Polycrystal Graph Neural Network

Minyi Dai\textsuperscript{1}, Mehmet F. Demirel\textsuperscript{2}, Xuanhan Liu\textsuperscript{2}, Yingyu Liang\textsuperscript{2}, Jia-Mian Hu\textsuperscript{1}\textsuperscript{*}

\textsuperscript{1}Department of Materials Science and Engineering, University of Wisconsin-Madison, Madison, WI, 53706, USA
\textsuperscript{2}Department of Computer Sciences, University of Wisconsin-Madison, Madison, WI, 53706, USA

Abstract

We develop a polycrystal graph neural network (PGNN) model for predicting the properties of three-dimensional (3D) polycrystalline microstructures. Compared to existing GNN models in this domain, the PGNN model considers the physical features of both the grains and grain boundaries, and therefore better aligns with the physical principles. To evaluate the property prediction and transfer learning performance of the PGNN model, a dataset of 5000 3D polycrystalline microstructures and corresponding effective properties is generated by high-throughput physics-based simulations. The effective Li-ion conductivities and elastic stiffness coefficients of the polycrystalline $\text{Li}_7\text{La}_3\text{Zr}_2\text{O}_{12}$ ceramic are utilized as the examples. Trained with 4000 data points, the PGNN model achieves a property prediction error of $<1.4\%$, which is superior to all the baseline machine learning models, including a linear artificial neural network, a convolutional neural network (CNN), and the ResNet. Furthermore, compared to the PGNN model trained from scratch, a pretrained PGNN model demonstrates a significant shorter training time and lower prediction error. Our fast, accurate, and transferable PGNN model is therefore well suited for harnessing large-scale 3D polycrystalline microstructure images that usually contain billions of voxels per image, which can be too slow to address with voxel-based machine learning models, such as the CNN.

\textsuperscript{*}E-mail: jhu238@wisc.edu
1. Introduction

Establishing the microstructure-property relationship lies at the core of materials science and engineering. A microstructure refers to the spatial distribution of an order parameter such as the crystallographic orientation, concentration, magnetization, and polarization, which describe the local structural, compositional, magnetic, and electric features of a material, respectively. In a three-dimensional (3D) characterization apparatus or computational model, a microstructure is typically represented by an image of 3D discretized voxels, and each voxel store the local physical features of interest. The effective property of a material measures the average of the individual responses of each voxel to an applied field [1]. However, such individual response is determined by not only the physical features stored in an individual voxel but also the physical fields (e.g., stress, magnetic, and electric field) generated by physical features from its neighboring and non-neighboring voxels. Therefore, an accurate prediction of the effective material properties requires a consideration of both the short-range and long-range physical (e.g., mechanical, magnetic, and electric) interactions among different voxels of a 3D microstructure [2].

Machine learning has recently emerged as a powerful tool to predict the microstructure-property relationship [3–5]. Compared to physics-based models, a machine learning model has two main advantages. First, once trained and validated, the prediction by a machine learning model is usually orders of magnitude faster. Second, for complex problems for which it is not possible to develop an accurate physics-based model but possible to acquire enough data by experiments, training a surrogate machine learning model would be the only viable option. Different types of machine learning models have been developed to predict the microstructure-property relationship [6–24]. Among them, the convolutional neural network (CNN) based methods [10–22] are particularly attractive. Specifically, CNN is well-established and can be directly utilized to extract a low-dimensional embedding of a microstructure image with little bias from human researchers. The obtained embedding can then be linked to the target properties via an artificial neural network (ANN) model. Moreover, the short-range interactions between neighboring voxels are automatically considered during the convolution process. However, the long-range interactions among the voxels are typically excluded. Furthermore, the voxel-by-voxel-based microstructure representation in a CNN model significantly increases the computational burden, making it challenging to apply CNN to large-scale microstructure data acquired via modern tomography methods such as high-energy diffraction microscopy, in which case each 3D image routinely contains billions of voxels [25–27].

To consider both the short-range and long-range interactions among the voxels and reduce the computational cost, graph-based machine learning models, such as the graph neural network (GNN), have recently been applied to predict the microstructure-property relationship [28–31]. All such applications are focused on polycrystalline materials, because it is straightforward to use a graph — which consists of a set of interacting nodes where the neighboring ones are connected by the edges — to represent a polycrystalline microstructure which contains a set of interacting grains where the neighboring grains are joined by the grain boundaries. Figure 1a-b show the process of building a microstructure graph from a 3D polycrystalline microstructure by labelling each grain and grain boundary and storing their physical properties. Such graph-based microstructure representation in GNN is computationally more efficient than the image-voxel-based representation approach in CNN, because one grain typically contains multiple voxels.
In this article, we develop a Polycrystal Graph Neural Network (PGNN) model for predicting the properties of polycrystalline microstructures. Compared to existing GNN models which either completely omit the grain boundaries (that is, no edge vectors) [30,31] or do not incorporate the physical features (e.g., conductivity and elastic moduli) of the grain boundaries in the edge feature vector [28,29], the present PGNN model considers the physical features of both the grains and grain boundaries. This is important because a grain boundary should be treated as an individual structure element that has its own structural, electrical, magnetic, and mechanical features. To evaluate the property prediction performance of our PGNN model, we generate a dataset of 5000 3D polycrystalline microstructures and their effective ion conductivities and elastic stiffness coefficients using high-throughput physical simulations. The PGNN model is then trained, validated, and tested based on this dataset. The optimized PGNN model gives a mean absolute relative error of < 1.4% on unseen testing dataset, and this error is lower than those from all three baseline machine learning models, which include a linear ANN model, a 3D CNN model that has recently been utilized to predict the effective elastic moduli of two-phase composites [32], and the ResNet [33] which is a widely used general-purpose CNN model. Furthermore, compared to a PGNN model trained from scratch, the PGNN model pre-trained using dataset of microstructures and effective ion conductivities shows a significantly shorter training time and lower prediction error in predicting the effective elastic stiffness coefficients of these microstructures.

2. Methods

2.1. Building a graph for each microstructure and the principles of feature selection

Figure 1a shows a representative 3D polycrystalline microstructure, where each grain is colored according to its crystallographic orientation defined by the three Euler angles \((\alpha, \beta, \gamma)\) and the grain boundary is colored black. To build a graph for this microstructure, each grain (a 3D object) and each grain boundary (a 2D plane) are labeled and represented by a node and an edge, respectively. For each node and edge, there is an associated vector (denoted as \(v_i\) and \(e_{ij}\)) which stores the physical features of the corresponding grain and grain boundary, respectively. The subscript of the vectors \((i,j=1,2,3,..,N)\) represent the label of a specific grain, where \(N\) is the total number of grains in one microstructure. For example, \(e_{12}\) refers to the edge that connects the two grains which are labeled as ‘1’ and ‘2’, as shown by Fig. 1b. Thus, the whole microstructure can be converted into an undirected graph with nodes and edges. Figure 1c shows the full microstructure graph, where each node is placed according to the center of the corresponding grain and is colored based on the number of edges connected to it.

The target properties are the effective ion conductivities \(\kappa_{xx}^{\text{eff}}, \kappa_{yy}^{\text{eff}}, \kappa_{zz}^{\text{eff}}\) or the effective elastic stiffness coefficients \(c_{11}^{\text{eff}}, c_{12}^{\text{eff}}, c_{44}^{\text{eff}}\). In this regard, the physical features in the vectors \(v_i\) and \(e_{ij}\) should include the ion conductivity or elastic moduli of individual grains or grain boundaries, in addition to their crystallographic (e.g., the Euler angles) and geometrical (e.g., grain size, grain boundary interface area and curvature [34]) features. Without including the local conductivities or local elastic moduli, the application of machine learning to predict the effective conductivities or moduli could fall into the pitfall of finding correlations that do not necessarily exist. Although one can exclude the local physical properties of the grains and grain boundaries if these properties are spatially uniform in the same microstructure and invariant among different microstructures of a
dataset, that would be an unrealistic condition. In fact, experiments have shown that the local physical features (e.g., the local conductivity of individual grain [35]) can be rather different even in the same microstructure.

With this in mind, we select 10 components for each node feature vector \( v_i \), including three for defining the coordinates of the grain center, one for the grain size, three for the grain orientation, and three for the local ion conductivity. Specifically, since each voxel has its unique coordinates \((p_x, p_y, p_z)\) as well as a label \( n_i \) indicating that it’s part of the \( i \)-th grain, the coordinates of the center of the \( i \)-th grain can then be calculated as the average coordinates \((<p_x>, <p_y>, <p_z>)\) of all the voxels labeled as \( n_i \). The grain size is defined as the number of voxels occupied by a grain. The grain orientation is represented by the three Euler angles \((\alpha, \beta, \gamma)\). The local ion conductivity is represented by the three diagonal components of the conductivity matrix \( \kappa^{g}_{xx}, \kappa^{g}_{yy}, \kappa^{g}_{zz} \), where the superscript ‘g’ refers to the grain and the prime symbol in the subscripts indicate that these properties are in the crystallographic coordinate system. For the edge feature vector, we select four components, including the grain boundary thickness and conductivity \( \kappa^{gb}_{xx}, \kappa^{gb}_{yy}, \kappa^{gb}_{zz} \). These four features were selected because they are part of the input parameters of our physics-based model for the prediction of effective ion conductivity. We also note that all the grain boundaries have an identical thickness of one voxel in our computer-generated polycrystalline microstructure (see Section 3.1). In practice, each grain boundary should have a non-uniform thickness distribution within its 2D plane [36]. Furthermore, the Euler angles in the voxels of the same grain boundary are randomly given in this dataset and different from each other. For simplicity, the orientation of the grain boundary [37] was not incorporated as the feature of the edge vector \( e_{ij} \) in this work. Geometrical features of grain boundaries such as interface area and curvature, which can be determined from topological analysis of voxel-based 3D polycrystalline microstructure image [34], are also neglected. The necessity and possible benefits of incorporating these additional grain boundary features will be addressed in future works.

Based on the node and edge feature vectors \( v_i \) and \( e_{ij} \), we can now construct three matrices for one microstructure graph \( G = \{F, A, E\} \), including an \( N \times 10 \) node feature matrix \( F \) that stores the \( v_i \) of each node, an \( N \times N \) adjacency matrix \( A \) that stores the adjacency relation (where \( A_{ij} = 1 \) if grain \( i \) and grain \( j \) are neighbors and \( A_{ij} = 0 \) otherwise), and an \( N \times N \times 4 \) edge feature matrix \( E \) that stores the \( e_{ij} \) of each edge. The information of the adjacency matrix \( A \) is contained in the edge vector \( e_{ij} \): in the edge feature matrix \( E \), \( E_{ij} = e_{ij} \) if \( A_{ij} = 1 \) and \( E_{ij} = 0 \) otherwise. These three matrices together constitute the graph input of the PGNN model.

2.2. Polycrystal Graph Neural Network (PGNN)

A graph neural network (GNN) model contains three main components: (1) the graph input \( G = \{F, A, E\} \) herein, as discussed above; (2) the neural network structure (see Fig. 1b); (3) the update function. One of the most unique features of the GNN is the ability to pass message from node to node [37]. The features of the edge that connects two neighboring nodes can be involved in the message passing. By passing messages for multiple times via the use of multiple graph convolutional layers (or referred to as ‘message passing layers’ more generally), it is possible to let the physical features stored in all edges and nodes interact with each other, thereby incorporating both the short-range and long-range physical interactions among the local structural
elements (grains, precipitates, dislocations) in a microstructure. The mathematical function that implements the message passing is called an ‘update function’. The selection of an update function could significantly influence the property prediction performance of a GNN model [38], and such selection highly depends on the graph input and the inherent physical laws that govern a specific material property. In the current polygrain graph neural network (PGNN) model, we select an update function (see Eq. (2) below) utilized in the crystal graph convolutional neural network (CGCNN) model [38] and adapt it to take microstructure graphs as the input. Regarding the neural network structure, there are four main layers in our PGNN model as shown in Fig. 1b.

(1) The embedding layer. The embedding layer takes the node feature matrix \( F \) as the input and converts the original node features into node embeddings \( F^0 \) through a linear transformation,

\[
F_e^0 = FW_e
\]

where \( W_e \) is a trainable weight matrix. The embedding layer allows for transferring the remaining layers to learn other datasets with different feature matrix sizes.

(2) The graph convolutional layer. Multiple graph convolutional layers can be used. The first graph convolutional layer takes three matrices as its input, including the node embedding matrix \( F_g^0 = F_e^0 \), the adjacency matrix \( A \), and the edge feature matrix \( E \). The graph convolutional layer updates the features of all the nodes through the same update function, while the matrix \( A \) and \( E \) remain unchanged in this process. Specifically, after passing the \( n \)th graph convolution layer \((n=1,2,3,...)\), the node embedding matrix \( F_g^{n-1} \) is updated to \( F_g^n \) through,

\[
F_g^n = ReLU \left( AF_g^{n-1} + \sum_{i=1}^{N} E[:, i, :] \odot F_g^{n-1} \right) W_g^n
\]

where \( \sum_{i=1}^{N} E[:, i, :] \) is the sum of the features of all edges connected to the \( i \)th node, which changes the matrix dimension from \( N \times N \times 4 \) to \( N \times 4 \); the symbol \( \odot \) means matrix concatenation; \( W_g^n \) is the trainable weight matrix of the \( n \)th graph convolutional layer, and \( ReLU \) (Rectified Linear Unit) is an activation function, where \( ReLU(x) = max(0, x) \). The updated features of a specific node are determined by the features of all the edges connected to it as well as the features of all its neighboring nodes. By utilizing multiple graph convolutional layers, the interactions among all grains and grain boundaries can be considered.

(3) The node-level fully connected layer. Multiple node-level fully connected layers can be used. The first node-level fully connected layer takes the node embedding matrix generated by the last graph convolutional layer as the input, that is, \( F_g^n \). After passing the \( m \)th layer, the node embedding matrix \( F_f^{m-1} \) is updated to \( F_f^m \) through,

\[
F_f^m = \text{softplus}(\text{softplus}(F_f^{m-1})W_f^m)
\]

where the \( W_f^m \) is the trainable weight matrix of the \( m \)th node-level fully connected layer; \( \text{softplus}(x) = \log(1 + \exp(x)) \) is utilized as the nonlinear activation function, but the \( ReLU \) can also be used. Such node-wise dense layers have previously been used in GNN models developed for predicting the properties of atomic crystal structures and molecules [38,39], and the
goal of introducing such layers is to provide more trainable parameters while maintain scalability with respect to the number of nodes.

(4) **Fully connected layer.** The node embedding matrix generated by the node-level fully connected layer is flattened to a one-dimensional vector \( f^0 \) and used as the input of the fully connected layer. After passing the \( l \)th layer, the vector \( f^{l-1} \) is updated to \( f^l \) through

\[
f^l = ReLU(f^{l-1}W^l),
\]

where \( W^l \) is the trainable weight matrix of the \( l \)th layer. In the last fully connected layer which is linked to the target properties, the \( ReLU \) activation function is removed to enable the prediction of negative targets.

### 3. Datasets Generation

To evaluate the property prediction performance of the PGNN model against the baseline machine learning models, a dataset of 5000 3D polycrystalline microstructures and their corresponding effective ion conductivities is generated. Specifically, each data point is denoted as \( \{X, Y\} \), where \( X \) is the microstructure input and has a dimension of \( d_x \times d_y \times d_z \times 3 \), where \( d_x = d_y = d_z = 64 \) are the number of voxels along the three Cartesian axes and the additional three channels store the three Euler angles; \( Y \) refers to the effective ion conductivities \( (\kappa_{xx}^{eff}, \kappa_{yy}^{eff}, \kappa_{zz}^{eff}) \), which is a 3×1 vector (see Fig. 1b). In this dataset, 4000, 500, and 500 data points are utilized for model training, validation, and testing, respectively. To evaluate the transfer learning performance of the PGNN model, a smaller dataset of 604 data points is generated, where each data point is denoted as \( \{X, Y'\} \). Here the microstructure input of 604 different \( X \) is randomly picked from the training dataset, and the \( Y' \) refers to the corresponding effective elastic stiffness coefficients \( (c_{11}^{eff}, c_{12}^{eff}, c_{44}^{eff}) \), which are utilized as the new target properties.

The generation of the microstructure input \( X \) is based on an in-house Voronoi Tessellation code (Section 3.1). The calculations of the effective ion conductivities \( Y \) and the effective elastic stiffness matrix \( Y' \) are performed based on high-throughput physics-based numerical simulations, which take the local Euler angles \( (\alpha, \beta, \gamma) \) and the local Li-ion conductivity or local elastic stiffness matrix of the polycrystalline Li\(_7\)La\(_3\)Zr\(_2\)O\(_{12}\) (LLZO) ceramics as the input. The Euler angles are randomly generated and assigned to each voxel of \( X \), while the local physical properties (local conductivities and elastic stiffness coefficients) are taken from literature (details are in Section 3.2 and 3.3). The LLZO is considered as a representative material because it is one of the most promising and well-studied solid electrolyte materials for all-solid-state lithium batteries [40,41]. The effective Li-ion conductivity and the effective elastic stiffness matrix are two key properties of the solid electrolytes for lithium batteries [42,43].

#### 3.1. Generation of 3D polycrystalline microstructure by Voronoi Tessellation

The 3D polycrystalline microstructures, each having a size of \( 64 \times 64 \times 64 \) voxels, are generated through an in-house Python code based on Voronoi Tessellation. We begin by randomly selecting \( N \) voxels as the seeds of \( N \) grains, and label them from 1 to \( N \). For each of the remaining voxels, its distances to all the \( N \) seeds are calculated. Each voxel is then given the same label as its nearest seed. As a result, the 3D cube is divided into \( N \) different regions with unique labels, representing
$N$ different grains. The grain boundary is then added manually based on the labels. Specifically, if the label of a voxel is different from those of its neighboring voxels, the voxel will be marked as the grain boundary and re-labeled as 0. The total number of seeds in each microstructure is randomly given.

After the generation of the grains and grain boundaries, voxels inside the same grain are given the same set of Euler angles ($\alpha$, $\beta$, $\gamma$), where the values of all three angles are randomly selected between $0^\circ$ and $360^\circ$. In different grains of a 3D microstructure, the Euler angles are different. The microstructure is not textured, as shown in Fig. 1a. In all voxels that are labeled as the grain boundary, the values of their Euler angles are also randomly selected. Figure 2a shows the distribution of local Li-ion conductivity in one 2D slice of the 3D microstructure shown in Fig. 1a. For this microstructure, the grain boundary has a higher Li-ion conductivity than the grain. This is consistent with the calculated distribution of local electric current density $J_z$ in this 2D slice upon applying an electric field of $100$ V/m along the +z direction, as shown in Fig. 2b. Figure 2c shows the statistical distributions of the grain conductivity $\kappa_{g'}^{g}$ and grain boundary conductivity $\kappa_{gb}^{g}$ for all the 5000 microstructures, and the distribution of the calculated effective conductivity $\kappa_{eff}$ is shown in Fig. 2d. The detailed setting of the conductivities is discussed in Section 3.2. As shown in Figs. 2c-d, although the distributions of both the input local grain and grain boundary conductivities are largely uniform, the distribution of the target effective conductivity is Gaussian. This suggests the presence of a strong nonlinear correlation between the input and output properties. Figure 2e shows the grain size distribution for the microstructure shown in Fig. 1a, where most of the grains have a size between 294 and 3442 voxels. In the whole dataset, the number of grains per microstructure image ($N$) has a uniform distribution between 10 and 400, as shown in Fig. 2f.

3.2. Calculation of the effective Li-ion conductivity matrix through physics-based simulations

The effective electrical conductivity $\kappa_{eff}$ is given by $\kappa_{eff} = \frac{(J_k(r))}{E_l(k,l=x,y,z)}$ where $E_l$ is the electrical field applied along the $l$ axis, and $(J_k(r))$ is the volumetric average of the $k$ component of the local electrical current density $J_k(r)$. The 3D spatial distribution of the $J_k(r)$ under the applied $E_l$ (an example is shown in Fig. 2c) is obtained by numerically solving the steady-state continuity equation,

$$\nabla \cdot J(r) = \nabla \cdot (\kappa(r)E) = 0$$

by a numerically efficient Fourier spectral iterative perturbation (FSIP) method [44,45]. Here $\kappa(r)$ is the local ion conductivity in the sample coordinate system. Assuming all the non-diagonal components are zero, $\kappa(r)$ is given by,

$$\kappa(r) = R\kappa'(r)R^T = R \begin{bmatrix} \kappa_{x'x'}(r) & 0 & 0 \\ 0 & \kappa_{y'y'}(r) & 0 \\ 0 & 0 & \kappa_{z'z'}(r) \end{bmatrix} R^T,$$

where the rotation matrix $R$ is defined based on the local Euler angles ($\alpha$, $\beta$, $\gamma$),
\[
\mathbf{R} = \begin{bmatrix}
\cos \alpha \cos \gamma - \cos \beta \sin \alpha \sin \gamma & -\cos \alpha \sin \gamma - \cos \beta \cos \alpha \sin \gamma & \sin \alpha \sin \beta \\
\sin \beta \sin \gamma & \cos \alpha \cos \beta & -\cos \alpha \sin \beta \\
\cos \beta \sin \alpha + \cos \alpha \cos \beta \sin \gamma & \cos \beta \cos \gamma - \cos \alpha \sin \gamma & \sin \alpha \sin \beta
\end{bmatrix}.
\]

(7)

\(\kappa_{x'x'}, \kappa_{y'y'}\) and \(\kappa_{z'z'}\) are the local Li-ion conductivities in the crystallographic coordinate system. In voxels that are labeled as the grain, \((\kappa_{x'x'}, \kappa_{y'y'}, \kappa_{z'z'}) = (\kappa_{x'x'}, \kappa_{y'y'}, \kappa_{z'z'})\). Here we assume that the grains in the LLZO ceramic have a tetragonal crystal symmetry due to residual stress. If all grains have a cubic crystal symmetry, \(\kappa_{x'x'}(\mathbf{r}) = \kappa_{y'y'}(\mathbf{r}) = \kappa_{z'z'}(\mathbf{r})\). In that case, the local conductivity matrix \(\kappa\) would be independent of the local grain orientations since \(\kappa = \mathbf{R} \kappa' \mathbf{R}^T = \kappa' \mathbf{l} = \kappa'\), where \(\mathbf{l}\) is the identity matrix. For this reason, we set \(\kappa_{x'x'} = \kappa_{y'y'} = 1.335 \times 10^{-5}\) S/m, which are the reported values for LLZO ceramics [46], but further set \(\kappa_{z'z'} = r_1 \kappa_{x'x'}\), where \(r_1\) is randomly chosen from a uniform distribution in the range of 0.8-1.2. In voxels that are labeled as the grain boundary, \((\kappa_{x'x'}, \kappa_{y'y'}, \kappa_{z'z'}) = (\kappa_{x'x'}, \kappa_{y'y'}, \kappa_{z'z'})\), where \(\kappa_{g^b_{x'x'}} = 10^{r_2} \kappa_{x'x'}, \kappa_{g^b_{y'y'}} = 10^{r_3} \kappa_{y'y'}, \kappa_{g^b_{z'z'}} = 10^{r_4} \kappa_{z'z'}\). Here, \(r_2, r_3\) and \(r_4\) are randomly selected from a uniform distribution in the range of -3 and 2 because the reported conductivity of grain boundaries in LLZO can be higher or lower than the grain conductivity [46]. Importantly, among the 5000 computer-generated polycrystalline microstructures, each one has a unique set of \(r_1, r_2, r_3\) and \(r_4\) values. Therefore, the distributions of local Li-ion conductivity in the 5000 microstructures are all different from each other. In each microstructure, the same set of \((\kappa_{g^b_{x'x'}}, \kappa_{g^b_{y'y'}}, \kappa_{g^b_{z'z'}})\) is utilized for all the grains, and the same set of \((\kappa_{x'x'}, \kappa_{y'y'}, \kappa_{z'z'})\) is utilized for all the grain boundaries.

3.3. Calculation of the effective elastic stiffness matrix through physics-based simulations

The effective elastic stiffness matrix \(c_{klmn}^{\text{eff}}\) is calculated by
\[
c_{klmn}^{\text{eff}} = \frac{\langle \sigma_{kl}(\mathbf{r}) \rangle}{\epsilon_{mn}} (k,l,m,n=x,y,z) \text{ where } \epsilon_{mn} \text{ is the applied strain, and } \langle \sigma_{kl}(\mathbf{r}) \rangle \text{ is the volumetric average of the local stress } \sigma_{kl}(\mathbf{r}). \text{ Likewise, the subscripts (with no primes) indicate that these quantities are in the sample coordinate system. The 3D spatial distribution of the } \sigma_{kl}(\mathbf{r}) \text{ under the applied strain } \epsilon_{mn} \text{ is obtained by solving the mechanical equilibrium equation,}
\]
\[
\frac{\partial \sigma_{kl}}{\partial r_j} = 0, \text{ i.e., } \nabla_j [c_{klmn}(\mathbf{r}) (\epsilon_{mn} - \epsilon_{mn}^0)] = 0
\]
\[
(8)
\]
by the FSIP method as well [47,48]. Here \(\epsilon_{mn}^0\) is the position-dependent eigenstrain related to grains, which is set as zero herein. \(c_{klmn}(\mathbf{r})\) is the local elastic stiffness coefficient in the sample coordinate system, given by
\[
c_{klmn}(\mathbf{r}) = R_{kp} R_{lq} R_{mr} R_{ns} c_{pqr}^{\text{eff}}(\mathbf{r})
\]
\[
(9)
\]
Where \(\mathbf{R}\) is the rotation matrix, the same as that in Eq. (7); \(c_{pqr}^{\text{eff}}(\mathbf{r})\) is the local elastic stiffness matrix in the crystallographic coordinate system. Since grains have a tetragonal crystal symmetry, the \(c'_{11}, c'_{12}, c'_{44}\) (expressed using the Voigt notation) are the three independent component of the 6x6 elastic stiffness matrix. For voxels that are labeled as grain, \((c_{11}', c_{12}', c_{44}') = (c'_{g,11}, c'_{g,12}, c'_{g,44})\). For half of the microstructures in the dataset, we set \(c'_{g,11}=1.87 \times 10^{11}\) Pa, \(c'_{g,12}=7.51 \times 10^{10}\) Pa, \(c'_{g,44}=7.10 \times 10^{10}\) Pa, which are reported values for Al-doped LLZO ceramics [49]. For the other
half, we set $c_{g,11}'=1.70\times10^{11}$ Pa, $c_{g,12}'=6.39\times10^{10}$ Pa, $c_{g,44}'=6.98\times10^{10}$ Pa, which are reported values for Ta-doped LLZO ceramics [49]. In voxels that are labeled as the grain boundary, $(c_{11}', c_{12}', c_{44}') = (c_{gb,11}', c_{gb,12}', c_{gb,44}')$, where $c_{gb,11}' = 10^{\alpha_5} c_{g,11}'$, $c_{gb,12}' = 10^{\alpha_6} c_{g,12}'$, and $c_{gb,44}' = 10^{\alpha_7} c_{g,44}'$. Here, $r_5$, $r_6$ and $r_7$ were randomly chosen from a uniform distribution in the range of -3 and 0 because the grain boundary is usually considered to be mechanically weaker than the grains due to the disordered nature.

4. Results

4.1. Model evaluation

The whole dataset is divided into the training dataset (4000 data points), the validation dataset (500 points) and the testing dataset (500 data points). For each epoch (that is, one complete pass of the training data points), all the weights mentioned in Section 2.2 are updated using the gradient descent method through,

$$W = W - \eta \frac{\partial L}{\partial W}$$

(10)

where $\eta$ is the learning rate; $L$ is the loss function, which is taken as the mean absolute error (MAE) between the predicted value $\hat{y}_i$ and the true value $y_i$ of the target in one batch, given by,

$$L = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| .$$

(11)

Here $n$ is the batch size, which is the number of data points used in each update process through Eq. (10). Both the $\hat{y}_i$ and $y_i$ have three components since there are three targets: $(\kappa_{xx}^{\text{eff}}, \kappa_{yy}^{\text{eff}}, \kappa_{zz}^{\text{eff}})$ or $(c_{11}^{\text{eff}}, c_{12}^{\text{eff}}, c_{44}^{\text{eff}})$.

We note that both the local and effective ion conductivity are in the range of $10^{-8}\sim10^{-2}$ S/m, which is small and comparable to the error of common machine learning models. Moreover, both the local and effective elastic stiffness matrix are in the range of $10^{7}\sim10^{12}$ Pa, which is extremely large and may cause exploding gradients due to an extremely large initial loss $L$. Thus, in both the PGNN and the baseline machine learning models, all the input and target conductivities and elastic stiffness coefficients are expressed in the log10 scale. Furthermore, the node features and edge features of the input graph are scaled using the min-max normalization. The minimum and maximum value of a specific feature, $f_{\text{min}}$ and $f_{\text{max}}$, are determined based on the prior knowledge and kept the same in different microstructures. The input feature $f$ is then scaled to $f'$ through,

$$f' = \frac{f - f_{\text{min}}}{f_{\text{max}} - f_{\text{min}}}$$

(12)

$f'$ ranges from 0 to 1. The $f_{\text{min}}$ and $f_{\text{max}}$ of each input feature are shown in Table I. Such normalization can rescale all the input features to the range of $[0,1]$. As a result, each feature will be treated equivalently, and the model performance will not be determined by large features. In addition, data normalization, as a linear transformation, can shorten the distance between the initial starting point and the minimum point [50] and thereby leads to a faster model training. Indeed, our control experiment shows that the min-max normalization of the input features results in a
faster and numerically more stable model training and significantly improves the property prediction performance.

Table I. The minimum and maximum value of each input features of the PGNN model.

| Features                  | $f_{\text{min}}$ | $f_{\text{max}}$ |
|---------------------------|------------------|------------------|
| Euler angle 1 $\alpha$   | 0                | 360              |
| Euler angle 2 $\beta$    | 0                | 180              |
| Euler angle 3 $\gamma$   | 0                | 360              |
| Position x, y, z          | 0                | 64               |
| Number of grids           | 0                | $64^3$           |
| $\log_{10} \kappa$       | -8               | -2               |
| $\log_{10} c$            | 7                | 12               |

The model performance is represented by the mean absolute relative error (MARE), given by,

$$\text{MARE} = \frac{1}{n} \sum_{i=1}^{n} \frac{|y_i - \hat{y}_i|}{|y_i|}$$

The MARE shows the relative difference between the predicted and the true values, and hence is commonly used when the values of the targets span across different scales.

The training of the PGNN model involves the optimization of different hyperparameters in two categories: (i) the network structure, which includes the number of graph convolutional layers and fully connected layers as well as the number of hidden units in the fully connected layers; (ii) the training parameters, which includes the number of epochs, batch size and learning rate. The PGNN model with different combinations of hyperparameters are independently trained using the training dataset. The model with the best performance on the validation dataset is identified as the optimized model. The hyperparameters of the optimized model are summarized in Table II.

Table II. Hyperparameters of the optimized PGNN model.

| Hyperparameter                               | Value |
|----------------------------------------------|-------|
| Number of graph convolutional layer          | 2     |
| Number of node-level fully connected layer   | 2     |
| Number of hidden units of the fully connected layer | $1024 (1^{st})/128 (2^{nd})/3 (3^{rd})$ |
| Batch size                                   | 10    |
| Learning rate                                | $1 \times 10^{-6}$ |
| Number of epochs                             | 50    |

The optimized model is tested independently on the testing dataset. Figure 4a shows the PGNN-predicted value vs. the true value of the $\kappa_{xx}^{\text{eff}}, \kappa_{yy}^{\text{eff}}, \kappa_{zz}^{\text{eff}}$ in log10 scale. For all three targets, the
MARE values are smaller than 1.4%. These very low prediction errors demonstrate the excellent prediction ability of the PGNN model on multiple targets.

4.2. Baselines

The PGNN model is benchmarked with three baseline machine learning models. It is found that the PGNN model, which considers both the short-range and long-range interactions among the grains and grain boundaries, shows the lowest MARE.

#1. A linear artificial neural network (ANN) with three fully connected layers. The numbers of hidden units of the fully connected layers are the same with those in Table II. The activation functions of all the fully connected layers are removed and hence the target properties are linearly related to the input features. This simple linear ANN model is chosen to investigate whether there exist strong linear correlations between input features and the model output.

#2. A convolutional neural network (CNN) model [32], which consists of several convolutional layers and fully connected layers. This CNN model was used to predict the multiple targets of effective elastic moduli of two-phase composites. The training parameters are slightly tuned to get an optimized model for the dataset we generated in this work.

#3. A ResNet model [33], which is one of the most successful modern CNN models and has a sophisticated network structure. Specifically, for conventional CNN models such as model #2, the input of the \( l + 1 \)th convolutional layer only includes the output of the \( l \)th convolutional layer. However, ResNet model utilizes residual blocks to take the outputs of both the \( l \)th convolutional layer and \( l + k \)th convolutional layer as the input of \( l + k + 1 \)th convolutional layer, thereby establishing a so-called ‘skip connection’ between the \( l + k + 1 \)th convolutional layer and the \( l \)th convolutional layer. The ResNet and the model #2 represent two extremes in the level of complexity for the network structure of a CNN model.

The inputs of all three baseline machine models are composed of the microstructure input \( \{ X \} \) (see Section 3) and the three local Li-ion conductivity component, which together form a matrix of 64 \( \times 64 \times 64 \times 6 \) for a single data point. For each data point, the target properties \( \{ Y \} \) are the three effective Li-ion conductivities \( (\kappa_{xx}^{\text{eff}}, \kappa_{yy}^{\text{eff}}, \kappa_{zz}^{\text{eff}}) \). The CNN and ResNet model directly take the 64 \( \times 64 \times 64 \times 6 \) matrix as the input with six channels (three Euler angles and three local conductivities). In the ANN model, this 64 \( \times 64 \times 64 \times 6 \) matrix is flattened to a 1D vector.

Figure 4 a-c show the predicted values vs. the true values of the baseline model #1-3, respectively. Model #1 gives a prediction error of around 5% on the testing dataset for all three targets. In model #2-3, the values of MARE are around 3% and 2.6%, respectively. The fact that the MARE is appreciably smaller in nonlinear machine learning models (that is, model #2-3 and the PGNN) demonstrates the existence of a robust nonlinear correlation between the input and output data, which is consistent with the statistical features of the input local ion conductivity (Fig. 2c) and the target effective ion conductivity (Fig. 2d).

4.3. Transfer learning results

To evaluate the transferability of the PGNN model, we compare the transferred model (denoted as the model ‘T’) and the model trained from scratch (the model ‘S’) on a new dataset of 604
microstructures (see details in Section 3) and their effective elastic stiffness coefficients ($c_{11}^{\text{eff}}, c_{12}^{\text{eff}}, c_{44}^{\text{eff}}$). Similarly, this new dataset of 604 data points was divided into a training dataset (80%), a validation dataset (10%), and a testing dataset (10%). The model ‘T’ and ‘S’ are trained on the same training dataset. The initial weights of the model ‘T’ are the trained weights of the optimized model based on the microstructure-conductivity dataset, while the initial weights of the model ‘S’ are randomly given. To improve the transfer learning performance, the target $\log_{10}\kappa^{\text{eff}}$ and $\log_{10}\sigma^{\text{eff}}$ are also normalized based on the $f_{\text{min}}$ and $f_{\text{max}}$ shown in Table I. The reason is the range of $\log_{10}\kappa^{\text{eff}}$ and $\log_{10}\sigma^{\text{eff}}$ are quite different. If a model trained on unnormalized conductivity dataset is directly transferred to the unnormalized stiffness coefficient dataset, the initial training MAE loss would be very high. This is different from the treatment in Sec 4.1 and Sec 4.2 (that is, for the results in Fig. 3) where only the input features are min-max normalized while the targets are not. As shown in Fig. 4a, the model ‘T’ gives a significantly lower initial training loss when the number of epochs is below five, and converges faster to a stable training MAE loss than the model ‘S’. The validation loss of the model ‘T’ is much lower than that of the model ‘S’. The training and validation loss curves indicate that transferring the PGNN model can accelerate the model training and give better prediction results on new datasets. Figure 4b shows the predicted values vs. the true values based on the testing dataset. As shown, the transferred model yields a low testing MARE of around 1.5% for all the three targets, which is particularly remarkable considering the small size of the training dataset (604 data points).

To further demonstrate the excellent transfer learning performance of the PGNN model, a data ablation study is performed based on five sub-datasets with a size of 100, 200, 300, 400 and 500 data points that are randomly selected from the whole 604 data points. Model ‘T’ and ‘S’ are then trained, validated, and tested on those sub-datasets. Three random number of data division are used to eliminate the influence of randomness in generating the sub-datasets. As shown in Fig. 4b, the model ‘T’ gives lower values of testing MARE with significantly lower standard deviation than the model ‘S’ for all five sub-datasets. Therefore, the PGNN model pretrained using a relatively large dataset of one property can significantly improve the prediction performance of another property with smaller available training dataset as well as the training speed.

5. Conclusion

In summary, we develop a polycrystal graph neural network (PGNN) model for predicting the effective properties of polycrystalline materials. The network structure and feature selection in this PGNN model allow for considering the physical features of both the grains and grain boundaries, which was not achieved by existing graph neural network (GNN) models developed for predicting the properties of polycrystals [28–31]. As another important contribution to the general field of microstructure informatics, we generate a large dataset of 5000 different 3D polycrystalline microstructures and their corresponding effective properties. The 3D polycrystalline microstructure input is generated using an in-house Voronoi Tessellation code, and their effective properties (ion conductivity and elastic stiffness matrix) are calculated using high-throughput physics-based simulations via the commercial software µ-Pro (mupro.co). The PGNN model trained using this dataset gives a testing error of < 1.4% on unseen datasets, which is superior to all the baseline machine learning models. Compared to a PGNN model trained from scratch, the
PGNN model pre-trained using the dataset of microstructure-conductivity yields a faster training speed and a lower error in predicting the effective mechanical properties of similar microstructures.

Overall, our PGNN model, which considers both the short-range and long-range interactions among the grains or grain boundaries of a microstructure, leads to a more accurate property prediction than the CNN models which at most considers the short-range interactions. Moreover, the grain-by-grain analysis of the polycrystalline microstructure in the PGNN is more memory-efficient than the voxel-by-voxel analysis in CNN and can potentially be integrated with the modern grain-by-grain microstructure characterization techniques such as the far-field high-energy diffraction microscopy (FE-HEDM) [43]. Furthermore, the excellent transfer learning performance of the PGNN model makes it valuable tool to predict the effective properties for which the available training dataset is small.

The property prediction performance of the present PGNN model could be further improved by feature engineering, such as introducing additional descriptors to represent the crystallographic and geometrical features of the grain boundaries [51]. The influence of feature selection on the property prediction performance of GNN models deserves further investigation. The property prediction performance of the current PGNN model could also be improved by optimizing the update function to better approximate the physical interactions among the local structure elements for building a physics-informed neural network [52–54]. Finally, we note that the graph-based presentation can in principle be extended to any other mesoscopic systems that contain a set of mutually interacting elements, such as a group of colloidal nanoparticles in liquid solutions, mechanically interacting particles in a composite battery cathode [55], electronic [56] or magnetic [57] quantum-dot cellular automata, and biological cells, thereby creating mesoscale graph inputs for a wider variety of downstream tasks (e.g., particle tracking [58], and prediction of the moving trajectories of particles [59,60]) in addition to the effective property prediction.

Acknowledgements

Acknowledgment is made to the donors of The American Chemical Society Petroleum Research Fund for partial support of this research, under the award PRF # 61594-DNI9 (M.D. and J.-M.H.). This work used the Extreme Science and Engineering Discovery Environment (XSEDE) Specifically, it used the Bridges-2 system at the Pittsburgh Supercomputing Center (PSC). Xuanhan Liu's participation was assisted by the National Science Foundation under grant OAC 2017072. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Dataset availability

Datasets of the microstructure-effective Li-on conductivity and the microstructure-effective elastic stiffness coefficients of polycrystalline LLZO ceramics are available in GitHub via https://github.com/mdai26/PGNN.

Code availability
The in-house codes of polycrystalline microstructure generation through Voronoi Tessellation and the PGNN model can both be accessed in GitHub via https://github.com/mdai26/PGNN.
Figure 1. (a) A typical 3D polycrystalline microstructure image in the dataset. There are 133 grains in this specific microstructure. Grains are colored based on their orientations (converted from their Euler angles). Grain boundaries are colored black. (b) Schematic of building an undirected microstructure graph from the microstructure image. (c) Microstructure graph built from the microstructure image in (a). (d) The neural network structure of the PGNN model.
Figure 2. (a) The distribution of the local Li-ion conductivity (represented by the conductivity matrix norm $||k||$) in a 2D slice of the microstructure shown in Fig. 1(a). (b) The distribution of local electrical current density component $J_z$ when an electric field of 100 V/m is applied along the $+z$ direction. (c) Distributions of grain conductivity $\log_{10} \kappa^g_{zz}$ and grain conductivity $\log_{10} \kappa^{gb}_{zz}$ of all the 5000 microstructures. (d) Distribution of effective conductivity $\log_{10} \kappa^{eff}_{zz}$ of all the 5000 microstructures. (e) Grain size distribution in the 3D microstructure shown in Fig. 1(a). (f) The distribution of the number of grains per microstructure in all 5000 microstructures.
Figure 3. The predicted values vs. the true values of the three targets of the testing dataset, obtained through (a) the PGNN model, (b) a linear ANN model, (c) a CNN model, and (d) the ResNet model. The PGNN model gives the lowest testing MARE in predicting all three targets. The three targets ($\kappa_{xx}^{\text{eff}}$, $\kappa_{yy}^{\text{eff}}$, $\kappa_{zz}^{\text{eff}}$) are the diagonal components of the effective Li-ion conductivity matrix of the LLZO ceramic.
Figure 4. (a) The MAE loss curves of the transferred model (model ‘T’) and the model trained from scratch (model ‘S’) on the training and validation datasets. (b) The testing MAE of the model ‘T’ and the model ‘S’ as a function of the size of the training dataset. (c) The predicted values vs. the true values of the three targets of the testing dataset, obtained using the model ‘T’. The three targets (\(c_{11}^{\text{eff}}\), \(c_{12}^{\text{eff}}\), \(c_{44}^{\text{eff}}\)) are the three independent components of the effective elastic stiffness matrix of the LLZO ceramic.
References

[1] J.F. Nye, Physical properties of crystals: their representation by tensors and matrices, Oxford university press, 1985.

[2] L.-Q. Chen, Y. Zhao, From classical thermodynamics to phase-field method, Prog Mater Sci. 124 (2022) 100868.

[3] D.L. McDowell, R.A. LeSar, The need for microstructure informatics in process–structure–property relations, MRS Bull. 41 (2016) 587–593.

[4] S.R. Kalidindi, Materials Science and Engineering: Chapter 18. Microstructure Informatics, Elsevier Inc. Chapters, 2013.

[5] S.R. Kalidindi, S.R. Niezgoda, A.A. Salem, Microstructure informatics using higher-order statistics and efficient data-mining protocols, Jom. 63 (2011) 34–41.

[6] A. Khosravani, A. Cecen, S.R. Kalidindi, Development of high throughput assays for establishing process-structure-property linkages in multiphase polycrystalline metals: Application to dual-phase steels, Acta Mater. 123 (2017) 55–69.

[7] Y.C. Yabansu, D.K. Patel, S.R. Kalidindi, Calibrated localization relationships for elastic response of polycrystalline aggregates, Acta Mater. 81 (2014) 151–160.

[8] Y.C. Yabansu, S.R. Kalidindi, Representation and calibration of elastic localization kernels for a broad class of cubic polycrystals, Acta Mater. 94 (2015) 26–35.

[9] M.I. Latypov, M. Kühbach, I.J. Beyerlein, J. Stinville, L.S. Toth, T.M. Pollock, S.R. Kalidindi, Materials Characterization Application of chord length distributions and principal component analysis for quantification and representation of diverse polycrystalline microstructures, Mater Charact. 145 (2018) 671–685.

[10] R. Kondo, S. Yamakawa, Y. Masuoka, S. Tajima, R. Asahi, Microstructure recognition using convolutional neural networks for prediction of ionic conductivity in ceramics, Acta Mater. 141 (2017) 29–38.

[11] Z. Yang, Y.C. Yabansu, R. Al-Bahrani, W. Liao, A.N. Choudhary, S.R. Kalidindi, A. Agrawal, Deep learning approaches for mining structure-property linkages in high contrast composites from simulation datasets, Comput Mater Sci. 151 (2018) 278–287.

[12] X. Li, Y. Zhang, H. Zhao, C. Burkhart, L.C. Brinson, W. Chen, A Transfer Learning Approach for Microstructure Reconstruction and Structure-property Predictions, Sci Rep. 8 (2018) 1–13.

[13] R. Cang, H. Li, H. Yao, Y. Jiao, Y. Ren, Improving direct physical properties prediction of heterogeneous materials from imaging data via convolutional neural network and a morphology-aware generative model, Comput Mater Sci. 150 (2018) 212–221.
[14] H. Chan, M. Cherukara, T.D. Loeffler, B. Narayanan, S.K.R.S. Sankaranarayanan, Machine learning enabled autonomous microstructural characterization in 3D samples, NPJ Comput Mater. 6 (2020) 1.

[15] C. Herriott, A.D. Spear, Predicting microstructure-dependent mechanical properties in additively manufactured metals with machine- and deep-learning methods, Comput Mater Sci. 175 (2020) 109599.

[16] C. Yang, Y. Kim, S. Ryu, G.X. Gu, Prediction of composite microstructure stress-strain curves using convolutional neural networks, Mater Des. 189 (2020) 108509.

[17] Q. Rong, H. Wei, X. Huang, H. Bao, Predicting the effective thermal conductivity of composites from cross sections images using deep learning methods, Compos Sci Technol. 184 (2019) 107861.

[18] J.R. Mianroodi, N. H Siboni, D. Raabe, Teaching solid mechanics to artificial intelligence—a fast solver for heterogeneous materials, NPJ Comput Mater. 7 (2021) 1–10.

[19] A. Frankel, K. Tachida, R. Jones, Prediction of the evolution of the stress field of polycrystals undergoing elastic-plastic deformation with a hybrid neural network model, Mach Learn Sci Technol. 1 (2020) 035005.

[20] A. Pandey, R. Pokharel, Machine learning based surrogate modeling approach for mapping crystal deformation in three dimensions, Scr Mater. 193 (2021) 1–5.

[21] A. Cecen, H. Dai, Y.C. Yabansu, S.R. Kalidindi, L. Song, Material structure-property linkages using three-dimensional convolutional neural networks, Acta Mater. 146 (2018) 76–84.

[22] X. Li, Z. Liu, S. Cui, C. Luo, C. Li, Z. Zhuang, Predicting the effective mechanical property of heterogeneous materials by image based modeling and deep learning, Comput Methods Appl Mech Eng. 347 (2019) 735–753.

[23] K. Choudhary, B. DeCost, C. Chen, A. Jain, F. Tavazza, R. Cohn, C.W. Park, A. Choudhary, A. Agrawal, S.J.L. Billinge, Recent advances and applications of deep learning methods in materials science, NPJ Comput Mater. 8 (2022) 1–26.

[24] S.R. Kalidindi, Feature engineering of material structure for AI-based materials knowledge systems, J Appl Phys. 128 (2020) 041103.

[25] U. Lienert, S.F. Li, C.M. Hefferan, J. Lind, R.M. Suter, J. v. Bernier, N.R. Barton, M.C. Brandes, M.J. Mills, M.P. Miller, B. Jakobsen, W. Pantleon, High-energy diffraction microscopy at the advanced photon source, JOM. 63 (2011) 70–77.

[26] R. Pokharel, J. Lind, S.F. Li, P. Kenesei, R.A. Lebensohn, R.M. Suter, A.D. Rollett, In-situ observation of bulk 3D grain evolution during plastic deformation in polycrystalline Cu, Int J Plast. 67 (2015) 217–234.
[27] H.F. Poulsen, Three-dimensional X-ray diffraction microscopy: mapping polycrystals and their dynamics, Springer Science & Business Media, 2004.

[28] D.C. Pagan, C.R. Pash, A.R. Benson, M.P. Kasemer, Graph Neural Network Modeling of Grain-scale Anisotropic Elastic Behavior using Simulated and Measured Microscale Data, ArXiv Preprint ArXiv:2205.06324. (2022).

[29] C. Shu, J. He, G. Xue, C. Xie, Grain Knowledge Graph Representation Learning: A New Paradigm for Microstructure-Property Prediction, Crystals (Basel). 12 (2022) 280.

[30] M. Dai, M.F. Demirel, Y. Liang, J.-M. Hu, Graph neural networks for an accurate and interpretable prediction of the properties of polycrystalline materials, NPJ Comput Mater. 7 (2021) 1–9.

[31] N.N. Vlassis, R. Ma, W. Sun, Geometric deep learning for computational mechanics Part I: anisotropic hyperelasticity, Comput Methods Appl Mech Eng. 371 (2020) 113299.

[32] C. Rao, Y. Liu, Three-dimensional convolutional neural network (3D-CNN) for heterogeneous material homogenization, Comput Mater Sci. 184 (2020) 109850.

[33] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016: pp. 770–778.

[34] D.J. Rowenhorst, A.C. Lewis, G. Spanos, Three-dimensional analysis of grain topology and interface curvature in a β-titanium alloy, Acta Mater. 58 (2010) 5511–5519.

[35] J. Song, Y. Zhou, N.P. Padture, B.D. Huey, Anomalous 3D nanoscale photoconduction in hybrid perovskite semiconductors revealed by tomographic atomic force microscopy, Nat Commun. 11 (2020) 1–9.

[36] P. Gao, R. Ishikawa, B. Feng, A. Kumamoto, N. Shibata, Y. Ikuhara, Atomic-scale structure relaxation, chemistry and charge distribution of dislocation cores in SrTiO3, Ultramicroscopy. 184 (2018) 217–224.

[37] J. Gilmer, S.S. Schoenholz, P.F. Riley, O. Vinyals, G.E. Dahl, Neural message passing for quantum chemistry, in: International Conference on Machine Learning, PMLR, 2017: pp. 1263–1272.

[38] T. Xie, J.C. Grossman, Crystal Graph Convolutional Neural Networks for an Accurate and Interpretable Prediction of Material Properties, Phys Rev Lett. 120 (2018) 145301.

[39] K.T. Schütt, H.E. Sauceda, P.-J. Kindermans, A. Tkatchenko, K.-R. Müller, Schnet—a deep learning architecture for molecules and materials, J Chem Phys. 148 (2018) 241722.

[40] Y. Ren, K. Chen, R. Chen, T. Liu, Y. Zhang, C. Nan, Oxide electrolytes for lithium batteries, Journal of the American Ceramic Society. 98 (2015) 3603–3623.
[41] C. Wang, K. Fu, S.P. Kammampata, D.W. McOwen, A.J. Samson, L. Zhang, G.T. Hitz, A.M. Nolan, E.D. Wachsman, Y. Mo, Garnet-type solid-state electrolytes: materials, interfaces, and batteries, Chem Rev. 120 (2020) 4257–4300.

[42] S. Kalnaus, W.E. Tenhaeff, J. Sakamoto, A.S. Sabau, C. Daniel, N.J. Dudney, Analysis of composite electrolytes with sintered reinforcement structure for energy storage applications, J Power Sources. 241 (2013) 178–185.

[43] M. Dixit, B. Vishugopi, W. Zaman, P. Kenesei, J.-S. Park, J. Almer, P. Mukherjee, K. Hatzell, Polymorphism of Garnet Solid Electrolytes and Its Implications on Grain Level Chemo-Mechanics, (2021).

[44] J.J. Wang, X.Q. Ma, Q. Li, J. Britson, L.-Q. Chen, Phase transitions and domain structures of ferroelectric nanoparticles: Phase field model incorporating strong elastic and dielectric inhomogeneity, Acta Mater. 61 (2013) 7591–7603.

[45] J.-M. Hu, B. Wang, Y. Ji, T. Yang, X. Cheng, Y. Wang, L.-Q. Chen, Phase-field based multiscale modeling of heterogeneous solid electrolytes: applications to nanoporous Li3PS4, ACS Appl Mater Interfaces. 9 (2017) 33341–33350.

[46] T.W. Heo, A. Grieder, B. Wang, M. Wood, T. Hsu, S.A. Akhade, L.F. Wan, L.-Q. Chen, N. Adelstein, B.C. Wood, Microstructural impacts on ionic conductivity of oxide solid electrolytes from a combined atomistic-mesoscale approach, NPJ Comput Mater. 7 (2021) 1–15.

[47] S. Bhattacharyya, T.W. Heo, K. Chang, L.-Q. Chen, A spectral iterative method for the computation of effective properties of elastically inhomogeneous polycrystals, Commun Comput Phys. 11 (2012) 726–738.

[48] G. Sheng, S. Bhattacharyya, H. Zhang, K. Chang, S.L. Shang, S.N. Mathaudhu, Z.K. Liu, L.Q. Chen, Effective elastic properties of polycrystals based on phase-field description, Materials Science and Engineering: A. 554 (2012) 67–71.

[49] S. Yu, R.D. Schmidt, R. Garcia-Mendez, E. Herbert, N.J. Dudney, J.B. Wolfenstine, J. Sakamoto, D.J. Siegel, Elastic properties of the solid electrolyte Li7La3Zr2O12 (LLZO), Chemistry of Materials. 28 (2016) 197–206.

[50] J. Sola, J. Sevilla, Importance of input data normalization for the application of neural networks to complex industrial problems, IEEE Trans Nucl Sci. 44 (1997) 1464–1468.

[51] C. Hu, Y. Zuo, C. Chen, S.P. Ong, J. Luo, Genetic algorithm-guided deep learning of grain boundary diagrams: addressing the challenge of five degrees of freedom, Materials Today. 38 (2020) 49–57.

[52] G.E. Karniadakis, I.G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, L. Yang, Physics-informed machine learning, Nature Reviews Physics. 3 (2021) 422–440.
[53] M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, J Comput Phys. 378 (2019) 686–707.

[54] W. Yan, J. Melville, V. Yadav, K. Everett, L. Yang, M.S. Kesler, A.R. Krause, M.R. Tonks, J.B. Harley, A novel physics-regularized interpretable machine learning model for grain growth, Mater Des. (2022) 111032.

[55] J. Li, N. Sharma, Z. Jiang, Y. Yang, F. Monaco, Z. Xu, D. Hou, D. Ratner, P. Pianetta, P. Cloetens, F. Lin, K. Zhao, Y. Liu, Dynamics of particle network in composite battery cathodes, Science (1979). 376 (2022) 517–521.

[56] I. Amlani, A.O. Orlov, G. Toth, G.H. Bernstein, C.S. Lent, G.L. Snider, Digital logic gate using quantum-dot cellular automata, Science (1979). 284 (1999) 289–291.

[57] R.P. Cowburn, M.E. Welland, Room temperature magnetic quantum cellular automata, Science (1979). 287 (2000) 1466–1468.

[58] J. Duarte, J.-R. Vlimant, Graph neural networks for particle tracking and reconstruction, in: Artificial Intelligence for High Energy Physics, World Scientific, 2022: pp. 387–436.

[59] V. Bapst, T. Keck, A. Grabska-Barwińska, C. Donner, E.D. Cubuk, S.S. Schoenholz, A. Obika, A.W.R. Nelson, T. Back, D. Hassabis, Unveiling the predictive power of static structure in glassy systems, Nat Phys. 16 (2020) 448–454.

[60] A. Sanchez-Gonzalez, J. Godwin, T. Pfaff, R. Ying, J. Leskovec, P. Battaglia, Learning to simulate complex physics with graph networks, in: International Conference on Machine Learning, PMLR, 2020: pp. 8459–8468.