

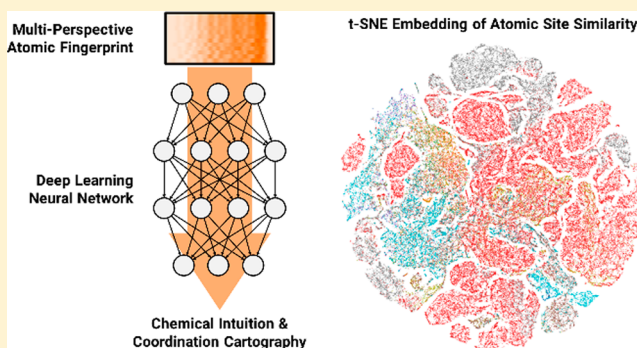
# Crystal Structure Prediction via Deep Learning

Kevin Ryan,\* Jeff Lengyel, and Michael Shatruk\*

Department of Chemistry and Biochemistry, Florida State University, Tallahassee, Florida 32306, United States

## Supporting Information

**ABSTRACT:** We demonstrate the application of deep neural networks as a machine-learning tool for the analysis of a large collection of crystallographic data contained in the crystal structure repositories. Using input data in the form of multiperspective atomic fingerprints, which describe coordination topology around unique crystallographic sites, we show that the neural-network model can be trained to effectively distinguish chemical elements based on the topology of their crystallographic environment. The model also identifies structurally similar atomic sites in the entire data set of ~50000 crystal structures, essentially uncovering trends that reflect the periodic table of elements. The trained model was used to analyze templates derived from the known crystal structures in order to predict the likelihood of forming new compounds that could be generated by placing elements into these structural templates in a combinatorial fashion. Statistical analysis of predictive performance of the neural-network model, which was applied to a test set of structures never seen by the model during training, indicates its ability to predict known elemental compositions with a high likelihood of success. In ~30% of cases, the known compositions were found among the top 10 most likely candidates proposed by the model. These results suggest that the approach developed in this work can be used to effectively guide the synthetic efforts in the discovery of new materials, especially in the case of systems composed of three or more chemical elements.



## INTRODUCTION

Crystal structure prediction has been the grand challenge for solid-state chemists for many decades.<sup>1</sup> From the early simplistic attempts to justify the structure of binary ionic compounds using the radius ratio rule,<sup>2–6</sup> to the more elaborate structure correlation diagrams mapping out important features that dictate the formation of binary and ternary structures,<sup>7–12</sup> and finally to the modern ab initio approaches relying on more expensive electronic structure methods and evolutionary algorithms,<sup>13–32</sup> the desire to facilitate synthetic discovery with theoretical guidance toward more likely compositions has been omnipresent in the search for new crystalline materials. Yet, despite the multifarious efforts, the theory-guided discovery of new crystal structures still remains a grand challenge in solid-state chemistry, except for the simplest well-defined structures with directional bonding, such as perovskites or zeolites. The situation becomes especially dire when approaching dense multielement structures with varying coordination numbers and nondirectional bonding, such as the diverse structures of intermetallic compounds.<sup>33,34</sup> The factors governing the formation of even known intermetallic structures remain an area of active research, making the reliable prediction of new ones seem quite a distant prospect.<sup>35–40</sup>

The problem of crystal structure prediction can be separated into two crucial steps: candidate crystal structures must be first generated and then evaluated to determine which of them are

likely to form. Modern structure generation approaches typically use some form of random initialization, often complemented by evolutionary algorithms.<sup>41,42</sup> The evaluation of generated crystal structures is dominated by quantum-mechanical methods.<sup>13,14,43–46</sup> Unfortunately, these approaches become prohibitively time-consuming with dense and/or complex crystal structures, especially the ones involving heavier elements or strongly correlated electrons. An alternative is offered by heuristics that simplify the evaluation to improve its speed but are usually limited to a specific type of structures and not easily generalizable. Efficient prediction schemes, which can supplement the pace of synthetic attempts at discovery of new materials, demand that the cost function is both fast and accurate.

In recent years, deep-learning methods have surpassed state-of-the-art models in a wide variety of fields.<sup>47</sup> Although deep learning has not seen widespread adoption to chemistry-relevant problems, the situation is quickly changing.<sup>48–55</sup> This is due in part to recent advances in affordable GPU hardware, along with the availability of open source frameworks such as TensorFlow,<sup>56</sup> PyTorch,<sup>57</sup> and Scikit-Learn.<sup>58</sup> The first examples of deep learning applied to crystallography have appeared very recently.<sup>49</sup> One of the important benefits of using deep-learning techniques is representation learning.

Received: April 11, 2018

Published: June 6, 2018

Engineering a method for transforming data into an appropriate representation for a model is one of the most time-consuming phases of model development. Deep learning allows the tedious representation design process to be incorporated into model fitting.<sup>59</sup>

The current, predominant model in deep learning is the deep neural network (DNN), which consists of multiple layers of nonlinear transformations. Such architecture allows DNN models to approximate functions of arbitrary complexity, in contrast to linear models such as Support Vector Machines, k-Nearest Neighbors, and logistic regression, which are unable to directly approximate nonlinear functions. The multilayered architecture of DNNs is well suited to learning hierarchical representations.

Vast information on existing crystal structures, which is available through the large open-access and commercialized repositories of crystallographic data, provides an excellent starting point for the implementation of deep-learning techniques to the discovery of hidden relationships that might be contained in such large data sets.<sup>60</sup> The machine learning algorithm can be thought of as a fitting procedure for a complicated heuristic model using a large amount of data.<sup>61,62</sup> This model is later tested to estimate its ability to generalize to unknown crystal structures in a holdout set, i.e., its predictive ability.

Herein, we describe a neural-network model trained to compare topologies of atomic sites in the known crystal structures and apply the knowledge gained from such comparison to predict possible compositions of unknown compounds that might be pursued by a synthetic chemist. We demonstrate that the DNN is capable of automatically “discovering” relevant descriptors from high-dimensional “raw representations” of the crystallographic data. Since the input data contain purely geometrical and topological information, any chemical knowledge residing within the DNN output must have been learned during training, and thus was “discovered”. The DNN’s learned representation of local topology shows evidence of known geometric and chemical trends not explicitly provided to the network during training.

## METHODS

**Software Libraries.** The following software libraries were used: Boost C++ Library, Computational Crystallography Toolbox (CCTBX), NumPy,<sup>63</sup> and TensorFlow.<sup>56</sup> In addition, we developed a multithreaded program written in C++14 to open and parse crystal structures in the Crystal Information File (CIF) format and to calculate positions of neighbor atoms (to a specific cutoff distance) around each crystallographically unique atomic site. The TensorFlow library was utilized for constructing, training, evaluating, and visualizing DNNs, as well as for the generation of atomic fingerprints. Code related to TensorFlow was written in Python 3.5. The t-distributed stochastic neighbor embeddings (t-SNE)<sup>64</sup> were calculated with Scikit-learn<sup>58</sup> and plotted with Matplotlib.<sup>65</sup>

**Crystal Structure Data Set.** The training data consisted of structures retrieved from the Inorganic Crystal Structure Database (ICSD, 2016) and Crystallographic Open Database (COD, 2016). Only structures with the final R-factor below 10% and the occupancy of each atomic site above 0.98 were selected. These limits were imposed to ensure the presence of high-quality experimental data and the lack of complications associated with crystallographic disorder. In addition, any structures with interatomic separations below 0.65 Å were discarded.<sup>66</sup> The final data set contained 704334 crystallographically unique sites in 51723 crystal structures.

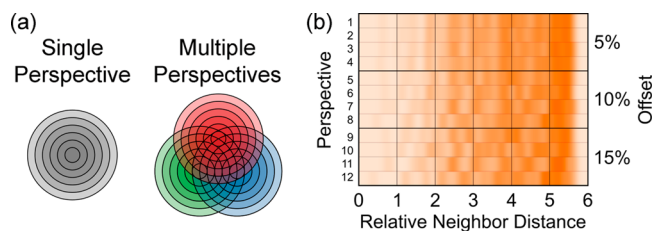
**Input Representation.** The model was trained using normalized atomic fingerprints, which represent the local topology around each

crystallographically unique atom. These atomic fingerprints (AFPs) were inspired by crystal structure fingerprints (CSFPs) described by Valle and Oganov.<sup>67</sup> The CSFP is a weighted sum of AFPs where the weights correspond to the relative abundance of each atom in the structure (the atomic site multiplicity divided by the total multiplicity of all unique atomic sites in the structure). Decoupling the contribution of each unique atomic site to the CSFP function produces the AFP function. When calculating the AFP for a specific site, all neighboring atoms were treated simply as other atoms, without any chemical identity. This corresponds to setting the element-specific *b* variable in the Valle and Oganov’s CSFP definition to 1. Thus, in the present definition, the AFP is meant to represent purely geometric relations between atoms in the crystal structure. The AFP of atom *i* at perspective *k* (see below) was calculated as

$$\text{AFP}_i^k(R) = \sum_j \delta \left( R - \frac{R_{ij}^k}{R_{i0}} \right) \quad (1)$$

where the sum runs over all atoms *j* within the threshold distance  $R_{\text{max}}$ ,  $\delta$  is a Kaiser–Bessel-smeared delta function,  $R_{ij}$  is the distance between atoms *i* and *j*, and  $R_{i0}$  is the distance to the nearest neighbor of atom *i*. The AFPs were calculated by summation over 1024 bins (from 0 to  $R_{\text{max}}$ ) and isotropically normalized by dividing all distances  $R_{ij}$  by the distance to the closest neighbor,  $R_{i0}$ , thus making the first peak in each AFP function to appear at  $R = 1$ . This normalization also removed the necessity of dividing the AFPs by shell volume during the summation procedure, as appeared in the original formula by Valle and Oganov.

The use of AFPs reduces the 3-dimensional (3D) crystal structures to 1-dimensional (1D) representations, which leads to an unavoidable loss of information. To mitigate this problem, we introduce a substantially new approach to the AFP description of crystal structures. More specifically, we use multiple perspectives, a method that has been proven effective for representing 3D objects using 2D representations.<sup>68</sup> Each perspective  $\text{AFP}_i^k$  is an individual AFP function (eq 1) calculated with the origin offset from atom *i* (Figure 1a). The total  $\text{AFP}_i$  is then a set of  $\text{AFP}_i^k$  functions (radial distribution functions), where *k* is simply an index used to enumerate the perspectives (Figure 1b). For a 1D function, such as  $\text{AFP}(R)$ , a minimum of 4 nonplanar equidistant perspectives are required to unambiguously denote locations in 3D space in an unbiased manner. We found that using 12 perspectives worked well in practice, in terms of providing good-quality results at reasonable computational costs.



**Figure 1.** (a) 2D example of a single and a multiperspective AFP. Each concentric circle represents a shell whose thickness is equal to the bin width (in our case,  $1/1024$  of  $R_{\text{max}}$ ). The displacements of the origin used to calculate the multiperspective AFPs are exaggerated. The variation in the color intensity serves just as a visual aid. (b) Visualization of a multiperspective AFP. Each of the 12 horizontal bands corresponds to the radial distribution of neighbors calculated from a different perspective. The first four perspectives are equidistant from one another and are offset from the origin by 5% of  $R_{i0}$  to form the shape of a tetrahedron. The next two sets of four perspectives correspond to offsets of 10% and 15% of  $R_{i0}$ , respectively. Shading indicates the relative number of neighbors (darker means more neighbors). The peak shifts, which become more apparent at larger offsets, allow the model to interpret 3D topologies using the combination of 12 radial distribution functions ( $\text{AFP}_i^k$ ).

These perspectives were created by choosing the origin as 3 sets of 4 points arranged around the atomic site  $i$  in the shape of the vertices of nested tetrahedra at increasing offset distances (0.05, 0.10, 0.15 of the nearest-neighbor distance,  $R_{i0}$ ). Larger offsets were more challenging for the model to learn while smaller offsets did not provide sufficient “depth perception”. To avoid potential bias toward specific orientations, a random 3D rotation was applied to the entire structure around the atomic site  $i$ , after which the offset was performed and the 12 AFP $_i^k$  functions were calculated. Prior to calculating each AFP $_i$ , the locations of neighbor atoms in the coordination sphere of interest (up to  $R_{\max}$ ) were also randomly shifted using normally distributed perturbations with a standard deviation of 0.01 Å. These randomization procedures were intended to minimize the bias in the input representations used by the model during training. Moreover, they make the model more robust to numerical errors introduced by binning the AFP functions during the summation and increase the diversity of the training data.

The multiperspective AFP representation has several notable advantages: translational invariance, fixed dimensionality, and the retention of 3D information with minor increase in computational costs. Translational invariance avoids the need to choose a comparable origin across all crystal structures. Fixed dimensionality facilitates the learning process by avoiding the need for models that can handle variable-length inputs, which are notoriously difficult to train.<sup>69</sup> Voxels are the typical method of representing 3D objects but they have prohibitively large memory requirements for high-resolution data. The multiperspective AFP involves a trade-off between the resolution of faraway objects and memory requirements, all while retaining 3D information. The drop-off in resolution as a function of distance from the central atom is tolerable in this application because of the relatively high importance of local information and high variability of structures at extended ranges.

In our present implementation, each AFP $_i^k$  was calculated using 1024 bins with a distance cutoff  $R_{\max} = 6R_{i0}$ . The Kaiser-Bessel smearing window had a width of 128 bins and the shape parameter  $\alpha = 14$ . After blurring, the AFPs were reduced to 256 dimensions by applying a fast Fourier transform (FFT), discarding higher-frequency terms, and then applying an inverse FFT to obtain the downsampled AFPs. This binning, blurring, and dimensionality reduction procedure is similar to nonuniform FFT, for which the Kaiser-Bessel smearing has been shown to result in less information loss relative to a Gaussian smearing.<sup>70</sup> We also note that the choice of blurring function (Kaiser–Bessel vs Gaussian) yielded no noticeable impact on the model performance.<sup>71</sup>

To provide spatial information lost due to the normalization of AFPs, two additional geometric descriptors were added: (1) the non-normalized distance ( $R_{i0}$ ) from atom  $i$  to its nearest neighbor and (2) the ratio of the smallest interatomic distance in the crystal structure to  $R_{i0}$ . These descriptors reflect, respectively, the actual (in Å) and relative size of site  $i$  in the crystal structure. During the model operation, as described below, they are used by the softmax and sigmoid classifiers, respectively. The use of  $R_{i0}$  can be thought of as the way to inform the model about the scale of topology.

**Class Imbalance and Oversampling.** The cleaned data set contained 704334 crystallographically unique atomic sites in 51723 crystal structures. A histogram of chemical elements present in these atomic sites is shown in Figure S1. As can be seen from the histogram, some elements are observed more frequently than the others, a situation known as class imbalance. This issue is addressed by oversampling instances of the minority classes.<sup>72,73</sup> It is in the classifier’s best interest to bias its predictions toward elements it sees most often. The oversampling procedure mitigates this bias by increasing the sampling frequency for the less frequently observed elements.

**Model Architecture.** A variational autoencoder (VAE) was used to allow the DNN to learn its own representation of the AFPs. The VAE was trained by minimizing reconstruction error and Kullback–Leibler (KL) divergence, which measured the deviation of the shape of the latent representation from an isotropic multidimensional unit Gaussian distribution.<sup>74</sup> Specifically, we trained a 42-layer convolu-

tional VAE on 12-perspective, 256-dimensional AFPs. The 64-dimensional latent representations generated by the VAE, along with the normalized geometric descriptor, were fed as input into a five-layer sigmoid classifier with 118 output neurons (one for each chemical element in the periodic table). The output from the sigmoid classifier, along with the non-normalized geometric descriptor, was then fed as input into a five-layer softmax classifier with 118 output neurons, corresponding to 118 known chemical elements. (Note that, while our data set only contains examples of 89 elements, the use of 118 output neurons makes the network architecture robust to the addition of the remaining elements as examples become available. The presence of “unused” neurons has only a minor impact on computational performance.) A scheme of the DNN architecture is shown in Figure 3 in the Results and Discussion. All weights were initialized using Xavier initialization.<sup>75</sup>

For the sigmoid classifier, only 1 out of 118 possible labels is correct for each example, which introduces a class imbalance. To counteract this problem, we apply a weight of 118 to the true classes. Note that since the unobserved classes (29 out of 118) are never correct, the sigmoid classifier learns that these elements are not present in our data set, and therefore they have zero likelihood to occur on any crystallographic site.

**Training Notes.** All AFPs have been preprocessed; the mean and standard deviation of each dimension of the AFP and geometric descriptors was estimated using  $\sim 10000$  randomly chosen examples. We allowed the network to learn an offset ( $\beta$ ) and a scale factor ( $\gamma$ ) parameters for each input dimension, which were initialized to the estimated mean and standard deviation, respectively. The AFP input was transformed to zero mean and unit variance by subtracting  $\beta$  and then dividing by  $\gamma$ . Conversely, output from the VAE was transformed by multiplying by  $\gamma$  and then adding  $\beta$ . The DNN was allowed to adjust these two parameters during training, if such adjustments improved the model’s performance. Batch normalization, which is common in current DNN implementations,<sup>76</sup> was applied only to the input of the softmax classifier, because sigmoid classifier’s logits and geometric descriptors would likely deviate from zero mean and unit variance over the course of training. Batch normalization was not employed elsewhere since the activation function utilized throughout the DNN was the exponential linear unit, which has been reported to not benefit significantly from batch normalization.<sup>48</sup>

The DNN was trained using the Adam optimizer with the following hyperparameters:  $\epsilon = 1 \times 10^{-8}$ ,  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$ . The learning rate was initially set to  $3 \times 10^{-7}$ , then increased by increments of  $3 \times 10^{-10}$  every batch to a maximum of  $3 \times 10^{-5}$ , then decreased to a value of  $2.25 \times 10^{-5}$  after 5000 batches (from the start of training). This learning rate schedule was designed to combat occasional instability encountered at higher learning rates during the start and end of training. Dropout probability was set to 5% along with a regularization weight  $L_2 = 5 \times 10^{-4}$ . A batch size of 16 was chosen for training. These hyperparameters were arrived at through trial-and-error experimentation.

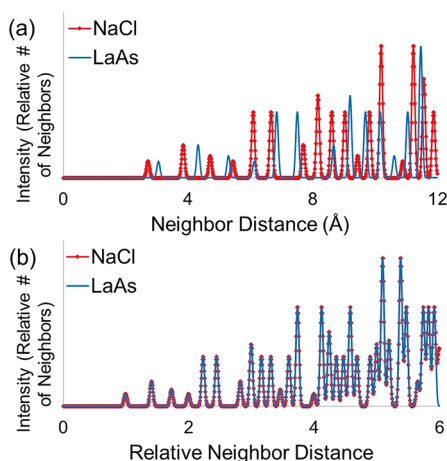
The VAE’s KL-divergence weight was increased from the initial value of 0.01 to the final value of 1.0 in increments of 0.001 per batch. The gradual increase of the weight to the intended value over the course of training reduces the risk of settling into local minima. At the start of training, the DNN’s weights are relatively far from “good” values. Starting with a small value avoids potential issues with the optimizer making changes at the beginning of training that only reap benefits from reducing KL-divergence while becoming trapped in a local minimum.

The network was trained for approximately 66 epochs,<sup>77</sup> at which point the slope of the sigmoid cross entropy loss function reached nearly zero (Figure S2), indicating that the sigmoid classifier was no longer improving. The meaning of epoch (single pass over all training examples) is slightly different when performing oversampling. Each example in classes with fewer examples is seen more often than those in classes with more examples. An epoch is still the same number of examples as appears in the training set, but because of oversampling this does not mean that every example is seen once per epoch; some examples are not seen at all, while some are seen multiple times.



## RESULTS AND DISCUSSION

**Atomic Fingerprints as Input Data.** Deep learning methods use representation learning. They convert the input data to automatically discover new representations for examining hidden correlations in the data set. Therefore, training of the model is critically dependent on the representation used to inform the model about the input data. The use of normalized atomic fingerprints (AFP) allows training the DNN on the structural topologies rather than on crystal structures with specific scale. In this sense, for example, the model should be able to consistently compare two rock-salt type structures, even if they exhibit very different unit cell parameters due to drastically different atomic sizes (Figure 2).



**Figure 2.** Original (a) and normalized (b) AFPs for crystal structures of NaCl and LaAs.

Thus, switching from regular AFPs to normalized AFPs reformulates the problem addressed by the structure evaluator from “does this *structure* make sense for this set of elements?” to “does this *structural topology* make sense for this set of elements?” The benefit of this approach is a significant reduction in the number of candidate structures which must be screened.

The computational cost of calculating the AFP function increases exponentially with the cutoff distance. Due to

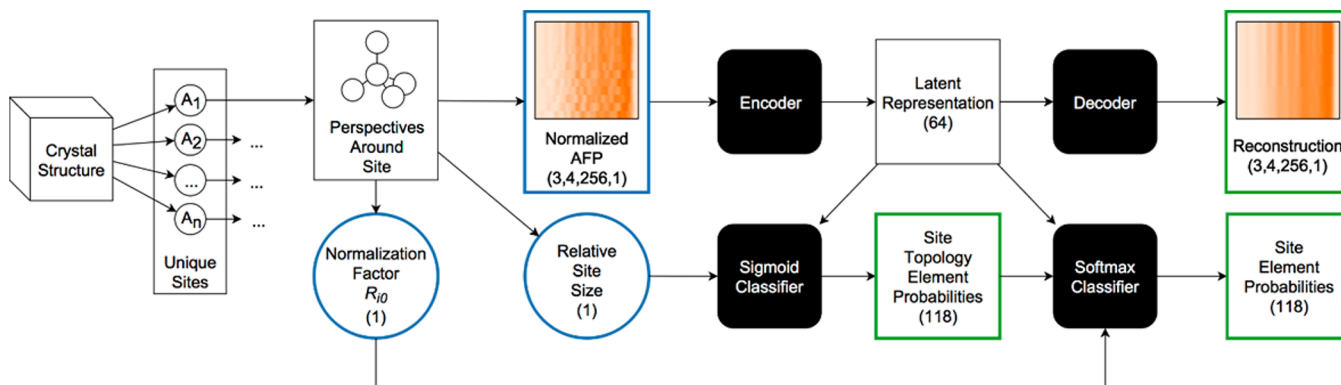
limitations in available computational power, we set the cutoff value to 6 times the nearest-neighbor distance for each AFP. For the normalized AFP, the nearest-neighbor distance  $R_0 = 1$ , and thus, the cutoff value becomes  $R_{\max} = 6$  after normalization.

AFPs were partitioned into 3 sets: training (60% of data), holdout (20% of data), and test (20% of data). The training set was used to train the DNN, the holdout set was used to evaluate the network’s performance on unseen data, and the test set was kept hidden from both researchers and the network to allow for a better measure of generalization performance at the end of model development.

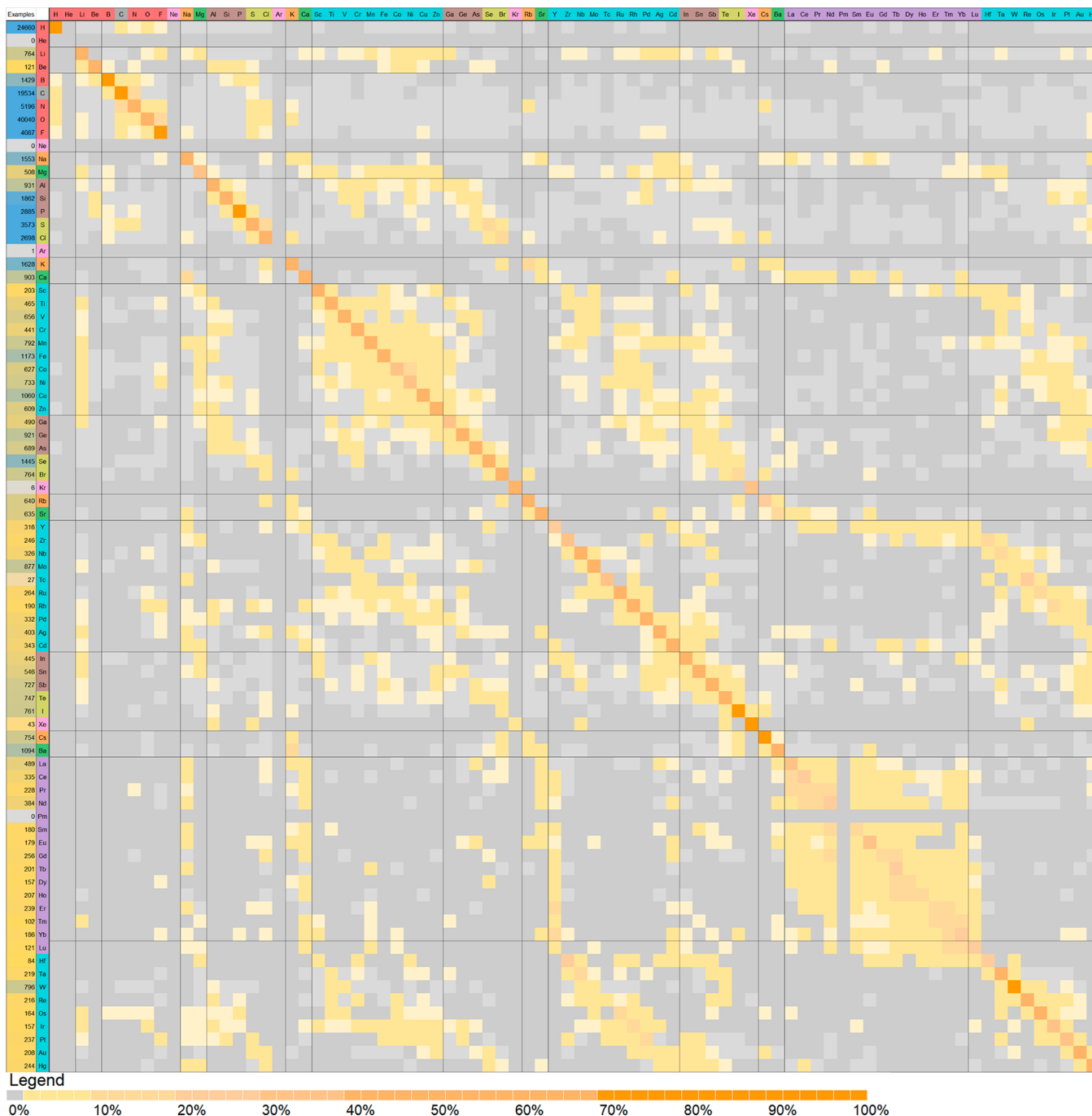
**Training of DNN Model.** The DNN architecture included three major components (Figure 3). The *variational autoencoder* was used to transform 3072-dimensional (12 256-dimensional perspectives) AFP input data into 64-dimensional latent representations to be used for deep learning. These latent representations were then used by the *sigmoid classifier* to predict what combinations of elements were likely to form specific structural topologies. The likelihoods obtained from this analysis were refined by an *auxiliary softmax classifier*, which used information about actual distances to predict what specific element corresponded to the input.

A powerful feature of DNN models is their ability to substantially improve the performance of classifiers that assign objects to specific categories, especially for high-dimensional sets of data. Classifiers operate by creating decision boundaries to partition different classes of data. In the case of complex high-dimensional data sets, such boundaries can be very complicated, which impedes data classification. By transforming the original data input into latent representations, DNNs discover a function to disentangle the data and afford better definition of decision boundaries.

To better visualize the improvement in the data representation by the DNN, we use t-distributed stochastic neighbor embedding (t-SNE), which is an iterative process for mapping high-dimensional data onto a lower-dimensional projection for the purposes of visualization.<sup>64</sup> This adaptive procedure applies different transformations to different regions of the high-dimensional space to create a nonlinear transformation that reflects both local and global aspects of the underlying data. The balance between the relative importance



**Figure 3.** Diagram of the deep neural network (DNN) architecture used to predict elements from crystallographic sites. The inputs and outputs to the DNN are outlined with blue and green, respectively. The numbers correspond to the dimensionality of each layer. Hidden layers use exponential linear unit activation functions. The latent and reconstruction output layers use linear activation functions. The Sigmoid Classifier’s element probabilities output layer uses a sigmoid activation function. The Softmax Classifier’s refined element probabilities output layer uses a softmax activation function. Dropout was applied between each layer (aside from output layers).



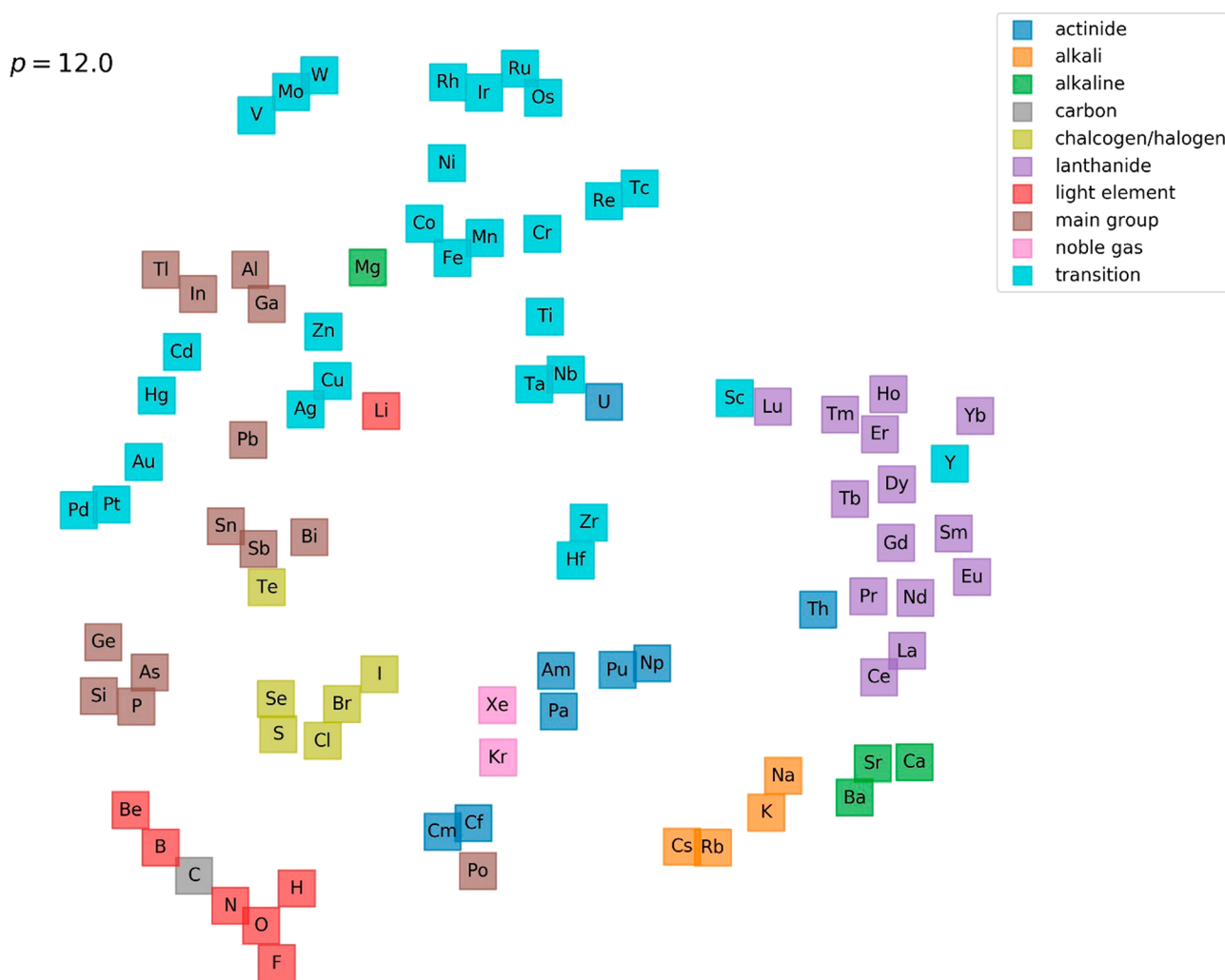
**Figure 4.** Confusion matrix of the softmax classifier obtained on the test set; rows correspond to ground truth labels (chemical elements on crystallographic sites in the input crystal structures), while columns correspond to predicted labels. The left-most column indicates the number of examples of each element in the test set. Cells are colored to indicate cell values according to the legend at the bottom. The color scheme was chosen to exaggerate mistakes; values in off-diagonal cells, which represent erroneous classifications, rarely exceed 10%. The average error rate for the test set was 31%, while the average error rates for the validation and training sets were 31% and 29%, respectively.

of local and global aspects is controlled using a parameter called perplexity ( $p$ ), which is loosely related to the notion of the number of close neighbors any given point is expected to have.<sup>78</sup> The t-SNE visualization of the grouping of normalized AFPs prior to and after transformation by the DNN clearly shows that the data classification using learned (latent) representations is more effective (Figure S3; also see Figure 5).

Note that in our structure evaluation (see the next section) we use the output from the *sigmoid* classifier, not from the *softmax* classifier. The purpose of the softmax classifier is

simply to help train the sigmoid classifier; the former is discarded after training. The intent of such approach is to utilize additional information about actual interatomic distances (which are absent in the normalized AFP), in order to steer the sigmoid classifier in a better direction. Auxiliary classifiers have been used to speed up training in record-setting DNN architectures.<sup>79</sup>

The softmax classifier allows us to measure an error rate for prediction (with mutually exclusive labels of chemical elements). The sigmoid classifier only evaluates the bonding



**Figure 5.** 2D t-SNE visualization of sigmoid classifier weights at the perplexity value  $p = 12$ . Each point can be thought of as the DNN’s “intuition” about the type of site topologies that a chemical element can adopt. Closeness of points indicates similar structural behavior of elements. (See Figure S6 for additional t-SNE visualizations.)

topology from isotropically normalized AFPs, which can be characteristic of a number of similar elements. Therefore, assigning a specific element label as being the only correct answer is not possible in such case. Our inability to decide which element labels assigned by the sigmoid classifier are false is addressed, in part, by weighting the sigmoid classifier toward minimizing false negatives. In order to do so, we use the auxiliary softmax classifier, which uses the data output by the sigmoid classifier to refine the prediction to a single element. This procedure assumes that, given information about the scale of the site topology, provided by the additional geometric descriptor  $R_{10}$ , the chemical elements should become, at least in part, mutually exclusive. This allows us to distinguish a correct label of an element for a specific site and assign the other 117 element labels as wrong, thus providing a measure of the error rate. We used this measure to build an element confusion matrix as described below.

The confusion matrix in Figure 4 shows the softmax classifier’s performance at predicting elements on sites from the validation set (the color-coded full confusion matrix is shown in Figure S4 and is also provided as numerical values in the form of an Excel spreadsheet in the Supporting Information). Ideal performance would result in values of

100% along the matrix diagonal and values of 0% everywhere else. While we see that the correlation values along the diagonal, in general, are higher, a number of off-diagonal confusions are observed. It is interesting, however, that the “mistakes” made by the auxiliary softmax classifier are predominantly chemically reasonable. For example, the confusion is especially well pronounced in the blocks of  $3d$  and  $4f$  elements, which underscores the similarity of typical coordination environments and atomic radii in the corresponding groups of elements. We also note such well-known cases of chemical similarity, in the structural sense, as Li or Mg vs  $3d$  metals and Ca or Y vs  $4f$  metals. These observations provide evidence that the classifier has gleaned some level of chemical “understanding” from the purely geometric description of local structural topology.

The sigmoid classifier’s performance can be also visualized by looking at the distribution of predicted likelihoods for correct and incorrect elements (Figure S5). For both training and test sets, the sigmoid classifier reliably assigns high site likelihood values for the elements known to exist in specific site topologies and nearly zero likelihoods for the other (incorrect) elements.

To visualize more clearly the chemical “knowledge” gained by the DNN model, we show in Figure 5 a t-SNE embedding of the sigmoid classifier weights for 88 elements encountered in our training data set. Since the t-SNE embedding is a statistical method used to present high-dimensional data on a 2D or 3D plot,<sup>64</sup> the appearance of the element distribution might not seem very familiar at first glance. Nevertheless, a closer look immediately reveals groupings similar to those seen in the periodic table. We can clearly distinguish the lanthanides. Remarkably, Y is positioned close to the smaller 4f elements, which is reasonable given the similarity of their ionic radii and coordination topologies. We also see the clustering of alkali and alkali-earth metals, 3d metals and Mg, and 4d and 5d noble metals. The groupings of second-row elements, chalcogens and halogens (S, Se, Cl, Br, I), tetrels and pnictogens (Si, Ge, P, As), and metalloids (Sn, Sb, Bi, Te) also make sense. They point not only to clustering that corresponds to groups or rows in the periodic table, but also to the diagonal similarity between elements, which is empirically well-known to chemists. It is obvious that the DNN model, which was given purely geometric information from crystallographic databases, has been able to discover the periodic trends and chemical similarities.

**Crystal Structure Evaluation.** While the DNN model determines the most likely chemical element for a given atomic site, it is not capable of evaluating the likelihood of the entire crystal structure. This limitation stems from the lack of examples of crystal structures which cannot exist. For machine-learning based methods it is necessary to have input examples of both positive and negative outcomes so that the model can learn what is possible and impossible. Unfortunately, there are no databases of crystal structures which are known *not* to exist. (This is in contrast to experimental synthetic chemistry, where unsuccessful reaction attempts can be documented.<sup>80</sup>) Without this crucial bit of information, there is not an obvious way for a model to be trained to recognize reasonable from unreasonable.

In order to mitigate this issue, we reformulated the crystal structure prediction problem into one of predicting the likelihoods of individual atomic sites in the structure. This approach relies on the approximation that the likelihood of a crystal structure to exist is equal to the product of likelihoods for each element to reside on a specific atomic site. The latter was determined using the DNN model described above. The array of values returned by the sigmoid classifier provided the likelihood of every one of the 118 chemical elements to exist on a given site. Of course, there is a possibility that a given crystal structure is not possible despite the high likelihood for the corresponding atomic sites to contain a specific set of elements.<sup>81</sup> Nevertheless, we believe that, provided a well-trained DNN model, such possibility is rare and has more to do with the experimental challenge of synthesizing the specific material or with insufficient thermodynamic stability of a compound with respect to the other structures that can form in the given system.<sup>82</sup>

The evaluation of the structure likelihood as the product of the site likelihoods also leads to two other consequences. First, the elements which do not correspond to certain types of sites act as counterexamples. In other words, the model learns what coordination topologies do and do not make sense for each chemical element. Second, a zero probability for an element to occupy a site in the structure also causes the likelihood to

obtain such structure to be zero, which quickly eliminates highly unlikely structures.

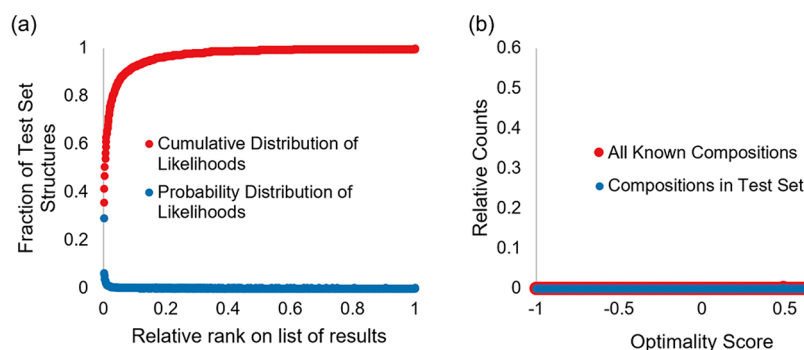
Once the neural-network model had been trained, the predicted likelihoods of all chemical elements on all crystallographic sites were calculated and stored to avoid costly recalculation during crystal structure prediction (described below). In this way, finding the probability of an element to adopt the topology of a specific site in a specific structure was rendered to a lookup procedure.

**Crystal Structure Prediction.** As mentioned in the Introduction, the crystal-structure prediction consists of two major steps: generation and evaluation. Having developed the DNN model for fast evaluation of a large number of possible crystal structure candidates, we now face the nontrivial problem of crystal structure generation. Modern structure prediction approaches typically tackle this problem by using random initialization complemented by evolutionary algorithms.<sup>41,42</sup> We would like to point out, however, that the great majority (~95%) of new crystal structures deposited in the ICSD nowadays represent already known structure types. Therefore, for the sake of simplicity in developing our structure prediction approach, we use the known structure types as the starting point for generating new crystal structures. The downside to such strategy is the inability of the model to discover novel structural arrangements (structure types). Nevertheless, since the generation and evaluation procedures are entirely separated in our approach, a superior crystal structure generator can be added to the model in the future without affecting the structure evaluation component performed by the DNN.

The unique crystallographic sites in the 51723 known crystal structures have been used as seeds, devoid of any chemical labels. Using these structural templates, the generator produced new crystal structures by combinatorial substitution across the series of all chemical elements. For a particular crystal structure template, such substitution was performed until all possible compositions and unique site configurations had been exhausted. For the initial implementation, we limited our predictions to binary and ternary compositions, which led to 623380 binary and 2703834 ternary candidate crystal structures. (It should be noted that some of these templates will be isostructural in the crystallographic sense. But since the DNN model was not supplied the conventional crystallographic information, such duplicate templates are not removed during the structure generation procedure. In fact, having more data improves the model's performance during training while the runtime penalty during structure generation is negligible.) One can extend this approach to compositions with an arbitrary number of chemical elements, with the provision that this number should be equal to or greater than the number of unique crystallographic sites in structural templates utilized by this procedure.

Using this structure generation approach, coupled to the structure evaluation protocol described above, we now can perform crystal structure prediction. The performance of the structure prediction model was tested on 5845 crystal structures, without restriction on the number of unique elements in a crystal structure. These structures were contained exclusively within the test set and did not have any common compositions in the training set. In other words, the DNN model had never seen these data during training. Using these holdout data, two separate tests were designed to estimate the model's performance.





**Figure 6.** (a) Probability (blue) and cumulative (red) distribution rankings over structure prediction results for the test set. The likelihood for a structure generated from a correct template to be ranked as the top candidate is given by the first point in the probability curve. A perfect evaluator would produce a Kronecker delta function centered at 0 for the probability curve. (b) Distribution of optimality scores, defined in eq 2, from all binary (290) and ternary (1719) element combinations in the test set. A perfect optimality score would be a Kronecker delta function centered at 1.

**Table 1. Most Likely Unique Chemical Compositions Predicted for the Mn–Ge and Li–Mn–Ge Systems**

| Mn–Ge system                          |   |  | Li–Mn–Ge system                                    |  |  |
|---------------------------------------|---|--|--|--|--|
| composition <sup>a</sup>              | template structure (ICSD Code)                            | product of site likelihoods <sup>b</sup> | composition <sup>a</sup>                           | template structure (ICSD Code)                           | product of site likelihoods <sup>b</sup> |
| 1. MnGe <sub>2</sub>                  | NiS <sub>2</sub> (68169)                                  | 0.890                                    | 1. LiMn <sub>2</sub> Ge <sub>2</sub>               | Li <sub>2</sub> CuO <sub>2</sub> (108666) <sup>c</sup>   | 0.819                                    |
| 2. MnGe <sub>3</sub>                  | CoSb <sub>3</sub> (62111)                                 | 0.886                                    | 2. LiMn <sub>2</sub> Ge <sub>4</sub>               | MgCr <sub>2</sub> O <sub>4</sub> (75623)                 | 0.813                                    |
| 3. MnGe <sub>4</sub>                  | CrP <sub>4</sub> (2790)                                   | 0.859                                    | 3. Li <sub>6</sub> MnGe <sub>6</sub>               | Li <sub>6</sub> UO <sub>6</sub> (48209)                  | 0.772                                    |
| 4. MnGe                               | NiS (151599)  | 0.831                                    | 4. <i>Li<sub>2</sub>MnGe</i>                       | PrCuSO (96345)   | 0.759                                    |
| 5. Mn <sub>2</sub> Ge <sub>3</sub>    | Fe <sub>2</sub> O <sub>3</sub> (164008)                   | 0.785                                    | 5. Li <sub>3</sub> Mn <sub>2</sub> Ge <sub>3</sub> | Ge <sub>3</sub> Rh <sub>2</sub> Se <sub>3</sub> (261240) | 0.755                                    |
| 6. Mn <sub>3</sub> Ge <sub>4</sub>    | Mn <sub>2</sub> ZnO <sub>4</sub> (166522)                 | 0.780                                    | 6. LiMnGe <sub>2</sub>                             | ScAgSe <sub>2</sub> (155115)                             | 0.754                                    |
| 7. Mn <sub>7</sub> Ge <sub>6</sub>    | Li <sub>6</sub> UO <sub>6</sub> (48209)                   | 0.762                                    | 7. LiMnGe <sub>3</sub>                             | MnCoO <sub>3</sub> (31854)                               | 0.753                                    |
| 8. Mn <sub>3</sub> Ge <sub>5</sub>    | Sm <sub>3</sub> Ge <sub>5</sub> (416581)                  | 0.739                                    | 8. LiMn <sub>6</sub> Ge <sub>6</sub>               | Li <sub>6</sub> UO <sub>6</sub> (48209)                  | 0.746                                    |
| 9. MnGe <sub>7</sub>                  | Li <sub>6</sub> TeO <sub>6</sub> (40247)                  | 0.729                                    | 9. LiMnGe  | CoAsS (69129)  | 0.738                                    |
| 10. Mn <sub>6</sub> Ge                | B <sub>6</sub> As (68151)                                 | 0.727                                    | 10. Li <sub>3</sub> MnGe <sub>2</sub>              | CoCu <sub>2</sub> O <sub>3</sub> (33996)                 | 0.732                                    |
| 12. Mn <sub>2</sub> Ge                | Cs <sub>2</sub> Se (41687)                                | 0.701                                    |  |  |  |
| 18. Mn <sub>3</sub> Ge                | LiNbO <sub>2</sub> (75880)                                | 0.608                                    |  |  |  |
| 26. Mn <sub>5</sub> Ge <sub>3</sub>   | Fe <sub>5</sub> Si <sub>3</sub> (99973)                   | 0.521                                    |  |  |  |
| 27. Mn <sub>5</sub> Ge <sub>2</sub>   | V <sub>4</sub> SiSb <sub>2</sub> (82564)                  | 0.520                                    |  |  |  |
| 239. Mn <sub>11</sub> Ge <sub>8</sub> | K <sub>8</sub> Tl <sub>11</sub> (370009)                  | 0.00852                                  |  |  |  |
| 332. Mn <sub>23</sub> Ge <sub>9</sub> | Li <sub>6</sub> Ca <sub>17</sub> Hg <sub>9</sub> (420846) | 0.00104                                  |  |  |  |

<sup>a</sup>Experimentally known compositions are italicized. <sup>b</sup>The likelihood values do not imply the exact probability but rather reflect the network's confidence that a specific composition is possible in a given structure template. <sup>c</sup>The structure template from the COD database.

**Test 1.** This test was designed to assess the predictive ability of the evaluation component, which relies on the pretrained DNN and the assumption that the likelihood of a crystal structure ( $P_{\text{crystal}}$ ) can be estimated by the product of atomic site likelihoods ( $P_i$ ) which are output by the sigmoid classifier of the DNN model:  $P_{\text{crystal}} = \prod_i P_i$ , where the product is taken over all unique atomic sites  $i$  in the crystal structure. Since all chemical information has been removed from the structure-generating templates, a known crystal structure, from which a template has been produced, should be also found in the set of generated structures. If the evaluation component works properly, then this known crystal structure should be ranked highly in the list of candidates produced with the same chemical composition. We find that there is 27% probability for this known crystal structure to appear as the top ranked candidate and 59% probability for it to appear in the top-10 predicted structures. Note that these prediction results frequently contain multiple candidates generated from isostructural crystal structure; they are indistinguishable from the known structure and compete for the top spot. Therefore, these statistics are likely to underestimate the model's true performance. This test shows that our evaluation model is

significantly better than a random chance at identifying the correct crystal structure for a given composition (Figure 6a).

**Test 2.** This test estimates the predictive ability of the algorithm by evaluating all binary and ternary element combinations contained in the test set (2009 in total) against all structures generated from the available templates for these element combinations. The test is designed to assess how highly a particular composition is ranked among all possible compositions produced by a given combination of elements using the available structural templates. Performing this test on the 2009 element combinations took approximately 6 h on a modern 8-core desktop computer and involved the generation and evaluation of 4.8 billion structures (120 million unique compositions). Note that only the predicted compositions (not the structures) affect the score of this test.

The outcome of test 2 was measured by an optimality score, which reflects the distribution of  $Q$  known compositions for a particular combination of elements over the entire set of  $N$  unique compositions generated with available structural templates:



$$\begin{aligned}
 \text{OS} &= 2 \times \left[ 1 - \frac{\sum_{i=1}^Q q_i - \frac{1}{2}(Q^2 + Q)}{\frac{1}{2}(2N - Q + 1)Q - \frac{1}{2}(Q^2 + Q)} \right] - 1 \\
 &= \frac{NQ + Q - 2 \sum_{i=1}^Q q_i}{NQ - Q^2} \quad (2)
 \end{aligned}$$

where  $q_i$  indicates the position of the known composition  $i$  ( $i = 1, \dots, Q$ ) in the list of generated compositions ( $q_i$  can take values from 1 to  $N$ ). An optimality of 1 would indicate that every known composition is at the top of the list of predicted compositions, while an optimality of  $-1$  would place every known composition at the bottom of the list. A list where the position of any given composition is random would have an average optimality score of 0. The distribution of the optimality scores (Figure 6b) is dominated by points with optimality scores approaching 1. This shows that our prediction model is significantly better than a random chance at identifying likely compositions to be found in a phase diagram. The prediction runtimes for binary and ternary compositions for a specific set of elements were about 2 and 10 s, respectively.

**Case Study.** A particular goal of this project is to apply the structure prediction algorithm as guidance for exploratory synthesis. If we would like to discover a new compound in a phase diagram (explored or unexplored), we could use the prediction algorithm to assess the likelihood of crystal structures within a specified range of compositions. This prediction should be followed by experimental attempts at synthesizing materials with the predicted chemical compositions. Compared to a blind search (commonly known as a shake-and-bake approach), such guided synthesis focuses on more promising areas of the compositional space and should yield a better success rate at discovering new materials, as suggested by the validation tests 1 and 2 described in the previous section.

Let us consider a Mn–Ge binary system. According to our data set, the Mn–Ge binary system contains seven known compositions, of which three appear in the top-20 unique compositions predicted by our algorithm, while the other four are ranked as 26, 27, 239, and 332 in the list of all 3051 unique compositions produced by the generator (Table 1). Even though the known Mn–Ge binary structures are contained within the templates used for structure generation, the specific templates from which the likely  $\text{Mn}_x\text{Ge}_y$  compositions have been derived can be different. For example, the known composition MnGe was predicted to have high probability to exist in the FeAs structural template. It is also reasonable that the compositions derived from more complex templates, viz.,  $\text{Mn}_{11}\text{Ge}_8$  and  $\text{Mn}_{23}\text{Ge}_9$ , appear lower ranked, both due to the higher variability of the structure and the product of site likelihoods involving more unique atomic sites.

Adding another element to this system, e.g., Li, results in 66652 unique  $\text{Li}_x\text{Mn}_y\text{Ge}_z$  compositions produced by the generator. The top 10 of these compositions are also listed in Table 1. Importantly, our entire data set did not contain any known compositions with this combination of elements. We did find, however, that  $\text{Li}_2\text{MnGe}$  was reported as a known compound in the Pearson's Crystal Data compilation, and this composition was suggested as the fourth choice by our algorithm. While this is just a solitary case, we reiterate that our structure prediction algorithm consistently identified known compositions, previously unseen by the DNN model (the test

set), as likely to form according to the optimality score statistics shown in Figure 6b.

**Concluding Remarks.** The DNN model developed in this work demonstrates that deep learning techniques can be effectively applied to extract chemically meaningful knowledge from large repositories of crystallographic data. Despite not having any a priori knowledge, our model was able to discover a method for distinguishing chemical elements and revealing similarities among them based on solely geometrical and topological considerations. This finding also attests to the significant amount of chemical information “encoded” in the crystal structures. Provided a sufficiently large set of input data, this information becomes accessible to deep learning techniques.

While the choice of our study examples was arbitrary, the statistical performance of the model confirms its ability to quickly identify chemical compositions and structures that are likely to form for a chosen combination of elements (Figure 6). It is important to emphasize that the crystal structure prediction is based on the model's ability to identify similarity between crystallographic sites and likelihood for a particular element to adopt certain coordination topologies. Hence, even if the exact structure type of the template is different from that of the known composition, the model's predictions can be considered “chemically” meaningful. Thus, we reiterate that the model serves as a guide for synthetic efforts, narrowing down the number of potentially promising compositions to be pursued. This might be trivial in the case of binary systems, where one can quickly scan the composition space in 10% increments, e.g.,  $\text{Mn}_{0.9}\text{Ge}_{0.1}$ ,  $\text{Mn}_{0.8}\text{Ge}_{0.2}$ , and so on, which leads to nine trial compositions. The situation becomes more complex in ternary systems, where a similar systematic scan of the composition space will lead to 36 trial compositions. Not surprisingly, while the known binary compositions cover 67% of all possible combinations of two elements (of the set of 89 elements), the known ternary compositions cover only ~17% of possible combinations. A systematic scan of four- or five-element systems becomes nearly intractable, and for such situations any reasonable guidance of synthetic efforts becomes highly valuable. Therefore, we view our approach as being especially useful for exploring ternary phase diagrams without any presently known compounds or any phase diagrams with four or more elements.

## ■ ASSOCIATED CONTENT

### § Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/jacs.8b03913.

histogram of the element abundance on crystallographic sites, the entropy loss and objective function vs the number of training iterations, the full confusion matrix of auxiliary classifier, a histogram of predicted likelihoods for site topologies, and additional t-SNE visualizations (PDF)

Figure S4 (XLSX)

## ■ AUTHOR INFORMATION

### Corresponding Authors

\*kryan@chem.fsu.edu

\*shatruck@chem.fsu.edu

ORCID 

Michael Shatruck: 0000-0002-2883-4694

## Notes

The authors declare no competing financial interest.

## ■ ACKNOWLEDGMENTS

This research was supported by the National Science Foundation (Award No. DMR-1507233) and by the U.S. Department of Energy via Oak Ridge National Laboratory (Subcontract No. 4000122380). We gratefully acknowledge Dr. Marat Mustyakimov for helpful ideas and discussions that led to the initiation of this project as well as for a number of computational routines that were used in this work.

## ■ REFERENCES

- (1) *Modern Methods of Crystal Structure Prediction*; Oganov, A. R., Ed.; Wiley-VCH: Weinheim, 2011.
- (2) Pauling, L. C. *The Nature of the Chemical Bond and the Structure of Molecules and Crystals. An Introduction to Modern Structural Chemistry*. 3rd ed.; Cornell University Press: Ithaca, 1960.
- (3) Goldschmidt, V. M. *Trans. Faraday Soc.* **1929**, *25*, 253–282.
- (4) Wasastjerna, J. A. Z. *Phys. Chem.* **1922**, *101*, 193–217.
- (5) Pauling, L. *Proc. R. Soc. London, Ser. A* **1927**, *114*, 181–211.
- (6) Burdett, J. K.; Price, S. L.; Price, G. D. *Solid State Commun.* **1981**, *40*, 923–926.
- (7) Mooser, E.; Pearson, W. B. *Acta Crystallogr.* **1959**, *12*, 1015–1022.
- (8) Zunger, A., A pseudopotential viewpoint of the electronic and structural properties of crystals. In *Structure and Bonding in Crystals*, O'Keeffe, M.; Navrotsky, A., Eds.; Academic Press: New York, 1981; Vol. 1, pp 73–135.
- (9) Bloch, A. N.; Schattelman, G. C., Quantum-defect orbital radii and the structural chemistry of simple solids. In *Structure and Bonding in Crystals*, O'Keeffe, M.; Navrotsky, A., Eds.; Academic Press: New York, 1981; Vol. 1, pp 49–72.
- (10) Pettifor, D. G. *Solid State Phys.* **1987**, *40*, 43–92.
- (11) Pettifor, D. G. *Mater. Sci. Technol.* **1988**, *4*, 675–691.
- (12) Pettifor, D. G.; Aoki, M. *Philos. Trans. R. Soc., A* **1991**, *334*, 439–449.
- (13) Curtarolo, S.; Morgan, D.; Persson, K.; Rodgers, J.; Ceder, G. *Phys. Rev. Lett.* **2003**, *91*, 135503.
- (14) Fischer, C. C.; Tibbetts, K. J.; Morgan, D.; Ceder, G. *Nat. Mater.* **2006**, *5*, 641–646.
- (15) Hautier, G.; Ong, S. P.; Jain, A.; Moore, C. J.; Ceder, G. *Phys. Rev. B: Condens. Matter Mater. Phys.* **2012**, *85*, 155208.
- (16) Urban, A.; Lee, J.; Ceder, G. *Adv. Energy Mater.* **2014**, *4*, 1400478.
- (17) Yan, F.; Zhang, X.; Yu, Y. G.; Yu, L.; Nagaraja, A.; Mason, T. O.; Zunger, A. *Nat. Commun.* **2015**, *6*, 7308.
- (18) Yang, D.; Lv, J.; Zhao, X.; Xu, Q.; Fu, Y.; Zhan, Y.; Zunger, A.; Zhang, L. *Chem. Mater.* **2017**, *29*, 524–538.
- (19) Zakutayev, A.; Zhang, X.; Nagaraja, A.; Yu, L.; Lany, S.; Mason, T. O.; Ginley, D. S.; Zunger, A. *J. Am. Chem. Soc.* **2013**, *135*, 10048–10054.
- (20) Gautier, R.; Zhang, X.; Hu, L.; Yu, L.; Lin, Y.; Sunde, T. O. L.; Chon, D.; Poeppelmeier, K. R.; Zunger, A. *Nat. Chem.* **2015**, *7*, 308–316.
- (21) Davies, D. W.; Butler, K. T.; Skelton, J. M.; Xie, C.; Oganov, A. R.; Walsh, A. *Chem. Sci.* **2018**, *9*, 1022–1030.
- (22) Dong, X.; Oganov, A. R.; Goncharov, A. F.; Stavrou, E.; Lobanov, S.; Saleh, G.; Qian, G.-R.; Zhu, Q.; Gatti, C.; Deringer, V. L.; Dronskowski, R.; Zhou, X.-F.; Prakapenka, V. B.; Konopkova, Z.; Popov, I. A.; Boldyrev, A. I.; Wang, H.-T. *Nat. Chem.* **2017**, *9*, 440–445.
- (23) Oganov, A. R.; Chen, J.; Gatti, C.; Ma, Y.; Ma, Y.; Glass, C. W.; Liu, Z.; Yu, T.; Kurakevych, O. O.; Solozhenko, V. L. *Nature* **2009**, *457*, 863–867.
- (24) Kirklin, S.; Saal, J. E.; Meredig, B.; Thompson, A.; Doak, J. W.; Aykol, M.; Ruhl, S.; Wolverton, C. *npj Comput. Mater.* **2015**, *1*, 15010.
- (25) Anand, S.; Xia, K.; Hegde, V. I.; Aydemir, U.; Kocovski, V.; Zhu, T.; Wolverton, C.; Snyder, G. J. *Energy Environ. Sci.* **2018**, DOI: 10.1039/C8EE00306H.
- (26) Isaacs, E. B.; Wolverton, C. *Chem. Mater.* **2018**, *30*, 1540–1546.
- (27) Zhou, Y.; Wang, H.; Zhu, C.; Liu, H.; Tse, J. S.; Ma, Y. *Inorg. Chem.* **2016**, *55*, 7026–7032.
- (28) Su, C.; Lv, J.; Li, Q.; Wang, H.; Zhang, L.; Wang, Y.; Ma, Y. *J. Phys.: Condens. Matter* **2017**, *29*, 165901.
- (29) Zhang, Y.; Wang, H.; Wang, Y.; Zhang, L.; Ma, Y. *Phys. Rev. X* **2017**, *7*, 011017.
- (30) Zurek, E.; Yao, Y. *Inorg. Chem.* **2015**, *54*, 2875–2884.
- (31) Shamp, A.; Terpstra, T.; Bi, T.; Falls, Z.; Avery, P.; Zurek, E. *J. Am. Chem. Soc.* **2016**, *138*, 1884–1892.
- (32) Avery, P.; Falls, Z.; Zurek, E. *Comput. Phys. Commun.* **2018**, *222*, 418–419.
- (33) Oliynyk, A. O.; Adutwum, L. A.; Rudyk, B. W.; Pisavadia, H.; Lotfi, S.; Hlukhyi, V.; Harynuk, J. J.; Mar, A.; Brgoch, J. *J. Am. Chem. Soc.* **2017**, *139*, 17870–17881.
- (34) Oliynyk, A. O.; Mar, A. *Acc. Chem. Res.* **2018**, *51*, 59–68.
- (35) *Complex Metallic Alloys: Fundamentals and Applications*. Dubois, J. M.; Belin-Ferre, E., Eds.; Wiley-VCH, 2011.
- (36) Fredrickson, D. C.; Lee, S.; Hoffmann, R. *Inorg. Chem.* **2004**, *43*, 6159–6167.
- (37) Fredrickson, D. C.; Lee, S.; Hoffmann, R. *Angew. Chem., Int. Ed.* **2007**, *46*, 1958–1976.
- (38) Fredrickson, D. C. *Acc. Chem. Res.* **2018**, *51*, 248–257.
- (39) Wang, F.; Miller, G. J. *Inorg. Chem.* **2011**, *50*, 7625–7636.
- (40) Berger, R. F.; Walters, P. L.; Lee, S.; Hoffmann, R. *Chem. Rev.* **2011**, *111*, 4522–4545.
- (41) Le Bail, A. *J. Appl. Crystallogr.* **2005**, *38*, 389–395.
- (42) Glass, C. W.; Oganov, A. R.; Hansen, N. *Comput. Phys. Commun.* **2006**, *175*, 713–720.
- (43) Curtarolo, S.; Kolmogorov, A. N.; Cocks, F. H. *CALPHAD: Comput. Coupling Phase Diagrams Thermochem.* **2005**, *29*, 155–161.
- (44) Gautier, R.; Zhang, X.; Hu, L.; Yu, L.; Lin, Y.; Sunde, T. O. L.; Chon, D.; Poeppelmeier, K. R.; Zunger, A. *Nat. Chem.* **2015**, *7*, 308–316.
- (45) Hofmann, D. W. M.; Apostolakis, J. *J. Mol. Struct.* **2003**, *647*, 17–39.
- (46) Oganov, A. R.; Glass, C. W. *J. Chem. Phys.* **2006**, *124*, 244704.
- (47) LeCun, Y.; Bengio, Y.; Hinton, G. *Nature* **2015**, *521*, 436–444.
- (48) Goh, G. B.; Hodas, N. O.; Vishnu, A. *J. Comput. Chem.* **2017**, *38*, 1291–1307.
- (49) Park, W. B.; Chung, J.; Jung, J.; Sohn, K.; Singh, S. P.; Pyo, M.; Shin, N.; Sohn, K.-S. *IUCr* **2017**, *4*, 486–494.
- (50) Wei, J. N.; Duvenaud, D.; Aspuru-Guzik, A. *ACS Cent. Sci.* **2016**, *2*, 725–732.
- (51) Patra, T. K.; Meenakshisundaram, V.; Hung, J.-H.; Simmons, D. S. *ACS Comb. Sci.* **2017**, *19*, 96–107.
- (52) Schütt, K. T.; Arbabzadah, F.; Chmiela, S.; Müller, K. R.; Tkatchenko, A. *Nat. Commun.* **2017**, *8*, 13890.
- (53) Coley, C. W.; Barzilay, R.; Jaakkola, T. S.; Green, W. H.; Jensen, K. F. *ACS Cent. Sci.* **2017**, *3*, 434–443.
- (54) Matlock, M. K.; Dang, N. L.; Swamidass, S. J. *ACS Cent. Sci.* **2018**, *4*, 52–62.
- (55) Segler, M. H. S.; Kogej, T.; Tyrchan, C.; Waller, M. P. *ACS Cent. Sci.* **2018**, *4*, 120–131.
- (56) Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; Kudlur, M.; Levenberg, J.; Monga, R.; Moore, S.; Murray, D. G.; Steiner, B.; Tucker, P.; Vasudevan, V.; Warden, P.; Wicke, M.; Yu, Y.; Zheng, X., TensorFlow: a system for large-scale machine learning. *12th USENIX Symposium on Operating Systems Design and Implementation*, Savannah, GA, USA, 2016.
- (57) Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer, A., Automatic differentiation in PyTorch. *NIPS 2017 Workshop Autodiff Submissions*, Long Beach, CA, 2017.

- (58) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; Duchesnay, E. *J. Machine Learn. Res.* **2011**, *12*, 2825–2830.
- (59) Bengio, Y.; Courville, A.; Vincent, P. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1798–828.
- (60) Deep learning is particularly well suited for the large data sets that we encounter in this work. We have only trained the model on about 50000 crystal structures from the Inorganic Crystal Structure Database and Crystallography Open Database, which resulted in about half a million examples of crystallographically unique atomic sites. When developing this project, we wanted to create a model which could easily accommodate future inclusion of the Cambridge Structural Database (or any other crystallographic repository) into the training set. As of January 2018, the CSD contains more than 900000 structures and many millions of atomic sites. Other machine learning algorithms would struggle with data sets of this size.
- (61) Musil, F.; De, S.; Yang, J.; Campbell, J. E.; Day, G. M.; Ceriotti, M. *Chem. Sci.* **2018**, *9*, 1289–1300.
- (62) Schmidt, J.; Shi, J.; Borlido, P.; Chen, L.; Botti, S.; Marques, M. A. L. *Chem. Mater.* **2017**, *29*, 5090–5103.
- (63) Jones, E.; Oliphant, T.; Peterson, P. *SciPy: open source scientific tools for Python*, 2001; <http://www.scipy.org/>.
- (64) Maaten, L. v. d.; Hinton, G. *J. Machine Learn. Res.* **2008**, *9*, 2579–2605.
- (65) Hunter, J. D. *Comput. Sci. Eng.* **2007**, *9*, 90–95.
- (66) The cutoff value of 0.65 Å was taken to be slightly below the length of the H–H bond in the H<sub>2</sub> molecule (0.74 Å)—the shortest possible distance in simple chemical species. Interatomic distances closer than this cutoff are not chemically reasonable, and thus, the cutoff served to filter out erroneous entries in the data set, removing 432 structures from the ICSD and 408 structures from the COD. We used a conservative estimate to avoid unintentionally removing legitimate crystal structures.
- (67) Valle, M.; Oganov, A. R. *Acta Crystallogr., Sect. A: Found. Crystallogr.* **2010**, *66*, 507–517.
- (68) Su, H.; Maji, S.; Kalogerakis, E.; Learned-Miller, E. Multi-view convolutional neural networks for 3D shape recognition. *Proc. IEEE Int. Conf. Comp. Vision*; Chile, 2015; IEEE, 2015; pp 945–953.
- (69) Pascanu, R.; Mikolov, T.; Bengio, Y., On the difficulty of training recurrent neural networks. In *International Conference on Machine Learning*; Atlanta, GA, 2013; IMLS, 2013; pp 1310–1318.
- (70) Chan, K. K. H.; Tang, S. *Opt. Express* **2011**, *19*, 26891–26904.
- (71) The choice of the Kaiser–Bessel over Gaussian blurring function was also influenced by the fact that the former quickly decays to zero, which improves computation efficiency.
- (72) He, H.; Garcia, E. A. *IEEE Trans. Knowl. Data Eng.* **2009**, *21*, 1263–1284.
- (73) Kotsiantis, S.; Kanellopoulos, D.; Pintelas, P. *GESTS Int. Trans. Comput. Sci. Eng.* **2006**, *30*, 25–36.
- (74) Kingma, D. P.; Welling, M. Auto-encoding variational bayes. *arXiv.org* **2013**, 1312.
- (75) Glorot, X.; Bengio, Y. *Proc. Machine Learn. Res.* **2010**, *9*, 249–256.
- (76) Ioffe, S.; Szegedy, C. *Proc. Machine Learn. Res.* **2015**, *37*, 448–456.
- (77) One epoch consists of one full training cycle on the training set.
- (78) Wattenberg, M.; Viégas, F.; Johnson, I. How to use t-SNE effectively. *Distill* **2016**, <http://distill.pub/2016/misread-tsne>.
- (79) Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A., Going deeper with convolutions. In *Proc. IEEE conf. comp. vision pattern recognition*; IEEE: Boston, MA, 2015.
- (80) Raccuglia, P.; Elbert, K. C.; Adler, P. D. F.; Falk, C.; Wenny, M. B.; Mollo, A.; Zeller, M.; Friedler, S. A.; Schrier, J.; Norquist, A. J. *Nature* **2016**, *533*, 73–76.
- (81) It is also worth mentioning that convolutional neural networks, used in the best image recognition models, share a similar shortcoming due to the pooling operation. See, for example: Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* **2014**, 580–587.
- (82) Le Page, Y. *MRS Bull.* **2006**, *31*, 991–994.