Towards Combinatorial Generalization for Catalysts: A Kohn-Sham Charge-Density Approach

Phillip Pope
University of Maryland, College Park
pepope@cs.umd.edu

David Jacobs
University of Maryland, College Park
dwj@umd.edu

Abstract

The Kohn-Sham equations underlie many important applications such as the discovery of new catalysts. Recent machine learning work on catalyst modeling has focused on prediction of the energy, but has so far not yet demonstrated significant out-of-distribution generalization. Here we investigate another approach based on the pointwise learning of the Kohn-Sham charge-density. On a new dataset of bulk catalysts with charge densities, we show density models can generalize to new structures with combinations of elements not seen at train time, a form of combinatorial generalization. We show that over 80% of binary and ternary test cases achieve faster convergence than standard baselines in Density Functional Theory, amounting to an average reduction of 13% in the number of iterations required to reach convergence, which may be of independent interest. Our results suggest that density learning is a viable alternative, trading greater inference costs for a step towards combinatorial generalization, a key property for applications.

1 Introduction

The Kohn-Sham (KS) equations are a nonlinear eigenvalue problem of the form $H[\rho]\Psi = E\Psi$, where H is a symmetric diagonally dominant matrix called the *Hamiltonian*, Ψ is an eigenvector, E the associated eigenvalue, and $\rho(\mathbf{r}) = \sum_i |\Psi_i(\mathbf{r})|^2$ is a real-valued scalar field called the *charge density*, which is unknown a priori [40, 30]. The KS equations are nonlinear in the sense that the matrix H depends on the charge density ρ, which in turn depends on the eigenvectors Ψ of H. Typically it is solved by fixed-point iteration, where an initial guess for ρ is made and then a sequence of *linear* eigenvalue problems are solved until convergence. In the computational chemistry literature this is referred to as the *self consistent field* (SCF) iteration. The cost of the iteration is dominated by the eigenvalue problem [41]. Consequently methods of reducing the number of requisite iterations, e.g. with machine learning, are of great interest.

The KS equations lie at the foundation of Density Functional Theory (DFT), an approach to electronic structure theory which reformulates the N-particle quantum many-body problem in terms of an effective one-particle density [31]. DFT is widely used in a number of physico-chemical applications. One such application with important economic and environmental consequences and growing attention from the machine-learning community in recent years is the discovery of new catalysts [54].

Perhaps the greatest difficulty of catalyst discovery is the combinatorial size of the search space of structures. Even when reducing the candidate set of elements to 55, as done in the Open Catalyst Project [8], the number of possible element combinations grows very quickly: we have $\binom{55}{3} = 26,235$, $\binom{55}{4} = 341,055$, and so forth. Additional factors like choice of crystal lattice, surface orientations, binding sites, and adsorbates further complicate matters, but nevertheless the number of possible element combinations is a dominating factor. Given the immensity of this search space, *generalization to new combinations* is a key aspect for the success of property-predicting models in applications. Put

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

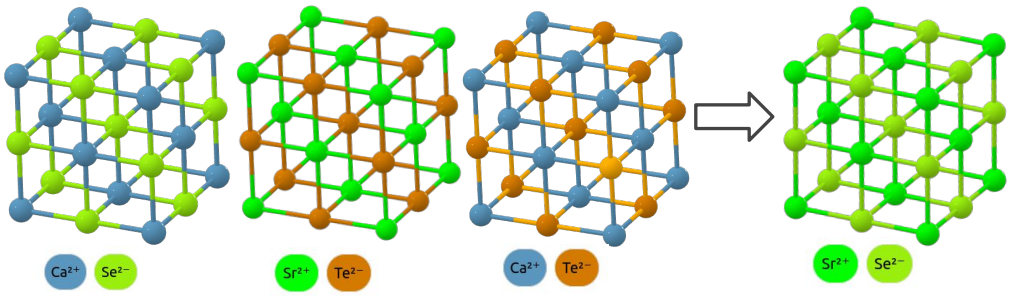

Figure 1: Simple illustration of a combinatorial generalization task with binary catalysts $\{\text{CaSe}, \text{SeTe}, \text{CaTe}\}$ for training and $\{\text{SrSe}\}$ for testing. Can density values of the structures on the left be used to predict the density values on the right? Note the combination of elements occurring in the test set does not occur in the training set, but its constituent elements do. Materials project IDs from left to right are `mp-1415`, `mp-1958`, `mp-1519`, and `mp-2758`. Structures visualized with the Materials Project web app [24]. NB: density values are not shown.

differently, predictive models that cannot combinatorially generalize will fail to cover large parts of the search space. Moreover some authors argue that combinatorial generalization is an intrinsically important property for machine learning models in general [4].

Recent research in machine-learning for catalyst modeling aims at directly predicting the energy and/or per-atom forces on large-scale benchmarks [8, 48]. The energy is a global property of a physical system which, among other reasons of theoretical importance, helps to assess the efficiency of catalytic reactions [54]. This line of work has spawned a number of innovations in equivariant and geometric graph learning [6, 55, 12, 11]. Despite these advances, generalization performance as measured by the Energy within Threshold (EWT) metric has not achieved greater than 20% of examples in any test split at the time of this writing [1]. It remains unclear if further development or scaling will lead to practical results.

An emerging alternative to energy prediction in the context of DFT systems is the *pointwise* prediction of the charge density [53, 17, 5, 27, 39]. The charge density is a local rather than global property of the system: a density value is associated to each point in the computational domain. The density is fundamental in DFT in the sense that all other properties may be computed from it, e.g. the energy by way of the eigenvalue problem, potentially eliminating the need for property-specific models in favor of a single model.

In this work we carry out an empirical investigation of density based models in catalyst systems. The focus of our investigation is whether such models can generalize to new combinations of elements. Our contributions are as follows:

- We compute a new large-scale DFT dataset of $\mathcal{O}(1000)$ relaxed bulk catalyst structures *with charge densities* spanning $\mathcal{O}(100\text{M})$ unique points using the open-source DFT software Quantum Espresso and workflow software AiiDA [37, 15, 21, 22, 49, 38].
- We design an evaluation methodology to measure when learned densities improve convergence versus standard baselines for density initialization in DFT. To do so we define a threshold-independent metric and also report the number of iterations saved at convergence.
- We show that density models improve convergence on new structures not seen at train time. Specifically we find that learned densities outperform baselines in 83% and 86% of test cases for binary and ternary catalyst respectively. These savings amount to a reduction of 13% in the number of SCF iterations needed to reach convergence.

To the best of our knowledge, our results are the first density-based approach to show improvement over standard baselines for density initialization in DFT on new structures not seen at train time.

2 Related Work

2.1 Catalyst Modeling

The discovery of new catalysts has received significant attention from the machine-learning community in recent years. Recent work in this area has primarily been lead by the Open Catalyst Project (OCP) [8]. Building on a line of works in geometric graph learning for atomic systems [51, 42], OCP has contributed a number of modeling developments [55, 12, 11, 46, 13] and large-scale datasets [8, 48]. See Joshi et al. [26] for a summary of recent approaches based on the level of geometric information incorporated in the model.

2.2 Learning the Kohn-Sham Density

A notable advantage of density-based approaches is that it naturally interfaces with the DFT algorithm, something not possible with energy models. Density predictions may be directly passed to the SCF cycle, which made be used to (1) better initialize a new SCF cycle versus standard baselines or (2) adaptively refine the predictions to increase the precision. In contrast the outputs of energy models are fixed and cannot be refined in this way.

Perhaps the most similar work to ours is that of Gong et al. [17], who use a graph-neural-network based approach for density prediction which encodes the local environment with a query node as we do. Using the CGGCNN [51] model, a predecessor to more advanced equivariant and geometric graph models that we use here, they perform a small-scale study of model transferability to new kinds of bonds not seen at train time, e.g. train on linear C-C-C and orthogonal C-O-C and test on orthogonal C-C-C (See Figure 7 of [17]).

Another similar work on density prediction is that of Zepeda-Núñez et al. [53], who represent an atomic structure as a *set* rather than a graph and define a novel network architecture with translation, rotation and permutation symmetries. They show up to four-digits of accuracy on non-catalyst test structures, including water, small organic molecules, and aluminum.

Other notable works include (1) Rackers et al. [39] who use Euclidean neural networks [14] to study generalization from smaller to larger clusters of water; (2) Brockherde et al. [5] who use a kernel-based approach to learn the dynamics of small organic molecules, Grisafi et al. [18] study transfer learning of density models from small molecules to larger ones such as hydrocarbons, (3) Jørgensen and Bhowmik [27], who show small-scale results with a message-passing network using "two-step approach" to encode the structure graph and local environment separately, in contrast to the query node approach that Gong et al. [17] and the present work uses, and (4) Schütt et al. [43] who learn elements of the Hamiltonian matrix directly for single molecule cases using a modified Schnet model [42], and show savings in SCF iterations in select cases.

2.3 Combinatorial Generalization

Combinatorial generalization (CG) is generalization from simpler data to more complex data [4]. CG and related ideas have been the subject of many machine learning works across several domains including generative models [34, 23, 33], vision and language [52, 47], reinforcement-learning [25, 7, 50], visual and analogical reasoning [20], and puzzle-solving [3].

To the best of our knowledge there are few priors works that explicitly address CG in chemistry-related machine learning. Fernando [10] explores related ideas in the context of chemical systems. Gui et al. [19] propose an out-of-distribution graph learning benchmark with molecular tasks but do not explicitly focus on any combinatorial structure as we do here.

2.4 High-precision learning

Most modern machine learning models trained through stochastic optimization do not typically achieve training loss values around machine-precision, 10^{-16} for double-precision `float64`. One

notable exception is Michaud et al. [32], wherein the authors demonstrate a training method enriched with the spectrum of the Hessian for achieving near machine-precision loss performance on low-dimensional examples such as $y = x^2$. It is unclear if such techniques scale to larger models . We note the distinction between high-precision at train time, a problem of optimization, and high-precision at test time, a problem of *generalization*. Colbrook et al. [9] argue there are fundamental limits to the precision obtainable by neural networks.

3 Methods

3.1 Modifying geometric graph learning for local property prediction

Standard geometric graph learning typically predicts a global property of a structure/molecule, e.g. the energy or band gap. Here we modify this approach for local property prediction by adding a "virtual atom" representing the query point to the graph. This approach is similar to Gong et al. [17], except our case deals with geometric graphs.

Let $s = \{(\mathbf{R}_i, Z_i)\}_{i=1}^{I}$ denote a structure s with N atoms at positions $\mathbf{R}_i \in \mathbb{R}^3$ with atomic numbers Z_i. Let $\rho_s : \mathbb{R}^3 \to \mathbb{R}$ denote the density function associated to a structure s, evaluated on a grid of query points $\{\mathbf{r}_j\}_{j=1}^{J}$, with all $\mathbf{r}_j \in \mathbb{R}^3$.

In the usual formulation, radial graphs are typically created from each structure, i.e. draw an edge between two atoms if they lie within some radius of each other. To instead make the graph local to a point, we first adjoin the query point to the structure as a "virtual" atom $(\mathbf{r}_j, Z_{I+1})$, where we interpret the atomic number Z_{I+1} simply as just an unused index on atom types. Note we use the same index for all query points. Letting $\mathcal{G}(\cdot)$ denote the operation of taking the radial graph, the localized graph then takes the form $\mathcal{G}(s \cup (\mathbf{r}_j, Z_{I+1}))$. Denote all such pairs for a structure as $\mathcal{D}_s = \{\mathcal{G}(s \cup (\mathbf{r}_j, Z_{I+1})), \rho_s(\mathbf{r}_j)\}$. Lastly, we complete the dataset by taking a union over all structures $\mathcal{D} = \bigcup_s \mathcal{D}_s$.

Given this data, we proceed as usual to stochastically optimize parameters θ of a graph neural network GNN to minimize the expected loss $\mathcal{L}$ of samples g, ρ drawn from a (uniform) probability measure $\mathbb{P}_{\mathcal{D}}$ on the dataset $\mathcal{D}$:

$$\min_{\theta} \mathop{\mathbb{E}}_{g,\rho\sim\mathbb{P}_{\mathcal{D}}} \Big[\mathcal{L}\left(\text{GNN}_\theta(g), \rho\right) \Big] \tag{1}$$

Since densities are continuous real-valued quantities, a regression loss is used.

3.2 Creating a dataset of catalyst charge densities with Quantum Espresso and AiiDA

Although there are emerging datasets of charge-densities [45], our evaluation requires that the settings of the data and solver be *exactly the same* to ensure numerical consistency. To that end, we generate our own dataset of relaxed bulk catalysts using the well-established and open-source DFT solver Quantum Espresso [16, 15] and modern computational workflow management system AiiDA [21]. In particular we utilize the relaxation workflows of Huber et al. [22], which automate the selection of computational parameters critical for achieving convergence based on expert knowledge.

For initial candidates for bulk catalysts we use the list of bulk catalysts identified by [8] from the Materials Project [24], which contains $11,410$ structures sampled from 1, 2 or 3 element combinations of a set of 55 elements. The total number of element combinations represented in this set is 5099.

Grid sizes per structures can vary by up to four orders of magnitude, which can strongly skew the training data toward larger structures. To mitigate this we restrict the maximum number of atoms to be 12 and the maximum volume of the (conventional) cells to 200 Å^3. We then remove candidates with oxidation states which only occur once in the dataset, since these are not well-represented enough to evaluate, using the oxidation state guesser in `pymatgen` [35].

We then partition the train/val/test splits as follows. First we assign all unary catalysts remaining to the train set. Then we randomly sample binaries with oxidation states not already represented in the unaries. Next, we remove oxidation states not present in the training set from the remaining binary and ternary candidates. This preprocessing step performed to ensure all states in the val/test splits

Table 1: Sample sizes of dataset

	$N_{\text{structures}}$	N_{samples}
Train (unary)	47	3.8M
Train (binary)	392	69M
Val (binary)	4	300k
Test (binary)	360	72M
Test (ternary)	1116	380M

were represented in the training set, e.g. training on H^+ cannot be expected to generalize to H^-. Finally we assign a few binaries to the validation split and the remaining binaries and ternaries to the testing split. Finally we validate that train and test splits are disjoint by element combinations, i.e. no combination of elements in training set is repeated in the test splits.

We relax each structure in these splits using Quantum Espresso with parameters set by the AiiDA relaxation workflow [22]. We use the `fast` protocol with the PBE exchange-correlation functional [36] for all relaxations. Initial structure positions were first perturbed with noise. Unconverged runs were dropped from the dataset. We report the number of structures and density samples in Table 1.

3.3 Evaluating convergence of learned densities versus standard baselines in DFT

An important baseline for density based models is the "atomic charge superposition" (ACS) initialization for charge densities, commonly used in DFT solvers [2]. The ACS baseline is derived from the pre-computed collection of atomic *pseudopotentials* used with the DFT solver, an important technique that reduces the number of electrons in the computation. For further details see [41, 30, 31, 29].

More precisely, given a configuration of atoms $\{(\mathbf{R}_i, Z_i)\}_{i=1}^I$ and associated atomic charge densities ρ_{Z_i}, the ACS initialization is $\rho_0^{ACS}(\mathbf{r}) = \sum_i^I \rho_{Z_i}(\mathbf{r} - \mathbf{R}_i)$. In other words it is simply a sum of radial functions. Note this initialization corresponds to a scenario of non-interacting atoms: poor performance may be expected when the atoms interact, i.e. when there is chemistry.

In this work we consider learned densities that do not improve upon this baseline to *not* be of interest. From a practical point of view, clearly initializations which do not beat this baseline would not be relevant. Nevertheless, given the difficulty of obtaining high precision results with learning-based approaches, this baseline is highly non-trivial.

The degree of precision required depends on the application. One could set a threshold and check which density reached convergence in fewest iterations. However, we wish to perform the evaluations in a threshold independent way to remain agnostic to the requirements of application, similar in spirit to "area-under-the-curve" (AUC) measures popular in classification.

To that end we propose a "signed area under curves" (s-AUC) measure, essentially the area between the curves of convergence for the baseline and learned densities. See Figure 6 for examples. Note the sign is necessary because the curves may cross. More formally, if $\{x_i\}_{i=1}^{n_x}$ and $\{y_j\}_{j=1}^{n_m}$ are the SCF accuracy values for the baseline and learned curves respectively, and $n = \min(n_x, n_y)$ we define

$$\text{s-AUC} = \frac{\sum_{i=1}^{n} \log(x_i) - \log(y_i)}{\sum_{i=1}^{n} |\max(\log(x_i), \log(y_i))|} \tag{2}$$

where we have used a log-scale to show rate of convergence and normalized by the pointwise max to facilitate comparison between difference curves.

We also report the percentage of iterations saved at convergence. Although this measure is threshold dependent, it has the benefit of being easily interpretable to the ML+DFT community.

3.4 Training with Spherical Channel Networks

We train a reduced sized Spherical Channel Network (SCN) for density prediction. We reduce the number of interactions to 10 and hidden channels to 256, resulting in 35M parameters, which is

smaller than state of the art models [55]. We use interaction cutoff to 6.0Å, and a maximum number of neighbors per node of 12. We list full hyperparameters in the Supplementary.

We train two replicates of SCNs for 420k steps with batch size 45 across 8 GPUs. Regarding compute, for training we used approximately 2000 GPU hours for both replicates across all GPUs. We use `NVIDIA A5000` cards with 24 GB of memory.

In initial experiments we also evaluated Dimenet++ [11] and GemNet [13] models but generally found performance on these models to be less than SCN models.

4 Results

4.1 Model performance

In training we obtain mean absolute errors (MAE) of $\mathcal{O}(10^{-4})$. Best validation MAE was 0.0011 for both replicates. One replicate was chosen to report the remainder the results. We show the distribution of MAE scores across each train/test split in Figure 2.

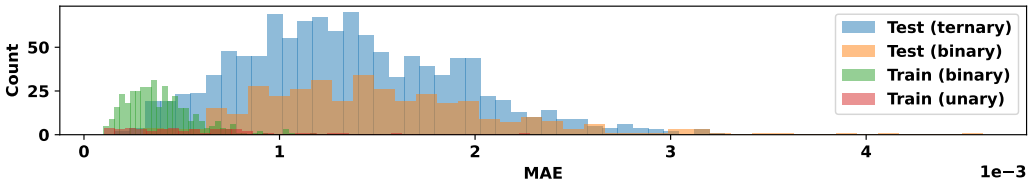

Figure 2: Distributions of MAE scores for train/test splits

4.2 Visualization of learned densities and errors

One advantage of local property prediction is the ability to directly visualize errors as a function of input points. In Figure 3 we show the true density, predicted density, and (scaled) absolute error for the top and bottom test predictions by MAE.

These visualizations allow us to inspect of the distribution of errors, which may facilitate e.g. model debugging, among other uses. Here we see a variety of patterns: ranging from nearly non-existent to quite strong errors near the core region of atoms. A closer analysis of these error patterns may inform future improvements to the dataset and model.

4.3 Evaluation of learned density convergence

We evaluate if initializing with learned densities leads to faster convergence in the SCF cycle. After predicting across the grid associated to each structure, we pass predictions to the DFT solver to initialize a new SCF cycle using *exactly the same runtime parameters* as the ground-truth run. Then we extract convergence information and compare them against the ground truth.

We show summary statistics of our results on SCF savings according to the s-AUC metric in Table 2. Notably we achieve positive savings in 83% of test binary cases and 86% of ternary cases, with a combined proportion of 86%. The mean and median savings in each case are positive.

In addition to the s-AUC metric, we also report savings in terms of the relative percentage of iterations saved at convergence in Table 3. The percentage of test structures with positive savings was 71% and 75% of binary and ternary test cases. The mean percentage of iterations saved was 13% for both binary and ternary splits.

We show the distribution of s-AUC scores and iteration savings across splits in Figures 4 and 5.

W plot the convergence behavior of top test cases by s-AUC in Figure 6. Positive savings can be seen when blue-dashed curves lie beneath the red-solid lines.

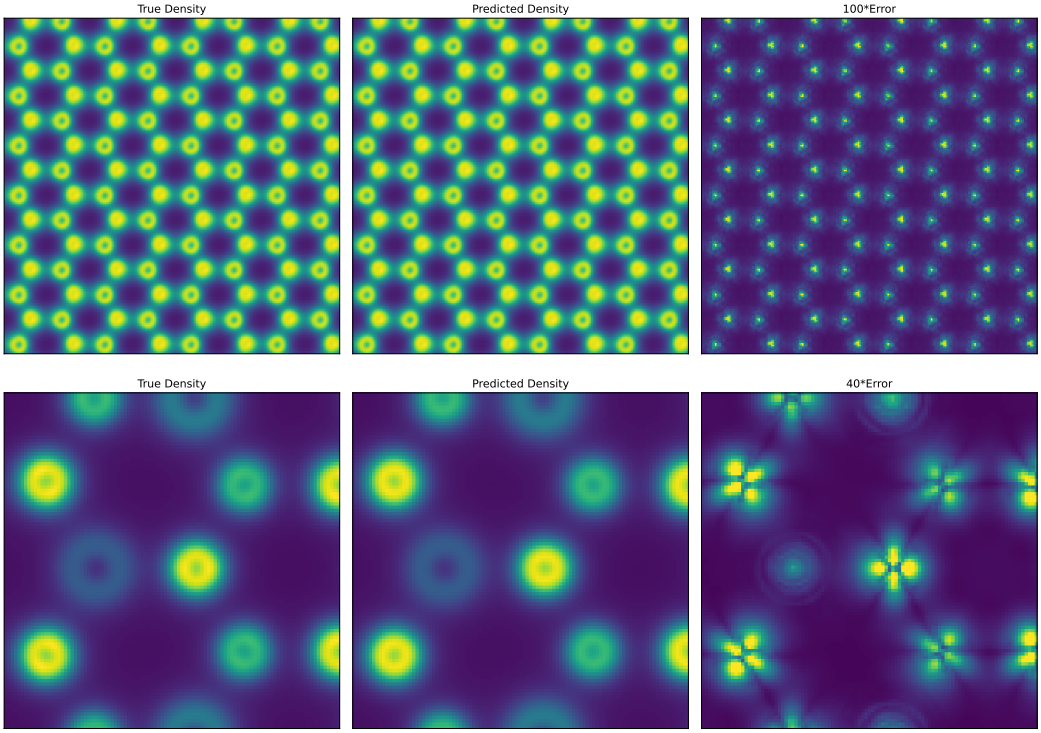

Figure 3: Test densities (2D projections): ground truth, predicted, and error. Top row is low error test structure MoS_4W(`mp-1023954`) and bottom row is higher error test structure Fe_6Re_2(`mp-865212`) with higher MAE error. Errors rescaled display more clearly. To visualize 3D densities are regridded to approximately a cube [44] and summed along one axis.

Table 2: Summary statistics of SCF savings with the s-AUC metric.

	N	N_+	N_+/N (%)	Mean	Median	Max	Min
Train (unary)	47	45	96	0.6	0.3	7.3	-0.1
Train (binary)	392	379	97	4.0	0.6	2.9×10^2	-0.5
Test (binary)	360	298	83	7.8	0.2	1.9×10^3	-3.7
Test (ternary)	1116	965	86	2.7	0.4	7.2×10^2	-2.8×10^2

Table 3: Summary statistics of relative iterations saved at convergence.

	N	N_+	N_+/N (%)	Mean (%)	Median (%)	Max (%)	Min (%)
Train (unary)	47	32	68	15	14	78	-53
Train (binary)	392	361	92	22	21	63	-50
Test (binary)	360	256	71	13	12	72	-53
Test (ternary)	1116	841	75	13	12	65	-90

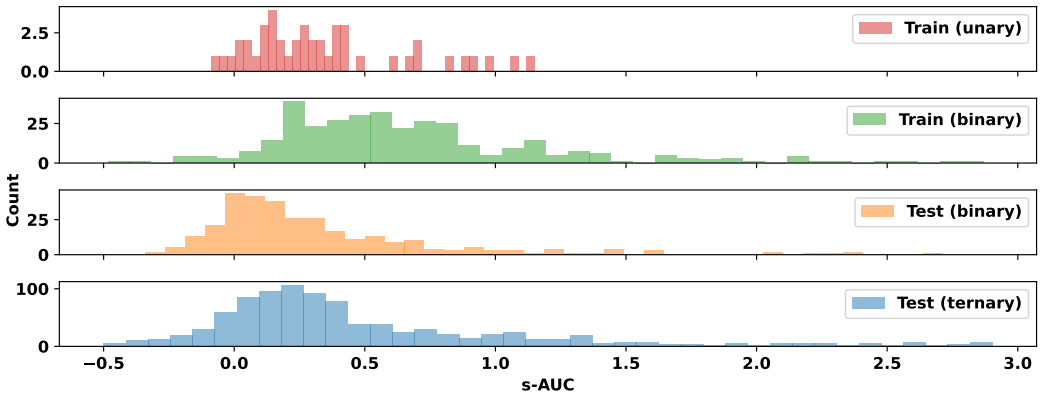

Figure 4: Distributions of `s-AUC` metrics. Outliers dropped for visualization.

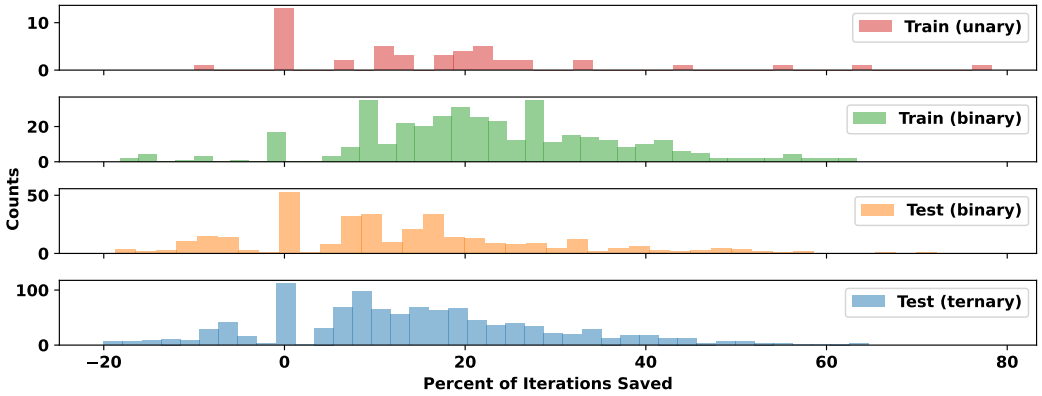

Figure 5: Distributions of relative iteration savings. Outliers dropped for visualization.

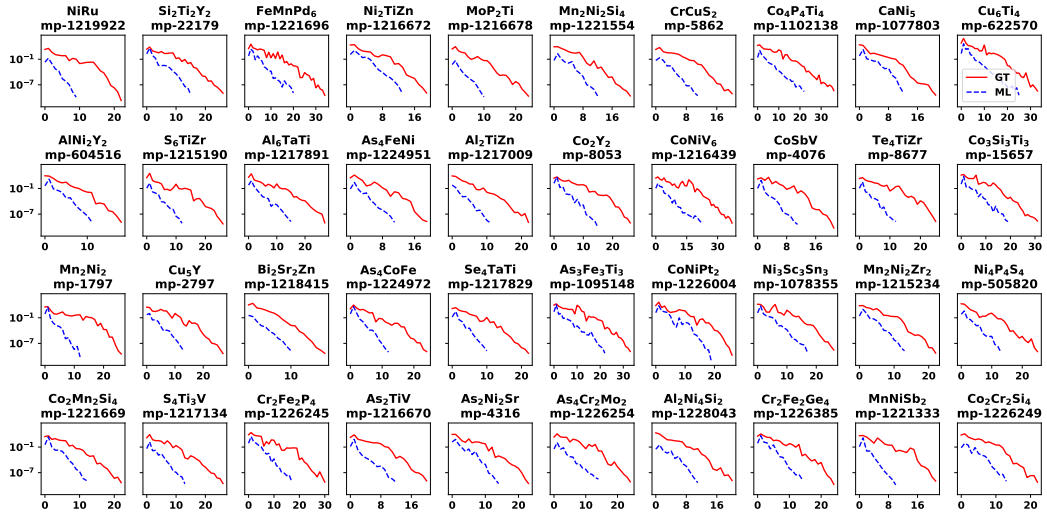

Figure 6: Top convergence results for test structures by s-AUC metric. Red solid lines are ground-truth DFT convergence, blue dashed lines are learned densities. Y-axis is SCF accuracy on a log-scale and x-axis is number of SCF iterations. Chemical formulas and material project ids given above each subplot. Note these structures were not seen at train time.

4.4 Validating learned densities

We perform two validations to check if learned densities converge to the same solution as the ground truth.

First we checked for any bias in converged energy values between ground-truth and learned densities. We find almost zero difference these energy values. On the test set we found 88% of structures had exactly zero difference in energy values. Of the cases with non-zero error, the relative error on average was 1.2×10^{-11} and at maximum was 1.5×10^{-9}.

Second, we check if a downstream property computed from the learned densities matches their ground-truth counterparts. We compute and compare the band gaps computed from the ground-truth and ML densities using the built-in tool in ASE [28]. We find 95% of values to be exactly the same, with maximum relative error at 7.8×10^{-5}.

5 Discussion

Overall we find in over 80% of test cases learned densities achieve faster convergence versus the baseline. Importantly these test cases are out-of-distribution, involving structures with new combinations of elements not seen at train time. Our results show that training on simpler structures, i.e. unary and binary, can lead to generalization on more complex structures, i.e. ternary.

6 Limitations

In this work we only test generalization for combinations of elements, and ignore stoichiometric ratios. In addition, we only investigate bulk catalysts, not the full complexity of catalyst systems as modeled in Chanussot et al. [8] which includes adsorbates, surfaces, and non-equilibrium points. Extending our analysis to these more complex systems is left to future work.

Our energy values are not directly comparable with those of Chanussot et al. [8], because we generated our dataset with a different DFT solver and pseudo-potentials. Systematic numerical differences may exist between different solvers, which makes comparison of energy values difficult.

Another limitation is that we do not report timings of our method versus traditional CPU-bound methods. Because of the high GPU-bound inference costs of our method, we do not expect timing improvement versus traditional methods. Sufficiently many GPUs will help mitigate this, as well as recent hardware development trends towards larger GPU memory.

7 Conclusion

We generated a new dataset of bulk catalyst relaxations with charge densities using open-source software. We trained a density-model pointwise on unary and binary catalysts. We showed this model can generalize to new binary and ternary catalysts, a form of combinatorial generalization. We found learned densities lead to faster convergence than standard baselines used to initialize the SCF cycle of DFT in over 80% of cases, amounting to an average of 13% saving in iterations.

7.1 Future Work

There are several directions for improvement to this work. The first is improving the proportion of test samples with positive savings, where clearly higher is better. In addition, expanding the class of structures tested, e.g. to quaternary structures and beyond, is of keen interest in applications.

A second direction is improving the rate of convergence of the learned densities. At present a number of SCF iterations are still necessary to reach high levels of convergence e.g. 10^{-9}. The ideal case would be to converge rapidly to a single or few SCF iterations.

Rapid convergence across many kinds of structures opens up the interesting possibility of an end-to-end hybrid learning/numerical model which first precisely predicts the density then computes the energy in one or a few SCF iterations. Such models may have different Pareto curve properties than current energy-based approaches, and may be useful in applications with high precision requirements.

8 Broader Impact

One way this work may have positive broader impact is through reducing costs of DFT computations. For example, supposing that the USA National Labs spend $100M on DFT per year, and a negligible cost of inference, then a 13% reduction in SCF iterations would yield a savings of $13M [1]. Although there are a number of practical details to resolve, we believe this work is an important first step.

The potential impacts of new catalysts discoveries may have significant economic and environmental consequences. Positive impacts may include accelerating the transition towards renewable energy, an important goal for future societies. Negative impacts may include unintended consequences such as contributing to unsustainable population growth. In addition, as with any new technology with potentially large impacts in the economy, there is risk that control of the technology may concentrate in the hands of the few and limit the distribution of benefits.

In early stages of research is difficult to discern if the aims will come to fruition. The computational discovery of new catalysts does not necessarily imply that the results are translatable into real-world materials, e.g. not all materials are known how to be made easily or economically. If such technology can be discovered, we believe that the benefits can be more equitably distributed by making this research completely open-source and accessible to global researchers. Our use of open-source DFT solvers is one notable step towards this goal over previous approaches [8].

9 Acknowledgements

PP thanks Professors Maria Cameron, Howard Elman, Tom Goldstein, Ramani Duraiswami, Pratyush Tiwary, and Hong-Zhou Ye, as well as Dr. Ruiyu Wang for feedback on this project. PP also thanks the University of Maryland Institute for Advanced Computer Studies (UMIACS) for providing the computing infrastructure used in this project.

[1] We thank an anonymous reviewer for suggesting this way of interpreting our results

References

[1] Open Catalyst Project - Public Leaderboard OC20. URL https://opencatalystproject.org/leaderboard.html.

[2] Quantum Espresso - pw.x input description: startingpot. URL https://www.quantum-espresso.org/Doc/INPUT_PW.html#idm899.

[3] David G. T. Barrett, Felix Hill, Adam Santoro, Ari S. Morcos, and Timothy Lillicrap. Measuring abstract reasoning in neural networks, 2018.

[4] Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

[5] Felix Brockherde, Leslie Vogt, Li Li, Mark E. Tuckerman, Kieron Burke, and Klaus-Robert Müller. Bypassing the Kohn-Sham equations with machine learning. *Nature Communications*, 8(1):872, October 2017. ISSN 2041-1723. doi: 10.1038/s41467-017-00839-3. URL https://www.nature.com/articles/s41467-017-00839-3. Number: 1 Publisher: Nature Publishing Group.

[6] Michael M. Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges, 2021.

[7] Michael Chang, Alyssa L. Dayan, Franziska Meier, Thomas L. Griffiths, Sergey Levine, and Amy Zhang. Neural constraint satisfaction: Hierarchical abstraction for combinatorial generalization in object rearrangement, 2023.

[8] Lowik Chanussot, Abhishek Das, Siddharth Goyal, Thibaut Lavril, Muhammed Shuaibi, Morgane Riviere, Kevin Tran, Javier Heras-Domingo, Caleb Ho, Weihua Hu, Aini Palizhati, Anuroop Sriram, Brandon Wood, Junwoong Yoon, Devi Parikh, C. Lawrence Zitnick, and Zachary Ulissi. Open Catalyst 2020 (OC20) Dataset and Community Challenges. *ACS Catalysis*, 11(10):6059–6072, May 2021. doi: 10.1021/acscatal.0c04525. URL https://doi.org/10.1021/acscatal.0c04525. Publisher: American Chemical Society.

[9] Matthew J. Colbrook, Vegard Antun, and Anders C. Hansen. The difficulty of computing stable and accurate neural networks: On the barriers of deep learning and smalersquo;s 18th problem. *Proceedings of the National Academy of Sciences*, 119(12), Mar 2022. doi: 10.1073/pnas.2107151119. URL https://doi.org/10.1073%2Fpnas.2107151119.

[10] Chrisantha Fernando. Design for a darwinian brain: Part 1. philosophy and neuroscience, 2013.

[11] Johannes Gasteiger, Janek Groß, and Stephan Günnemann. Directional message passing for molecular graphs. *arXiv preprint arXiv:2003.03123*, 2020.

[12] Johannes Gasteiger, Florian Becker, and Stephan Günnemann. Gemnet: Universal directional graph neural networks for molecules. *Advances in Neural Information Processing Systems*, 34:6790–6802, 2021.

[13] Johannes Gasteiger, Muhammed Shuaibi, Anuroop Sriram, Stephan Günnemann, Zachary Ulissi, C. Lawrence Zitnick, and Abhishek Das. Gemnet-oc: Developing graph neural networks for large and diverse molecular simulation datasets, 2022.

[14] Mario Geiger and Tess Smidt. e3nn: Euclidean neural networks, 2022. URL https://arxiv.org/abs/2207.09453.

[15] P. Giannozzi, O. Andreussi, T. Brumme, O. Bunau, M. Buongiorno Nardelli, M. Calandra, R. Car, C. Cavazzoni, D. Ceresoli, M. Cococcioni, N. Colonna, I. Carnimeo, A. Dal Corso, S. de Gironcoli, P. Delugas, R. A. DiStasio, A. Ferretti, A. Floris, G. Fratesi, G. Fugallo, R. Gebauer, U. Gerstmann, F. Giustino, T. Gorni, J. Jia, M. Kawamura, H.-Y. Ko, A. Kokalj, E. Küçükbenli, M. Lazzeri, M. Marsili, N. Marzari, F. Mauri, N. L. Nguyen, H.-V. Nguyen, A. Otero-de-la Roza, L. Paulatto, S. Poncé, D. Rocca, R. Sabatini, B. Santra, M. Schlipf, A. P. Seitsonen, A. Smogunov, I. Timrov, T. Thonhauser, P. Umari, N. Vast, X. Wu, and S. Baroni. Advanced capabilities for materials modelling with Quantum ESPRESSO. *Journal of Physics: Condensed Matter*, 29(46):465901, October 2017. ISSN 0953-8984. doi: 10.1088/1361-648X/aa8f79. URL https://doi.org/10.1088/1361-648x/aa8f79. Publisher: IOP Publishing.

[16] Paolo Giannozzi, Stefano Baroni, Nicola Bonini, Matteo Calandra, Roberto Car, Carlo Cavazzoni, Davide Ceresoli, Guido L Chiarotti, Matteo Cococcioni, Ismaila Dabo, Andrea Dal Corso, Stefano de Gironcoli, Stefano Fabris, Guido Fratesi, Ralph Gebauer, Uwe Gerstmann, Christos Gougoussis, Anton Kokalj, Michele Lazzeri, Layla Martin-Samos, Nicola Marzari, Francesco Mauri, Riccardo Mazzarello, Stefano Paolini, Alfredo Pasquarello, Lorenzo Paulatto, Carlo Sbraccia, Sandro Scandolo, Gabriele

Sclauzero, Ari P Seitsonen, Alexander Smogunov, Paolo Umari, and Renata M Wentzcovitch. QUANTUM ESPRESSO: a modular and open-source software project for quantum simulations of materials. *Journal of Physics: Condensed Matter*, 21(39):395502, September 2009. ISSN 0953-8984, 1361-648X. doi: 10.1088/0953-8984/21/39/395502. URL https://iopscience.iop.org/article/10.1088/0953-8984/21/39/395502.

[17] Sheng Gong, Tian Xie, Taishan Zhu, Shuo Wang, Eric R. Fadel, Yawei Li, and Jeffrey C. Grossman. Predicting charge density distribution of materials using a local-environment-based graph convolutional network. *Physical Review B*, 100(18):184103, November 2019. doi: 10.1103/PhysRevB.100.184103. URL https://link.aps.org/doi/10.1103/PhysRevB.100.184103. Publisher: American Physical Society.

[18] Andrea Grisafi, Alberto Fabrizio, Benjamin Meyer, David M. Wilkins, Clemence Corminboeuf, and Michele Ceriotti. Transferable Machine-Learning Model of the Electron Density. *ACS Central Science*, 5 (1):57–64, January 2019. ISSN 2374-7943. doi: 10.1021/acscentsci.8b00551. URL https://doi.org/10.1021/acscentsci.8b00551. Publisher: American Chemical Society.

[19] Shurui Gui, Xiner Li, Limei Wang, and Shuiwang Ji. Good: A graph out-of-distribution benchmark. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 2059–2073. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/0dc91de822b71c66a7f54fa121d8cbb9-Paper-Datasets_and_Benchmarks.pdf.

[20] Felix Hill, Adam Santoro, David Barrett, Ari Morcos, and Timothy Lillicrap. Learning to make analogies by contrasting abstract relational structure. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=SylLYsCcFm.

[21] Sebastiaan P Huber, Spyros Zoupanos, Martin Uhrin, Leopold Talirz, Leonid Kahle, Rico Häuselmann, Dominik Gresch, Tiziano Müller, Aliaksandr V Yakutovich, Casper W Andersen, et al. Aiida 1.0, a scalable computational infrastructure for automated reproducible workflows and data provenance. *Scientific data*, 7 (1):300, 2020.

[22] Sebastiaan P Huber, Emanuele Bosoni, Marnik Bercx, Jens Bröder, Augustin Degomme, Vladimir Dikan, Kristjan Eimre, Espen Flage-Larsen, Alberto Garcia, Luigi Genovese, et al. Common workflows for computing material properties using different quantum engines. *npj Computational Materials*, 7(1):136, 2021.

[23] Geonho Hwang, Jaewoong Choi, Hyunsoo Cho, and Myungjoo Kang. MAGANet: Achieving combinatorial generalization by modeling a group action. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 14237–14248. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/hwang23b.html.

[24] Anubhav Jain, Shyue Ping Ong, Geoffroy Hautier, Wei Chen, William Davidson Richards, Stephen Dacek, Shreyas Cholia, Dan Gunter, David Skinner, Gerbrand Ceder, et al. Commentary: The materials project: A materials genome approach to accelerating materials innovation. *APL materials*, 1(1):011002, 2013.

[25] Yiding Jiang, Shixiang Shane Gu, Kevin P Murphy, and Chelsea Finn. Language as an abstraction for hierarchical deep reinforcement learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[26] Chaitanya K. Joshi, Cristian Bodnar, Simon V. Mathis, Taco Cohen, and Pietro Liò. On the expressive power of geometric graph neural networks, 2023.

[27] Peter Bjørn Jørgensen and Arghya Bhowmik. Deepdft: Neural message passing network for accurate charge density prediction, 2020.

[28] Ask Hjorth Larsen, Jens Jørgen Mortensen, Jakob Blomqvist, Ivano E Castelli, Rune Christensen, Marcin Dułak, Jesper Friis, Michael N Groves, Bjørk Hammer, Cory Hargus, Eric D Hermes, Paul C Jennings, Peter Bjerre Jensen, James Kermode, John R Kitchin, Esben Leonhard Kolsbjerg, Joseph Kubal, Kristen Kaasbjerg, Steen Lysgaard, Jón Bergmann Maronsson, Tristan Maxson, Thomas Olsen, Lars Pastewka, Andrew Peterson, Carsten Rostgaard, Jakob Schiøtz, Ole Schütt, Mikkel Strange, Kristian S Thygesen, Tejs Vegge, Lasse Vilhelmsen, Michael Walter, Zhenhua Zeng, and Karsten W Jacobsen. The atomic simulation environment—a python library for working with atoms. *Journal of Physics: Condensed Matter*, 29(27):273002, 2017. URL http://stacks.iop.org/0953-8984/29/i=27/a=273002.

[29] Lin Lin and Jianfeng Lu. *A mathematical introduction to electronic structure theory*. SIAM, 2019.

[30] Lin Lin, Jianfeng Lu, and Lexing Ying. Numerical methods for Kohn–Sham density functional theory. *Acta Numerica*, 28:405–539, May 2019. ISSN 0962-4929, 1474-0508. doi: 10.1017/S0962492919000047. URL https://www.cambridge.org/core/journals/acta-numerica/article/numerical-methods-for-kohnsham-density-functional-theory/755DFB88349DD5F1EE1E360AD61661BF. Publisher: Cambridge University Press.

[31] Richard M Martin. *Electronic structure: basic theory and practical methods*. Cambridge university press, 2020.

[32] Eric J. Michaud, Ziming Liu, and Max Tegmark. Precision machine learning. *Entropy*, 25(1), 2023. ISSN 1099-4300. doi: 10.3390/e25010175. URL https://www.mdpi.com/1099-4300/25/1/175.

[33] Milton L Montero, Jeffrey S Bowers, Rui Ponte Costa, Casimir JH Ludwig, and Gaurav Malhotra. Lost in latent space: Disentangled models and the challenge of combinatorial generalisation. *arXiv preprint arXiv:2204.02283*, 2022.

[34] Milton Llera Montero, Casimir JH Ludwig, Rui Ponte Costa, Gaurav Malhotra, and Jeffrey Bowers. The role of disentanglement in generalisation. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=qbH974jKUVy.

[35] Shyue Ping Ong, William Davidson Richards, Anubhav Jain, Geoffroy Hautier, Michael Kocher, Shreyas Cholia, Dan Gunter, Vincent L Chevrier, Kristin A Persson, and Gerbrand Ceder. Python materials genomics (pymatgen): A robust, open-source python library for materials analysis. *Computational Materials Science*, 68:314–319, 2013.

[36] John P. Perdew, Kieron Burke, and Matthias Ernzerhof. Generalized gradient approximation made simple. *Phys. Rev. Lett.*, 77:3865–3868, Oct 1996. doi: 10.1103/PhysRevLett.77.3865. URL https://link.aps.org/doi/10.1103/PhysRevLett.77.3865.

[37] Giovanni Pizzi, Andrea Cepellotti, Riccardo Sabatini, Nicola Marzari, and Boris Kozinsky. AiiDA: automated interactive infrastructure and database for computational science. *Computational Materials Science*, 111:218–230, January 2016. ISSN 0927-0256. doi: 10.1016/j.commatsci.2015.09.013. URL https://www.sciencedirect.com/science/article/pii/S0927025615005820.

[38] Gianluca Prandini, Antimo Marrazzo, Ivano E Castelli, Nicolas Mounet, and Nicola Marzari. Precision and efficiency in solid-state pseudopotential calculations. *npj Computational Materials*, 4(1):72, 2018.

[39] Joshua A. Rackers, Lucas Tecot, Mario Geiger, and Tess E. Smidt. Cracking the quantum scaling limit with machine learned electron densities, 2022.

[40] Yousef Saad. *Numerical methods for large eigenvalue problems: revised edition*. SIAM, 2011.

[41] Yousef Saad, James R. Chelikowsky, and Suzanne M. Shontz. Numerical Methods for Electronic Structure Calculations of Materials. *SIAM Review*, 52(1):3–54, January 2010. ISSN 0036-1445, 1095-7200. doi: 10.1137/060651653. URL http://epubs.siam.org/doi/10.1137/060651653.

[42] K. T. Schütt, H. E. Sauceda, P.-J. Kindermans, A. Tkatchenko, and K.-R. Müller. SchNet – A deep learning architecture for molecules and materials. *The Journal of Chemical Physics*, 148(24):241722, June 2018. ISSN 0021-9606. doi: 10.1063/1.5019779. URL https://aip.scitation.org/doi/10.1063/1.5019779. Publisher: American Institute of Physics.

[43] K. T. Schütt, M. Gastegger, A. Tkatchenko, K. R. Müller, and R. J. Maurer. Unifying machine learning and quantum chemistry – a deep neural network for molecular wavefunctions, 2019.

[44] Jimmy-Xuan Shen. mp-pyrho, October 2022. URL https://doi.org/10.5281/zenodo.7227448. If you use this software, please cite it using the metadata from this file.

[45] Jimmy-Xuan Shen, Jason M. Munro, Matthew K. Horton, Patrick Huck, Shyam Dwaraknath, and Kristin A. Persson. A representation-independent electronic charge density database for crystalline materials, 2021.

[46] Muhammed Shuaibi, Adeesh Kolluru, Abhishek Das, Aditya Grover, Anuroop Sriram, Zachary Ulissi, and C. Lawrence Zitnick. Rotation Invariant Graph Neural Networks using Spin Convolutions. *arXiv:2106.09575 [cs]*, June 2021. URL http://arxiv.org/abs/2106.09575. arXiv: 2106.09575.

[47] Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. Winoground: Probing vision and language models for visio-linguistic compositionality, 2022.

[48] Richard Tran, Janice Lan, Muhammed Shuaibi, Brandon M. Wood, Siddharth Goyal, Abhishek Das, Javier Heras-Domingo, Adeesh Kolluru, Ammar Rizvi, Nima Shoghi, Anuroop Sriram, Fé lix Therrien, Jehad Abed, Oleksandr Voznyy, Edward H. Sargent, Zachary Ulissi, and C. Lawrence Zitnick. The open catalyst 2022 (OC22) dataset and challenges for oxide electrocatalysts. *ACS Catalysis*, 13(5):3066–3084, feb 2023. doi: 10.1021/acscatal.2c05426. URL https://doi.org/10.1021%2Facscatal.2c05426.

[49] Martin Uhrin, Sebastiaan P Huber, Jusong Yu, Nicola Marzari, and Giovanni Pizzi. Workflows in aiida: Engineering a high-throughput, event-based engine for robust and modular computational workflows. *Computational Materials Science*, 187:110086, 2021.

[50] Marin Vlastelica, Michal Rolinek, and Georg Martius. Neuro-algorithmic policies enable fast combinatorial generalization. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 10575–10585. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/vlastelica21a.html.

[51] Tian Xie and Jeffrey C. Grossman. Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. *Phys. Rev. Lett.*, 120:145301, Apr 2018. doi: 10.1103/PhysRevLett.120.145301. URL https://link.aps.org/doi/10.1103/PhysRevLett.120.145301.

[52] Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it?, 2023.

[53] Leonardo Zepeda-Núñez, Yixiao Chen, Jiefu Zhang, Weile Jia, Linfeng Zhang, and Lin Lin. Deep Density: Circumventing the Kohn-Sham equations via symmetry preserving neural networks. *Journal of Computational Physics*, 443:110523, October 2021. ISSN 0021-9991. doi: 10.1016/j.jcp.2021.110523. URL https://www.sciencedirect.com/science/article/pii/S0021999121004186.

[54] C. Lawrence Zitnick, Lowik Chanussot, Abhishek Das, Siddharth Goyal, Javier Heras-Domingo, Caleb Ho, Weihua Hu, Thibaut Lavril, Aini Palizhati, Morgane Riviere, Muhammed Shuaibi, Anuroop Sriram, Kevin Tran, Brandon Wood, Junwoong Yoon, Devi Parikh, and Zachary Ulissi. An Introduction to Electrocatalyst Design using Machine Learning for Renewable Energy Storage. *arXiv:2010.09435 [cond-mat]*, October 2020. URL http://arxiv.org/abs/2010.09435. arXiv: 2010.09435.

[55] C. Lawrence Zitnick, Abhishek Das, Adeesh Kolluru, Janice Lan, Muhammed Shuaibi, Anuroop Sriram, Zachary Ward Ulissi, and Brandon M Wood. Spherical channels for modeling atomic interactions. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=5Z3GURcqwT.