Beyond Atoms: Evaluating Electron Density Representation for 3D Molecular Learning

Patricia Suriana
Prescient Design, Genentech
SSF, USA
surianap@gene.com

Joshua A. Rackers
Prescient Design, Genentech
SSF, USA

Ewa M. Nowara
Prescient Design, Genentech
SSF, USA

Pedro O. Pinheiro
Prescient Design, Genentech
SSF, USA

John M. Nicoloudis
Structural Biology, Genentech
SSF, USA

Vishnu Sresht
Prescient Design, Genentech
SSF, USA

Abstract

Machine learning models for 3D molecular property prediction typically rely on atom-based representations, which may overlook subtle physical information. Electron density maps—the direct output of X-ray crystallography and cryo-electron microscopy—offer a continuous, physically grounded alternative. We compare three voxel-based input types for 3D convolutional neural networks (CNNs): atom types, raw electron density, and density gradient magnitude, across two molecular tasks—protein–ligand binding affinity prediction (PDBbind) and quantum property prediction (QM9). We focus on voxel-based CNNs because electron density is inherently volumetric, and voxel grids provide the most natural representation for both experimental and computed densities. On PDBbind, all representations perform similarly with full data, but in low-data regimes, density-based inputs outperform atom types, while a shape-based baseline performs comparably—suggesting that spatial occupancy dominates this task. On QM9, where labels are derived from Density Functional Theory (DFT) but input densities from a lower-level method (XTB), density-based inputs still outperform atom-based ones at scale, reflecting the rich structural and electronic information encoded in density. Overall, these results highlight the task- and regime-dependent strengths of density-derived inputs, improving data efficiency in affinity prediction and accuracy in quantum property modeling.

1 Introduction

Machine learning (ML) has become an essential component of structure-based small molecule discovery, supporting tasks such as virtual screening, lead optimization, and protein–ligand binding affinity prediction [1, 2]. A critical challenge in these applications is selecting a representation that effectively captures the physical and chemical complexity of molecular systems. Most current methods rely on atom-based features, using atomic coordinates and element types extracted from experimentally resolved or computationally modeled structures.

Although widely used, atom-based representations abstract away important physical details. Atomic coordinates are not measured directly; they are inferred by fitting an atomistic model to experimental electron density maps obtained from X-ray crystallography or cryo-electron microscopy. This model-building process depends on expert interpretation and heuristic refinement procedures, introducing potential bias and error. Structural inaccuracies in public databases such as the Protein Data Bank (PDB) [3] are well documented [4, 5]. Moreover, by discretizing molecules into point-like atoms,

Accepted at the AI for Science workshop (NeurIPS 2025).

these representations ignore the continuous distribution of electron density that governs molecular interactions.

Electron density maps, the direct output of X-ray crystallography and cryo-electron microscopy (cryo-EM), offer a more direct and physically grounded alternative. These maps represent a 3D scalar field describing the spatial distribution of electrons. Unlike fitted atomistic models, they encode both the extent and overlap of electron clouds and inherently capture features such as conformational heterogeneity and structural uncertainty, which manifest as diffuse or mixed density in flexible regions.

Electron density has several properties that make it appealing for ML applications:

- **Directly derived from experiment:** Density maps are obtained directly from physical measurements, bypassing the lossy step of atomic model fitting [6].
- **Continuous encoding of interactions:** Because molecular forces arise from electron distribution, density may enable more physically faithful modeling of interaction strength and geometry.
- **Implicit representation of flexibility and uncertainty:** Conformational variability appears naturally in the map, without additional modeling assumptions.

These characteristics suggest that density-based representations may offer a richer signal for learning molecular properties—particularly those governed by electronic structure and interactions, such as binding affinity and quantum mechanical properties. Intuitively, one might expect this richer information to translate into improved model performance, especially under data-limited conditions.

At the same time, atom-type representations provide strong chemical priors by explicitly labeling atoms with their identities (e.g., C, N, O). These priors embed known chemical patterns and may be especially useful in high-data or chemically diverse regimes. This raises a central question: when do density-based representations offer an advantage over atom-based ones?

To address this, we compare three voxel-based representations for 3D convolutional neural networks (CNNs): atom-type channels, raw electron density, and the gradient magnitude of density. The gradient captures rapid spatial changes in density and may highlight features relevant to chemical interactions, such as bonding regions or non-covalent contacts [7]. We use 3D CNNs because they are well suited for volumetric data, as electron density forms a continuous field in three-dimensional space. Although computed densities (e.g., from XTB or DFT) can be expressed in alternative bases, voxelization provides a unified spatial framework for both experimental and theoretical sources. Because our objective is to benchmark different voxel representations rather than optimize architectures, we focus on CNNs, which can directly process volumetric data, whereas graph- or transformer-based networks [8–11] cannot natively represent continuous 3D density fields. This makes 3D CNNs a natural architectural choice for evaluating volumetric representations.

We evaluate these representations on two tasks: (1) protein–ligand binding affinity prediction using the PDBbind dataset, and (2) quantum property prediction for small molecules using QM9. These tasks differ in physical scale, label origin, and modeling assumptions, providing complementary perspectives. Specifically, we ask: Do density-based inputs improve model performance in low-data settings? Do these benefits persist at scale, even when the input densities are approximate or noisy? By analyzing how representation interacts with data regime and model capacity, we aim to clarify the conditions under which density-derived inputs improve 3D molecular learning.

2 Related Work

Structure-Based Learning with Atomic Coordinates. Most machine learning models for structure-based molecular discovery rely on atomic coordinate–based representations. Graph neural networks (GNNs) encode molecules as graphs with atom and bond features, often extended to include 3D geometric information [8, 9, 12]. Point-cloud models treat molecules as unordered sets of atomic coordinates, requiring networks that are invariant or equivariant to rotation and translation [11, 10]. Voxel-based 3D CNNs, which are directly relevant to this work, project atomic features onto a 3D grid and have been widely applied to pose prediction and binding-affinity scoring in docking pipelines [13, 2, 1]. Unlike graph or point-based models that operate on discrete atomic representations, 3D CNNs can directly process continuous volumetric data such as electron density maps. Because our

goal is to compare representational domains rather than model architectures, we adopt voxel-based CNNs as a consistent framework for learning from density fields. In contrast, GNNs and related architectures are designed for atom- and bond-level inputs and cannot directly handle volumetric electron-density data without new formulations that incorporate such information.

Electron Density in Structural Biology. Electron density maps are not merely alternative inputs for ML but are the fundamental data products of experimental techniques such as X-ray crystallography and cryo-EM, representing the time- and ensemble-averaged spatial distribution of electrons. Structural biologists interpret these maps to build and refine atomic models that best explain the observed density [6]. This process involves fitting atomic templates, often requiring expert knowledge, and can be subjective—particularly in regions of lower resolution or higher flexibility. While essential for generating interpretable models, this step inherently involves assumptions and may lose subtle information present in the raw density. Using the density directly for ML bypasses this modeling step and can preserve more of the original experimental signal.

Machine Learning on Electron Density. Although less common than atom-based approaches, applying machine learning directly to electron density is an emerging area of research. Recent work has explored density maps for generative modeling—for example, Wang et al. [7] introduced a diffusion model that generates ligands conditioned on the electron density of protein pockets and showed that regions of rapid density change may correspond to non-covalent interactions. These findings motivate our use of both raw electron density and its gradient magnitude, which highlight complementary aspects of the electronic environment. However, few controlled studies have systematically compared density-based and atom-based representations across 3D molecular prediction tasks. Our work addresses this gap by directly evaluating both input types on two benchmark molecular prediction tasks.

3 Experiments and Methods

3.1 Binding Affinity Prediction

We assess model performance using the PDBbind v2021 dataset [14], a widely used benchmark for protein-ligand binding affinity prediction, which includes $\sim$20,000 complexes with experimentally measured pK values. Following Pinheiro et al. [15], we voxelize the ligand and its surrounding protein pocket into separate 3D grids, both centered on the ligand's center of mass, and pass them as input to the model (See Supplement A.1 for details).

To ensure proper generalization and avoid data leakage, we split the data using both receptor sequence and ligand similarity. Two complexes are assigned to the same split only if: (1) their receptor sequence identity exceeds 50%, or (2) their receptor sequence identity exceeds 40% and their ligand Tanimoto similarity is above 0.9. To further diversify the test set, all targets similar to those in the DEKOIS 2.0 benchmark [16]—which spans a wide range of protein families—are reserved for the test set. The final split includes approximately 14,258 complexes for training, 1,171 for validation, and 5,554 for testing.

We report Spearman correlation (ρ) on the test set as the primary evaluation metric. This choice reflects real-world applications of binding models, where compound ranking is often more critical than absolute value prediction.

3.1.1 Input Feature Representations

All inputs are voxelized into a $64\times64\times64$ grid at 0.25 Å resolution. For density-based representations, we first resample all experimental electron density maps to a uniform resolution of 0.25 Å, as raw maps vary in resolution across structures. We compare the following four input types:

- `Atom-Type`: A multichannel representation where atom types are encoded as 3D Gaussians centered at atomic coordinates [15]. Ligands use 7 channels (C, O, N, S, F, Cl, P), and protein pockets use 4 channels (C, O, N, S).
- `Shape-Only`: A single-channel baseline where all atoms are treated as carbon, removing chemical identity to focus on shape.

- `Density`: A single-channel grid of experimental 2mFo–DFc electron density values, extracted from crystallographic MTZ files using Phenix [6].
- `GradMag`: A single-channel grid encoding the spatial gradient magnitude of the 2mFo–DFc map, highlighting regions of rapid density change, which may correlate with interaction sites [7].

3.1.2 Model Architectures and Training

To isolate the effect of input representation, we use a consistent family of 3D convolutional networks across experiments. Models are evaluated at three capacity levels:

- `Tiny` ($\sim$0.4M parameters): Based on GNINA's `Default2018Affinity` architecture [17].
- `Small` ($\sim$4M) and `Default` ($\sim$58M): Use the encoder half of the VoxBind 3D U-Net architecture [15], followed by a multi-layer perceptron (MLP) for affinity prediction.

See Supplement A.1 for detailed architectural descriptions.

All models are trained using the Adam optimizer (learning rate 1×10^{-5}), batch size 32, and mean squared error (MSE) loss, for 1000 epochs. Random 3D rotations are applied to voxel inputs during training to promote rotational invariance. Reported results are averaged over three random seeds.

3.1.3 Data Efficiency Evaluation

To assess data efficiency, we train models on increasingly larger subsets of the training set: 1%, 5%, 10%, 25%, 50%, and 100%. This allows us to compare performance trends across input representations and model sizes as data availability increases.

3.2 Quantum Property Prediction

We evaluate our models on the QM9 dataset [18], which contains approximately 134,000 small organic molecules (each with up to 9 heavy atoms), along with quantum chemical properties computed using Density Functional Theory (DFT). We train separate models to predict four scalar regression targets: dipole moment (μ), isotropic polarizability (α), energy of the highest occupied molecular orbital (E_{HOMO}), and energy of the lowest unoccupied molecular orbital (E_{LUMO}). The dataset is randomly split into 80% training, 10% validation, and 10% test sets.

Unlike the PDBbind task, which leverages experimental electron density maps derived from crystallographic data, the QM9 dataset does not contain any experimental structural or density measurements. As a result, we generate approximate electron density maps using the GFN2-xTB semiempirical method via the XTB package [19], based on the molecular geometries provided in the dataset. These computed densities are used as input for our density-based voxel representations.

We use mean absolute error (MAE) as the evaluation metric, following standard practice in prior work [9, 12]. Our goal is not to achieve state-of-the-art accuracy, but to evaluate how different representations affect predictive performance.

3.2.1 Input Representations

QM9 molecules are voxelized into $32 \times 32 \times 32$ grids at 0.25 Å resolution, centered on the molecular center of mass. We use the same four input types as in the binding affinity task (Section 3.1): `Atom-Type`, `Shape-Only`, `Density`, and `GradMag`. The only difference lies in the atom-type representation: for QM9, we use five channels corresponding to C, H, O, N, and F. All other representations use a single channel.

3.2.2 Model Architectures and Training

To study the effect of model capacity, we use three versions of a 3D CNN architecture adapted from VoxMol [20], varying in size: `Tiny` ($\sim$4M parameters), `Small` ($\sim$15M), and `Default` ($\sim$58M). For architectural details, see Supplement A.2.

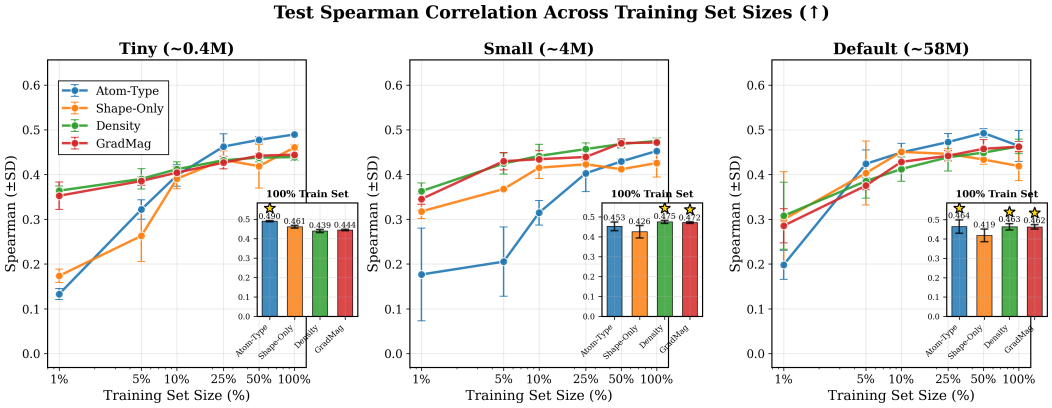

Figure 1: Test Spearman correlation for binding affinity prediction on the PDBbind test set across training set sizes (1%–100%) and model capacities (∼0.4M, ∼4M, ∼58M parameters). We compare four voxel-based input representations: atom types (`Atom-Type`), atoms mapped to carbon (`Shape-Only`), electron density values (`Density`), and electron density gradient magnitude (`GradMag`). Performance generally improves with training data size but plateaus beyond 10%. With the full training set, all representations perform similarly across model sizes. In the low-data regime (1%), density-based inputs outperform `Atom-Type`, and the `Shape-Only` baseline—despite discarding chemical identity—performs comparably to density-based inputs. This counterintuitive result suggests that simple spatial occupancy alone may be highly predictive in this dataset, potentially due to biases in the benchmark or the use of static, bound structures. Insets show performance at 100% training data, with stars marking best-performing models within standard deviation.

All models are trained using the Adam optimizer with a learning rate of 1×10^{-5}, batch size 128, and Mean Squared Error (MSE) loss. To improve invariance to molecular orientation, we apply random 3D rotations to voxelized inputs during training.

Prior to training, all target labels are normalized to zero mean and unit variance. Predictions are rescaled during evaluation to report metrics in the original units. Each model is initially trained for 1500 epochs. Training continues for up to 5000 epochs if the validation loss has not converged, using early stopping with a patience of 50 epochs. Final results are reported as the mean and standard deviation of the MAE across three runs with different random seeds.

3.2.3 Data Efficiency Evaluation

To evaluate data efficiency, we train the models on subsets of the training data: 0.15%, 1%, 10%, and 100%. Performance across these subsets helps assess how each input representation scales with data availability.

4 Results

4.1 Binding Affinity Prediction (PDBbind)

Figure 1 shows the Spearman correlation on the PDBbind test set across training set sizes and model capacities. We observe that increasing model size has little effect on performance. The `Tiny` model (∼0.4M parameters) performs similarly to the largest model (`Default`, ∼58M), consistent with previous findings that small architectures can perform well in structure-based docking and virtual screening tasks [17].

Model performance improves with more training data but plateaus after 10%. At full data scale, all four input representations—`Atom-Type`, `Shape-Only`, `Density`, and `GradMag`—perform similarly within error margins.

Interestingly, in the low-data regime (1% of the training set, or ∼100 complexes), the density-based inputs (`Density` and `GradMag`) already achieved relatively strong performance—close to their performance when trained on the full dataset—and outperformed the atom-type input (`Atom-Type`)

across all models. Even more surprisingly, the `Shape-Only` input, which treats all atoms as carbon and discards atomic identity, performed comparably to the density-based inputs at this data size (except for the `Tiny` model). This result is counter-intuitive: one might expect `Shape-Only` to perform the worst as we removing the atom types. However, these findings suggest that in this dataset—where we use experimentally resolved structures in their bound, low-energy conformations—spatial occupancy alone (i.e., how well the ligand fills the binding pocket) may be a strong predictive signal. Prior work has noted that hydrophobic and shape-complementary interactions are often the dominant contributors to binding affinity, while atom-type–specific interactions (e.g., hydrogen bonds, salt bridges) tend to govern binding specificity [21].

The small difference between 10% and 100% training data further supports this: models may already extract most of the relevant information early on. In contrast, the `Atom-Type` input performs worse in low-data settings, likely due to its higher dimensionality (7 channels for ligand atoms and 4 for protein atoms), leading to sparser inputs and greater risk of overfitting.

4.2 Quantum Property Prediction

Figure 2 shows test MAE across training set sizes for QM9 target properties. As expected, performance improves consistently with more training data, regardless of model size, input type, or prediction target. This contrasts with the PDBbind results, where performance plateaus after 10%, and highlights QM9's greater sensitivity to data quantity.

Figure 3 summarizes results at 100% training data. Accuracy improves steadily with model size—from `Tiny` ($\sim$4M) to `Small` ($\sim$15M) and `Default` ($\sim$58M)—unlike the binding task, where model size had little effect. Atom-type information plays a more important role here: removing atomic identity (`Shape-Only`) consistently reduces performance across all models and data sizes, in line with the expectation that quantum properties are driven by electronic structure, not just shape.

In low-data regimes, performance varies across input types with no clear winner. However, at full data scale, density-based inputs (`Density`, `GradMag`) consistently outperform atom-type representations. These densities are generated using the semiempirical XTB method, which is significantly less computationally expensive—but also less accurate—than the DFT calculations used to derive the target molecular properties. This mismatch in the level of theory introduces a potential source of error; for example, XTB tends to systematically overestimate dipole moments relative to DFT (see Supplement, Figure 4).

Additional approximation error comes from voxelizing continuous densities. The supplemental figure reports error based on continuous basis-function densities, whereas our models use discretized voxel inputs, which add further discrepancy. Despite these limitations, `Density` and `GradMag` outperform both `Atom-Type` and `Shape-Only` (Figure 3), suggesting that voxelized density still captures important aspects of electronic structure not present in other representations.

Note on external comparisons. We do not compare to prior PDBbind models, as published results use different data splits and are not directly comparable without retraining. Moreover, recent work has shown that the splits used in previous studies may suffer from data leakage due to high sequence similarity between training and test proteins [22, 23]. Our experiments use controlled splits with minimal train-test overlap to fairly assess the effect of input representations (see Section 3.1). For QM9, we include SchNet [9] performance (Figure 2) only as a sanity check to verify that our 3D CNN benchmarks produce results within a reasonable error range. Our CNN models were not further tuned or optimized for performance, as the goal is to compare voxel-based representations under consistent training conditions, rather than to outperform graph-based or other state-of-the-art methods published in the literature. Because SchNet operates on graph-based atomic representations rather than volumetric grids, its results are not directly comparable to our voxel-based models.

5 Conclusion

This study systematically evaluated voxel-based molecular representations—`Atom-Type`, electron density (`Density`), and density gradient magnitude (`GradMag`)—as inputs to 3D convolutional neural networks for molecular property prediction. Motivated by the hypothesis that density-derived representations offer a richer, more physically grounded encoding than discrete atom-type models,

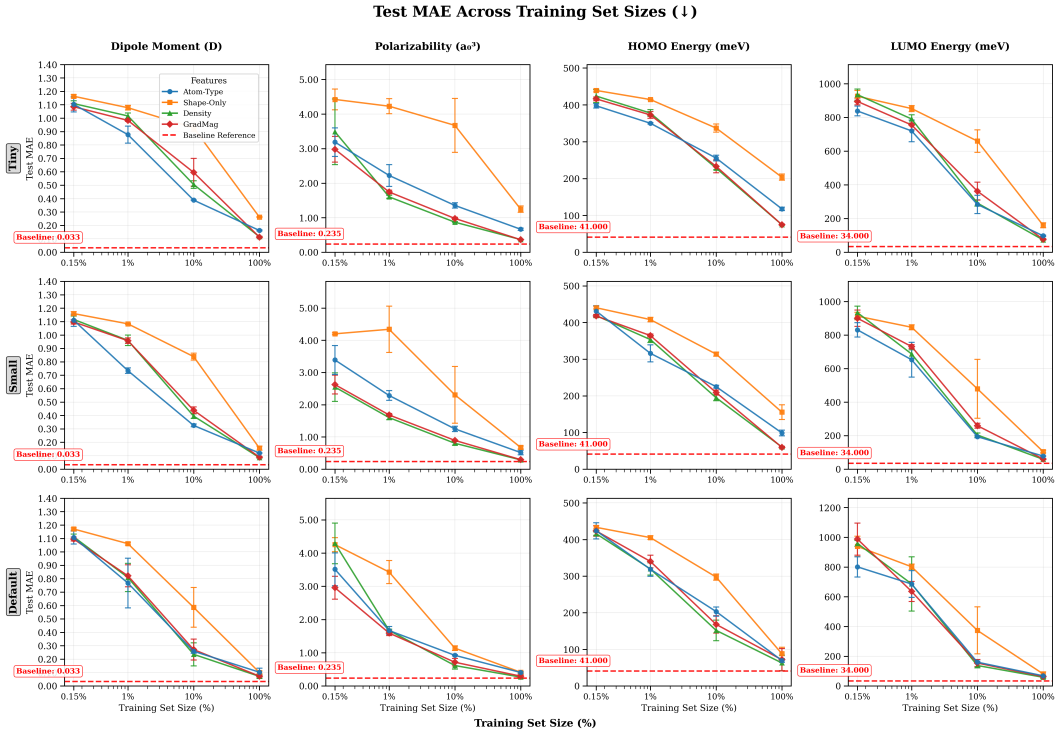

Figure 2: Test MAE across training set sizes for QM9 target properties. Each row corresponds to a different model size (`Tiny` ∼4M, `Small` ∼15M, `Default` ∼58M), and each column to a regression target (Dipole Moment, Polarizability, HOMO Energy, LUMO Energy). Lower MAE indicates better performance. Across all settings, increasing training data consistently reduces error. Density-based inputs (`Density`, `GradMag`) outperform atom-based ones at full data scale, while the poor performance of `Shape-Only` (orange) highlights the value of chemically informative features. Red dashed lines mark reported SchNet [9] results, shown only as a sanity check to confirm that our 3D CNN benchmarks yield reasonable error ranges. Our models were not tuned for state-of-the-art performance—the goal is to compare voxel-based representations under consistent conditions. Because SchNet is a graph neural network operating on atom-level graphs, it cannot directly represent volumetric density data without substantial reformulation, and its results are therefore not directly comparable to ours.

we benchmarked these approaches across two prediction tasks: 3D binding affinity prediction using experimental protein–ligand structures (PDBbind) [14], and quantum property prediction of small molecules (QM9) [18].

For 3D binding affinity prediction, increasing model size—from ∼0.4M to ∼58M parameters—had minimal impact on performance, consistent with prior work showing the effectiveness of compact architectures (e.g., GNINA [17]) for structure-based drug discovery. Model accuracy improved with additional training data but plateaued beyond ∼10%, indicating that most predictive information is captured early. In the low-data regime (1%), density-based inputs (`Density`, `GradMag`) outperformed atom-type inputs and achieved near-peak performance. Interestingly, a simplified input that removes atom-type information (`Shape-Only`) performed comparably to density-based inputs. These findings suggest that in this setup—using bound-state experimental structures where ligand poses and steric complementarity are already resolved—geometric occupancy alone may provide a strong predictive signal. In such cases, atom-type information may add limited benefit and can increase overfitting risk in low-data settings due to higher input sparsity.

In contrast, the quantum property prediction task on QM9 exhibited markedly different behavior. Performance improved consistently with both model capacity and data scale, with larger networks yielding better accuracy, indicating a greater need for representational expressiveness in this setting. Atom-type information also played a more critical role: removing atom identity (`Shape-Only`)

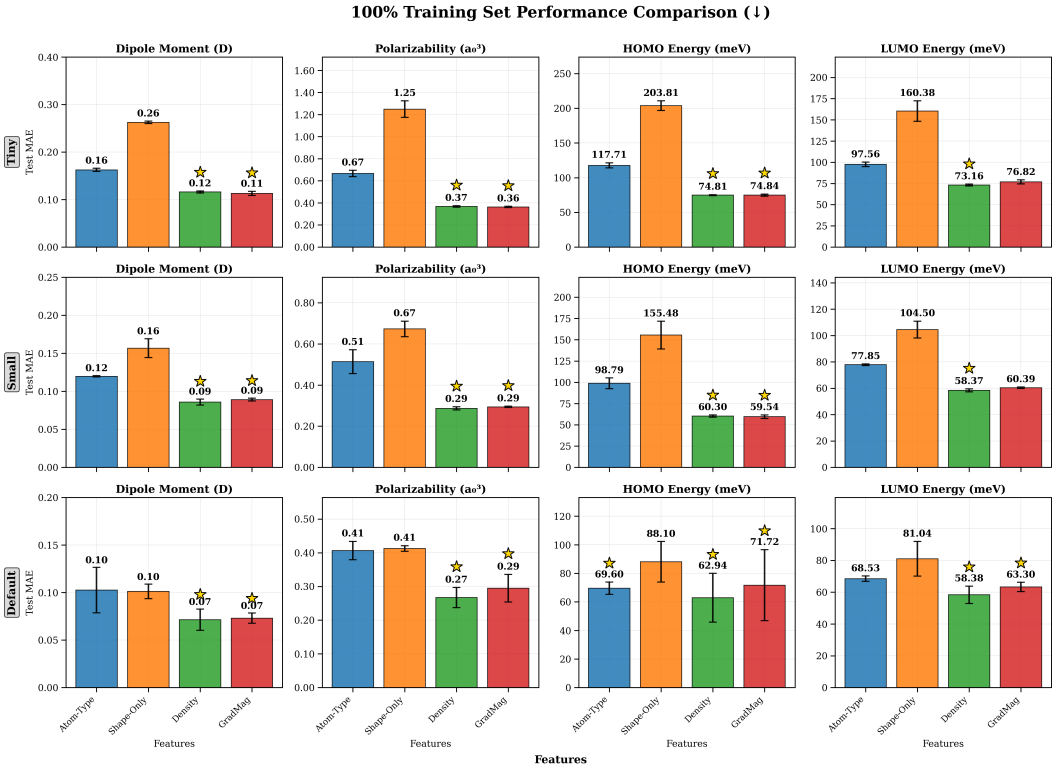

Figure 3: Test MAE at 100% training set size for all input types, targets, and model sizes. Each row shows results for a different model size, and each column corresponds to one of the QM9 target properties. Bars show mean test MAE with standard deviation across three seeds. Density-based inputs (`Density`, `GradMag`) consistently yield the lowest errors across all targets and model sizes. `Shape-Only` inputs perform worst overall, highlighting the value of chemically meaningful information in voxel inputs.

substantially degraded performance across all targets and model sizes, reflecting the importance of chemical specificity in quantum behavior. While no representation was consistently superior in low-data settings, density-based inputs (`Density`, `GradMag`) consistently outperformed atom-type inputs when trained on the full dataset. These densities were computed using the semiempirical XTB method, which is significantly less computationally expensive—but also less accurate—than the DFT methods used to generate the QM9 target properties. This introduces two distinct sources of approximation: (1) differences in the level of theory—e.g., XTB systematically overestimates dipole moments relative to DFT (see Supplement, Figure 4)—and (2) discretization error from voxelizing continuous densities. Despite these limitations, density-based inputs yielded the best performance overall, suggesting that electron density captures essential information about electronic structure that is not easily recovered from atom types or spatial geometry alone.

While these findings highlight the strengths of density-based voxel representations, several limitations and opportunities remain. This study focuses on 3D CNNs because they are the most natural choice for processing volumetric electron density data from both experimental and computed sources. Extending these ideas to graph-based or equivariant architectures would require new formulations capable of representing density within their atom-centric frameworks. Moreover, voxel-based representations are computationally intensive—storing and training on high-resolution volumetric grids can be prohibitively expensive for large datasets. Developing more compact or adaptive encodings could make density-based learning more practical at scale. It would also be valuable to explore hybrid approaches that combine atom-type specificity with density-derived features, potentially capturing complementary geometric and electronic information. Ultimately, the optimal molecular representation depends on the prediction task, data regime, and underlying physical principles most relevant to the property being modeled.

References

[1] José Jiménez, Miha Skalic, Gerard Martinez-Rosell, and Gianni De Fabritiis. K deep: protein–ligand absolute binding affinity prediction via 3d-convolutional neural networks. *Journal of chemical information and modeling*, 58(2):287–296, 2018.

[2] Matthew Ragoza, Joshua Hochuli, Elisa Idrobo, Jocelyn Sunseri, and David Ryan Koes. Protein–ligand scoring with convolutional neural networks. *Journal of chemical information and modeling*, 57(4):942–957, 2017.

[3] Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, Talapady N Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. The protein data bank. *Nucleic acids research*, 28(1):235–242, 2000.

[4] Zbigniew Dauter, Alexander Wlodawer, Wladek Minor, Mariusz Jaskolski, and Bernhard Rupp. Avoidable errors in deposited macromolecular structures: an impediment to efficient data mining. *IUCrJ*, 1(3):179–193, 2014.

[5] Alexander Wlodawer, Zbigniew Dauter, Przemyslaw J Porebski, Wladek Minor, Robyn Stanfield, Mariusz Jaskolski, Edwin Pozharski, Christian X Weichenberger, and Bernhard Rupp. Detect, correct, retract: How to manage incorrect structural models. *The FEBS journal*, 285(3): 444–466, 2018.

[6] Paul D Adams, Pavel V Afonine, Gábor Bunkóczi, Vincent B Chen, Ian W Davis, Nathaniel Echols, Jeffrey J Headd, L-W Hung, Gary J Kapral, Ralf W Grosse-Kunstleve, et al. Phenix: a comprehensive python-based system for macromolecular structure solution. *Biological crystallography*, 66(2):213–221, 2010.

[7] Lvwei Wang, Rong Bai, Xiaoxuan Shi, Wei Zhang, Yinuo Cui, Xiaoman Wang, Cheng Wang, Haoyu Chang, Yingsheng Zhang, Jielong Zhou, et al. A pocket-based 3d molecule generative model fueled by experimental electron density. *Scientific reports*, 12(1):15100, 2022.

[8] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. Pmlr, 2017.

[9] Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. *Advances in neural information processing systems*, 30, 2017.

[10] Nathaniel Thomas, Tess Smidt, Steven Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018.

[11] Fabian Fuchs, Daniel Worrall, Volker Fischer, and Max Welling. Se (3)-transformers: 3d roto-translation equivariant attention networks. *Advances in neural information processing systems*, 33:1970–1981, 2020.

[12] Johannes Gasteiger, Janek Groß, and Stephan Günnemann. Directional message passing for molecular graphs. In *International Conference on Learning Representations (ICLR)*, 2020.

[13] Izhar Wallach, Michael Dzamba, and Abraham Heifets. Atomnet: a deep convolutional neural network for bioactivity prediction in structure-based drug discovery. *arXiv preprint arXiv:1510.02855*, 2015.

[14] Renxiao Wang, Xueliang Fang, Yipin Lu, and Shaomeng Wang. The pdbbind database: Collection of binding affinities for protein- ligand complexes with known three-dimensional structures. *Journal of medicinal chemistry*, 47(12):2977–2980, 2004.

[15] Pedro O. Pinheiro, Arian Jamasb, Omar Mahmood, Vishnu Sresht, and Saeed Saremi. Structure-based drug design by denoising voxel grids. In *ICML*, 2024.

[16] Matthias R Bauer, Tamer M Ibrahim, Simon M Vogel, and Frank M Boeckler. Evaluation and optimization of virtual screening workflows with dekois 2.0–a public library of challenging docking benchmark sets. *Journal of chemical information and modeling*, 53(6):1447–1462, 2013.

[17] Andrew T McNutt, Paul Francoeur, Rishal Aggarwal, Tomohide Masuda, Rocco Meli, Matthew Ragoza, Jocelyn Sunseri, and David Ryan Koes. Gnina 1.0: molecular docking with deep learning. *Journal of cheminformatics*, 13(1):43, 2021.

[18] Raghunathan Ramakrishnan, Pavlo O Dral, Matthias Rupp, and O Anatole Von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific data*, 1(1):1–7, 2014.

[19] Stefan Grimme, Christoph Bannwarth, and Philip Shushkov. A robust and accurate tight-binding quantum chemical method for structures, vibrational frequencies, and noncovalent interactions of large molecular systems parametrized for all spd-block elements (z= 1–86). *Journal of chemical theory and computation*, 13(5):1989–2009, 2017.

[20] Pedro O Pinheiro, Joshua Rackers, Joseph Kleinhenz, Michael Maser, Omar Mahmood, Andrew Watkins, Stephen Ra, Vishnu Sresht, and Saeed Saremi. 3d molecule generation by denoising voxel grids. *Advances in Neural Information Processing Systems*, 36:69077–69097, 2023.

[21] Caterina Bissantz, Bernd Kuhn, and Martin Stahl. A medicinal chemist's guide to molecular interactions. *Journal of medicinal chemistry*, 53(14):5061–5084, 2010.

[22] Jie Li, Xingyi Guan, Oufan Zhang, Kunyang Sun, Yingze Wang, Dorian Bagni, and Teresa Head-Gordon. Leak proof pdbbind: A reorganized dataset of protein-ligand complexes for more generalizable binding affinity prediction. *ArXiv*, pages arXiv–2308, 2024.

[23] David Graber, Peter Stockinger, Fabian Meyer, Siddhartha Mishra, Claus Horn, and Rebecca Buller. Gems–enhancing generalizable binding affinity prediction by removing data leakage and integrating language model embeddings into graph neural networks. *bioRxiv*, pages 2024–12, 2024.

[24] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural networks*, 107:3–11, 2018.

[25] Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018.

[26] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

A Model Architectures

A.1 Binding Affinity Model Architecture (PDBbind)

We evaluate three 3D CNN architectures for predicting protein–ligand binding affinity using the PDBbind v2021 dataset [14]. Each complex is voxelized into two separate 64^3 grids—one for the ligand and one for the protein pocket—centered on the ligand's center of mass and resampled to 0.25 Å resolution.

Input Representations. Channel dimensions vary based on the representation:

- `Atom-Type`: 7 channels for ligand atoms (C, O, N, S, F, Cl, P), 4 channels for protein pocket atoms (C, O, N, S).
- `Shape-Only`, `Density`, `GradMag`: 1 channel each for ligand and protein pocket atoms.

Ligand and Pocket Encoders. The ligand and pocket are processed independently using identical 3D CNN encoders adapted from VoxBind [15]. Each encoder consists of a single residual block comprising two padded $3 \times 3 \times 3$ convolutional layers with 16 channels, followed by `SiLU` activations [24]. Each produces an output of shape 16×64^3. The two outputs are summed element-wise to produce a fused embedding.

Fused Representation Encoder. The fused representation is then processed by one of the following architectures depending on model capacity:

- `Tiny` ($\sim$0.4M parameters): The fused embedding is passed through the GNINA `Default2018` CNN architecture [17]. This consists of five 3D convolutional layers with interleaved average pooling and `ReLU` activations [25], followed by a fully connected linear layer to produce a scalar affinity prediction.
- `Small` ($\sim$4M parameters): The fused embedding is passed through the encoder portion of the 3D U-Net from VoxBind [15]. This U-Net encoder follows the original VoxBind design, using four resolution levels with channel multipliers `[1, 2, 2, 4]` and base channel width $n_{\text{ch}} = 8$. Each resolution level contains two residual blocks, and each residual block consists of two padded $3 \times 3 \times 3$ convolutions with 16 channels followed by `SiLU` activations [24]. Group normalization [26] with 4 groups is applied throughout. The encoder outputs a bottleneck feature map of shape 128×8^3.
- `Default` ($\sim$58M parameters): Same as `Small`, but with $n_{\text{ch}} = 32$, yielding a bottleneck of 512×8^3. Group normalization uses 16 groups.

MLP Prediction Head (Small and Default only). The bottleneck feature map is passed through a shared multi-layer perceptron (MLP) head to produce the final affinity prediction. This consists of:

- `AvgPool3d`(8)
- Linear layers with `LayerNorm`, `SiLU`, and `Tanhshrink` activations:

$$\texttt{Small:} \quad 128 \to 128 \to 64 \to 1$$

$$\texttt{Default:} \quad 512 \to 512 \to 64 \to 1$$

A.2 Quantum Property Prediction Model Architecture (QM9)

We use a 3D CNN encoder adapted from VoxMol [20] for scalar regression on voxelized QM9 molecules [18]. Each molecule is represented as a 32^3 grid.

Input Representations.

- `Atom-Type`: 5 channels (H, C, N, O, F)
- `Shape-Only`, `Density`, `GradMag`: 1 channel each

Encoder. The input is first projected to n_{ch} base channels using a $3 \times 3 \times 3$ convolution. The encoder applies four downsampling stages using channel multipliers `[1, 2, 2, 4]`, resulting in a final output of size $16n_{\text{ch}} \times 4^3$. Each stage contains two residual blocks with 3D convolution, `GroupNorm`, and `SiLU` activations. Self-attention is used in the final two stages.

MLP Prediction Head. The encoder output is passed to a multi-layer perceptron for scalar quantum property prediction:

$$16n_{\text{ch}} \to 64 \to 32 \to 1$$

Each layer includes `LayerNorm`, `SiLU`, and `Tanhshrink` activations.

Model Variants. The QM9 models share this architecture but vary in channel width and normalization:

- `Tiny` (∼4M): $n_{\text{ch}} = 8$, $n_{\text{groups}} = 4$
- `Small` (∼15M): $n_{\text{ch}} = 16$, $n_{\text{groups}} = 8$
- `Default` (∼58M): $n_{\text{ch}} = 32$, $n_{\text{groups}} = 16$

B Distribution of Dipole Moment Errors: XTB vs DFT

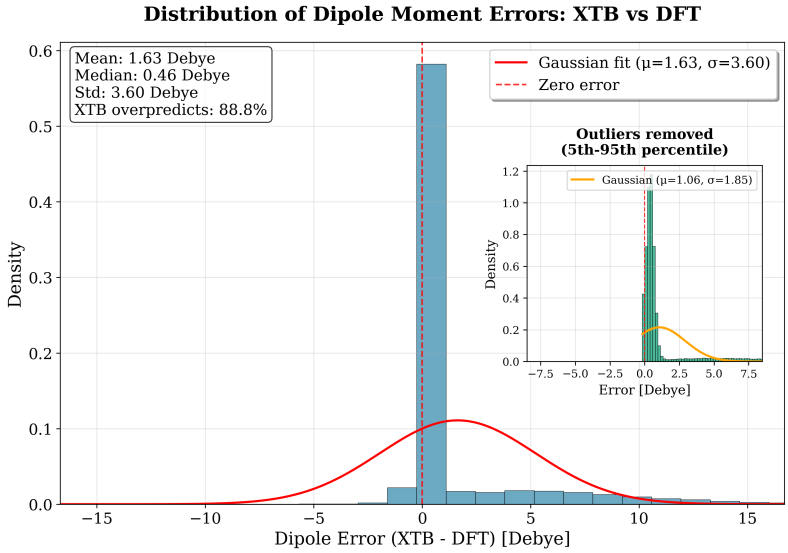

Figure 4: **Distribution of Dipole Moment Errors: XTB vs DFT.** Histogram of dipole moment errors (XTB − DFT) across the QM9 dataset. The red solid line shows a Gaussian fit to the full error distribution ($\mu = 1.63$, $\sigma = 3.60$ Debye), while the dashed red line indicates zero error. The inset shows the distribution with outliers removed (5th–95th percentile), highlighting the skew and overprediction tendency of XTB. XTB overpredicts dipole moments in 88.8% of cases.