

AtomBench: A Benchmark for Generative Atomic Structure Models using GPT, Diffusion, and Flow Architectures

Charles Rhys Campbell,[†] Aldo H. Romero,[†] and Kamal Choudhary^{*,‡,¶}

[†]*Department of Physics and Astronomy, West Virginia University, Morgantown, WV 26506, USA*

[‡]*Department of Materials Science and Engineering, Whiting School of Engineering, The Johns Hopkins University, Baltimore, MD 21218, USA*

[¶]*Department of Electrical and Computer Engineering, Whiting School of Engineering, Whiting School of Engineering, The Johns Hopkins University, Baltimore, MD 21218, USA*

E-mail: kchoudh2@jhu.edu

Abstract

Generative models have become significant assets in the exploration and identification of new materials, enabling the rapid proposal of candidate crystal structures that satisfy target properties. Despite the increasing adoption of diverse architectures, a rigorous comparative evaluation of their performance on materials datasets is lacking. In this work, we present a systematic benchmark of three representative generative models: AtomGPT (a transformer-based model), Crystal Diffusion Variational Autoencoder (CDVAE), and FlowMM (a Riemannian flow matching model). These models were trained to reconstruct crystal structures from subsets of two publicly available superconductivity datasets: JARVIS Supercon 3D and DS-A/B from the Alexandria database. Performance was assessed using the Kullback-Leibler (KL)

divergence between predicted and reference distributions of lattice parameters, as well as the mean absolute error (MAE) of individual lattice constants. For the computed KLD and MAE scores, CDVAE performs most favorably, followed by AtomGPT, and then FlowMM. All benchmarking code and model configurations will be made publicly available at https://github.com/atomgptlab/atombench_inverse.

Introduction

Electrons in crystalline solids can organize into remarkable collective states. High-temperature superconductivity is a prime example that lies at the forefront of condensed-matter physics and materials engineering. The conventional in-silico discovery pipeline relies heavily on density functional theory (DFT), which, though accurate in many cases, typically requires substantial computational effort per structure, scaling cubically with system size, thereby limiting throughput to only a few candidates at a time. DFT and its extensions, such as superconducting DFT (SCDFT), are reliable when superconductivity is driven by electron-phonon coupling, as in conventional superconductors.¹⁻³ However, this approach becomes problematic for unconventional superconductors such as cuprates and iron-based compounds-where strong electron-electron correlations play a major role and no consensus exists on a fully predictive ab initio theory.⁴⁻⁶ Moreover, standard DFT methods can underestimate critical temperatures by significant margins. For example, nearly 50% in some hydride systems-highlighting the need for correction schemes or more advanced formalisms.⁷⁻⁹ Still, DFT-based methods remain the most widely used and trusted approach for describing conventional (electron-phonon-mediated) superconductors, where the pairing mechanism is well understood and reliably captured within current first-principles frameworks. For this reason, DFT serves as the principal basis for generating the training data used in this study. In contrast, for unconventional superconductors-where strong electronic correlations dominate and no universally accepted ab initio theory exists. At this point, DFT is insufficient, and model training would require fundamentally different datasets and theoretical assumptions.

While theoretical efforts are focused on developing more accurate or more explicit models for high T_c superconductivity, the use of artificial intelligence methods can be a game changer, since they provide extrapolation techniques capable of deriving hitherto unknown information from the dataset.¹⁰ By training on well characterized DFT corpora with specified pseudopotentials, convergence settings, and other specified methodologies, generative models inherit the DFT configuration’s physical constraints, chemical validity assumptions, and systematic biases while sharply lowering the cost of candidate generation and property prediction. As a surrogate of the DFT-induced distribution, the model cannot, without further correction or transfer learning, achieve accuracy beyond that of its teacher. The overarching goal is not to replace DFT entirely but to use machine learning to propose high-quality candidates that are statistically likely to exhibit desired properties, which may subsequently be validated with more precise yet costly ab initio methods. Moreover, machine learning can also be used to quickly estimate properties at high-throughput when it would be too costly to do so by means of DFT.

With that said, high-quality DFT databases form the structural backbone for machine-learning-driven studies of conventional (electron-phonon-mediated) superconductors. For example, the JARVIS-DFT infrastructure contains over 90,000 materials with extensive computed properties-ranging from structural, electronic, mechanical, to phonon-related data-including a refined subset of about 1,058 superconductors characterized via electron-phonon coupling and the McMillan-Allen-Dynes formula to estimate T_c .^{11–15} Similarly, high-throughput screening within JARVIS-DFT has evaluated over 1,000 two-dimensional materials, yielding 34 dynamically stable superconductors with $T_c > 5\text{ K}$ ¹⁶ and high-pressure hydride superconductors.¹⁷ Beyond this, the Alexandria database curated by Marques and collaborators provides an even broader foundation-now including more than 4.4 million inorganic compounds, covering multiple dimensionalities (3D, 2D, 1D),¹⁸ with computed properties accessible under a permissive open license. This vast repository enables machine-learning-accelerated workflows that have already suggested promising hydride superconductors among more than

one million candidate compounds.^{19–21}

These repositories not only accelerate conventional high-throughput screening but also serve as the training ground for machine-learning approaches, which generally fall into two categories: forward design and inverse design. Forward design, also known as the direct or predictive problem, involves determining a material’s macroscopic properties based on a complete specification of its atomic crystallographic structure. In the context of crystalline materials, essential information comes from the Bravais lattice vectors and the atomic positions within the unit cell. Otherwise, the system can be defined by means of the Wyckoff positions, cell parameters, and the specification of the space group. In both cases, it is also necessary to specify the chemical identities of the constituent elements. Traditionally, the structure/property mapping is computed using first-principles methods, most notably DFT, which solves the many-body Schrödinger equation to predict observables such as formation energy, band gap, elastic moduli, and superconducting critical temperature.^{22,23} While DFT offers high accuracy and deep physical insight, it is computationally expensive—typically scaling as $\mathcal{O}(N^3)$ with system size—which limits its use in large-scale screening of the vast chemical compound space.

To overcome this limitation, machine learning has emerged as an efficient surrogate for DFT. Approaches such as graph neural networks (GNNs) and equivariant message-passing networks are trained on existing DFT databases to learn the complex, nonlinear relationship between crystal structures and properties. These forward models encode the atomic species and spatial arrangement into a structured representation—often referred to as a crystal graph—and approximate the function $f : (\mathbf{A}, \mathbf{X}, \mathbf{L}) \mapsto \mathbf{y}$, where $\mathbf{A}$ represents the atomic species, $\mathbf{X}$ the fractional coordinates, $\mathbf{L}$ the set of lattice vectors, and $\mathbf{y}$ the target property vector. Forward models are typically trained on large corpora of experimental or DFT-relaxed structures $\{(\mathbf{A}_i, \mathbf{X}_i, \mathbf{L}_i, \mathbf{y}_i)\}_{i=1}^N$. Once trained, such models can evaluate millions of hypothetical structures in minutes to hours, making them suitable for high-throughput virtual screening and accelerating the discovery pipeline by several orders of magnitude.

Inverse design poses a complementary and substantially more challenging problem: starting from a desired property vector $\mathbf{y}$, identify one or more crystal structures $\mathbf{M} = (\mathbf{A}, \mathbf{X}, \mathbf{L})$ that are likely to realize it. This inverse problem is inherently ill-posed; the structure-property mapping is many-to-one, nonlinear, and discontinuous, meaning that no unique or closed-form inverse exists. Recent advances in generative modeling offer a probabilistic path forward by learning the conditional distribution $p_\phi(\mathbf{A}, \mathbf{X}, \mathbf{L} \mid \mathbf{y})$, which can be sampled to generate chemically valid, symmetry-consistent crystals that are biased toward the target properties. Similar to forward models, these inverse models are trained on large corpora of experimental or DFT-relaxed structures $\{(\mathbf{A}_i, \mathbf{X}_i, \mathbf{L}_i, \mathbf{y}_i)\}_{i=1}^N$, optimizing likelihood or divergence-based objectives to align the learned generative distribution with the empirical one. Once trained, inverse design reduces to sampling from the learned conditional distribution p_ϕ , conditioned on the vector of properties $\mathbf{y}^*$, to generate crystal structures.

Notably, a diverse array of model architectures follows the inverse design paradigm from convolutional autoencoders to diffusion graph models.²⁴ Despite the proliferation of inverse design architectures, the field lacks standardized, reproducible benchmarks to assess accuracy. This ambiguity limits methodological progress and provides little guidance to practitioners seeking reliable tools for downstream discovery. To address this gap, we benchmark three inverse-design models on two DFT-relaxed datasets, measuring per-structure reconstruction errors and crystal distribution divergences to assess how accurately each model recovers ground-truth atomic arrangements from target superconducting critical temperatures. Particularly, we use (i) Generative pretrained transformers (GPTs) tokenize atomic species, lattice parameters, and fractional coordinates into sequences, modeling crystal generation as a language modeling task.^{25,26} Their self-attention mechanisms capture long-range chemical dependencies, and their flexible conditioning capabilities make them well-suited for property-targeted generation. (ii) Diffusion variational autoencoders (VAEs) adopt a denoising approach, gradually converting random noise into plausible crystal structures.^{27,28} (iii) Riemannian flow matching models define a generative process as an ordinary differential

equation (ODE) on a geometric manifold, transforming samples from a known prior into the space of valid crystal structures.²⁹

The following section outlines the study design, datasets, evaluation metrics, and the three inverse-model architectures. Subsequent sections present the results obtained for each model, followed by a discussion of the findings and concluding remarks.

Methods

In this work, we performed a systematic comparison of three inverse design models: AtomGPT,³⁰ a large language model; Crystal Diffusion Variational Autoencoder (CDVAE),³¹ a diffusion variational autoencoder model; and FlowMM,²⁹ a Riemannian flow-matching network. Two datasets were used in this study for model training and testing (labeled JARVIS Supercon-3D³² and Alexandria DS-A/B³³), and they are both comprised of DFT calculations. Each dataset comprises input–output pairs, where the input is a parameterized graph representation of the material and the output is its superconducting transition temperature computed from DFT. A separate instance of each model was trained on each dataset, resulting in six trained models in total. For each of the six model instances, 10% of the training data was withheld from training, and each model was tasked with reproducing the unseen 10% of its corresponding dataset. We then performed a statistical assessment of each model’s ability to reconstruct the held-out data, and the resulting performance metrics were uploaded to the JARVIS-Leaderboard,³⁴ an open-source benchmarking platform for materials AI models. https://atomgptlab.github.io/jarvis_leaderboard/Special/AtomGenBench/.

For the statistical assessment, we measure the structural deviations both per-structure and over the whole distribution using three metrics. The first metric quantifies the mean absolute deviation of individual reconstructed crystal lattices from their ground-truth counterparts. The second metric assesses how closely the overall set of reconstructed crystal lattices matches the statistical distribution of lattices found in the true dataset. The third

metric measures how closely the reconstructed atomic coordinates align with those of the ground-truth structures. Before detailing the statistical metrics, we must define how a crystal structure is parametrized. A crystal is described by six lattice parameters $\{a, b, c, \alpha, \beta, \gamma\}$, which define the shape of the unit cell parallelepiped through the metric tensor G , and by a set of atomic coordinates $\mathbf{r}_i$ expressed in reduced (fractional) units within that cell. Together, these parameters fully specify the periodic arrangement of atoms used as input for both training and reconstruction analysis. Prior to computing statistical metrics, all reconstructed structures are transformed into their Niggli-reduced cells.³⁵ This canonical change of basis removes degeneracies due to lattice-vector permutations and prevents artifacts introduced by differing conventions in lattice-parameter labeling among models.

For the first statistical metric, we have employed the Mean Absolute Error (MAE) to quantify the average deviation between predicted and ground-truth lattice parameters. Specifically, the MAE is computed for each of the six lattice parameters $\{a, b, c, \alpha, \beta, \gamma\}$ by averaging the absolute differences between their predicted values $\hat{y}_i$ and reference values y_i , $\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$. Consequently, for each trained model instance, six Mean Absolute Error (MAE) values are produced. These values serve as a direct indicator of the model’s precision in replicating the crystal geometry. An accurate reproduction of the lattice is essential for determining the correct space group and the crystal family. The MAE is a standard and interpretable metric widely used to assess the average magnitude of prediction errors.

For the second statistical metric, we use Kullback-Leibler Divergence (KLD)³⁶ to measure the divergence between the histogrammed distribution of a single ground-truth crystal lattice parameter and the histogrammed distribution of a single predicted crystal lattice parameter for all six crystal lattice parameters $\{a, b, c, \alpha, \beta, \gamma\}$. Similarly to MAE, this means that there are six KLD computations for each trained model instance under consideration. The KLD between a true distribution P and a predicted distribution Q quantifies the expected log-ratio of probabilities of events $p(x)$ and $q(x)$ under distributions P and Q respectively:

$$D_{\text{KL}}(P\|Q) = \sum_x p(x) \log \frac{p(x)}{q(x)} \quad (1)$$

By construction, KLD is greater than zero if $P \neq Q$, and KLD approaches zero as Q approaches P . This means that a lower KLD corresponds to greater similarity between the predicted and ground-truth lattice parameter histograms.

For the third statistical metric, we employ the normalized Root Mean Square Error (RMSE) to evaluate how closely the reconstructed atomic coordinates align with those of the ground-truth structures. The RMSE between predicted and reference atomic positions $\hat{\mathbf{r}}_i$ and $\mathbf{r}_i$ is computed as $\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N \|\mathbf{r}_i - \hat{\mathbf{r}}_i\|^2}$ and then averaged over all matched materials in the held-out test subset. Unlike the earlier metrics, there is only one average RMSE value produced for each trained model instance. Because interatomic distances vary substantially across materials, the RMSE is normalized by an appropriate structural length scale to ensure comparability across datasets. Lower normalized RMSE values indicate that a model more faithfully reproduces the spatial arrangement of atoms within each crystal, providing a direct measure of local structural accuracy that complements the global lattice and distributional metrics.

Results & Discussion

A schematic of the current work design is summarized in Figure 1. First, we benchmark on the JARVIS Supercon-3D superconductor dataset.³² This was the first full atomic structure information database for superconductors. The authors start from 55,723 JARVIS-DFT structures and pre-screen using the Debye temperature θ_D (derived from elastic tensors) and the electronic DOS at the Fermi level, retaining materials with $\theta_D > 300$ K (5,618 remain) and $N(0) > 1$ states eV⁻¹ per valence electron (1,736 remain). To keep DFPT tractable, they then restrict to primitive cells with ≤ 5 atoms, yielding 1,058 candidates. For these, electron-phonon coupling (EPC) is computed via DFPT in Quantum ESPRESSO with

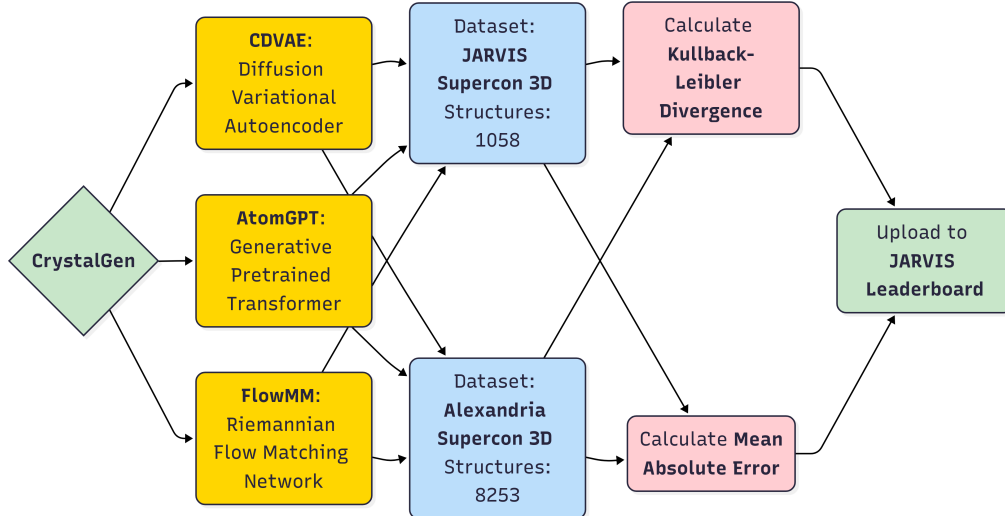


Figure 1: Diagram showing the inverse model benchmarking study design. We compare three generative AI inverse models-AtomGPT,³⁰ a language model; CDVAE,³¹ a diffusion variational autoencoder; and FlowMM,²⁹ a flow-matching network-each separately trained on two superconductivity DFT datasets (JARVIS Supercon-3D³² and Alexandria DS-A/B³³) for a total of six benchmarks. Ten percent of each dataset is held out to test reconstruction performance, which we statistically quantify before submitting these results to the JARVIS-Leaderboard.

GBRV pseudopotentials³⁷ and the PBEsol exchange-correlation functional;³⁸ T_c is estimated using the McMillan-Allen-Dynes equation with $\mu^* = 0.09$, where μ^* is the Morel-Anderson effective Coulomb pseudopotential.³⁹ Dynamical stability is assessed from the DFPT phonon spectra: a material is labeled “stable” only if no imaginary (negative) phonon frequencies appear at any sampled $\mathbf{q}$ -point across the Brillouin zone (i.e., all mode eigenfrequencies satisfy $\omega_{qj}^2 > 0$ within numerical tolerances). This yields 626 dynamically stable structures, of which 105 have $T_c \geq 5$ K. The EPC convergence strategy is deliberately lightweight: reuse the JARVIS-converged k -point meshes, employ at least a $2 \times 2 \times 2$ q -mesh, and apply Gaussian broadening ≈ 0.05 Ry to stabilize λ at modest cost. In our benchmarks, we use the full 1,058-structure set to evaluate property-conditioned reconstruction; if prioritizing dynamical stability, one could restrict to the 626-structure stable subset.

Subsequently, the Alexandria DS-A/B dataset³³ was developed utilizing a high-throughput methodology enhanced by machine learning techniques. This comprehensive dataset encom-

passes a total of 8,253 well-converged EPC entries, which include various compounds such as nitrides, hydrides, and intermetallics. The source structures are screened from the Alexandria database for metallic, non-magnetic compounds on or near the convex hull (typically $E_{\text{hull}} < 50$ meV/atom, with a secondary sweep allowing $50 \leq E_{\text{hull}} < 100$ meV/atom for small cells), excluding semiconductors, insulators, semimetals, and very low density of states entries. A Debye-temperature filter $T_D > 300$ K (estimated via an ALIGNN model²³) is applied, and structural complexity is limited to ≤ 8 atoms per primitive cell with space-group number ≥ 100 (favoring tetragonal and cubic lattices). All calculations use PBEsol³⁸ in Quantum ESPRESSO with PseudoDojo pseudopotentials, tight stopping criteria on energies/forces/stresses (10^{-8} a.u., 10^{-6} a.u., 5×10^{-2} kbar), a double-grid electron-phonon coupling (EPC) strategy, and Methfessel-Paxton smearing ≈ 0.05 Ry. Dynamic stability is assessed from DFPT phonons; a practical tolerance allows for up to three small-magnitude imaginary modes at Γ ($\lesssim 35$ cm⁻¹) to account for numerical artifacts, with false positives removed at higher accuracy. Superconducting labels report the dimensionless EPC strength λ , the logarithmic-average phonon frequency ω_{log} , and T_c from the McMillan-Allen-Dynes formula using $\mu^* = 0.10$ (where μ^* is the Morel-Anderson effective Coulomb pseudopotential). In this work, we concatenate the original DS-A (training) and DS-B (validation) splits into a single DS-A/B dataset for model training and evaluation.

Subsequently, a crucial step is to explicitly define the relationship between two sets of crystals: (1) the 10% subset of fully specified test set crystals and (2) the corresponding collection of crystals whose geometries are predicted by the inverse models. Each crystal in the test set is fully specified, meaning that its chemical species, lattice structure parameters, atomic coordinates, and superconducting temperature are known. For each test set crystal, there is a corresponding partially-specified crystal that shares chemical species and superconducting temperature information with the ground-truth crystal. By construction, there is a one-to-one correspondence between the set of ground-truth crystals and the set of partially-specified crystals, established via dataset indexing. However, the partially-specified crystal

does not share lattice parameters or atomic coordinates with its corresponding ground-truth, nor does it contain lattice parameters or atomic coordinates at all. It is the purpose of the inverse model to map the partially-specified crystal’s existing stoichiometry and superconducting temperature data to a new set of crystallographic parameters as a function of the inverse model’s learned parameters. We denote the set of partially-specified crystals that have undergone this mapping to be the set of predicted crystals, or, interchangeably, the set of reconstructed crystals. In the CDVAE setting, test structures $\mathbf{M}_i$ are encoded into latent representations z_i and decoded back into test structures $\hat{\mathbf{M}}_i$. For consistency with the above protocol, the decoded test structures $\hat{\mathbf{M}}_i$ are treated as reconstructions of $\mathbf{M}_i$ and are evaluated with the same metrics; model-specific details appear in the CDVAE section.

Next, we discuss the three machine learning architectures in greater detail, including dataset encoding, training procedures, and the inference process. First, AtomGPT is a generative, pretrained transformer (GPT) adapted to predict crystal structures and target properties via text generation. This framing not only enables the model to predict lattice parameters and atomic coordinates by generating sequences of text but also makes the interface more intuitive for researchers, since structures can be queried and generated through natural language rather than specialized code. Since this study is focused on testing AtomGPT’s inverse design accuracy, its ability to predict target properties as a function of lattice structure will not be discussed; we will only further discuss its ability to predict lattice structures as a function of a desired property.

At its core, AtomGPT utilizes a pretrained language model, and for this study, the Mistral-7b-BNB-4bit GPT was used for its low computational cost and strong performance on benchmarks.⁴⁰ Out of the box, this pretrained language model is a generalist, and it has no particular expertise in crystal and materials problems relative to other problem domains. However, there is strong evidence that resuming model training (finetuning) on crystal structures strongly improves the performance of language models for predicting material properties and lattice structures.^{30,41} To perform finetuning, AtomGPT utilizes the Hug-

ging Face SFTTrainer to intake a dataset of crystal-property pairs and update the model weights accordingly. The crystal-property pairs used for finetuning follow a common textual schema, and the following example illustrates how crystal-property pairs are represented as text during training.

Input: The crystal’s chemical formula is Nb₃Sn, and the superconducting transition temperature is 18.3 K. Generate atomic structure description with lattice lengths, angles, coordinates and atom types.

Output: 5.32 5.32 5.32
 90 90 90
 Sn 0.000 0.000 0.000
 Nb 0.000 0.500 0.500
 Nb 0.500 0.000 0.500
 Nb 0.500 0.500 0.000

During finetuning, the model generates a predicted crystal structure for each input prompt. The cross-entropy loss is then computed token-by-token against the textual encoding of the corresponding ground-truth structure, and model weights are updated via gradient descent to minimize this loss. After finetuning, inference is performed using the finetuned checkpoint loaded via the Hugging Face Transformers library. Hyperparameters, the forked repository, and implementation details are found in the appendix.

The next model, CDVAE, is an inverse model for materials design that generates periodic and physically plausible materials using a diffusion variational autoencoder architecture. It has three main functionalities: reconstruction, generation, and property optimization. For this study, only the reconstruction task was used; therefore, we will constrain our discussion of CDVAE to cover only reconstruction and concepts that are relevant to it. Under the

hood, CDVAE reconstructs materials using a three-step process. First, an SE(3)-equivariant periodic graph neural network encoder $PGNN_{ENC}(\mathbf{M})$ maps a crystal structure $\mathbf{M}$ to a latent representation $\mathbf{z}$. Importantly, the crystal structure $\mathbf{M} = (\mathbf{A}, \mathbf{X}, \mathbf{L})$ is fully described by three lists, $\mathbf{A}$, $\mathbf{X}$, and $\mathbf{L}$, which contain the crystal’s atom types, atomic coordinates, and Bravais lattice vectors respectively. Second, three distinct multilayer perceptrons (MLPs) map $\mathbf{z}$ to a set of three aggregated properties: the atomic composition $\mathbf{A}$, the Bravais lattice vectors $\mathbf{L}$, and the number of atoms N . After these aggregated properties are predicted, a provisional structure is instantiated by assigning atom types according to the predicted composition $\mathbf{A}$, placing them at uniformly sampled fractional coordinates within the unit cell defined by $\mathbf{L}$, and perturbing these positions with Gaussian noise to obtain the noisy material $\tilde{\mathbf{M}}$. Third, a conditional score-matching decoder $PGNN_{DEC}(\mathbf{M} | \mathbf{z})$ parameterized by an SE(3)-equivariant periodic graph neural network denoises $\tilde{\mathbf{M}}$ via annealed Langevin dynamics to produce a reconstruction of the original crystal $\mathbf{M}$. During training, the encoder, decoder, and aggregate property heads are jointly optimized with a master loss function that is a linear combination of the individual loss functions from each neural network used in the CDVAE architecture: $\mathcal{L} = \mathcal{L}_{AGG} + \mathcal{L}_{DEC} + \mathcal{L}_{KL}$. In this expression, $\mathcal{L}_{AGG}$ represents the atom type classification and lattice regression loss, $\mathcal{L}_{DEC}$ represents the denoising score-matching loss, and $\mathcal{L}_{KL}$ represents the Kullback-Leibler divergence regularization loss for the encoder, which is characteristic of many VAE architectures. Hyperparameters, the forked repository, and implementation details are found in the appendix.

Finally, FlowMM is an inverse model for materials design that utilizes a Riemannian flow matching architecture to predict stable crystal structures with either known or novel compositions. FlowMM is built to perform two tasks: crystal structure prediction (CSP) and *de novo* generation (DNG). For this study, only the CSP task was used, so we constrain our discussion of FlowMM to cover only CSP and concepts relevant to it. CSP is the process by which FlowMM predicts the atomic fractional coordinates and Bravais lattice vectors of a crystal with only its composition specified, and FlowMM represents crystals by their

position on a product manifold $\mathcal{C} := \mathcal{A} \times \mathcal{F} \times \mathcal{L}$. The submanifold $\mathcal{F}$ is a collection of $n \times 3$ flat tori representing the periodic space of atomic fractional coordinates, the submanifold $\mathcal{L}$ is the space of lattice parameters $\{a, b, c\} \in \mathbb{R}^{+3}$ and $\{\alpha, \beta, \gamma\} \in [60, 120]^3$, subject to the Niggli reduction.³⁵ Because of domain boundaries in $\{\alpha, \beta, \gamma\}$, FlowMM represents lattice parameters in an unconstrained coordinate system via an invertible transformation ϕ ; training and inference take place in this flat space, and $\{\alpha, \beta, \gamma\}$ are recovered by applying the inverse transformation ϕ^{-1} . The submanifold $\mathcal{A}$ represents compositions, which in CSP are fixed h -dimensional one-hot vectors. Since composition is specified during both training and inference, the learned vector field has no active components along $\mathcal{A}$, effectively reducing the manifold to $\mathcal{C}' := \{\mathbf{A}\} \times \mathcal{F} \times \mathcal{L}$, where $\mathbf{A} \in \mathcal{A}$. Because $\mathcal{F}$ and $\mathcal{L}$ are flat, closed-form geodesics exist and take the form of straight line segments, however, geodesics in the $\mathcal{F}$ submanifold wrap around due to the toroidal nature of $\mathcal{F}$.

On this space, a time-dependent flow ψ_t is defined as the solution of the differential equation

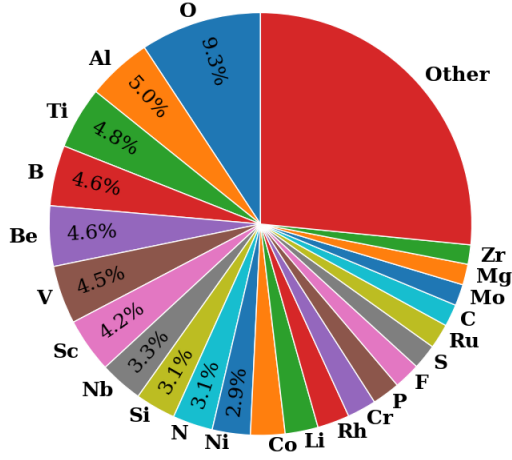
$$\frac{d}{dt}\psi_t(x) = u_t(\psi_t(x)), \quad \psi_0(x) = x,$$

which pushes an initial density p_0 along a probability path p_t to a target distribution $p_1 = q$. The initial probability density on the manifold is comprised of the uniform distribution on $\mathcal{F}$, the LogNormal distribution for $\{a, b, c\}$, and the uniform distribution for $\{\alpha, \beta, \gamma\}$ pushed through ϕ . In $\mathcal{F}$, the conditional targets use toroidal logarithmic displacements with the mean tangent translation across atoms removed, yielding a translation-invariant marginal path.²⁹ The target distribution q is the empirical measure supported on the training set, $q = \frac{1}{N} \sum_{i=1}^N \delta_{x_i}$, from which we sample x_1 during training. In practice, training proceeds by sampling an initial point $x_0 \sim p_0$, a time $t \sim \text{Uniform}(0, 1)$, and a training example $x_1 \sim q$. The conditional flow matching construction guaranties the existence of a velocity field $u_t(x|x_1)$ that connects x_0 to x_1 along a straight path (because $\mathcal{C}'$ is Euclidean).⁴² A permutation-equivariant, translation-aware GNN then predicts a velocity vector $v_t^\theta(x)$ in the tangent space at the interpolated location x_t . The loss is the squared Riemannian

distance between $v_t^\theta(x_t)$ and the target velocity $u_t(x_t|x_1)$, averaged over samples. Minimizing this objective aligns the learned velocity field with the analytic conditional vector fields, ensuring that the trained model transports the base distribution p_0 toward the empirical data distribution q . Permutation invariance follows from relabel-equivariant message passing, and rotation invariance follows from using Niggli-reduced lattice parameters in $\mathcal{L}$, so the induced density is S_n - and $SO(3)$ -invariant by construction.²⁹

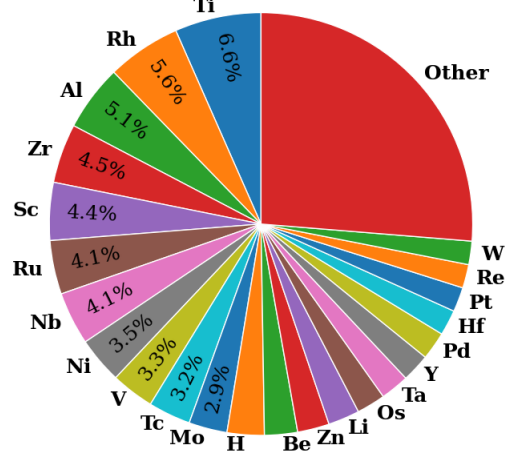
At inference time, CSP reduces to integrating the learned flow forward in time: given a composition $\mathbf{A}$, a point is sampled from the base distribution p_0 on $\mathcal{F} \times \mathcal{L}$ and transported to $t = 1$ under the learned flow. The result is a predicted set of lattice parameters and atomic fractional coordinates consistent with the specified composition. Hyperparameters, the repository fork, and implementation details are found in the appendix.

Element Proportions in the JARVIS Supercon-3D Dataset



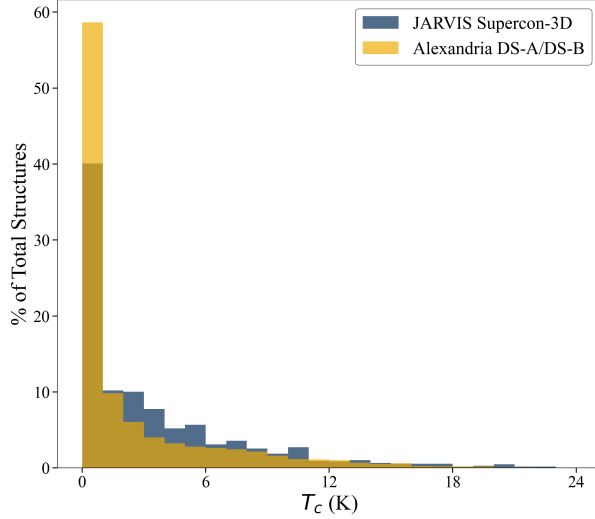
(a)

Element Proportions in the Alexandria DS-A/B Dataset



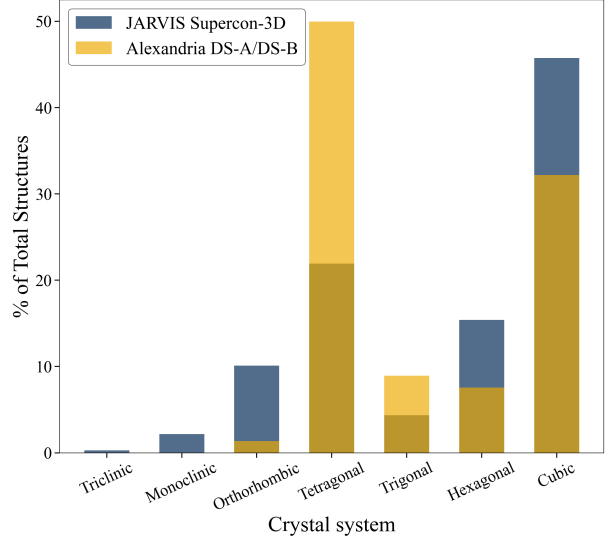
(b)

Distribution of T_c
JARVIS vs Alexandria



(c)

Crystal Systems
JARVIS vs Alexandria



(d)

Figure 2: Statistical comparison of the JARVIS Supercon-3D and Alexandria DS-A/B superconductivity datasets. (a,b) Pie charts show elemental compositions for the 23 most represented elements. In JARVIS, oxygen, aluminum, and titanium are most common (9.3%, 5.0%, and 4.8%), while in Alexandria, titanium, rhodium, and aluminum dominate (6.6%, 5.0%, and 5.1%). (c) The overlain histogram compares superconducting critical temperature (T_c) distributions, both decaying exponentially with T_c , though Supercon-3D contains a larger fraction of high- T_c materials. (d) The overlain bar chart shows crystal system distributions: DS-A/B is dominated by cubic and tetragonal phases (>80%), whereas Supercon-3D exhibits greater diversity, with ~45% cubic and a broader spread across all seven systems.

Alexandria DS-A/B Reconstruction KLD

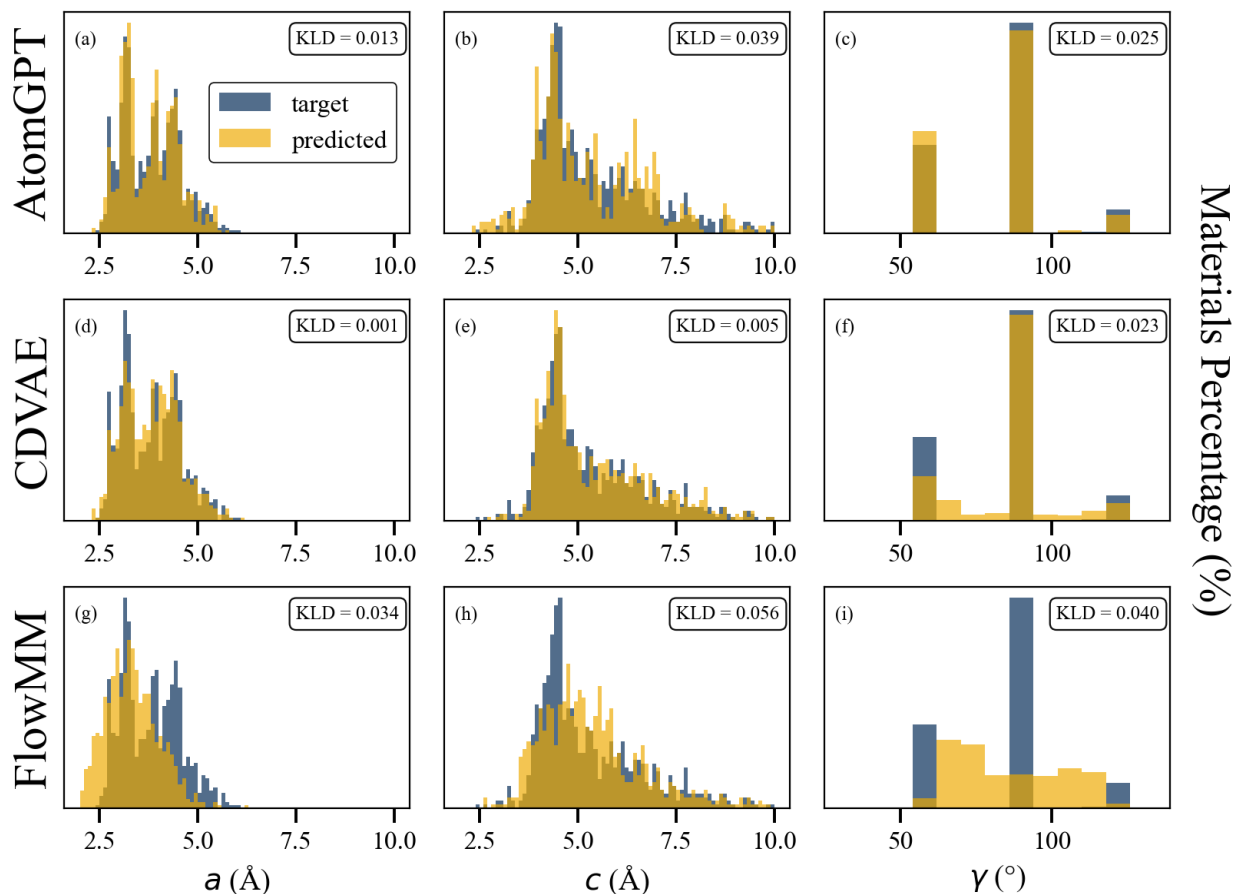


Figure 3: Reconstruction performance of AtomGPT, CDVAE, and FlowMM on the Alexandria DS-A/B test set (825 structures) for three representative Niggli-reduced lattice parameters, a , c , and γ . The blue distributions are the target distributions directly obtained from the dataset, and the gold distributions are the predictions made by the models with the goal of matching the target distributions. CDVAE appears to match the target distribution most closely, followed by AtomGPT, and with FlowMM’s predictions matching the target the least closely.

JARVIS Supercon-3D Reconstruction KLD

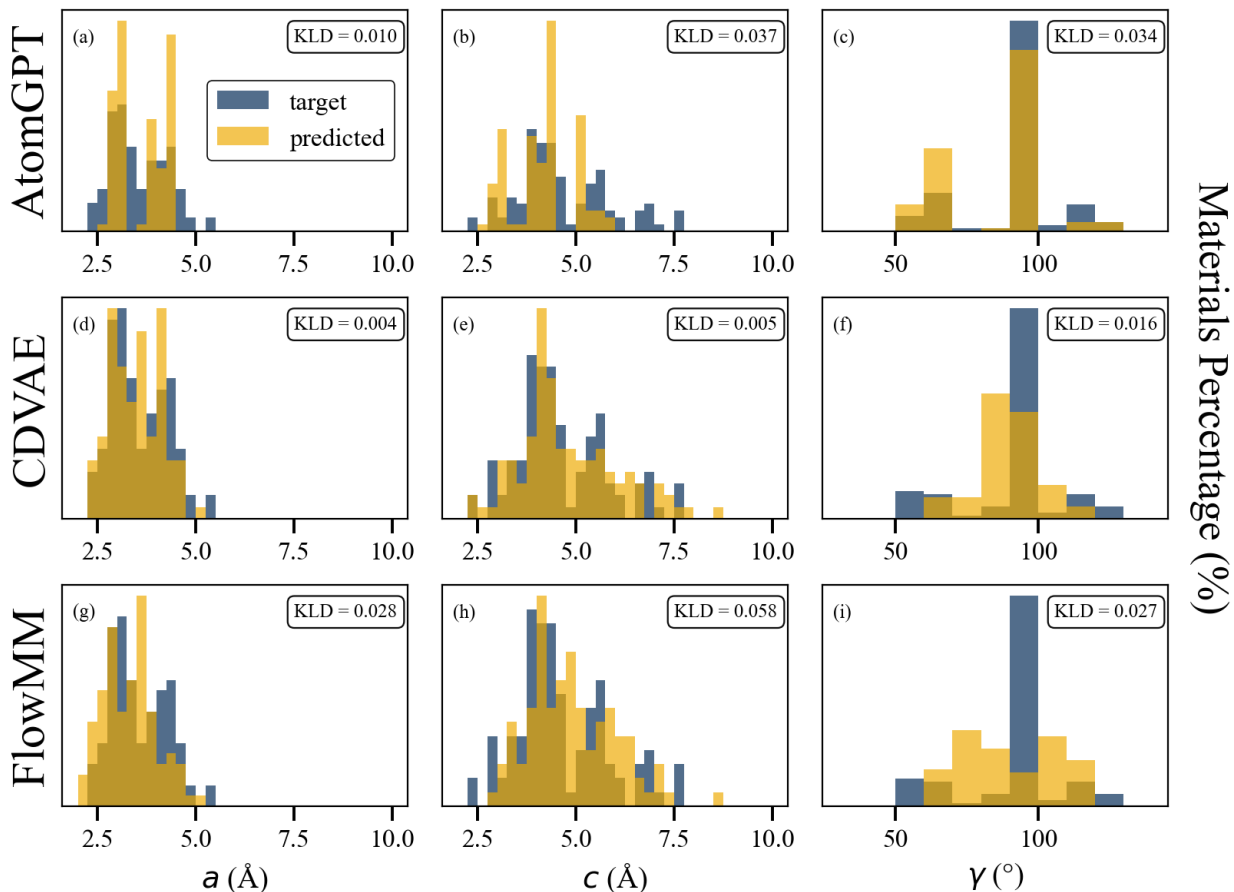


Figure 4: Reconstruction performance of AtomGPT, CDVAE, and FlowMM on the JARVIS Supercon-3D test set (105 structures) for three representative Niggli-reduced lattice parameters, a , c , and γ . The blue distributions are the target distributions directly obtained from the dataset, and the gold distributions are the predictions made by the models with the goal of matching the target distributions. On the average, CDVAE has the lowest KLD, followed by AtomGPT and then FlowMM. It is ambiguous which model’s predictions visually appear to follow the target distributions most closely, reasonably due to the lower volume of test structures.

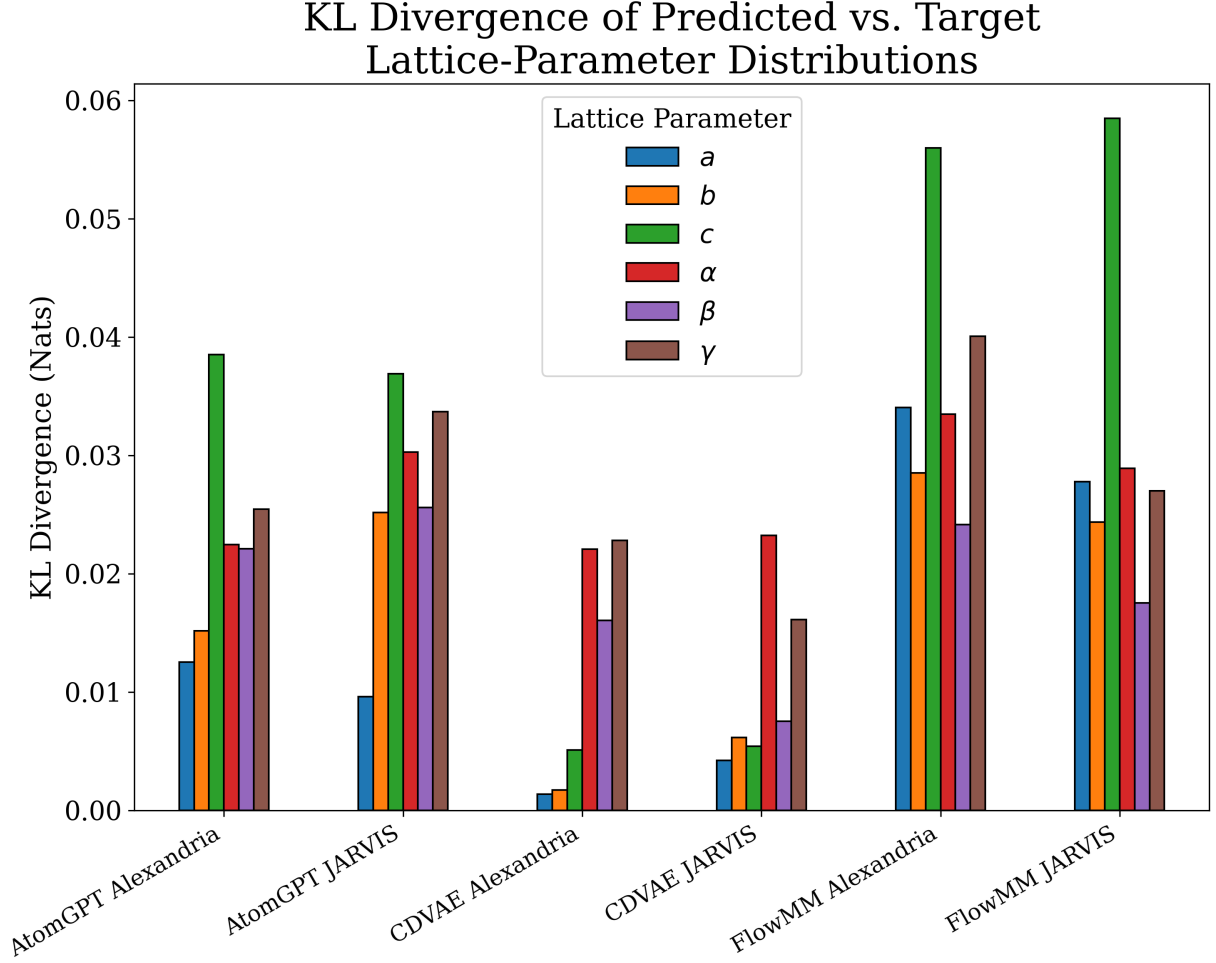


Figure 5: Kullback-Leibler Divergence in units of nats between the predicted and target distributions for all six Niggli-reduced lattice parameters (a , b , c , α , β , γ) for a total of six experiments using three models and two datasets. CDVAE has the most favorable KLD scores for both datasets, followed by AtomGPT and then FlowMM.

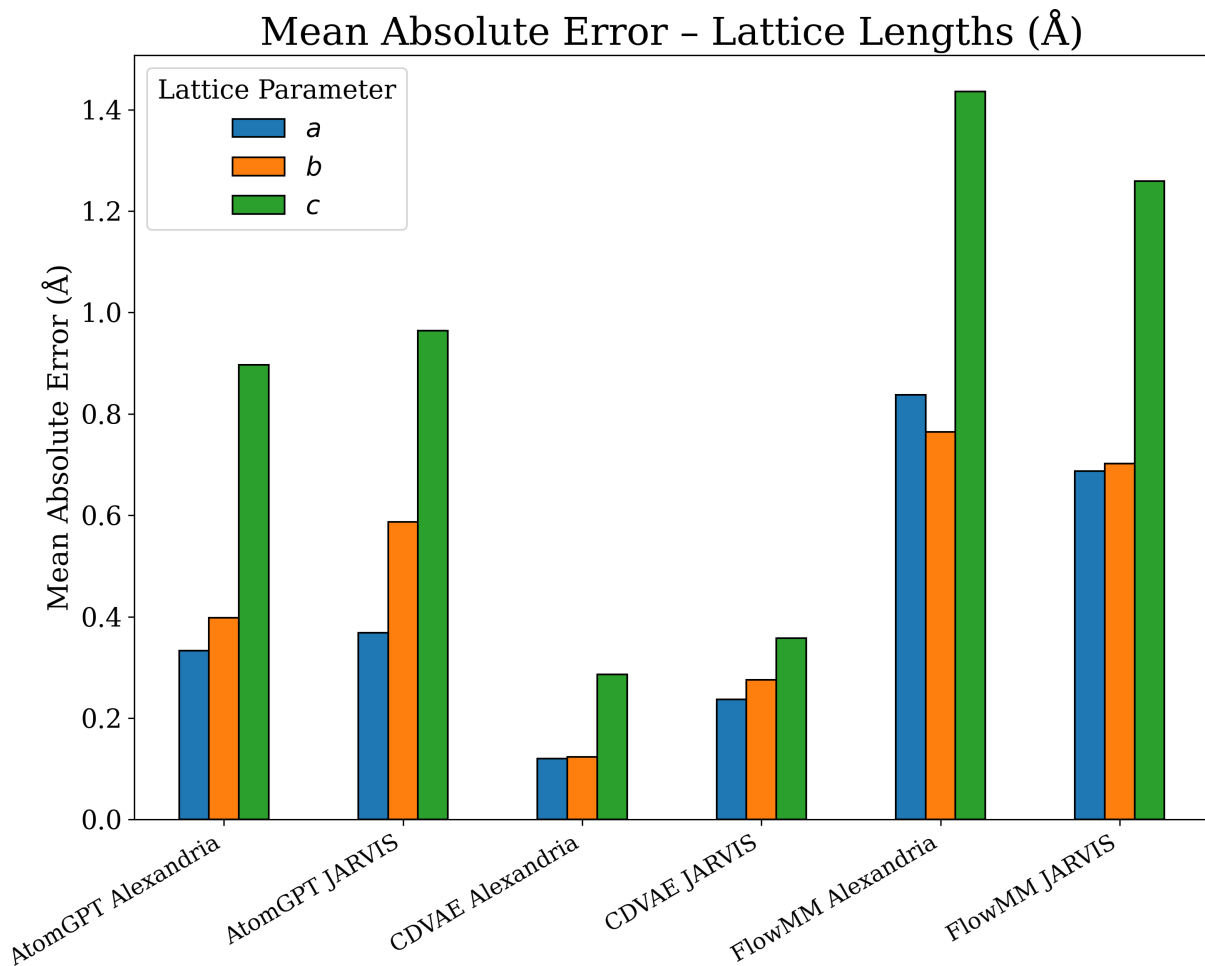


Figure 6: Mean absolute error in units of angstroms between the predicted and target distributions for the (*a*, *b*, *c*) Niggli-reduced lattice lengths for a total of six experiments using three models and two datasets. CDVAE has the most favorable lattice length MAE scores for both datasets, followed by AtomGPT and then FlowMM.

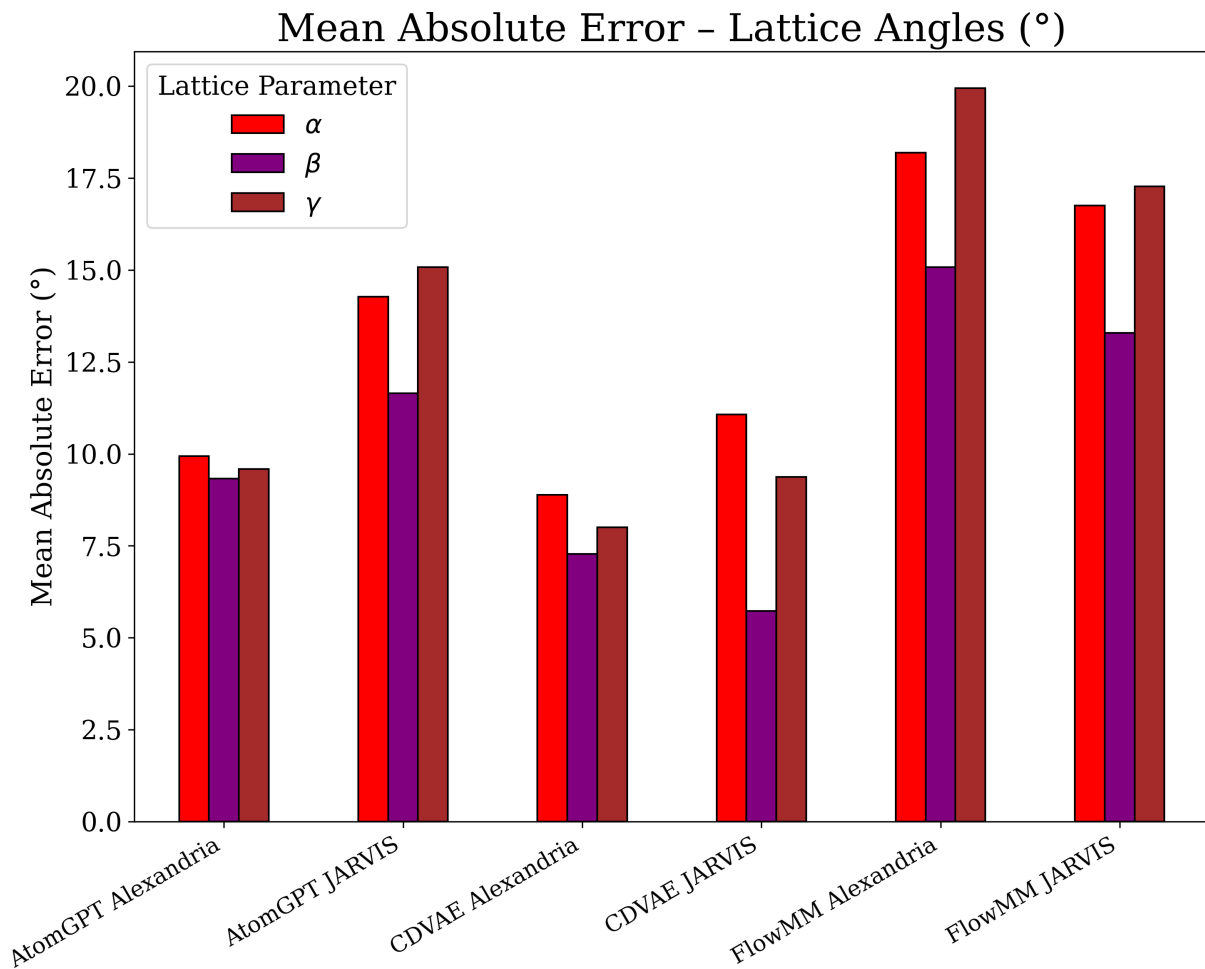


Figure 7: Mean absolute error in units of degrees between the predicted and target distributions for the (α, β, γ) Niggli-reduced lattice angles for a total of six experiments using three models and two datasets. CDVAE has the most favorable lattice angle MAE scores for both datasets, followed by AtomGPT and then FlowMM.

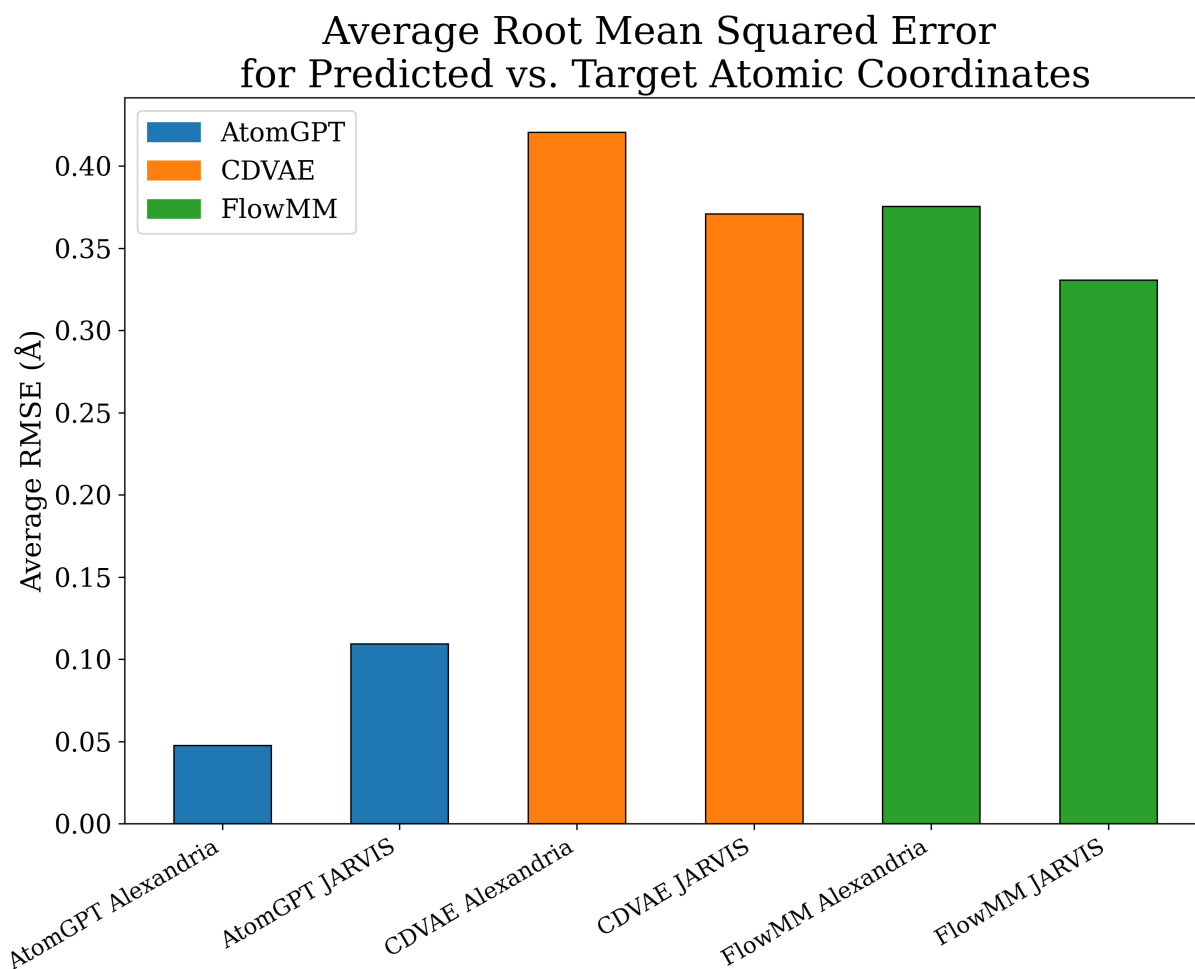


Figure 8: Average root mean squared error in units of Angstroms between the predicted and target atomic coordinates for a total of six experiments using three models and two datasets. AtomGPT has the most favorable average RMSE scores for both datasets, followed by FlowMM and then CDVAE.

Across the Alexandria DS-A/B benchmarks, CDVAE achieved the lowest reconstruction error for both lattice KLD and MAE, while AtomGPT achieved the lowest error for atomic coordinate RMSE. In terms of lattice reconstruction, AtomGPT performed intermediately between CDVAE and FlowMM, with FlowMM exhibiting the highest error. Conversely, for atomic coordinate reconstruction, FlowMM performed between AtomGPT and CDVAE, with CDVAE showing the highest error in this category. On the JARVIS Supercon-3D benchmark, AtomGPT and FlowMM produced comparable KLD values, but AtomGPT attained a lower MAE for both lattice lengths and angular parameters. The three models differ in the amount of prior information they receive during reconstruction, and these input differences help explain the performance gap. CDVAE denoises a latent embedding, expressed as the mapping $\mathbf{z} = f^\theta(\mathbf{A}, \mathbf{X}, \mathbf{L})$, that is already rich in structural information; AtomGPT predicts $(\mathbf{X}, \mathbf{L})$ from scratch given the $\mathbf{A}$ and T_c ; FlowMM must infer solely from the $\mathbf{A}$ alone. For generation (not reconstruction), FlowMM can instead compute the flow starting from a large-language-model-suggested position on the manifold given a composition (called FlowLLM). The LLM component is structured similarly to AtomGPT, but it does not condition on target property information, though such conditioning could be added straightforwardly via the prompt schema. From an information-theoretic view, every crystal structure is specified by a finite amount of information, and these generative models are high-dimensional conditional probability distributions from which crystal reconstructions are sampled given some quantity of information about the target crystals. Since the models are supplied with different amounts of information before performing reconstruction, the total information recovered by the models is inversely proportional to the amount of information they start with. The expected surprisal of the reconstructions obey

$$H(\mathbf{M} \mid f^\theta(\mathbf{A}, \mathbf{X}, \mathbf{L})) < H(\mathbf{M} \mid \mathbf{A}, T_c) < H(\mathbf{M} \mid \mathbf{A}), \quad (2)$$

where H is the conditional entropy associated with each model, f^θ is the CDVAE encoder

acting to produce a latent $\mathbf{z}$, and $\mathbf{M}$ is the target crystal structure. This aligns with the observed reconstruction error for lattice parameters, but not for fractional coordinates. The deviation suggests that the T_c conditioning unique to AtomGPT may strongly influence atomic coordinate predictions, but further analysis will be required to test this hypothesis.

Future work should supply each model with equivalent information prior to reconstruction, enabling fine-grained architectural comparisons without the confound of unequal information regarding the target crystal. Following the inverse-design paradigm outlined in the introduction, we focus on mapping a specified composition $\mathbf{A}$ and superconducting transition temperature T_c to atomic coordinates $\mathbf{X}$ and lattice vectors $\mathbf{L}$. AtomGPT already fits this setup, but FlowMM and CDVAE would need modifications. In the context of FlowMM, one potential approach involves the introduction of a submanifold characterized by a scalar property, denoted as $\mathcal{T}$. This technique is aimed at ensuring that the learned flow accommodates conditions based on both composition and the temperature denoted as T_c . The reconstruction process would continue as it is currently implemented, but it would now incorporate this additional conditioning when computing flows. For CDVAE, one could fold T_c into the latent $\mathbf{z}$, add a head that recovers T_c from $\mathbf{z}$, and make the decoder explicitly conditional on the recovered T_c during denoising. This may be more robust than CDVAE’s current property-optimization approach (using an external predictor with latent-space gradient ascent) because conditioning on T_c is propagated throughout the architecture rather than confined to a single component.

Conclusion

In this work, we conducted a systematic benchmark of three inverse materials design models, AtomGPT, CDVAE, and FlowMM, across two superconducting datasets, JARVIS Supercon-3D and Alexandria DS-A/B. The objective was to establish a fair and quantitative comparison of model architectures under controlled conditions, addressing the current absence of

standardized performance benchmarks in data-driven inverse materials design. We find that CDVAE demonstrates superior accuracy in reconstructing lattice parameters, whereas AtomGPT excels in reproducing atomic coordinates, with FlowMM generally performing lower across both categories. We find that, for the reconstruction task, each model operates with differing amounts of information regarding the target crystal, and that reconstruction accuracy may be correlated with the amount of information provided. Future work will focus on comparing model architectures under experimental conditions in which each receives an equivalent amount of information about the target crystal. Establishing such parity will enable a rigorous evaluation of the intrinsic inductive biases that underlie each model’s design.

Data availability

The datasets used in this work are available at <https://doi.org/10.6084/m9.figshare.6815699> and <https://doi.org/10.6084/m9.figshare.27174897> . The code used in this study will be made available at https://github.com/atomgptlab/atombench_inverse .

Acknowledgements

We thank the Pittsburgh Supercomputer Center (Bridges2) and San Diego Supercomputer Center (Expanse) through allocation DMR140031 from the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program, which is supported by National Science Foundation grants #2138259, #2138286, #2138307, #2137603, and #2138296. We also recognize the computational resources provided by the WVU Research Computing Dolly Sods HPC cluster, which is funded in part by NSF OAC-2117575. Support for the West Virginia group came from West Virginia High Education Policy Commission under the call Research Challenge Grand Program 2022, Award RCG 23-007.

References

- (1) Giustino, F. Electron-phonon interactions from first principles. *Reviews of Modern Physics* **2017**, *89*, 015003.
- (2) Oliveira, L. N. d.; Gross, E.; Kohn, W. Density-functional theory for superconductors. *Physical review letters* **1988**, *60*, 2430.
- (3) Lüders, M.; Marques, M.; Lathiotakis, N.; Floris, A.; Profeta, G.; Fast, L.; Continenza, A.; Massidda, S.; Gross, E. Ab initio theory of superconductivity. I. Density functional formalism and approximate functionals. *Physical Review B—Condensed Matter and Materials Physics* **2005**, *72*, 024545.
- (4) Furness, J. W.; Zhang, Y.; Lane, C.; Buda, I. G.; Barbiellini, B.; Markiewicz, R. S.; Bansil, A.; Sun, J. An accurate first-principles treatment of doping-dependent electronic structure of high-temperature cuprate superconductors. *Communications Physics* **2018**, *1*, 11.
- (5) Pokharel, K.; Lane, C.; Furness, J. W.; Zhang, R.; Ning, J.; Barbiellini, B.; Markiewicz, R. S.; Zhang, Y.; Bansil, A.; Sun, J. Sensitivity of the electronic and magnetic structures of cuprate superconductors to density functional approximations. *npj Computational Materials* **2022**, *8*, 31.
- (6) Kent, P.; Saha-Dasgupta, T.; Jepsen, O.; Andersen, O. K.; Macridin, A.; Maier, T. A.; Jarrell, M.; Schulthess, T. C. Combined density functional and dynamical cluster quantum Monte Carlo calculations of the three-band Hubbard model for hole-doped cuprate superconductors. *Physical Review B—Condensed Matter and Materials Physics* **2008**, *78*, 035132.
- (7) Chen, S.; Wei, Y.; Monserrat, B.; Tomczak, J. M.; Poncé, S. Impact of electronic correlations on the superconductivity of high-pressure CeH₉. *arXiv preprint arXiv:2507.12506* **2025**,

- (8) Held, K.; Andersen, O.; Feldbacher, M.; Yamasaki, A.; Yang, Y. Bandstructure meets many-body theory: theLDA+ DMFT method. *Journal of Physics: Condensed Matter* **2008**, *20*, 064202.
- (9) Selisko, J.; Amsler, M.; Wever, C.; Kawashima, Y.; Samsonidze, G.; Haq, R. U.; Tacchino, F.; Tavernelli, I.; Eckl, T. Dynamical mean field theory for real materials on a quantum computer. *arXiv preprint arXiv:2404.09527* **2024**,
- (10) Choudhary, K.; DeCost, B.; Chen, C.; Jain, A.; Tavazza, F.; Cohn, R.; Park, C. W.; Choudhary, A.; Agrawal, A.; Billinge, S. J.; others Recent advances and applications of deep learning methods in materials science. *npj Computational Materials* **2022**, *8*, 59.
- (11) Choudhary, K.; Garrity, K. F.; Reid, A. C.; DeCost, B.; Biacchi, A. J.; Hight Walker, A. R.; Trautt, Z.; Hattnick-Simpers, J.; Kusne, A. G.; Centrone, A.; others The joint automated repository for various integrated simulations (JARVIS) for data-driven materials design. *npj computational materials* **2020**, *6*, 173.
- (12) Choudhary, K. The JARVIS infrastructure is all you need for materials design. *Computational Materials Science* **2025**, *259*, 114063.
- (13) Choudhary, K.; Garrity, K. Designing high-TC superconductors with BCS-inspired screening, density functional theory, and deep-learning. *npj Computational Materials* **2022**, *8*, 244.
- (14) Choudhary, K.; Wines, D.; Li, K.; Garrity, K. F.; Gupta, V.; Romero, A. H.; Krogel, J. T.; Saritas, K.; Fuhr, A.; Ganesh, P.; others JARVIS-Leaderboard: a large scale benchmark of materials design methods. *npj Computational Materials* **2024**, *10*, 93.
- (15) Wines, D.; Gurunathan, R.; Garrity, K. F.; DeCost, B.; Biacchi, A. J.; Tavazza, F.; Choudhary, K. Recent progress in the JARVIS infrastructure for next-generation data-driven materials design. *Applied Physics Reviews* **2023**, *10*.

- (16) Wines, D.; Choudhary, K.; Biacchi, A. J.; Garrity, K. F.; Tavazza, F. High-throughput DFT-based discovery of next generation two-dimensional (2D) superconductors. *Nano letters* **2023**, *23*, 969–978.
- (17) Wines, D.; Choudhary, K. Data-driven design of high pressure hydride superconductors using DFT and deep learning. *Materials futures* **2024**, *3*, 025602.
- (18) Schmidt, J.; Cerqueira, T. F.; Romero, A. H.; Loew, A.; Jäger, F.; Wang, H.-C.; Botti, S.; Marques, M. A. Improving machine-learning models in materials science through large datasets. *Materials Today Physics* **2024**, *48*, 101560.
- (19) Sanna, A.; Cerqueira, T. F.; Fang, Y.-W.; Errea, I.; Ludwig, A.; Marques, M. A. Prediction of ambient pressure conventional superconductivity above 80 K in hydride compounds. *npj Computational Materials* **2024**, *10*, 44.
- (20) Gao, K.; Cui, W.; Cerqueira, T. F.; Wang, H.-C.; Botti, S.; Marques, M. A. Enhanced Superconductivity in X₄H₁₅ Compounds via Hole-Doping at Ambient Pressure. *Advanced Science* **2025**, e08419.
- (21) Cerqueira, T. F.; Sanna, A.; Marques, M. A. Sampling the materials space for conventional superconducting compounds. *Advanced Materials* **2024**, *36*, 2307085.
- (22) Gross, E. K.; Dreizler, R. M. *Density functional theory*; Springer Science & Business Media, 2013; Vol. 337.
- (23) Choudhary, K.; DeCost, B. Atomistic line graph neural network for improved materials property predictions. *npj Computational Materials* **2021**, *7*, 185.
- (24) De Breuck, P.-P.; Wang, H.-C.; Rignanese, G.-M.; Botti, S.; Marques, M. A. Generative AI for Crystal Structures: A Review. *arXiv preprint arXiv:2509.02723* **2025**,
- (25) Antunes, L. M.; Butler, K. T.; Grau-Crespo, R. Crystal structure generation with autoregressive large language modeling. *Nature Communications* **2024**, *15*, 10570.

- (26) Cao, Z.; Luo, X.; Lv, J.; Wang, L. Space group informed transformer for crystalline materials generation. *Science Bulletin* **2025**,
- (27) Xie, T.; Fu, X.; Ganea, O.-E.; Barzilay, R.; Jaakkola, T. Crystal diffusion variational autoencoder for periodic material generation. *arXiv preprint arXiv:2110.06197* **2021**,
- (28) Jiao, R.; Huang, W.; Lin, P.; Han, J.; Chen, P.; Lu, Y.; Liu, Y. Crystal structure prediction by joint equivariant diffusion. *Advances in Neural Information Processing Systems* **2023**, *36*, 17464–17497.
- (29) Miller, B. K.; Hsu, J.; Macke, S.; Li, S.; Ham, J.; Liu, Z. FlowMM: Generating Crystal Structures with Riemannian Flow Matching. *arXiv 2402.12345* **2024**,
- (30) Choudhary, K.; Campbell, C. R.; Overly, L.; Kumar, A. AtomGPT: Generative Transformer Models for Atomic Structure Discovery. *arXiv 2309.12345* **2023**,
- (31) Xie, T.; Geiger, M.; Friederich, P.; Batzner, S.; Kozinsky, B. Crystal Diffusion Variational Autoencoder. Proceedings of the International Conference on Learning Representations (ICLR). 2022.
- (32) Choudhary, K.; Garrity, K. Designing high-TC superconductors with BCS-inspired screening, density functional theory, and deep-learning. *npj Computational Materials* **2022**, *8*.
- (33) Cerqueira, T. F. T.; Fang, Y.-W.; Errea, I.; Sanna, A.; Marques, M. A. L. Searching Materials Space for Hydride Superconductors at Ambient Pressure. *Advanced Functional Materials* **2024**, *34*, 2404043.
- (34) Choudhary, K. et al. JARVIS-Leaderboard: a large scale benchmark of materials design methods. *npj Computational Materials* **2024**, *10*.
- (35) Shi, H.-L.; Li, Z.-A. Niggli reduction and Bravais lattice determination. *Journal of Applied Crystallography* **2022**, *55*, 204–210.

- (36) Kullback, S.; Leibler, R. A. On Information and Sufficiency. *The Annals of Mathematical Statistics* **1951**, *22*, 79–86.
- (37) Giannozzi, P.; Baroni, S.; Bonini, N.; Calandra, M.; Car, R.; Cavazzoni, C.; Ceresoli, D.; Chiarotti, G. L.; Cococcioni, M.; Dabo, I.; others QUANTUM ESPRESSO: a modular and open-source software project for quantumsimulations of materials. *Journal of physics: Condensed matter* **2009**, *21*, 395502.
- (38) Perdew, J. P.; Ruzsinszky, A.; Csonka, G. I.; Vydrov, O. A.; Scuseria, G. E.; Constantin, L. A.; Zhou, X.; Burke, K. Restoring the density-gradient expansion for exchange in solids and surfaces. *Physical Review Letters* **2008**, *100*, 136406.
- (39) Morel, P.; Anderson, P. W. Calculation of the Superconducting State Parameters with Retarded Electron-Phonon Interaction. *Physical Review* **1962**, *125*, 1263–1271.
- (40) Jiang, A. Q. et al. Mistral 7B. 2023; <https://arxiv.org/abs/2310.06825>.
- (41) Gruver, N.; Sriram, A.; Madotto, A.; Wilson, A. G.; Zitnick, C. L.; Ulissi, Z. Fine-Tuned Language Models Generate Stable Inorganic Materials as Text. 2025; <https://arxiv.org/abs/2402.04379>.
- (42) Chen, R. T. Q.; Lipman, Y. Flow Matching on General Geometries. 2024; <https://arxiv.org/abs/2302.03660>.

Appendix

AtomGPT

Table 1: AtomGPT hyperparameters and implementation details.

Parameter	Value
Model	unsloth/mistral-7b-bnb-4bit
Epochs	2
Batch size (global)	2
Per-device train batch size	2
Gradient accumulation steps	4
Learning rate	2×10^{-4}
Optimizer	AdamW (8-bit)
LR scheduler	Linear
Max sequence length	2048
Test ratio	0.1
Seed	3407
Load in 4-bit	true
Quantization mode	bnb-4bit
Alpaca-style prompt	Instruction:{} Input:{} Output:{} Below is a description of a superconductor material.
Instruction prompt	Generate atomic structure description with lattice lengths, angles, coordinates and atom types.
Output prompt	

The authors of this paper are developers of AtomGPT, so no repository fork was used to compute benchmarks. The repository can be found at github.com/atomgptlab/atomgpt.

AtomGPT Tutorial Notebook: github.com/knc6/jarvis-tools-notebooks/blob/master/jarvis-tools-notebooks/AtomGPT_example.ipynb

CDVAE

Table 2: CDVAE hyperparameters and implementation details.

Parameter	Value
Max atoms per structure	20
Training epochs (max)	100
Early stopping patience	5
Teacher forcing (max epoch)	20
Data split (train / val / test)	0.8 / 0.1 / 0.1
Batch size (train / val / test)	64 / 64 / 64
Encoder settings	Default
Decoder settings	Default
Optimizer settings	Default
Training settings	Default

A fork of CDVAE was utilized for this study. The changes made to the original CDVAE repository are removing Weights & Biases logging, adding configuration files for the JARIVS Supercon-3D and Alexandria DS-A/B datasets, and fixing a bug that led to erroneous values of the project root path. The fork can be found at github.com/crhysc/cdvae.

CDVAE Tutorial Notebook: github.com/crhysc/jarvis-tools-notebooks/blob/master/jarvis-tools-notebooks/cdvae_example.ipynb

FlowwMM

Table 3: FlowMM hyperparameters and implementation details.

Parameter	Value
Max atoms per structure	24
Training epochs (max)	100
Early stopping patience	5
Teacher forcing (max epoch)	20
dim_coords	3
Data split (train / val / test)	0.8 / 0.1 / 0.1
Batch size (train / val / test)	64 / 64 / 64
Vector field network	Default
Model settings	Default
Optimizer settings	Default

A fork of FlowMM was utilized for this study. The changes made to the original FlowMM repository are removing Weights & Biases logging, adding configuration files for the JARVIS Supercon-3D and Alexandria DS-A/B datasets, and modifying the FlowMM hardcode to accept these datasets. FlowMM was not shipped to automatically let users train on arbitrary datasets, but code modifications discussed in our FlowMM tutorial notebook explain the changes necessary for computing these benchmarks.

FlowMM Tutorial Notebook: github.com/crhyse/jarvis-tools-notebooks/blob/master/jarvis-tools-notebooks/flowmm_example.ipynb