npj | computational materials

Review

Published in partnership with the Shanghai Institute of Ceramics of the Chinese Academy of Sciences

https://doi.org/10.1038/s41524-025-01881-2

Generative AI for crystal structures: a review

Pierre-Paul De Breuck[1,2] ✉, Hai-Chen Wang[1,2], Gian-Marco Rignanese[3,4], Silvana Botti[1,2] & Miguel A. L. Marques[1,2]

The rapid rise of generative artificial intelligence is reshaping materials discovery by offering new ways to propose crystal structures and, in some cases, even predict desired properties. This review provides a comprehensive survey of recent advancements in generative models specifically for inorganic crystalline materials. We outline architectures, representations, conditioning mechanisms, data sources, metrics, and applications, and organize existing models into a unified taxonomy.

The rapid advancement of technology is closely tied to the discovery and development of new materials[1–3]. From energy storage and conversion to electronics and catalysis, progress in these areas relies on materials with well-designed properties. A key step in this process is the computational design and characterization of stable or metastable crystal structures, which can then be prioritized for experimental synthesis and testing. This principle, called materials screening, is central in today's materials discovery.

One of the earliest problems in this context is enumerating plausible crystal structures for a given chemical formula. Traditionally, this task—known as crystal structure prediction (CSP)—used methods to explore the potential energy surface to identify global low-energy configurations. Techniques such as genetic algorithms (USPEX[4,5], random search[6]), particle swarm optimization (CALYPSO[7]), and minima hopping[8] have been widely used to navigate this complex landscape, iteratively refining candidate structures by evaluating their energies. These approaches mimic natural processes and have successfully enabled the design of novel materials[9,10].

However, CSP methods should be distinguished from fully generative approaches. The former traditionally start from a predefined chemical composition and often a specified number of atoms per unit cell. Moreover, they are typically based on an iterative screening process where each candidate structure must undergo an explicit energy calculation. This step is computationally expensive, particularly with high-accuracy methods. Furthermore, the search space in CSP grows exponentially with system size, posing a significant combinatorial challenge.

In contrast, generative methods from artificial intelligence learn the underlying data distribution and chemical rules from large crystal structure databases. Consequently, they can directly suggest novel and plausible crystal structures without a priori constraints on chemistry or stoichiometry. This bypasses the computationally intensive search and initial energy evaluation steps inherent to CSP. Moreover, many generative models can be conditioned on specific target properties, enabling a more direct and efficient path to discovering materials with desired characteristics.

This represents a paradigm shift for materials discovery and design. Historically, discovery was driven by empirical trial-and-error methods, as illustrated by Edison's search for light bulb filaments. Over time, the field has evolved from heuristic approaches to theory-guided synthesis and high-throughput computational screening. Today, generative AI enables proactive material generation, prioritizing candidates in silico before experimental validation.

A number of recent reviews and perspectives have surveyed the broader landscape of inverse and generative design methods in materials science[11–17]. This review aims to provide a comprehensive overview of recent developments in generative models for inorganic crystalline materials. We focus on the design and architecture of these models, their representations, and their ability to incorporate constraints through conditioning. We emphasize that this review is limited to in silico approaches for generating crystal structures, such as in the form of crystallographic information files (CIFs), and does not address synthesis, processing, or experimental characterization.

The remainder of this review is structured as follows: section "Preliminaries" introduces the foundational concepts of generative modeling, including the mathematical formulation of probability distributions, common algorithms, and material representations. Section "Data sources" discusses the importance of training data and currently available datasets. Section "Taxonomy" surveys various generative models applied to inorganic materials, organizing them into a taxonomy based on representation, architecture, conditioning, and materials domain. Section "Evaluation metrics and benchmarks" covers benchmarking efforts and current gaps in evaluation metrics. Finally, the section "Applications" presents practical use cases, the section "Discussion and future directions" outlines future directions, and the section "Conclusion" provides our conclusions.

[1]Interdisciplinary Centre for Advanced Materials Simulation (ICAMS), Ruhr University Bochum, Bochum, Germany. [2]Research Center Future Energy Materials and Systems, University Alliance Ruhr, Bochum, Germany. [3]Institute of Condensed Matter and Nanosciences (IMCN), UCLouvain, Louvain-la-Neuve, Belgium. [4]WEL Research Institute, Wavre, Belgium. ✉e-mail: pierre-paul.debreuck@rub.de

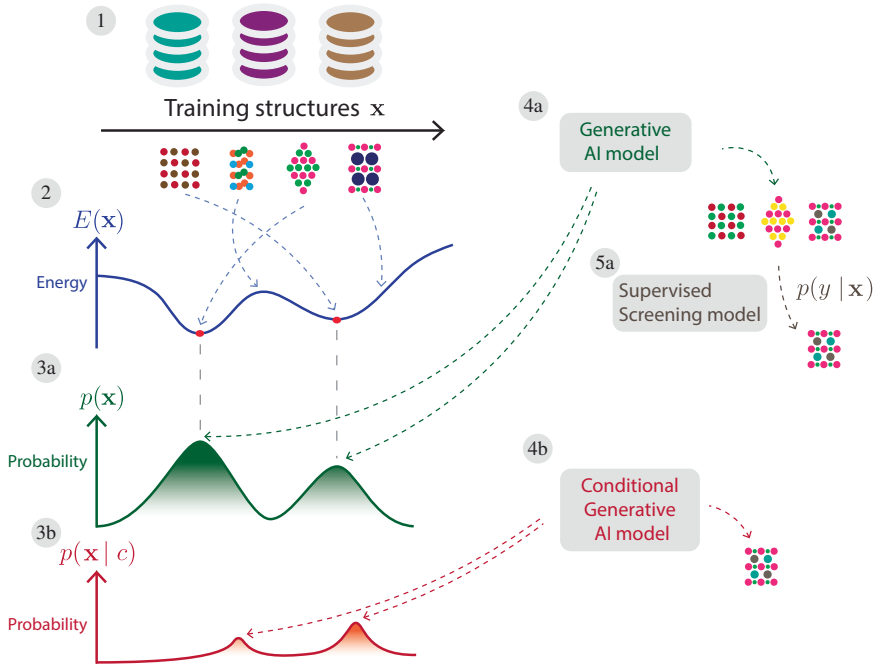

Fig. 1 | Schematic overview of generative AI for crystal structure generation. (1) Crystal structure training data is collected from various databases. (2) The data follows a distribution $\mathbf{x}$ in materials space and which can be converted into an energy distribution $E(\mathbf{x})$. (3) This distribution is implicitly learned by the generative model, providing the basis for the sampling probability $p(\mathbf{x})$. Two approaches are possible: (4a) Unconditional sampling, optionally followed by post-processing such as (5a) property screening, $p(y|\mathbf{x})$ on a target property y; or (3b) Conditional learning of $p(\mathbf{x}|c)$, enabling (4b) direct conditional sampling.

Preliminaries

Problem statement

Generative modeling constitutes a foundational paradigm within artificial intelligence and machine learning, primarily focused on learning the underlying probability distribution, $p(\mathbf{x})$, of a given dataset. Unlike supervised models, which learn a mapping from inputs to outputs to predict a label y given an input $\mathbf{x}$ (i.e., learning conditional probability $p(y|\mathbf{x})$), generative models capture the inherent $p(\mathbf{x})$ of the data itself.

Their core objective is to train a model parameterized by θ such that $p_\theta(\mathbf{x}) \approx p(\mathbf{x})$, allowing to generate from $p_\theta(\mathbf{x})$ novel samples $\mathbf{x}'$ that are statistically indistinguishable to samples drawn from the true data distribution $p(\mathbf{x})$.

This framework naturally extends to solid crystalline materials, where $\mathbf{x}$ denotes a complete crystal structure configuration (i.e., the lattice vectors, the set of atomic species, and their corresponding fractional coordinates) and $p(\mathbf{x})$ denotes the associated probability distribution over such structures. In practice, some configurations are significantly more likely than others, effectively reducing the space of all possible (random) atomic arrangements to those near global or local energy minima. For example, assuming a Boltzmann distribution, $p(\mathbf{x}) \propto \exp(-E(\mathbf{x})/k_B T)$. As a result, low-energy configurations, corresponding to (meta-)stable materials, form the high-probability modes of this distribution and are the primary targets for generative sampling. This is schematically illustrated in Fig. 1.

Rather than relying on explicit chemical rules or direct energy evaluation, generative models implicitly learn the complex, multimodal probability distribution $p(\mathbf{x})$ from datasets of known structures. In doing so, they capture the essential structural motifs and bonding mechanisms of different materials classes.

Generative architectures

In this section, we outline the theoretical foundations underlying the main generative architectures used for materials generation. Here, "architecture" refers to the probabilistic framework that defines how a model samples from $p(\mathbf{x})$. We discuss variational autoencoders (VAEs), generative adversarial networks (GANs), transformers, normalizing flows, and diffusion models. In addition, we highlight large language models (LLMs) as a specific application of the transformer architecture distinguished by their training paradigm. A schematic illustration of these different architectures and paradigms is provided in Fig. 2.

Variational autoencoders (VAEs). VAEs, introduced by Kingma and Welling[18], combine dimensionality reduction with probabilistic generative modeling. They extend the classical autoencoder architecture, i.e., an encoder network that maps a high-dimensional input $\mathbf{x}$ to a lower-dimensional latent vector $\mathbf{z}$ and a decoder network that reconstructs $\mathbf{x}$ from $\mathbf{z}$, by introducing a probabilistic treatment of the latent space.

The VAEs define an approximate posterior distribution $q_\phi(\mathbf{z}|\mathbf{x})$, typically chosen as a multivariate Gaussian $\mathcal{N}(\boldsymbol{\mu}_\phi(\mathbf{x}), \mathrm{diag}(\boldsymbol{\sigma}^2_\phi(\mathbf{x})))$. From this distribution, one can draw a sample $\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})$ and pass it to the decoder, which models the likelihood $p_\theta(\mathbf{x}|\mathbf{z})$. This stochastic encoding makes the decoder more robust to variations in the latent space and enables the generation of novel samples by sampling directly from the prior.

The prior distribution $p(\mathbf{z})$, usually set to a standard normal $\mathcal{N}(0, I)$, imposes a smooth, continuous structure on the latent manifold. VAE training consists of maximizing the evidence lower bound (ELBO) on the marginal log-likelihood $\log p(\mathbf{x})$:

$$\mathcal{L}_{\mathrm{ELBO}}(\mathbf{x};\theta,\phi) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p(\mathbf{z})). \quad (1)$$

The first term, the *reconstruction* loss, encourages accurate recovery of $\mathbf{x}$ from $\mathbf{z}$, while the second one, the Kullback–Leibler *divergence*, regularizes the approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$ toward the prior $p(\mathbf{z})$. This ensures that the latent space is distributed in a way that permits smooth sampling.

Once the VAE is trained, the generation of new samples $\mathbf{x}'$ is straightforward: sample $\mathbf{z}' \sim p(\mathbf{z})$ and compute $\mathbf{x}' \sim p_\theta(\mathbf{x}|\mathbf{z}')$. This two-step process transforms the complex problem of modeling the data distribution $p(\mathbf{x})$ into learning and sampling from a well-behaved latent space, enabling the generation of new, plausible data points.

Generative adversarial networks (GANs). Instead of explicitly learning the probability distribution $p(\mathbf{x})$, GANs, introduced by Goodfellow et al.[19,20], are trained to produce samples through an antagonistic process involving two distinct neural networks: a generator (G) and a discriminator (D). The generator tries to fool the discriminator into believing its outputs are real, while the discriminator attempts to distinguish between real samples and synthetic ones produced by the generator.

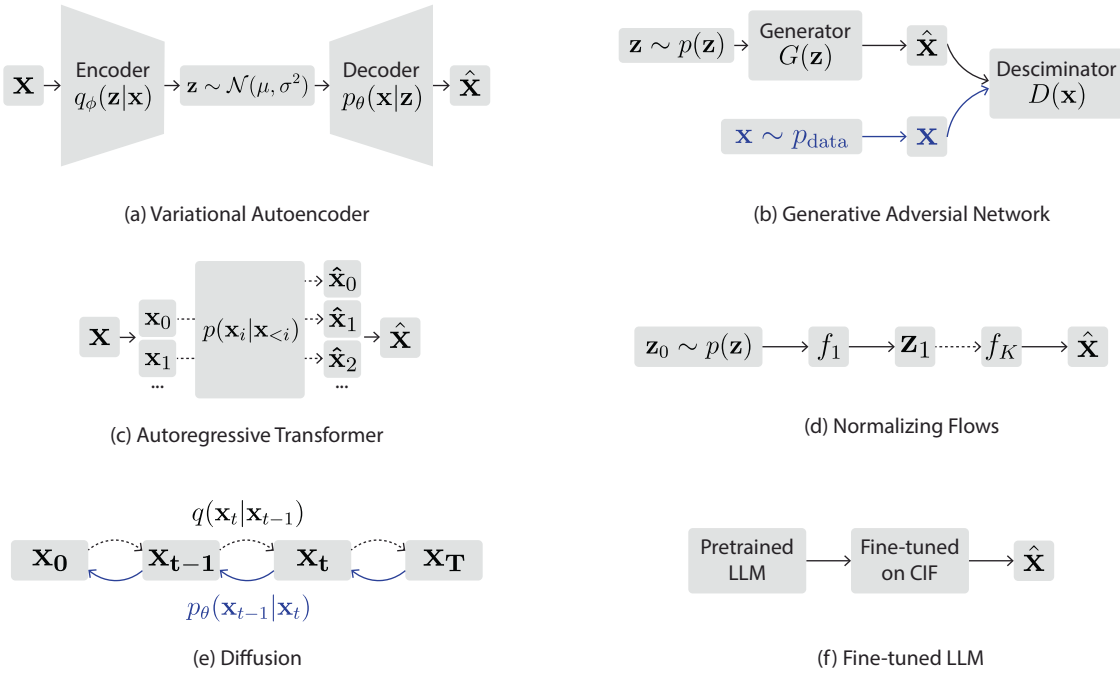

Fig. 2 | Schematic overview of the main generative model architectures and paradigms. a Variational autoencoder (VAE), **b** generative adversarial network (GAN), **c** autoregressive transformer, **d** normalizing flow, **e** diffusion model, and **f** large language model (LLM) as a transformer-based paradigm distinguished by its training strategy rather than by a distinct architecture. For detailed mathematical formulations and discussions, see the main text.

In this two-players-game scenario, the generator learns to map the latent vector $\mathbf{z}$, typically drawn from a simple prior distribution such as a standard normal $\mathcal{N}(0, I)$, to a data-like output $\tilde{\mathbf{x}} = G(\mathbf{z})$. Meanwhile, the discriminator aims to maximize the probability of correctly classifying generated data versus real samples from the true data distribution $p_{\text{data}}(\mathbf{x})$. The adversarial training objective is formally expressed as:

$$\mathcal{L} = \min_G \max_D \quad \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}(\mathbf{x})}[\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))] \tag{2}$$

With this loss, the generator iteratively improves its ability to produce samples that are indistinguishable from the true data, while the discriminator becomes more adept at identifying subtle artifacts. Ideally, training converges to a Nash equilibrium where the discriminator cannot reliably distinguish real from generated samples, i.e., $D(\mathbf{x}) = 0.5$ for all inputs, and the generator captures the underlying data distribution $p(\mathbf{x})$.

Unlike VAEs, which explicitly optimize a likelihood-based objective and impose structure on the latent space through regularization, GANs do not require an explicit form for $p(\mathbf{x})$ or a reconstruction loss.

However, GANs often suffer from instabilities during training and mode collapse, where the generator only produces a limited diversity of samples.

Transformers. The transformer, introduced by Vaswani et al.[21], is a foundational deep learning model renowned for its scalability, parallelizability, and ability to capture long-range dependencies. It dispenses with recurrence and convolution, relying instead on self-attention mechanisms to process input sequences.

The model operates on a sequence of input embeddings, $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$, which are enhanced with positional encodings to preserve token order. In materials science, structural information like atoms, bonds, or symmetry descriptors can be represented as sequences, making the transformer well suited for learning from material structure data.

Decoder-only transformers are used for generative tasks by learning the joint probability distribution of sequences. This is achieved by factorizing it using the chain rule:

$$p(\mathbf{x}) = \prod_{i=1}^{N} p(\mathbf{x}_i|\mathbf{x}_{<i}) \tag{3}$$

Here, x_i represents the i-th token in the sequence. This autoregressive formulation allows for the generation of new material structures by sequentially sampling tokens from the learned distribution.

To maintain this autoregressive property, the decoder employs a masked self-attention layer, which prevents access to future positions during training. The self-attention mechanism computes a weighted sum of all token representations in the sequence, allowing it to capture contextual dependencies regardless of their distance. For each attention head, the input is projected into queries ($\mathbf{Q}$), keys ($\mathbf{K}$), and values ($\mathbf{V}$), and the output is calculated as:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right)\mathbf{V} \tag{4}$$

where d_k is the dimensionality of the key vectors. This is extended by multi-head attention, which performs several such computations in parallel to capture diverse relationships across different sub-spaces of representations.

The efficient and flexible design of the transformer has led to state-of-the-art performance in numerous sequence modeling tasks, driving its adoption in fields ranging from natural language processing to protein folding and materials discovery.

Normalizing flows. Normalizing flows are a class of generative models that transform a simple probability distribution into a complex one through a sequence of invertible and differentiable mappings. Unlike VAEs or GANs, flows provide exact log-likelihood estimation and tractable sampling by explicitly modeling the data distribution[22,23].

Given a base distribution $p_{\mathbf{z}}(\mathbf{z})$, typically a standard normal $\mathcal{N}(0, I)$, a normalizing flow defines a bijective transformation $f_\theta : \mathbf{z} \leftrightarrow \mathbf{x}$ such that the output $\mathbf{x}$ follows the target data distribution. The inverse $f_\theta^{-1} : \mathbf{x} \leftrightarrow \mathbf{z}$ allows

for the computation of the exact density using the change of variables formula:

$$\log p_{\mathbf{x}}(\mathbf{x}) = \log p_{\mathbf{z}}(f_\theta^{-1}(\mathbf{x})) + \log\left|\det\left(\frac{\partial f_\theta^{-1}(\mathbf{x})}{\partial \mathbf{x}}\right)\right|. \quad (5)$$

By composing multiple simple transformations $\mathbf{z}_0 \leftrightarrow \mathbf{z}_1 \leftrightarrow \cdots \leftrightarrow \mathbf{z}_K = \mathbf{x}$, complex distributions can be modeled while maintaining tractability. Each transformation must be invertible and have a Jacobian determinant that is easy to compute.

Beyond discrete flows, continuous normalizing flows[23] replace the sequence of mappings with an ordinary differential equation parameterized by a neural network, enabling more flexible transformations via continuous-time dynamics. More recently, flow matching (FM)[24] provides a simplified training objective for learning such continuous flows by directly matching vector fields between distributions, leading to stable and scalable generative models. By learning smooth mappings between structured latent spaces and high-dimensional material representations, these models can learn to generate new materials.

Diffusion. Diffusion models have emerged as a highly successful class of generative models, synthesizing data by systematically reversing a gradual noising process, with early application to images[25], point clouds[26], and molecular conformations[27], often exceeding GANs in quality. Originally inspired by non-equilibrium thermodynamics (see the work of Sohl-Dickstein et al.[28]), these models consist of two main parts: a fixed forward process that incrementally adds noise to data until it resembles pure noise, and a learned reverse process that reconstructs realistic samples by denoising at each step.

We explain here the denoising diffusion probabilistic models (DDPMs)[29]. The forward process is defined as a Markov chain that adds Gaussian noise to a data sample $\mathbf{x}_0$ over T discrete time steps according to a fixed variance schedule $\{\beta_t\}_{t=1}^T$:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}). \quad (6)$$

As $t \to T$, the distribution $q(\mathbf{x}_T)$ converges to an isotropic Gaussian distribution $\mathcal{N}(0, \mathbf{I})$. The core task is to learn the reverse process, $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$, which approximates the true (but intractable) posterior $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$. This is typically achieved by training a neural network, $\boldsymbol{\epsilon}_\theta$, to predict the noise that was added at a given step t. The training objective simplifies to the following loss:

$$\mathcal{L} = \mathbb{E}_{t,\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right\|^2\right], \quad (7)$$

where $\boldsymbol{\epsilon}$ is the sampled noise and $\mathbf{x}_t$ is the noised version of $\mathbf{x}_0$ at step t.

To generate a new sample, one starts with pure noise $\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$ and iteratively applies the learned denoising function to step backward in time, eventually yielding a clean sample $\mathbf{x}_0$. Key variants of this are the denoising diffusion implicit models, which introduce a deterministic sampling path that allows for much faster generation with fewer steps. Another variant includes modeling the gradient[30] and a further generalization casts the process in continuous time, leading to score-based models that solve stochastic differential equations (SDEs) to transform noise into data[31].

Large language models (LLMs). A final category involves the LLMs[32–34], often fine-tuned on crystallographic data. It is important to emphasize that these models are not a distinct architectural class: they are fundamentally based on the transformer architecture. What differentiates LLMs is their training paradigm, in which a large transformer model is first pre-trained on massive general text corpora and subsequently fine-tuned for specialized scientific domains. Because many generative approaches in materials science build on pre-trained LLMs without modifying the underlying architecture, we present them in a separate section to highlight this paradigm.

The core idea is to treat structural information as a form of language. CIFs, simplified molecular-input line-entry systems (SMILES), or other textual representations of atomic structures are used as training data. The LLM is then fine-tuned on this dataset, adapting its learned linguistic capabilities to understand the "grammar" and "vocabulary" of crystal structures, i.e., the rules governing symmetry, atomic arrangements, and bonding. By retraining the model with the specific objectives of generating valid text-based structural descriptions, it learns to produce novel CIFs or other representations that can be further interpreted (by machine or human) into physically plausible materials.

Representations

Modeling the probability distribution $p(\mathbf{x})$ of materials requires an appropriate representation for $\mathbf{x}$. Unlike supervised learning, where simple descriptors (e.g., average mass or volume) may suffice, generative modeling demands an invertible representation—one that preserves all structural/chemical information and allows exact reconstruction of materials from their representations.

Ideally, representations should be invariant to translation, rotation, and permutation. However, these invariances are not strict requirements, as data augmentation can help models learn such symmetries[35,36]. Below, we summarize the main invertible representations used in crystal generative AI models.

Point cloud or AXL. One of the most straightforward ways to describe a crystal structure is as a set of points in $\mathbb{R}^3$, where each point corresponds to an actual atom. The set of total N atoms in a crystal cell can be expressed as a set $\{s_i, (x_i, y_i, z_i)\}_{i=1}^N$, where s_i and (x_i, y_i, z_i) are respectively the chemical species and Cartesian coordinate of the i-th atom. Next, the periodicity of the cell is captured by including the lattice $\mathbf{L}$, which is typically defined by the lengths of the lattice vectors (a, b, c) and the angles between the lattice vectors (α, β, γ). Thus, the complete representation becomes $\mathbf{P} = \{s_i, (x_i, y_i, z_i)\}_{i=1}^N, (a, b, c, \alpha, \beta, \gamma)$. In the literature, this is also often referred to as the $(\mathbf{A}, \mathbf{X}, \mathbf{L})$ representation, where $\mathbf{A}$ corresponds to the species, $\mathbf{X}$ to the coordinates, and $\mathbf{L}$ to the lattice. This representation is invertible as required, since with any given $(\mathbf{A}, \mathbf{X}, \mathbf{L})$, the corresponding crystal structure can be reconstructed.

Regarding symmetry, this representation explicitly specifies atomic species and positions but does not directly encode point-group or space-group symmetries beyond translational periodicity. The translational symmetry is captured through the lattice parameters, as the lengths and angles of the lattice vectors implicitly define the underlying crystal system.

Voxel grid. The voxel grid representation discretizes the 3D space of the crystal into a regular grid of voxels, similar to a 3D image. Each voxel in the grid holds a value that reflects the presence or density of atoms within that region. Therefore, atoms are represented as continuous functions, typically Gaussians, centered at their fractional (reduced) coordinates within the unit cell. This provides a smooth representation of atomic positions. For materials with multiple chemical species, the voxel grid can have multiple channels, with each channel dedicated to a specific element. The resolution of the grid is a crucial hyperparameter, dictating the level of detail the representation captures and the computational cost. The crystal lattice itself can be encoded by applying transformations to a canonical unit cell. For instance, a unit-centered Gaussian within the unit cell can be generated and then transformed according to the lattice parameters $(a, b, c, \alpha, \beta, \gamma)$ into a cubic grid. This representation allows the model to operate on image-like 3D data and leverage convolutional operations. As it naturally drew inspiration from the image processing field, voxel was one of the earliest representations used in crystal generative AI models.

Similar to the atomic-coordinate representation, the voxel grid does not explicitly encode crystal symmetries beyond translational periodicity, which is implicitly included through the lattice transformation.

Graph. The graph representation encodes a crystal structure as a multigraph $G = (V, E)$, where each node $v_i \in V$ corresponds to an atom and each edge $e_{ij} \in E$ represents an interaction or bond between atoms i and j. Nodes are typically annotated with elemental features (e.g., atomic number, electronegativity, etc.), while edges commonly carry information about distances (bond length) or bond types. Optionally, secondary and further graphs can be added, to capture higher-order interactions (e.g. bond angles). Unlike point cloud or voxel representations, graphs are not inherently invertible and lose exact geometric information, making them unsuitable as stand-alone representations for generative models. Instead, they are often used jointly with point cloud representations to iteratively update atomic positions during generation.

The graph representation does not explicitly encode crystallographic symmetry, as it captures only local atomic connectivity without reference to the full space group. Furthermore, graphs are generally ill suited for representing unit cell parameters[37]. However, while not explicitly incorporating crystal symmetry, most modern graph-based models are designed to enforce invariance or equivariance to Euclidean transformations (translations, rotations, and reflections) within their message-passing framework.

Reciprocal space. The reciprocal space representation leverages the Fourier transform of the crystal atomic positions to express periodicity and long-range order. A central quantity in this framework is the *structure factor* $F(h, k, l)$, defined as:

$$F(h,k,l) = \sum_j f_j \, e^{-2\pi i(hx_j + ky_j + lz_j)}, \tag{8}$$

where f_j is the atomic scattering factor (or form factor) of atom j, and (x_j, y_j, z_j) are the corresponding fractional coordinates. In practical implementations, scattering factors may be replaced or augmented by learned elemental embeddings, and the sum can be decomposed into multiple channels.

To ensure invertibility, the direct lattice L, defined by its parameters $(a, b, c, \alpha, \beta, \gamma)$, must be encoded alongside the set of structure factors. Given the information of L, the inverse Fourier transform can uniquely reconstruct the periodic positions and crystal structure from the reciprocal space representation.

This representation inherently includes crystallographic symmetry but does not make it explicit. Translational symmetry appears as the discrete periodicity of reciprocal lattice points. Point-group and space-group symmetries manifest as systematic relationships among equivalent structure factors or as systematic absences (extinction rules). Consequently, symmetry information resides in the pattern of Fourier coefficients rather than in explicit symmetry operators.

Wyckoff positions. This representation encodes a structure based on its crystallographic information. Each atom in a crystal is assigned to a *Wyckoff position*, i.e., a set of symmetry-equivalent points defined within the space group of the crystal. In three-dimensional crystallography, there are 230 space groups comprising 1771 distinct Wyckoff positions.

A Wyckoff position is specified by its multiplicity and label (e.g., 4*a*, 4*b*, 8*c*, etc.), the associated space group, and any free parameters. These parameters define the coordinates of a representative point within the asymmetric unit, while the symmetry operations of the space group generate the complete set of symmetrically equivalent atomic positions. Given information about the atomic occupancy (i.e., which atoms occupy which Wyckoff positions) along with the lattice parameters, one can uniquely reconstruct the complete crystal structure.

Unlike previous representations, the Wyckoff formalism explicitly encodes crystallographic symmetry through the space group and its operations. All equivalent atomic sites are generated from a single asymmetric unit by applying these symmetry operations, eliminating redundancy and providing a compact, symmetry-consistent description of the crystal.

Conditioned generation

Many applications in materials discovery require sampling from conditional distributions of the form $p(\mathbf{x}|\mathbf{c})$, where c represents a specific constraint or target attribute, such as a desired chemical composition, space group symmetry, or functional properties such as the electronic band gap, thermal expansion coefficient, superconductivity transition temperature, etc. Conditional models, therefore, enable the targeted generation of materials that are not only structurally valid but also optimized for specific applications. When trained appropriately, such models capture which structures are most likely to exhibit the desired property, as summarized schematically in Fig. 1(3b), (4b). In essence, conditioning parameterizes the generative process on $\mathbf{c}$, such that sampling $\mathbf{x}\prime \sim p_\theta(\mathbf{x}|\mathbf{c})$ yields structures with desirable properties.

In practice, conditioning can be implemented through several general mechanisms, depending on the underlying generative architecture. In the following, we outline these foundational strategies to provide a conceptual framework; specific models employing these techniques are discussed in detail later in the section "Taxonomy".

Direct input augmentation. The most common approach augments the model input with the condition $\mathbf{c}$. The condition, encoded as a scalar, vector, or one-hot representation, is concatenated or otherwise injected into latent variables, intermediate features, or input tokens.

In VAEs, both the encoder and decoder are extended to include the condition

$$q_\phi(\mathbf{z}|\mathbf{x},\mathbf{c}), \quad p_\theta(\mathbf{x}|\mathbf{z},\mathbf{c}) \tag{9}$$

so that the latent representation $\mathbf{z}$ explicitly encodes the target property. In GANs, the generator and discriminator are conditioned as

$$G(\mathbf{z},\mathbf{c}), D(\mathbf{x},\mathbf{c}) \tag{10}$$

For transformer-based architectures, conditioning can be achieved by prepending $\mathbf{c}$ as a special token (e.g., band gap = 3 eV) to the input sequence or injecting it via cross-attention layers. This defines an autoregressive conditional distribution:

$$p_\theta(\mathbf{x}|\mathbf{c}) = \prod_{i=1}^{N} p_\theta(\mathbf{x}_i|\mathbf{x}_{<i},\mathbf{c}), \tag{11}$$

For diffusion and flow-matching models, the condition is embedded into the score or vector-field network,

$$\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, \mathbf{c}), \quad f_\theta^{-1}(\mathbf{x},\mathbf{c}), \tag{12}$$

often combined with classifier-free guidance. In this scheme, both conditional and unconditional models are trained, and mixed during inference to control conditioning strength:

$$\boldsymbol{\epsilon}_{\text{guided}} = \boldsymbol{\epsilon}_{\text{uncond}} + s\big(\boldsymbol{\epsilon}_{\text{cond}} - \boldsymbol{\epsilon}_{\text{uncond}}\big), \tag{13}$$

where $s > 1$ is the guidance scale.

Finally, LLM frameworks achieve conditioning through textual prompts explicitly describing the desired properties. This can be viewed as a text-based form of input augmentation.

Conditional latent priors. Conditional latent priors provide an alternative approach to incorporating property constraints by shaping the latent space itself according to the target property. Instead of sampling from a fixed, unconditional prior $p(\mathbf{z})$, a conditional prior $p_\theta(\mathbf{z}|\mathbf{c})$ is

Table 1 | Common inorganic crystal datasets used for training generative models

Dataset	Description	Size	URL	License
Materials Project[40]	DFT-optimized inorganic crystals (PBE GGA/GGA+U), including experimental and hypothetical structures	155K	https://materialsproject.org	CC BY 4.0
Alexandria[42,43]	Large-scale DFT-optimized inorganic crystals, compatible with MP settings	4.5M	https://alexandria.icams.rub.de	CC BY 4.0
ICSD[41]	Experimentally known inorganic crystal structures; subset included in MP	300K	https://icsd.fiz-karlsruhe.de	Commercial (restricted)
AFLOWLIB[47]	Automated high-throughput DFT database of materials properties	3.5M	http://aflowlib.org	Academic use only
OQMD[48]	DFT database focused on thermodynamic stability and phase diagrams	1.3M	http://oqmd.org	CC BY 4.0
JARVIS[130]	DFT database including properties calculated using PBEsol, OptB88vdW, and TBmBJ	80K	https://jarvis.nist.gov	Open access (MIT for tools)

learned, such that latent codes $\mathbf{z}$ drawn from this distribution are likely to decode into structures satisfying the desired property $\mathbf{c}$.

In practice, this typically involves a two-step procedure: first, a standard generative model (e.g., a VAE) is trained to encode and decode structures; second, a separate model is trained to map property vectors $\mathbf{c}$ to the latent space. Sampling from this learned conditional prior biases generation toward regions corresponding to the target property. However, this technique is relatively uncommon, as training can be challenging for complex or high-dimensional latent spaces, and the conditional prior may not fully capture all feasible structures for a given property.

Latent optimization via property prediction. Another strategy jointly trains a property predictor $F_\psi(\mathbf{z})$ alongside the generative model, encouraging a smooth and monotonic relationship between latent variables and target properties. This allows continuous optimization in the latent space via gradient ascent:

$$\mathbf{z}_{t+1} = \mathbf{z}_t + \eta \nabla_{\mathbf{z}} F_\psi(\mathbf{z}_t), \tag{14}$$

where η is the step size. Decoding intermediate latent points produces a series of crystals with gradually increasing values of the target property. A useful addition is conservative regularization to prevent "latent hacking," i.e., unphysical extrapolations arising from an overfitted property predictor.

Reward guided. Another class of conditioning strategies relies on reinforcement learning or reward-guided generation. Here, a reward or energy function $R(\mathbf{x}, \mathbf{c})$ quantifies how well a generated structure satisfies the target property. The generative model is then trained to produce samples in accordance with this reward, either by maximizing expected reward, as in standard RL, or by sampling proportionally to a reward-based distribution, as in GFlowNet-based approaches:

$$p_\theta(\mathbf{x}|\mathbf{c}) \propto R(\mathbf{x}, \mathbf{c}). \tag{15}$$

This formulation allows flexible incorporation of arbitrary property predictors and constraints into the generation process. A practical limitation is that models typically need to be retrained for each distinct conditioning target $\mathbf{c}$.

Constrained data training. A final and conceptually simple approach is to train or fine-tune the generative model directly on a dataset restricted to structures satisfying the desired property, $\mathcal{D}(\mathbf{x}|\mathbf{c})$. This strategy applies to all generative architectures and implicitly learns $p(\mathbf{x}|\mathbf{c})$ without explicit conditioning. As in reward-guided generation, a separate model or fine-tuning is required for each target property.

To summarize, several conditioning strategies exist, including direct input augmentation, conditional latent priors, latent optimization, reward-guided training, and constrained data training. As discussed further in section "Taxonomy," the most recent models rely on direct input augmentation, often fine-tuned on smaller, property-specific datasets. Reinforcement learning and latent optimization remain less common but are promising, as they can incorporate external property predictors with the potential to be more generalizable, better regularized, and trainable within active learning loops. Overall, conditional generative modeling for materials is still in its early stages, with current work addressing only a limited set of target properties. Extending these methods to a broader set of physical and functional constraints, including multi-objective or Pareto-guided optimization, represents an important direction for future research.

Data sources

Beyond the choice of architecture (e.g., VAE, GAN) and structure representation, access to high-quality training data is crucial. Such data typically consist of low-energy, thermodynamically stable crystal structures, sampled from the underlying distribution $p(\mathbf{x})$. Table 1 summarizes key resources in this area.

In practice, most datasets rely on density functional theory (DFT) calculations at the generalized gradient approximation (GGA) level[38], typically using the Perdew, Burke, and Ernzerhof (PBE) functional[39]. The fidelity of the generated crystal structures and their predicted properties is fundamentally constrained by the accuracy of the underlying DFT training data, as models will naturally reproduce both the strengths and limitations of the density functional approximations used in their training datasets. Furthermore, as DFT calculations scale poorly with system size, all large-scale materials datasets are inherently biased toward smaller unit cells, typically containing fewer than 20–30 atoms. Although exceptions exist (e.g., certain complex oxides), these represent only a small fraction of entries. It should therefore be kept in mind that this bias directly impacts the learned structural distribution and can influence model performance and generalizability.

A widely used source is the Materials Project (MP)[40], which contains a diverse set of DFT-relaxed inorganic structures. Many of these entries are derived from the Inorganic Crystal Structure Database (ICSD)[41], providing a link to experimentally observed compounds. Filtering MP entries by ICSD provenance or by applying a threshold on the energy above the hull (e.g., <80 meV/atom) is a common heuristic to bias the dataset toward thermodynamically stable or metastable phases, although it does not strictly guarantee stability in generated compounds.

The Alexandria database[42,43] significantly expands the available data, offering over 4.5 million 3D structures computed under MP-compatible settings using VASP[44]. This massive size enables significant improvement in quality in generative frameworks such as MatterGen[45] and Matra-Genoa[46] that were trained on its contents.

Additional valuable resources include AFLOWLIB[47], OQMD[48], and JARVIS[49], all emphasizing high-throughput DFT calculations. In this framework, it is worth mentioning the OPTIMADE API, which provides users with an easy common access to many of these world-leading materials databases[50,51]. A full list of the databases implementing the API is available on the OPTIMADE providers dashboard.

Finally, Xie et al.[52] introduced task-specific datasets for generative models, including Perov-5, a curated set of ~19,000 perovskites designed for water splitting[53,54], and Carbon-24, which contains over 100,000 carbon allotropes generated via ab initio random structure searching[6]. However,

Table 2 | Taxonomy for generative models applied to crystal structures

Category	Description	Possible values
Representation	How the crystal is encoded as input/output	Voxel grid
		Graph Wyckoff positions
		Point cloud
		Reciprocal space
		etc.
Architecture	Core generative model framework	Variational Autoencoder (VAE)
		Generative Adversarial Network (GAN)
		Normalizing flow
		Diffusion model
		Transformer
		Fine-tuned LLM
		etc.
Conditioning	Extra information used to guide generation	Unconditional (no external constraints)
		Composition (elemental fractions)
		Space group or crystal system
		Band gap (numeric)
		Formation energy
		Target density or porosity
		etc.
Material Domain	Subset of materials modeled or generated	Oxides, halides, or any limited chemistry
		Perovskites, cubic materials, or any structural prototype
		Alloys
		All structures (unconstrained)
		etc.

Generative models can be distinguished by four key aspects: the choice of data representation for the crystal, the core model architecture, the conditioning information used to guide material generation, and the target material domain (limited by the architecture or training data).

these datasets, as recently discussed in ref. 55, have some limitations due to duplicates or polymorphic structures present in the train and test set that can lead to misleading benchmark metrics.

Taxonomy

The methodology and intrinsic workings of generative models for crystal structures can be systematically understood through four key pillars: *representation*, *architecture*, *conditioning*, and *materials domain*. Together, these elements define the design space of existing approaches and form the basis of the taxonomy presented in Tables 2 and 3.

The *representation* specifies how crystal structures are encoded, ranging from point clouds and voxel grids to graphs, reciprocal-space descriptors, and Wyckoff positions, as explained above. This choice determines how symmetry, periodicity, and atomic details are captured. In practice, many combinations are possible, and models often employ multiple representations simultaneously. For example, FTCP[56] integrates reciprocal- and real-space features into a hybrid representation, while graph-based models are commonly paired with point clouds to iteratively refine atomic positions.

The *architecture*, as defined here, refers to the foundational generative framework, including GANs, VAEs, transformers, normalizing flows, diffusion models, and fine-tuned LLMs. These architectures learn the underlying distribution of crystal structures based on the chosen representation.

The *conditioning* extends generative modeling from learning an unconditional distribution $p(\mathbf{x})$ to a conditional distribution $p(\mathbf{x}|\mathbf{c})$, where $\mathbf{c}$ encodes the vector representation of the target property. While formation energy is often implicitly modeled, some approaches explicitly condition on additional properties such as magnetism, electrical conductivity, or thermal expansion to guide generation. Conditioning is marked as "yes" only when the authors explicitly demonstrate generation conditioned on a functional property; otherwise, "no" is stated. When conditioning is used, the number in parentheses indicates the mechanism, as numbered in the section "Preliminaries" subsection "Conditioned generation".

The *materials domain* defines the scope over which a model can operate, reflecting both its architectural design and training data. This can be either limited by chemistry (e.g., oxides, halides), structural prototypes (e.g., perovskites, cubic materials), or other subdomains such as alloys or 2D materials. However, actual performance across the full domain is not guaranteed, due to limitations in the training data such as fidelity, bias toward smaller unit cells, and other intrinsic biases (see "Data sources" section).

Figure 3 illustrates how the first two dimensions of the taxonomy—architecture and representation—create a discrete space for organizing current models, thereby visually summarizing the current landscape.

The following sections review these models in detail, grouping them by their common architectural and representational strategies into common subsections.

GANs and VAEs

Inspired by successes in image generation, early generative models in materials science adapted similar techniques. GANs and VAEs operating on voxel representations emerged as the first methods for crystal structure generation. These models treat 3D atomic arrangements as image-like density grids, compressing them into continuous latent spaces for sampling new structures.

Hoffman et al.[57] represented structures within a 10 Å cube, discretized into a $30 \times 30 \times 30$ grid. Each voxel held a density value derived from Gaussian distributions centered on atoms, scaled by atomic number. A U-Net architecture was then used to segment this density grid to determine atomic positions. Despite data augmentation and high accuracy on the MP database, the model struggled to generate physically stable structures.

iMatGen[58] extended the voxel approach using fractional coordinates and separate grids for unit cell representation (using a transformed Gaussian). A hierarchical, two-step VAE encoded the materials into a latent space. The channel-based design, in principle, allowed this model to be extended to other chemical systems.

ICSG3D[59] represented crystals using voxelized electron density grids with summed Gaussian densities. It used a 3D U-Net for segmentation to recover atomic coordinates. A one-hot encoding of the energy was appended to the latent space of the VAE to condition on formation energy. Trained on cubic binary alloys, ternary perovskites, and Heusler compounds, ICSG3D demonstrated the ability to interpolate between known structures—for instance, varying the A-site element in a perovskite across a periodic row by traversing its latent space.

CCDCGAN[60] introduced a two-stage approach in which a VAE was first trained to create a structured latent space, which subsequently served as input for a GAN. This framework incorporated a formation energy predictor directly into the loss function, compelling the generator to find low-energy structures by seeking minima in the latent space. The work was later extended to multi-component systems[61].

While voxel-based methods are prominent, GANs using other representations are also found in the literature. For instance, CrystalGAN[62] and GANCSP[63] operated directly on point clouds of atomic coordinates, with the latter also enabling conditioning on the composition. More recent approaches evolved the voxel concept for greater efficiency; as an example, Uni-3DAR[64] used an octree-compressed voxel representation to convert a 3D structure into a compact token sequence. A transformer was then autoregressively used on these tokens to rapidly sample new crystals.

Table 3 | Summary of generative models found in literature following the presented taxonomy

Acronym	Representation	Architecture	Condit.	Domain	Ref.	Year
CrystalGAN	Point Cloud	GAN	No	Hydrides	62	2019
DD3DCS	Voxel	GAN	No	All Structures	57	2019
CondGAN	Bag of Atoms	GAN	Yes (1)	Compositions	139	2019
iMatGen	Voxel Grid	VAE	No	Vanadium Oxides	58	2019
MatGAN	One-Hot Elements	GAN	Yes (5)	Compositions	140	2020
GANCSP	Point Cloud	GAN	Yes (1)	All Structures	63	2020
ICSG3D	Voxel	VAE	Yes (1)	Cubic Alloys, Perovskites, Heusler Compounds	59	2020
CubicGAN	Point Cloud	GAN	Yes (1)	Cubic Structures	123	2021
CCDCGAN	Voxel Grid	VAE, GAN	Yes (3)	Bi-Se Materials	60	2021
FTCP	Reciprocal Space, Point Cloud	VAE	No	All Structures	56	2022
CDVAE	Point Cloud, Graph	VAE, Diffusion	Yes (3)	All Structures	52	2022
CCDCGAN	Voxel Grid	VAE, GAN	Yes (3)	All Structures	61	2022
PGCGM	Wyckoff	GAN	No	Ternaries	83	2023
XYZTransformer	Point Cloud	LLM	No	All Structures	91	2023
PCVAE	Composition, Prototype	VAE	No	Prototypes	141	2023
LCOM	Graph, Point Cloud	VAE, Diffusion	Yes (3)	All Structures	65	2023
CHGlownet	Graph	GFlowNet	No	All Structures	109	2023
SLI2Cry	Graph (SLICES)	RNN	No	All Structures	110	2023
SyMat	Graph, Point Cloud	Diffusion	No	All Structures	66	2023
Crystal-GFN	Prototype (Space Group)	GFlowNet	Yes (4)	Structural Prototypes	108	2023
GemsDiff	Graph, Point Cloud	Diffusion	No	All Structures	67	2023
CGMD	Point Cloud	Diffusion, VAE, Flow Matching	No	All Structures	111	2024
DP-CDVAE	Point Cloud, Graph	Diffusion	No	All Structures	68	2024
CrysTens	Point Cloud (Pairwise Distance)	GAN, Diffusion	No	All Structures	112	2024
CrystalTextLLM	Point Cloud	LLM	Yes (1)	All Structures	92	2024
DiffCSP	Point Cloud, Graph	Diffusion	No	All Structures	69	2024
DiffCSP++	Wyckoff	Diffusion	No	All Structures	70	2024
Con-CDVAE	Point Cloud, Graph	VAE, Diffusion	Yes (2)	All Structures	71	2024
NSGAN	Composition Vector	GAN, GA	No	Alloys	113	2024
UniMat	Point Cloud	Diffusion	Yes (1)	All Structures	81	2024
FlowMM	Point Cloud (Flat Manifold)	Flow Matching	No	All Structures	104	2024
StructRepDiff	Embedded Atom Density	Diffusion	No	All Structures	142	2024
CrystalFormer	Wyckoff	Transformer	Yes (1)	All Structures	88	2024
VGD-CG	Composition One-Hot	GAN, VAE, Diffusion	Yes (1)	Compositions	115	2024
LCMGM	Reciprocal Space	VAE, GAN	No	Perovskites	143	2024
GenMS	Point Cloud	LLM, Diffusion	Yes (1)	All Structures	99	2024
WyCryst	Wyckoff	VAE	Yes (3)	All Structures	84	2024
MatExpert	Point Cloud (Conversational)	LLM	Yes (1)	All Structures	100	2024
FlowLLM	Point Cloud (Flat Manifold)	LLM, Flow Matching	Yes (1)	All Structures	105	2024
Cond-CDVAE	Point Cloud, Graph	VAE, Diffusion	Yes (1)	All Structures	72	2024
CrystaLLM	CIF File	LLM, Transformer	Yes	All Structures	95	2024
CrysText	CIF File	LLM	Yes (1)	All Structures	96	2024
CGWGAN	Wyckoff	GAN	No	All Structures	86	2024
Matra-Genoa	Wyckoff	Transformer	Yes (1)	All Structures	46	2025
EH-Diff	Hypergraph, Point Cloud	Diffusion	No	All Structures	73	2025
CrysBFN	Point Cloud (Hyper-Torus)	Bayesian	No	All Structures	106	2025
TransVAE-CSP	Graph, Point Cloud, RBF	VAE, Diffusion	No	All Structures	74	2025
CrystalFlow	Graph, Point Cloud	Continuous Normalizing Flow	Yes (1)	All Structures	144	2025
MatLLMSearch	Text (JSON)	LLM, Evolutionary Search	Yes (3)	All Structures	101	2025
Mat2Seq	Invariant Sequence	LLM	Yes (1)	All Structures	102	2025

Table 3 (continued) | Summary of generative models found in literature following the presented taxonomy

Acronym	Representation	Architecture	Condit.	Domain	Ref.	Year
MatterGen	Point Cloud, Graph	Diffusion	Yes (1)	All Structures	45	2025
TGDMat	Point Cloud, Contextual	Diffusion	Yes (1)	All Structures	75	2025
NatureLM-Mat3D	Point Cloud	LLM	Yes (1)	All Structures	103	2025
CrystalGRW	Graph, Manifold	Geodesic Random Walk	Yes (1)	All Structures	79	2025
UniGenX	Point Cloud	Transformer, Diffusion	Yes (1)	All Structures	82	2025
DAO-G	Graph, Point Cloud	Diffusion (EBM)	No	All Structures	77	2025
Uni-3DAR	Voxel (Compressed)	Transformer	No	All Structures	64	2025
Chemeleon	Graph, Point Cloud, Text	Diffusion, Contrastive Learning	Yes (1)	All Structures	76	2025
SymmCD	Wyckoff (Binary Matrix), Graph	Diffusion	No	All Structures	89	2025
WyckoffDiff	Wyckoff, Graph	Diffusion	No	Protostructures	90	2025
KLDM	Manifold, Graph	Diffusion	No	All Structures	80	2025
OMatG	Point Cloud, Graph	Stochastic Interpolants	No	All Structures	116	2025
CHGGen	Point Cloud, Graph	Diffusion (with inpainting)	No	All Structures	78	2025

Refer to Table 2 and the main text for the definition of "representation," "architecture," and "conditioning." Conditioning is marked as "yes" only when the authors explicitly demonstrate generation conditioned on a functional property. The number in parentheses indicates the conditioning mechanism, as numbered in section "Preliminaries" subsection "Conditioned generation".

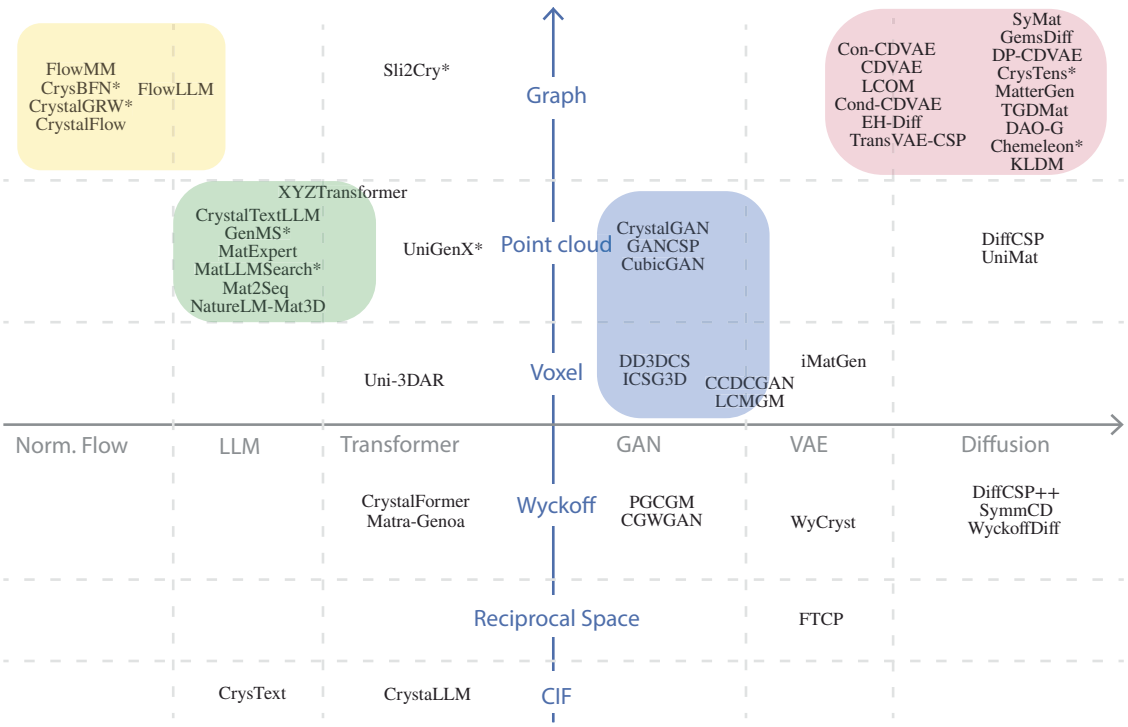

Fig. 3 | Schematic representation of the taxonomy of models in a two-dimensional architecture-representation space. The space is discrete, indicated by dashed lines. Some models lie at intersections, combining multiple architectures. The main categories are highlighted with colored bubbles, reflecting historically established groupings with significant contributions: GANs on point clouds or voxels (blue), graph diffusion (red), LLMs on point clouds (green), and normalizing flows on point clouds (yellow). Models with architectural variations relative to the indicated category are denoted by (*), see Table 3 for additional details.

Models based on diffusion

In 2021, Xie et al.[52] introduced the Crystal Diffusion Variational Auto-encoder (CDVAE) for periodic material generation. This model has since served as a foundation for many subsequent iterations of diffusion-based approaches. The original generation pipeline began with an encoder, i.e., a periodic graph neural network (PGNN) that mapped materials into a latent space. From this latent representation, the model predicted composition, lattice parameters, and number of atoms, which served as the initial structure. A diffusion decoder subsequently refined this initial structure by performing Langevin dynamics, which simultaneously denoised atomic coordinates and updated atom types. A noise-conditioned score network guided the denoising process. This network used another PGNN that estimated the gradient of the probability density at various noise levels while explicitly incorporating interactions across periodic boundaries. CDVAE maintained permutation, translation, and rotation invariances through SE(3) equivariant graph neural networks; however failed to achieve translational invariance in denoising. Nevertheless, benchmark evaluations demonstrated that CDVAE achieved relatively high validity and coverage for materials generation.

Several extensions were built on the CDVAE architecture. Qi et al. proposed LCOM[65], which extended CDVAE by introducing a surrogate conservative model aimed at minimizing formation energy in the latent space. Similarly, the SyMat[66] framework followed the CDVAE architecture but further strengthened invariance with respect to translation. The GemmsDiff model developed by Klipfell et al.[67], also employed an equivariant GNN, further extending CDVAE. It allowed lattice parameters to evolve within the diffusion process, in contrast to the original CDVAE that fixed them after the VAE.

Another variation, DP-CDVAE[68], modified CDVAE by replacing its score network with a DDPM. This modification enabled the model to generate structures that were closer to their ground states compared to the original CDVAE. Jiao et al. introduced DiffCSP[69], which performed joint diffusion on both lattice parameters and reduced coordinates using a periodic E(3) equivariant denoising model. Unlike CDVAE, DiffCSP started from randomly initialized structures drawn from prior distributions, eliminating the need for a VAE. The model demonstrated excellent validity and coverage. Its successor, DiffCSP++[70], further incorporated crystal symmetry by conditioning the diffusion process on a specific space group. Con-CDVAE[71] extends CDVAE to conditional crystal generation based on user-defined properties such as band gap, formation energy, and crystal system. It introduces property embeddings appended to the latent variable and a diffusion-based prior network that learns to generate latent vectors conditioned on target properties. Cond-CDVAE[72] also extended CDVAE to conditional generation, based on both chemical composition and external pressure. Trained on a combination of the MP dataset and high-pressure CALYPSO CSP data, Cond-CDVAE could generate crystal structures under user-defined conditions via injecting a conditional vector into both the latent space and Langevin dynamics.

MatterGen[45] further evolved the CDVAE architecture by removing the VAE entirely (as in DiffCSP) and employing a fully joint diffusion process over atom types, coordinates, and lattice parameters. Thus, MatterGen allowed the lattice to change during generation. Instead of applying Gaussian noise in Cartesian coordinates, MatterGen diffused in fractional coordinates using a wrapped normal distribution, which inherently respected periodicity and accommodates varying unit cells. MatterGen also used larger training datasets compared to previous diffusion models and implemented conditional generation through inserting lightweight adapter modules into the score network. This enabled property control without retraining the full model.

Recent developments explored various architectural enhancements. EH-Diff[73] replaced the graph representation by a hypergraph, linking more than two atoms by an edge, capturing better multi-atomic interactions. TransVAE-CSP[74] used a methodology similar to CDVAE but enhanced the encoder by using a transformer with equivariant dot product attention. TGDMat[75] used joint diffusion on atom types, fractional coordinates, and lattice structure using a periodic E(3) equivariant denoising model, and further integrated textual information to guide generation with user-defined prompts. In a similar fashion, Chemeleon[76] also integrated textual descriptors. The embeddings for textual and structural data were trained using contrastive learning. Then these embeddings were used in diffusion, enabling material sampling with simple prompts such as "$LiMnO_4$ with orthorhombic structure." DAO[77] used two models for diffusion by incorporating energy considerations into the diffusion paths. The approach was also pre-trained on non-stable structures for enhancing model capability. CHGGen[78] introduced crystal host-guided inpainting generation, conditioning diffusion on symmetrized host structures to mitigate locality bias in graph-based models and therefore enhance symmetric crystal generation.

Other geometric approaches offered alternative perspectives. CrystalGRW[79] proposed diffusion defined on manifolds, i.e., torus for coordinates and simplices for atom types, using geodesic random walks instead of Euclidean Gaussian noise. This method respected the natural geometry of crystal representations better. Drawing inspiration from physical Langevin dynamics, KLDM[80] used manifolds for structure representation but performed diffusion on auxiliary velocity variables in the associated Euclidean space. This avoided complex diffusion directly on curved manifolds while maintaining geometric constraints.

In another direction, UniMat[81] proposed a universal representation in which materials were encoded as a point cloud shaped like the periodic table, with DDPM-based diffusion applied on top. This approach generated structures with lower decomposition energies compared to CDVAE. It is also worth noting that diffusion has been integrated with transformer architectures. For instance, UniGenX[82] generated element types (i.e., chemical formulas) autoregressively via a transformer decoder, while refining atomic coordinates using a diffusion head applied to the next-token embeddings of the transformer. This hybrid design offered controllability under both compositional and structural constraints.

Models based on Wyckoff descriptors

One of the earliest works to incorporate symmetry information into generative models was PGCGM by Zhao et al.[83]. This approach represented symmetry operations using affine matrices and encoded a single fractional coordinate for each set of equivalent sites, thereby implicitly preserving symmetry throughout the generation process. A GAN architecture was used, combined with physics-inspired loss functions: specifically, inter- and intra-atomic distance-based and symmetry-compliant-based losses, as well as average full-coordinate losses. However, the method was limited to ternary systems, though it could likely be extended to more complex compositions.

Built on the use of Wyckoff positions, the WyCryst[84] model encoded them as one-hot matrices-referred to as Wyckoff genes, which were then embedded into a latent space via a VAE. This latent space was guided by a property-prediction branch, ensuring that the space was structured along property gradients for targeted sampling. While this enabled symmetry-aware and property-driven generation, the model does not directly learn the fractional coordinates. Consequently, an additional step using PyXtal[85] is required to heuristically estimate these coordinates.

A related approach is CGWGAN[86] proposed by Su et al., which also leveraged Wyckoff descriptors but adopted a two-step generation process. In the first step, a GAN was used to generate crystal templates defined by the asymmetric unit, space group, and lattice vectors. In the second step, atoms were filled into these templates, guided by a supervised force field model (specifically, M3GNet[87]) to assess thermodynamic and mechanical stabilities.

More recently, Cao et al. introduced CrystalFormer[88], a decoder-only transformer model that generated crystal structures by sampling Wyckoff descriptors conditioned on the space group. The model iteratively predicted Wyckoff sites, fractional coordinates, and lattice parameters, achieving high validity and diversity while outperforming earlier methods in maintaining symmetry consistency. A parallel work by De Breuck et al.[46] introduced Matra-Genoa, an autoregressive transformer that encoded crystals in a fully invertible sequence representation based on Wyckoff positions, including both discrete sites and continuous free coordinates. Trained on over two million structures, Matra-Genoa could also be conditioned on thermodynamic stability by explicitly incorporating the distance to the convex hull during training using a stability token. Together with CrystalFormer, Matra-Genoa combined for the first time discrete Wyckoff information with continuous positional data.

Finally, Wyckoff descriptors can also be fully utilized using other types of architectures, such as diffusion. SymmCD[89] learned the joint distribution of the asymmetric unit cell and corresponding symmetry operations using diffusion. It encodes crystals as a reduced set of crystallographic orbits with a binary symmetry representation and allows conditioning on space group and number of orbits, producing complete 3D structures with atomic coordinates. WyckoffDiff[90] also employed diffusion but operated purely on Wyckoff positions, starting from all positions and diffusing element assignments to produce protostructures, defined by space group and Wyckoff letters. As in WyCryst, an additional heuristic step is required to obtain the corresponding fractional coordinates.

Models based on LLMs

One of the earliest explorations of applying LLMs to structural data files, such as XYZ, CIF, and PDB formats, was carried out by Flam-Shepherd and Aspuru-Guzik[91]. They trained a GPT-like transformer architecture from scratch using next-token prediction, rather than fine-tuning an existing language model. While this means that their work could also be categorized under "transformer" approaches, the focus was on demonstrating that an LLM could directly generate novel and valid structures without relying on advanced, domain-specific representations. This pioneering result sparked further research into leveraging more advanced LLMs for modeling crystal structures directly from text-based formats. Gruver et al. introduced CrystalTextLLM[92], a model that fine-tuned Llama 2-70B[93] on textual point cloud descriptions of crystals. The model tokenized numerical values digit-by-digit, enabling the language model to process them. This setup supported flexible text-conditioned generation, allowed users to prompt the model with specifications like "Below is a description of a bulk material. [The chemical formula is Pm2ZnRh]. Generate a description of the lengths and angles of the lattice vectors and then the elements and coordinates for each atom within the lattice." The model also performed infilling by masking parts of the input and proposing replacements, giving a flexible framework for the user.

Antunes et al. explored CIF generation by training a LLaMA-based[94] language model from scratch (CrystaLLM[95]). Given a chemical formula and the number of formula units, their model predicted complete CIFs token-by-token, successfully generating a wide range of structure types, including perovskites, spinels, zeolites, and metal-organic frameworks.

Mohanty et al. introduced CrysText[96], which combined the CIFs representation from CrystaLLM with the pre-trained LLM approach of CrystalTextLLM. Using a Llama 3.1-8B[97] model fine-tuned with QLoRA[98], CrysText learned the relationships between crystallographic parameters while retaining the flexibility of LLM-based conditioning on natural language prompts.

Moving beyond direct CIFs generation, GenMS[99] proposed a hierarchical approach in which an LLM first predicted plausible intermediate chemical formulas from natural language descriptions. A diffusion model then generated detailed 3D atomic structures conditioned on these formulas. This two-step process outperformed direct CIF generation via prompting, produced structures with higher validity, lower formation energies, and better adherence to user specifications.

Other works focused on making LLM-powered crystal generation more agentic. Ding et al. developed MatExpert[100], a conversational multi-step agent that started with a set of user-defined property requirements. The system then retrieved similar materials from a pre-embedded database using contrastive learning, after which a fine-tuned LLM proposed modifications to improve the target properties. For example, it might suggest: "Mg would be replaced by Na, which coordinates differently with sulfur, forming NaS_6 octahedra with a mix of edge- and corner-sharing geometries." This design not only generated final crystal structures but also explained the reasoning behind changes, making the process more transparent to humans.

Other recent approaches continued to demonstrate the versatility of LLM-based crystal generation. MatLLMSearch[101] showed that LLMs could propose novel structures without fine-tuning. They combined an LLM with evolutionary search: the LLM received two parent structures and was asked to propose new structures, therefore performing implicit crossovers and mutations. The proposed crystal structures were then optimized, filtered, and fed back into the LLM for another round of evolution. Another work, Mat2Seq[102] introduced a systematic way to encode crystals as unique, invariant 1D sequences to enable language-model-based crystal generation. NatureLM-Mat3D[103] used a fine-tuned Llama 3-8B model that represented crystals as lists of elements, space groups, and coordinates. It demonstrated strong performance in crystal generation and had the advantage of being trained on multiple scientific domains by converting them all to sequences, providing a larger science foundation model.

Models using flow matching

Miller et al. introduced FlowMM[104], the first application of FM to crystal structure generation. By learning symmetry-aware vector fields on manifolds that reflect the geometries and periodicity of crystals, FlowMM performed Riemannian FM to efficiently model both CSP and de novo generation. It outperformed prior methods like CDVAE and DiffCSP in accuracy and stability while requiring significantly fewer integration steps, achieving up to three times faster inference with comparable or better results. The authors later extended this work by using an LLM for the base distribution[105]. This approach added a fine-tuned LLM at the start to sample a noisy material description that was subsequently refined using Riemannian FM. This yielded structures with a better stability rate than FlowMM.

Although not directly related to FM and closer to diffusion approaches, CrysBFN[106] used Bayesian flow networks, a class of models in which iterative Bayesian updates are used to update parameters of the target distribution. Another work in this category includes CrystalFlow[106], which introduced continuous FM in combination with a graph-based equivariant neural network. It also demonstrated the possibility of conditioning on external variables such as pressure.

Other models

In this section, we briefly mention other works that do not fall into the previous main categories, as they combine multiple or different representations and architectures. Generative Flow Networks (GFlowNets[107]) are a general generative modeling framework that sequentially samples actions using a policy model trained on a reward signal. By enabling sampling proportional to a reward distribution, GFlowNets efficiently explored multiple modes in the search space, making them a promising approach for crystal generation. Within this line of work, the Crystal-GFN[108] model represented one of the first attempts, although it did not generate full structures. CHGlowNet[109] adopted a hierarchical GFlowNet-based design for crystal generation.

A different approach focused on reversible graph representations: Sll2Cry[110] introduced an invertible SLICES-based representation using RNNs, achieving successful crystal generation. Additional efforts include CGMD[111], which combined point cloud inputs with diffusion, VAE, and flow-matching models. CrysTens[112] converted atomic coordinates into an image-like representation based on pairwise distances, applying both GANs and diffusion models, with diffusion showing superior results. NSGAN[113] integrated GANs with a genetic algorithm operating in the latent space of aluminum alloys. StructRepDiff[113] employed a diffusion model in an invariant descriptor space derived from the embedded atom method[114], reconstructing atomic positions via gradient optimization to preserve symmetries.

Further work involves VGD-CG[115], a composition generator that combines VAE, GAN, and diffusion, conditioned on properties such as decomposition enthalpy, synthesizability, and band gap. The predicted compositions were then converted into structures using a template-based predictor.

Finally, OMatG[116] unifies the diffusion and flow-matching paradigms through stochastic interpolants (SIs), providing a flexible framework that encompasses stochastic differential equations (diffusion) and ordinary differential equations (FM) as limiting cases.

Evaluation metrics and benchmarks

Assessing whether a generative model works well requires a comprehensive multi-metric evaluation. A truly effective generative model should demonstrate not only the ability to discover novel materials with desired properties that are stable and synthesizable, but also the computational efficiency and scalability to do so in practical applications.

To achieve this comprehensive assessment, the evaluation of model performance involves both domain-knowledge-based criteria and data statistical metrics. Early models applied the *validity metric*, where the generated structure is considered valid when certain chemical or physical

constraints are satisfied. For example, the charges (oxidation states) must be balanced, the shortest distances between atoms must be reasonable, and the space groups assigned to the structures by the model must match those from symmetry analysis. For successful models, the validity rates, i.e., the fraction of generated structures that are physically meaningful, are often > 90%. However, these domain-knowledge-based validity checks do not guarantee the stability of the generated structures, which are more stringently evaluated later through DFT calculations.

The *coverage metric* evaluates how well the generated materials match that of a reference test set by computing precision (the fraction of generated structures that are valid and match the reference) and recall (the fraction of reference structures successfully predicted). However, this metric strongly depends on the choice and completeness of the test set. Missing structures in the test set are not inherently problematic—novelty and creativity are desirable in generative models—so the metric has clear limitations and, in our view, is not always well suited for comparing models.

The *realism metric* measures the distance between realistic references and the generated structures. To quantify this distance, one can use the root-mean-square displacement (RMSD) between reference and generated structures. Another approach is to check the difference in terms of symmetries using structure-matching algorithms. Distances (e.g., Wasserstein distance, WD) in distributions of properties (energies, volumes, or targets for conditional models) have also been used. However, such distribution-based distances make comparisons challenging, as smaller improvements do not necessarily indicate a better model but only a closer match to the reference set. Metrics include the number of optimization steps required to relax a generated geometry to a local minimum, which can serve as a distance measure. Shorter optimization paths between generated and realistic structures indicate that the samples are largely physically plausible. In practice, DFT-optimized counterparts or algorithm-matched experimental structures are typically used as realistic references. Another related measure is the synthesizability metric, which estimates how feasible it would be to synthesize a generated structure under current laboratory conditions. However, synthesizability is difficult to standardize due to the scarcity of large-scale, high-throughput synthesis validation. As a result, energy-based metrics, described hereafter, often provide a more robust alternative.

The *stability metrics* only appeared much later, making comparison with prior works difficult. These metrics rely on the evaluation of the thermodynamic and/or dynamic stabilities of the generated structures. Commonly, thermodynamic stability (with respect to competing phases) is represented by the distance to the convex hull. Usually, a structure is considered to be (meta-)stable when it lies within a threshold (e.g., 100 meV/atom) above the hull. For structures to be dynamically stable, the harmonic phonons across the Brillouin zone must exhibit real frequencies. Due to the computational expense of phonon calculations via DFT, dynamic stability checks are applied only in select cases, e.g., for benchmarking of the WyCryst model[84]. Recently, with the improving accuracy of universal machine learning interatomic potentials (uMLIPs)[117,118], some researchers have explored the possibility of replacing DFT with uMLIPs in stability checks[45]. Recent work by Szymanski and Bartel[119] further highlighted that stability outcomes are highly sensitive to the chosen energy threshold (e.g., 0 vs. 100 meV/atom above the convex hull) and advocated for explicit baselines-such as random enumeration and template-based generation-together with low-cost uMLIP post-filtering to improve stability yields efficiently. Nevertheless, there remains neither a universal threshold nor a standardized reference convex hull across benchmarking studies of generative models. As a result, comparisons based on stability metrics are often not straightforward and should be interpreted with caution.

The *uniqueness metric* measures the ability of the model to generate diverse candidates. However, diversity is difficult to quantify. In practice, the uniqueness of the samples is used instead. A generated structure is considered unique if it does not duplicate any other structure generated in the *same batch*. Another metric, the *novelty metric*, measures how many structures are new compared to the training set. Unlike uniqueness, which refers to whether generated samples duplicate among themselves, this metric refers to whether a generated structure is different from a fixed reference (usually the training) dataset. When compared against all reported structures, novelty indicates a new *phase* for a given composition. However, novelty with respect to known structures should be interpreted carefully, as structural matching algorithms can sometimes fail or erroneously distinguish ordered configurations from their disordered counterparts. This issue has been highlighted in several studies[120–122], so extra caution is advised when claiming "new compounds" or "new phases."

A similar metric is the *rediscovery rate*, which measures the percentage of the test set being recovered by the model. For these metrics, the algorithm used to judge the match between structures plays a key role. For example, some algorithms treat different ordered configurations for a disordered alloy as unique structures. Therefore, it is crucial to account for differences in algorithms when comparing model performance. Moreover, these three metrics are functions of the generated batch size and the reference datasets. With increasing batch size, the novelty and uniqueness ratios usually drop while the recovery rate increases. Nevertheless, within the same batch, a model might generate each novel structure multiple times, resulting in high novelty but low uniqueness. A successful model should be able to generate stable, unique, and novel structures. Therefore, these three metrics are often combined as the *S.U.N. metric*[45].

In some studies, particular attention is given to *prototypes*, either by classifying generated structures into prototype families or by identifying novel prototypes. A prototype is defined as a class of crystal structures that share identical crystallographic characteristics, independent of chemical composition. This classification typically considers stoichiometry, space group, and occupied Wyckoff positions, although the exact criteria may vary depending on context. Prototype-based metrics enable the evaluation of a model's ability to explore new structural motifs, rather than merely producing chemical variations of already known structures.

The *efficiency metric* represents a critical practical consideration for real-world deployment of generative models. Efficiency encompasses multiple dimensions, including computational training cost, data efficiency, scalability, etc. Recent studies reveal significant disparities in resource consumption across different models, with training duration varying dramatically depending on the model architecture and dataset size. For instance, TGDMat required only 500 epochs to train on Perov-5 and Carbon-24 datasets, compared to more than 3000 epochs for CDVAE and DiffCSP[69]. Some approaches obviate the computational costs for curating data that, in some cases, are several orders of magnitude larger than the training costs. The scaling behavior becomes particularly important when models are applied to larger chemical spaces or high-throughput screening scenarios, where thousands or millions of candidate structures need to be generated. However, most benchmarks focus exclusively on generation quality while leaving efficiency metrics overlooked.

We note that the metrics discussed above provide a robust and transferable suite of evaluation tools, primarily designed for unconditional generation, i.e., de novo materials design. This setting also benefits from *diversity metrics*, which quantify the variety of generated elements, crystal symmetries, stoichiometries, and structural scales (e.g., atoms per cell, density). Such measures reflect a model's capacity for broad and unbiased exploration of materials space. In contrast, conditional generation tasks (e.g., constrained by chemical space, composition, prototype, or target property) often require more task-specific evaluation criteria. For CSP, top-(k)matching metrics are particularly relevant, measuring the fraction of compositions for which the correct ground-state structure appears within the top (k) generated candidates, together with the corresponding RMSD error. For property-conditioned generation, additional validation against the intended target property is essential. Finally, application-specific criteria, such as supply-risk indicators (e.g., the Herfindahl-Hirschman Index, HHI), may be incorporated when discovery goals include compositional, economic, or sustainability constraints.

We conclude this section by emphasizing the lack of unified standards for evaluating generative materials models. Firstly, early models were

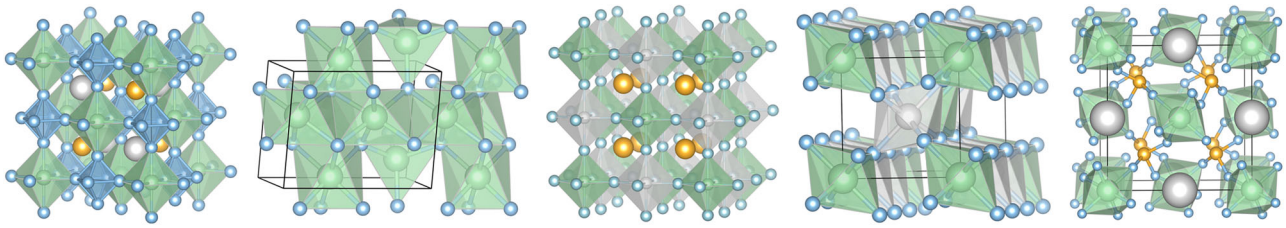

Fig. 4 | Examples of generated crystal structures. a cubicGAN, **b** iMatGen, **c** CrystaLLM, **d** MatterGen, and **e** Matra-Genoa models, replotted from the CIF files in refs. 123,[58],[95],[45], and [46], respectively using VESTA[145].

constrained by limited train and test datasets available at the time, such as Perov-5 or Carbon-24, which were restricted to specific structure motifs or chemical compositions and, as previously discussed, have some serious limitations with random test splits[55]. While recent models leverage substantially larger datasets, as shown in Table 1, test splits are still not standard.

Secondly, additional factors such as matching algorithms and reference convex hulls vary significantly across the literature, as previously stated. Even when studies employ identical reference datasets and algorithms, they frequently apply different evaluation criteria and thresholds, severely hampering cross-study comparisons. A particularly concerning issue is that many models rely exclusively on formation energies for stability metrics, either lacking proper convex hull analysis for competing phases or disregarding it entirely. Additionally, matching algorithms might erroneously find "new phases," as discussed previously for the novelty metric.

Thirdly, while validity metrics and recovery rates are commonly treated as key performance indicators, simplistic metrics such as "machine readability of generated structure files" or "successful termination of DFT calculations" are insufficient for meaningful evaluation. The fundamental balance between validity (which can reflect memorization) and creativity (which enables exploration of chemical space) deserves greater attention in benchmarking frameworks. Current approaches often prioritize structural correctness without adequately assessing genuine materials discovery potential.

Last but not least, the realism metrics, particularly synthesizability assessments, often suffer from bias present in reference datasets—a limitation that remains poorly discussed in the literature. These biases can systematically skew evaluation results and misrepresent model capabilities. All these interconnected aspects highlight the critical need for comprehensive, multi-criteria benchmarking standards.

For these reasons, we do not attempt to directly compare or benchmark models in this review. The absence of standardized evaluation metrics makes such comparisons unreliable, highlighting the need for universal benchmarking test sets. Without such standards, meaningful cross-model comparisons remain fundamentally compromised.

Applications

Beyond benchmarking, the real value of generative models in materials science lies in their ability to discover new, stable compounds. Beyond simply reproducing known data, these models can explore novel chemistries, prototypes, and topologies, as shown by the generated structure for several models in Fig. 4. This section reviews applications in two stages. First, *large-scale, unconstrained screening*, where models generate vast libraries of candidate structures across broad structural classes. Second, *targeted, property-driven generation*, where models are steered toward specific functional objectives, such as battery performance, superconductivity, or optoelectronic properties.

Large-scale screening of crystal structures

The CubicGAN model[123] is an early example of large-scale generative screening in materials science. Trained to produce structures in three cubic space groups (#216, #225, and #221), it generated 10 million candidates—recovering most known cubic crystals from MP and ICSD and identified 24 novel prototypes (defined by stoichiometry and space group, ignoring the Wyckoff positions). Filtered by (i) CIF validity, (ii) composition uniqueness, (iii) charge neutrality, and (iv) negative formation energy, the model produced 17,303 ternary and 91,594 quaternary materials. DFT optimization succeeded for 83.4% (14,433) of ternaries and 24.5% (22,414) of quaternaries. Mechanical and dynamical stability screening yielded 506 stable structures, including four new prototypes. The method generalized beyond cubic symmetry[83], produced 1869 DFT-optimized non-cubic structures, ~5.3% of which lay within 250 meV/atom of the MP convex hull.

Variants of the CDVAE model[52] have been applied to low-dimensional materials generation. Lyngby et al.[124] trained on 2615 2D structures from the C2DB database[125] and generated 10,000 candidates, 89% of which passed geometric and charge neutrality checks. After deduplication and elemental substitution augmentation, systematic DFT calculations showed that the generated set has similar formation energies to the training set but with distinct compositions and structures. The approach was later extended to 1D materials[126], produced 1895 candidates, of which 1008 had a value of E_{hull} less than 200 meV/atom (based on DFT calculations with the OQMD dataset as the hull reference[48]). This set also included completely new prototypes.

MatterGen, trained on 607,683 structures from MP and Alexandria, further showed the ability of generative models to find S.U.N. structures. Interestingly, the uniqueness declined from 100% for 1000 samples to 52% for 10 million, while novelty stayed near 60% up to 1 million samples before dropping to ~ 47%. It recovered over 2000 experimental ICSD structures absent from the training set. DFT validation of 1000 samples showed RMSE < 0.076 Å between generated and optimized structures, with 75% within 100 meV/atom of the convex hull.

Generation of target functional materials

Generative models can be fine-tuned or conditioned to produce candidates meeting specific functional targets. MatterGen, for example, supports conditioning on composition, space group, and mechanical, electronic, or magnetic properties. Parida et al.[127] targeted Li-ion battery materials with $E_{\text{hull}} \leq 30$ meV/atom, and generated 32,600 candidates. Using the MatterSim interatomic potential[128] for screening, they identified 12,550 unique novel structures within 100 meV/atom, including 817 Li-compounds without heavy elements, ~11% of which met the target stability threshold.

MatterGen was also employed to generate metal oxides for thermochemical water-splitting (TCH) applications. Conditioned on $E_{\text{hull}} \leq 10$ meV/atom and 13 desired chemical systems, the model generated around 15,000 samples. Further, the samples were pre-filtered using S.U.N. checks (for stability, the E_{hull} values were also predicted by the MatterSim potential). The most notable candidate is a novel thermodynamically stable quinary oxide, $Ba_2SrInFeO_6$, that exhibits oxygen defect formation energies appropriate for TCH.

CDVAE was also used to design new conventional superconductors[129]. Trained on 1058 JARVIS structures[130], it generated 3000 candidates, which were property-screened using a pre-trained ALIGNN model[131]. While element abundance was similar to the training set, the generated set included novel stoichiometries and structures, though all belonged to space group *P*1,

showing a limitation of the CDVAE model in generating high-symmetry crystal structures.

The GANCSP model was applied to generate Mg–Mn–O ternaries for photoanode applications[63], and identified 23 new compounds with promising aqueous stability and band gaps. Two structures ($MgMn_4O_8$ and Mg_2MnO_4) were at or near the convex hull, representing stable new phases within DFT accuracy.

Recently, Okabe et al. showed that structural constraints can be integrated into any diffusion-based generative model by strategic masking of the denoising step to steer the generation[132]. Using DiffCSP trained on the MP-20 dataset as the diffusion model, the SCIGEN model generated 7.87 million compounds in Archimedean lattices. Pre-filtering DFT calculations were performed on 26,000 samples, and 13,800 materials were found to be converged to a maximum force smaller than 0.01 eV/Å.

Another example of targeting specific structural conditions can be found in ref. 133. Via conditioning the CDVAE model, the ConditionCDVAE+ model was trained on a dataset of 19,926 two-dimensional Janus III-VI van der Waals heterostructures and could generate novel heterostructures for specific chemical compositions.

Discussion and future directions

Generative AI for inorganic crystal structures has evolved from early feasibility demonstrations on narrow material classes to high-quality models that can produce symmetry-aware, stable, and sometimes property-targeted candidates. Despite this progress, much remains to be done. Below, we outline and discuss a research agenda organized around methodology, physics, control, data, and deployment.

As shown in Fig. 3, several regions of the methodological design space remain unexplored. While some regions make less sense (e.g., diffusion models or LLMs on voxel), some remain underexplored (e.g., diffusion in reciprocal space or normalizing flows on Wyckoff positions). Other directions include masked denoising, manifold-constrained diffusion, and flow-matching on crystallographic manifolds (e.g., fixed space group/Wyckoff sets), which can enforce space-group consistency, charge neutrality, and minimal interatomic distances by design, thereby reducing the fraction of invalid samples and post-hoc filtering. We also expect more hybrid approaches combining complementary architectures and representations.

On the physics side, most current models assume fully ordered crystals at $T = 0$ K with full-site occupancies. Extending representations to include partial occupancies, i.e., alloying, vacancies, and point defects, will be critical for applications in catalysis, solid electrolytes, and thermoelectrics. Addressing these non-idealities may require stepping beyond the unit cell and modeling larger-scale structures. Additional auxiliary information, such as magnetic moments and charge states, can also be integrated into structure representations. Moreover, conditioning generative models on thermodynamic variables (temperature, pressure, and chemical potentials) would allow targeting of metastable phases and entropically stabilized structures.

Beyond standard likelihood-based learning of $p(\mathbf{x})$ or $p(\mathbf{x}|\mathbf{c})$, coupling to energy surrogates $E_\phi(\mathbf{x})$ (e.g., uMLIPs or differentiable property predictors) enables Boltzmann-like likelihoods of the form $p(\mathbf{x}) \propto \exp(-\beta E_\phi(\mathbf{x}))$. This can be realized through energy-based models (EBMs), which directly parameterize $E_\theta(x)$ and rely on Markov chain Monte Carlo sampling[134,135]. Related approaches use energy as a reward signal: reinforcement learning can exploit $f(E(\mathbf{x}))$ for reward shaping, while Generative Flow Networks (GFlowNets[136]) learn policies that sample proportional to a reward function.

Score-based methods, which learn $\nabla_x \log p(\mathbf{x})$ rather than $p(\mathbf{x})$ itself, can also be connected to energy-based perspectives since the score can be derived from the energy function. Diffusion models, for example, can be interpreted as score-based generative models[31]. These generative methods are therefore conceptually related to global optimization approaches—such as simulated annealing[137] and minima hopping[8]—but are, in principle, more powerful, as they operate over high-dimensional probability distributions that encompass composition and other relevant aspects.

Further developments will also include better conditional control. Models with hard conditional guarantees (e.g., composition, charge balance, space group, prototype class) will increase success rates. Many applications also require balancing trade-offs, such as E_{hull}, band gap, ionic conductivity, critical temperature, and abundance/toxicity. Multi-objective approaches such as GFlowNets and reinforcement learning with vector-valued rewards can explicitly learn to sample across Pareto fronts rather than optimize a single scalar. Natural-language conditioning is also promising: as demonstrated by Chemeleon, LLM-conditioned generators can parse free-text prompts (e.g., "wide band gap, nontoxic, layered *p*-doped perovskite with large polaron mobility") into structured constraints, supporting interactive dialog.

On the data side, models are limited by the data they are trained on. Existing datasets are biased toward certain chemistries, prototypes, small unit cells, and GGA-level energetics. Better curation is needed to align training data with generation goals. This includes careful dataset design, clear licenses, transparent recipes, and provenance tracking for reproducibility. Online training, where new data are generated and used to fine-tune models, offers an attractive direction. Where available, attaching synthesis metadata (e.g., temperature, atmosphere, precursors, success/failure) can help bridge the gap between generative proposals and laboratory realization by enabling models to suggest not just what to make, but also how to make it.

On the evaluation side, as noted earlier, there is a lack of rigorous benchmarks. There is a need for improvement from simple sanity checks (validity, coverage, precision-recall rates) to real comparative benchmarks that include stability, novelty, and time complexity. Community-maintained suites should include frozen convex hulls to assess stability, fixed training sets with standard matching algorithms for novelty, and explicit reporting of the computational cost for scalability. Benchmarks should span conditional tasks across chemistries and properties, not only unconditional generation. Following the example of property prediction[138], incorporating calibrated uncertainties for structures and properties would enable more reliable screening and prioritization.

At scale, generative models will serve as seeds for new candidates in autonomous materials discovery. Crucially, generative AI should not be viewed as a one-shot predictor of "the best" material, but rather as a component in iterative pipelines. Fast, lightweight models are needed to propose candidates for downstream filtering by uMLIPs, high-throughput DFT, and experimental validation. Active learning loops, where models propose, labs validate, and data are recycled, will be central. Sustainability constraints—earth abundance, toxicity, embodied energy, and end-of-life—must also be built into discovery pipelines. Finally, models with interpretable outputs will allow experts to audit and steer generation, improving trust and accelerating insight.

Conclusion

Generative AI is beginning to reshape how we explore crystalline materials. By learning probability distributions from large databases, models such as VAEs, GANs, flows, diffusion, transformers, and LLMs can propose novel structures and, in some cases, target desired properties. In this review, we outlined the core architectures, representations, as well as data sources and organized them through a taxonomy of representations, conditioning strategies, and application domains. More than 50 models have been discussed, with applications spanning large-scale screening, discovery of novel phases, and early advances in energy and catalysis.

Despite these advances, generative models for crystal structures are still in an early stage of development. Benchmarks are still fragmented, stability assessments vary, and synthesizability is often overlooked. Disorder, finite-temperature effects, and multi-objective trade-offs are not yet fully addressed. Fortunately, many promising paths exist to improve them—through advances in methodology, physics, control, data, and integration into discovery pipelines. With continued development, generative models could become indispensable tools for computational materials scientists, accelerating the discovery and understanding of the whole materials landscape.

Data availability

No datasets were generated or analyzed during the current study.

Received: 3 September 2025; Accepted: 17 November 2025;
Published online: 06 December 2025

References

1. Lewis, N. S. Toward cost-effective solar energy use. *Science* **315**, 798–801 (2007).
2. Magee, C. L. Towards quantification of the role of materials innovation in overall technological development. *Complexity* **18**, 10–25 (2012).
3. Snyder, G. & Toberer, E. Complex thermoelectric materials. *Nat. Mater*. **7**, 105–114 (2008).
4. Oganov, A. R. & Glass, C. W. Crystal structure prediction using ab initio evolutionary techniques: principles and applications. *J. Chem. Phys.* **124**, 244704 (2006).
5. Podryabinkin, E. V., Tikhonov, E. V., Shapeev, A. V. & Oganov, A. R. Accelerating crystal structure prediction by machine-learning interatomic potentials with active learning. *Phys. Rev. B* **99**, 064114 (2019).
6. Pickard, C. J. & Needs, R. J. Ab initio random structure searching. *J. Phys. Condens. Matter* **23**, 053201 (2011).
7. Wang, Y., Lv, J., Zhu, L. & Ma, Y. Crystal structure prediction via particle-swarm optimization. *Phys. Rev. B* **82**, 094116 (2010).
8. Goedecker, S. Minima hopping: an efficient search method for the global minimum of the potential energy surface of complex molecular systems. *J. Chem. Phys.* **120**, 9911–9917 (2004).
9. Yamashita, T. et al. CrySPY: a crystal structure prediction tool accelerated by machine learning. *Sci. Technol. Adv. Mater. Methods* **1**, 87–97 (2021).
10. Falls, Z., Avery, P., Wang, X., Hilleke, K. P. & Zurek, E. The XtalOpt evolutionary algorithm for crystal structure prediction. *J. Phys. Chem. C* **125**, 1601–1620 (2021).
11. Handoko, A. D. & Made, R. I. Artificial intelligence and generative models for materials discovery—a review. *WS Ann. Rev. Funct. Mater*. **3**, 2540001 (2025).
12. Liu, Y. et al. Generative artificial intelligence and its applications in materials science: current situation and future perspectives. *J. Materiomics* **9**, 798–816 (2023).
13. Long, T., Zhang, Y. & Zhang, H. Generative deep learning for the inverse design of materials. Preprint at https://doi.org/10.48550/arXiv.2409.19124 (2024).
14. Noh, J., Gu, G. H., Kim, S. & Jung, Y. Machine-enabled inverse design of inorganic solid materials: promises and challenges. *Chem. Sci.* **11**, 4871–4881 (2020).
15. Park, H., Li, Z. & Walsh, A. Has generative artificial intelligence solved inverse materials design? *Matter* **7**, 2355–2367 (2024).
16. Takahara, I., Mizoguchi, T. & Liu, B. Accelerated inorganic materials design with generative AI agents. Preprint at https://doi.org/10.48550/arXiv.2504.00741 (2025).
17. Zunger, A. Inverse design in search of materials with target functionalities. *Nat. Rev. Chem.* **2**, 1–16 (2018).
18. Kingma, D. P. & Welling, M. Auto-encoding variational bayes. Preprint at https://doi.org/10.48550/arXiv.1312.6114 (2022).
19. Goodfellow, I. J. et al. Generative adversarial nets. In *Proc. Advances in Neural Information Processing Systems* Vol. 27 (eds Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. & Weinberger, K.) (Curran Associates, Inc., 2014).
20. Radford, A., Metz, L. & Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. Preprint at https://doi.org/10.48550/arXiv.1511.06434 (2016).
21. Vaswani, A. et al. Attention is all you need. In *Proc. Advances in Neural Information Processing Systems* Vol. 30 (eds Guyon, I. et al.) (Curran Associates, Inc., 2017).
22. Rezende, D. & Mohamed, S. Variational inference with normalizing flows. In *Proc. 32nd International Conference on Machine Learning* Vol. 37 of *Proceedings of Machine Learning Research* (eds Bach, F. & Blei, D.) 1530–1538 (PMLR, 2015).
23. Dinh, L., Sohl-Dickstein, J. & Bengio, S. Density estimation using real NVP. In *Proc. ICLR (Poster)* (OpenReview.net, 2017).
24. Lipman, Y., Chen, R. T. Q., Ben-Hamu, H., Nickel, M. & Le, M. Flow matching for generative modeling. Preprint at https://doi.org/10.48550/arXiv.2210.02747 (2023).
25. Dhariwal, P. & Nichol, A. Diffusion models beat GANs on image synthesis. In *Proc. Advances in Neural Information Processing Systems* Vol. 34 (eds Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P. & Vaughan, J. W.) 8780–8794 (Curran Associates, Inc., 2021).
26. Cai, R. et al. Learning gradient fields for shape generation. In *Eur. Conf. on Computer Vision* 364–381 (Springer, 2020)
27. Shi, C., Luo, S., Xu, M. & Tang, J. Learning gradient fields for molecular conformation generation. In *Proc. 38th International Conference on Machine Learning* 9558–9568 (PMLR, 2021).
28. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N. & Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proc. 32nd International Conference on Machine Learning* Vol. 37 of *Proceedings of Machine Learning Research* (eds Bach, F. & Blei, D.) 2256–2265 (PMLR, 2015).
29. Ho, J., Jain, A. & Abbeel, P. Denoising diffusion probabilistic models. In *Proc. Advances in Neural Information Processing Systems* Vol. 33, 6840–6851 (Curran Associates, Inc., 2020).
30. Song, Y. & Ermon, S. Generative modeling by estimating gradients of the data distribution. In *Proc. Advances in Neural Information Processing Systems* Vol. 32 (Curran Associates, Inc., 2019).
31. Song, Y. et al. Score-based generative modeling through stochastic differential equations. In *Proceedings of International Conference on Learning Representations* (ICLR) (2021).
32. Radford, A., Narasimhan, K., Salimans, T. & Sutskever, I. Improving language understanding by generative pre-training. Preprint at https://paperswithcode.com/paper/improving-language-understanding-by (2018).
33. Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. 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)* (2018).
34. Brown, T. et al. Language models are few-shot learners. In *Proc. Advances in Neural Information Processing Systems* (eds Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. & Lin, H.) Vol. 33, 1877–1901 (Curran Associates, Inc., 2020).
35. Quiroga, F. M., Ronchetti, F., Lanzarini, L. & Fernandez-Bariviera, A. Revisiting data augmentation for rotational invariance in convolutional neural networks. In *Modelling and Simulation in Management Sciences: Proceedings of the International Conference on Modelling and Simulation in Management Sciences (MS-18)*, 127–141 (Springer, 2020).
36. Mazitov, A. et al. PET-MAD, a lightweight universal interatomic potential for advanced materials modeling. Preprint at https://doi.org/10.48550/arXiv.2503.14118 (2025).
37. Gong, S. et al. Examining graph neural networks for crystal structures: limitations and opportunities for capturing periodicity. *Sci. Adv.* **9**, eadi3245 (2023).
38. Lehtola, S., Steigemann, C., Oliveira, M. J. & Marques, M. A. Recent developments in LIBXC—a comprehensive library of functionals for density functional theory. *SoftwareX* **7**, 1–5 (2018).
39. Perdew, J. P., Burke, K. & Ernzerhof, M. Generalized gradient approximation made simple. *Phys. Rev. Lett.* **77**, 3865–3868 (1996).
40. Jain, A. et al. The Materials Project: a materials genome approach to accelerating materials innovation. *APL Mater.* **1**, 011002 (2013).

41. Belsky, A., Hellenbrandt, M., Karen, V. L. & Luksch, P. New developments in the Inorganic Crystal Structure Database (ICSD): accessibility in support of materials research and design. *Acta Crystallogr. Sect. B Struct. Sci*. **58**, 364–369 (2002).
42. Schmidt, J. et al. Machine-learning-assisted determination of the global zero-temperature phase diagram of materials. *Adv. Mater.* **35**, 2210788 (2023).
43. Wang, H.-C., Schmidt, J., Marques, M. A. L., Wirtz, L. & Romero, A. H. Symmetry-based computational search for novel binary and ternary 2D materials. *2D Mater.* **10**, 035007 (2023).
44. Kresse, G. & Furthmüller, J. Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set. *Phys. Rev. B* **54**, 11169–11186 (1996).
45. Zeni, C. et al. A generative model for inorganic materials design. *Nature* **639**, 624–632 (2025).
46. Breuck, P.-P. D., Piracha, H. A., Rignanese, G.-M. & Marques, M. A. L. A generative material transformer using Wyckoff representation. Preprint at https://doi.org/10.48550/arXiv.2501.16051 (2025).
47. Curtarolo, S. et al. AFLOWLIB.ORG: a distributed materials properties repository from high-throughput ab initio calculations. *Comput. Mater. Sci.* **58**, 227–235 (2012).
48. Saal, J. E., Kirklin, S., Aykol, M., Meredig, B. & Wolverton, C. Materials design and discovery with high-throughput density functional theory: the Open Quantum Materials Database (OQMD). *JOM* **65**, 1501–1509 (2013).
49. Choudhary, K. et al. The joint automated repository for various integrated simulations (JARVIS) for data-driven materials design. *npj Comput. Mater.* **6**, 173 (2020).
50. Andersen, C. W. et al. OPTIMADE, an API for exchanging materials data. *Sci. Data* **8**. https://doi.org/10.1038/s41597-021-00974-z (2021).
51. Evans, M. L. et al. Developments and applications of the OPTIMADE API for materials discovery, design, and data exchange. *Digit. Discov.* **3**, 1509–1533 (2024).
52. Xie, T., Fu, X., Ganea, O.-E., Barzilay, R. & Jaakkola, T. Crystal diffusion variational autoencoder for periodic material generation. In *Proc. International Conference on Learning Representations (ICLR)* (2022).
53. Castelli, I. E. et al. Computational screening of perovskite metal oxides for optimal solar light capture. *Energy Environ. Sci.* **5**, 5814–5819 (2012).
54. Castelli, I. E. et al. New cubic perovskites for one- and two-photon water splitting using the computational materials repository. *Energy Environ. Sci.* **5**, 9034–9043 (2012).
55. Martirossyan, M. M. et al. All that structure matches does not glitter. Preprint at https://doi.org/10.48550/arXiv.2509.12178 (2025).
56. Ren, Z. et al. An invertible crystallographic representation for general inverse design of inorganic crystals with targeted properties. *Matter* **5**, 314–335 (2022).
57. Hoffmann, J. et al. Data-driven approach to encoding and decoding 3-D crystal structures. Preprint at https://doi.org/10.48550/arXiv.1909.00949 (2019).
58. Noh, J. et al. Inverse design of solid-state materials via a continuous representation. *Matter* **1**, 1370–1384 (2019).
59. Court, C. J., Yildirim, B., Jain, A. & Cole, J. M. 3-D inorganic crystal structure generation and property prediction via representation learning. *J. Chem. Inf. Model.* **60**, 4518–4535 (2020).
60. Long, T. et al. Constrained crystals deep convolutional generative adversarial network for the inverse design of crystal structures. *npj Comput. Mater.* **7**, 1–7 (2021).
61. Long, T. et al. Inverse design of crystal structures for multicomponent systems. *Acta Mater.* **231**, 117898 (2022).
62. Nouira, A., Sokolovska, N. & Crivello, J.-C. CrystalGAN: learning to discover crystallographic structures with generative adversarial networks. Preprint at https://doi.org/10.48550/arXiv.1810.11203 (2019).
63. Kim, S., Noh, J., Gu, G. H., Aspuru-Guzik, A. & Jung, Y. Generative adversarial networks for crystal structure prediction. *ACS Cent. Sci.* **6**, 1412–1420 (2020).
64. Lu, S. et al. Uni-3DAR: unified 3D generation and understanding via autoregression on compressed spatial tokens. Preprint at https://doi.org/10.48550/arXiv.2503.16278 (2025).
65. Qi, H. et al. Latent conservative objective models for data-driven crystal structure prediction. Preprint at https://doi.org/10.48550/arXiv.2310.10056 (2023).
66. Luo, Y., Liu, C. & Ji, S. Towards symmetry-aware generation of periodic materials. In *Proc. Advances in Neural Information Processing Systems* Vol. 36 (2024).
67. Klipfel, A., Fregier, Y., Sayede, A. & Bouraoui, Z. Vector field oriented diffusion model for crystal material generation. In *Proceedings of the AAAI Conference on Artificial Intelligence,* Vol. 38 (2024)
68. Pakornchote, T. et al. Diffusion probabilistic models enhance variational autoencoder for crystal structure generative modeling. *Sci. Rep.* **14**, 1275 (2024).
69. Jiao, R. et al. Crystal structure prediction by joint equivariant diffusion. In *Proc. Thirty-seventh Conference on Neural Information Processing Systems (NeurIPS)* (2023).
70. Jiao, R., Huang, W., Liu, Y., Zhao, D. & Liu, Y. Space group constrained crystal generation. Preprint at https://doi.org/10.48550/arXiv.2402.03992 (2024).
71. Ye, C.-Y., Weng, H.-M. & Wu, Q.-S. Con-CDVAE: a method for the conditional generation of crystal structures. *Compt. Mater. Today* **1**, 100003 (2024).
72. Luo, X. et al. Deep learning generative model for crystal structure prediction. *npj Comput. Mater.* **10**, 1–10 (2024).
73. Liu, Y. et al. Equivariant hypergraph diffusion for crystal structure prediction. Preprint at https://doi.org/10.48550/arXiv.2501.18850 (2025).
74. Chen, Z. et al. Transformer-enhanced variational autoencoder for crystal structure prediction. Preprint at https://doi.org/10.48550/arXiv.2502.09423 (2025).
75. Das, K. et al. Periodic materials generation using text-guided joint diffusion model. In *The Thirteenth18449 International Conference on Learning Representations (ICLR)* (2025).
76. Park, H., Onwuli, A. & Walsh, A. Exploration of crystal chemical space using text-guided generative artificial intelligence. *Nat. Commun.* **16**, 4379 (2025).
77. Wu, L. et al. Siamese foundation models for crystal structure prediction. Preprint at https://doi.org/10.48550/arXiv.2503.10471 (2025).
78. Zhong, P., Dai, X., Deng, B., Ceder, G. & Persson, K. A. Crystal structure prediction with host-guided inpainting generation and foundation potentials. *Mater. Horiz.* https://doi.org/10.1039/D5MH00774G (2025).
79. Tangsongcharoen, K. et al. CrystalGRW: generative modeling of crystal structures with targeted properties via geodesic random walks. Preprint at https://doi.org/10.48550/arXiv.2501.08998 (2025).
80. Cornet, F. et al. Kinetic Langevin diffusion for crystalline materials generation. Preprint at https://doi.org/10.48550/arXiv.2507.03602 (2025).
81. Yang, S. et al. Scalable diffusion for materials generation. In *Proc. International Conference on Learning Representations (ICLR*) (2024).
82. Zhang, G. et al. UniGenX: unified generation of sequence and structure with autoregressive diffusion. Preprint at https://doi.org/10.48550/arXiv.2503.06687 (2025).
83. Zhao, Y. et al. Physics guided deep learning for generative design of crystal materials with symmetry constraints. *npj Comput. Mater.* **9**, 1–12 (2023).

84. Zhu, R., Nong, W., Yamazaki, S. & Hippalgaonkar, K. WyCryst: Wyckoff inorganic crystal generator framework. *Matter* **7**, 3469–3488 (2024).
85. Fredericks, S., Parrish, K., Sayre, D. & Zhu, Q. PyXtal: a Python library for crystal structure generation and symmetry analysis. *Comput. Phys. Commun.* **261**, 107810 (2021).
86. Su, T., Cao, B., Hu, S., Li, M. & Zhang, T.-Y. CGWGAN: crystal generative framework based on Wyckoff generative adversarial network. *J. Mater. Inf.* **4**, N/A–N/A (2024).
87. Chen, C. & Ong, S. P. A universal graph deep learning interatomic potential for the periodic table. *Nat. Comput. Sci.* **2**, 718–728 (2022).
88. Cao, Z., Luo, X., Lv, J. & Wang, L. Space group informed transformer for crystalline materials generation. *Sci. Bull.* **70**, 3522–3533 (2025).
89. Levy, D. et al. SymmCD: symmetry-preserving crystal generation with diffusion models. Preprint at https://doi.org/10.48550/arXiv.2502.03638 (2025).
90. Kelvinius, F. E. et al. WyckoffDiff—a generative diffusion model for crystal symmetry. Preprint at https://doi.org/10.48550/arXiv.2502.06485 (2025).
91. Flam-Shepherd, D. & Aspuru-Guzik, A. Language models can generate molecules, materials, and protein binding sites directly in three dimensions as XYZ, CIF, and PDB files. Preprint at https://doi.org/10.48550/arXiv.2305.05708 (2023).
92. Gruver, N. et al. Fine-tuned language models generate stable inorganic materials as text. In *Proc. 12th Int. Conf. Learn. Represent. (ICLR)* (2024).
93. Touvron, H. et al. Llama 2: open foundation and fine-tuned chat models. Preprint at https://doi.org/10.48550/arXiv.2307.09288 (2023).
94. Touvron, H. et al. LLaMA: open and efficient foundation language models. Preprint at https://doi.org/10.48550/arXiv.2302.13971 (2023).
95. Antunes, L. M., Butler, K. T. & Grau-Crespo, R. Crystal structure generation with autoregressive large language modeling. *Nat. Commun.* **15**, 10570 (2024).
96. Mohanty, T., Mehta, M., Sayeed, H. M., Srikumar, V. & Sparks, T. D. CrysText: a generative AI approach for text-conditioned crystal structure generation using LLM. Preprint at https://doi.org/10.26434/chemrxiv-2024-gjhpq-v2 (2024).
97. Grattafiori, A. et al. The Llama 3 herd of models. Preprint at https://doi.org/10.48550/arXiv.2407.21783 (2024).
98. Dettmers, T., Pagnoni, A., Holtzman, A. & Zettlemoyer, L. QLoRA: efficient finetuning of quantized LLMs. Preprint at https://doi.org/10.48550/arXiv.2305.14314 (2023).
99. Yang, S. et al. Generative hierarchical materials search. In *Advances in Neural Information Processing Systems, 37 (NeurIPS)* 2024.
100. Ding, Q., Miret, S. & Liu, B. MatExpert: decomposing materials discovery by mimicking human experts. Preprint at https://doi.org/10.48550/arXiv.2410.21317 (2024).
101. Gan, J. et al. MatLLMSearch: Crystal Structure Discovery with Evolution-Guided Large Language Models. Preprint at https://doi.org/10.48550/arXiv.2502.20933 (2025).
102. Yan, K. et al. Invariant tokenization of crystalline materials for language model enabled generation. Preprint at https://doi.org/10.48550/arXiv.2503.00152 (2025).
103. Xia, Y. et al. Nature language model: deciphering the language of nature for scientific discovery. Preprint at https://doi.org/10.48550/arXiv.2502.07527 (2025).
104. Miller, B. K., Chen, R. T. Q., Sriram, A. & Wood, B. M. FlowMM: generating materials with Riemannian flow matching. Preprint at https://doi.org/10.48550/arXiv.2406.04713 (2024).
105. Sriram, A., Miller, B. K., Chen, R. T. Q. & Wood, B. M. FlowLLM: flow matching for material generation with large language models as base distributions. Preprint at https://doi.org/10.48550/arXiv.2410.23405 (2024).
106. Wu, H. et al. A periodic Bayesian flow for material generation. Preprint at https://doi.org/10.48550/arXiv.2502.02016 (2025).
107. Bengio, E., Jain, M., Korablyov, M., Precup, D. & Bengio, Y. Flow network based generative models for non-iterative diverse candidate generation. *Adv. Neural Inf. Process. Syst.* **34**, 27381–27394 (2021).
108. AI4Science, M. et al. Crystal-GFN: sampling crystals with desirable properties and constraints. Preprint at https://doi.org/10.48550/arXiv.2310.04925 (2023).
109. NeurIPS hierarchical GFlowNet for crystal structure generation. https://neurips.cc/virtual/2023/78549 (2023).
110. Xiao, H. et al. An invertible, invariant crystal representation for inverse design of solid-state materials using generative deep learning. *Nat. Commun.* **14**, 7027 (2023).
111. Novitskiy, L. et al. Unleashing the power of novel conditional generative approaches for new materials discovery. Preprint at https://doi.org/10.48550/arXiv.2411.03156 (2024).
112. Alverson, M. et al. Generative adversarial networks and diffusion models in material discovery. *Digit. Discov.* **3**, 62–80 (2024).
113. Li, Z. & Birbilis, N. NSGAN: A non-dominant sorting optimisation-based generative adversarial design framework for alloy discovery. *npj Comput. Mater.* **10**, 1–15 (2024).
114. Zhang, Y., Hu, C. & Jiang, B. Embedded atom neural network potentials: efficient and accurate machine learning with a physically inspired representation. *J. Phys. Chem. Lett.* **10**, 4962–4967 (2019).
115. Qin, C. et al. Inverse design of semiconductor materials with deep generative models. *J. Mater. Chem. A* **12**, 22689–22702 (2024).
116. Hoellmer, P. et al. Open materials generation with stochastic interpolants. Preprint at https://doi.org/10.48550/arXiv.2502.02582 (2025).
117. Riebesell, J. et al. A framework to evaluate machine learning crystal stability predictions. *Nat. Mach. Intell.* **7**, 836–847 (2025).
118. Loew, A., Sun, D., Wang, H.-C., Botti, S. & Marques, M. A. L. Universal machine learning interatomic potentials are ready for phonons. *npj Comput. Mater.* **11**, https://doi.org/10.1038/s41524-025-01650-1 (2025).
119. Szymanski, N. J. & Bartel, C. J. Establishing baselines for generative discovery of inorganic crystals. *Mater. Horiz.* **12**, 8000–8011 (2025).
120. Cheetham, A. K. & Seshadri, R. Artificial intelligence driving materials discovery? Perspective on the article: Scaling deep learning for materials discovery. *Chem. Mater.* **36**, 3490–3495 (2024).
121. Leeman, J. et al. Challenges in high-throughput inorganic materials prediction and autonomous synthesis. *PRX Energy* **3**, https://doi.org/10.1103/PRXEnergy.3.011002 (2024).
122. Juelsholt, M. Continued challenges in high-throughput materials predictions: mattergen predicts compounds from the training dataset. Preprint at *ChemRxiv* https://doi.org/10.26434/chemrxiv-2025-mkls8 (2025).
123. Zhao, Y. et al. High-throughput discovery of novel cubic crystal materials using deep generative neural networks. *Adv. Sci.* **8**, 2100566 (2021).
124. Lyngby, P. & Thygesen, K. S. Data-driven discovery of 2D materials by deep generative models. *npj Comput. Mater.* **8**, https://doi.org/10.1038/s41524-022-00923-3 (2022).
125. Gjerding, M. N. et al. Recent progress of the computational 2D materials database (C2DB). *2D Mater.* **8**, 044002 (2021).
126. Moustafa, H., Lyngby, P. M., Mortensen, J. J., Thygesen, K. S. & Jacobsen, K. W. Hundreds of new, stable, one-dimensional materials from a generative machine learning model. *Phys. Rev. Mater.* **7**, https://doi.org/10.1103/PhysRevMaterials.7.014007 (2023).
127. Parida, C., Roy, D., Lastra, J. M. G. & Bhowmik, A. Mining chemical space with generative models for battery materials. Preprint at https://doi.org/10.26434/chemrxiv-2025-q48jr (2025).

128. Yang, H. et al. MatterSim: a deep learning atomistic model across elements, temperatures and pressures. Preprint at https://doi.org/10.48550/arXiv.2405.04967 (2024).
129. Wines, D., Xie, T. & Choudhary, K. Inverse design of next-generation superconductors using data-driven deep generative models. *J. Phys. Chem. Lett.* **14**, 6630–6638 (2023).
130. Choudhary, K. et al. The joint automated repository for various integrated simulations (Jarvis) for data-driven materials design. *npj Comput. Mater*. **6**, https://doi.org/10.1038/s41524-020-00440-1 (2020).
131. Choudhary, K. & DeCost, B. Atomistic line graph neural network for improved materials property predictions. *npj Comput. Mater*. **7**, https://doi.org/10.1038/s41524-021-00650-1 (2021).
132. Okabe, R. et al. Structural constraint integration in a generative model for the discovery of quantum materials. *Nat. Mater.* https://doi.org/10.1038/s41563-025-02355-y (2025).
133. Gao, S. et al. Deep generative model for the inverse design of van der waals heterostructures. *Sci. Rep*. **15**, https://doi.org/10.1038/s41598-025-06432-9 (2025).
134. Lecun, Y., Chopra, S., Hadsell, R., Ranzato, M. A. & Huang, F. J. A tutorial on energy-based learning. in *Predicting Structured Data* (eds Bakir, G., Hofman, T., Scholkopt, B., Smola, A. & Taskar, B.) (MIT Press, 2006).
135. Du, Y. & Mordatch, I. Implicit generation and modeling with energy based models. In *Proc. Advances in Neural Information Processing Systems* Vol. 32 (eds Wallach, H. et al.) (Curran Associates, Inc., 2019).
136. Bengio, E., Jain, M., Korablyov, M., Precup, D. & Bengio, Y. Flow network based generative models for non-iterative diverse candidate generation. In *Proc. Advances in Neural Information Processing Systems* Vol. 34, 27381–27394 (Curran Associates, Inc., 2021).
137. Kirkpatrick, S., Gelatt, C. D. & Vecchi, M. P. Optimization by simulated annealing. *Science* **220**, 671–680 (1983).
138. De Breuck, P.-P., Evans, M. L. & Rignanese, G.-M. Robust model benchmarking and bias-imbalance in data-driven materials science: A case study on MODNet. *J. Phys.: Condens. Matter* **33**, 404002 (2021).
139. Sawada, Y., Morikawa, K. & Fujii, M. Study of deep generative models for inorganic chemical compositions. Preprint at https://doi.org/10.48550/arXiv.1910.11499 (2019).
140. Dan, Y. et al. Generative adversarial networks (GAN) based efficient sampling of chemical composition space for inverse design of inorganic materials. *npj Comput. Mater.* **6**, 1–7 (2020).
141. Liu, K., Gao, S., Yang, K. & Han, Y. PCVAE: a physics-informed neural network for determining the symmetry and geometry of crystals. In *Proc. 2023 International Joint Conference on Neural Networks (IJCNN)* 1–8 (IEEE, 2023).
142. Sinha, A., Jia, S. & Fung, V. Representation-space diffusion models for generating periodic materials. Preprint at https://doi.org/10.48550/arXiv.2408.07213 (2024).
143. Chenebuah, E. T., Nganbe, M. & Tchagang, A. B. A deep generative modeling architecture for designing lattice-constrained perovskite materials. *npj Comput. Mater.* **10**, 1–21 (2024).
144. Luo, X. et al. CrystalFlow: a flow-based generative model for crystalline materials. *Nat. Commun.* **16**, 9267 (2025).
145. Momma, K. & Izumi, F. VESTA 3 for three-dimensional visualization of crystal, volumetric and morphology data. *J. Appl. Crystallogr.* **44**, 1272–1276 (2011).

Author contributions

P.-P.D.B.: conceptualization, analysis, taxonomy, visualization, writing—original draft. H.-C.W.: metrics, applications, analysis. G.-M.R.: writing—review and editing. S.B.: supervision, funding acquisition, writing—review and editing. M.A.L.M.: supervision, funding acquisition, writing—review and editing.

Competing interests

S. Botti serves as an Associate Editor of npj Comput. Mater. but was not involved in the journal's review or decision process for this manuscript.

Additional information

Correspondence and requests for materials should be addressed to Pierre-Paul De Breuck.