MOLECULE PROPERTY PREDICTION WITH MOLECULAR ORBITALS

Anonymous authors
Paper under double-blind review

ABSTRACT

Molecular orbitals describe the distribution of electrons in a molecule and are frequently used by chemists to understand properties of molecules, yet deep learning has largely neglected them for property prediction so far. In many setups where atom coordinates are computed, they can be obtained cheaply and quickly – a useful source of complementary information, particularly when data is scarce. We give an introduction to molecular orbitals for a machine learning audience and propose models to process three different representations of them. Experiments on a dataset with experimental properties show that including molecular orbitals significantly improves performance and sample efficiency over a pretrained molecular foundation model in real-world settings.

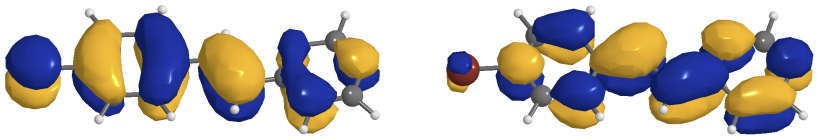

Figure 1: Two molecular orbitals of the same molecule, HOMO (left) and LUMO (right). Blue and yellow regions indicate the sign of the level set and are interchangeable.

1 INTRODUCTION

Machine learning requires quality and quantity of data to perform well. Unfortunately, in domains like materials discovery, this data can be hard to come by: oftentimes, properties of interest cannot be simulated accurately, but must be measured experimentally after chemical synthesis. This synthesis can be time-consuming and expensive. For example, synthesizing new organic light-emitting diode molecules (OLEDs, present in modern phone and TV displays, among others) can take several months of lead time and cost thousands of dollars with only a throughput of a few molecules per month, even in industrial labs. Therefore, it is unrealistic to obtain a substantial quantity of different molecules to use as a dataset. Meanwhile, such domains where data is scarce would also benefit the most from improved property predictors to help with identifying promising molecules for synthesis. In such cases, it is usually worth it to improve the quality of data at the cost of some extra compute.

One avenue is to make use of extra computational information. For state-of-the-art results, a set of atom positions for which the molecule is stable (ground-state conformation) is commonly used as input to geometric deep learning models. It is usually obtained by computational chemistry software like ORCA (Neese, 2012) – implementing methods like DFT (Hohenberg & Kohn, 1964; Kohn & Sham, 1965) and GFN2-xTB (Bannwarth et al., 2019) – which moves atom positions to minimize the molecule's total energy. A simple molecule development pipeline then looks as follows: candidate molecules are proposed (e.g. domain experts, generative models), analysis is performed by DFT or similar followed by geometric deep learning models (utilizing atom coordinates), and the most promising candidates are considered for synthesis. Such a pipeline makes sense when synthesis is costly, as it maximizes the quality of property predictions by obtaining atom coordinates through DFT or similar high-accuracy computational methods. This is the main setting we are interested in.

These software packages also produce other relevant data *at the same time* as atom positions. Molecular orbitals (**MO**s; Hund, 1928; Mulliken, 1928) are routinely analyzed by chemists to infer molecule properties (Szabo & Ostlund, 2012): they contain information about how electrons are

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

distributed in a molecule, enabling a richer representation of molecules than atom positions alone. Conveniently, to obtain atom positions using computational chemistry software, *molecular orbitals usually have to be computed anyway* (we explain this in section 2). In fact, MOs are frequently saved alongside the atom positions in the computation logs, so it is only a matter of loading instead of discarding them. By making use of MOs, we intend to strengthen the quality of predictions without the need for any additional computational chemistry calculations in the standard pipeline.

Contributions Surprisingly, relatively little work has been done that utilizes these MOs as inputs in deep learning despite their apparent utility in chemistry. Thus, we contribute the following:

1. We give a pedagogical introduction to molecular orbitals intended for a machine learning audience (section 2). This data modality has remained largely unexplored in deep learning for molecules and materials for the purposes of property prediction (we discuss a number of related works in-depth in Appendix A). Our exposition does not require significant chemistry background to be useful for machine learning researchers.
2. We propose three ways to represent MOs for processing by deep learning models, discuss their properties, and describe appropriate models for learning with these representations (section 3). The choice of representation and accommodation of symmetries is crucial for sample efficiency. Our innovation is not in the architectures themselves – a strength of our method is that we can use existing convolutional and rotation-equivariant graph neural networks – but how they are used to encode the specific symmetry properties of MOs.
3. Based on work by Joung et al. (2020), we propose a dataset of 2,336 OLED molecules with *experimentally-measured* properties that allows for better evaluation of models using molecular orbitals (section 4). This is important because many existing datasets use computational properties as targets, which may exhibit the same simulation errors as MOs compared to reality. By targeting experimentally-measured properties, we ensure that the task is relevant to inferring molecular properties in real-world settings.
4. We evaluate our methods on our dataset and the QM9 dataset against (and in combination with) a number of strong baselines (section 5). We show that incorporating MOs consistently improves results and sample efficiency of a state-of-the-art pretrained foundation model, as well as significantly outperforming standard methods that utilize electronic structure information.

2 Molecular orbitals for machine learning

What are molecular orbitals? In chemistry, they are wave-like functions in 3D space that provide a compact description of electronic structure at particular energy levels. For machine learning, they can be represented as vectors of coefficients with a particular structure. In this section, we aim to bridge these two worlds. We begin by providing context on the chemistry side to explain where molecular orbitals come from, and then discuss the implications on the machine learning side. We necessarily omit many details to get the most important points across concisely. For more details, we recommend textbook references such as by Jensen (2017a); Szabo & Ostlund (2012).

2.1 Background on molecular orbitals

A molecule is made up of atoms that are linked by chemical bonds. When forming these bonds, the distribution of electrons is rearranged to minimize the total energy. *Orbitals* are useful for describing this electronic structure and can be represented as wave-like functions over 3D space. In a one-electron system, a normalized orbital $\phi : \mathbb{R}^3 \to \mathbb{C}$ can be interpreted as describing where to find that electron, with $|\phi(\mathbf{r})|^2$ being the probability density (integrating to 1).

Electrons around individual atoms are often described using atomic orbitals (AOs), which are orbitals centered on the atom. When atoms form a molecule, these AO *basis functions* combine to form molecular orbitals (MOs), which provide a convenient representation of the electronic structure and are used (e.g., in Kohn–Sham DFT) to construct the electron density $\rho(\mathbf{r})$ for the molecule.

Their shape, as reflected in the regions where $|\phi_i(r)|^2$ is large, can be analyzed to develop chemical intuition and to help explain various molecular properties. For example, in molecules such as N_2, O_2, and F_2, certain atomic orbitals on each atom called 2p combine to form bonding and antibonding molecular orbitals that extend over the entire molecule. The way these molecular orbitals are

occupied explains both the trend in bond order (3, 2, and 1 for N_2, O_2, and F_2, respectively) and the paramagnetism of O_2, which arises from two unpaired electrons in antibonding π^* orbitals. This illustrates how the structure and energies of MOs can encode chemically meaningful properties.

Each MO is associated with a scalar (the MO energy), and MOs are typically sorted in ascending energy. A particularly important subset of MOs is known as the *frontier molecular orbitals* (Fukui et al., 1952). These are the MOs corresponding to the highest MO energy that is filled with electrons (highest occupied MO, HOMO) and the next higher MO energy, which is the lowest energy MO that is not filled with electrons (lowest unoccupied MO, LUMO). The HOMO-LUMO gap (the difference in energy between the HOMO and LUMO) is frequently targeted in property prediction as higher gaps tend to indicate more stable molecules. Adjacent frontier MOs can also be important (Berson, 1972); the ones below the HOMO are denoted as HOMO-1, HOMO-2, ..., and the ones above the LUMO are LUMO+1, LUMO+2, ... We choose to work with six frontier MOs (HOMO-2 to HOMO, LUMO to LUMO+2), as this is a common upper range we find in chemistry papers.

Representation So far, the discussion has treated orbitals as functions defined over 3D space. To work with them in a more tractable manner in software, they are discretized from (infinite-dimensional) objects to finite-dimensional vectors.

To accomplish this, MOs are represented as a *linear combination of atomic orbitals* (LCAO) in computational chemistry software. In this representation, an MO is a *vector of coefficients* for a number of *basis functions* that roughly model the shape of atomic orbitals. Each atom is associated with a number of basis functions centered on the atom, and the precise shape of these basis functions is defined by a *basis set*. There are many different basis sets with names such as 6-31G(d,p) and def2-svp with different trade-offs in size, speed, accuracy, and type of system being analyzed. The basis set and the atoms with their positions determine how the MOs of a molecule are represented as coefficients. It is standard and always possible to use real-valued coefficients and basis functions. A real molecular orbital Ψ_i is:

$$\Psi_i = \sum_{k=1}^{K_{\text{total}}} c_{k,i}\phi_k \tag{1}$$

specifying how to combine the K_{total} basis functions $\phi_k : \mathbb{R}^3 \to \mathbb{R}$ in the molecule to obtain the ith MO Ψ_i. The MO coefficients $c_{k,i}$ – with the knowledge of the basis set and atom positions – give us a finite-dimensional representation for a set of MOs. Additionally, there is a block structure: the coefficients $c_{1,i}, \ldots, c_{K_1,i}$ are the coefficients for the K_1 basis functions of the first atom, the next $c_{K_1+1,i}, \ldots, c_{K_1+K_2,i}$ are the coefficients for the K_2 basis functions of the second atom, and so on.

To design a model that takes this representation as input, we must encode not only the MO coefficients but also the basis functions – the basis changes between different molecules (with a different set of atoms), but also when the atom positions are changed. Each basis function ϕ_k is usually centered at a corresponding atom position. It maps a point in space, parametrized by its radial distance r to the atom center and spherical coordinates θ_1 and θ_2 to indicate direction, to a scalar value.

$$\phi_k(r, \theta_1, \theta_2) = R_k(r, l)Y_{l,m}(\theta_1, \theta_2) \tag{2}$$

This is done through a radial basis $R_k : \mathbb{R} \to \mathbb{R}$ and the real spherical harmonics $Y_{l,m} : S^2 \to \mathbb{R}$, a standard orthonormal basis for functions defined on the sphere. The basis set defines how many ϕ_k there are for each atom, and for each ϕ_k the shape of R_k and the angular momentum order l. We provide additional information in Appendix B, though it is not required for the following exposition.

Computation In Kohn-Sham DFT, a self-consistent field procedure is commonly used to solve the Kohn-Sham equations (Kohn & Sham, 1965), leading to a generalized eigenvalue problem:

$$FC = SCE \tag{3}$$

Of particular note here is that C will contain the MO coefficients of each MO as column eigenvectors, $C_{ki} = c_{k,i}$. F refers to the so-called Fock matrix, $S_{jk} = \langle \phi_j, \phi_k \rangle$ is the overlap matrix, and E is a diagonal matrix containing the MO energies λ_i (all matrices here are $\mathbb{R}^{K_{\text{total}} \times K_{\text{total}}}$). It is called self-consistent because the orbitals C (and their energies E) are derived from F, while F itself is constructed from the electron density implied by C. Thus, the goal is to find an F (or equivalently a pair (C, E)) that satisfies this equation self-consistently.

For a system of atoms and their positions, given a guess for F, computational chemistry software obtains C and E through the generalized eigendecomposition – C contains column eigenvectors (MOs) and E contains eigenvalues (MO energies). The eigendecomposition is considered generalized because the basis is typically not orthogonal; S specifies how to transform it to an orthogonal basis (if the basis is orthogonal, S is simply identity). From the occupied columns of C, DFT and methods like GFN2-xTB reconstruct the electron density and then use standard procedures (which are not important here) to build a new F. This procedure is repeated until F has converged.

To obtain atom positions that minimize energy, the geometry optimization procedure starts with an initial guess for the atom positions and solves Equation 3 (a single-point calculation), with the atom positions (and thus the basis) treated as fixed. Then, the atom positions are moved slightly, which changes the atom-centered basis functions, and we solve Equation 3 again. This is repeated until the atom positions have converged. We can see that geometry optimization requires solving the electronic-structure problem and thus yields MOs as part of the procedure. Therefore, when DFT or other standard computational chemistry methods are already part of the workflow (e.g., for geometry optimization), obtaining MO coefficients can be considered "free", requiring the practitioner only to store and use them. Depending on the size of molecules and accuracy desired, either DFT or faster methods like GFN2-xTB or FMO-xTB can be used, which scale to molecules beyond 20,000 atoms – we discuss exact costs for our cases in Appendix F. Even if coordinates are obtained with a different method such as machine learning force fields, a single-point computation on the fixed coordinates is far cheaper than full geometry optimization, which makes MOs feasible to obtain in large-scale settings.

3 Molecular orbital representations

Now that we have given an introduction to molecular orbitals, we need to decide on how to represent them for the purpose of processing with machine learning models. We propose three different approaches (formal definitions in Appendix C), their corresponding properties, and how to design models for them.

3.1 Direct MO representation

In this first approach, we use the existing MO coefficients $c_{k,i}$ directly, without modifying any of their values. For the information to be complete, we also need to encode the position and (radial and angular) shape of the basis functions. One MO would therefore be encoded as a set of pairs, each consisting of a scalar MO coefficient and the corresponding basis function information. The benefit of this approach is that it is closest to the data coming from DFT and reuses the well-established basis functions of the particular basis set. Conversely, while these basis functions work well for DFT, their suitability for machine learning purposes is unknown.

By maintaining some additional structure in the set to keep certain basis functions grouped together, we may be able to gain some further modeling benefits. We consider two practical levels of partitions, or *blocks*, of the basis elements. This choice governs the level of abstraction our model will be working with. A **subshell block** contains all the basis functions of the same atom, same radial basis R_k, and same l, but different m (see details in Appendix B). The number of m components in the same subshell differs with l. An l has $2l + 1$ valid values for m, meaning that the subshell block size varies accordingly as well. On the other hand, **atom blocks** group the basis functions based on which atom the basis functions belong to and are centered on. Therefore, each atom block consists of multiple subshell blocks. Each atom block has a unique position (two atoms cannot share the same position in space), while subshell blocks of the same atom share the same position. Since different chemical elements have different sets of basis functions, atom blocks will differ in size between atoms of different chemical elements.

Symmetries One key property is the **permutation symmetry** of the set of individual or blocks of pairs. If we change the order of the K_{total} basis functions and their corresponding coefficients, then the MO being expressed is still the same. This is due to the linear combination in Equation 1, which does not care about the order of summing the terms. One choice is therefore to design a model that is *equivariant* to this transformation: permuting the basis and coefficients (or blocks of them) should change the prediction predictably. If we want molecule-level properties, then we typically desire invariance: permutations should not change the predicted property values.

Another important property is the **3D rototranslation symmetry**. Molecules are positioned in 3D space, but oftentimes the orientation and position of the molecule as a whole in the coordinate system is arbitrary; the physics and properties do not change when moving and rotating the molecule.

Last and perhaps most interesting is the **eigenspace symmetry** of MOs. This is due to the fact that each MO is also an eigenvector of the Fock matrix. Thus, there is an arbitrary choice in sign for the eigenvector: if $c_{:,i}$ for a fixed i is an eigenvector, then $-c_{:,i}$ is the same eigenvector. In practice, we want to take a group of frontier MOs as input, so this sign symmetry becomes eigenspace symmetry (Lim et al., 2023): eigenvectors with different eigenvalues exhibit sign symmetry, but there is a free choice of basis in the subspace spanned by the eigenvectors when multiple eigenvalues are the same. In that case, the MOs are called degenerate.

Models How can we design models for this unique combination of properties? There are plenty of existing works in molecular modeling that address *two* of them: permutation and 3D rototranslation symmetry. We identify rotation-equivariant graph neural networks that use *atoms as nodes* and *spherical harmonics as node features* as particularly suitable for MOs, since the MO basis functions are already expressed in terms of spherical harmonics. With the atom block approach, we can embed each block as additional node features of the corresponding atom in the graph. Once embedded appropriately, we can use the existing graph neural network architecture without any modifications. This is a key benefit: advances in such models can be naturally integrated with MO representations.

We use a state-of-the-art architecture, eSEN (Fu et al., 2025), as an example. It uses a number of *spherical channels* (Zitnick et al., 2022) as node features, each of which represents a function on the sphere and is a vector of coefficients for spherical harmonics of $l = 0, \ldots, L_{\max}$. To embed each subshell block of an atom block, we copy the coefficients of the subshell into the part of a spherical channel corresponding to the same spherical harmonics, matching up both l and m. Thus, we end up with the same number of spherical channels as subshell blocks in the atom block. The shape of R_k is encoded implicitly by each channel index being associated to a different set of input weights. We perform this for each MO in the input in order of ascending energy.

There are two problems left: the differing number of subshells depending on the chemical element, and the eigenspace symmetry. To solve the former, we propose to use a separate embedding model for each chemical element. The embedding models have different numbers of input channels depending on the chemical elements, but the same number of output channels. These outputs can then be used to initialize the eSEN node features. To solve the eigenspace symmetry problem, we use the SignNet (Lim et al., 2023) approach to make our embedding models invariant to sign flips for each MO individually: any function f can be made sign-invariant with $\bar{f}(x) = f(x) + f(-x)$. We deliberately omit the full basis invariance here due to the higher complexity in modeling it correctly (Lim et al., 2023) and the fact that DFT is unlikely to produce exactly equal eigenvalues due to numerical errors unless the molecular structure exhibits high degrees of symmetry.

3.2 Cuboid MO representation

Another approach is to represent each molecular orbital (MO) directly in real space, as depicted in Figure 1. We sample the MO values on a regular 3D grid to obtain a voxel cuboid for each MO. This representation avoids committing to a specific analytic expansion of the orbital in a chosen basis, although the computed orbitals still depend on the electronic-structure method and basis set used in the underlying calculation. 3D voxels are a relatively common data domain, so many models for processing them exist. This representation exhibits translation and sign (eigenspace) symmetries, while rotational symmetry is handled approximately in practice due to voxel discretization.

Models We design a simple 3D CNN to process this representation, details in Appendix F. A drawback of standard CNNs is that while they are mostly translation-equivariant, they are not rotation-equivariant. It is difficult to make CNNs rotation-equivariant to the same degree as an approach based on spherical harmonics due to data discretization. Thus, we rely on data augmentation, where the xyz axes can be permuted (3! options) and individually flipped (2^3 options), yielding 48 different augmentations. In some domains, flips may not be allowed because properties can depend on the chirality of the molecule, in which case there are 24 allowed augmentations.

For the eigenspace sign symmetry, we use an approach inspired by canonicalization (Kaba et al., 2023). For each MO cuboid, we create *two* inputs: an absolute value channel, and three sign flip channels (one for each xyz axis). The former takes the absolute value of the cuboid's values. For the latter, we perform edge detection, placing a 1 on a voxel if the sign of the cuboid flips at that location in that axis. Both of these are sign-invariant, which accomplishes the same goal as SignNet with less computation.

3.3 Unified MO representation

A key concern with the direct MO representation is that the embedding model's parameters are not shared across chemical elements, which may lead to worse performance when data is scarce. We can solve this by converting the cuboid representation back to a representation where basis functions are *shared* across chemical elements. We further reduce discretization error by replacing the cuboid's coordinates with DFT integration grid coordinates (similar to Luise et al. (2025)), which are used in DFT software for numerical integration. So, we evaluate each MO at the integration grid coordinates and project back to a shared basis, represented as spherical channels corresponding to different radii. This representation shares the same three symmetries as the direct MO approach. We note that King et al. (2025) develop a representation with some similarities, which we discuss in Appendix A.

Models We do the same as for the direct MO representation: embed coefficients from the unified basis into spherical channels, process with an embedding model, and add them to the eSEN node features. The only difference is that there is a *shared* embedding model across chemical elements.

3.4 MO energies

While the encoding of MO shapes is our primary concern, we can also use the MO energies λ_i to enhance the three representations. We process the scalar energies corresponding to the frontier MOs with a small multilayer perceptron and add this representation to the MO embedding.

4 Dataset: ChromophoreV3-DCM

There is an abundance of chemistry-related datasets in machine learning that uses computational properties from DFT (Zhang et al., 2025, sec. 5.2.9). For example, the popular QM9 dataset (Ramakrishnan et al., 2014) is often used with DFT atom positions as input and DFT molecule properties as output. The problem is that these simulated properties may not match reality – DFT is not a perfect method, with the "best" choice of parameters that most accurately model experimental results being highly dependent on the type of system being analyzed (Mardirossian & Head-Gordon, 2017) and certain types of systems and interactions not being modeled well by DFT (Jensen, 2017b). If we use atom coordinates or MOs from DFT to predict properties from DFT, it may not reflect real improvements, as they could have the same simulation errors. We want to note that this does not mean that these tasks are not useful: predicting results of expensive calculations using cheap calculations as input is still meaningful in saving computation time.

Instead, we propose to focus on *experimental properties*, which are more directly relevant to many practical use cases. While experimental data is hard to obtain in large quantities, these are exactly the domains where property predictors can have the biggest impact in speeding up materials discovery. We create a dataset based on work by Joung et al. (2020; 2024), which collects experimental measurements of 20,247 OLEDs from literature. OLEDs have applications in e.g. display technology, bioimaging dyes, and solar cells. We use four properties: the peak wavelength of the emission spectrum (colour of emitted light), the full-width at half maximum (FWHM) of the emission ("purity" of the colour), and the same peak/FWHM for the absorption spectrum.

We filter the dataset to only use measurements from the same solvent (dichloromethane, the most common with 2,336 samples) and split the data into 10 folds for cross-validation. We note that while the number of samples might seem small, this setting more accurately mirrors real-world use cases where experimental datasets are likely to be even smaller. To avoid potential for data leakage, we ensure that if multiple papers measure the same molecule, the papers are assigned to the same fold. We run DFT geometry optimization in Gaussian09 (Frisch et al., 2009) to obtain atom positions and MOs using B3LYP/6-31G(d,p) and switch to the def2-svp basis set at the optimum.

Table 1: Different MO representations on ChromophoreV3-DCM. Mean absolute error (lower is better), $\text{mean}_{\text{standard error}}$ of 10-fold cross-validation over 5 repeats.

Representation	Emi_{Peak}	Emi_{FWHM}	Abs_{Peak}	Abs_{FWHM}
Constant input	$76.92_{0.29}$	$20.44_{0.02}$	$80.49_{0.37}$	$20.90_{0.30}$
HOMO, LUMO	$42.45_{1.88}$	$20.34_{0.04}$	$43.83_{2.03}$	$19.88_{0.11}$
vS1, vT1	$40.68_{0.69}$	$19.74_{0.16}$	$41.08_{0.45}$	$19.56_{0.29}$
Coulomb	70.12	20.43	67.81	19.94
SPAHM	63.97	19.79	61.58	19.84
SLATM	48.71	18.59	42.90	18.10
Direct MO	$43.40_{0.34}$	$17.63_{0.19}$	$37.88_{0.27}$	$17.59_{0.16}$
Cuboid MO	$34.21_{0.18}$	$17.86_{0.06}$	$29.98_{0.41}$	$17.33_{0.19}$
Unified MO	$\mathbf{32.73}_{0.17}$	$\mathbf{17.54}_{0.19}$	$\mathbf{28.39}_{0.26}$	$\mathbf{16.28}_{0.08}$

Table 2: Ablation of MO input features on ChromophoreV3-DCM. cMO: Cuboid MO, uMO: Unified MO, E: MO energy. Mean absolute error (lower is better), $\text{mean}_{\text{standard error}}$ of 10-fold cross-validation over 5 repeats.

cMO	E	Emi_{Peak}	Emi_{FWHM}	Abs_{Peak}	Abs_{FWHM}
		$76.92_{0.29}$	$20.44_{0.02}$	$80.49_{0.37}$	$20.90_{0.30}$
✓		$49.20_{0.43}$	$18.31_{0.29}$	$45.52_{0.41}$	$17.65_{0.11}$
	✓	$40.57_{0.17}$	$20.36_{0.05}$	$37.92_{0.26}$	$20.10_{0.14}$
✓	✓	$\mathbf{34.21}_{0.18}$	$\mathbf{17.86}_{0.06}$	$\mathbf{29.98}_{0.41}$	$\mathbf{17.33}_{0.19}$

uMO	E	Emi_{Peak}	Emi_{FWHM}	Abs_{Peak}	Abs_{FWHM}
		$46.14_{0.22}$	$17.98_{0.09}$	$40.31_{0.09}$	$17.92_{0.09}$
✓		$38.12_{0.17}$	$16.22_{0.07}$	$35.20_{0.34}$	$\mathbf{15.63}_{0.13}$
	✓	$33.58_{0.13}$	$17.95_{0.05}$	$29.20_{0.13}$	$16.16_{0.13}$
✓	✓	$\mathbf{32.73}_{0.17}$	$\mathbf{17.54}_{0.19}$	$\mathbf{28.39}_{0.26}$	$16.28_{0.08}$

5 Experiments

In the following experiments, we evaluate various aspects of using MOs for property prediction: the effect of the MO representations (subsection 5.1), MO-related features on their own (subsection 5.2), more realistic setups that combine MOs with larger models (subsection 5.3), sample efficiency (subsection 5.4), and lastly general applicability to a different dataset (subsection 5.5). We publish our code and dataset at `https://anonymous.website` (see supplementary material).

5.1 Different representations

We begin by evaluating the effect of the different representation and modeling approaches in section 3 for a prediction task that only takes MO features as input. Direct MO and unified MO are used with eSEN (6.6M parameters), the cMO representation is processed by a convolutional neural network (2.1M parameters). This is compared to a constant baseline and two linear regressions that either use HOMO and LUMO energies from DFT, or the vS1 and vT1 energies from a more expensive time-dependent DFT. We also include three standard electronic structure methods for machine learning as an alternative to the MO information: Coulomb (Rupp et al., 2012), SPAHM (Fabrizio et al., 2022), and SLATM (Huang & von Lilienfeld, 2016). For experiments on ChromophoreV3-DCM, the variance over 10-fold cross-validation would be uninformative because the folds are not i.i.d.; so, we report variance over 5 repeats of different seeds. We list more experimental details in Appendix F.

Results In Table 1, the cuboid MO (cMO) and unified MO (cMO) representations produce results consistently better than the DFT baseline methods. This shows the general feasibility of using MOs over using DFT to predict these experimental properties, as DFT alone is insufficient to accurately predict these properties. Additionally, our methods strongly outperform the electronic structure baselines, showing that our approach of equivariant modeling is significantly stronger in learning from this type of information. Note that the cMO approach does not have knowledge of atom positions, whereas the direct MO and uMO representations do (since they are required to locate the basis functions). We deem the cMO and uMO results sufficiently close for the following experiments.

5.2 Different MO features

Next, we evaluate the effect of the representation of the MO shape compared to the MO energies. The results in Table 2 (full table: Appendix D) show that there is a clear benefit to including each MO-related feature for both cMO and uMO representations. For example, while MO energies contribute the most individually to the Emi_{Peak} and Abs_{Peak} predictions, they are less useful than the MO shape to Emi_{FWHM} and Abs_{FWHM}. In fact, they lead to slight overfitting for uMO when combining both MO and energy features. Here, we see clearly that while uMO performance is slightly better than cMO, this can be attributed to the presence of atom positions: the eSEN model with zeroed MO inputs (but with atom coordinates) performs much better than the convolutional network with zeroed inputs, which does not have access to any knowledge about atoms.

Table 3: Molecule models combined with MO features on ChromophoreV3-DCM. Mean absolute error (lower is better), $\text{mean}_{\text{standard error}}$ of 10-fold cross-validation over 5 repeats.

Model	Emi_{Peak}	Emi_{FWHM}	Abs_{Peak}	Abs_{FWHM}
DimeNet++	$47.27_{0.34}$	$19.37_{0.07}$	$40.71_{0.43}$	$18.41_{0.27}$
DimeNet++ cMO	$\mathbf{31.65}_{0.31}$	$\mathbf{17.16}_{0.30}$	$\mathbf{27.14}_{0.20}$	$\mathbf{16.21}_{0.26}$
eSEN	$45.53_{0.38}$	$18.06_{0.11}$	$40.65_{0.32}$	$18.20_{0.06}$
eSEN cMO	$33.39_{0.31}$	$\mathbf{17.36}_{0.12}$	$\mathbf{28.04}_{0.33}$	$\mathbf{16.33}_{0.17}$
eSEN uMO	$\mathbf{32.61}_{0.12}$	$17.37_{0.12}$	$28.58_{0.06}$	$16.45_{0.17}$
UMA	$40.57_{0.76}$	$17.45_{0.17}$	$34.65_{0.69}$	$17.17_{0.24}$
UMA cMO	$\mathbf{31.65}_{0.45}$	$\mathbf{17.05}_{0.26}$	$\mathbf{26.63}_{0.23}$	$15.99_{0.19}$
UMA uMO	$32.03_{0.28}$	$17.20_{0.16}$	$27.96_{0.18}$	$\mathbf{15.85}_{0.09}$
UMA cMO *with GFN2-xTB MOs*	$34.64_{0.36}$	$17.36_{0.28}$	$30.01_{0.44}$	$16.19_{0.23}$

5.3 Combining MOs with existing models

Here, we evaluate the combination of existing models with MOs. This is the experiment most relevant to how we envision MOs being used: a standard (possibly pretrained) molecular model is enhanced with features derived from MOs. We choose two well-established base models for this. DimeNet++ (Gasteiger et al., 2020a) is a 7.6M parameter model that uses angles between atoms and has been shown to work well on a variety of property prediction tasks. eSEN (Fu et al., 2025) is a state-of-the-art machine learning force fields (MLFF) method. At the time of writing, it is one of the top entries on the matbench-discovery benchmark (Riebesell et al., 2025), which measures both accuracy of MLFF as well as downstream property prediction performance. For eSEN, we use the UMA-S mixture-of-experts version with 150M parameters (6.6M active parameters) (Wood et al., 2025); we will use "eSEN" to refer to the 150M parameter model that we trained from scratch, and "UMA" to refer to the same model that is initialized from the pretrained UMA-S 1.1 weights and finetuned. To combine the respective base model with the cMO model, we concatenate their global (molecule-level) features, followed by a 3-layer MLP as output block. For uMO, we add the uMO features to the eSEN/UMA node features as described in subsection 3.3.

Results Table 3 shows that across all properties and all models, MO features provide consistent improvements over the baseline, including the 150M parameter pretrained UMA model. This demonstrates the potential of using MO features to enhance existing models. The models enhanced with cMO representations are now slightly better than uMO, likely because they are able to access atom positions and types through the base eSEN and UMA models. Looking at the cMO-only and uMO-only results in Table 1, they are already competitive to the baseline models here, even though cMO does not have any knowledge of atoms. The combination of these pure MO models with eSEN and UMA improves even further, which shows that the electronic information within MOs is *complementary* to the atom positions. In the last row, we show that when MOs originate from the lower-quality but more scalable GFN2-xTB, we still observe performance improvements over the UMA baseline, with the gap being smaller than with the higher-quality DFT MOs. This confirms that our method is robust to different sources of MOs.

5.4 Sample efficiency in low data regimes

We continue with measuring the benefit of MO features in a small-sample scenario. A key motivation for introducing these extra MO features is to improve in settings where data is hard to acquire in large quantities. To simulate this, we still perform 10-fold cross-validation as in all previous experiments on ChromophoreV3-DCM, but use only a small fraction of the training examples in each fold. We vary this fraction from 32% (∼673 training samples per fold) down to 1% (only ∼21 training samples per fold). We focus on the UMA model here because it is pretrained – in settings where data is this scarce, we expect the finetuning of foundation models to be most representative and realistic. We combine it with cMO due to its good performance on the previous experiment.

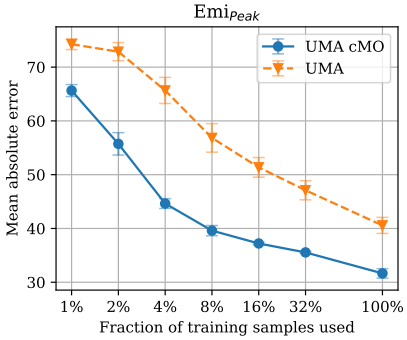

Figure 2: Low data regime experiments on ChromophoreV3-DCM. 1% of the data corresponds to only 21 training samples. Error bars show 95% confidence interval from 5 random seeds.

Table 4: Mean absolute error on QM9 properties. Results including uMO in Appendix D.

Property	μ	α	$\langle R^2 \rangle$	ZPVE	U_0	U	H	G	c_v
Unit	D	a^3	${a_0}^2$	meV	meV	meV	meV	meV	$\frac{\text{cal}}{\text{mol K}}$
DimeNet++	0.0404	0.0732	1.694	2.53	25.35	25.62	25.63	25.16	0.0305
DimeNet++ cMO	**0.0272**	**0.0648**	**1.299**	**2.09**	**22.09**	**22.22**	**22.30**	**21.81**	**0.0293**
eSEN	0.0172	0.0530	1.074	2.70	23.19	23.44	23.68	23.49	0.0260
eSEN cMO	**0.0140**	**0.0472**	**0.881**	**2.19**	**19.57**	**19.68**	**19.74**	**19.37**	**0.0239**
UMA	0.0162	0.0479	1.093	2.37	19.90	20.13	20.23	20.06	0.0244
UMA cMO	**0.0131**	**0.0459**	**0.751**	**2.01**	**16.75**	**16.91**	**16.93**	**16.98**	**0.0224**

Results Figure 2 shows that including cMOs substantially improves the performance of UMA when data is scarce (full numerical results in Appendix D). The performance at only 1% / 21 samples of training data when using MOs is similar to that of the model without MOs at 4% / 84 samples. At 4% of data, the MO version beats the non-MO UMA at 32%. In practice, when each sample could take thousands of dollars and months to acquire, these are substantial improvements, especially given that MO data is often cheaply available when atom positions are being used – which geometric deep learning models already assume. When acquiring samples is expensive, the cost of performing computational chemistry methods like DFT is negligible and standard practice. Even when molecular systems are too large for DFT (i.e. 1000+ atoms), more scalable semi-empirical methods can be used.

5.5 Simulated properties on the QM9 dataset

Last but not least, we perform experiments on the well-established QM9 dataset (Ramakrishnan et al., 2014), which contains 134k molecules and a variety of property types (electronic, energetic, and thermodynamic) to predict. The atom coordinates and MOs come from DFT geometry optimization (gradients of energy), whereas the energetic and thermodynamic properties require a much more expensive vibrational analysis (Hessian of energy). The meaningfulness of the task comes from this prediction of expensive DFT properties with only cheap DFT quantities as input. We obtain the MOs from an eigendecomposition of the Hamiltonians computed in QH9 (Yu et al., 2024).

Results Table 4 shows that including MOs improves each base model on *every* property. This shows that MOs are not only useful for a specific type of domain (ChromophoreV3-DCM), but provide general improvements on a variety of types of properties. It is also clear that certain properties benefit more from MOs than others, with heat capacity c_v improving the least (average of 6.7% relative improvement) compared to others like electronic spatial extent $\langle R^2 \rangle$ (24% relative improvement), which matches the fact that MO encodes electronic information.

6 Discussion

Molecular orbitals are a promising modality to improve the performance of any molecule property predictor that uses atom coordinates already, which includes most state-of-the-art molecular models. They are readily obtained from existing computational chemistry pipelines performed in many standard molecular benchmarks. We proposed MO representations and performed extensive evaluations to demonstrate their performance in predicting experimental properties, especially when data is scarce.

When coordinates come from a generative model rather than a computational chemistry method, the cost of obtaining MOs with those coordinates becomes a relatively cheap single-point calculation rather than full geometry optimization (or can at least be used to jumpstart the geometry optimization). While the cost of obtaining MOs is nonzero in this case, combining it with cheaper methods like GFN2-xTB or FMO-xTB (solving 20,000+ atom systems in a few minutes on one CPU (Einsele & Mitric, 2026)) means that MOs can also be used for significantly larger systems at faster speeds, potentially enabling more accurate high-throughput virtual screening. In regimes with large systems, the molecule foundation model likely becomes the bottleneck rather than any MO computation.

The unique combination of equivariance properties required also presents interesting research problems. For instance, an interesting extension would be consider the symmetries for open-shell systems, with our current approach limited to closed-shell systems. We believe that cross-disciplinary collaborations will continue to shine in areas where domain knowledge from chemistry helps identify new sources of interesting data, and machine learning helps design models for those new types of data.

AI USE STATEMENT

In this work, the primary use of generative AI tools was to proofread the finished manuscript, once. This identified some minor typos and a few details that were missing. We fixed these issues manually.

When looking for related works, the AI summary that some search engines automatically provide may have surfaced works that we would have otherwise missed. We always made sure to manually check any reference, regardless of source, for the actual relevance based on the underlying publication and added these citations by hand when deemed relevant.

Aside from these two aspects, no generative AI was used at any stage in the writing, nor for any aspect of the experiments and coding, nor for the ideas.

ETHICS STATEMENT

As with any work improving methods for materials discovery, there is a dual-use risk. The advancements in this paper can be applied to domains beneficial to society, such as safer, more effective drugs for medical treatment, more environmentally friendly materials, catalysts for carbon capture, and many other applications useful to society. These are the main use cases that we envision the field embracing. On the other hand, this could also include harmful drugs or technology used for warfare. By developing models targeted at domains where data is scarce, we reduce the development costs of both positive and negative advances. We focus on the relatively benign data from OLED molecules for display technology in this paper, but we acknowledge that our methods can be applied equally to harmful datasets. We believe that it is imperative that researchers focus on data that poses minimal risk to the development of harmful technology.

REPRODUCIBILITY STATEMENT

Beyond the model descriptions, we give a rough description of experiments in the experiments section, give extra details in the appendix, and we will publicly release the attached code for full reproducibility (the current version is in the supplementary material already). Scripts to reproduce the dataset creation (for data splitting and filtering of the publicly available dataset collected by Joung et al. (2024)), as well as scripts containing the exact command line arguments (including random seeds) to run all the experiments are included in the code base. We expect results to be exactly reproducible, up to non-determinism of GPU execution and parallelized data loading. The computed molecular orbitals for ChromophoreV3-DCM will be part of the released dataset.

REFERENCES

Christoph Bannwarth, Sebastian Ehlert, and Stefan Grimme. Gfn2-xtb—an accurate and broadly parametrized self-consistent tight-binding quantum chemical method with multipole electrostatics and density-dependent dispersion contributions. *Journal of Chemical Theory and Computation*, 15 (3):1652–1671, 2019. doi: 10.1021/acs.jctc.8b01176. URL https://doi.org/10.1021/acs.jctc.8b01176. PMID: 30741547.

Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, 2003. doi: 10.1162/089976603321780317.

Jerome A. Berson. Orbital-symmetry-forbidden reactions. *Accounts of Chemical Research*, 5(12):406–414, 1972. doi: 10.1021/ar50060a003. URL https://doi.org/10.1021/ar50060a003.

Daniil A. Boiko, Thiago Reschützegger, Benjamin Sanchez-Lengeling, Samuel M. Blau, and Gabe Gomes. Advancing molecular machine learning representations with stereoelectronics-infused molecular graphs. *Nature Machine Intelligence*, 7(5):771–781, May 2025. ISSN 2522-5839. doi: 10.1038/s42256-025-01031-9. URL https://doi.org/10.1038/s42256-025-01031-9.

Ksenia R. Briling, Yannick Calvino Alonso, Alberto Fabrizio, and Clemence Corminboeuf. Spahm(a,b): Encoding the density information from guess hamiltonian in quantum machine learning representations. *Journal of Chemical Theory and Computation*, 20(3):1108–1117, 2024.

doi: 10.1021/acs.jctc.3c01040. URL https://doi.org/10.1021/acs.jctc.3c01040. PMID: 38227222.

Lixue Cheng, Matthew Welborn, Anders S. Christensen, and III Miller, Thomas F. A universal density matrix functional from molecular orbital-based machine learning: Transferability across organic molecules. *The Journal of Chemical Physics*, 150(13):131103, 04 2019. ISSN 0021-9606. doi: 10.1063/1.5088393. URL https://doi.org/10.1063/1.5088393.

Lixue Cheng, Jiace Sun, and Thomas F. III Miller. Accurate molecular-orbital-based machine learning energies via unsupervised clustering of chemical space. *Journal of Chemical Theory and Computation*, 18(8):4826–4835, 2022. doi: 10.1021/acs.jctc.2c00396. URL https://doi.org/10.1021/acs.jctc.2c00396. PMID: 35858242.

Taco S. Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical CNNs. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=Hkbd5xZRb.

Richard Einsele and Roland Mitric. Fmo-xtb: Fragment molecular orbital method with gfn1-xtb for large-scale quantum-mechanical simulations, 2026. URL https://arxiv.org/abs/2606.28022.

Alberto Fabrizio, Ksenia R. Briling, and Clemence Corminboeuf. Spahm: the spectrum of approximated hamiltonian matrices representations. *Digital Discovery*, 1:286–294, 2022. doi: 10.1039/D1DD00050K. URL http://dx.doi.org/10.1039/D1DD00050K.

M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, G. Scalmani, V. Barone, B. Mennucci, G. A. Petersson, H. Nakatsuji, M. Caricato, X. Li, H. P. Hratchian, A. F. Izmaylov, J. Bloino, G. Zheng, J. L. Sonnenberg, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, T. Vreven, J. A. Montgomery, J. E. Peralta, F. Ogliaro, M. Bearpark, J. J. Heyd, E. Brothers, K. N. Kudin, V. N. Staroverov, R. Kobayashi, J. Normand, K. Raghavachari, A. Rendell, J. C. Burant, S. S. Iyengar, J. Tomasi, M. Cossi, N. Rega, J. M. Millam, M. Klene, J. E. Knox, J. B. Cross, V. Bakken, C. Adamo, J. Jaramillo, R. Gomperts, R. E. Stratmann, O. Yazyev, A. J. Austin, R. Cammi, C. Pomelli, J. W. Ochterski, R. L. Martin, K. Morokuma, V. G. Zakrzewski, G. A. Voth, P. Salvador, J. J. Dannenberg, S. Dapprich, A. D. Daniels, Ö Farkas, J. B. Foresman, J. V. Ortiz, J. Cioslowski, and D. J. Fox. Gaussian 09 Revision A.2, 2009.

Xiang Fu, Brandon M Wood, Luis Barroso-Luque, Daniel S. Levine, Meng Gao, Misko Dzamba, and C. Lawrence Zitnick. Learning smooth and expressive interatomic potentials for physical property prediction. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 17875–17893. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/fu25h.html.

Kenichi Fukui, Teijiro Yonezawa, and Haruo Shingu. A Molecular Orbital Theory of Reactivity in Aromatic Hydrocarbons. *Journal of Chemical Physics*, 20(4):722–725, April 1952. doi: 10.1063/1.1700523.

Johannes Gasteiger, Shankari Giri, Johannes T. Margraf, and Stephan Günnemann. Fast and uncertainty-aware directional message passing for non-equilibrium molecules. In *Machine Learning for Molecules Workshop @ NeurIPS*, 2020a. URL https://arxiv.org/abs/2011.14115.

Johannes Gasteiger, Janek Groß, and Stephan Günnemann. Directional message passing for molecular graphs. In *International Conference on Learning Representations*, 2020b. URL https://openreview.net/forum?id=B1eWbxStPH.

Pierre Hohenberg and Walter Kohn. Inhomogeneous electron gas. *Physical review*, 136(3B):B864, 1964.

Bing Huang and O. Anatole von Lilienfeld. Understanding molecular representations in machine learning: The role of uniqueness and target similarity. *The Journal of Chemical Physics*, 145(16): 161102, 10 2016. ISSN 0021-9606. doi: 10.1063/1.4964627. URL https://doi.org/10.1063/1.4964627.

Friedrich Hund. Zur Deutung der Molekelspektren. IV. *Zeitschrift fur Physik*, 51:759–795, 1928. doi: 10.1007/BF01400239.

Tamara Husch, Jiace Sun, Lixue Cheng, Sebastian J. R. Lee, and III Miller, Thomas F. Improved accuracy and transferability of molecular-orbital-based machine learning: Organics, transition-metal complexes, non-covalent interactions, and transition states. *The Journal of Chemical Physics*, 154(6):064108, 02 2021. ISSN 0021-9606. doi: 10.1063/5.0032362. URL https://doi.org/10.1063/5.0032362.

Frank Jensen. *Introduction to Computational Chemistry 3rd. ed.* Wiley, 3rd edition, 2017a. ISBN 9781118825990.

Frank Jensen. *Introduction to Computational Chemistry 3rd. ed.*, chapter 6.7. Wiley, 3rd edition, 2017b. ISBN 9781118825990.

Joonyoung F. Joung, Minhi Han, Minseok Jeong, and Sungnam Park. Experimental database of optical properties of organic compounds. *Scientific Data*, 7(295), 2020. doi: https://doi.org/10.1038/s41597-020-00634-8.

Joonyoung F. Joung, Minhi Han, Minseok Jeong, and Sungnam Park. DB for chromophore, 10 2024. URL https://figshare.com/articles/dataset/DB_for_chromophore/12045567.

Sékou-Oumar Kaba, Arnab Kumar Mondal, Yan Zhang, Yoshua Bengio, and Siamak Ravanbakhsh. Equivariance with learned canonicalization functions. In *International Conference on Machine Learning*, pp. 15546–15566. PMLR, 2023.

Beom Seok Kang, Vignesh C. Bhethanabotla, Amin Tavakoli, Maurice D. Hanisch, William A. Goddard III, and Anima Anandkumar. Orbitall: A unified quantum mechanical representation deep learning framework for all molecular systems, 2025. URL https://arxiv.org/abs/2507.03853.

Manasa Kaniselvan, Benjamin Kurt Miller, Meng Gao, Juno Nam, and Daniel S. Levine. Learning from the electronic structure of molecules across the periodic table. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=PS1YS8Wv4t.

Daniel S. King, Daniel Grzenda, Ray Zhu, Nathaniel Hudson, Ian Foster, Bingqing Cheng, and Laura Gagliardi. Cartesian equivariant representations for learning and understanding molecular orbitals. *Proceedings of the National Academy of Sciences*, 122(48):e2510235122, 2025. doi: 10.1073/pnas.2510235122. URL https://www.pnas.org/doi/abs/10.1073/pnas.2510235122.

Walter Kohn and Lu Jeu Sham. Self-consistent equations including exchange and correlation effects. *Physical review*, 140(4A):A1133, 1965.

Daniel S. Levine, Muhammed Shuaibi, Evan Walter Clark Spotte-Smith, Michael G. Taylor, Muhammad R. Hasyim, Kyle Michel, Ilyes Batatia, Gábor Csányi, Misko Dzamba, Peter Eastman, Nathan C. Frey, Xiang Fu, Vahe Gharakhanyan, Aditi S. Krishnapriyan, Joshua A. Rackers, Sanjeev Raja, Ammar Rizvi, Andrew S. Rosen, Zachary Ulissi, Santiago Vargas, C. Lawrence Zitnick, Samuel M. Blau, and Brandon M. Wood. The open molecules 2025 (omol25) dataset, evaluations, and models, 2025. URL https://arxiv.org/abs/2505.08762.

He Li, Zun Wang, Nianlong Zou, Meng Ye, Runzhang Xu, Xiaoxun Gong, Wenhui Duan, and Yong Xu. Deep-learning density functional theory hamiltonian for efficient ab initio electronic-structure calculation. *Nature Computational Science*, 2(6):367–377, 2022. ISSN 2662-8457. doi: 10.1038/s43588-022-00265-6. URL https://doi.org/10.1038/s43588-022-00265-6.

Derek Lim, Joshua David Robinson, Lingxiao Zhao, Tess Smidt, Suvrit Sra, Haggai Maron, and Stefanie Jegelka. Sign and basis invariant networks for spectral graph representation learning. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Q-UHqMorzil.

Stiv Llenga and Ganna Gryn'ova. Matrix of orthogonalized atomic orbital coefficients representation for radicals and ions. *The Journal of Chemical Physics*, 158(21):214116, 06 2023. ISSN 0021-9606. doi: 10.1063/5.0151122. URL https://doi.org/10.1063/5.0151122.

Giulia Luise, Chin-Wei Huang, Thijs Vogels, Derk P. Kooi, Sebastian Ehlert, Stephanie Lanius, Klaas J. H. Giesbertz, Amir Karton, Deniz Gunceler, Megan Stanley, Wessel P. Bruinsma, Lin Huang, Xinran Wei, José Garrido Torres, Abylay Katbashev, Rodrigo Chavez Zavaleta, Bálint Máté, Sékou-Oumar Kaba, Roberto Sordillo, Yingrong Chen, David B. Williams-Young, Christopher M. Bishop, Jan Hermann, Rianne van den Berg, and Paola Gori-Giorgi. Accurate and scalable exchange-correlation with deep learning, 2025. URL https://arxiv.org/abs/2506.14665.

Narbe Mardirossian and Martin Head-Gordon. Thirty years of density functional theory in computational chemistry: an overview and extensive assessment of 200 density functionals. *Molecular Physics*, 115(19):2315–2372, 2017. doi: 10.1080/00268976.2017.1333644. URL https://doi.org/10.1080/00268976.2017.1333644.

Robert S. Mulliken. The assignment of quantum numbers for electrons in molecules. i. *Phys. Rev.*, 32: 186–222, Aug 1928. doi: 10.1103/PhysRev.32.186. URL https://link.aps.org/doi/10.1103/PhysRev.32.186.

F. Neese. The orca program system. *Wiley Interdiscip. Rev. Comput. Mol. Sci.*, 2(1):73–78, 2012. doi: 10.1002/wcms.81.

Zhuoran Qiao, Matthew Welborn, Animashree Anandkumar, Frederick R. Manby, and Thomas F. Miller III. Orbnet: Deep learning for quantum chemistry using symmetry-adapted atomic-orbital features. *The Journal of Chemical Physics*, 153(12):124111, 09 2020. ISSN 0021-9606. doi: 10.1063/5.0021955. URL https://doi.org/10.1063/5.0021955.

Zhuoran Qiao, Anders S. Christensen, Matthew Welborn, Frederick R. Manby, Anima Anandkumar, and Thomas F. Miller. Informing geometric deep learning with electronic interactions to accelerate quantum chemistry. *Proceedings of the National Academy of Sciences*, 119(31): e2205221119, 2022. doi: 10.1073/pnas.2205221119. URL https://www.pnas.org/doi/abs/10.1073/pnas.2205221119.

Raghunathan Ramakrishnan, Pavlo Dral, Matthias Rupp, and O. Anatole von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules, Jul 2014. URL https://springernature.figshare.com/collections/Quantum_chemistry_structures_and_properties_of_134_kilo_molecules/978904/5.

Janosh Riebesell, Rhys E. A. Goodall, Philipp Benner, Yuan Chiang, Bowen Deng, Gerbrand Ceder, Mark Asta, Alpha A. Lee, Anubhav Jain, and Kristin A. Persson. A framework to evaluate machine learning crystal stability predictions. *Nature Machine Intelligence*, 7:836–847, 2025. doi: https://doi.org/10.1038/s42256-025-01055-1.

Matthias Rupp, Alexandre Tkatchenko, Klaus-Robert Müller, and O. Anatole von Lilienfeld. Fast and accurate modeling of molecular atomization energies with machine learning. *Phys. Rev. Lett.*, 108:058301, Jan 2012. doi: 10.1103/PhysRevLett.108.058301. URL https://link.aps.org/doi/10.1103/PhysRevLett.108.058301.

K. T. Schütt, M. Gastegger, A. Tkatchenko, K.-R. Müller, and R. J. Maurer. Unifying machine learning and quantum chemistry with a deep neural network for molecular wavefunctions. *Nature Communications*, 10:5024, 2019. doi: https://doi.org/10.1038/s41467-019-12875-2.

Qiming Sun, Timothy C. Berkelbach, Nick S. Blunt, George H. Booth, Sheng Guo, Zhendong Li, Junzi Liu, James D. McClain, Elvira R. Sayfutyarova, Sandeep Sharma, Sebastian Wouters, and Garnet Kin-Lic Chan. Pyscf: the python-based simulations of chemistry framework. *WIREs Computational Molecular Science*, 8(1):e1340, 2018. doi: https://doi.org/10.1002/wcms.1340. URL https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/wcms.1340.

Attila Szabo and Neil S Ostlund. *Modern quantum chemistry: introduction to advanced electronic structure theory*. Courier Corporation, 2012.

Oliver Thorsten Unke, Mihail Bogojeski, Michael Gastegger, Mario Geiger, Tess Smidt, and Klaus Robert Muller. SE(3)-equivariant prediction of molecular wavefunctions and electronic densities. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=auGY2UQfhSu.

Matthew Welborn, Lixue Cheng, and Thomas F. III Miller. Transferability in machine learning for electronic structure via the molecular orbital basis. *Journal of Chemical Theory and Computation*, 14(9):4772–4779, 2018. doi: 10.1021/acs.jctc.8b00636. URL https://doi.org/10.1021/acs.jctc.8b00636. PMID: 30040892.

Brandon M. Wood, Misko Dzamba, Xiang Fu, Meng Gao, Muhammed Shuaibi, Luis Barroso-Luque, Kareem Abdelmaqsoud, Vahe Gharakhanyan, John R. Kitchin, Daniel S. Levine, Kyle Michel, Anuroop Sriram, Taco Cohen, Abhishek Das, Ammar Rizvi, Sushree Jagriti Sahoo, Zachary W. Ulissi, and C. Lawrence Zitnick. Uma: A family of universal models for atoms, 2025. URL https://arxiv.org/abs/2506.23971.

Haiyang Yu, Zhao Xu, Xiaofeng Qian, Xiaoning Qian, and Shuiwang Ji. Efficient and equivariant graph networks for predicting quantum hamiltonian. In *International Conference on Machine Learning*, pp. 40412–40424. PMLR, 2023. URL https://proceedings.mlr.press/v202/yu23i.html.

Haiyang Yu, Meng Liu, Youzhi Luo, Alex Strasser, Xiaofeng Qian, Xiaoning Qian, and Shuiwang Ji. Qh9: A quantum hamiltonian prediction benchmark for qm9 molecules, 2024. URL https://arxiv.org/abs/2306.09549.

Xuan Zhang, Limei Wang, Jacob Helwig, Youzhi Luo, Cong Fu, Yaochen Xie, Meng Liu, Yuchao Lin, Zhao Xu, Keqiang Yan, Keir Adams, Maurice Weiler, Xiner Li, Tianfan Fu, Yucheng Wang, Alex Strasser, Haiyang Yu, YuQing Xie, Xiang Fu, Shenglong Xu, Yi Liu, Yuanqi Du, Alexandra Saxton, Hongyi Ling, Hannah Lawrence, Hannes Stärk, Shurui Gui, Carl Edwards, Nicholas Gao, Adriana Ladera, Tailin Wu, Elyssa F. Hofgard, Aria Mansouri Tehrani, Rui Wang, Ameya Daigavane, Montgomery Bohde, Jerry Kurtin, Qian Huang, Tuong Phung, Minkai Xu, Chaitanya K. Joshi, Simon V. Mathis, Kamyar Azizzadenesheli, Ada Fang, Alán Aspuru-Guzik, Erik Bekkers, Michael Bronstein, Marinka Zitnik, Anima Anandkumar, Stefano Ermon, Pietro Liò, Rose Yu, Stephan Günnemann, Jure Leskovec, Heng Ji, Jimeng Sun, Regina Barzilay, Tommi Jaakkola, Connor W. Coley, Xiaoning Qian, Xiaofeng Qian, Tess Smidt, and Shuiwang Ji. Artificial intelligence for science in quantum, atomistic, and continuum systems. *Foundations and Trends in Machine Learning*, 18(4):385–849, 07 2025. ISSN 1935-8237. doi: 10.1561/2200000115. URL https://doi.org/10.1561/2200000115.

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 (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=5Z3GURcqwT.

A Related Work

CEONet The closest work to ours is the concurrent research of King et al. (2025), which proposes a model that takes individual molecular orbitals as input. We consider this the pioneering work in using molecular orbitals in geometric deep learning. Their representation is most similar to the unified MO representation (subsection 3.3). However, there is an important difference in that they represent functions with a Cartesian basis, while we utilize the spherical harmonic basis. As the current state-of-the-art modeling approaches are based on the theory of spherical harmonics, we believe that our representation fits into these existing models better.

There is also an important difference in problem setup: they are concerned with making *orbital-level predictions* such as whether an individual MO is bonding or antibonding (manually labeled), or the energy and entropy of that MO (obtained from DFT). On the other hand, we are interested in making *molecule-level* property predictions, primarily for experimental properties. That is why we use multiple MOs as input, not a single MO at a time, which also changes the type of equivariance required. We believe that our setting better matches realistic practical usage: it is clear that improving the prediction of experimental properties is beneficial, whereas inferring properties of individual MOs is less directly relevant to concrete applications.

An important scientific aspect of our work is that we precisely characterize the effects of different MO representations and MO-related features (shape versus energy) in our experiments. King et al. (2025) instead propose a single method using MOs, so it is not possible to separate the contribution of their model architecture from the contribution of using MOs in general. The benefit of our work is that it integrates easily within existing state-of-the-art approaches for molecular modeling. Regardless, we acknowledge that we are not the first to combine molecular orbitals with geometric deep learning. However, we believe that we make a significant contribution in terms of generality and applicability of results for more widespread use in the field of molecular modeling.

MOB-ML There are a number of methods that derive features from MO-related inputs, which are then processed using various (mostly traditional) machine learning techniques. The key difference is that these typically use hand-designed features, whereas we take the deep learning approach of letting the model do more of the feature learning with a less-processed input. MOB-ML (molecular orbital-based machine learning) (Welborn et al., 2018; Cheng et al., 2019; Husch et al., 2021; Cheng et al., 2022) specifically addresses the problem of using features derived from molecular orbitals to predict *correlation energies* from post-Hartree-Fock methods (a class of methods usually significantly more expensive than Kohn-Sham DFT). This cost motivates the use of machine learning to bridge the gap between Hartree-Fock and post-Hartree-Fock.

MOB-ML methods typically rely on localized MOs, a unitary transformation of the canonical MOs we use. The task is then to predict a correlation energy component for each pair of localized MOs, for which a variety of traditional machine learning techniques can be used. This approach is different from ours in that it targets correlation energies specifically (which admit this decomposition as a sum over pairwise MO energies), which does not apply to general property prediction setups.

SPAHM (Fabrizio et al., 2022; Briling et al., 2024) uses the density matrix constructed from the occupied MOs as input. From this, they compute atom-localized densities, from which they derive rotation-invariant features for each atom. Their density matrix comes from the initial guess of the Fock matrix, similar to MAOC (Llenga & Gryn'ova, 2023). This is something we could use with our model as well if we have a setup where SCF iterations are not performed, with the likely cost of less informative MOs. MAOC is another method that uses a specific localization method to assign densities to atoms. SIMG (Boiko et al., 2025) uses information from Natural Bond Orbital analysis of the converged Fock matrix to construct augmented molecular graphs with new nodes and invariant features. A common thread among these approaches is that they localize the density to obtain atom- or bond-specific features, with many different localization methods having been developed. In our setup, while the direct MO and unified MO approaches represent the MOs in terms of atom features, there is no need for an explicit localization method; we use the canonical MOs from the DFT directly. However, it would be interesting to consider the application of localized MOs in our setup – one drawback is that localized MOs usually no longer correspond to explicit energies, which we found to be helpful for predicting certain properties.

OrbNet OrbNet (Qiao et al., 2020; 2022; Kang et al., 2025) forms another line of related work. They propose to use various DFT-related matrices (Fock, core Hamiltonian, density, and overlap matrices). This is opposed to the much smaller set of eigenvectors that we make use of. Since our eigenvectors are derived from the Fock and overlap matrix, their method uses a superset of the information that we do. For example, if there are 500 atomic orbitals in a molecule, we use six 500d vectors (3,000d in total) corresponding to the six frontier MOs, while they take four 500x500 matrices (1,000,000d) as input. We believe that a benefit of focusing on the frontier MOs is their known chemical relevance and the greater ease of working with data from the smaller size, which enables our method to be easily combined with existing models. While their method is applied to molecule-level properties, their experiments are limited to simulated data. A key aspect of our work is demonstrating the utility of these features for experimental properties. Without this, it would be unclear whether the extra information only helps because the simulated input data contains spurious information about other simulated properties, but not experimental properties.

DFT quantities as output There is a separate line of work using DFT information that aims to *predict* the Fock matrix (so indirectly the MOs) given the atom types and positions (Schütt et al., 2019; Unke et al., 2021; Li et al., 2022; Yu et al., 2023). These either aim to speed up DFT calculations by initializing Equation 3 with a good guess for the Fock matrix, or in the case of (Kaniselvan et al., 2026), be used for improved representation learning compared to predicting the forces and total energy only, as is usually done in MLFF pretraining. The latter in particular shows that the Fock matrix contains plenty of salient information that we can utilize. We recommend the survey of Zhang et al. (2025) for a more in-depth introduction, including coverage of other modalities in AI4Science.

Eigenvectors Lastly, eigenvector inputs have also been used for positional embeddings of generic graphs (Belkin & Niyogi, 2003; Lim et al., 2023). These are typically eigenvectors of the graph Laplacian or other matrices related to the graph structure, and therefore do not have the same block permutation and 3D rotation symmetries as our MO coefficients.

B Further details on basis set definition

Continuing from the main text, the basis set specifies for each chemical element a set of pairs, consisting of radial basis R and angular momentum $l = 0, 1, \ldots$. Depending on the l, there are a different number of spherical harmonic components indexed by m, all with the same shared radial basis R. Whenever a particular l is included by the basis set, the corresponding m takes values from $-l, \ldots, l$. This is referred to as an *orbital subshell*. For example, the basis set can specify one parametrization of the radial basis R and $l = 2$, which results in five basis functions that use this R with spherical harmonics from $Y_{2,-2}$ to $Y_{2,2}$. In Equation 2, we abuse some notation to re-index these l and m combinations under the common index k; e.g. $k = 1, 2, 3, 4, \ldots$ could correspond to $(l, m) = (0, 0), (1, -1), (1, 0), (1, 1), \ldots$ with $k = 2, 3, 4$ sharing the same function R. While R is parametrized further with fixed coefficients, they are unimportant here for our purposes.

In the direct MO representation, each R corresponds to a different channel. Therefore, the model can learn their different effects directly without explicit knowledge of the exact shape. For the cuboid and unified MO representations, R is used during the integration grid calculation and does not enter explicitly into the model.

C Representation details

In this section, we specify the representations of section 3 more formally. We define the ith molecular orbital Ψ_i with the help of a function $\gamma_k : \mathbb{R}^3 \to \mathbb{R}^3$ that converts a global 3D position $p = (x, y, z)$ to spherical coordinates that are centered at the position of the kth atomic orbital.

$$\Psi_i(p) = \sum_{k=1}^{K_{\text{total}}} c_{k,i} \phi_k(\gamma_k(p)) \tag{4}$$

We use F to denote the set of frontier orbital indices for a molecule; in our paper, we fix $|F|$ to 6.

C.1 DIRECT MO

In the following, we describe the atom block version. Assuming AO basis functions are indexed sequentially by atom, let $s(a)$ denote the lowest basis function index that is centered on atom index a. The number of basis functions associated with an atom index a is $K_a = s(a+1) - s(a)$. Then, the representation $A_a^{\mathrm{dMO}} \in \mathbb{R}^{|F| \times K_a}$ of atom index a across frontier MOs $i \in F$ is defined as:

$$(A_a^{\mathrm{dMO}})_{i,k} = c_{s(a)+k,i} \quad \text{for } 0 \leq k < K_a \tag{5}$$

Note that different chemical elements can have a different number K_a of basis functions.

C.2 CUBOID MO

The cuboid grid is defined as a regular 3D grid with indices (u, v, w): $0 \leq u < G_u, 0 \leq v < G_v, 0 \leq w < G_w$. We use $t(u, v, w)$ to map from grid indices to 3D Cartesian coordinates. We evaluate the molecular orbitals Ψ_i on those coordinates to obtain representation $A^{\mathrm{cMO}} \in \mathbb{R}^{|F| \times G_u \times G_v \times G_w}$ across frontier MOs $i \in F$:

$$(A^{\mathrm{cMO}})_{i,u,v,w} = \Psi_i(t(u, v, w)) \tag{6}$$

C.3 UNIFIED MO

Let q be a pre-defined function from N 3D atom coordinates to a set of Cartesian quadrature integration coordinates $\{p_1, \ldots, p_Q\} \subset \mathbb{R}^3$ and associated integration weights $\{w_1, \ldots, w_Q\} \subset \mathbb{R}$. Let $\{\phi_{a,k}^{\mathrm{shared}}\}_{k=1}^{K_{\mathrm{shared}}}$ denote a set of spherical harmonic basis functions centered on atom a. The unified representation $A_a^{\mathrm{uMO}} \in \mathbb{R}^{|F| \times K_{\mathrm{shared}}}$ for an atom index a is obtained by evaluating Ψ_i on the integration grid coordinates and projecting onto each ϕ^{shared}.

$$(A_a^{\mathrm{uMO}})_{i,k} = \sum_{q=1}^{Q} w_q \phi_{a,k}^{\mathrm{shared}}(p_q) \Psi_i(p_q) \tag{7}$$

In our case, ϕ^{shared} is always centered on the corresponding atom, with spherical harmonics and radial function being shared across atoms.

D EXTRA EXPERIMENTAL RESULTS

In this section, we provide additional experimental results.

In Table 5, we perform an extended ablation of the cuboid MO model corresponding to subsection 5.2. We split the MO shape representation into its two components, the absolute value features and sign features (see subsection 3.2). We see that while the absolute value provides more informative features on its own, the sign features still provide a minor contribution to overall performance.

Figure 3 corresponds to the sample efficiency experiment of subsection 5.4. In it, we show the performance on the other properties aside from $\mathrm{Emi}_{\mathrm{Peak}}$. In Table 6, we provide a tabular version of the same data for easier comparability. We see that for $\mathrm{Abs}_{\mathrm{Peak}}$, it follows similar trends as $\mathrm{Emi}_{\mathrm{Peak}}$ with the cMO-enhanced model vastly outperforming the baseline. The story is a bit more complicated for the FWHM properties. Overall, cMO performs slightly better in general, though $\mathrm{Emi}_{\mathrm{FWHM}}$ on 1% and 2% of data are worse than the baseline. We believe this to be an effect of including MO energies (which increased overfitting in other experiments as well) and some random variation from the extremely small training set size.

In Table 7, we present the full results on QM9, specifically including the uMO variants of the models. In contrast to before, UMA uMO performs worse than base UMA and even eSEN with or without uMO features. We believe that this is due to the addition of the MO features disturbing the existing learned node initialization of UMA. This is in comparison to eSEN cMO vs eSEN uMO, where depending on the property, one of the two is superior. In either case, the cMO results always outperform the base model results for all of the three base models, which is why we focusd on these numbers in the main text.

Table 5: Contribution of different MO input features on ChromophoreV3-DCM for cuboid MO, split into absolute channel (A), sign-flip channels (S), and MO energies (E). Mean absolute error (lower is better), the results shown are the average of 10-fold cross-validation.

A	S	E	Emi_{Peak}	Emi_{FWHM}	Abs_{Peak}	Abs_{FWHM}
			$76.92_{0.29}$	$20.44_{0.02}$	$80.49_{0.37}$	$20.90_{0.30}$
		✓	$40.57_{0.17}$	$20.36_{0.05}$	$37.92_{0.26}$	$20.10_{0.14}$
	✓		$58.81_{0.38}$	$21.26_{0.34}$	$51.51_{0.52}$	$20.34_{0.15}$
✓			$49.99_{0.48}$	$18.80_{0.34}$	$47.20_{0.27}$	$18.51_{0.14}$
	✓	✓	$38.34_{0.12}$	$20.24_{0.12}$	$33.81_{0.20}$	$18.74_{0.28}$
✓		✓	$34.40_{0.21}$	$18.04_{0.05}$	$30.73_{0.28}$	$17.48_{0.15}$
✓	✓		$49.20_{0.43}$	$18.31_{0.29}$	$45.52_{0.41}$	$17.65_{0.11}$
✓	✓	✓	$34.21_{0.18}$	$17.86_{0.06}$	$29.98_{0.41}$	$17.33_{0.19}$

Table 6: Sample efficiency results of Figure 2 and Figure 3 in tabular form. Mean absolute error (lower is better), the results shown are the average of 5 repeats of 10-fold cross-validation.

	Data	Emi_{Peak}	Emi_{FWHM}	Abs_{Peak}	Abs_{FWHM}
UMA	100%	$40.57_{0.76}$	$17.45_{0.17}$	$34.65_{0.69}$	$17.17_{0.24}$
UMA cMO	100%	$31.65_{0.45}$	$17.05_{0.26}$	$26.63_{0.23}$	$15.99_{0.19}$
UMA	32%	$47.10_{0.91}$	$18.97_{0.21}$	$41.74_{1.09}$	$19.28_{0.18}$
UMA cMO	32%	$35.55_{0.09}$	$18.63_{0.17}$	$30.29_{0.49}$	$17.49_{0.13}$
UMA	16%	$51.36_{0.93}$	$19.68_{0.13}$	$45.97_{1.05}$	$20.51_{0.17}$
UMA cMO	16%	$37.21_{0.14}$	$19.06_{0.21}$	$31.89_{0.40}$	$18.50_{0.11}$
UMA	8%	$56.84_{1.35}$	$20.12_{0.30}$	$51.95_{1.29}$	$20.73_{0.22}$
UMA cMO	8%	$39.60_{0.49}$	$19.71_{0.31}$	$35.29_{0.86}$	$19.25_{0.17}$
UMA	4%	$65.66_{1.24}$	$20.38_{0.31}$	$60.95_{1.48}$	$21.13_{0.44}$
UMA cMO	4%	$44.63_{0.47}$	$20.11_{0.25}$	$39.88_{0.55}$	$19.97_{0.48}$
UMA	2%	$72.87_{0.86}$	$20.28_{0.19}$	$71.86_{1.39}$	$20.98_{0.14}$
UMA cMO	2%	$55.73_{1.06}$	$21.35_{0.20}$	$53.87_{1.17}$	$20.84_{0.56}$
UMA	1%	$74.26_{0.51}$	$20.71_{0.21}$	$75.24_{0.69}$	$21.15_{0.24}$
UMA cMO	1%	$65.64_{0.57}$	$21.66_{0.28}$	$65.45_{1.08}$	$21.40_{0.43}$

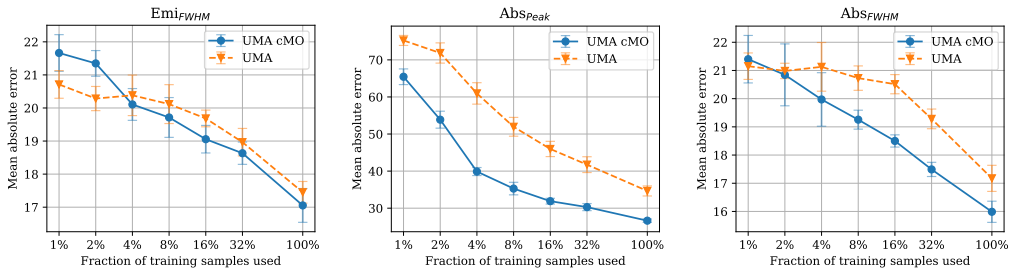

Figure 3: Low data regime experiments on ChromophoreV3-DCM. Error bars show 95% confidence interval from 5 runs.

Table 7: Mean absolute error on QM9 properties. Models are trained in a multi-task setting.

Property Unit	μ D	α a^3	$\langle R^2 \rangle$ ${a_0}^2$	ZPVE meV	U_0 meV	U meV	H meV	G meV	c_v $\frac{\text{cal}}{\text{mol K}}$
DimeNet++	0.0404	0.0732	1.694	2.53	25.35	25.62	25.63	25.16	0.0305
DimeNet++ cMO	**0.0272**	**0.0648**	**1.299**	**2.09**	**22.09**	**22.22**	**22.30**	**21.81**	**0.0293**
eSEN	0.0172	0.0530	1.074	2.70	23.19	23.44	23.68	23.49	0.0260
eSEN cMO	**0.0140**	**0.0472**	**0.881**	2.19	19.57	19.68	19.74	19.37	**0.0239**
eSEN uMO	0.0176	0.0488	0.958	**1.97**	**19.00**	**19.14**	**19.17**	**19.16**	0.0254
UMA	0.0162	0.0479	1.093	2.37	19.90	20.13	20.23	20.06	0.0244
UMA cMO	**0.0131**	**0.0459**	**0.751**	**2.01**	**16.75**	**16.91**	**16.93**	**16.98**	**0.0224**
UMA uMO	0.0195	0.0577	1.072	2.85	25.41	25.76	25.95	25.33	0.0280

E Comparability of results on QM9

There are a number of differences that limit comparability to existing results in literature for this dataset. Since the purpose of this experiment is not to demonstrate state-of-the-art results, but to show that including MO features can improve any base model, this does not detract from our results. The following explanation is provided for completeness, to explain why our results are somewhat "worse" on an absolute level than existing numbers in the literature.

There are several differences that limit comparability. First of all, we use additional data (coming from MOs) that existing models do not. Even if this additional data is easy to obtain, it is still extra data and thus not a like-for-like comparison. Another key difference is that we perform multi-task training while Gasteiger et al. (2020b) shows that training single-task models on each property individually performs much better. This also means that during training, we have a particular weighting of loss components determined by the scaling of the individual properties. To reproduce their results, we would require 9x the training cost by training on each property separately, which we do not believe to be meaningful with regards to the message that this experiment is intended to provide. All the absolute numbers would be better, but we expect the relative ordering to remain the same, which would mirror existing DimeNet results.

Next, we train for much fewer steps with different learning rate schedules. The results of DimeNet++ (Gasteiger et al., 2020a) in their paper used a budget of 3,000,000 steps with early stopping at a batch size of 32 (90M samples seen) for every property separately. Meanwhile, our DimeNet++ reproduction is trained for 981,300 steps (300 epochs at batch size 32, 31M samples seen), not using early stopping and a single model for all properties at once. This is also in comparison to our training runs for the eSEN and UMA models, which use 1,308,400 steps (100 epochs at batch size 8, 10M samples seen). Again, we note that reproducing the DimeNet++ exactly is not the point here, especially for the non-DimeNet models of eSEN and UMA. We emphasize that all results in our table use the same setup, which makes them comparable without issue.

We note there is an issue with the dataset in comparison to non-DimeNet results from the literature. Due to an off-by-one error in DimeNet data processing, the set of molecules included is slightly different to the original set of QM9 data. We note that QH9 and DimeNet++ use this version with the off-by-one error. To be clear, this means that a small percentage of samples is different from the original QM9, not that labels are incorrect.

Lastly, we did not take care to use exactly the same random dataset split into train, validation, and test sets as existing results.

The reason for these discrepancies is because this experiment is somewhat orthogonal to the main thesis of our paper and we see no need or benefit to exactly reproduce the existing results of DimeNet++ in literature. Our main comparisons between an existing model and the exact same model but using MOs remains fully valid, and our experiments on ChromophoreV3-DCM should be considered the main experiments that are more relevant to the problem setups we are mainly addressing.

F Experimental details

While we provide details here for completeness, we also recommend to consult our reference implementation at `https://anonymous.website` (see supplementary material during peer review).

F.1 Dataset details

In ChromophoreV3-DCM, the largest number of atoms is 150, and the largest number of heavy atoms is 88, which we handle without problems with B3LYP/def2-SVP in DFT. We think that larger systems up to twice the size would still be reasonable with these settings. For larger systems, scalability can be increased by switching to simpler functionals and smaller basis sets if the accuracy of DFT is required. The general maximum range of DFT is considered to be in the few-hundred-atom range with standard consumer-level hardware.

For the QM9 setting, the molecules are relatively small, so geometry optimization and vibrational analysis are expected to take similar amounts of time (e.g. 15 minutes for each); in this case, predicting these properties (with or without MOs) saves those 15 minutes per molecule. With the much larger ChromophoreV3-DCM molecules, vibrational analysis can easily take multiple days per molecule due to its worse scaling behaviour, thus making the time-saving benefits from direct prediction more substantial. Our work shows that doing so using information that is already computed by the preceding geometry optimization improves the prediction quality

Alternatives to DFT While we have focused on MOs obtained from DFT in the paper, there are faster, less accurate techniques that also perform geometry optimization and compute MOs, such as the semi-empirical GFN2-xTB (Bannwarth et al., 2019). It is often used with systems containing over 1,000 atoms. We used it only in one experiment (Table 3) because our applications did not require it; DFT was sufficiently fast, so we preferred the higher accuracy from it. GFN2-xTB takes in the order of 10s of minutes for ChromophoreV3-DCM-sized molecules for full geometry optimization. Meanwhile, (Einsele & Mitric, 2026) is designed for large scale systems beyond 20,000 samples with almost linear scaling behaviour, allowing single-point calculations of such scale to be completed in a few minutes on one CPU.

Experiment with GFN2-xTB In Table 3, we ran GFN2-xTB *on the DFT coordinates* for the purposes of this ablation of evaluating prediction quality under lower quality MOs. The reason is to keep the numbers comparable to the existing ones in the paper: if we had used the xTB-optimized coordinates, it would negatively affect the performance of the base UMA as well due to the less optimized geometry, so it would not be a like-for-like comparison to the existing results in the paper. By using DFT coordinates + xTB MOs, we ensure that the only difference is the quality of MOs – in principle, these coordinates could have come from any method, so this comparison is purely between DFT (B3LYP/def2-svp) MOs versus GFN2-xTB MOs. In practice with large molecular systems where DFT is no longer feasible, we would of course do all the computations with xTB only.

Alternative basis sets We have not tried different basis sets beyond def2-svp and the GFN2-xTB basis set in our experiments. We primarily focus on def2-svp as a reasonable, standard choice for the types of molecular systems we considered. Given that the orbitals are unified in cMO and uMO during preprocessing, we expect that the effect of choice of basis set is relatively minor; the difference in results would be primarily due to the task-specific suitability of the basis set to the experimental property. This choice would primarily be informed by domain experts depending on the type of system and desired properties to be modeled.

Another option that we did not explore is mixing basis sets within the same dataset. Since uMO and cMO project the existing MO information into a different representation, we do not expect any issues with with, since those differences disappear during preprocessing. Only dMO (which had worse results anyway) would be problematic with mixed basis sets. For basis ordering and normalization conventions, these are standardized by the preprocessing so they make no difference to the results for any of the MO representations

F.2 Compute requirements

CPU The main CPU cost arises from the DFT calculations. Geometry optimization, which takes the bulk of the time, takes an average of 8.5 hours per molecule for us on ChromophoreV3-DCM. This can be trivially parallelized across multiple molecules with access to CPU compute infrastructure. This constitutes around 20,000 CPU hours for the creation of our dataset. This is realistically achievable with a modest amount of compute that is available for 200 USD with public cloud computing services, which can be significantly less than the synthesis of just a single molecule.

Other works have computed DFT on a much larger scale than us, showing that this cost is usually not a big problem. For example, the OMol25 dataset (Levine et al., 2025) containing 83 million unique molecular systems with 140 million DFT calculations has larger molecules (up to 350 atoms) and uses the more accurate (much slower) functional of ωB97M-V/def2-TZVPD. They report an average of 50 CPU core-hours per molecule, with a total of 6.6 billion CPU core hours.

Again, we stress that existing molecular models that use molecule geometries already require DFT (or other computational chemistry) computations, so this is a cost that needs to be paid either way,

with or without MOs. As we discussed, in domains where obtaining samples is expensive and time-consuming, this is typically a negligible cost to pay.

Preprocessing The preprocessing once the MOs are obtained is fairly cheap. On a single consumer CPU core without parallelization it takes:

- dMO: 50 milliseconds per ChromophoreV3-DCM molecule (58 MB total)
- cMO: 0.2 seconds per ChromophoreV3-DCM molecule (2000 MB total)
- uMO: 2.3 seconds per ChromophoreV3-DCM molecule (880 MB total)
  - this is easily tunable to be much faster by sampling fewer points to evaluate the wave function on, currently the sampling grid is much denser than cMO

The sizes are the size of the SQLite database file, so they include database-related metadata and other molecule properties such as coordinates. We expect the voxelized cMO representations to scale less well than the spherical harmonics based representations to larger systems, as there will likely be a lot more empty space that would be encoded in the volume. The scaling of the spherical harmonics based representations should be fine as they are linear in the number of atoms (same as the atom coordinates) and has the benefit of being focused on the areas where we expect nontrivial amplitudes of the wave function.

On QM9, we did not keep track of processing speeds, but since molecules are generally much smaller, it is quite a bit faster per molecule as well. cMO uses 25 GB and uMO uses 14 GB.

GPU All experiments were performed on Nvidia L40S GPUs as part of a compute cluster. Each experiment run was performed on a single GPU. In total, we estimate that we used 850 GPU hours for experiments, which does not include preliminary experimentation. We give a rough estimate of 1200 GPU hours in total if we include preliminary experiments.

The direct MO and unified MO models in section 5.1 and section 5.2 takes around 25 minutes each, while the cMO runs take around 12 minutes each.

In 5.3, the base models for DimeNet++, eSEN/UMA take 11 minutes and 48 minutes. With the inclusion of the uMO representation, eSEN/UMA runtime increases to 51 minutes. With the inclusion of cMO features, the runtime increases to 19 minutes and 64 minutes respectively for DimeNet++ and eSEN/UMA.

In 5.4, the 32% experiment takes 19 minutes without cMO and 24 minutes with cMO. The 1% experiment takes 6 minutes without cMO and 6 minutes with cMO (execution time dominated by fixed size validation set).

All of the preceding individual run numbers are multiplied by 50, as we perform 10-fold cross-validation a total of 5 times. This is done because the folds are not distributed i.i.d., so the variance over those 10 folds would have mostly measured the shift in data distribution rather than model variance.

In section 5.5, the runtime is a bit more variable due to experiments being run on slightly varying hardware (but the same GPU type), so there is no clear trend. All of the experiments using eSEN/UMA here take around 48 hours, and using DimeNet++ take around 30 hours.

Model sizes

- base eSEN/UMA: 146,597,284 params
- with uMO representation: 146,848,164 params
- with cMO representation: 151,399,428 params

For the number of active parameters, uMO adds 250k to the 6.6M active parameters of eSEN/UMA, while cMO adds 4.8M.

F.3 Experimental procedure

Most architecture and hyperparameter development was performed on a separate private dataset with no overlap to ChromophoreV3-DCM. The hyperparameter configuration selected on that private dataset was transferred unchanged to ChromophoreV3-DCM and is the configuration used for all reported results.

One ChromophoreV3-DCM fold (out of 10) was used only for a limited qualitative robustness check. We temporarily varied general training choices, such as the learning rate and number of epochs, to verify that the observed advantage of incorporating MOs was not brittle to reasonable perturbations. These experiments were not used to choose the reported configuration: none of the tested changes was retained, and no model choice or hyperparameter used in the reported results was selected or modified based on performance on that fold.

The configuration was then kept fixed and evaluated across all ten cross-validation folds. Within every baseline-versus-MO comparison, the training settings, data splits, and random seeds are identical; the only difference is whether the MO encoder is enabled.

F.4 Convolutional neural network for Cuboid MO

We mostly follow a standard architecture that uses 3D convolutions, SiLU activations, and batch normalization. Our network is split into two halves: the first half processes each of the six MOs independently of each other, the second half combines them as separate channels.

The absolute channel and three sign-flip channels are encoded separately with a one layer CNN and two layer CNN (each uses a stride of 2 to halve the resolution), then added to a representation of the MO energies that was processed by a 2-layer MLP. Once the information from these three sources is combined, we process this with another strided convolution. After this point, we stack the channels of each MO belonging to the same molecule together, followed by a convolution and two residual convolution blocks. Each block has two sequences of BatchNorm, SiLU, and convolution. Lastly, we follow this with average and max pooling of the spatial features and a 3-layer MLP output block with the number of properties to predict as output and ReLU activations.

All convolutions use 3x3x3 convolution filters.

F.5 Embedding model for direct MO and unified MO

We use S^2 activations (Cohen et al., 2018; Zitnick et al., 2022) as embedding model. That is, the spherical functions that the spherical channels represent are sampled on a grid, which is then processed by an MLP on each grid position, which is then converted back to spherical channels.

There is a slight difference in how the direct MO and unified MO are processed in this approach. For direct MO, a 3-layer MLP that takes all 6 frontier MOs as input channels is used on the grid representation, and this MLP applied to the positive and negative sign (following SignNet). The result is converted back to spherical channels, which are used to initialize the eSEN node features. As this is only takes into account the global sign flip symmetry, we additionally perform data augmentation that individually flips the signs of orbitals.

For the unified MO, we keep them separate for slightly longer (like the cuboid MO model) and process their grids separately with a 3 layer MLP first, with positive and negative signs added at the end of the MLP. Then, the different MOs (still in the grid representation) of the same molecule are reorganized to be different channels and transformed by another 2 layer MLP. Lastly, we convert this back to spherical channels to obtain the initialization of eSEN node features. For this approach, sign flip augmentations are not needed.

F.6 Training details and hyperparameters

For the cuboid MO representation, we use a resolution of 0.8 Bohr (each voxel has this side length) with a box margin of 3.0 Bohr around the molecule.

For the unified MO representation, we choose an L_{max} of 4 for ChromophoreV3-DCM and 3 for QM9. For both datasets, we use 16 channels in the unified representation. These correspond to 16

evenly-spaced Gaussians up to a radial distance of 3.0 Bohr – coordinates of the integration grid up to 3.0 Bohr distance to an atom are captured in the 16 spherical channels. For the integration grid, we use the PySCF (Sun et al., 2018) integration grid of the lowest level 0.

All models on the ChromophoreV3-DCM dataset are trained for 50 epochs with a batch size of 16. As mentioned in Appendix E, the DimeNet models on QM9 are trained for 300 epochs with batch size 32, while the eSEN/UMA models are trained for 100 epochs with a batch size of 8 due to GPU memory limitations of these 150M parameter models.

The eSEN and DimeNet++ models use a hidden channel dimension of 128. For eSEN, this is necessitated by the fact that the UMA-S 1.1 weights use this dimensionality. In subsection 5.1 and subsection 5.2, the cMO convolutional neural network has 64 hidden channels, while for the remaining experiments when combined with either DimeNet++ or eSEN/UMA, we increase this to 128.

All of our experiments use the same learning rate schedule using the Adam optimizer. The first 4% of training steps are used to linearly warm up from a learning rate of 6×10^{-5} to 3×10^{-4}, which then follows a cosine schedule to lower to $3 \times 10^{-}6$ at the end of training.

We use an exponential moving average over weights with a momentum of 0.99 to obtain the model used for test set predictions. We apply gradient clipping with a parameter of 0.5. For both ChromophoreV3-DCM and QM9, we apply z-standardization on the output properties during training.