

Cite this: DOI: 00.0000/xxxxxxxxxx

CryinGAN: Design and evaluation of point-cloud-based generative adversarial networks using disordered materials – application to $\text{Li}_3\text{ScCl}_6\text{-LiCoO}_2$ battery interfaces

Adrian Xiao Bin Yong,^{*ac} and Elif Ertekin^{*bc}Received Date
Accepted Date

DOI: 00.0000/xxxxxxxxxx

Generative models have received significant attention in recent years for materials science applications, particularly in the area of inverse design for materials discovery. While current efforts have mainly focused on bulk materials with relatively small unit cells, the possibility of generative models for more complex, disordered materials would extent the current capabilities of generative modeling to a significantly wider range of materials. Generative models would also benefit from better ways to assess their performance. They are typically evaluated on the new, unverified materials being generated, which give limited metrics to evaluate how well the model has learnt from the training data. In this work, we design and evaluate models intended for disordered interface structure generation. Using a disordered $\text{Li}_3\text{ScCl}_6\text{-LiCoO}_2$ battery interface as a model disordered material interface, we tested different point-cloud-based generative adversarial network architectures that further include bond distance information in the discriminator, rather than only atomic coordinates. By working with a fixed material system, we evaluated the model performance through direct comparisons between training and generated structures. PointNet-based 1D convolutions, while lacking explicit knowledge of each atom's neighbors, have good sampling sensitivity and resulted in stable training. On the other hand, graph convolutions led to discriminators with overly high sampling sensitivity, such that the models struggled to train useful generators. The best performing architecture, Crystal Interface Generative Adversarial Network (CryinGAN), uses two separate 1D convolutional discriminators, one that accepts coordinates and another that accepts bond distances as input. We demonstrate that CryinGAN is able to successfully generate low-interface-energy structures for systems with >250 atoms, in which the generated interfaces are structurally similar to the training structures. Local coordination motif analysis of the disordered region revealed that the generated structures exhibited the low-energy features of the training structures. This study highlights the capabilities of a relatively simple generative model in generating large disordered materials, and discusses the limitations of the point cloud representation. Insights are provided to help guide the development of future generative models that are useful to not just disordered, but also ordered materials.

1 Introduction

Generative modeling has emerged as a powerful tool for tackling problems in materials science^{1,2}. Initially limited to simpler molecules^{3,4} and proteins⁵, generative modeling has since advanced to include inorganic materials^{6–8} as well. The main interest involving generative modeling has been its potential for the in-

verse design of materials^{9–11}, in which the ultimate goal is to generate new materials with targeted properties, instead of screening known materials for desired properties. Generative models are distinguished from discriminative models, as the latter learns the conditional probability $p(y | x)$ of observing a property (y) given a material representation (x). Instead, a generative model learns the joint probability distribution $p(x, y)$ of the data that it was trained on, and samples from the distribution of structures.

Most generative modeling efforts for inorganic materials^{9,11,12} have been focused on simpler bulk crystals, comparatively less attention has been given to disordered systems, even though disordered materials are relevant to a wide range of applications^{13–15}. Disordered systems usually have complex and irregular structures, necessitating large atomic representations and requiring

^a Department of Materials Science and Engineering, University of Illinois Urbana-Champaign, Urbana, Illinois, USA. E-mail: axyong2@illinois.edu

^b Department of Mechanical Science and Engineering, University of Illinois Urbana-Champaign, Urbana, Illinois, USA. Email: ertekin@illinois.edu

^c Materials Research Laboratory, University of Illinois Urbana-Champaign, Urbana, Illinois, USA.

† Electronic Supplementary Information (ESI) available: [details of any supplementary information available should be included here]. See DOI: 00.0000/00000000.

more powerful generative models than those developed for simple crystals. They include structures that completely lack crystal lattices such as amorphous materials, as well as structures beyond bulk materials such as surfaces and interfaces (refer to Fig. 1a). In direct physical modeling, disordered materials are most typically represented by large so-called “supercells” that (spuriously) introduce periodicity at larger length scales. Databases of disordered materials are growing^{16–18} and they present exciting opportunities for the inverse design of metal-organic frameworks, porous amorphous materials, amorphous battery materials, etc. Beyond materials discovery, generative modeling can be used to generate amorphous structures of arbitrarily large sizes upon training on smaller samples that capture the correlation length of the material¹⁹. This capability allows for better studies of properties that are sensitive to size effects such as thermal conductivity and mechanical properties. Generative modeling can also be used to optimize atomic structures to match experimental observations²⁰, in which the structures being optimized are often disordered^{21,22}. However, the capabilities of generative modeling for disordered systems remain largely limited (e.g., generation of 2D morphology¹⁹ instead of actual atomic structures), and generative models have been reported to fail for large systems^{11,23}. To reap the benefits of generative modeling for disordered systems, better generative models need to be developed and tested on disordered systems.

When building a generative model, two major design decisions are the type of generative model and the material representation (i.e., the input used to describe the material). The compatibility of these two choices is important as well. The types of generative models that have been used for materials include variational autoencoders (VAEs)^{6,7,9,11}, generative adversarial networks (GANs)^{8,10,24,25}, diffusion models^{26–29}, and language models^{30–32}. Generative models were initially developed with two main types of material representations: (1) voxels^{6,7,11,12} and (2) point clouds^{8–10,25}. Voxel representation treats cells as voxelized density maps. New structures are generated as density maps that are subsequently reconstructed back into atomic coordinates. One disadvantage of this representation is that it is memory intensive, resulting in limitations in the voxelization resolution and thus number of atoms (e.g., Court et al.¹¹ restricted the number of atoms to < 40 atoms per cell). Reconstruction issues were also reported¹¹ for non-cubic cells. These considerations make voxel representations less suitable for disordered materials. On the other hand, point clouds directly represent structures using their atomic coordinates and lattice parameters, circumventing the memory and reconstruction issues of the voxel representation. The simple representation makes them a highly scalable and better choice for disordered materials. However, the design of point cloud architectures that are symmetry-invariant is not trivial, as the commonly used PointNet architecture³³ does not include the desired symmetry invariances. More recently, other representations such as graphs^{26,29}, coordinate-based representations (e.g., UniMat²⁸, CrysTens²⁷), and text-based representations^{30–32} have also been explored.

To compare different generative models and make design choices, it is necessary to sufficiently evaluate the generated struc-

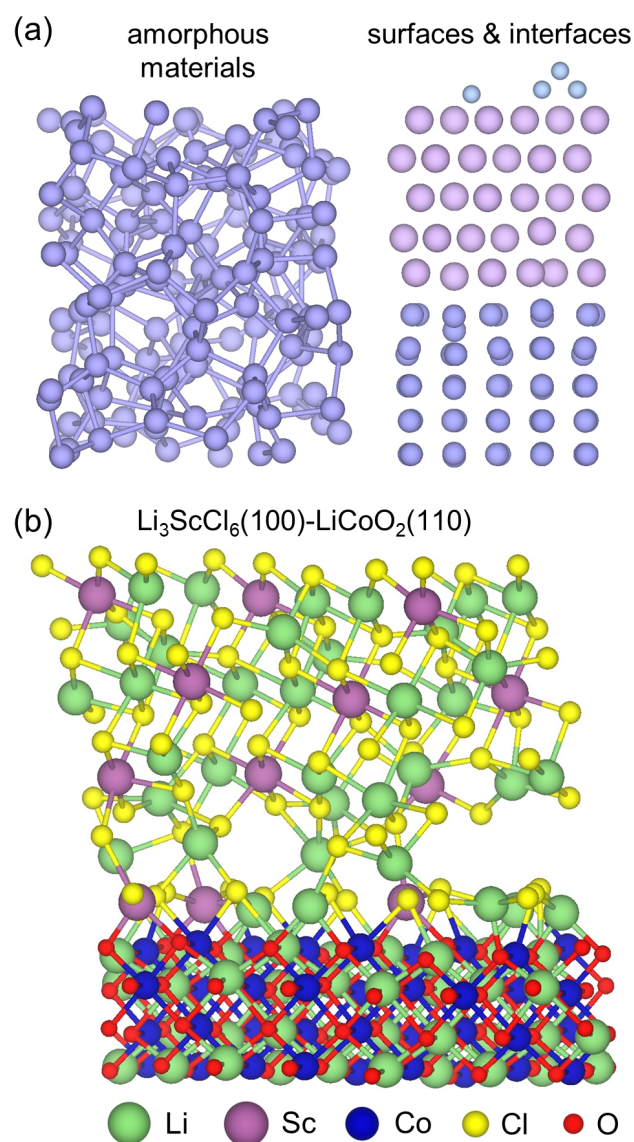


Fig. 1 (a) Examples of disordered structures including amorphous materials, surfaces, and interfaces. (b) $\text{Li}_3\text{ScCl}_6(100)\text{-LiCoO}_2(110)$ interface system of this study. A disordered interface region is formed between the two slabs.

tures for their validity. Training generative models for materials discovery inherently makes the evaluation of the models’ performance difficult. In more conventional problems such as image generation or speech synthesis, it is relatively easy to discern if the model has learned from the training data to generate realistic images or speech. However, it is much more difficult to determine if a newly generated material is realistic, or if the model is simply generating noise. With only limited and subjective metrics describing material validity, it is difficult to evaluate which model design choices yield better performance. One approach to circumventing this issue is to train the model on a fixed set of materials (e.g., perovskites). Restricting the material space allows for easier determination of whether the correct structure is being generated, and direct comparison of the properties between

the generated and training structures can be performed. In this scenario, however, the materials on which the model is trained should be sufficiently challenging to provide meaningful evaluation of the model's performance. In this regard, disordered materials are good candidates for the task, given that generative models can fail to generate even a single type of disordered material (e.g., amorphous silicon²³).

In this work, we demonstrate a design and evaluation process for generative models for a fixed set of disordered materials. We show how this process is valuable to improving generative models in general (and not just for disordered materials). We chose the commonly used point cloud representation for this work, and tested multiple different GAN architectures where we included bond distance information explicitly in the GANs. The performances of these GANs were evaluated using a $\text{Li}_3\text{ScCl}_6(100)$ - $\text{LiCoO}_2(110)$ disordered battery interface system (see Fig. 1b). The GANs were tasked with generating interface structures with low interface energy. Comparison of GAN performances reveal the architecture that is most effective at accomplishing this goal, as well as reasons why the other architectures do not perform as well. We demonstrate that the best architecture, Crystal Interface Generative Adversarial Network (CryinGAN), can generate large disordered interfaces with low interface energy, and similar structural features to the training structures. Through this work, we provide insights regarding generative model design and promising directions to pursue for future models of disordered materials.

2 Results and discussion

2.1 Interface disorder mechanism

The disordered interface considered in this work is a solid-state battery interface between the LiCoO_2 (LCO) cathode and the Li_3ScCl_6 (LSC) solid electrolyte. LCO is one of the most commonly used cathode materials in commercial Li-ion batteries³⁴. LSC³⁵ belongs to the class of halide solid electrolytes, which can achieve both high ionic conductivity and high-voltage stability, enabling the use of high-voltage cathodes in all-solid-state batteries^{36,37}. In previous work³⁸ focused on the interface between LCO and a different halide solid electrolyte, Li_3YCl_6 (LYC), we found the formation of a highly disordered LCO-LYC interfacial structure. This disordered interfacial structure was similarly observed for the calculated LCO-LSC interfaces considered in this work, despite the different compositions and crystal structures of the halide solid electrolytes.

Disordered interfaces have been observed experimentally across diverse material systems. They arise for a variety of reasons including elemental segregation to the interface^{39,40}, intermixing across the interface⁴¹, and ion irradiation⁴². For the material system of this study, we found that chlorides have an innate tendency to form disordered interfaces with oxides. In this subsection, we provide an explanation of the interface disorder mechanism for completeness, before presenting on generative modeling of the LCO-LSC interfaces. While many heterointerfaces can adopt regular epitaxial registries connecting two materials with well-defined crystalline orientations, other heterointerfaces show irregular, disordered interfacial patterns. As shown in Fig. 2, us-

ing binary instead of ternary materials, we found that disordered interfacial structures are obtained with only the combination of Li_2O and LiCl . We suggest two interdependent reasons for their occurrence. The first reason is the bond formation between O (in the oxide) and Li (in the chloride). Fig. 2a shows the interface structure of $\text{Li}_2\text{O}(100)$ - $\text{LiCl}(100)$ for $\text{Li}_2\text{O}(100)$ slabs that are either Li-terminated or O-terminated. When $\text{Li}_2\text{O}(100)$ is Li-terminated, there is minimal rearrangement at the interface with only some Li-Cl bond formation. In contrast, when $\text{Li}_2\text{O}(100)$ is O-terminated, significant atomic rearrangement is observed with a combination of Li-O and Li-Cl bond formation. The stronger interaction between O and Li is likely due to the higher charge density of O compared to Cl. However, Li-O bond formation alone does not lead to highly disordered interfaces, since such degree of disorder is not observed in other O-containing interfaces such as oxide-oxide and oxide-sulfide interfaces^{43,44}.

The second reason that the chlorides can form disordered interfaces is due to their intermediate mechanical stiffness. To form bonds with the O in the oxide slab, the other slab must distort its crystal structure to align its atoms with O, and the stiffness of the material determines the extent of the distortion allowed. Fig. 2b shows the interface structure between Li_2O and materials in increasing order of mechanical stiffness (Li, LiCl , and MgS). MgS has the same rock salt crystal structure as LiCl but with higher stiffness^{45,46}. Due to its higher stiffness, it is unable to distort significantly to bond with O. On the other hand, LiCl is soft enough that it can distort itself, whilst maintaining some of its structure, resulting in a more distorted interface structure. For a very soft material like Li metal⁴⁷, the atoms can rearrange themselves relatively freely to bond with O, so large voids such as those observed for LiCl are less likely to form. The intermediate stiffness of the chlorides, combined with bond formation with O, results in the chloride solid electrolytes' ability to form highly disordered interface structures with the oxide cathode. The disordered nature of the LCO-LSC interfaces, as shown in Fig. 1b, is therefore understood from the two considerations above.

2.2 Dataset generation

To train the interface GAN models, we first generated a dataset of LSC-LCO interface structures. A brief overview of the methodology is outlined here, further details can be found in the Methods section. $\text{Li}_3\text{ScCl}_6(100)$ - $\text{LiCoO}_2(110)$ interfaces (see Fig. 1b for reference) were generated by randomly generating 3 formula units of LSC atoms in the interface region between the LSC and LCO slabs. For each structure, the thickness of the interface region was randomly selected between 4 and 6 Å, and a random lateral displacement was applied to the LSC slab (translation allowed along the full range of both lateral directions). The randomly generated structures were relaxed using density functional theory (DFT) calculations.

To reduce the time required for DFT relaxations, we trained from scratch a machine learning interatomic potential, M3GNet⁴⁸, for the LSC-LCO interfaces, and used it to pre-relax structures prior to DFT relaxation. A total of 15,484 training structures consisting of optimized structures and intermediate

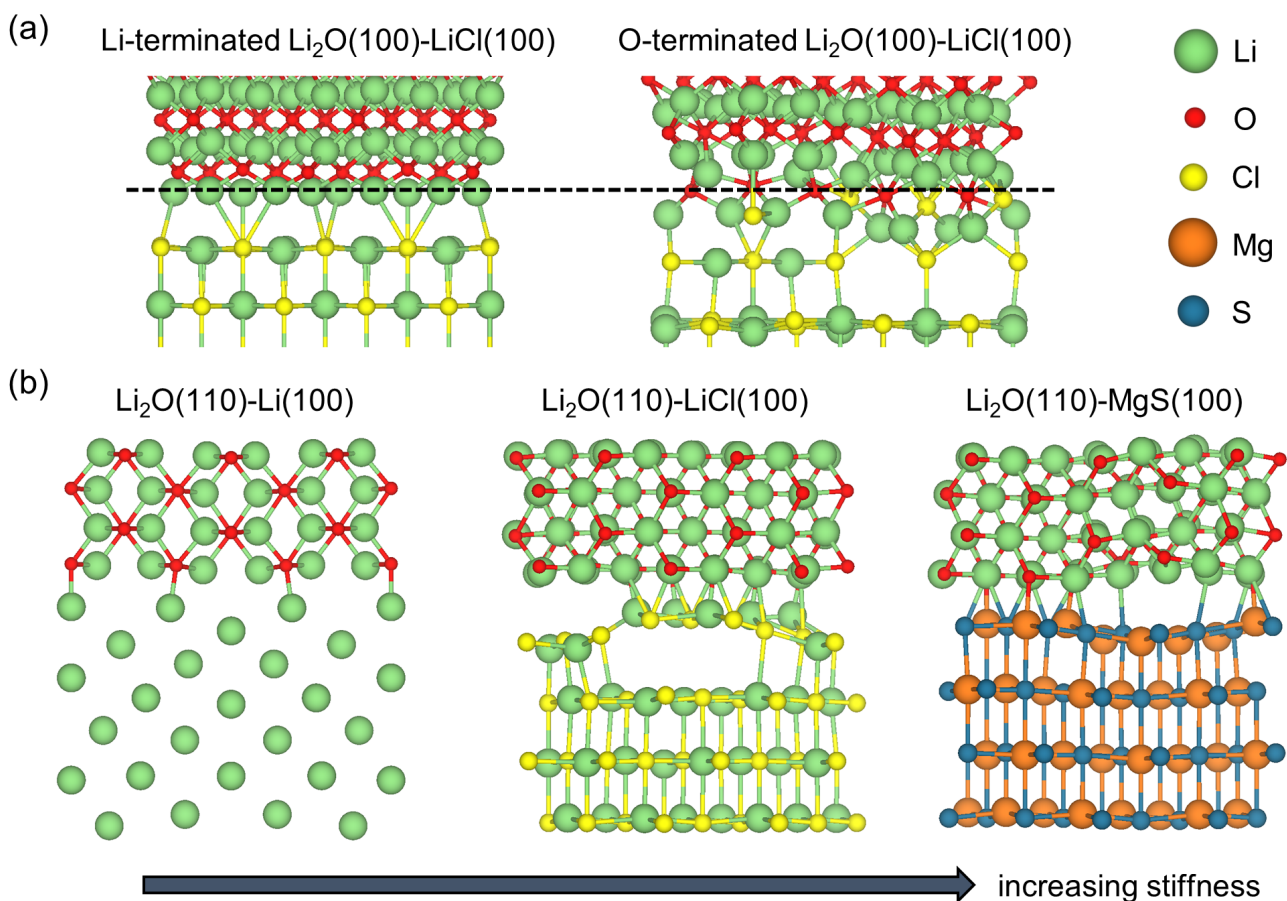


Fig. 2 (a) Interface structures of $\text{Li}_2\text{O}(100)$ - $\text{LiCl}(100)$ for Li-terminated and O-terminated $\text{Li}_2\text{O}(100)$. The dashed line indicates the termination layer of $\text{Li}_2\text{O}(100)$. (b) Interface structures for $\text{Li}_2\text{O}(110)$ - $\text{Li}(100)$, $\text{Li}_2\text{O}(110)$ - $\text{LiCl}(100)$, and $\text{Li}_2\text{O}(110)$ - $\text{MgS}(100)$. The mechanical stiffness of the bottom slab material increases from left to right.

ionic steps of the DFT relaxations were used to train the M3GNet interatomic potential. The M3GNet model achieved low test set mean absolute errors (MAEs) of 2.70 meV/atom, 20.9 meV/Å, and 0.0146 GPa for energy, force, and stress respectively. Machine learning interatomic potentials with similar (or higher) MAEs showed good performance in relaxations and molecular dynamics simulations when applied to other Li-ion conductors^{48,49}. We then relaxed randomly generated interface structures using the M3GNet interatomic potential, followed by subsequent DFT relaxations. The interface energies of the relaxed structures were distributed across a wide range (approximately 1.4 J/m²) as shown in ESI Fig. S1. We defined structures with interface energies no higher than 0.4 J/m² relative to the lowest energy structure to be low-interface-energy structures. For reference, the observation frequency of a grain boundary in aluminum metal decreases by 95% when the grain boundary energy increases by around 0.35 J/m²⁵⁰. The intended task of the GANs was to generate these low-interface-energy structures. We assembled 1,500 low-interface-energy structures that were all relaxed by M3GNet + DFT, for use as the GAN training dataset.

2.3 GAN architecture comparison

A typical GAN consists of two neural networks: a generator and a discriminator. The role of the generator is to generate material structures from input noise, whereas the role of the discriminator is to distinguish between the real (training) structures and the fake (generated) structures. The generator and discriminator compete with each other during training to progressively improve the quality of the generated structures. The point cloud representation was used for the GANs, and since all of the interface structures share the same lattice parameters, each structure was simply represented by the fractional coordinates of its atoms. We tested a couple of different GAN architectures as summarized in Fig. 3. We will present each architecture in succession and how they compare against each other.

The base GAN model (CryinGAN) was adapted from the Composition-Conditioned Crystal GAN (CCCGAN) presented by Kim et al.²⁵, which was used to generate Mg-Mn-O ternary materials. One-dimensional (1D) convolutions were used in the discriminator to extract the latent features of structures, an inspiration taken from PointNet³³, a 3D object classification and segmentation network. These 1D convolutions have been similarly implemented in other point-cloud-based crystal generative mod-

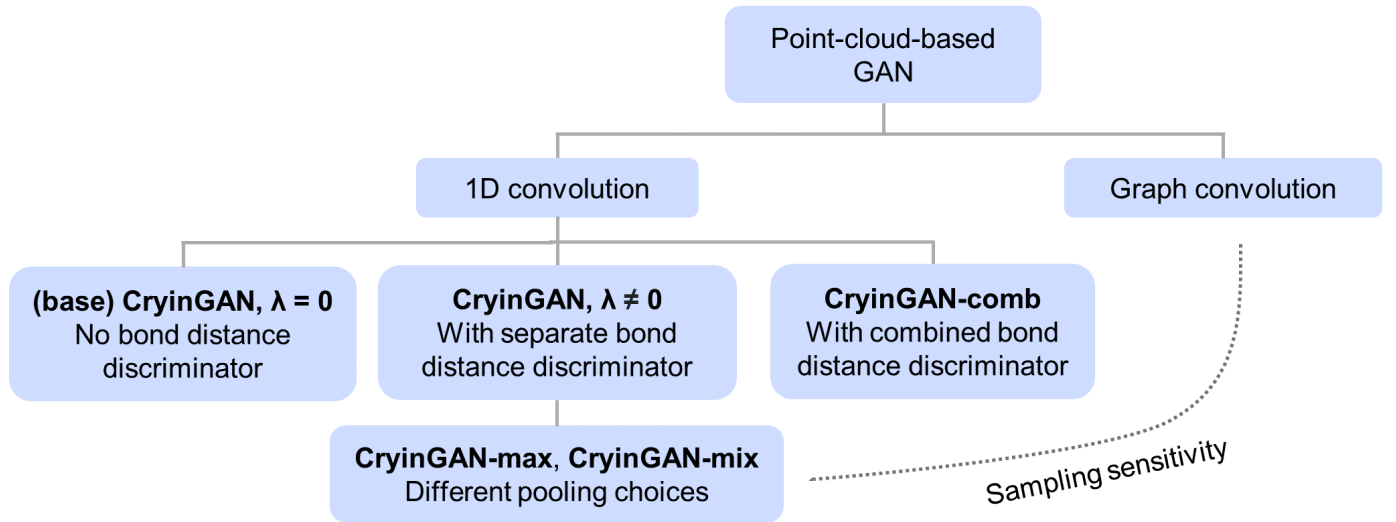


Fig. 3 Schematic of GAN architectures tested. The discriminator either uses 1D convolutions (PointNet) or graph convolutions (CGCNN). For the 1D-convolution-based discriminators, the primary CryinGAN design consists of a fractional coordinate discriminator and a separate bond distance discriminator, where the output of the latter is weighted by λ . Alternate pooling choices were tested (CryinGAN-max and CryinGAN-mix). Graph convolution and different types of pooling affect the sampling sensitivity of the discriminator. CryinGAN-comb combines the two discriminators into a single discriminator.

els such as FTCP-VAE⁹ and CubicGAN¹⁰. We simplified and generalized CCCGAN to be used for any periodic system with fixed lattice, composition, and number of atoms. The discriminator of the original CCCGAN relied solely on atomic coordinates and lattice parameters to distinguish between real and fake structures. To further provide explicit bond distance information to the discriminator, we added a second discriminator to the CryinGAN model that has the same architecture, but accepts bond distances as input instead of coordinates.

The CryinGAN model architecture is shown in Fig. 4. Note that each interface structure has a total of 264 atoms. The generator accepts random gaussian noise as input, and produces fractional coordinates of structures as output. Of the two discriminators, one accepts fractional coordinates as input, and the other accepts bond distances (6 nearest-neighbors of each atom) as input. In the first convolutional layer of the discriminators, the fractional coordinates/bond distances are convoluted along each row separately (i.e., separate 1D convolutions for each atom). CryinGAN implements the Wasserstein loss, which was shown to provide more stable training by preventing the vanishing gradients issue of the traditional GAN^{51,52}. The discriminator loss function with a gradient penalty term for improved stability⁵² is as follows:

$$L_{\text{disc}} = \mathbb{E}_{\tilde{x} \sim \mathbb{P}_g} [D(\tilde{x})] - \mathbb{E}_{x \sim \mathbb{P}_r} [D(x)] + \mu \mathbb{E}_{\hat{x} \sim \mathbb{P}_{\hat{x}}} [(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1)^2], \quad (1)$$

where D is the discriminator output, \tilde{x} and x are the inputs for generated and real structures respectively. The distribution $\mathbb{P}_{\hat{x}}$ is taken over interpolated samples between the distribution of real structures \mathbb{P}_r , and the distribution of generated structures \mathbb{P}_g . In the code implementation, \hat{x} is obtained by interpolating between the fractional coordinates/bond distances of training structures and generated structures. The interpolation point between

any two data points is chosen randomly. The parameter μ is the gradient penalty coefficient set to 10 similar to past Wasserstein GANs^{10,25,52}. The total discriminator loss, to be minimized, is a weighted sum of the losses from both discriminators

$$L_{\text{disc,total}} = L_{\text{disc,coord}} + \lambda L_{\text{disc,bond}}. \quad (2)$$

Here, $L_{\text{disc,coord}}$ and $L_{\text{disc,bond}}$ are the losses from the fractional coordinate discriminator and bond distance discriminator respectively, and λ is the weight of the bond distance discriminator loss. The total generator loss, to be maximized, is computed similarly according to the following equations:

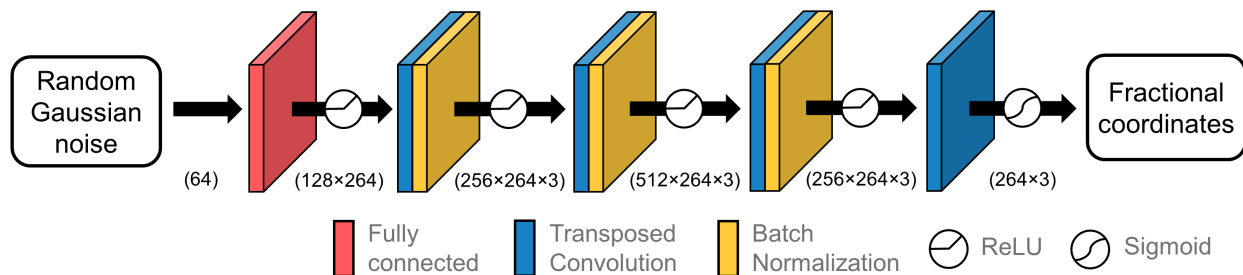
$$L_{\text{gen}} = \mathbb{E}_{\tilde{x} \sim \mathbb{P}_g} [D(\tilde{x})], \quad (3)$$

$$L_{\text{gen,total}} = L_{\text{gen,coord}} + \lambda L_{\text{gen,bond}}. \quad (4)$$

Note that the base CryinGAN model uses only the fractional coordinate discriminator ($\lambda = 0$), which we use as the baseline reference model.

CryinGAN models were trained on the assembled dataset of low interface energy structures with varying λ values. For each λ value, three separate models were trained and the combined results were used for analysis. A visualization of the training process is shown in ESI Movie S1, where CryinGAN progressively learns to generate low interface energy structures over training epochs. Due to noise in the models, the generated structures often have a small fraction of atoms that are too close to each other; in these cases we manually moved the atoms apart before relaxation. As λ increases, we found that the number of pairs of atoms generated too close together quickly decreases and then levels off around $\lambda = 0.05$ - 0.1 (see ESI Fig. S2). This observation is an indicator that including the bond distance discriminator is effective.

Generator



Discriminator × 2

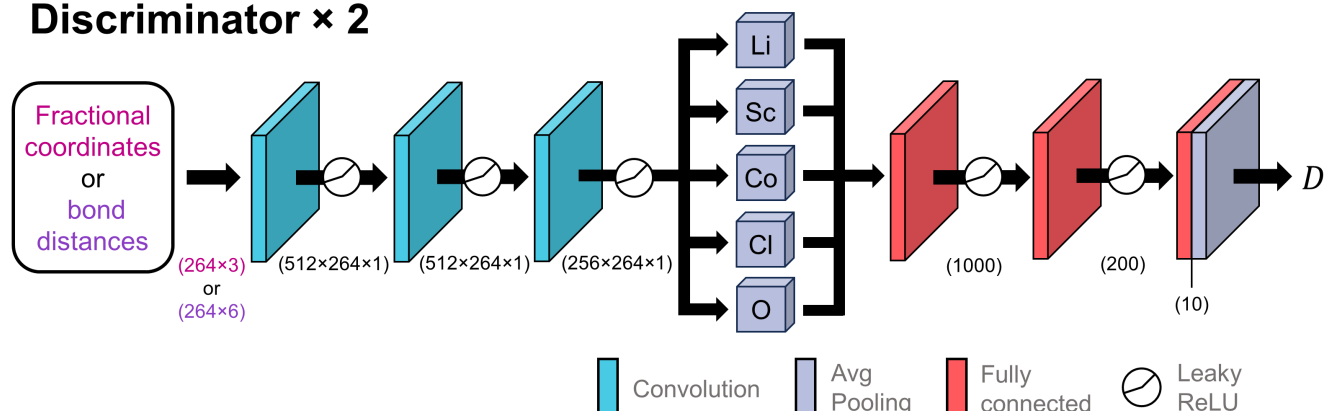


Fig. 4 CryinGAN architecture. The generator takes in random Gaussian noise as input and produces fractional coordinates as output. There are two discriminators with the same architecture, but with different inputs, i.e., fractional coordinates or bond distances. D is the discriminator output used to calculate the losses. The numbers in brackets represent the dimensions of tensors before/after a layer, where the batch dimension is omitted. The size of the layers shown in the figure do not reflect the tensor dimensions.

tive for training the generator to create structures with more reasonable atom-atom distances, in terms of minimal spacing. The generated structures were relaxed using M3GNet to evaluate their interface energies for comparison between models. We found that M3GNet relaxation alone provided sufficiently accurate interface energies without further DFT relaxation (see Table 3 in Methods section).

The objective was to obtain a model with the lowest interface energy distribution. Fig. 5 shows the interface energy distribution for the models trained with different parameters λ . For each value of λ , three separate models were trained and the spread of the interface energy distributions is indicated by the shading. The interface energy distribution of all trained models (without shading) is provided in ESI Fig. S3. The results indicate that as λ increases from 0 to 0.05, the interface energy distribution shifts to lower energies. However, as λ further increases to 0.1, the interface energy distribution shifts to higher energies. This observation shows that optimal values of λ improve model performance, but excessive weight on the bond distance discriminator causes the generator to prioritize the bond distances too much over positioning the atoms correctly. The use of a second discriminator does slow down training compared to using only a single discriminator, requiring around twice as long to train the model for the same number of epochs. However, we found that two discriminators still outperform a single discriminator given

the same amount of training time (see ESI Fig. S4), justifying the benefits of the bond distance discriminator. Also, whereas a large percentage of structures (around 34%) generated using the models trained with $\lambda = 0$ failed to converge in the M3GNet relaxations, only around 11% of structures generated using the models trained with $\lambda = 0.05$ did not converge. These results show that with an appropriately tuned λ , the bond distance discriminator helps to improve the model performance.

We considered a different architecture that circumvents the need to tune λ , where both fractional coordinate and bond distance discriminators were combined into a single discriminator, which we refer to as CryinGAN-comb (see ESI Fig. S5). The outputs were combined after the pooling layer, and the model was allowed to learn the relative importance of the coordinate and bond distance latent features on its own through fully connected layers. For CryinGAN-comb, the bond distances were directly obtained using fractional coordinates (instead of Cartesian coordinates), to provide a more direct link to the atomic coordinates which are also represented in fractional coordinates. CryinGAN-comb was trained using the same dataset, then structures were generated from it and relaxed using M3GNet. Compared to CryinGAN trained with $\lambda = 0$, CryinGAN-comb generates structures with fewer atoms too close together (around 2 times fewer pairs of atoms ≤ 1.5 Å). However, as shown in Fig. 6a, the interface energy distribution of (relaxed) structures generated from

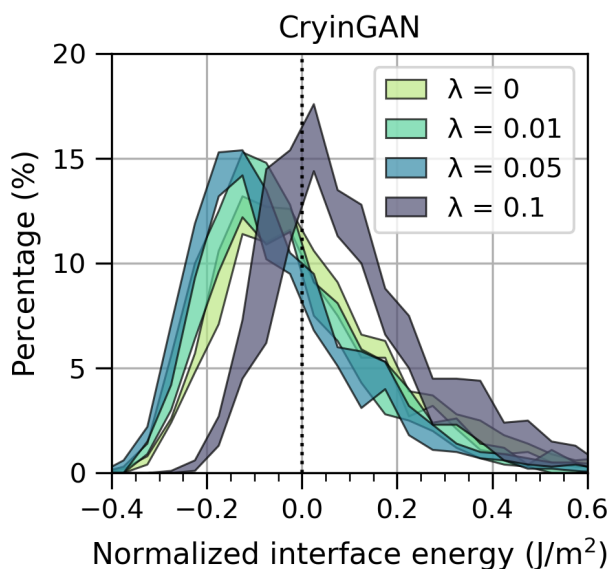


Fig. 5 Normalized interface energy distributions of structures generated using CryinGAN trained with different λ values (0, 0.01, 0.05, and 0.1). The generated structures were relaxed using M3GNet, and the interface energies shown are based on M3GNet-calculated energies. For each λ value, three separate models were trained, and the spread of the interface energy distributions is indicated by the shading. As λ increases, the interface energy distribution initially shifts to lower energies, then shifts to higher energies.

CryinGAN-comb is significantly higher than CryinGAN. For reference, the interface energy distribution of all trained models (without shading) is provided in ESI Fig. S6. These results show that combining the discriminators in CryinGAN-comb leads to less useful discriminator gradients to train the generator, resulting in poor quality of structures being generated. In contrast, separating the discriminators as in CryinGAN helps the GAN to learn more effectively.

Next, we considered different choices of the pooling layer for the discriminator. Wang et al.⁵³ found that the type of pooling operation affected the sampling sensitivity of the discriminator and the overall performance of the GAN. The sampling sensitivity describes how sensitive the discriminator is to changes in point density or the sampling pattern of the input point cloud. Their results suggest that max pooling produces a discriminator with lower sampling sensitivity than average pooling. The CryinGAN discriminators use average pooling, and we tested an architecture which uses max pooling instead, referred to as CryinGAN-max (see ESI Fig. S7a). We also tested an architecture named CryinGAN-mix that uses the mix pooling operation proposed by Wang et al.⁵³, where both max and average pooling operations are used together (see ESI Fig. S7b). CryinGAN-max and CryinGAN-mix were trained using $\lambda = 0$ and 0.05. For $\lambda = 0.05$, only the pooling layer of the bond distance discriminator was changed, and we kept the average pooling for the fractional coordinate discriminator. This choice allows us to study the effect of the pooling choice on the two discriminators independently.

The interface energy distributions of relaxed structures gen-

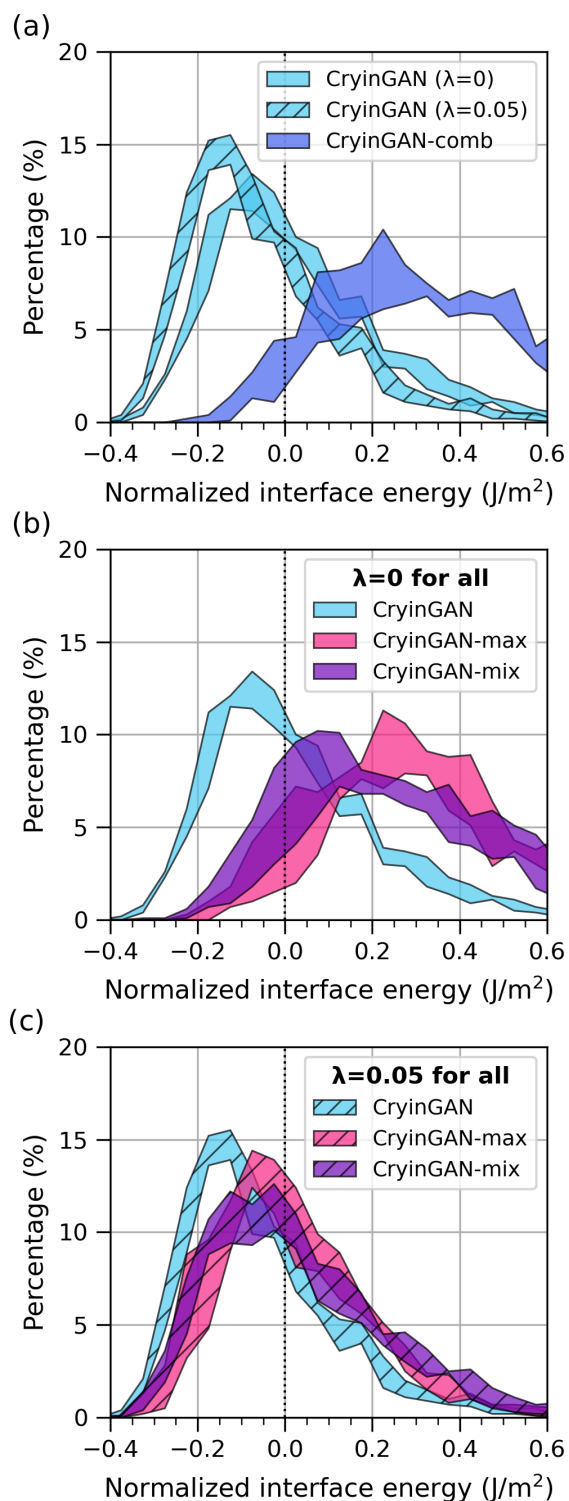


Fig. 6 (a) Normalized interface energy distributions of structures generated using CryinGAN ($\lambda = 0, 0.05$) and CryinGAN-comb. (b) and (c) show the normalized interface energy distributions of structures generated using CryinGAN (average pooling), CryinGAN-max (max pooling), and CryinGAN-mix (mix pooling). (b) shows models without the bond distance discriminator ($\lambda = 0$), and (c) shows models with the bond distance discriminator ($\lambda = 0.05$). All structures were relaxed using M3GNet, and the interface energies shown are based on M3GNet-calculated energies.

erated using CryinGAN, CryinGAN-max, and CryinGAN-mix are shown in Fig. 6b and c. The interface energy distribution of all trained models (without shading) is provided in ESI Fig. S8 for reference. For $\lambda = 0$ (Fig. 6b), we observe that CryinGAN (average pooling) significantly outperforms CryinGAN-max and CryinGAN-mix. Although the use of max pooling was beneficial for the generation of 3D objects where the object shape is the most important aspect to capture⁵³, it appears that a higher sampling sensitivity is needed for atomic configurations, for which the local coordination environment is the important aspect to capture. For $\lambda = 0.05$ (Fig. 6c), we observe that the distributions are more similar across the three architectures, but CryinGAN still has the lowest energy distribution. The pooling choice of the bond distance discriminator does not relate to the sampling sensitivity like the fractional coordinate discriminator, and the choice appears to have a smaller effect on model performance. Overall, we find that average pooling works best for both discriminators.

Finally, we attempted a graph convolutional discriminator by adapting the commonly used Crystal Graph Convolutional Neural Networks (CGCNN)⁵⁴. A higher sampling sensitivity can be achieved through the use of graph convolutions, where each atom is convoluted with its surrounding atoms and bonds. Unlike the PointNet-based 1D convolutional architecture, graph convolutions include explicit information of the neighbor relationship between atoms. However, the training losses diverged and it was not possible to train a GAN that could generate useful structures (see ESI Fig. S9). This result is consistent with the findings of Wang et al.⁵³, whose graph convolutional GAN also failed to produce point clouds of 3D objects. Graph convolutions are highly sensitive to sampling, making them prone to overfocus on the sampling pattern of a point cloud instead of the overall structure. This overfocus bears similarity to the behavior observed here for CryinGAN at high λ values. For graph convolutions to be implemented in point-cloud-based GANs, we expect that a carefully designed architecture will be required to take advantage of its sampling sensitivity without destabilizing training.

2.4 Evaluation of CryinGAN-generated structures

Since the best GAN architecture was found to be CryinGAN using both the fractional coordinate and bond distance discriminators, we trained a CryinGAN model ($\lambda = 0.05$) to use for further evaluation of the generated structures. The model was trained for a higher number of epochs with a shorter interval between generator trainings (see Methods section for more details). Fig. 7 shows the interface energy distribution of (relaxed) structures generated using the CryinGAN model, compared to randomly generated structures (also relaxed). The shift of the interface energy distribution to the left shows that CryinGAN has learnt to generate low-interface-energy structures. With random generation, only around 48% of the structures had low interface energy, whereas with CryinGAN, the percentage increased to around 85%. However, although CryinGAN is capable of generating low-interface-energy structures, it does not reproduce the interface energy distribution of the training dataset (yellow line). Presence of noise in the model, or mode collapse⁵⁵ where the GAN becomes bi-

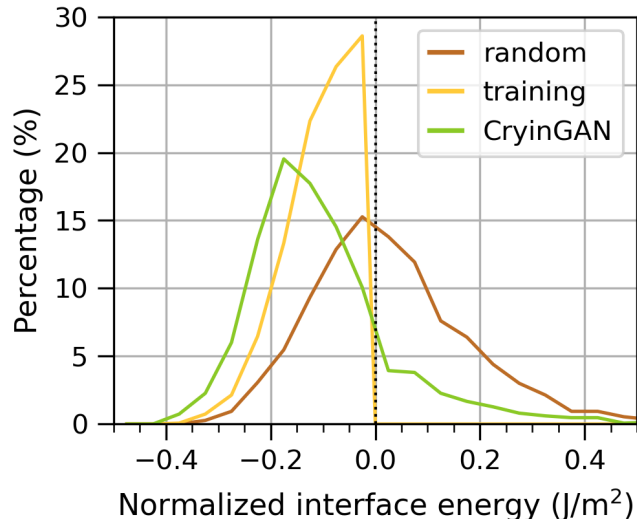


Fig. 7 Normalized interface energy distribution of structures generated using the best CryinGAN model (green), compared against structures that were randomly generated (brown). All structures shown were relaxed using M3GNet followed by DFT calculations, and the energies shown are with respect to DFT-calculated energies. Randomly generated structures with (relaxed) normalized interface energy ≤ 0 J/m² were used as the CryinGAN training dataset (yellow).

ased towards generating certain types of structures only, is often the source of such differences between the generated and training distribution. We will further discuss the differences between the generated and the training structures in subsequent structural analysis.

We filtered CryinGAN structures with low interface energy (≤ 0 J/m²) for structural comparison with the training structures. We studied the structural similarity between the CryinGAN and training structures by analyzing the coordination environment (motifs) of atoms in the disordered interface region. The CrystalNN-Fingerprint⁵⁶, which contains the coordination likelihoods and local structure order parameters of a given atom, was calculated and averaged across structures in each dataset. We calculated the fingerprints of the cations (Li and Sc) in the interface region coordinated to the anions (Cl and O). For Li species, Table 1 shows the Euclidean distance and cosine similarity between the (average) interface fingerprint of the training and CryinGAN datasets. For comparison, we also performed the same analysis for a dataset of high-interface-energy structures (interface energy > 0 J/m²) that were randomly generated and relaxed. Compared to the high energy dataset, the CryinGAN dataset has a lower Euclidean distance and a higher cosine similarity to the low energy dataset, indicating that the interface Li atoms in the CryinGAN structures are more similar to the low-interface-energy training structures, than the high-interface-energy structures.

To study the structural features in more detail, Fig. 8 shows the distribution of the most likely coordination motif of each interfacial Li atom. Although Li atoms are octahedrally coordinated in the bulk of LCO and LSC, it is found in a wide range of coordination motifs in the disordered interface region. Despite the

Table 1 Euclidean distance and cosine similarity between the interface Li site fingerprints of the training structures and the CryinGAN-generated/high-interface-energy structures. The 95% bootstrap confidence intervals are shown in brackets.

Dataset	Euclidean distance (95 % CI)	Cosine similarity (95 % CI)
CryinGAN	0.02374 (0.01622 to 0.02763)	0.9996 (0.9996 to 0.9999)
High energy	0.08987 (0.08204 to 0.09741)	0.9942 (0.9933 to 0.9953)

complex structure of the interface, the Li coordination motif distribution of the CryinGAN-generated structures closely resembles that of the training structures. On the other hand, the distribution of the high-interface-energy structures is shifted towards motifs with lower coordination number (shifted upwards), indicating that the fewer bonds lead to higher energies. These results reflect the higher fingerprint similarity of the CryinGAN dataset to the training dataset as seen in Table 1, and show that CryinGAN is able to reproduce the Li coordination environment of the training structures.

Table 2 shows the Euclidean distance and cosine similarity between the interface Sc fingerprint of the CryinGAN/high energy dataset and the training dataset. Unlike the Li fingerprints, both metrics are similar for the CryinGAN and high energy datasets. This finding indicates that the average Sc coordination environment in the CryinGAN structures is not more similar to the training structures than the high energy structures. However, the Sc environments of the CryinGAN structures and the high-energy structures still exhibit differences with each other, as discussed directly below.

Table 2 Euclidean distance and cosine similarity between the interface Sc site fingerprints of the training structures and the CryinGAN/high-interface-energy structures. The 95 % bootstrap confidence intervals are shown in brackets.

Dataset	Euclidean distance (95 % CI)	Cosine similarity (95 % CI)
CryinGAN	0.09407 (0.07568 to 0.11193)	0.9895 (0.9859 to 0.9940)
High energy	0.10508 (0.09029 to 0.11642)	0.9899 (0.9879 to 0.9932)

Examination of the Sc coordination motif distribution shows that although the CryinGAN dataset and the high energy dataset both vary from the training dataset, they vary in different ways. Fig. 9 shows the distribution of selected coordination motifs around Sc species, focusing on the motifs that show the largest differences across the datasets. Each motif is further subdivided based on the number of O atoms in the coordination shell of Sc (the distribution of all coordination motifs is provided in ESI Fig. S10b for reference). Compared to the training dataset, the high energy dataset shows significantly higher percentages of motifs with no O atoms present, and lower percentages of motifs with O atoms. This finding indicates that the lower frequency of bonding between O (in LCO) and Sc leads to weaker interface binding and higher interface energy. In contrast, the CryinGAN

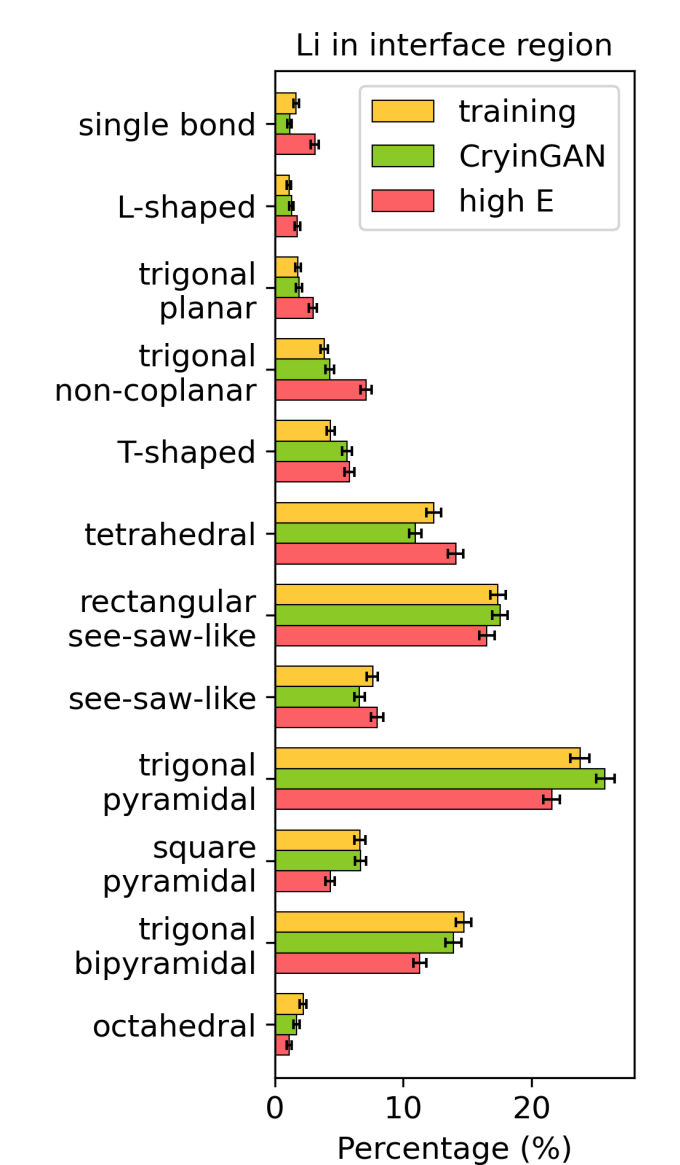


Fig. 8 Coordination motif distributions of Li in the interface region for three datasets: (1) training structures with low interface energy, (2) CryinGAN-generated structures, and (3) structures with high interface energy. All structures were relaxed using M3GNet followed by DFT calculations. Error bars represent 95 % bootstrap confidence intervals. The coordination motifs are ordered in increasing coordination number from top to bottom.

dataset shows a higher frequency of Sc-O bonding than the training dataset, with higher percentages of motifs with 2 O atoms (trigonal bipyramidal and pentagonal pyramidal), and lower percentages of motifs with no O atoms (trigonal bipyramidal and octahedral). The average O bond count per Sc atom provided in ESI Fig. S11b also shows that the high energy structures have fewer Sc-O bonds, whereas the CryinGAN structures have higher number of Sc-O bonds. This analysis shows that the Sc coordination environments of the CryinGAN structures differ from the training structures due to a higher frequency of Sc-O bonding. As Sc-O bonds appear to induce low-interface-energy structures, the

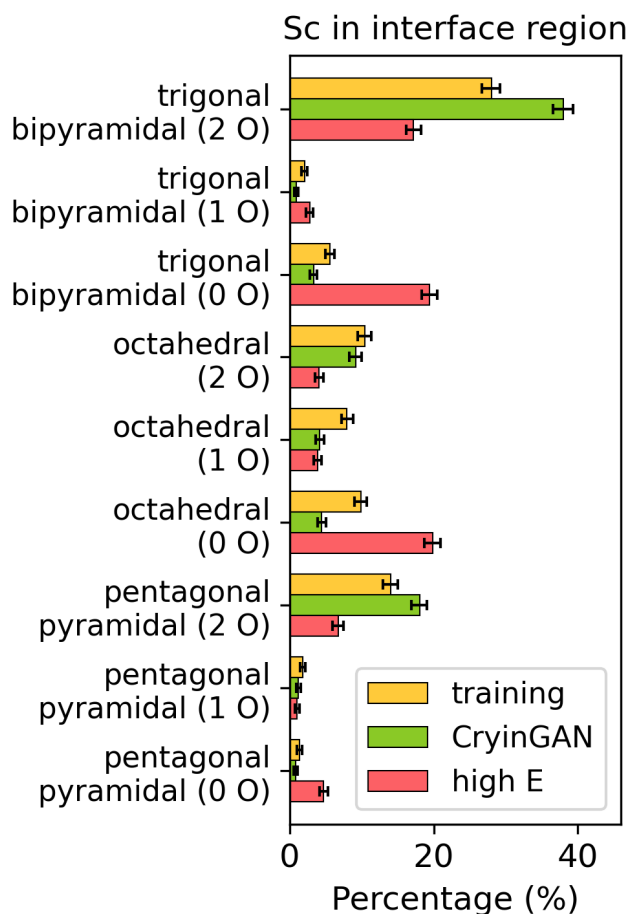


Fig. 9 Percentages of selected coordination motifs for Sc in the interface region. The motifs are subdivided based on the number of O atoms in the motif as indicated in brackets in the y-labels. The percentages of coordination motifs not shown are similar across the three datasets. Error bars represent 95 % bootstrap confidence intervals.

overpopulation of Sc-O bonds in the CryinGAN configurations is likely responsible for the lower interface energy distribution of the CryinGAN structures compared to the training dataset in Fig. 7.

We also analyzed the radial distribution functions (RDFs) of the interface Li and Sc atoms of the datasets. The RDFs confirm the trends revealed by the coordination environment analysis, in which the CryinGAN dataset shows higher similarity to the training dataset than the high energy dataset. ESI Fig. S12a-c show respectively the Sc-O, Sc-Sc, and Li-Li RDFs. For the Sc-O RDF (ESI Fig. S12a), the high energy dataset exhibits peaks with lower magnitude, reflecting its reduced degree of Sc-O bonding. On the other hand, the Sc-Sc RDF (ESI Fig. S12b) shows a higher peak around 4 Å, indicating that the high energy structures have a higher proportion of Sc cations closer together. The associated Sc-Sc repulsion raises the energy of the structures. In the Li-Li RDF (ESI Fig. S12c), a new peak appears around 2.5 Å. This short Li-Li interatomic distance again raises the energy of the structures due to Li-Li repulsion. In contrast, the CryinGAN dataset does not show any of these high-energy features, and its RDFs are similar

to the training dataset.

3 Conclusions

Overall, we have shown that CryinGAN has learnt to generate disordered interface structures that not only have low interface energy, but also have good structural similarity with the training structures. This performance was achieved by training and evaluating on a fixed set of disordered materials, rather than training across compositions and space groups to generate new materials. By generating structures that can directly be compared with the training structures, it was possible to compare between model architectures and inform design choices. Including a second bond distance discriminator provided explicit information about the bond distances and improved model performance. The non-intuitive choice of using two discriminators led to better results than a single combined discriminator, despite the need for an extra tuning step. GANs also benefit from discriminators with high sampling sensitivity for generating atomic structures, but overly sensitive discriminators lead to training instability. The CryinGAN-generated interfaces exhibited the desired low-energy features of the training structures, even if the interface energy and coordination motif distributions showed differences with the training structures. Model evaluation through a fixed set of materials helps make the model development more robust and insightful, and disordered materials are sufficiently complex to be good datasets to help with this evaluation. However, generative models that incorporate symmetry constraints^{57,58} may not be able to take advantage of disordered materials for evaluation, unless the constraints can be made optional.

Amongst the options considered, the point cloud representation paired with the 1D convolutional architecture, despite its simplicity, was able to generate large, complex interface structure. Considering that the model lacks the symmetry invariances for translation, rotation, and cell repetition, its good performance is notable. However, there are some limitations to its performance. For example, we further tested CryinGAN on a dataset of amorphous silicon configurations⁵⁹ to see if it is able to generate these highly disordered structures, but found that CryinGAN unfortunately failed to generate amorphous silicon. The generated structures did not exhibit the tetrahedral coordination of amorphous silicon, and resembled randomly generated atoms (see ESI Fig. S13). It appears that to use the point cloud representation for such challenging structures, more sophisticated architectures beyond 1D convolutions are required. Graph convolution is an effective method to learn atomic structures, but implementing them in GANs requires designing a novel architecture specifically for graphs, which have already found success in VAEs and diffusion models^{26,60,61}. Autoencoders can also be combined with GANs^{12,23}, where the encoded latent vectors are used as input to a GAN. Graphs are one of many options for invariant representations, among others such as smooth overlap of atomic position (SOAP) vectors⁶² and the atom-centered symmetry functions (ACSFs)⁶³. These invariant representations should be further explored, even if reconstruction back to atomic coordinates may be needed. Reconstruction is not necessarily a barrier to their implementation, as demonstrated by Fung et al.⁶⁴ who showed that

reconstruction can be achieved through gradient-based optimization using automatic differentiation. Building the next generation of generative models will rely on not only developing better architectures and representations, but also adopting better evaluation methods that provide useful feedback about model performance. We hope that the findings of this work will help advance future generative models for both ordered and disordered materials.

4 Methods

4.1 GAN and Dataset

The dataset used for training the GANs consists of 1,500 relaxed low-interface-energy structures. All structures had the same lattice parameters and composition, so each structure was represented by its 2D matrix of fractional coordinates only. CryinGAN models were trained with $\lambda = 0, 0.01, 0.05$, and 0.1 for 100,000 epochs to study the effect of λ on model performance. Three separate models were trained for each λ value. A batch size of 32 was used, and Adam optimizers with learning rate of 5×10^{-5} were used for the generator and discriminator(s). The generator was trained only once every 5 batches to help stabilize the training. CryinGAN-comb, CryinGAN-max, and CryinGAN-mix models were trained using the same procedure, and three separate models were trained for each model configuration. CryinGAN-max and CryinGAN-mix, which use two discriminators, were trained using $\lambda = 0$ and 0.05 . For $\lambda = 0$, max pooling and mix pooling were used for the fractional coordinate discriminator of CryinGAN-max and CryinGAN-mix respectively. For $\lambda = 0.05$, max pooling and mix pooling were used for the bond distance discriminator of CryinGAN-max and CryinGAN-mix respectively, whereas average pooling was used for the fractional coordinate discriminator of both models. For each trained model, structures were generated, relaxed using the M3GNet interatomic potential⁴⁸, and the (normalized) interface energy distribution was calculated using 1,000 relaxed structures.

We also tested a graph convolutional neural network architecture by adapting CGCNN⁵⁴ as the discriminator. Instead of predicting material properties, the output of CGCNN was used to estimate the Wasserstein distance between the training and generated structures. Default values for the hyperparameters were used, as provided in the code repository of CGCNN. The GAN was optimized using the Adam optimizer and a batch size of 32, where we tested learning rates of 10^{-2} , 10^{-3} , 10^{-4} , and 10^{-5} . We found that the losses diverged and the GAN was unable to generate meaningful structures. We did not further pursue the development of graph convolutional neural networks as discriminators, considering similar failures reported in literature for point cloud GANs⁵³.

The best GAN architecture was found to be CryinGAN using both discriminators, where $\lambda = 0.05$ was found to be optimum among the values tested. We trained CryinGAN with $\lambda = 0.05$ to use for further structural analysis of the generated structures. The generator was trained once every 2 batches to speed up the training. We found the training to be stable at this frequency as shown by the Wasserstein distance plot in ESI Fig. S14a. Although the Wasserstein distance plateaued early in the training, the per-

centage of (relaxed) low-interface-energy structures continued to gradually increase (see ESI Fig. S14b and c). We stopped the training when the energy improvements have mostly plateaued, and used the last trained model as the best GAN model. Structures were generated using the model, and relaxed using M3GNet followed by DFT calculations to determine the interface energy distribution and perform subsequent structural analysis.

4.2 Structural analysis

Three datasets were used to perform the structural analysis: (1) CryinGAN-generated structures with low interface energy (≤ 0 J/m²), (2) randomly generated structures with low interface energy (i.e., the training structures), and (3) randomly generated structures with high interface energy (> 0 J/m²). Each dataset contains 1,500 structures that were relaxed using the M3GNet interatomic potential followed by DFT calculations.

To evaluate the structural similarity between the CryinGAN dataset and the training dataset, we compared the coordination environment of atoms in the interface region. The same comparison was carried out between the high energy dataset and the training dataset. For each structure, the CrystalNNFingerprint⁵⁶ was calculated for Li and Sc atoms in the interface region, allowing only the anions (Cl and O) to be considered as neighbors. The CrystalNNFingerprint is a vector containing coordination likelihoods, and local structure order parameters (LoStOPs) that describe the extent to which an observed coordination environment matches a perfect coordination motif. For each atom type (Li/Sc), an average fingerprint was calculated by averaging the site fingerprints of each structure, then averaging across all structures in a dataset. The Euclidean distance and cosine similarity between the average fingerprint of the CryinGAN/high energy dataset and the training dataset were calculated. To visualize the coordination motif distribution of the datasets, the most likely coordination motif of each interface Li/Sc atom was identified by first selecting the coordination number with the highest likelihood, then selecting the coordination motif of this coordination number with the highest LoStOP. We only plotted coordination motifs that appeared in > 1 % of the interface Li/Sc atoms in the training structures. 95 % bootstrap pivotal confidence intervals⁶⁵ were calculated using 1,000 bootstrap samples with 1,500 structures in each sample.

4.3 DFT calculations

All DFT calculations were performed using the Vienna Ab initio Simulation Package (VASP)^{66–69}, with the projector augmented-wave (PAW) method^{70,71}. The Li ($1s^2 2s^1$), Sc ($3s^2 3p^6 3d^2 4s^1$), Co ($3d^8 4s^1$), Cl ($3s^2 3p^5$), O ($2s^2 2p^4$), Mg ($2p^6 3s^2$), and S ($3s^2 3p^4$) electrons were treated as valence electrons in the pseudopotentials. The generalized gradient approximation (GGA) with the Perdew-Burke-Ernzerhof (PBE) exchange-correlation functional⁷² was used. The orbitals were expanded using a plane wave basis with cutoff energy of 520 eV for bulk structures, and 450 eV for interface structures (to lower computational cost). The DFT+U approach⁷³ was used to account for the electron localization of the Co-3d states, and we selected a U value of 4 eV as

reported in other literature⁷⁴.

To investigate the origin of the interfacial disorder observed for chlorides, interfaces of simpler materials (Li_2O -Li, Li_2O -LiCl, and Li_2O -MgS) were studied. Structural relaxations were first performed on unit cells of Li_2O ($Fm\bar{3}m$), Li ($Im\bar{3}m$), LiCl ($Fm\bar{3}m$), and MgS ($Fm\bar{3}m$). The cell shapes, cell volumes, and atom positions were allowed to relax, until the force on each atom was below 0.001 eV/Å. The Brillouin zone was sampled using a $(8\times8\times8)$ Monkhorst-Pack k-point grid for Li, and a $(6\times6\times6)$ Monkhorst-Pack k-point grid for Li_2O , LiCl, and MgS. The interface structures were constructed using the MPInterfaces package⁷⁵, which implements the lattice matching algorithm proposed by Zur et al.⁷⁶. The configurations of interfaces constructed and their lattice mismatches are listed in ESI Table S1. All interfaces were constructed with vacuum spacings of at least 14 Å. We chose the (100) orientation as a representative plane for Li, LiCl, and MgS.

To study the effect of different terminations of Li_2O surfaces on the interfacial structure with LiCl, we chose the $\text{Li}_2\text{O}(100)$ orientation which can either be Li-terminated or O-terminated. The surfaces of the $\text{Li}_2\text{O}(100)$ slabs are polar, so half of the Li/O atoms were moved from one surface to the other to neutralize the polarity (resulting in 'Tasker Type 2b' surfaces^{77,78}). Structural relaxations were performed on $\text{Li}_2\text{O}(100)$ -LiCl(100) interfaces for both terminations, allowing the cell shapes, cell volumes, and atom positions to relax until the force on each atom was below 0.05 eV/Å. The Brillouin zone was sampled using a $(3\times2\times1)$ gamma-centered k-point grid. To study the effect of mechanical stiffness on interfacial structure, we chose the $\text{Li}_2\text{O}(110)$ orientation, which exposes both Li and O at its surface, and paired it with Li(100), LiCl(100), and MgS(100). Structural relaxations were performed on these interfaces using the same procedure. The Brillouin zone was sampled using a $(8\times2\times1)$, $(3\times2\times1)$, and $(3\times2\times1)$ gamma-centered k-point grid for $\text{Li}_2\text{O}(110)$ -Li(100), $\text{Li}_2\text{O}(110)$ -LiCl(100), and $\text{Li}_2\text{O}(110)$ -MgS(100) respectively.

For the LCO-LSC interface system, we began by performing structural relaxations on unit cells of LCO ($R\bar{3}m$) and LSC ($C2/m$ ³⁵), allowing the cell shapes, cell volumes, and atom positions to relax, until the force on each atom was below 0.001 eV/Å. The Brillouin zone was sampled using a $(9\times9\times1)$ gamma-centered k-point grid for LCO, and a $(6\times4\times6)$ Monkhorst-Pack k-point grid for LSC. To construct the interface structures, we chose the orientations of LCO(110), a Li fast-diffusing plane⁷⁹, and LSC(100), a representative plane. The surfaces of the LSC(100) slab are polar, so half of the Cl atoms were moved from one surface to the other to neutralize the polarity.

The configuration of the lattice-matched interface is given in ESI Table S1, where the average lattice mismatch is 2.17 %. In a preliminary calculation, the interface was constructed by simply placing the LCO(110) and LSC(100) slabs in contact with each other, and we obtained a relaxed structure similar to that depicted in Fig. 1b. An interface region with disordered LSC atoms formed with a thickness of around 5 Å. Using this structure as reference, we then generated random interface structures using the CALYPSO package^{80–82}. We used LCO(110) and LSC(100) slabs with 4 and 7 layers respectively. For each structure, an interface region thickness was randomly chosen between 4 and 6 Å, then

the region was randomly populated with 3 formula units of LSC (9 Li, 3 Sc, and 18 Cl atoms). A random lateral displacement (in-plane direction parallel to the interface) was also applied to the LSC slab. A vacuum spacing of 14 Å was included in all the interface structures. For a subset of the generated structures, random mutual exchanges were performed between atoms in the interface region and the top layer of LCO, where we allowed up to 3 Sc \leftrightarrow Co, 3 Li \leftrightarrow Co, and 6 Cl \leftrightarrow O exchanges per structure. All interface structures have a total of 264 atoms per structure.

A total of 350 structures were generated without mutual exchanges, and 600 structures were generated with mutual exchanges. Structural relaxations were performed on all interface structures, where the lateral lattice vectors were fixed to the optimized values of the LCO slab (the elastic moduli of LCO⁸³ is significantly larger than LSC⁸⁴). The relaxations were performed in two stages. In the first stage, a kinetic energy cutoff of 374.3 eV was used, and only the atoms in the interface region were allowed to relax for 50 ionic steps. In the second stage, a kinetic energy cutoff of 450 eV was used, and all atoms were allowed to relax until the force on each atom was below 0.1 eV/Å. The Brillouin zone was sampled at the gamma point only for both stages.

4.4 M3GNet

We trained the M3GNet interatomic potential⁴⁸ on the LCO-LSC interface structures to allow us to perform relaxations quickly. The dataset used for M3GNet training and evaluation included the 950 relaxed interface structures and 14,534 intermediate ionic steps from the DFT relaxations (second stage only). The intermediate steps were sampled starting from the 5th ionic step of every relaxation with an interval of 10 steps. The dataset was split into 80 % training, 10 % validation, and 10% test data. We used the Adam optimizer⁸⁵ to optimize the loss function, L , as follows:

$$L = \text{MSE}_E + \text{MSE}_F + 0.1(\text{MSE}_S) \quad (5)$$

where MSE_E , MSE_F , and MSE_S are the mean squared error of energy, force and stress respectively. We tested different learning rates and batch sizes, and trained M3GNet models for 24 hours each using Tesla V100 GPUs. We found that the losses plateaued within the given training time, and there was little difference in errors between the different hyperparameters (see ESI Table S2). We chose the model with the smallest loss (learning rate = 0.001, batch size = 4) as the working interatomic potential of this work (see ESI Fig. S15 for the loss curves). The test set mean absolute errors for energy, force, and stress are 2.70 meV/atom, 20.9 meV/Å, and 0.0146 GPa respectively.

The GAN training dataset was obtained by randomly generating interface structures, then pre-relaxing them using the M3GNet interatomic potential. No mutual ion exchanges were performed when the structures were generated since the exchanges mostly led to high-interface-energy structures. The structures were optimized using a conjugate gradient algorithm⁸⁶ through the atomic simulation environment (ASE) package⁸⁷. The relaxations were stopped when the force on each atom was below 0.05 eV/Å. Then, the structures were relaxed through DFT calculations using a kinetic energy cutoff of 450 eV. Only the atom positions

were allowed to relax until the force on each atom was below 0.1 eV/Å. The Brillouin zone was sampled at the gamma point only. 1,500 low-interface-energy structures were filtered from the relaxed structures as the training dataset. The initial 950 structures that were relaxed using DFT calculations only were just used for training the M3GNet interatomic potential, but not for GAN training. This is to ensure a fair comparison between the GAN-generated and training structures, since we intended to pre-relax the GAN-generated structures with the M3GNet interatomic potential. Table 3 shows the errors between the M3GNet- and DFT-calculated total energies of the interface structures. The errors are generally low even when compared to the final DFT-relaxed structures, showing that the M3GNet relaxations are able to give accurate predictions of the energy, and yield structures close to DFT convergence.

Table 3 Performance of M3GNet relaxations in reaching DFT convergence. Interface structures are first optimized using M3GNet relaxations followed by DFT relaxations. n_{steps} is the number of DFT ionic steps required to fully relax the M3GNet-optimized structures. $|\Delta E_{\text{M-opt}}|$ is the absolute energy difference between the M3GNet and DFT energies of the M3GNet-optimized structure. $|\Delta E_{\text{D-opt}}|$ is the absolute energy difference between the M3GNet energy of the M3GNet-optimized structure and the DFT energy of the final DFT-optimized structure. The mean and standard deviation for each quantity are listed.

Structure type	n_{steps}		$ \Delta E_{\text{M-opt}} $ (meV/atom)		$ \Delta E_{\text{D-opt}} $ (meV/atom)	
	mean	std	mean	std	mean	std
low energy	17.6	25.7	3.14	3.41	2.15	1.42
high energy	56.1	51.8	11.0	15.3	4.75	4.51
all	31.7	41.8	6.03	10.4	3.10	3.21

Before relaxing the GAN-generated structures, the structures were first preprocessed by moving apart atoms that were too close together until all atomic distances were > 1.5 Å. For model comparison between architectures, the structures were relaxed using the M3GNet interatomic potential only (no DFT relaxation). For the best GAN model, the structures were relaxed using the M3GNet interatomic potential followed by DFT calculations. The normalized interface energies, $\tilde{\gamma}_{\text{int}}$, of relaxed structures were calculated using the following equation:

$$\tilde{\gamma}_{\text{int}} = \frac{E_{\text{int}} - N(-4.78 \text{ eV/atom})}{A} \times (1.60218 \times 10^{-19} \text{ J/eV}) \quad (6)$$

where E_{int} is the total energy of the interface in eV, $N = 264$ is the total number of atoms, and A is the interface area. The interface energies are normalized such that structures with interface energy ≤ 0 J/m² are considered to be low-interface-energy structures. The normalization threshold of -4.78 eV/atom was chosen such that the low-interface-energy structures fall within a range of 0.4 J/m². For reference, the observation frequency of a grain boundary in aluminum metal decreases by 95 % when the grain boundary energy increases by around 0.35 J/m²⁵⁰.

Data Availability

The CryinGAN code and datasets used in this study are available at <https://github.com/ertekin-research-group/CryinGAN>.

Author Contributions

A.X.B.Y. and E.E conceived the idea. A.X.B.Y. wrote the code, carried out the calculations/experiments, and performed the analysis. E.E. supervised and guided the project. The manuscript was prepared by A.X.B.Y. Both authors reviewed and edited the manuscript.

Conflicts of interest

There are no conflicts to declare.

Acknowledgements

This work was supported by the Defense Advanced Research Projects Agency (DARPA) HR00112220028. We also acknowledge partial financial support from the U.S. Army CERL, awarded under Grant No. W9132T-19-2-0008. This work made use of the Illinois Campus Cluster, operated by the Illinois Campus Cluster Program (ICCP) in conjunction with the National Center for Supercomputing Applications (NCSA) and supported by the University of Illinois at Urbana-Champaign. This work also utilized the Hardware-Accelerated Learning (HAL) cluster⁸⁸, supported by the National Science Foundation's Major Research Instrumentation program, grant #1725729, as well as the University of Illinois at Urbana-Champaign. This work also used Bridges-2⁸⁹ at Pittsburgh Supercomputing Center through allocation MAT220011 from the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program, which is supported by National Science Foundation grants #2138259, #2138286, #2138307, #2137603, and #2138296.

Notes and references

- 1 D. M. Anstine and O. Isayev, *Journal of the American Chemical Society*, 2023, **145**, 8736–8750.
- 2 A. S. Fuhr and B. G. Sumpter, *Frontiers in Materials*, 2022, **9**, 1–13.
- 3 A. Kadurin, A. Aliper, A. Kazennov, P. Mamoshina, Q. Vanhaelen, K. Khrabrov and A. Zhavoronkov, *Oncotarget*, 2017, **8**, 10883–10890.
- 4 A. Kadurin, S. Nikolenko, K. Khrabrov, A. Aliper and A. Zhavoronkov, *Molecular Pharmaceutics*, 2017, **14**, 3098–3104.
- 5 S. Sinai, E. Kelsic, G. M. Church and M. A. Nowak, *Variational auto-encoding of protein sequences*, 2018.
- 6 J. Hoffmann, L. Maestrati, Y. Sawada, J. Tang, J. M. Selier and Y. Bengio, *Data-Driven Approach to Encoding and Decoding 3-D Crystal Structures*, 2019.
- 7 J. Noh, J. Kim, H. S. Stein, B. Sanchez-Lengeling, J. M. Gregoire, A. Aspuru-Guzik and Y. Jung, *Matter*, 2019, **1**, 1370–1384.
- 8 A. Noura, N. Sokolovska and J.-C. Crivello, *CrystalGAN: Learning to Discover Crystallographic Structures with Generative Adversarial Networks*, 2019.
- 9 Z. Ren, S. I. P. Tian, J. Noh, F. Oviedo, G. Xing, J. Li, Q. Liang, R. Zhu, A. G. Aberle, S. Sun, X. Wang, Y. Liu, Q. Li, S. Jayavelu, K. Hippalgaonkar, Y. Jung and T. Buonassisi, *Matter*, 2022, **5**, 314–335.

- 10 Y. Zhao, M. Al-Fahdi, M. Hu, E. M. D. Siriwardane, Y. Song, A. Nasiri and J. Hu, Advanced Science, 2021, **8**, 2100566.
- 11 C. J. Court, B. Yildirim, A. Jain and J. M. Cole, Journal of Chemical Information and Modeling, 2020, **60**, 4518–4535.
- 12 T. Long, N. M. Fortunato, I. Opahle, Y. Zhang, I. Samatharakis, C. Shen, O. Gutfleisch and H. Zhang, npj Computational Materials, 2021, **7**, 66.
- 13 M. F. Thorpe and L. Tichy, Properties and applications of amorphous materials, Springer Dordrecht, 2001, vol. 9.
- 14 T. Yang, Y. L. Zhao, W. P. Li, C. Y. Yu, J. H. Luan, D. Y. Lin, L. Fan, Z. B. Jiao, W. H. Liu, X. J. Liu, J. J. Kai, J. C. Huang and C. T. Liu, Science, 2020, **369**, 427–432.
- 15 Y. Xie, J. Cai, Y. Wu, X. Hao, Z. Bian, S. Niu, X. Yin, Z. Pei, D. Sun, Z. Zhu, Z. Lu, D. Niu and G. Wang, ACS Materials Letters, 2021, **3**, 1738–1745.
- 16 Y. G. Chung, J. Camp, M. Haranczyk, B. J. Sikora, W. Bury, V. Krungleviciute, T. Yildirim, O. K. Farha, D. S. Sholl and R. Q. Snurr, Chemistry of Materials, 2014, **26**, 6185–6192.
- 17 R. Thyagarajan and D. S. Sholl, Chemistry of Materials, 2020, **32**, 8020–8033.
- 18 H. Zheng, E. Sivonxay and K. Persson, MRS Spring Meeting, 2023.
- 19 M. Kilgour, N. Gastellu, D. Y. T. Hui, Y. Bengio and L. Simine, The Journal of Physical Chemistry Letters, 2020, **11**, 8532–8537.
- 20 V. S. C. Kolluru, D. G. Unruh, J. T. Paul and M. K. Chan, APS March Meeting, 2023.
- 21 J. Guo, A. Mannodi-Kanakkithodi, F. G. Sen, E. Schwenker, E. S. Barnard, A. Munshi, W. Sampath, M. K. Y. Chan and R. F. Klie, Applied Physics Letters, 2019, **115**, 1–5.
- 22 E. Schwenker, V. S. C. Kolluru, J. Guo, R. Zhang, X. Hu, Q. Li, J. T. Paul, M. C. Hersam, V. P. Dravid, R. Klie, J. R. Guest and M. K. Y. Chan, Small, 2022, **18**, 2102960.
- 23 M. Comin and L. J. Lewis, Physical Review B, 2019, **100**, 094107.
- 24 B. Kim, S. Lee and J. Kim, Science Advances, 2020, **6**, eaax9324.
- 25 S. Kim, J. Noh, G. H. Gu, A. Aspuru-Guzik and Y. Jung, ACS Central Science, 2020, **6**, 1412–1420.
- 26 T. Xie, X. Fu, O.-E. Ganea, R. Barzilay and T. Jaakkola, International Conference on Learning Representations, 2022.
- 27 M. Alverson, S. G. Baird, R. Murdock, S.-H. Ho, J. Johnson and T. D. Sparks, Digital Discovery, 2024, **3**, 62–80.
- 28 M. Yang, K. Cho, A. Merchant, P. Abbeel, D. Schuurmans, I. Mordatch and E. D. Cubuk, Scalable Diffusion for Materials Generation, 2023.
- 29 C. Zeni, R. Pinsler, D. Zügner, A. Fowler, M. Horton, X. Fu, S. Shysheya, J. Crabbé, L. Sun, J. Smith, B. Nguyen, H. Schulz, S. Lewis, C.-W. Huang, Z. Lu, Y. Zhou, H. Yang, H. Hao, J. Li, R. Tomioka and T. Xie, MatterGen: a generative model for inorganic materials design, 2024.
- 30 D. Flam-Shepherd and A. Aspuru-Guzik, Language models can generate molecules, materials, and protein binding sites directly in three dimensions as XYZ, CIF, and PDB files, 2023.
- 31 L. M. Antunes, K. T. Butler and R. Grau-Crespo, Crystal Structure Generation with Autoregressive Large Language Modeling, 2024.
- 32 N. Gruver, A. Sriram, A. Madotto, A. G. Wilson, C. L. Zitnick and Z. Ulissi, Fine-Tuned Language Models Generate Stable Inorganic Materials as Text, 2024.
- 33 C. R. Qi, H. Su, K. Mo and L. J. Guibas, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
- 34 Y. Lyu, X. Wu, K. Wang, Z. Feng, T. Cheng, Y. Liu, M. Wang, R. Chen, L. Xu, J. Zhou, Y. Lu and B. Guo, Advanced Energy Materials, 2021, **11**, 2000982.
- 35 J. Liang, X. Li, S. Wang, K. R. Adair, W. Li, Y. Zhao, C. Wang, Y. Hu, L. Zhang, S. Zhao, S. Lu, H. Huang, R. Li, Y. Mo and X. Sun, Journal of the American Chemical Society, 2020, **142**, 7012–7022.
- 36 X. Li, J. Liang, X. Yang, K. R. Adair, C. Wang, F. Zhao and X. Sun, Energy & Environmental Science, 2020, **13**, 1429–1461.
- 37 C. Wang, J. Liang, J. T. Kim and X. Sun, Science Advances, 2022, **8**, eadc9516.
- 38 B. Zahiri, A. Patra, C. Kiggins, A. X. B. Yong, E. Ertekin, J. B. Cook and P. V. Braun, Nature Materials, 2021, **20**, 1392–1400.
- 39 T. Yang, Y. L. Zhao, W. P. Li, C. Y. Yu, J. H. Luan, D. Y. Lin, L. Fan, Z. B. Jiao, W. H. Liu, X. J. Liu, J. J. Kai, J. C. Huang and C. T. Liu, Science, 2020, **369**, 427–432.
- 40 G. H. Balbus, J. Kappacher, D. J. Sprouster, F. Wang, J. Shin, Y. M. Eggeler, T. J. Rupert, J. R. Trelewicz, D. Kiener, V. Maier-Kiener and D. S. Gianola, Acta Materialia, 2021, **215**, 116973.
- 41 D. Hudry, R. Popescu, D. Busko, M. Diaz-Lopez, M. Abeykoon, P. Bordet, D. Gerthsen, I. A. Howard and B. S. Richards, Journal of Materials Chemistry C, 2019, **7**, 1164–1172.
- 42 S. R. Spurgeon, T. C. Kaspar, V. Shutthanandan, J. Gigax, L. Shao and M. Sassi, Advanced Materials Interfaces, 2020, **7**, 1901944.
- 43 B. Gao, R. Jalem and Y. Tateyama, ACS Applied Materials & Interfaces, 2021, **13**, 11765–11773.
- 44 S. He, K. Chen, M. Saunders, J. Li, C. Q. Cui and S. P. Jiang, Journal of The Electrochemical Society, 2017, **164**, F1437.
- 45 J. Wang, M. Deng, Y. Chen, X. Liu, W. Ke, D. Li, W. Dai and K. He, Materials Chemistry and Physics, 2020, **244**, 122733.
- 46 Y.-D. Guo, Z.-J. Yang, Q.-H. Gao and W. Dai, Physica B: Condensed Matter, 2008, **403**, 2367–2371.
- 47 A. Masias, N. Felten, R. Garcia-Mendez, J. Wolfenstine and J. Sakamoto, Journal of Materials Science, 2019, **54**, 2585–2600.
- 48 C. Chen and S. P. Ong, Nature Computational Science, 2022, **2**, 718–728.
- 49 B. Deng, P. Zhong, K. Jun, J. Riebesell, K. Han, C. J. Bartel and G. Ceder, Nature Machine Intelligence, 2023, **5**, 1031–1041.
- 50 E. A. Holm, G. S. Rohrer, S. M. Foiles, A. D. Rollett, H. M. Miller and D. L. Olmsted, Acta Materialia, 2011, **59**, 5250–5256.

- 51 M. Arjovsky, S. Chintala and L. Bottou, *Proceedings of the 34th International Conference on Machine Learning*, 2017.
- 52 I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin and A. Courville, *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017.
- 53 H. Wang, Z. Jiang, L. Yi, K. Mo, H. Su and L. J. Guibas, *Rethinking Sampling in 3D Point Cloud Generative Adversarial Networks*, 2020.
- 54 T. Xie and J. C. Grossman, *Physical Review Letters*, 2018, **120**, 145301.
- 55 H. Thanh-Tung and T. Tran, *2020 International Joint Conference on Neural Networks (IJCNN)*, 2020, pp. 1–10.
- 56 N. E. R. Zimmermann and A. Jain, *RSC Advances*, 2020, **10**, 6063–6081.
- 57 Y. Zhao, E. M. D. Siriwardane, Z. Wu, N. Fu, M. Al-Fahdi, M. Hu and J. Hu, *npj Computational Materials*, 2023, **9**, 38.
- 58 R. Jiao, W. Huang, Y. Liu, D. Zhao and Y. Liu, *Space Group Constrained Crystal Generation*, 2024.
- 59 V. L. Deringer, N. Bernstein, G. Csányi, C. Ben Mahmoud, M. Ceriotti, M. Wilson, D. A. Drabold and S. R. Elliott, *Nature*, 2021, **589**, 59–64.
- 60 C. Vignac, N. Osman, L. Toni and P. Frossard, *MiDi: Mixed Graph and 3D Denoising Diffusion for Molecule Generation*, 2023.
- 61 B. Samanta, A. De, G. Jana, V. Gázquez, P. Chattaraj, N. Ganguly and M. Gomez-Rodriguez, *Journal of Machine Learning Research*, 2020, **21**, 1–33.
- 62 S. De, A. P. Bartók, G. Csányi and M. Ceriotti, *Physical Chemistry Chemical Physics*, 2016, **18**, 13754–13769.
- 63 J. Behler, *The Journal of Chemical Physics*, 2011, **134**, 1–13.
- 64 V. Fung, S. Jia, J. Zhang, S. Bi, J. Yin and P. Ganesh, *Machine Learning: Science and Technology*, 2022, **3**, 045018.
- 65 L. Wasserman, in *The Bootstrap*, ed. L. Wasserman, Springer New York, New York, NY, 2004, pp. 107–118.
- 66 G. Kresse and J. Hafner, *Physical Review B*, 1993, **47**, 558–561.
- 67 G. Kresse and J. Hafner, *Physical Review B*, 1994, **49**, 14251–14269.
- 68 G. Kresse and J. Furthmüller, *Physical Review B*, 1996, **54**, 11169–11186.
- 69 G. Kresse and J. Furthmüller, *Computational Materials Science*, 1996, **6**, 15–50.
- 70 P. E. Blöchl, *Physical Review B*, 1994, **50**, 17953–17979.
- 71 G. Kresse and D. Joubert, *Physical Review B*, 1999, **59**, 1758–1775.
- 72 J. P. Perdew, K. Burke and M. Ernzerhof, *Physical Review Letters*, 1996, **77**, 3865–3868.
- 73 S. L. Dudarev, G. A. Botton, S. Y. Savrasov, C. J. Humphreys and A. P. Sutton, *Physical Review B*, 1998, **57**, 1505–1509.
- 74 C. Wang, J. Liang, M. Jiang, X. Li, S. Mukherjee, K. Adair, M. Zheng, Y. Zhao, F. Zhao, S. Zhang, R. Li, H. Huang, S. Zhao, L. Zhang, S. Lu, C. V. Singh and X. Sun, *Nano Energy*, 2020, **76**, 105015.
- 75 K. Mathew, A. K. Singh, J. J. Gabriel, K. Choudhary, S. B. Sinnott, A. V. Davydov, F. Tavazza and R. G. Hennig, *Computational Materials Science*, 2016, **122**, 183–190.
- 76 A. Zur and T. C. McGill, *Journal of Applied Physics*, 1984, **55**, 378–386.
- 77 P. W. Tasker, *Journal of Physics C: Solid State Physics*, 1979, **12**, 4977–4984.
- 78 J. D. Gale and A. L. Rohl, *Molecular Simulation*, 2003, **29**, 291–341.
- 79 P. J. Bouwman, B. A. Boukamp, H. J. M. Bouwmeester and P. H. L. Notten, *Journal of The Electrochemical Society*, 2002, **149**, A699.
- 80 Y. Wang, J. Lv, L. Zhu and Y. Ma, *Physical Review B*, 2010, **82**, 094116.
- 81 Y. Wang, J. Lv, L. Zhu and Y. Ma, *Computer Physics Communications*, 2012, **183**, 2063–2070.
- 82 B. Gao, P. Gao, S. Lu, J. Lv, Y. Wang and Y. Ma, *Science Bulletin*, 2019, **64**, 301–309.
- 83 E. J. Cheng, N. J. Taylor, J. Wolfenstine and J. Sakamoto, *Journal of Asian Ceramic Societies*, 2017, **5**, 113–117.
- 84 M. Jiang, S. Mukherjee, Z. W. Chen, L. X. Chen, M. L. Li, H. Y. Xiao, C. Gao and C. V. Singh, *Physical Chemistry Chemical Physics*, 2020, **22**, 22758–22767.
- 85 D. P. Kingma and J. Ba, *International Conference on Learning Representations*, 2015.
- 86 *Conjugate Gradient Methods*, ed. J. Nocedal and S. J. Wright, Springer New York, New York, NY, 2006, pp. 101–134.
- 87 A. Hjorth Larsen, J. Jørgen Mortensen, J. Blomqvist, I. E. Castelli, R. Christensen, M. Dułak, J. Friis, M. N. Groves, B. Hammer, C. Hargus, E. D. Hermes, P. C. Jennings, P. Bjerre Jensen, J. Kermode, J. R. Kitchin, E. Leonhard Kolsbjerg, J. Kubal, K. Kaasbjerg, S. Lysgaard, J. Bergmann Maronsson, T. Maxson, T. Olsen, L. Pastewka, A. Peterson, C. Rossgaard, J. Schiøtz, O. Schütt, M. Strange, K. S. Thygesen, T. Vegge, L. Vilhelmsen, M. Walter, Z. Zeng and K. W. Jacobsen, *Journal of Physics: Condensed Matter*, 2017, **29**, 273002.
- 88 V. Kindratenko, D. Mu, Y. Zhan, J. Maloney, S. H. Hashemi, B. Rabe, K. Xu, R. Campbell, J. Peng and W. Gropp, *Practice and Experience in Advanced Research Computing*, New York, NY, USA, 2020, p. 41–48.
- 89 S. T. Brown, P. Buitrago, E. Hanna, S. Sanielevici, R. Scibek and N. A. Nystrom, *Practice and Experience in Advanced Research Computing*, New York, NY, USA, 2021.

Supplementary Information

CryinGAN: Design and evaluation of point-cloud-based generative adversarial networks using disordered materials – application to $\text{Li}_3\text{ScCl}_6\text{-LiCoO}_2$ battery interfaces

Adrian Xiao Bin Yong^{*1, 3} and Elif Ertekin^{*2, 3}

¹Department of Materials Science and Engineering, University of Illinois Urbana-Champaign, 1304 West Green Street, Urbana 61801 IL, USA

²Department of Mechanical Science and Engineering, University of Illinois Urbana-Champaign, 1206 West Green Street, Urbana 61801 IL, USA

³Materials Research Laboratory, University of Illinois Urbana-Champaign, 104 South Goodwin Avenue, Urbana 61801 IL, USA

^{*}Corresponding authors; Emails: axyong2@illinois.edu, ertekin@illinois.edu

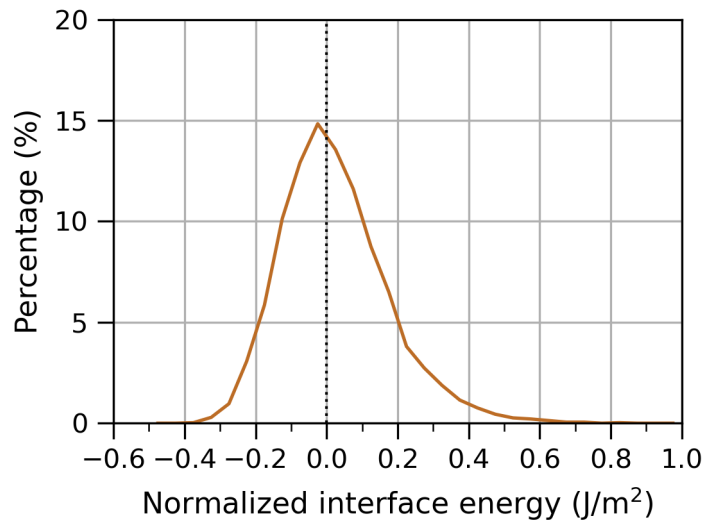


Figure S1: Normalized interface energy distribution of randomly generated structures. All structures were relaxed using M3GNet followed by DFT calculations, and the energies shown are with respect to DFT-calculated energies.

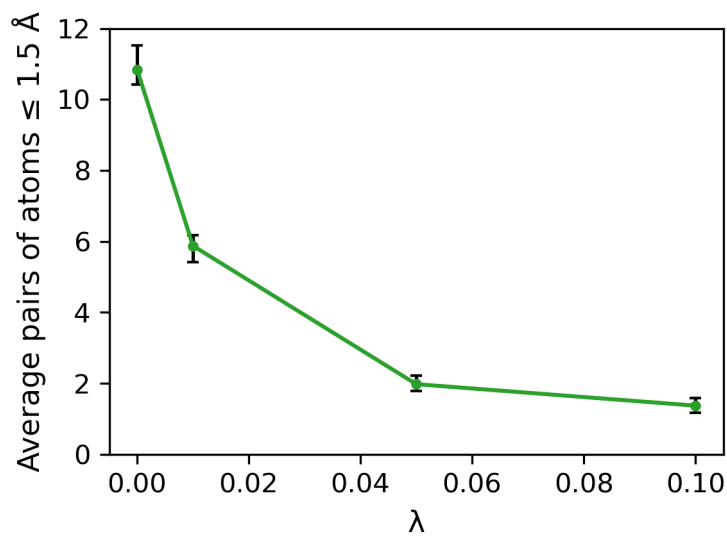


Figure S2: Average number of unique pairs of atoms with bond distance ≤ 1.5 Å for structures generated using CryinGAN trained with different λ values. As λ increases, fewer atoms are generated too close to each other. For each λ value, 3 separate models were trained, 1000 structures were generated using each model, and the counts were averaged across the models.

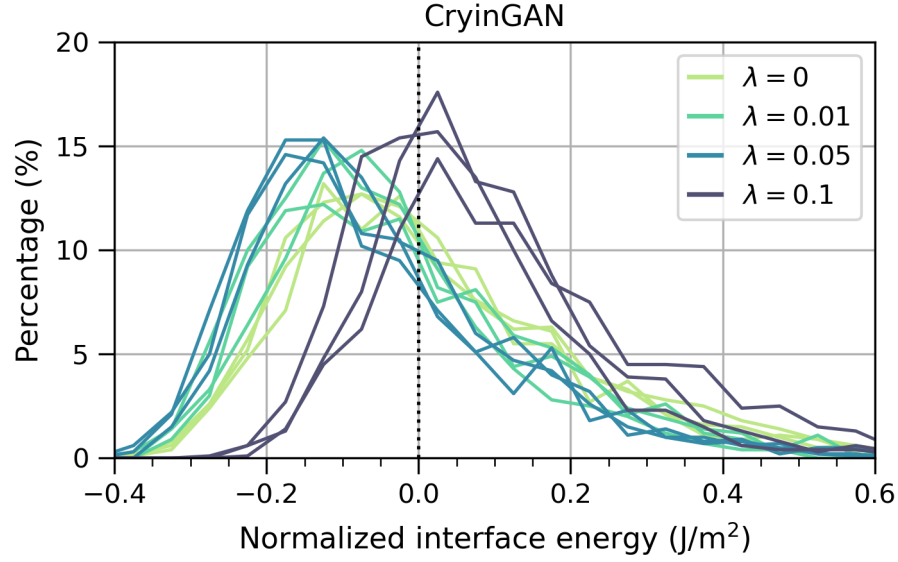


Figure S3: Normalized interface energy distributions of structures generated using CryinGAN trained with different λ values (0, 0.01, 0.05, and 0.1). For each λ value, 3 separate models were trained. The generated structures were relaxed using M3GNet, and the interface energies shown are based on M3GNet-calculated energies.

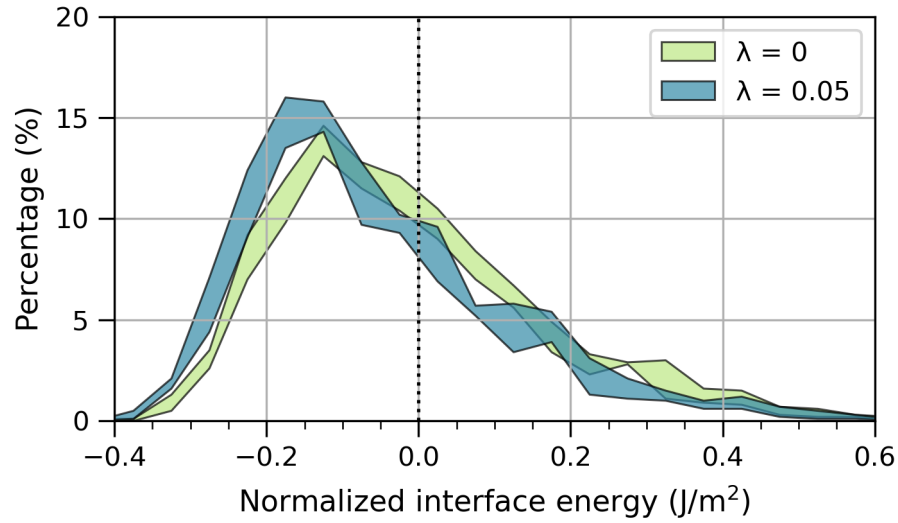


Figure S4: Normalized interface energy distributions of structures generated using CryinGAN trained with $\lambda = 0$ (200,000 epochs) and $\lambda = 0.05$ (100,000 epochs). The generated structures were relaxed using M3GNet, and the interface energies shown are based on M3GNet-calculated energies. For each λ value, and the spread of the interface energy distributions is indicated by the shading. CryinGAN runs approximately two times faster if the bond distance discriminator is not used ($\lambda = 0$), but the use of the bond distance discriminator still yields a lower interface energy distribution for the same amount of training time.

CryinGAN-comb discriminator

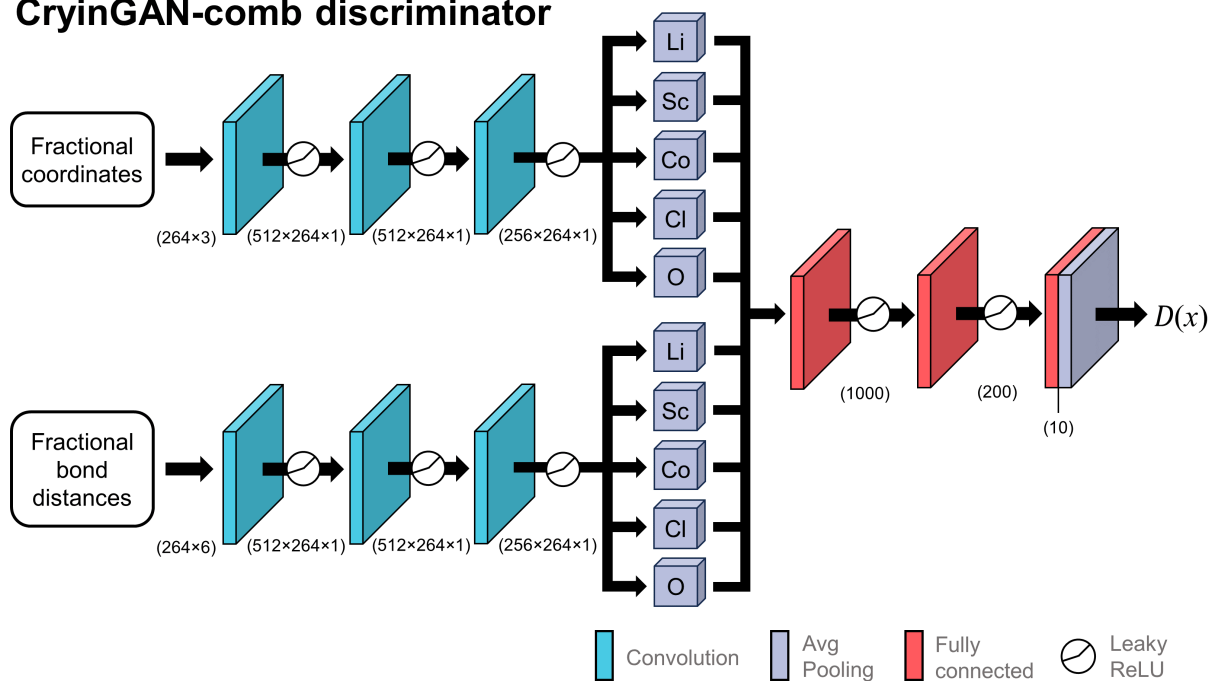


Figure S5: Discriminator architecture of CryinGAN-comb. The output of the two discriminators of CryinGAN are combined after pooling.

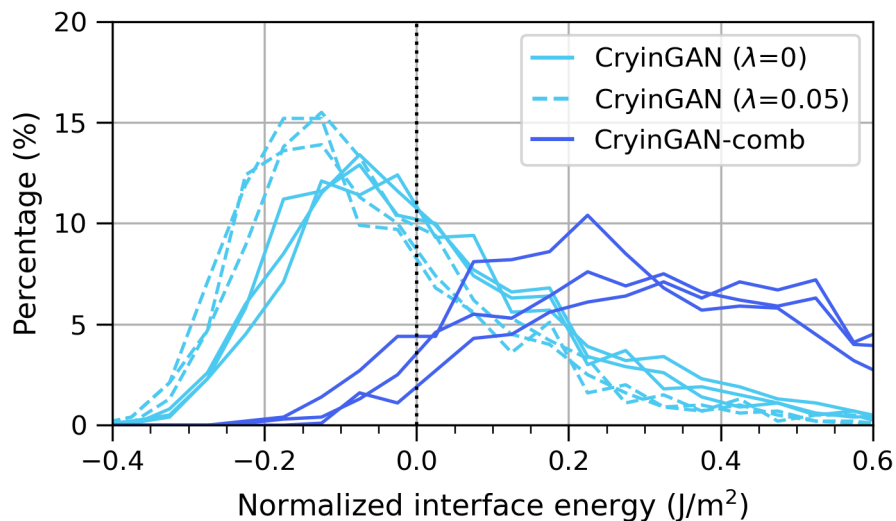
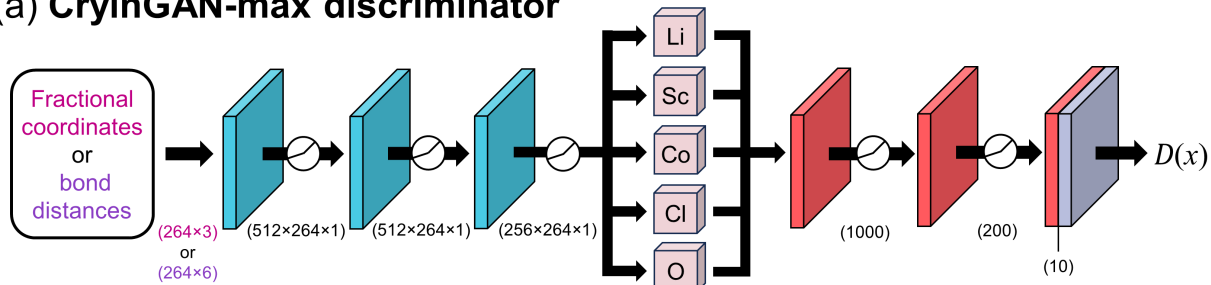


Figure S6: Normalized interface energy distributions of structures generated using CryinGAN ($\lambda = 0, 0.05$) and CryinGAN-comb. For each model configuration, 3 separate models were trained. All structures were relaxed using M3GNet, and the interface energies shown are based on M3GNet-calculated energies.

(a) CryinGAN-max discriminator



(b) CryinGAN-mix discriminator

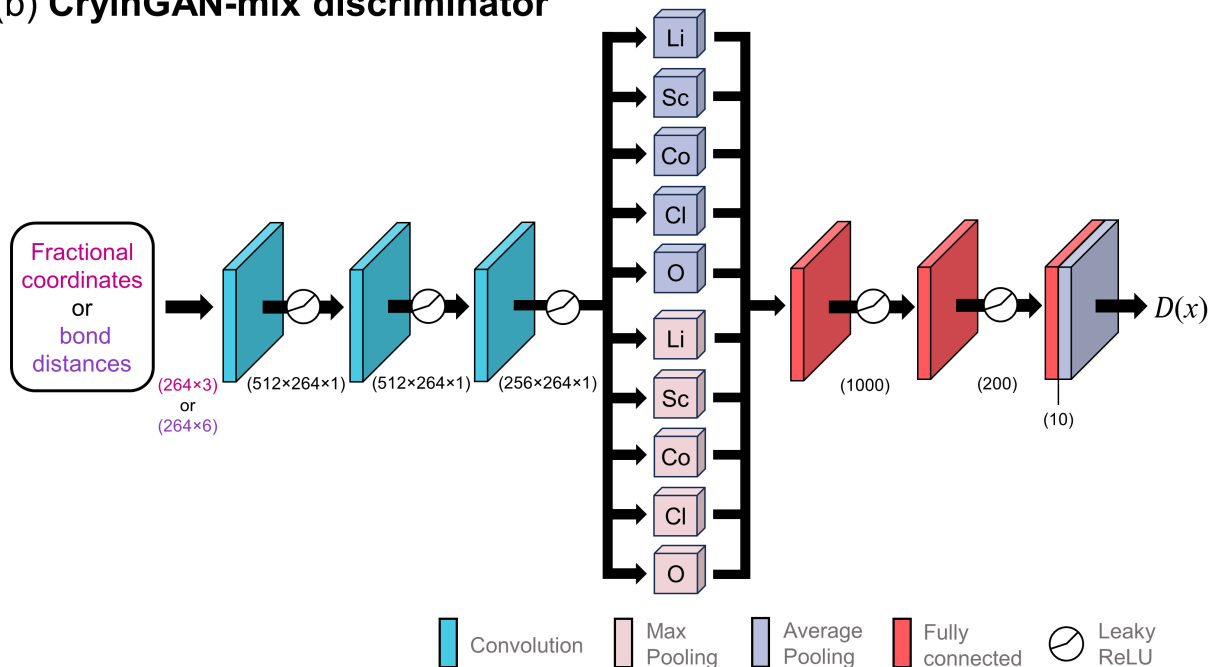


Figure S7: Discriminator architectures of (a) CryinGAN-max and (b) CryinGAN-mix. When the bond distance discriminator is not used, max/mix pooling is applied to the fractional coordinate discriminator. When the bond distance discriminator is used, max/mix pooling is applied to the bond distance discriminator, and average pooling is applied to the fractional coordinate discriminator.

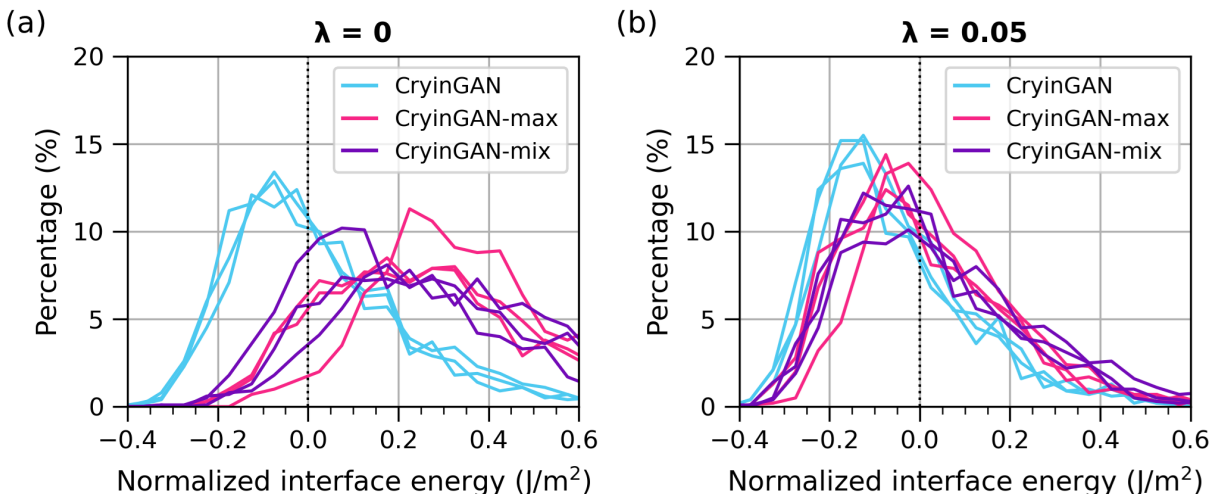


Figure S8: Normalized interface energy distributions of structures generated using CryinGAN (average pooling), CryinGAN-max (max pooling), and CryinGAN-mix (mix pooling). (a) Models without the bond distance discriminator ($\lambda = 0$), where pooling is varied for the fractional coordinate discriminator. (b) Models with the bond distance discriminator ($\lambda = 0.05$), where pooling is varied for the bond distance discriminator only (average pooling is used for the fractional coordinate discriminator). For each model configuration, 3 separate models were trained. All structures were relaxed using M3GNet, and the interface energies shown are based on M3GNet-calculated energies.

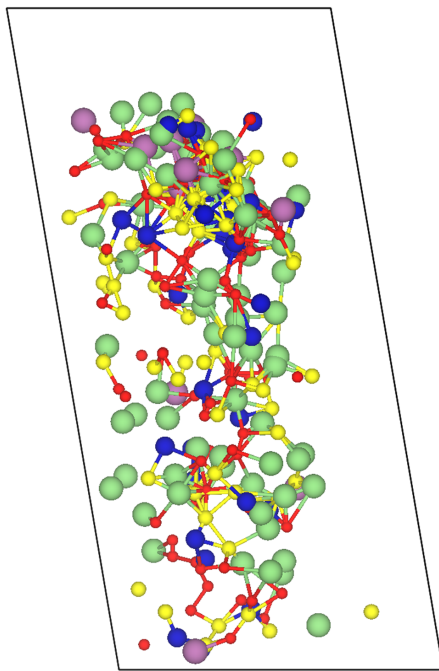


Figure S9: Example generated structure of a GAN model using Crystal Graph Convolutional Neural Networks (CGCNN) as the discriminator. The GAN was unable to learn to generate a meaningful structure.

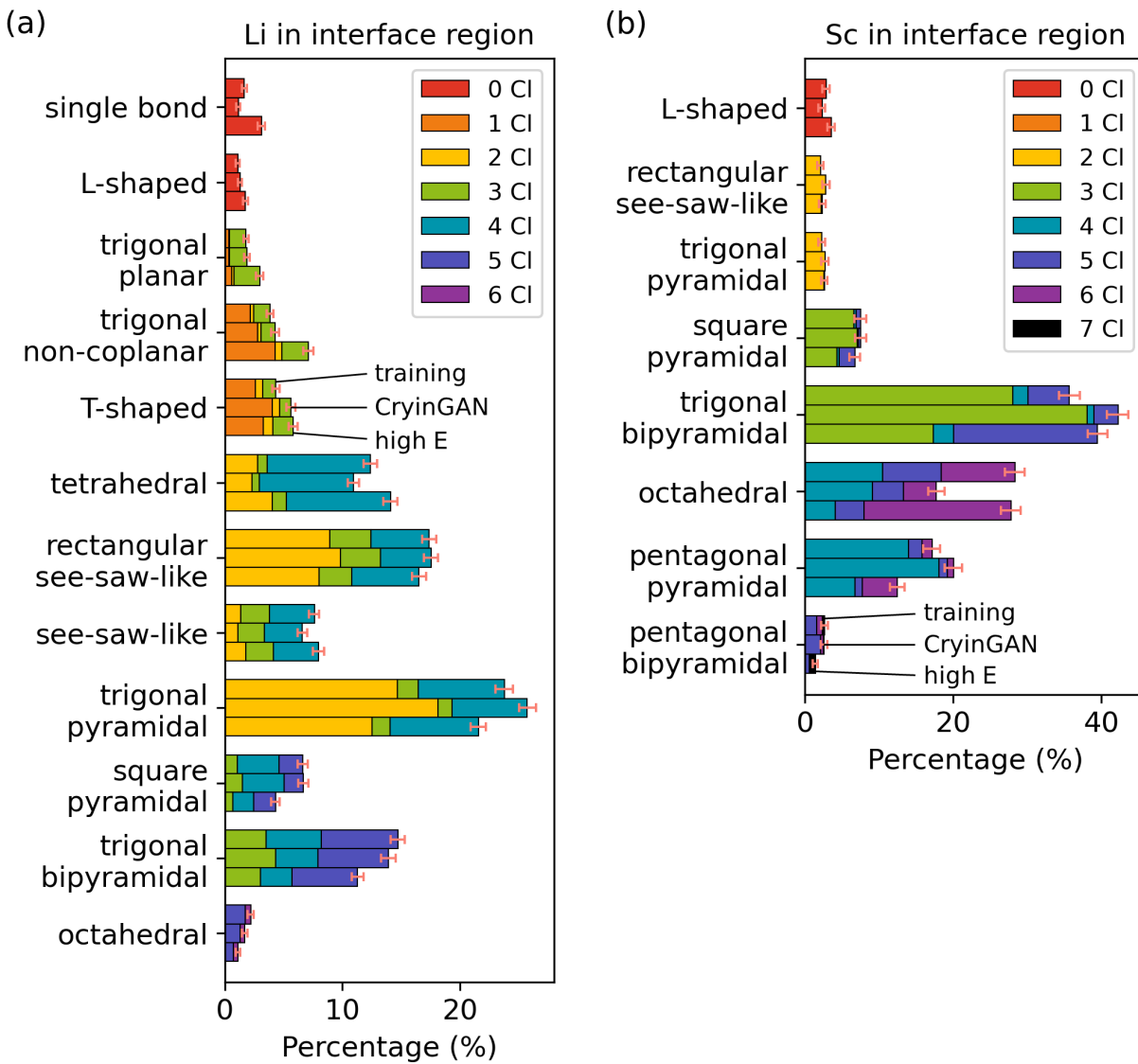


Figure S10: Coordination motif distributions of (a) Li and (b) Sc in the interface region, where each coordination motif is further subdivided based on the number of Cl bonds. The distributions of three datasets are shown: (1) training structures with low interface energy, (2) CryinGAN-generated structures, and (3) structures with high interface energy. All structures were relaxed using M3GNet followed by DFT calculations. Error bars represent 95 % bootstrap confidence intervals.

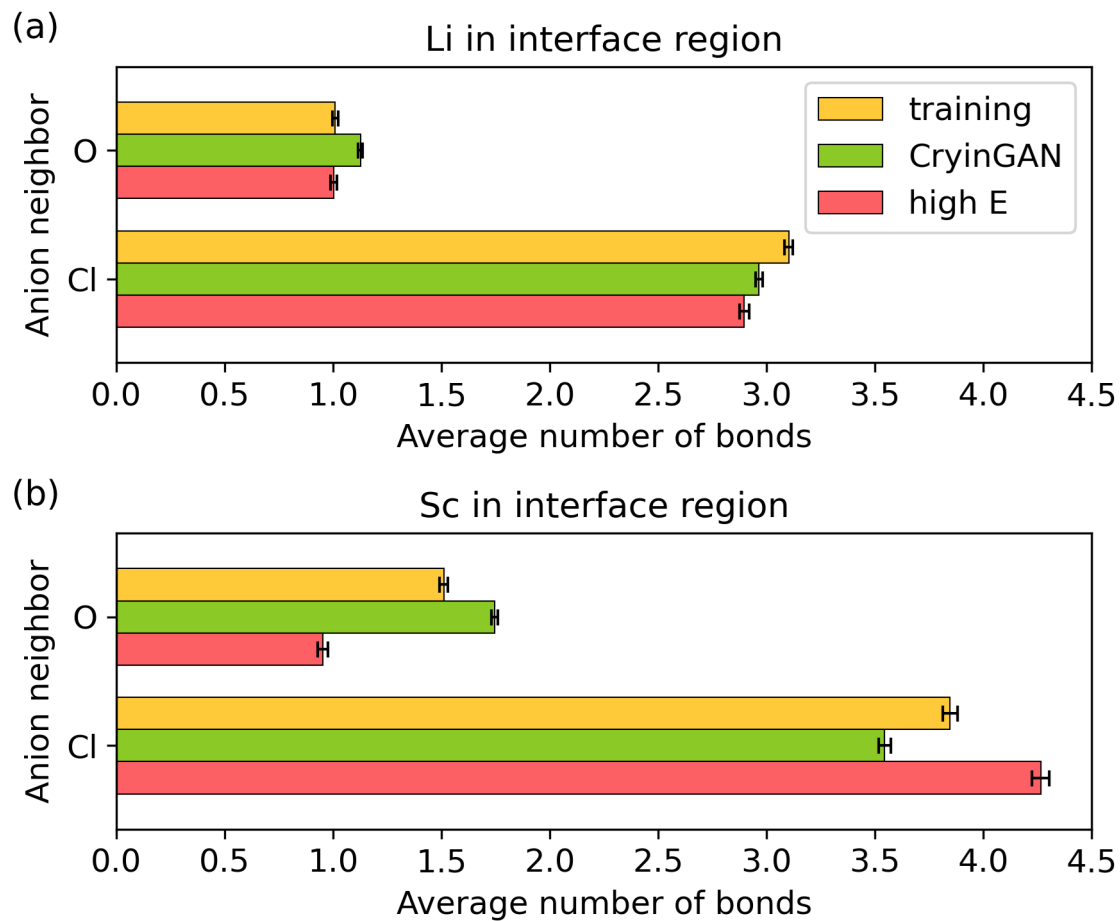


Figure S11: Histograms of the average number of O and Cl bonds for (a) Li and (b) Sc in the interface region. Error bars represent 95 % bootstrap confidence intervals.

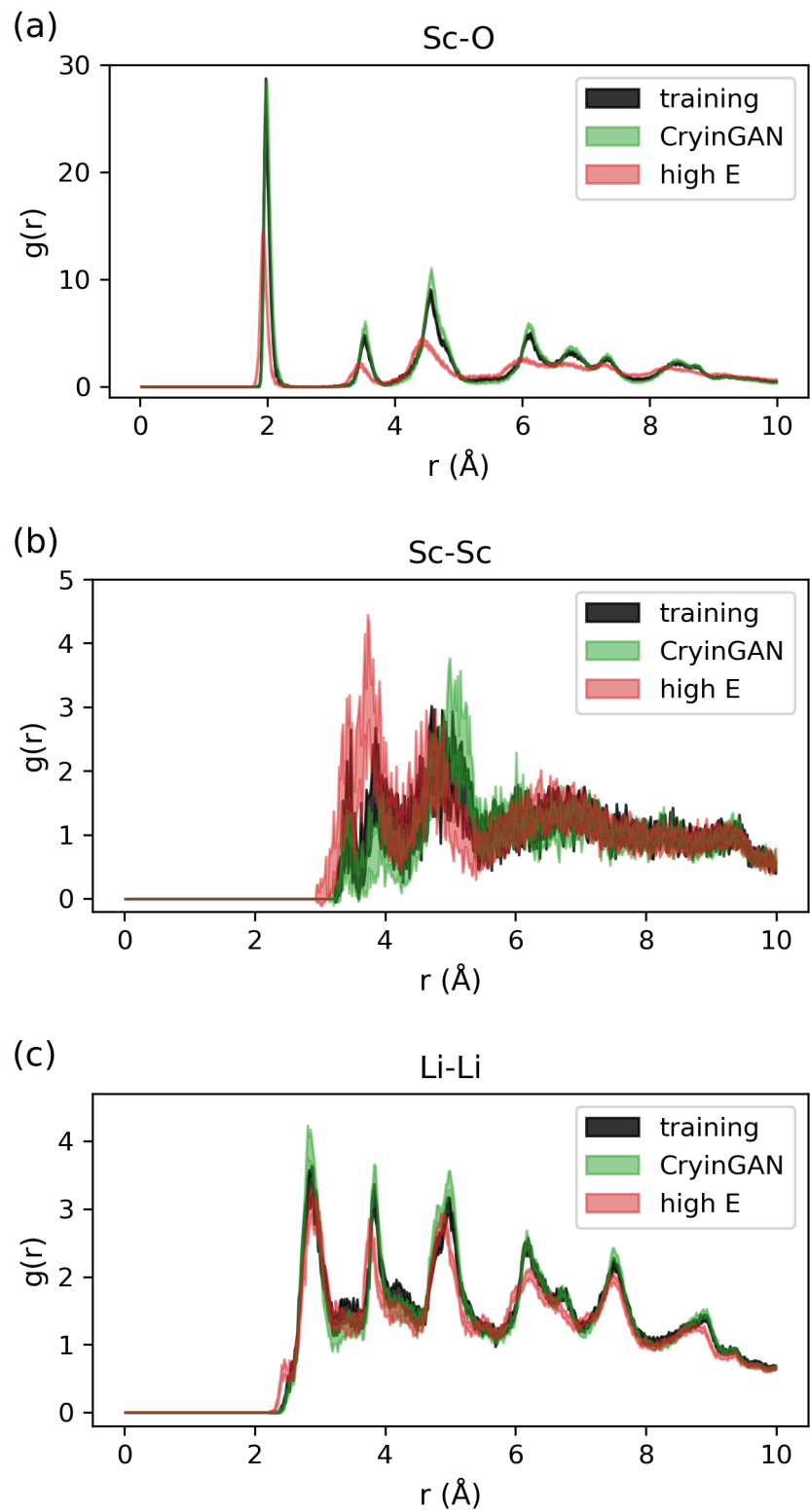


Figure S12: Radial distribution functions of atoms in the interface region for (a) Sc-O, (b) Sc-Sc, and (c) Li-Li. The thickness of each curve represents the 95 % bootstrap confidence interval.

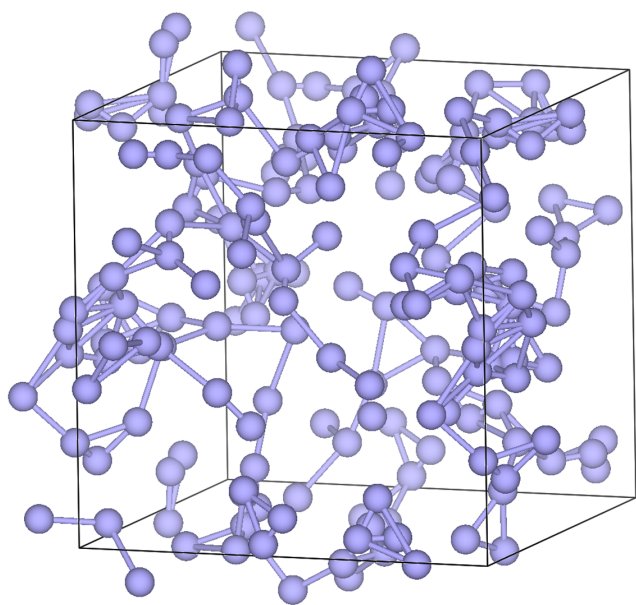


Figure S13: Example generated structure of a CryinGAN model trained on amorphous silicon structures.

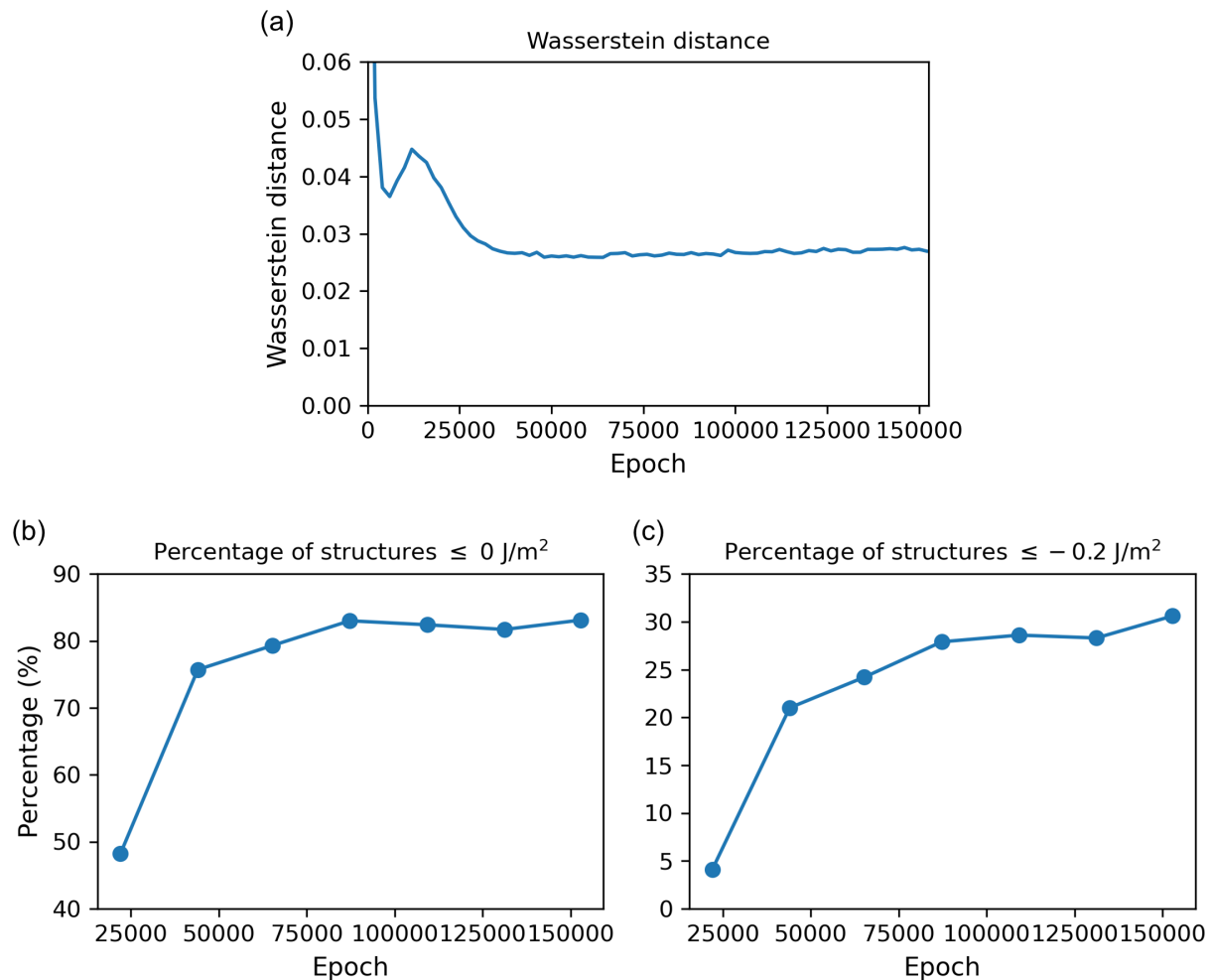


Figure S14: (a) Wasserstein distance as a function of epoch for CryinGAN trained with $\lambda = 0.05$. Note that the Wasserstein distance shown here does not include the gradient penalty term of the Wasserstein loss function. (b) Percentage of relaxed structures with normalized interface energy $\leq 0 \text{ J/m}^2$ as a function of epoch. (c) Percentage of relaxed structures with normalized interface energy $\leq -0.2 \text{ J/m}^2$ as a function of epoch. Structures were generated using a CryinGAN model (trained for a certain number of epochs) and relaxed using the M3GNet interatomic potential. Each data point was calculated using 1,000 relaxed structures.

Table S1: Configurations of the interfaces calculated. The slab orientation, number of layers, and average lattice mismatch between any given two slabs are listed. M represents the transformation matrix used to transform the lattice vectors of a given slab surface (\vec{u}, \vec{v}) into the superlattice vectors of the interface (\vec{u}_s, \vec{v}_s) , according to the relation $(\vec{u}_s, \vec{v}_s) = M \cdot (\vec{u}, \vec{v})$. Note that the number of layers of the $\text{Li}_3\text{ScCl}_6(100)$ slab is indicated for the randomly generated $\text{LiCoO}_2(110)$ - $\text{Li}_3\text{ScCl}_6(100)$, and does not include the atoms randomly generated in the interface region.

Slab 1	number of layers	M	Slab 2	number of layers	M	average mismatch (%)
$\text{Li}_2\text{O}(100)$	9	$\begin{pmatrix} 1 & 2 \\ -4 & 2 \end{pmatrix}$	$\text{LiCl}(100)$	4	$\begin{pmatrix} 2 & 0 \\ 0 & 4 \end{pmatrix}$	0.545
$\text{Li}_2\text{O}(110)$	4	$\begin{pmatrix} 1 & 0 \\ 0 & 3 \end{pmatrix}$	$\text{Li}(100)$	6	$\begin{pmatrix} 1 & 0 \\ 0 & 3 \end{pmatrix}$	3.00
$\text{Li}_2\text{O}(110)$	4	$\begin{pmatrix} -2 & 1 \\ 3 & 2 \end{pmatrix}$	$\text{LiCl}(100)$	4	$\begin{pmatrix} 2 & 1 \\ -2 & 3 \end{pmatrix}$	2.13
$\text{Li}_2\text{O}(110)$	4	$\begin{pmatrix} -2 & 1 \\ 3 & 2 \end{pmatrix}$	$\text{MgS}(100)$	4	$\begin{pmatrix} 2 & 1 \\ -2 & 3 \end{pmatrix}$	2.17
$\text{LiCoO}_2(110)$	4	$\begin{pmatrix} 1 & 2 \\ -3 & 3 \end{pmatrix}$	$\text{Li}_3\text{ScCl}_6(100)$	9	$\begin{pmatrix} 1 & 1 \\ -2 & 1 \end{pmatrix}$	2.17

Table S2: Validation set mean absolute errors (MAEs) and losses of M3GNet models trained with different learning rates and batch sizes. The loss, L , is as defined in the main text. For each model, the epoch with the smallest L is shown. The model with the smallest L is highlighted in bold font. Note that we also trained a model with a learning rate of 0.005 and a batch size of 4, but we found the training to be unstable so the results are omitted here.

Learning rate	Batch size	Energy MAE (meV/atom)	Force MAE (meV/Å)	Stress MAE (GPa)	Loss, L
0.001	4	2.75	20.9	0.0151	0.0251
0.0005	4	2.82	22.2	0.0179	0.0268
0.001	2	3.45	21.9	0.0203	0.0273
0.001	6	2.88	21.3	0.0156	0.0257

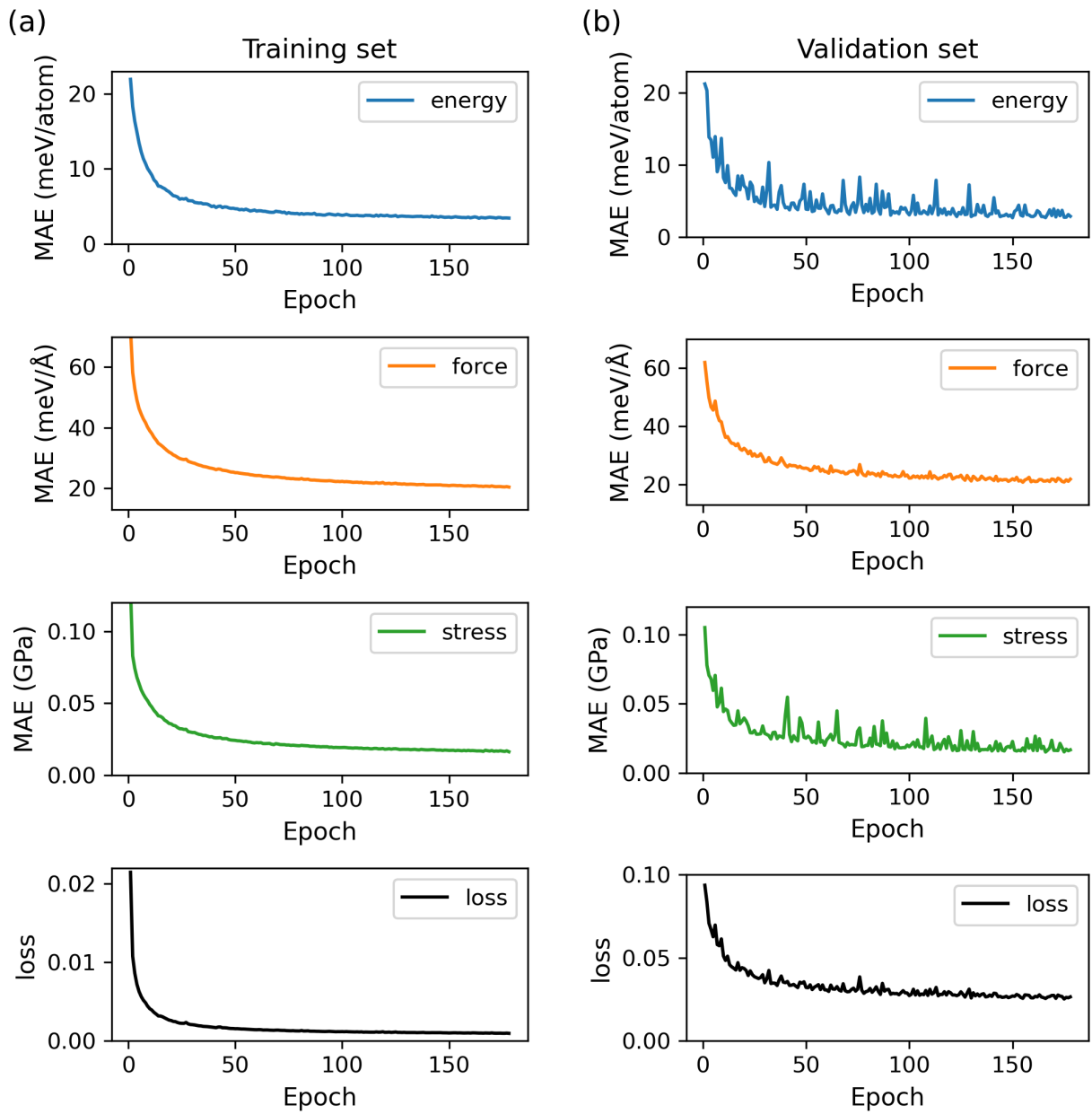


Figure S15: M3GNet training curves of the (a) training set and (b) validation set for the model trained with a learning rate of 0.001 and a batch size of 4. The mean absolute errors (MAEs) for energy, force, and stress, as well as the loss function are plotted against training epoch.