Learning Material Elasticity Tensors Using Tensor Field Networks

Nima Leclerc, Abhinav Ramkumar

December 16, 2020

Abstract

We utilize tensor field networks to learn elasticity tensors from space group information for different materials in the Materials Project Database (MP). This is the *first attempt* at utilizing this network architecture to take in rank-4 tensor inputs. This project utilizes the Euclidean Neural Network (e3nn) library developed by Tess Smidt at the Lawrence Berkeley National Laboratory (LBNL) that can take in space groups in euclidean basis, convert them to irreducible basis or spherical harmonics, and perform point convolutions on space group particle positions to learn filters that preserve equivariant tensor properties. In this report, we will explain the features that make tensor field networks unique, how they relate to convolution neural networks (CNNs), and the potential benefits of these network architectures to problems in materials science and beyond. While we are able to overfit the network on individual points, we were unable to train the network on the entire dataset to predict elasticity tensor for different phases of SiO_2, indicating that this architecture might have issues with the heterogeneity of the dataset.

Keywords: Tensor Field Networks, Elasticity, Equivariance, and Point Convolutions

1 Introduction

Convolution neural networks (CNNs) afford *translational equivariance*, which is important in detecting features in any location of a given input. In the course, we studied how filters are designed to *convolve* on different local neighborhoods of an image and output a set of signals that correspond to a specific feature. The addition of a *maximum pooling layer* results in *translational invariance*, thus detecting a feature no matter where it is located in the input. In this report, we discuss the application of a family of networks that are equivariant to symmetries present in 3D Euclidean space, called **tensor field networks**.

One important application of tensor field networks is in predicting properties of crystalline materials that depend on local unit structures, called *space groups*. Such systems have *extensive* properties that depend on symmetry considerations such as resistivity and thermal conductivity, which are important for technological applications in photonics and computer chips. In conjunction, there are *intensive* properties that are *not* measurable, but are inferred from experiments. For example, elasticity is a intensive property of a material that relates the mechanical displacement of a crystal along different axes to the applied load, and is thus represented as a *geometric tensor*. Predicting these properties currently require computationally intensive approaches such as density functional theory (DFT) that involve evaluating the wave-functions of all atoms to determine macroscopic properties [1].

Tensor field networks are a good solution to such problems as it extends the notion of equivariance in CNNs to material systems in *euclidean space* to allow for rotational (eg. the ability to identify a feature in any 3D rotation and its orientation) and translational equivariance. These networks naturally encode *geometric tensors*, mathematical objects that transform under geometric transformations of rotation and translations. These networks differ from CNNs in three ways [2,3]:

1. These networks operate on point clouds using continuous convolutions. The network layers act on 3D coordinates of the points and features at those points.
2. The filters are constrained to be the product of learnable radial functions and a spherical harmonic.
3. The choice of filter requires that the structure of the network be compatible with the algebra of *geometric tensors*.

2 Related Work

Tensor field networks build upon existing work on Harmonic Networks [4], which achieve 2D rotation equivariance using discrete convolutions and filters composed of circular harmonics, and SchNet [5] that presents a rotation-invariant network with continuous convolutions for modelling quantum interactions. Such networks are emulated in tensor field networks, but suffer from one main limitation, namely that rotations in 3D do not commute in the same ways 2D rotations; that is for rotation matrices A and B, $AB \neq BA$. Other works have dealt with similar issues of invariance and equivariance under particular transformations. For example, group convolution networks (G-CNNs) [6] guarantee equivariance with respect to *finite* symmetry groups. Cohen et al. [7] uses spherical harmonics and Wigner D-matrices for spherical signals (2D surface data of spheres), but do not extend to 3D data. One of the main limitations in these work is their inability to be applied to points in space that define a unit cell of a crystal, which is achieved by tensor field networks through point convolutions.

3 Our Project

We demonstrate the application of tensor field networks in predicting elasticity tensors from space group data from the Materials Project (MP) Database. The MP Database is an open-source repository that contains information about important material properties, such as elasticity and band structures. These properties are generated from high-throughput DFT predictions and experimental measurements. We trained a regression model on 22 different phases of SiO_2 that predicts how changes to structural symmetry properties affect the macroscopic elasticity tensor. There are some of noteworthy challenges, namely (1) the lack of labelled data for elasticity tensors, (2) adapting the architecture to higher rank tensors, and (3) testing the network since there isn't a well-defined test case for this problem. While we were able to overfit the network to a single data point, it proved infeasible to train the network on the entire dataset. Despite this setback, the ability to overfit this network is a promising sign that with more work, this architecture could be scaled to higher ranked tensors, which would be a novel result. We plan to build on this course project to achieve this goal and publish our results.

4 Materials Science Background

4.1 Crystal Structures

A crystal structure is composed of a unit cell that periodically repeats (infinitely) in all directions by defining vectors $\vec{A}$, $\vec{B}$, and $\vec{C}$ that represent the lattice parameters of the unit cell [2,3]. In addition, there are three angles that define the unit cell, namely α, β, and γ. **Figure 1** illustrates these parameters:

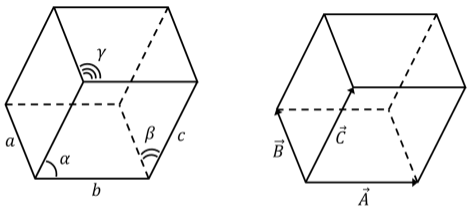

FIGURE 1.1: Unit cell given by side lengths and angles

FIGURE 1.2: Unit cell given by vectors

Figure 1: Unit Cell Diagram

Different combinations of these parameters result in *space group* symmetries. There are total of 230 space groups, which contain a combination of translation, rotations, reflections, and inversions that result in different types of material packing.

4.2 Elastic Tensor

The elastic (or elasticity) tensor is a rank-4 tensor that quantifies the response of a body to deformations, which appears in the generalized form of Hooke's law in continuum mechanics shown in Eqtn. 1:

$$\sigma_{ij} = C_{ijkn}\epsilon_{kn} \qquad i,j,k,n \in \{1,2,3\} \tag{1}$$

where σ is the **stress tensor**, ϵ is the **strain tensor**, C is the **elasticity tensor**, and $1,2,3 \equiv \{x,y,z\}$. The elastic tensor C has the following symmetry elements shown in Eqtn. 2:

$$\sigma_{ij} = \sigma_{ji} \quad \epsilon_{ij} = \epsilon_{ji} \quad C_{ijkn} = C_{jkni} = C_{nijk} \tag{2}$$

As a consequence of these symmetries, the elastic tensor has at most 21 independent components out of $3^4 = 81$ possible components. Thus, a common way to represent the reduced form of this tensor is **Voight notation**, and the reduced form is expressed in Eqtn. 3:

$$\tilde{\sigma}_i = \tilde{C}_{ij}\tilde{\epsilon}_j \qquad i,j \in \{1,\dots 6\} \tag{3}$$

where the mapping for indices is $11 \to 1$, $22 \to 2$, $33 \to 3$, $23 \to 4$, $13 \to 5$, and $12 \to 6$. Depending on the crystal system, there can be additional point symmetries and as a consequence, fewer than 21 independent components. For the purposes of designing the neural network, we consider the most general tensor possible with 21 components [2,3].

5 Theory

Group Representations in 3D
Let's denote a representation for a group G, D. Let $g, h \in G$.

$$D(g)D(h) = D(gh) \tag{4}$$

For some loss-function $\mathcal{L} : \mathcal{X} \to \mathcal{Y}$, it must be equivariant with respect to group G for $g \in G$ if,

$$\mathcal{L} \circ D^{\mathcal{X}}(g) = D^{\mathcal{Y}}(g) \circ \mathcal{L} \tag{5}$$

which is the composition of the network with the group representation. If the network is equivariant with respect to the transformations g and h, then it must be equivariant to the *composition* of the transformations [8,9]. These networks will act on individual points (or sets of points) and output vectors corresponding to each point. This mapping is given by Eqtn. 6,

$$\mathcal{L}(r_\alpha, x_\alpha) = (r_\alpha, y_\alpha) \tag{6}$$

with r as the input point, x as the input feature vector, and y as the output feature vector. Here $r \in R^3$, $x \in \mathcal{X}$, and $y \in \mathcal{Y}$. Importantly, these networks posses 3 types of equivariance: (1) permuatation, (2) translational, and (3) rotational. These different forms of equivariance are baked into the definitions of network layers, described below.

Tensor Field Layers
Point convolution: One type of layer in this network is the point convolution. At each layer, there's an input structure $V^{(l)}_{acm}$. This operation acts the same for each point a. At each layer, spherical harmonics $Y^{(l)}_m$ are used which are equivariant to $SO(3)$. Let $g \in SO(3)$. Then according to Eqtn. 7:

$$Y^{(l)}_m(\mathcal{R}(g)\hat{r}) = \sum_{m'} D^{(l)}mm'(g)Y^{(l)}_{m'}(\hat{r}) \tag{7}$$

To ensure that the filters are rotation-equivariant, we restrict them to have the form:

$$F^{(l_f,l_i)}_{cm} = R^{(l_f,l_i)}_c Y^{(l)}_{m'}(\hat{r}) \tag{8}$$

where $F_{cm}^{(l_f,l_i)}$ is the product of the learnable radial function, $R_c^{(l_f,l_i)}$, and the equivariant spherical harmonics $Y_{m'}^{(l)}(\hat{r})$. Hence, each filter and layer will carry l and m indices. Here, we can combine representations through a tensor product over the vector spaces of each representation:

$$D^{\mathcal{X}} \otimes D^{\mathcal{Y}} = D^{\mathcal{X} \otimes \mathcal{Y}} \tag{9}$$

Take two representations $u \in l_1$ and $v \in l_2$, then $u \otimes v \in l_1 \otimes l_2$. Explicitly, we write this tensor product as:

$$(u \otimes v)_m^{(l)} = \sum_{m_1=-l_1}^{l_1} \sum_{m_2=-l_2}^{l_2} C_{(l_1,m_1),(l_2,m_2)}^{(l,m)} u_{m_1}^{(l_1)} v_{m_1}^{(l_1)} \tag{10}$$

where $C_{(l_1,m_1),(l_2,m_2)}^{(l,m)}$ is the **Clebsch-Gordon** coefficient for the transformations of the representations between Cartesian and radial basis functions. It is important to note that only parameter being learned here is the the point-convolution filter $F_{cm}^{(l_f,l_i)}$, which only works on inputs given in terms of radial basis functions. The Clebsch-Gordon coefficients are parameters that are deterministic that map network outputs to Cartesian coordinates.

Layer Definition: Now that we've defined the filters and convolutional operation, we can define the network's layers. This is given by,

$$\mathcal{L}_{acm_o}(r_a V_{acm_i}) = \sum_{m_2,m_1} C_{(l_1,m_1),(l_2,m_2)}^{(l,m)} \sum_{b \in S} F_{cm_2}^{l_2,l_1} V_{bcm_1} \tag{11}$$

since the filter F is equivariant under $SO(3)$, then the entire is also equivariant under these operations. In a typical implementation of a tensor field network, one would typically use 3-4 point-convolution and gated-block nonlinearities to obtain a state-of-the-art performance. To date, this network architecture has **not** been tried on *anything higher* than rank-2 tensors, so getting a good training accuracy and especially test error for this problem would be a novel finding in the research community and *worthy* of publication. **Figure 2** shows an example architecture with all these components for predicting rank-1 tensors.

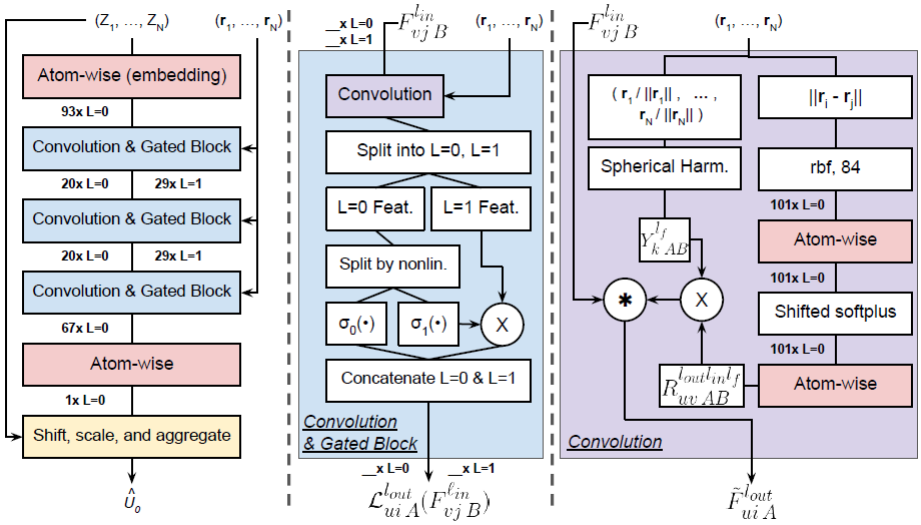

Figure 2: L1Net architecture (left), the convolution & gated block (center), and the convolution (right) [9].

Gated-Block: The gated-nonlinearities will act as scalar transformations in the l spaces. The right middle subfigure shows how different nonlinearities are applied to the outputs. In the L1Net architecture, scalars ($l = 0$) are handled normally, i.e. $\mathcal{L}(F^{l=0}) = \text{Softplus}(\tilde{F}^{l=0})$, which is denoted as $\sigma_0(.)$, while vectors ($l = 1$) features are multiplied by a scalar passed through an activation function. Specifically, $\mathcal{L}(F_u^{l=1}) = \text{Sigmoid}(\tilde{F}_{u+\mathcal{O}}^{l=0}) \tilde{F}_u^{l=1}$, which is denoted as $\sigma_1(.)$. This gated-block thus introduces nonlinearities while maintaining equivariance. The previous layers provide extra learned scalar features of multiplicity u_1 with index offset $\mathcal{O}$ to use this nonlinearity. In this project, we modify the architecture to extend the gated-block to apply nonlinearities on $l = \{0, 1, 2, 3, 4\}$ spaces [2,3,9].

6 Dataset

In this implementation, we trained our model on a data set containing elasticity tensors corresponding to 22 phases of SiO_2 (with composition conserved) existing at different temperatures and pressures. Each crystal structure corresponding to a single phase of SiO_2 possesses a unique symmetry that is reflected in its corresponding elasticity tensor. The elasticity tensors queried directly from the MP database were originally in Voigt (engineering) notation making it intractable to directly feed into the network. Hence, we first converted each tensor from Voigt to Cartesian notation to generate tensors each with 81 elements. The Cartesian elasticity tensors were then each converted to an irreducible form containing 21 non-zero elements, using a linear transformation. The small size of our data set is due to the computational cost of performing large quantum mechanical predictions of elasticity tensor and the limitations set by thermodynamics for SiO_2 to take on multiple phases over a specified temperature-composition space. Hence, we were only able to train our model on 22 points because of the computational and experimental costs predicting and measuring these elasticity tensors.

7 Approach

We first curated and post-processed our data set as described in Section 6. Following this, we constructed a 3-layer Gated Convolutional Parity Network with point convolutions. The point convolutional filters are represented as a tensor product of a radial and spherical component as represented in Eqtn. 8. Here, we chose to use a radial function for the radial component and then trained the network to learn the parameters belonging to the radial basis function, while setting $l = 4$ since we are training on fourth-rank tensor data and used a convolution normalization constant of 50. We trained the model with the Adam optimizer, using a learning rate of 0.01 and used an l_2 loss function. Due to the small size of the data set, we set the batch size to 1. Nonlinearities were applied following each layer using the gated non-linearity routine described in Section 5 and we extended previous implementations to account for the fourth-rank tensor nature of the data.

8 Experiments

We carried out the approach as described in Section 7 to train the tensor field network on the SiO_2 elasticity data set. Due to the novelty of this architecture and the data set being used to train the model, we first attempted to overfit the model on one element of our training set to ensure that the network can operate on fourth-rank tensors. We trained the model over 170 epochs to look for behavior of convergence in the training loss. We found that at 140 epochs, the loss was able to achieve a convergence on the order of $l(w_{t+1}) - l(w_t) = 10^{-6}$ and the training loss was able to drop from $l = 2,650$ in the first epoch to $l = 3.92$ in the convergence regime. This model took 40 minutes to to train on a single point, making it inefficient. This result suggests that the model was able to sufficiently overfit on the training data and is a demonstration of training a tensor field network on fourth-rank tensor data.

We then proceeded to train our model on the entire SiO_2 elasticity data set. Using a batch size of 1, we attempted to train the model on the data set over 100 epochs. However, we were unable to achieve convergence in the training loss here. We attribute this problem to the fact that our data set is heterogeneous in the symmetry of crystal structure and tensor. Additionally, the network performed poorly on our data set since the model was not exposed to multiple examples of crystals with the same symmetry. We also attempted to train the model using learning rates of 0.1 and 0.001, where it showed the same convergence behavior.

9 Acknowledgments

We would like to thank Tess Smidt for her support with getting us situated with the e3nn library, and Kostiantyn Lapchevskyi for providing us background to using this network on different material properties. We would also like to thank Prof. Pratik Chaudhari and the TA team for timely insights and feedback throughout the semester for our project.

References

[1] K. T. Schütt, F. Arbabzadah, S. Chmiela, K. R. Müller, and A. Tkatchenko. *Quantum-chemical insights from deep tensor neural networks.* 8:13890 EP, Jan 2017. URL http://dx.doi.org/10.1038/ncomms13890. Article.

[2] Thomas, N., Smidt, T., Kearnes, S. M., Yang, L., Li, L., Kohlhoff, K., Riley, P. (2018). *Tensor Field Networks: Rotation- and Translation-Equivariant Neural Networks for 3D Point Clouds.* CoRR, abs/1802.08219. http://arxiv.org/abs/1802.08219

[3] K. Lapchevski, *Predicting Properties of Crystals*; Master's Thesis, Ukranian Catholic University, Supervised by Tess Smidt

[4] D. E. Worrall, S. J. Garbin, D. Turmukhambetov, and G. J. Brostow. *Harmonic networks: Deep translation and rotation equivariance.* In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017.

[5] Schütt, K. T., Sauceda, H. E., Kindermans, P.-J., Tkatchenko, A., Müller, K.-R. (2018). *SchNet – A deep learning architecture for molecules and materials.* The Journal of Chemical Physics, 148(24), 241722. https://doi.org/10.1063/1.5019779

[6] T. S. Cohen and M. Welling. *Group equivariant convolutional networks.* In Proceedings of the International Conference on Machine Learning (ICML), 2016.

[7] T. Cohen, M. Geiger, J. Köhler, and M. Welling. Spherical CNNs. *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=Hkbd5xZRb.

[8] Smidt, T. E. (2020). *Euclidean Symmetry and Equivariance in Machine Learning.* Trends in Chemistry. https://doi.org/https://doi.org/10.1016/j.trechm.2020.10.006

[9] Miller, B., Geiger, M., Smidt, T., Noé, F. (2020). *Relevance of Rotationally Equivariant Convolutions for Predicting Molecular Properties.* ArXiv, abs/2008.08461.