



Equivariant graph neural networks for fast electron density estimation of molecules, liquids, and solids

Jørgensen, Peter Bjørn; Bhowmik, Arghya

Published in:
npj Computational Materials

Link to article, DOI:
[10.1038/s41524-022-00863-y](https://doi.org/10.1038/s41524-022-00863-y)

Publication date:
2022

Document Version
Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

Citation (APA):
Jørgensen, P. B., & Bhowmik, A. (2022). Equivariant graph neural networks for fast electron density estimation of molecules, liquids, and solids. *npj Computational Materials*, 8(1), Article 183. <https://doi.org/10.1038/s41524-022-00863-y>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

ARTICLE OPEN



Equivariant graph neural networks for fast electron density estimation of molecules, liquids, and solids

Peter Bjørn Jørgensen¹✉ and Arghya Bhowmik¹✉

Electron density $\rho(\vec{r})$ is the fundamental variable in the calculation of ground state energy with density functional theory (DFT). Beyond total energy, features and changes in $\rho(\vec{r})$ distributions are often used to capture critical physicochemical phenomena in functional materials. We present a machine learning framework for the prediction of $\rho(\vec{r})$. The model is based on equivariant graph neural networks and the electron density is predicted at special query point vertices that are part of the message-passing graph, but only receive messages. The model is tested across multiple datasets of molecules (QM9), liquid ethylene carbonate electrolyte (EC) and Li_xNi_yMn_zCo(1-y-z)O₂ lithium ion battery cathodes (NMC). For QM9 molecules, the accuracy of the proposed model exceeds typical variability in $\rho(\vec{r})$ obtained from DFT done with different exchange-correlation functionals. The accuracy on all three datasets is beyond state of the art and the computation time is orders of magnitude faster than DFT.

npj Computational Materials (2022)8:183; <https://doi.org/10.1038/s41524-022-00863-y>

INTRODUCTION

Simulations are as critical as experiments now in materials discovery. At the atomic scale, quantum mechanics based simulations are frequently used in the computational search of novel functional materials and molecules¹. Within the well-known cost-accuracy trade-off associated with such methods, Kohn-Sham density functional theory (DFT) is the most widely used method due to the right balance between computational cost and accuracy. The electron density $\rho(\vec{r})$ is one of the fundamental variables in the state of the art iterative scheme of DFT. The electron density uniquely determines the ground state properties of a system². DFT is an $O(n^3)$ complexity method and thus is limited to a few hundred atoms in system size that can be simulated. System size limits prohibit us from fully exploiting DFT for simulating critical technologically and scientifically important systems. For example, one would need large simulation cells for portraying engineering ceramics (with many types of atoms in small fractions) or mixed liquid electrolytes (with many component molecules and additives). Not just system size, in many materials design problems, the enormity of phase space to be explored can also be a bottleneck in using DFT. Total energy is the most commonly used output from DFT simulations. Significant recent developments towards high accuracy machine learning potentials for molecules and condensed matter phases have been able to provide QM accuracy total energy at a much lower computational cost^{3–5}. However, for functional materials, electronic structure is important as well. Electronic density distribution and its modulation due to structural and chemical modifications are descriptors for many chemical properties. For example, Bader charge analysis is frequently used to understand redox processes and related phenomena in intercalation battery cathodes^{6–9}. Charge densities are critical for solar cell materials properties as well^{10,11}. Similarly, charge density redistribution is often used to understand trends in catalytic activity^{12–15}. In liquid electrolytes, the analysis of gradients and other features in charge density helps us understand intermolecular interactions^{16–19}. Charge density also gives us access to functional properties (through surrogate models) that are costly to calculate directly. For

example, charge density maps can give us direct access to optimum intercalation sites²⁰ as well as ion migration pathway and barriers²¹. We exemplify a few of the many possible ways fast machine learning prediction of charge density can help us find new better functional materials. We can explore larger phase space as well as evaluate functional properties of materials that require large simulation boxes.

Electron density is inherently more information rich than total energy and therefore learning from the density could lead to machine learning models that generalize better from small datasets. For example,²² and²³ both found that learning the electron density and then predicting the total energy gives better accuracy when extrapolating from small to large systems in comparison to direct energy prediction. In the last few years, a number of articles have been published on electron density prediction. Pioneering works by^{24,25} use a basis representation for the density and predict the basis function coefficients using kernel ridge regression. The model's efficacy was demonstrated on molecular dynamics trajectories of small molecules, but by construction the model is not transferable to new molecules. A transferable model based on symmetry-adapted Gaussian process regression²⁶ (SA-GPR) was introduced later^{27,28}. The transferability is achieved by decomposing the density into atom-centered contributions and the local environment around each atom is mapped to a set of basis coefficients using the SA-GPR framework. One of the downsides of kernel regression is that the computational complexity of the model grows cubically with the number of training examples and in most practical problems we will need thousands of training examples to cover the system of interest.

Deep neural network models are highly flexible and are generally well suited for absorbing large datasets. A 3D convolutional neural network has been²⁹ trained with thousands of small molecules. However, by using a voxel based 3D U-Net³⁰ architecture the model is dependent on the image resolution and is not equivariant to rotations. Equivariance to rotation has been achieved in different ways. The aforementioned SA-GPR model^{27,28} has symmetry built into its kernel function.

¹Department of Energy Conversion and Storage, Technical University of Denmark, Anker Engelundsvej 1, Kgs. Lyngby 2800, Denmark. ✉email: pbjo@dtu.dk; arbh@dtu.dk

For deep learning models, equivariance has been achieved by constructing a fingerprint for every point in space that is invariant to rotations around the point for which the fingerprint has been created, but not to rotations around other points^{31–34}.

The models are local, which means that a cutoff distance defines the range for which the atoms no longer influence the electron density. Message-passing neural networks^{35,36} provide a mechanism for propagating atomic interactions over longer distances in a computationally efficient manner, by computing local messages that represents an atom and its environment and then propagate this information via the edges of a graph representation of the system. Message-passing neural networks are also being applied to the electron density prediction problem. In the work by³⁷ a message-passing network is used as part of the algorithm, but a new graph, representing the local neighborhood, is created for every point in space, which makes the method computationally inefficient and the model was therefore only trained on a relatively small number of points. A more efficient approach was later presented by^{38,39} in which atom-centered density contributions are predicted with a message passing neural network. This approach allows the energy of the system to be calculated analytically, but the accuracy of the density predictions are inferior to our previously published framework, DeepDFT, which predicts the densities directly point by point⁴⁰. In the previous message passing solutions^{37–40} an invariant representation is used. However, in this first generation of message passing algorithms, the models are unable to resolve angular information through the message passing⁴¹. Recent developments in equivariant message passing neural networks^{41–46} make it possible to propagate directional information through the message passing steps from which angular information can be extracted. Common for the equivariant message passing models is that the hidden states of the graph nodes are now representing directional vectors (in three dimensions) that rotate with the rotation of the molecule.

In this work, we present the equivariant DeepDFT model, which is a machine learning model for predicting the electron density $\rho(\vec{r})$. The model is based on equivariant message passing on a graph and uses special probe nodes inserted into the graph, for which the density is computed. In contrast to the OrbNet-Equi model⁴⁶, which uses features calculated by the GFN-xTB semiempirical electronic structure method, our method is purely data driven in the sense that the only inputs required to make a prediction are the atomic numbers and the coordinates of the atoms (including the unit cell parameters for periodic structures). We also do not induce any bias in terms of using a predefined basis set for the density, i.e., the density is purely learned from data examples.

Previous models have shown the ability to predict electron density on a number of different systems, including molecular dynamics of a single molecule or slab^{24,32}, different hydrocarbon molecules^{27,33}, large datasets of small organic molecules^{29,38,46}, carbon nanotubes³⁴, crystalline polymers and zeolites³⁷ and peptides^{28,39}. To showcase the universal applicability, we benchmark the equivariant DeepDFT model on three diverse datasets, (a) the QM9 dataset^{47,48} often used for benchmarking molecular machine learning models, which is a large dataset of 134k small organic molecules (b) a dataset of mixed transition metal layered oxide lithium ion battery cathode materials and (c) a dataset consisting of a molecular dynamics trajectory with ethylene carbonate molecules—a commonly used organic electrolyte.

RESULTS

Invariant and equivariant message passing models

We have developed and trained two models for charge density prediction with different architectures, but both work on the principles of message passing on molecular graphs³⁵. In this

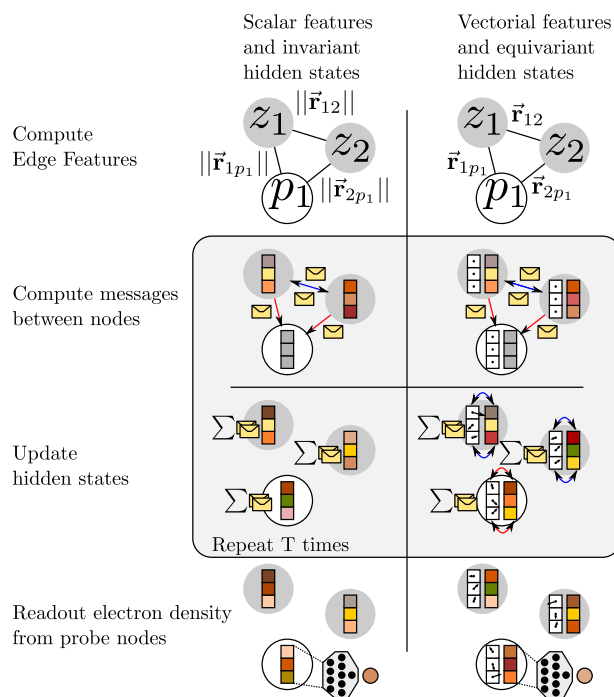


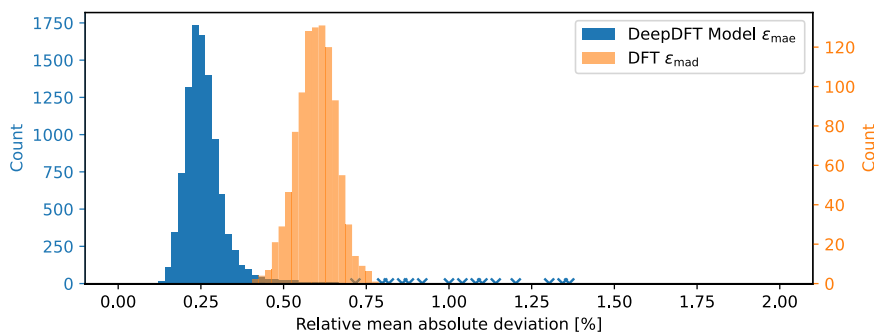
Fig. 1 Conceptual overview of the two message passing architectures used in DeepDFT. The left column illustrates the invariant DeepDFT model and the right column illustrates the equivariant DeepDFT model.

section we will be referring to both of the models—the invariant DeepDFT model and the equivariant DeepDFT model. A conceptual overview of the equivariant and invariant DeepDFT models are shown in Fig. 1.

The models use a 3D-embedded graph representation of the molecule or crystal structure. The graph has a vertex for each atom in the molecule or for each atom in the crystal structure unit cell. Edges are defined by a constant cutoff distance, i.e. we draw an edge between vertices if the distance between them is less than a certain cutoff distance (chosen here to be 4 Å). The edges may cross the periodic boundary as in quotient graphs^{49,50}. Special probe vertices, that only accept incoming edges, are placed at each query point where electron density prediction is to be made. Each vertex has a hidden state, which represents the atom or probe and its environment and the state is initialized based on the atom type or is zero for the probe vertices. The vertices interact by receiving messages from other vertices via the incoming edges. After each exchange of messages, the vertices update their hidden state based on the sum of incoming messages. Artificial neural networks (ANN) are used to model - (a) how the content of the messages depend on sending/receiving vertices and (b) how the hidden state updates depend on the message sum. ANNs representing these interactions can be trained using data examples. After a number of interaction steps T another neural network is used to map the hidden state of each probe vertex to the predicted density at that point in space. In the invariant version of the DeepDFT model, the edge feature is the distance between the vertices, while in the equivariant version, the distance as well as the direction of the edges are used as features. The directional features affect the hidden state of the vertices. To maintain the directionality of the hidden states, the equivariant model contains two sets of hidden states, an array of vector valued features for each vertex in addition to the array of scalar valued features for each vertex. The equivariance of the hidden states is preserved by restricting the allowed operations to scaling, inner products with other equivariant features and the addition of

Table 1. Datasets and prediction errors for invariant and equivariant DeepDFT model. We also compare with OrbNet-Equi and the superposition of atomic densities as implemented in VASP.

Dataset	Dataset Splits			Test Set Error ($\epsilon_{\text{mae}}\%$)			
	Train	Val.	Test	invDeepDFT	eqDeepDFT	OrbNet-Equi ⁴⁶	Init VASP
QM9	123835	50	10000	0.36	0.27	0.21	15
LIB Cathode	1450	50	500	0.09	0.06		7.1
Ethylene Carb.	7330	50	4000	0.53	0.18		13

**Fig. 2** Histogram of test errors on QM9 in comparison with variations in DFT computed density using different exchange correlations functional. The markers show bins with a count of one.

linear combinations of other equivariant features. The details of the invariant DeepDFT model are described in our previous NeurIPS workshop paper⁴⁰ and more details of the equivariant model can be found in the methods section.

Dataset and setup

To assess the models, we use three diverse datasets. The first is the QM9 dataset^{47,48} (134k small molecules with up to nine heavy atoms (CNOF)) that is widely used for benchmarking machine learning models for molecular property prediction. Additionally, we also train and test with electron density data from crystalline and liquid state materials. 1: A class of industrially important mixed transition metal layered oxide lithium ion battery (LIB) cathode materials. Configurations are generated through random crystal site occupation of transition metal ions (Ni/Mn/Co) and lithium/vacancy to represent varieties of chemistry and lithiation states. 2: Liquid ethylene carbonate—the most used LIB electrolyte. 12000 disordered configurations are generated through high temperature (3000 K) accelerated molecular identity preserving molecular dynamics. In all three cases, the electron densities are obtained using the VASP code⁵¹. See the methods section for more details on the computational setup.

Prediction accuracy

In this numerical experiment we assess the average prediction accuracy of the model using the three aforementioned datasets. We split the datasets into training, validation (for model selection) and test set with uniform random splitting. The sizes of the splits are shown in Table 1. To evaluate the model's accuracy in predicting charge density for a given atomic structure, we integrate the mean absolute error (MAE) over the whole simulation box normalized by the total number of electrons Equation (1) following recently published work^{27,28}. To obtain the metric for a given test VASP electron density, the model is probed at every point corresponding to the positions of the electron density grid saved by VASP, and the integrals are evaluated numerically as a sum. The DFT calculated density is used as ground truth $\rho(\vec{r})$ and the trained models output the predicted

density $\hat{\rho}(\vec{r})$.

$$\epsilon_{\text{mae}} = \frac{\int_{\vec{r} \in V} |\rho(\vec{r}) - \hat{\rho}(\vec{r})|}{\int_{\vec{r} \in V} |\rho(\vec{r})|} \quad (1)$$

The MAE of the models are shown in Table 1. As a baseline model we use the superposition as of atomic densities, which is used by VASP to initialize the electron densities.

The equivariant DeepDFT achieves notably lower prediction error than the invariant model, especially for the ethylene carbonate dataset. Both methods produce roughly two orders of magnitude lower error than the superposition of atomic densities baseline. The QM9 accuracy is below that of OrbNet-Equi⁴⁶, which uses additional input information, i.e. the results of GFN1-xTB semiempirical electronic structure method are used as input features. Thus it can not be considered as a pure data driven model. As another reference, we also look at the variation in electron density between DFT simulations performed using different exchange-correlation functionals in VASP. This can be seen as an appraisal of variability in DFT derived charge density. We randomly select 1000 examples of the QM9 test set and recalculate them with eight different XC functionals in VASP (BEEF, Perdew-Burke-Ernzerhof, Perdew-Wang 91, Ceperley-Alder, Perdew-Zunger, revised Perdew-Burke-Ernzerhof, revPBE, PBEsol)⁵²⁻⁵⁹. For every grid point, we select the median across the eight calculated densities as the reference point and compute the mean absolute deviation around this point as shown in Equation (2).

$$\epsilon_{\text{mad}} = \frac{\int_{\vec{r} \in V} \frac{1}{K} \sum_{k=1}^K |\rho_k(\vec{r}) - \rho_{\text{median}}(\vec{r})|}{\int_{\vec{r} \in V} |\rho_1(\vec{r})|} \quad (2)$$

For every molecule, the deviation is numerically integrated over the simulated volume. This serves as an estimate of inherent variations in DFT derived density and is analogous to the error measure Equation (1) used for the machine learning models. The DFT variation of 1000 molecules is shown along with the prediction test set (10000 molecules) errors in Fig. 2. The average DFT variation across molecules is 0.60 %, which is generally higher than the DeepDFT error.

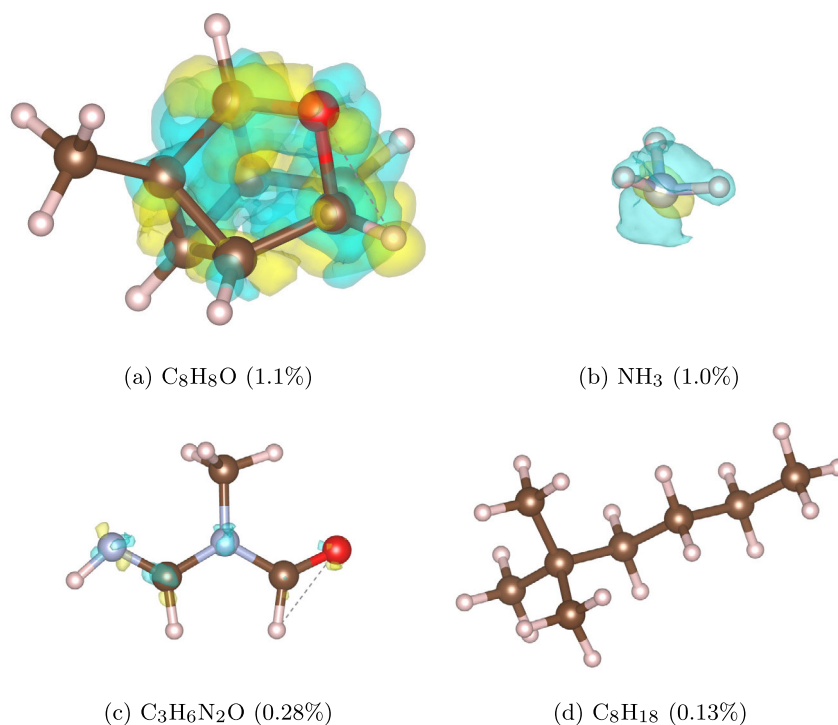


Fig. 3 Prediction error isosurfaces ± 0.001 e Bohr⁻³ for four example QM9 molecules from the test set with hydrogen (white), carbon (brown), oxygen (red) and nitrogen (blue). The error percentages in parenthesis denote the normalized mean absolute error ϵ_{mae} for each molecule. The chosen examples are two very high error molecules (a) and (b), an example with average error (c) and the molecule with the lowest error of all in the test set (d).

However, notice that the distribution of errors from the DeepDFT model has a much longer tail than DFT variation. There is even a single test point with 11 % error (not visible on the histogram figure). This is not the case for the two other datasets and is caused by the large chemical and structural variations within the QM9 dataset. Modeling the uncertainty and detecting outliers is therefore important for future models. Error isosurfaces for four example molecules are shown in Fig. 3. We are showing two very high error examples (Fig. 3a, b), an average error datapoint (Fig. 3c) and the lowest error example (Fig. 3d) in the test set. The highest error example (Fig. 3a) is a clear outlier and has very unnatural bond angles. The ammonia example (Fig. 3b) is an isolated group and is therefore not well represented in the training data. The average and lowest error points (Fig. 3c, d respectively) are well represented in the dataset and especially the hydrocarbon (Fig. 3d) is easily learned. Previous work focusing only on hydrocarbons could create models reaching an accuracy of 1.26 % using <1000 DFT densities³³. Our models are more accurate for hydrocarbons while generalizing to other molecules as well.

To understand the accuracy of the model from an energy perspective, we take all the charge densities of the test sets and replace the density values with those predicted by the equivariant DeepDFT model (while keeping the PAW augmentation charges fixed), and run a single-point non-self-consistent energy calculation using VASP. The obtained energies are compared with the self-consistent energies and the energy errors are shown in Fig. 4. The distribution of energy errors is heavily distributed around 0 with a long narrow tail, so we show the distribution in a log-log histogram. For the QM9 dataset (Fig. 4a) there are three molecules out of 10,000 with absolute error above 1×10^{-2} V/atom, while the mean absolute error for the remaining molecules is 8.5×10^{-5} eV/atom. For the two other datasets (Fig. 4b, c) we observe less extreme outliers, but the tails are still present. The MAEs are

4.2×10^{-4} eV/atom and 1.2×10^{-4} eV/atom for the LIB cathode and ethylene carbonate datasets, respectively.

Learning curve

In the sections above, we have looked at the average test errors for specific training and test set sizes. To better understand the effectiveness of the learning method and the data efficiency of the models, it can be very useful to look at learning curves⁶⁰, i.e. test error as a function of the training set size plotted on a log-log scale. The validation and test sets are the same as above, but we randomly sample a subset of the training data to reduce the training set size. The result of this numerical experiment is shown in Fig. 5. Ideally the learning curves should follow a straight line in the log-log plot and be as steep as possible⁶⁰. Initially all the learning curves are steep, but they flatten out with an increasing number of training examples.

The flattening of a learning curve is usually caused by either noise in the data, by a non-unique input representation or by lack of flexibility in the model.

If the saturation was caused by noise we would expect both models to converge to the same error. In contrast, for all three datasets, the equivariant models outperform the invariant models. The input information presented to the equivariant models is the relative positions of atoms and their atomic numbers, which is (given a large enough cutoff) enough to distinguish between all the inputs. This indicates that the model lacks the flexibility to capture all the details and inter-dependencies modeled by density functional theory and this is not surprising, given the simpler architecture of the deep learning models and the approximate nature of the learned functions. The training curves show (Supplementary Figs. 1, 2) that with 15000 training examples or more the model is not able to significantly overfit the QM9 dataset, but in all other cases the root mean squared error (RMSE)

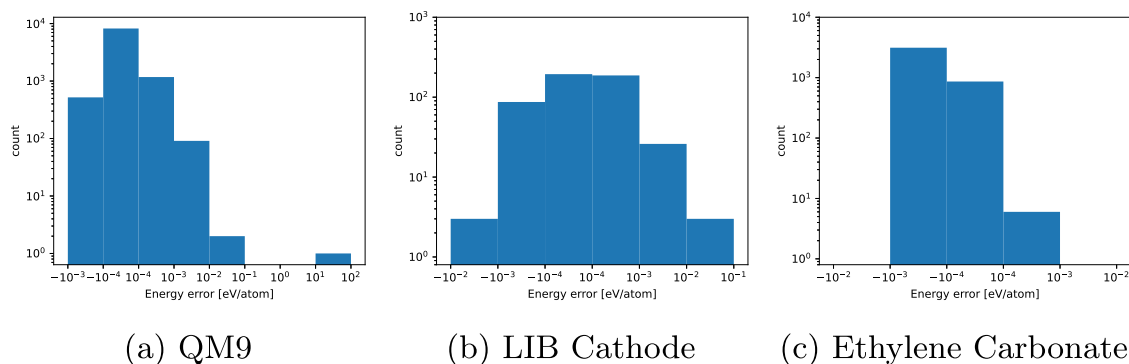


Fig. 4 Distribution of energy errors obtained from running non-self-consistent single point energy calculations with VASP using the predicted DeepDFT charge densities. The distribution of normalized (by number of atoms) energy errors are shown for three different datasets QM9 (a), LIB Cathode (b) and Ethylene Carbonate (c). The histograms use logarithmic bins and count scale to clearly show both the heavy concentration around 0 eV/atom and the long, narrow tail of errors.

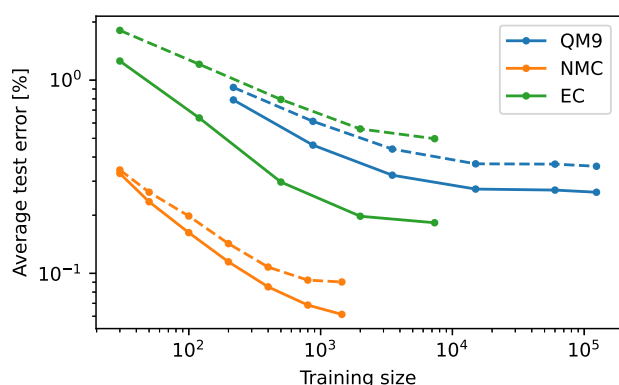


Fig. 5 Learning curves for invariant (dashed lines) and equivariant (solid lines) models for three different datasets. The plot shows the average normalized test errors (Equation 1) for the QM9 (blue) LIB Cathode (orange) and Ethylene Carbonate (green) datasets as functions of the number of training examples, which are randomly sampled subsets of the full training sets.

is lower on the training set than on the validation set. During development we have briefly tried to double the number of interaction layers in the equivariant model and increased the cutoff radius to 5 Å, but we did not see an improvement in the accuracy of the model. However, it might be possible to improve the model with a more systematic hyperparameter search or by introducing higher order equivariant internal representations as used in recently developed interatomic potentials, for example NequIP⁶¹.

Improved models are expected to provide higher accuracy and our analysis encourages further research and development of even more expressive deep learning architectures.

Among the three datasets, the ethylene carbonate dataset shows the largest difference in error between the two models. The added representation of directionality in the equivariant model is better at modeling highly polar molecules and the intermolecular interaction between them, which is further discussed in the following section.

Intermolecular interactions

To visualize the difference in prediction accuracy between the two models trained on the liquid ethylene carbonate structures, in Fig. 6 we show an error isosurface for one of the test set examples.

We see that most of the error is located around the oxygen atoms (red atoms). To investigate this quantitatively we partition

the electron density into volumes around each atom according to Bader partitioning⁶². For the 4000 test examples this leads to 29%, 16%, 55% of the total volume and 12%, 20%, 68% of the total electron charge to be assigned to H, C, O respectively. To understand how different elements contribute to the overall error, we calculate ϵ_{mae} following Equation (1); but for each atom type we only integrate over the Bader volumes associated with atoms of that type. Instead of normalizing with respect to the target density for each atom type in Equation (1), we also normalize with respect to the total error and calculate the total error share. The error decomposition for the invariant and equivariant models are shown in Table 2.

As the error isosurface figure also showed, the majority of the error is assigned to the oxygen atoms (66% for the invariant model and 59% for the equivariant model), but most of the target electron density is also found within the oxygen Bader volumes (68%). We also notice that out of the three elements it is the oxygen volume that benefits the most from using the equivariant model.

Runtime and scalability

To demonstrate the scalability of the model we measure the runtime for calculating the electron density of systems of increasing sizes. We use a single cell of 12 atoms $\text{Li}_3\text{Co}_2\text{NiO}_6$ with periodic boundary conditions and repeat the unit cell to show how model run time and DFT run time scales with system size. The result of the scalability test is plotted in Fig. 7. As expected we observe a linear trend for DeepDFT and it is therefore much faster than DFT for large systems, because of the cubic complexity of DFT. However, DeepDFT is also an order of magnitude faster even for small systems. Notice that DeepDFT is only running on a single GPU core and can be optimized to utilize more GPU cores in parallel for actual deployment towards high throughput tasks. The model is implemented in PyTorch for research purposes and even though it is orders of magnitude faster than density functional theory for large systems, it can still be made to run faster, e.g. utilizing several GPUs in parallel when making predictions or by simplifying the readout network, because this part of the network needs to be run for point in the simulation grid and thus are completely parallelizable with no communication overhead.

Charge transfer in NMC cathode

Charge transfer in intercalation cathodes is not only a key descriptor for the ion intercalation process²⁰ and related energetics, but can also be used as a tool towards capturing degradation processes like oxygen evolution⁶³. Thus understanding the electrochemical properties of cathode materials

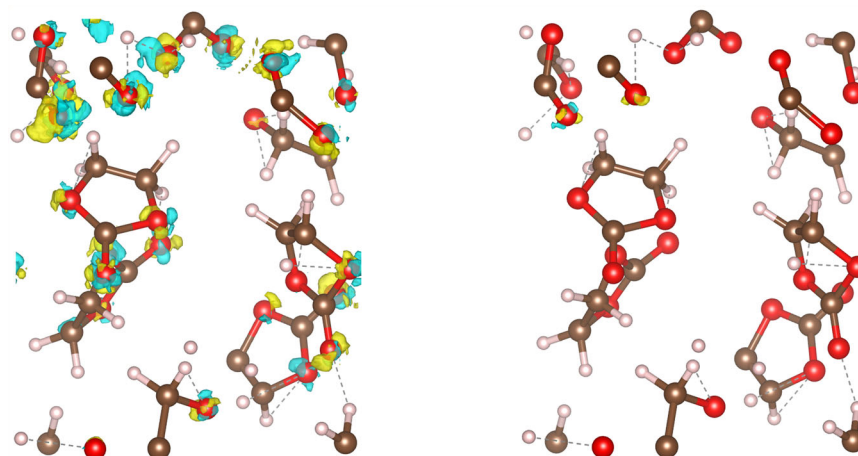


Fig. 6 Prediction error isosurfaces of $\pm 0.003 e \text{ Bohr}^{-3}$ with model based on invariant representation to the left and model based on equivariant representation to the right. The ethylene carbonate molecule contains hydrogen (white), carbon (brown) and oxygen (red).

Table 2. Prediction errors of liquid ethylene carbonate data test set decomposed into Bader volumes for each atom type.

Model	Element	ϵ_{mae}	Total Error share
Invariant	H	0.71%	17%
	C	0.41%	16%
	O	0.49%	66%
Equivariant	H	0.32%	21%
	O	0.16%	59%

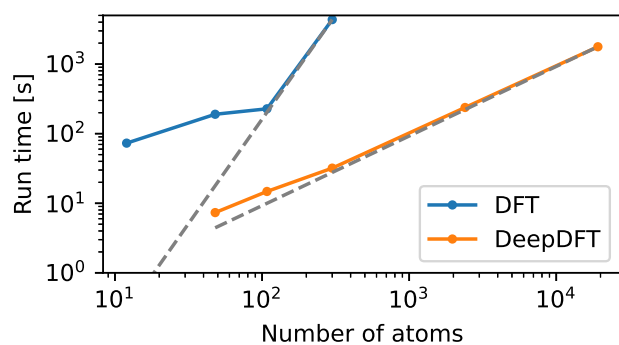


Fig. 7 Computation time for VASP running on 2*20 core Intel Skylake Xeon CPU vs DeepDFT running on a single RTX 3090 GPU. The dashed lines show the expected asymptotic behaviors ax^3 and ax respectively.

and computational screening for materials need access to charge density at various lithiated states. In this section we investigate the applicability of the model to accurately track charge transfer in NMC cathode materials. To test the accuracy of the trained DeepDFT model, we randomly sample 30 structures from the NMC dataset (that are not part of the training dataset), remove one Li atom and relax the structure with DFT. The DeepDFT model is then used to predict the electron density of the initial structure and of the one with one Li removed. If DeepDFT can capture the electron transfer redox process with good accuracy, we can use such models for screening optimal NMC compositions. We do note that this numerical experiment is only meaningful for assessing the ability of the model to track charge

transfer. In a real computational screening setting the structure relaxation would need to also be replaced by a machine learning model.

To assign charge to each atom we use the Bader charge of each atom calculated with the Bader program⁶⁴. The change in charge of each atom for all the 30 systems is shown as a histogram in Fig. 8. Notice that we have two clusters of charge differences in the dataset. The largest cluster is atoms that are not or very little affected by the removal of the Li atom. The smaller cluster is the one that is influenced by the Li removal to a larger degree. The prediction error is small for both clusters as shown in Fig. 8. This reinforces the capability of DeepDFT as a practical ML model that can be deployed for inexpensive large phase space exploration for high performance materials. Instead of looking at individual errors we can also compute the total error of each structure. We calculate the sum of absolute errors in the charge difference across the unit cell and average across the 30 structures of the test set:

$$\epsilon_{\text{total}} = \frac{1}{30} \sum_{i=1}^{30} \sum_{n=1}^{Q_i} \|\Delta q_n^i - \hat{\Delta q}_n^i\| \quad (3)$$

where i is the index for the 30 systems in the test set, Δq_n^i is the change in Bader charge of the n th atom of that system and Q_i is the number of atoms in the i th system. The average total error is 0.060 e, which means that on average 6.0 % of the electron charge is distributed incorrectly. However, a large proportion of that error is due to fluctuations in charges far away from the removed Li. If we only consider the near atoms in the inner sum of Equation (3), the error decreases to 1.0 %.

DISCUSSION

In this work we have presented the equivariant DeepDFT model which achieves at par or beyond the state of the art prediction accuracy on three widely different datasets including solid, liquid, and gas phases. Implementation of the equivariant framework (improvement on the invariant one⁴⁰) greatly helped in achieving high accuracy, especially in liquid state simulations as it helped learning better representations of inter- and intramolecular structure variations. Although the training is done on discrete grid points, the learnt function is fully differentiable and thus can be used for further mathematical derivations, such as derivative-based visualization of interactions using Interaction Region Indicator (IRI)⁶⁵ or Density Overlap Regions Indicator (DORI)⁶⁶. Additionally, the inference mechanism itself is parallelizable and

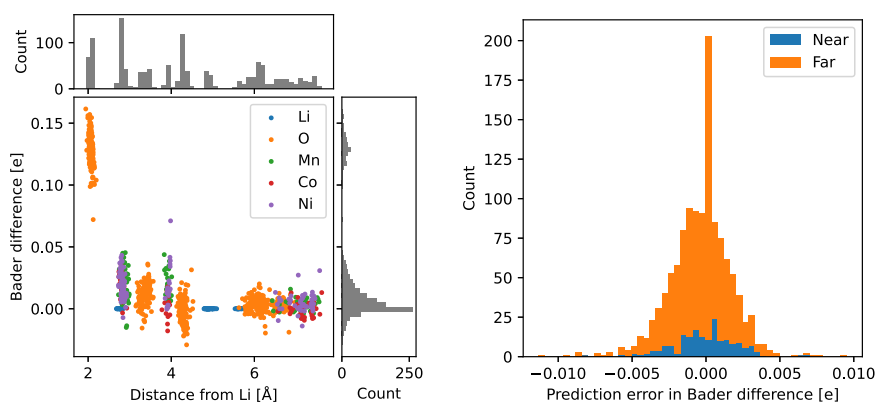


Fig. 8 Analysis of Bader charge difference of remaining atoms when removing a Li atom from NMC cathode. The scatter plot to the left shows the change in Bader charge for all atoms in the systems as a function of the distance from the Li being removed. The stacked histogram to the right shows the distribution of prediction errors of the machine learning model.

can be distributed to a larger number of GPUs, which makes electron density simulation of millions of atoms feasible.

Surrogate models for energy and forces have helped perform high accuracy molecular simulations at long length and time scale or conduct high throughput screening at an unprecedented speed in the last few years. For functional materials discovery (e.g. battery electrodes, catalysts etc.) combining energy models with DeepDFT will let us model and improve properties where electron transfer redox reactions and charge density are critical for better functionality. Because of the low runtime and linear scaling with system size DeepDFT will let us explore far larger materials phase space than it is currently possible with DFT derived charge densities.

We foresee high throughput screening and large scale simulations pertaining to redox reactions and charge transfer properties will be tackled with DeepDFT in the near future.

The error distribution for structures not well represented in the training data creates a long tail with high errors (as seen for a few unique molecules in QM9 dataset). For using DeepDFT in high throughput studies and for building trained models for new classes of materials in a data efficient manner with active learning, it would be necessary to also model the uncertainty of the predictions, e.g. by using an ensemble of models. In future it will also be beneficial to benchmark the model on other density derived properties such as the total energy, as utilizing density as the underlying variable can be more data efficient strategy than predicting the properties directly. The DeepDFT code is made available for the community and we are looking forward to seeing applications of the model in the simulation of materials and molecules as well as adaption into other models expanding on our codebase.

METHODS

Equivariant neural message-passing network

In this section we describe more formally and in more detail how neural message passing is used to model the electron density around atoms. The DeepDFT density model is framed in the neural message passing framework devised by³⁵. A simple example graph with four atoms and three query points is illustrated in Fig. 9. Each vertex has a hidden state that is updated in a number of interaction steps.

The previously introduced DeepDFT model⁴⁰ uses an atom-to-atom interaction architecture very similar to the SchNet model³⁶. However, in this type of model the hidden states of the vertices are scalar arrays, which contain no explicit information about directionality and are invariant to rotations. This for example, means that a change in angles can not be resolved through message passing steps⁴¹. This shortcoming has already been addressed by graph neural networks using spherical harmonics as the irreducible representation for the group of rotations in three dimensions^{44,45,61}. In these cases the hidden states become higher order

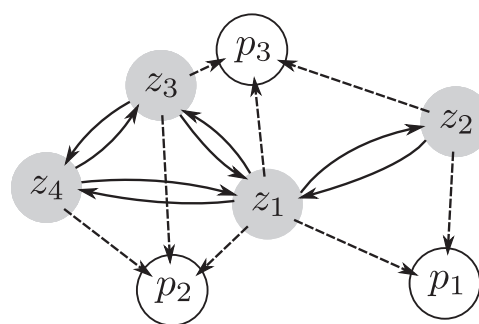


Fig. 9 Neural message-passing with probe nodes. Neural network computed messages are exchanged between atom vertices in several steps while the probe nodes only receive messages. The contents of the messages depend on the hidden state of the vertices and the states are updated after each message-passing step.

tensors and they rotate (equivariantly) with the molecule. A special case of equivariant neural networks, for which the equivariant states are Cartesian tensors, was recently introduced by⁴¹. In the so called polarizable atom interaction neural network (PaiNN) model the hidden vertex state contains a scalar array state as well as a vectorial state. Equivariance of the vectorial state is conserved by restricting the vectorial states to interact only via cross products, inner products and scaling. The new version of DeepDFT uses a variant of the PaiNN architecture as backend and the architectural details are given in this section. See the article introducing PaiNN⁴¹ for more explanation on the method itself.

The input to the model are the atomic numbers $\{z_1, \dots, z_N\} \in \mathbb{N}$, the xyz-coordinate of each atom $\{\vec{\mathbf{r}}_1, \dots, \vec{\mathbf{r}}_N\} \in \mathbb{R}^3$ and of each probe point $\{\vec{\mathbf{p}}_1, \dots, \vec{\mathbf{p}}_M\} \in \mathbb{R}^3$ and in the case of crystal structure the unit cell vectors are also required $C \in \mathbb{R}^{3 \times 3}$. We use the arrow superscript $\vec{\mathbf{x}}$ to emphasize vectors that are treated as geometric vectors, as opposed to arrays of scalars such as weight matrices.

From this information a directed graph is constructed with a vertex for each atom and another vertex type for each probe point. The edges of the graph are drawn between atoms when they are within the cutoff distance (4 Å) and incoming edges to the probe points are drawn when they are within the cutoff distance of an atom. The atom scalar nodes are initialized with a learned embedding for each atom type $\mathbf{s}_i^0 = \mathbf{a}_z \in \mathbb{R}^{F \times 1}$ and the vectorial state is initialized to zeros $\vec{\mathbf{v}}_i^0 = \mathbf{0} \in \mathbb{R}^{F \times 3}$. The update in scalar state is given by a sum over messages from neighboring atoms

$$\Delta \mathbf{s}_i^m = \sum_{j \in N(i)} \Phi_s(\mathbf{s}_j) \circ \mathbf{W}_s(\|\vec{\mathbf{r}}_{ij}\|) \cdot f_{\text{cut}}(\|\vec{\mathbf{r}}_{ij}\|) \quad (4)$$

where Φ is a 2-layer neural network with hidden layer and output layer size F , \circ is element-wise vector multiplication and $\mathbf{W}_s(\|\vec{\mathbf{r}}_{ij}\|)$ is a continuous filter function. The feature-wise filter function is implemented as F linear combinations of the distance expanded in sinc-like radial basis function⁶⁷

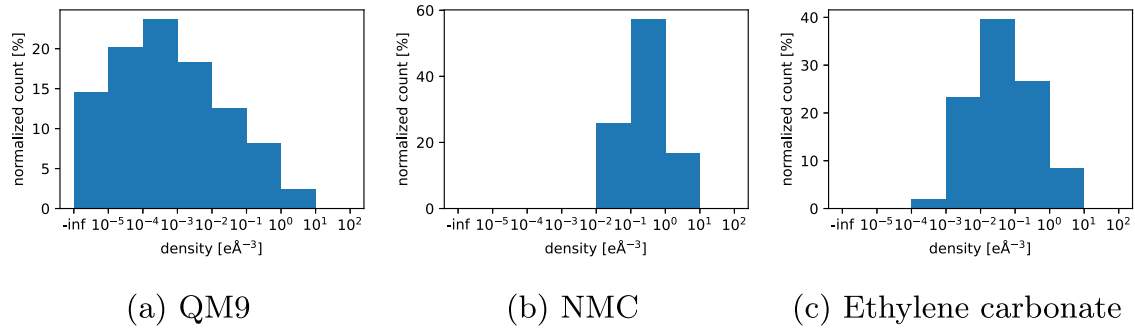


Fig. 10 Normalized histograms with logarithmic bins for the data examples of the three electron density datasets. QM9 (a) LIB Cathode (b) and Ethylene Carbonate (c). The histograms use logarithmic bins and show the number of data examples that fall into each bin as a percentage of the total number of examples.

$\sin\left(\frac{\pi}{r_{\text{cut}}}\|\vec{\mathbf{r}}_{ij}\|\right)/\|\vec{\mathbf{r}}_{ij}\|$ with $1 \leq n \leq 20$. The cutoff function $f_{\text{cut}}(\|\vec{\mathbf{r}}_{ij}\|) = 0.5(\cos(\pi\|\vec{\mathbf{r}}_{ij}\|/r_{\text{cut}}) + 1)$ for $\|\vec{\mathbf{r}}_{ij}\| < r_{\text{cut}}$ and 0 otherwise as proposed by⁶⁸. The cutoff function ensures a smooth transition when neighboring atoms enter the cutoff region. The update in vectorial state is given by:

$$\begin{aligned} \Delta \vec{\mathbf{v}}_i^m = & \sum_{j \in N(i)} \vec{\mathbf{v}}_j \circ \Phi_{\text{vv}}(\mathbf{s}_j) \circ \mathbf{W}_{\text{vv}}(\|\vec{\mathbf{r}}_{ij}\|) \cdot f_{\text{cut}}(\|\vec{\mathbf{r}}_{ij}\|) \\ & + \sum_{j \in N(i)} \Phi_{\text{vs}}(\mathbf{s}_j) \circ \mathbf{W}_{\text{vs}}(\|\vec{\mathbf{r}}_{ij}\|) \frac{\vec{\mathbf{r}}_{ij}}{\|\vec{\mathbf{r}}_{ij}\|} \cdot f_{\text{cut}}(\|\vec{\mathbf{r}}_{ij}\|) \end{aligned} \quad (5)$$

The first sum is the convolution of scaled equivariant features $\vec{\mathbf{v}}_j \circ \Phi_{\text{vv}}(\mathbf{s}_j)$ with an invariant (only distance-dependent) filter function. This is the only operation in the architecture where equivariant features are propagated through the network. This allows directional information obtained through previous message passing interactions to propagate through the network. The second term in Equation (5) is the only operation in the architecture where new directional information is added to the hidden state. Unit vectors corresponding to edges between vertices are scaled by $\Phi_{\text{vs}}(\mathbf{s}_j)$ and by the invariant filter function $\mathbf{W}_{\text{vs}}(\|\vec{\mathbf{r}}_{ij}\|)f_{\text{cut}}(\|\vec{\mathbf{r}}_{ij}\|)$ and added to the vectorial node state.

For more expressiveness another two update equations are introduced, which operates atomwise across the scalar and vectorial features. The updates for the scalar features are as follows

$$\Delta s_i^u = \mathbf{a}_{\text{ss}}(\mathbf{s}_i, \|\mathbf{V}\vec{\mathbf{v}}_i\|) + \mathbf{a}_{\text{sv}}(\mathbf{s}_i, \|\mathbf{V}\vec{\mathbf{v}}_i\|) \langle \mathbf{U}\vec{\mathbf{v}}_i, \mathbf{V}\vec{\mathbf{v}}_i \rangle \quad (6)$$

where (\cdot, \cdot) means concatenation of features and \mathbf{a}_{ss} and \mathbf{a}_{sv} are 2-layer neural networks. The vectorial features use the following update:

$$\Delta \vec{\mathbf{v}}_i^u = \mathbf{a}_{\text{vv}}(\mathbf{s}_i, \|\mathbf{V}\vec{\mathbf{v}}_i\|) \mathbf{U}\vec{\mathbf{v}}_i \quad (7)$$

which is a scaling of a linear combination of equivariant vector features. The message passing algorithm works by computing Equations (4) and (5) in parallel for all the node vectors and add the computed values to the current scalar and vectorial states respectively. Then the nodes are updated by running Equations (6) and (7) in parallel and update the nodes atomwise. The message passing and update equations are repeated in several layers with different neural network weights at each layer. In the PaiNN method and in this work we use three layers of message passing and update layers.

The hidden state of the special probe vertices are initialized with zeros (the scalar features are zero and the equivariant features are the zero vector). They use the same message passing and update equations as for the atom vertices above, but the neural network weights are not shared between the two. Furthermore, instead of using the residuals in Equations (4) and (5) directly, we introduce a gating network that determines which features of the message sum to include and which part of the features to ignore.

$$\mathbf{s}_i^{\text{new}} = \mathbf{G}_s(\Delta s_i^m) \circ \mathbf{s}_i + (1 - \mathbf{G}_s(\Delta s_i^m)) \circ \Delta s_i^m \quad (8)$$

$$\vec{\mathbf{v}}_i^{\text{new}} = \mathbf{G}_v(\Delta s_i^m) \circ \vec{\mathbf{v}}_i + (1 - \mathbf{G}_v(\Delta s_i^m)) \circ \Delta \vec{\mathbf{v}}_i^m \quad (9)$$

The gating neural networks are two-layer neural networks with SiLU activation function for the hidden units and a sigmoid output activation function. This allows the network to ignore parts of the incoming messages dependent on the total sum of messages. After the final interaction steps the final state of each probe vertex is mapped to a single scalar for each

probe state \mathbf{s}_i . As in the original PaiNN model we also use a two-layer neural network with SiLU activation function on the first layer and a linear activation function on the output layer. In principle the vectorial representation also enables prediction of vectorial properties at each probe vertex or higher order tensors constructed from a rank-1 tensor decomposition, as described in the original PaiNN article⁴¹.

Model parameters and training/validation setup

In all experiments we use a feature size $F=128$ and cutoff distance $r_{\text{cut}}=4 \text{ \AA}$. When using the invariant message passing model we set $T=6$ and use $T=3$ for the equivariant message passing model. The invariant model then has 2.1×10^6 parameters and the equivariant model 1.5×10^6 parameters. Because of the large memory requirement for the electron density (CHGCAR) files, we use a rotating pool of 20 atomic configurations during training. New configurations are continuously loaded from the full dataset on disk into the rotating pool. In each training step we sample two atomic structures and for each structure 1000 probe points are uniformly sampled from the VASP electron density grid. The cost function is the mean squared error of the probe points. The Adam optimizer is used with initial learning rate 10^{-4} and the learning rate is exponentially decayed during training, i.e. the learning rate is $0.96^{s \cdot 10^{-5}}$ at gradient step number s . The validation set is used for early stopping. The cost on the validation set is computed every 5000 gradient steps and the final model is the one with the lowest error on the validation set. To reduce the variance and improve computational efficiency the probes of the validation set are kept fixed during the training and we use 5000 probes for each configuration.

Datasets

For all datasets the electron densities are calculated with VASP which is a projector-augmented wave based implementation of DFT. Four hundred electronvolt is used for the wavefunction cutoff and a Gaussian smearing of 0.1 eV is used for the electronic states. The first Brillouin zone is sampled only at the zone center for QM9 molecules, with a $3 \times 3 \times 1$ Monkhorst-Pack k-point mesh for NMC data and $2 \times 2 \times 2$ for liquid ethylene carbonate simulations. Perdew-Burke-Ernzerhof (PBE) exchange-correlation functional is used for all datasets. However to understand the effect of exchange correlation-functional on the variability of DFT calculated charge density we did DFT simulations with eight different functionals for small subset of QM9 molecules. Model training was done with PBE densities. The electron densities are output on a grid and the grid spacing is $\sim 0.1 \text{ \AA}$ in all three datasets. The average number of grid points per configuration is 700 k, 400 k and 1 M for the QM9, NMC and ethylene carbonate datasets, respectively. The distribution of the data points for each dataset is shown in Fig. 10. The NMC dataset (Fig. 10b) is the most dense dataset while the QM9 (Fig. 10a) dataset has a lot of low electron density data examples. For datasets with even larger areas of “empty space” than the QM9 dataset it might be beneficial to remove some of the low density data points or in other ways weight the sampling of the training examples to avoid biasing the training towards predicting zeros.

QM9 dataset. The geometries for the QM9 dataset^{47,48} are obtained from Figshare repository as deposited by⁶⁹. The VASP package only supports periodic boundary conditions so we use simulation cells with vacuum around molecules such that there is a gap of at least 4 \AA or more. As pointed out by a reviewer, the gap is not large enough to avoid all

interactions between adjacent molecules. To quantify the interaction we have doubled the size of the unit cell of 1000 test set molecules and calculated the electron density at the same grid points as in the original dataset. The average ϵ_{mae} Equation (1) between the recalculated and original densities is 0.18%.

NMC dataset. This dataset contains charge densities for NMC $2 \times 2 \times 1$ supercell (12 transition metal atoms and 12 Li/vacancy site) with varying levels of Li content. For each structure we first randomly sample the number of Mn, Ni and Co atoms given that the total number of transition metal atoms is 12 and then randomly assign them to the transition metal positions of the lattice. Similarly the number of vacancies is uniformly sampled between 0 and 12 and vacancies are assigned to the Li site. The generated configurations are then relaxed in two steps: First we relax the atom positions with fixed cell parameters and then we allow both positions and cell parameters to relax. We keep only the electron density (CHGCAR) file after the last cell relaxation step. The atoms are relaxed until forces on each atom are <0.01 eV/Å.

Ethylene carbonate molecular dynamics trajectory. This dataset consists of charge densities of individual snapshots from a molecular dynamics trajectory. We insert 8 ethylene carbonate molecules in the simulation box. To quickly explore a large part of the configurational space we put Hookean constraints on the molecular bonds (to maintain molecular identity such that molecules are not torn apart at such high temperature) and run Langevin molecular dynamics with thermostat temperature of 3000 K. The simulation was run for 12380 steps of 0.5 fs and we discard the first 1000 steps to reach equilibration.

DATA AVAILABILITY

The datasets are openly accessible at DTU figshare^{70–72}.

CODE AVAILABILITY

A codebase for both models as well as pretrained PyTorch models are available on Github⁷³.

Received: 1 December 2021; Accepted: 1 August 2022;

Published online: 23 August 2022

REFERENCES

- Marzari, N., Ferretti, A. & Wolverton, C. Electronic-structure methods for materials design. *Nat. Mater.* **20**, 736–749 (2021).
- Payne, M. C., Teter, M. P., Allan, D. C., Arias, T. A. & Joannopoulos, J. D. Iterative minimization techniques for ab initio total-energy calculations: molecular dynamics and conjugate gradients. *Rev. Mod. Phys.* **64**, 1045 (1992).
- Unke, O. T. Machine learning force fields. *Chem. Rev.* **121**, 10142–10186 (2021).
- Deringer, V. L. et al. Gaussian process regression for materials and molecules. *Chem. Rev.* **121**, 10073–10141 (2021).
- Behler, J. Four generations of high-dimensional neural network potentials. *Chem. Rev.* **121**, 10037–10072 (2021).
- Kondrakov, A. O. Charge-transfer-induced lattice collapse in ni-rich ncm cathode materials during delithiation. *J. Phys. Chem. C* **121**, 24381–24388 (2017).
- Dixit, M., Markovsky, B., Schipper, F., Aurbach, D. & Major, D. T. Origin of structural degradation during cycling and low thermal stability of ni-rich layered transition metal-based electrode materials. *J. Phys. Chem. C* **121**, 22628–22636 (2017).
- Varanasi, A. K., Bhowmik, A., Sarkar, T., Waghmare, U. V. & Bharadwaj, M. D. Tuning electrochemical potential of licoo 2 with cation substitution: first-principles predictions and electronic origin. *Ionics* **20**, 315–321 (2014).
- Kuo, L.-Y., Guillon, O. & Kaghazchi, P. On the origin of non-monotonic variation of the lattice parameters of lini 1/3 co 1/3 mn 1/3 o 2 with lithiation/delithiation: a first-principles study. *J. Mater. Chem. A* **8**, 13832–13841 (2020).
- Chang, J. et al. Lead-free perovskite compounds csn1-xgexi3-ybry explored for superior visible-light absorption. *Phys. Chem. Chem. Phys.* **23**, 14449–14456 (2021).
- Barragan-Yani, D. & Albe, K. Atomic and electronic structure of perfect dislocations in the solar absorber materials cuinse 2 and cugase 2 studied by first-principles calculations. *Phys. Rev. B* **95**, 115203 (2017).
- Castellani, N. J., Branda, M. M., Neyman, K. M. & Illas, F. Density functional theory study of the adsorption of au atom on cerium oxide: effect of low-coordinated surface sites. *J. Phys. Chem. C* **113**, 4948–4954 (2009).
- Palmer, C. Methane pyrolysis with a molten cu–bi alloy catalyst. *ACS Catal.* **9**, 8337–8345 (2019).
- Zaffran, J. & Toroker, M. C. Metal–oxygen bond ionicity as an efficient descriptor for doped nioph photocatalytic activity. *ChemPhysChem* **17**, 1630–1636 (2016).
- Vasileff, A. Selectivity control for electrochemical co2 reduction by charge redistribution on the surface of copper alloys. *ACS Catal.* **9**, 9411–9417 (2019).
- Cao, B. et al. Reversibility of imido-based ionic liquids: a theoretical and experimental study. *RSC Adv.* **7**, 11259–11270 (2017).
- Mangiatordi, G. F., Hermet, J. & Adamo, C. Modeling proton transfer in imidazole-like dimers: a density functional theory study. *J. Phys. Chem. A* **115**, 2627–2634 (2011).
- del Olmo, L., Morera-Boado, C., López, R. & de la Vega, J. M. G. Electron density analysis of 1-butyl-3-methylimidazolium chloride ionic liquid. *J. Mol. Model.* **20**, 1–10 (2014).
- Armaković, S., Armaković, S. J., Vraneš, M., Tot, A. & Gadžurić, S. Dft study of 1-butyl-3-methylimidazolium salicylate: a third-generation ionic liquid. *J. Mol. Model.* **21**, 1–10 (2015).
- Shen, J.-X., Horton, M. & Persson, K. A. A charge-density-based general cation insertion algorithm for generating new li-ion cathode materials. *npj Comput. Mater.* **6**, 1–7 (2020).
- Kahle, L., Marcolongo, A. & Marzari, N. High-throughput computational screening for solid-state li-ion conductors. *Energy Environ. Sci.* **13**, 928–948 (2020).
- Tsubaki, M. & Mizoguchi, T. Quantum deep field: data-driven wave function, electron density generation, and atomization energy prediction and extrapolation with machine learning. *Phys. Rev. Lett.* **125**, 206401 (2020).
- Lewis, A. M., Grisafi, A., Ceriotti, M. & Rossi, M. Learning electron densities in the condensed phase. *J. Chem. Theory and Comput.* **17**, 7203–7214 (2021).
- Brockherde, F. et al. Bypassing the Kohn-Sham equations with machine learning. *Nat. Commun.* **8**, 872 (2017).
- Bogojeski, M. et al. Efficient prediction of 3D electron densities using machine learning. *arXiv* <https://doi.org/10.48550/arXiv.2205.05475> (2018)
- Grisafi, A., Wilkins, D., Csányi, G. & Ceriotti, M. Symmetry-adapted machine-learning for tensorial properties of atomistic systems. *Phys. Rev. Lett.* **120**, 036002 (2018).
- Grisafi, A. et al. Transferable machine-learning model of the electron density. *ACS Cent. Sci.* **5**, 57–64 (2019).
- Fabrizio, A., Grisafi, A., Meyer, B., Ceriotti, M. & Corninboeuf, C. Electron density learning of non-covalent systems. *Chem. Sci.* **10**, 9424–9432 (2019).
- Sinitskiy, A.V., Pande, V. S. Deep neural network computes electron densities and energies of a large set of organic molecules faster than density functional theory (DFT). *arXiv* <https://doi.org/10.48550/arXiv.1809.02723> (2018).
- Ronneberger, O., Fischer, P. & Brox, T. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI). Lecture Notes in Computer Science* Vol. 9351 (eds. Navab, N. et al.) 234–241 (Springer, 2015).
- Zepeda-Núñez, L. et al. Deep density: Circumventing the kohn-sham equations via symmetry preserving neural networks. *J. Comput. Phys.* **443**, 110523 (2021).
- Chandrasekaran, A. Solving the electronic structure problem with machine learning. *npj Comput. Mater.* **5**, 1–7 (2019).
- Kamal, D., Chandrasekaran, A., Batra, R. & Ramprasad, R. A charge density prediction model for hydrocarbons using deep neural networks. *Mach. Learn. Sci. Technol.* **1**, 025003 (2020).
- Alred, J. M., Bets, K. V., Xie, Y. & Yakobson, B. I. Machine learning electron density in sulfur crosslinked carbon nanotubes. *Compos. Sci. Technol.* **166**, 3–9 (2018).
- Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., Dahl, G. E. Neural message passing for quantum chemistry. In *International Conference on Machine Learning*. 1263–1272 (ACM, 2017).
- Schütt, K. T., Sauceda, H. E., Kindermans, P.-J., Tkatchenko, A. & Müller, K.-R. SchNet—A deep learning architecture for molecules and materials. *J. Chem. Phys.* **148**, 241722 (2018).
- Gong, S. Predicting charge density distribution of materials using a local-environment-based graph convolutional network. *Phys. Rev. B* **100**, 184103 (2019).
- Cuevas-Zuñiga, B. & Pacios, L. F. Analytical model of electron density and its machine learning inference. *J. Chem. Inf. Model.* **60**, 3831–3842 (2020).
- Cuevas-Zuñiga, B. & Pacios, L. F. Machine learning of analytical electron density in large molecules through message-passing. *J. Chem. Inf. Model.* **61**, 2658–2666 (2021).
- Jørgensen, P. B., Bhowmik, A. DeepDFT: Neural Message Passing Network for Accurate Charge Density Prediction. Preprint at <https://arxiv.org/abs/2011.03346> (2020).
- Schütt, K., Unke, O. & Gastegger, M. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In *Proceedings of the 38th*

- International Conference on Machine Learning* Vol. 139 (eds. Meila, M. & Zhang, T.) 9377–9388 (PMLR, 2021).
42. Cohen, T. S., Cohen, T. S. & Ni, U. *33rd International Conference on Machine Learning* (PMLR, 2016).
 43. Kondor, R., Lin, Z. & Trivedi, S. *Advances in Neural Information Processing Systems*, Vol. 31 (MIT Press, 2018).
 44. Thomas, N. et al. Tensor field networks: Rotation- and translation-equivariant neural networks for 3D point clouds. Preprint at <https://arxiv.org/abs/1802.08219> (2018).
 45. Anderson, B., Hy, T. S. & Kondor, R. *Advances in Neural Information Processing Systems* Vol. 32 (NIPS, 2019).
 46. Qiao, Z. et al. Informing geometric deep learning with electronic interactions to accelerate quantum chemistry. *Proc. Natl. Acad. Sci.* **119**, e2205221119 (2022).
 47. Ruddigkeit, L., van Deursen, R., Blum, L. C. & Reymond, J.-L. Enumeration of 166 billion organic small molecules in the chemical universe database GDB-17. *J. Chem. Inf. Model.* **52**, 2864–2875 (2012).
 48. Ramakrishnan, R., Dral, P. O., Rupp, M. & von Lilienfeld, O. A. Quantum chemistry structures and properties of 134 kilo molecules. *Sci. Data* **1**, 140022 (2014).
 49. Chung, S. J., Hahn, T. & Klee, W. E. Nomenclature and generation of three-periodic nets: The vector method. *Acta Crystallogr. A* **40**, 42–50 (1984).
 50. Klee, W. E. Crystallographic nets and their quotient graphs. *Cryst. Res. Technol.* **39**, 959–968 (2004).
 51. Hafner, J. Ab-initio simulations of materials using vasp: density-functional theory and beyond. *J. Comput. Chem.* **29**, 2044–2078 (2008).
 52. Wellendorff, J. Density functionals for surface science: exchange-correlation model development with bayesian error estimation. *Phys. Rev. B* **85**, 235149 (2012).
 53. Perdew, J. P., Burke, K. & Ernzerhof, M. Generalized gradient approximation made simple. *Phys. Rev. Lett.* **77**, 3865 (1996).
 54. Perdew, J. P. & Wang, Y. Accurate and simple analytic representation of the electron-gas correlation energy. *Phys. Rev. B* **45**, 13244 (1992).
 55. Ceperley, D. M. & Alder, B. J. Ground state of the electron gas by a stochastic method. *Phys. Rev. Lett.* **45**, 566 (1980).
 56. Perdew, J. P. & Zunger, A. Self-interaction correction to density-functional approximations for many-electron systems. *Phys. Rev. B* **23**, 5048 (1981).
 57. Hammer, B., Hansen, L. B. & Nørskov, J. K. Improved adsorption energetics within density-functional theory using revised perdew-burke-ernzerhof functionals. *Phys. Rev. B* **59**, 7413 (1999).
 58. Zhang, Y. & Yang, W. Comment on “generalized gradient approximation made simple”. *Phys. Rev. Lett.* **80**, 890 (1998).
 59. Csonka, G. I. Assessing the performance of recent density functionals for bulk solids. *Phys. Rev. B* **79**, 155107 (2009).
 60. Huang, B., Symonds, N. O., Lilienfeld, O. A. v. (eds.) *Quantum Machine Learning in Chemistry and Materials* (Springer, 2018).
 61. Batzner, S. E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nat. Commun.* **13**, 2453 (2022).
 62. Tang, W., Sanville, E. & Henkelman, G. A grid-based bader analysis algorithm without lattice bias. *J. Phys. Condens. Matter* **21**, 084204 (2009).
 63. Jung, R., Metzger, M., Maglia, F., Stinner, C. & Gasteiger, H. A. Oxygen release and its effect on the cycling stability of linixmycozo2 (nmc) cathode materials for lithium batteries. *J. Electrochem. Soc.* **164**, 1361 (2017).
 64. Arnaldsson, A. et al. *Code: Bader Charge Analysis*. <http://theory.cm.utexas.edu/henkelman/code/bader/> (2020).
 65. Lu, T. & Chen, Q. Interaction region indicator: a simple real space function clearly revealing both chemical bonds and weak interactions. *Chem. Methods* **1**, 231–239 (2021).
 66. de Silva, P. & Corminboeuf, C. Simultaneous visualization of covalent and non-covalent interactions using regions of density overlap. *J. Chem. Theory Comput.* **10**, 3745–3756 (2014).
 67. Gasteiger, J., Groß, J., Günnemann, S. *International Conference on Learning Representations* (ICLR, 2020).
 68. Behler, J. & Parrinello, M. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Phys. Rev. Lett.* **98**, 146401 (2007).
 69. Ramakrishnan, R., Dral, P., Rupp, M. & Anatole von Lilienfeld, O. Quantum chemistry structures and properties of 134 kilo molecules. <https://doi.org/10.6084/m9.figshare.c.978904.v5>. (2014).
 70. Jørgensen, P. B. & Bhowmik, A. *QM9 VASP Dataset*. Technical University of Denmark. <https://doi.org/10.11583/DTU.16794500> (2022).
 71. Jørgensen, P. B., Bhowmik, A. *NMC Dataset*. Technical University of Denmark. <https://doi.org/10.11583/DTU.16837721> (2022).
 72. Jørgensen, P. B. & Bhowmik, A. *Ethylene Carbonate Molecular Dynamics Dataset*. Technical University of Denmark. <https://doi.org/10.11583/DTU.16691825> (2022).
 73. Jørgensen, P. B. *DeepDFT Model Implementation*. <https://github.com/peterbjorgensen/DeepDFT> (2022).

ACKNOWLEDGEMENTS

Peter Bjørn Jørgensen and Arghya Bhowmik acknowledges financial support from VILLUM FONDEN by a research grant (00023105) for the DeepDFT project.

AUTHOR CONTRIBUTIONS

A.B. has designed and outlined the project and its research goal. P.B.J. and A.B. have designed and developed the model and method. P.B.J. has carried out the software engineering, including implementation of model and training framework, implementation of data workflows for the datasets, and analysis of data and results. A.B. has supervised the research and designed the numerical experiments. A.B. and P.B.J. have both written the manuscript through mutual discussion.

COMPETING INTERESTS

The authors declare no competing interests

ADDITIONAL INFORMATION

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41524-022-00863-y>.

Correspondence and requests for materials should be addressed to Peter Bjørn Jørgensen or Arghya Bhowmik.

Reprints and permission information is available at <http://www.nature.com/reprints>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2022