HyperDiffusionFields (HyDiF): Diffusion-Guided Hypernetworks for Learning Implicit Molecular Neural Fields

Sudarshan Babu*
CZ Biohub Chicago
Chicago, IL
sudarshan.babu@czbiohub.org

Phillip Lo*
CZ Biohub Chicago
Chicago, IL
phillip.lo@czbiohub.org

Xiao Zhang
University of Chicago
Chicago, IL
zhang7@uchicago.edu

Aadi Srivastava
Indian Institute of Technology Madras
Chennai, TN
aadisrivastava.iitm@gmail.com

Ali Davariashtiyani
University of Chicago
Chicago, IL
davari@uchicago.edu

Jason Perera
CZ Biohub Chicago
Chicago, IL
jason.perera@czbiohub.org

Michael Maire
University of Chicago
Chicago, IL
mmaire@uchicago.edu

Aly A. Khan
University of Chicago
CZ Biohub Chicago
Chicago, IL
aakhan@czbiohub.org

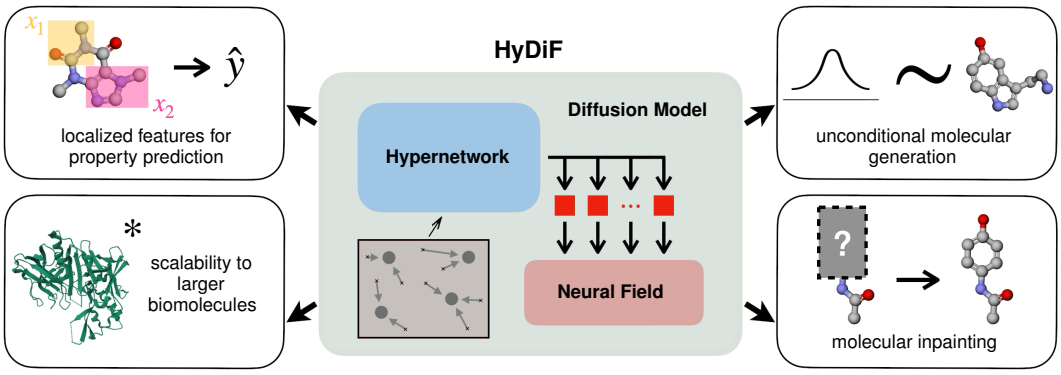

Figure 1: HyperDiffusionFields (HyDiF) is a framework for learning molecular conformers capable of a variety of tasks, including unconditional generation of molecules, inpainting, and representation learning for molecular property prediction. Furthermore, HyDiF scales to larger biomolecules.

Abstract

We introduce HyperDiffusionFields (HyDiF), a framework modeling 3D molecular conformers as continuous fields rather than discrete atomic coordinates or graphs. At the core of our approach is the Molecular Directional Field (MDF), a vector field that maps any point in space to the direction of the nearest atom of a particular type. We represent MDFs using molecule-specific neural implicit fields, which

*Equal contribution.

The NeurIPS 2025 Workshop on AI Virtual Cells and Instruments: A New Era in Drug Discovery and Development (AI4D3 2025), San Diego, California, USA, 2025.

we call Molecular Neural Fields (MNFs). To enable learning across molecules and facilitate generalization, we adopt an approach where a shared hypernetwork, conditioned on a molecule, generates the weights of the given molecule's MNF. To endow the model with generative capabilities, we train the hypernetwork as a denoising diffusion model, enabling sampling in the function space of molecular fields. Our design naturally extends to a masked diffusion mechanism to support structure-conditioned generation tasks, such as molecular inpainting, by selectively noising regions of the field. Beyond generation, the localized and continuous nature of MDFs enables spatially fine-grained feature extraction for molecular property prediction, something not easily achievable with graph or point cloud based methods. Furthermore, we demonstrate that our approach scales to larger biomolecules, illustrating a promising direction for field-based molecular modeling.

1 Introduction

Modeling complex, structured 3D data is a fundamental challenge in machine learning, with critical applications ranging from drug discovery and materials science to robotics and computer graphics [2–8]. Traditional approaches typically rely on discrete representations such as point clouds, meshes, or voxel grids. However, each of these come with limitations in resolution, expressiveness, or computational efficiency [9–13]. To overcome this, we use implicit neural representations (INRs), which model continuous fields over 3D space using coordinate-conditioned neural networks, have shown remarkable success in capturing high-fidelity geometry and generalizing across spatial scales (e.g., SIREN [14], NeRF [15]). By operating directly in continuous space, INRs are uniquely suited to capture fine-grained local geometric relationships crucial for modeling 3D objects [16, 17].

Our work focuses on 3D molecular modeling, where precise geometric understanding is essential for tasks such as binding affinity prediction, docking and molecular design [11, 18]. Most existing approaches represent molecules as discrete objects—typically as graphs or point clouds—limiting the ability to capture continuous, fine-grained spatial relationships [18, 19]. We lift these discrete representations into continuous space via the Molecular Directional Field (MDF), which maps any point in space to the direction of the nearest atom of a given type. This dense, continuous representation encodes local geometry, provides high-resolution structural information, and offers robustness to perturbations by encoding correlated directional signals at nearby points [16, 17].

To model the MDF, we propose **Hy**per**Di**ffusion**F**ields (HyDiF, illustrated in Figure 2)—a generative framework for learning continuous molecular representations. At the core of our approach is the Molecular Neural Field (MNF), a coordinate-conditioned neural implicit that models the MDF. Since training a separate MNF for each molecule prevents learning shared structure across molecules, we adopt an approach similar to HyperFields [20], where a shared hypernetwork generates the parameters of per-molecule MNFs, enabling cross-molecular generalization. This adoption is further motivated by recent findings that hypernetworks often exhibit improved generalization in out-of-distribution (OOD) settings [20–22]. To enable generative capabilities, we train the hypernetwork as a denoising diffusion model [23] over the space of MDFs, conditioning it to produce per-molecule MNFs. This allows explicit generation in the continuous function space of molecular fields, rather than discrete coordinate representations.

Building on the model's ability to capture fine-grained local geometry through the MDF, we extend our framework to support molecular inpainting, a critical task for *in silico* drug discovery [24–27]. This leverages the locality of our representation to enable structure-conditioned generation—a more practical and relevant capability than unconditional generation. The ability of HyDiF to selectively regenerate parts of a molecule makes it well-suited for applications that require incorporating known structural constraints while completing or optimizing the rest.

In addition to generation, we utilize HyDiF for feature extraction in downstream property prediction tasks. Generative models inherently learn rich, data-driven representations and recent work has demonstrated that such models excel at representation learning [28–32], enabling a unified framework that supports both generation and representation. Unlike traditional molecular encoders that produce a single global representation, our model naturally produces spatially localized features. Finally,

*Protein structure from [1].

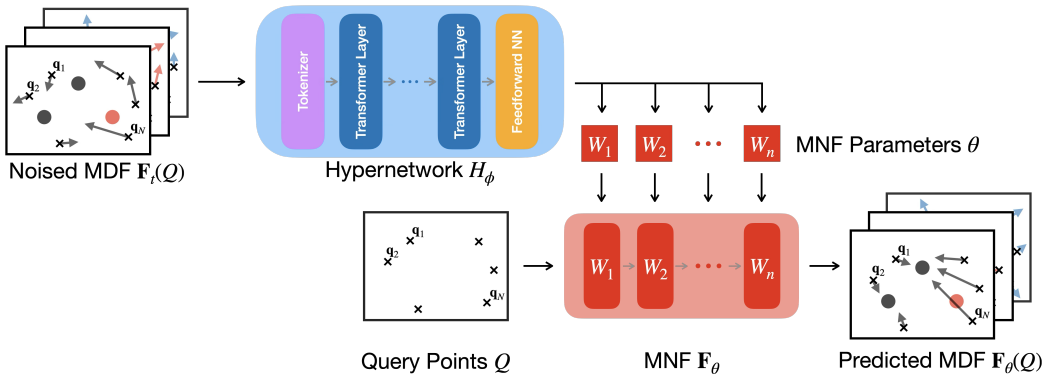

Figure 2: Overview of the HyDiF architecture. A hypernetwork H_ϕ takes a noised direction field $\mathbf{F}_t(Q)$, query points Q, and noising timestep t, and produces the parameters θ of an MNF $\mathbf{F}_\theta$. The neural field is evaluated at the same query points to generate a denoised field, which is compared against the ground truth to compute the training loss. This enables HyDiF to model molecules in a spatially localized manner.

we demonstrate the scalability of our approach by applying HyDiF to larger biomolecules such as proteins, which were previously intractable with existing methods. This extension underscores the flexibility of our framework in handling molecules of varying size and complexity. Overall, our work highlights a promising new direction in molecular modeling—one that shifts from discrete, rigid representations toward continuous, generative field-based models that unify representation learning and generation within a single framework.

2 Related Work

In this section, we position our work relative to molecular representations, molecular generative and inpainting models, implicit neural representations, and hypernetworks.

2.1 Molecular Representations

Traditional approaches often represent molecules as strings (e.g., SMILES [33–35], SELFIES [36, 37], InChI [38]) or graphs that capture atom connectivity but treat 3D structure as auxiliary. To explicitly incorporate spatial information, recent models use 3D point clouds [39–41] or voxel grids [42]. However, point clouds are sparse and permutation-sensitive, while voxel grids are resolution-limited and memory-intensive.

To address these limitations, we introduce the Molecular Directional Field (MDF)—a continuous vector field that maps 3D coordinates to atom-directed vectors, enabling dense, localized encoding of molecular geometry. This representation supports spatial querying, interpretability, and serves as a robust inductive bias for generative modeling.

2.2 Models for 3D Molecular Generation

Existing methods employ variational autoencoders (VAEs) [43–45], normalizing flows [46–48], or generative adversarial networks (GANs) [49–51] over strings or molecular graphs. These models often lack fine-grained control over 3D geometry, a critical limitation for tasks like conformer generation or structure-based drug design [52–56].

Denoising diffusion models have recently advanced the state-of-the-art in 3D molecular generation, with methods such as EDM [39], GeoLDM [40], and MiDi [41], applying noise to conformers and denoising in coordinate or latent spaces. These models often rely on equivariant architectures, but they still operate on sparse atomic representations. VoxMol [42] instead uses voxelized occupancy grids, providing a denser spatial representation, though at the cost of resolution and memory efficiency.

FuncMol [57] models 3D molecules using neural occupancy fields and performs diffusion in a learned latent space, capturing global molecular geometry while operating on compressed representations.

Their approach requires a two-stage training process—first learning the field representation, then training a separate diffusion model in latent space. Another related model MCF (Molecular Conformer Fields [58]) approaches conformer generation by learning a distribution over functions that map molecular graph elements to 3D coordinates using diffusion.

In contrast to these approaches, HyDiF performs diffusion directly in the function space of Molecular Directional Fields (MDFs)—dense, spatially continuous vector fields that map arbitrary 3D points to atom-directed vectors. Unlike latent-based methods, our framework is trained end-to-end and supports localized generation via masked denoising. While MCF models a mapping from molecular graphs to conformers and is limited to graph-to-structure generation, HyDiF models a function from 3D space to directional vectors, enabling both generation from noise and spatially localized structure-conditioned editing.

2.3 Molecular Inpainting and Scaffold-Constrained Generation

Scaffold-based methods like ScaffoldGVAE [59], Sc2Mol [60], and MoLeR [61] operate on SMILES or molecular graphs, generating molecules by conditioning on a core scaffold and elaborating it via learned motifs or side-chain transformations. Others such as DiffSBDD [52] perform 3D inpainting in the context of protein-ligand binding.

In contrast, HyDiF performs localized editing directly in continuous 3D space via masked denoising on Molecular Directional Fields. Unlike prior work, our edits are not constrained to predefined scaffolds or external protein contexts; instead, we enable user-specified, spatially targeted edits by masking arbitrary regions in the molecular field, offering a flexible, geometry-aware approach to molecular structure manipulation.

2.4 Hypernetworks

Hypernetworks [62] are neural networks that generate the weights of another network, enabling dynamic and flexible parameterization. They have proven particularly effective for generating implicit neural representations (INRs), where a shared hypernetwork maps latent codes to the weights of coordinate-based MLPs that model continuous signals [20, 63]. Additionally, works highlight that hypernetworks facilitate out-of-distribution generalization by decoupling representation learning from signal generation [21, 22]. Hence, we use hypernetworks in HyDiF to model a distribution over implicit molecular fields. This enables parameter sharing across molecules, supports generation and inference over unseen structures, and allows us to train on large libraries of molecule-specific neural implicits in a unified framework.

2.5 Generative Models as Feature Extractors

Recent work in computer vision has shown that diffusion models can serve as effective feature extractors, with intermediate activations capturing rich semantic information necessary for generation [29, 64, 65]. Motivated by this, we explore the use of intermediate representations from HyDiF for downstream molecular tasks. This dual role—both generative and descriptive—makes HyDiF the first model in molecular conformer modeling to unify generation and feature extraction within a single framework.

3 Method

In this section, we describe the HyDiF framework. Each molecule is represented as a dense field that maps every point in 3D space to a vector pointing toward the nearest atom, yielding a locally structured and geometrically aware representation. Our model learns to generate such fields by reversing a diffusion process using a transformer-based hypernetwork that predicts the parameters of a neural field conditioned on noisy observations. This formulation supports both unconditional generation and molecular inpainting, and enables local feature extraction from molecular structure. HyDiF brings together diffusion models for flexible generative modeling, hypernetworks for task-specific adaptivity, and implicit neural representations for capturing fine-grained 3D molecular structure.

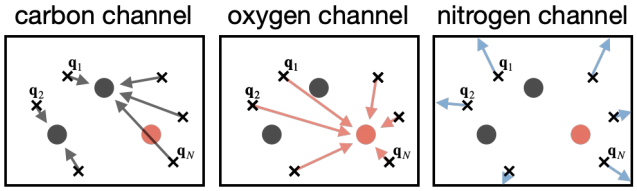

Figure 3: Direction field channels for a molecule with two carbons, one oxygen, and no nitrogens. Each query point is mapped to the nearest atom of the corresponding type. Observe the nitrogen channel points outwards.

3.1 Molecular Direction Fields (MDFs)

We represent 3D molecular conformers using *Molecular Direction Fields*—vector fields over continuous space that capture local geometric structure. At every point in space, the field points toward the nearest atom in the molecule, offering a dense representation of molecular shape.

To build intuition for MDFs, consider constructing a direction field for a toy molecule consisting of atoms of a single type—e.g., carbon. Let $\mathcal{A} = \{\mathbf{a}_1, \ldots, \mathbf{a}_n\}$ denote the set of atomic coordinates. We define the vector field $\mathbf{F} : \mathbb{R}^3 \to \mathbb{R}^3$ where for any query point $\mathbf{q} \in \mathbb{R}^3$, the output $\mathbf{F}(\mathbf{q})$ is the vector pointing from $\mathbf{q}$ to the nearest atom in $\mathcal{A}$:

$$\mathbf{F}(\mathbf{q}) = \mathbf{a}_{\text{nearest}} - \mathbf{q}, \text{ where } \mathbf{a}_{\text{nearest}} = \arg\min_{\mathbf{a} \in \mathcal{A}} \|\mathbf{q} - \mathbf{a}_i\|_2. \tag{1}$$

To extend to molecules with multiple atom types, we construct a separate direction field $\mathbf{F}^{(k)}$ for each atom type k over the subset $\mathcal{A}_k \subset \mathcal{A}$ which contains only atoms of type k. The resulting representation is a multi-channel vector field $\mathbf{F} : \mathbb{R}^3 \to \mathbb{R}^{K \times 3}$, where K is the number of distinct atom types in the dataset. Each channel $\mathbf{F}^{(k)}(\mathbf{q})$ is the vector from $\mathbf{q}$ to the nearest atom of type k.

To extend to molecules with multiple atom types, we construct a separate direction field $\mathbf{F}^{(k)}$ for each atom type k over the subset $\mathcal{A}_k \subset \mathcal{A}$ which contains only atoms of type k. The resulting representation is a multi-channel vector field $\mathbf{F} : \mathbb{R}^3 \to \mathbb{R}^{K \times 3}$, where K is the number of distinct atom types in the dataset. Each channel $\mathbf{F}^{(k)}(\mathbf{q})$ is the vector from $\mathbf{q}$ to the nearest atom of type k.

Each molecule in our dataset is thus represented by a K-channel direction field, one channel for each possible atom type. If a particular type k is not present in a given molecule, we populate that channel with outward-pointing vectors that radiate from the molecule's center of mass toward a bounding sphere. This ensures that the representation remains dense and consistent across all molecules. This construction is illustrated in Figure 3.

While MDFs provide a rich representation for molecules, for downstream applications we are still interested in human-parsable graph or string representations; we describe how we recover molecular graphs from MDFs in §D in the supplementary material.

3.2 Molecular Neural Fields (MNFs)

We represent Molecular Direction Fields using *Molecular Neural Fields* (MNFs)—coordinate-based neural networks trained to approximate the ground truth MDF of a molecule. Each MNF is a function $\mathbf{F}_\theta : \mathbb{R}^3 \to \mathbb{R}^{K \times 3}$ that maps a query point to a multi-channel output, where the kth channel predicts the direction to the nearest atom of type k. The network is trained to match $\mathbf{F}_\theta$ to $\mathbf{F}$ at sampled points and is parameterized as an MLP with sinusoidal activations known as a SIREN [14], which is well-suited for modeling high-frequency geometric signals.

3.3 Forward Diffusion Process

To model a distribution over Molecular Direction Fields, we adopt the denoising diffusion probabilistic model (DDPM) framework [23], applied to vector fields. The forward diffusion process defines a Markov chain that gradually transforms a clean field into Gaussian noise over T discrete steps.

Given a ground-truth direction field $\mathbf{F}$, evaluated at a set of spatial query points $Q = \{\mathbf{q}_j\}_{j=1}^N$, we construct the noised field at timestep t as:

$$\mathbf{F}_t(Q) = \sqrt{\bar{\alpha}_t} \cdot \mathbf{F}(Q) + \sqrt{1 - \bar{\alpha}_t} \cdot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, I), \tag{2}$$

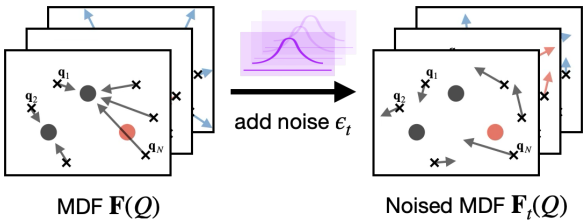

Figure 4: The forward noising process for an MDF, where i.i.d. Gaussian noise is added to each component of each vector at each query point. Here, $\boldsymbol{\epsilon}_t$ is short for $\sqrt{1-\bar{\alpha}_t} \cdot \boldsymbol{\epsilon}$.

where $\bar{\alpha}_t = \prod_{i=1}^{t}(1-\beta_i)$, and $\{\beta_i\}_{i=1}^{T}$ is a fixed noise schedule. We adopt the cosine schedule proposed in [66], which improves stability over the linear schedule in high-resolution spatial domains.

At timestep $t = 0$, the noised field reduces to the clean field, while at the final step $t = T$, the noised field $\mathbf{F}_T(Q)$ becomes almost indistinguishable from isotropic Gaussian noise. This progression forms a trajectory through field space, allowing the model to learn to denoise under varying signal-to-noise ratios. The noise is applied independently to each vector component at each spatial location and for each channel in the field. Each vector $\mathbf{F}_t(\mathbf{q})$ is corrupted by additive Gaussian noise scaled according to the diffusion schedule (see Figure 4).

3.4 Architecture

The core idea of HyDiF (see Figure 2) is to use a hypernetwork to generate the parameters of a MNF that denoises a noised molecular direction field. The hypernetwork, denoted H_ϕ, takes three inputs: the noised direction field at timestep t, denoted $\mathbf{F}_t(Q)$; the corresponding 3D query points $Q = \{\mathbf{q}_j\}_{j=1}^{N}$; and the diffusion timestep t itself. The output is a set of parameters θ, which define a coordinate-based neural network $\mathbf{F}_\theta$ that serves as the denoised MNF. Formally, we have

$$\theta = H_\phi(\mathbf{F}_t(Q), Q, t). \tag{3}$$

We include further details about the HyDiF architecture in §B of the supplementary material.

3.5 Reverse Process and Training

Once the parameters θ are generated by the hypernetwork, we use the resulting MNF $\mathbf{F}_\theta$ to reverse the forward noising process. Specifically, we evaluate $\mathbf{F}_\theta$ at the same query points used in the forward process to predict the clean direction field. These predictions are then compared to the ground-truth direction vectors in the following training objective:

$$\min_{\phi} \mathcal{L}_{\text{vec}}, \text{ where } \mathcal{L}_{\text{vec}} = \frac{1}{|Q|}\sum_{\mathbf{q}\in Q}\sum_{k=1}^{K}\left\|\mathbf{F}_\theta^{(k)}(\mathbf{q}) - \mathbf{F}^{(k)}(\mathbf{q})\right\|_1 \text{ and } \theta = H_\phi(\mathbf{F}_t(Q), Q, t). \tag{4}$$

In practice, we find this vector-based objective to be unstable. Molecular direction fields often contain high-frequency discontinuities, particularly where the identity of the nearest atom can change abruptly across neighboring spatial points. These discontinuities make the direction field difficult to fit with compact MLPs, especially when training with noisy inputs. To address this, we instead train the MNF to predict the scalar *distance field*, defined as the Euclidean distance from a point $\mathbf{q}$ to the nearest atom of each type. Define

$$f^{(k)}(\mathbf{q}) = \|\mathbf{a}_{\text{nearest}}^{(k)} - \mathbf{q}\|, \text{ where } \mathbf{a}_{\text{nearest}}^{(k)} = \arg\min_{\mathbf{a}\in\mathcal{A}_k}\|\mathbf{q}-\mathbf{a}\|_2. \tag{5}$$

This scalar function is smoother and easier for neural networks to approximate. We correspondingly define the MNF as a function $f_\theta : \mathbb{R}^3 \to \mathbb{R}^K$, where each output channel predicts the distance to the nearest atom of type k. The associated training objective is then

$$\min_{\phi} \mathcal{L}_{\text{dist}}, \text{ where } \mathcal{L}_{\text{dist}} = \frac{1}{|Q|}\sum_{\mathbf{q}\in Q}\sum_{k=1}^{K}\left|f_\theta^{(k)}(\mathbf{q}) - f^{(k)}(\mathbf{q})\right| \text{ and } \theta = H_\phi(\mathbf{F}_t(Q), Q, t). \tag{6}$$

In this setup, each MNF predicts one scalar distance per atom type for every spatial location. Through experimentation, we observe that the best performance is achieved when the hypernetwork

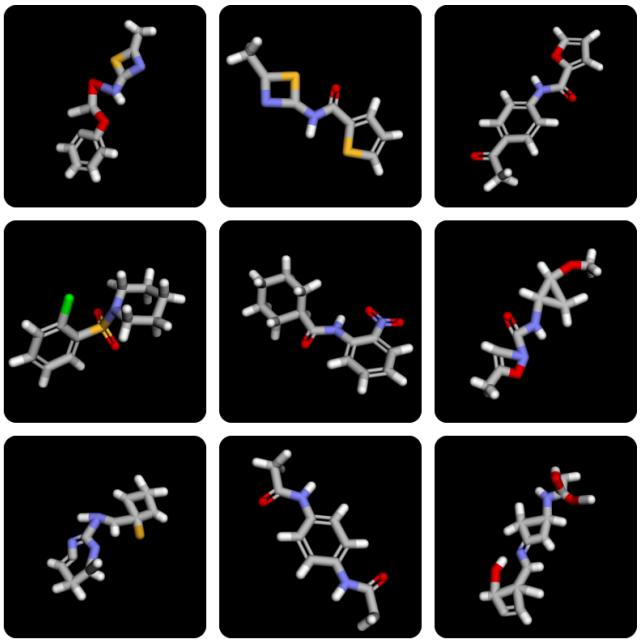

Figure 5: Nine curated samples from unconditional generation. HyDiF generates chemically plausible and structurally diverse molecules when sampling unconditionally.

is *conditioned on* a noised direction field, but trained to *output* a distance field. This hybrid setup leverages the high-frequency information present in the direction field for conditioning, while taking advantage of the smoother, lower-frequency distance field as the target for regression. We provide ablation studies supporting this choice in §I in the supplementary material. Full algorithms for forward noising and generation are provided in the supplementary material §H.

3.6 Masked Diffusion Procedure for Molecular Inpainting

The geometric nature of the HyDiF architecture supports a molecular inpainting task: given a partially masked molecule, the model completes the missing regions, enabling selective modification of portions of a molecule. This is a crucial step in the drug discovery pipeline commonly referred to as *lead optimization* [67–70]. We discuss the details of our masked diffusion procedure more in §E.

4 Results

We pretrain HyDiF on the gold standard GEOM (Geometric Ensemble of Molecules [71]) dataset—a collection of high-quality conformers computed *in silico* using semi-empirical tight-binding density functional theory. We use the same GEOM-drugs dataset and splits as [57], discarding molecules containing elements other than C, H, O, N, F, S, Cl, and Br; this yields a training set of 1.1M conformers across 242K species. We show unconditional generation results after pretraining in Figure 5, where nine curated samples demonstrate that HyDiF generates diverse and realistic conformers when sampling from pure noise. In §F of the supplementary material, we describe and show preliminary results of fitting HyDiF to proteins.

4.1 Molecular Inpainting

In the drug discovery pipeline, a candidate drug for a particular task is known as a lead. Medicinal chemists are then interested in finding a large number of structural analogues to this lead, i.e., a variety of molecules with similar structure, in hopes of having a large number of candidates with similar interaction with a biological target, but perhaps more favorable absorption, etc. For this reason, the conditional generation task (molecular inpainting) is much more relevant in real-world applications, as opposed to the unconditional task that most other works evaluate their models on.

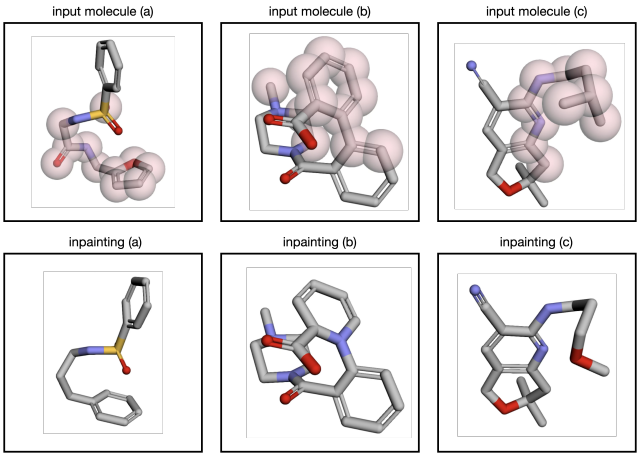

Figure 6: Three molecules from the validation set and their corresponding inpaintings. Yellow denotes the region of the molecule we selected for inpainting. The above examples demonstrate the ability of HyDiF to inpaint effectively. We attribute this to HyDiF's ability to locally model conformers via MDFs. (a) replacement of the five-member ring with a six-member ring, and the replacement of a nitrogen (blue) with carbon (gray). (b) replacement of a carbon in a benzene ring with a nitrogen. (c) replacement of a branched chain with a non-branched chain.

Table 1: Molecular inpainting metrics on GEOM (higher is better). Metrics for EDM and HyDiF are computed over 100 generated conformers. HyDiF's high-resolution molecular representation is able to inpaint more chemically reasonable molecules than EDM's discrete representation.

	Atom Stable %	Mol. Stable %	Validity %
data	100.0%	100.0%	96.9%
EDM	93.1%	13%	85%
HyDiF	**95.1**%	**36**%	**93**%

We train HyDiF on the molecular inpainting task similarly to the pretraining method in §3.5 but with the masked diffusion procedure in §E. In Figure 6, we exhibit a selection of partially masked conformers from the validation set and the corresponding inpainted molecules. In each case, HyDiF is able to successfully perform local edits on a molecule while maintaining chemical validity.

To quantify the quality of inpainting, we compute three metrics on generated conformers: *atom stability*, *molecule stability*, and *validity*. Atom stability measures the percentage of atoms with correct valency; molecule stability measures the percentage of molecules where all atoms have correct valency; validity reports the fraction of molecules that pass RDKit's sanitization check. Table 1 compares HyDiF to the EDM baseline on these metrics.

The inpainting task highlights the utility of our novel field-based representation of molecules and our accompanying hypernetwork architecture, which performs diffusion directly in molecular space rather than latent space; this is a departure from most other methods. This is why we are only able to compare to EDM (which performs diffusion in atomic coordinates space); most other models such as FuncMol and GeoLDM perform diffusion in latent space. Latent diffusion models cannot be straightforwardly modified for the conditional generation task. On the other hand, for EDM and our method, the diffusion in molecular space makes it straightforward to perform diffusion on a partially masked molecule.

4.2 Molecular Property Prediction

Datasets. To evaluate how well our pretrained model generalizes to downstream property prediction tasks, we use a subset of benchmarks from MoleculeNet [72], which span regression and binary classification tasks (see §C in the supplementary material for details).

Table 2: Comparison on MoleculeNet tasks. HyDiF alone compares favorably with baselines which learn global representations, highlighting the efficacy of locally learned representations.

	Classification (AUROC ↑)		**Regression (RMSE ↓)**		
	BBBP	BACE	Lipo	ESOL	FreeSolv
ChemBERTa-2	0.70	0.79	0.81	0.98	2.12
EDM	0.67	0.78	0.94	1.03	2.25
GeoLDM	0.68	0.78	0.97	1.26	2.60
HyDiF	0.71	0.79	0.83	0.90	1.82
HyDiF + ChemBERTa-2	**0.73**	**0.80**	**0.78**	**0.89**	**1.75**

Baselines. We compare our method against a large-scale sequence-based foundational model and two structure-based diffusion models trained on the same dataset as ours. ChemBERTa-2 [73] is a RoBERTa-style [74] transformer pretrained on 77M SMILES strings using masked language modeling. As a foundational model trained on orders of magnitude more data than ours, it serves as a strong baseline for molecular property prediction. In contrast, EDM [39] and GeoLDM [40] are pretrained diffusion models trained on GEOM. Both operate directly on 3D molecular structures using graph neural networks: EDM employs a SE(3)-GNN, while GeoLDM uses SE(3)-GNN with latent diffusion. To repurpose EDM and GeoLDM as feature extractors, we extract a global molecular embedding by averaging per-atom feature vectors from intermediate layers of the GNN.

HyDiF. HyDiF is pretrained on the GEOM dataset using the procedure described earlier in §3.5. For a given molecule, we generate its MDF and sample a dense set of query points uniformly throughout the molecular volume. The MDF is passed through a pretrained HyDiF, from which we extract intermediate activations as features. This allows us to obtain localized representations at high spatial resolution. We average these local features to produce a single global embedding for the molecule.

HyDiF + Chemberta-2. In addition to using HyDiF on its own, we also evaluate a fusion model that concatenates the global embedding from HyDiF with ChemBERTa-2's representation. This combined feature vector is then passed to the same MLP head for downstream property prediction. This allows us to test the complementarity of geometry-aware and sequence-based representations.

Comparison. In Table 2, across all tasks, HyDiF compares favorably to both ChemBERTa-2 and the structure-based diffusion baselines. This is despite the fact that ChemBERTa-2 is trained on 77M unique species, while HyDiF is trained on 242K unique species. We attribute this performance to a fundamental difference in how our model represents molecules: rather than producing a single global embedding directly, HyDiF aggregates localized features learned from spatially resolved query points. This allows the model to capture fine-grained geometric information that is otherwise lost in global or sequence-based representations.

5 Conclusion

This work introduces HyDiF, a generative framework that models 3D molecular structure via direct diffusion on continuous vector fields and neural function spaces. The continuous nature of our MDF representation allows for HyDiF to learn rich localized representations, enabling highly competitive feature generation for molecular property prediction as well as the ability to make localized edits to molecules via molecular inpainting. Moreover, the coordinate- and grid-free nature of our representation allows HyDiF to scale to larger biomolecules. We believe the contributions of HyDiF represent a promising direction for future work in geometric deep learning, one that bridges the gap between high-fidelity representation and flexible, spatially aware generation.

References

[1] T.T. Chen, W.Y. Chen, and Y.C. Xu. Crystal structure of BACE1 with its inhibitor, November 2012. URL http://dx.doi.org/10.2210/pdb3uqu/pdb.

[2] Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric Deep Learning: Going beyond Euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017. doi: 10.1109/MSP.2017.2693418.

[3] R. Qi Charles, Hao Su, Mo Kaichun, and Leonidas J. Guibas. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 77–85, 2017. doi: 10.1109/CVPR.2017.16.

[4] Yin Zhou and Oncel Tuzel. VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4490–4499, 2018. doi: 10.1109/CVPR.2018.00472.

[5] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip Torr, and Vladlen Koltun. Point Transformer. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16239–16248, 2021. doi: 10.1109/ICCV48922.2021.01595.

[6] Rana Hanocka, Amir Hertz, Noa Fish, Raja Giryes, Shachar Fleishman, and Daniel Cohen-Or. MeshCNN: a network with an edge. *ACM Trans. Graph.*, 38(4), July 2019. ISSN 0730-0301. doi: 10.1145/3306346.3322959.

[7] Patrick Reiser, Marlen Neubert, André Eberhard, Luca Torresi, Chen Zhou, Chen Shao, Houssam Metni, Clint van Hoesel, Henrik Schopmans, Timo Sommer, and Pascal Friederich. Graph neural networks for materials science and chemistry. *Communications Materials*, 3(1), November 2022. ISSN 2662-4443. doi: 10.1038/s43246-022-00315-6.

[8] Amil Merchant, Simon Batzner, Samuel S Schoenholz, Muratahan Aykol, Gowoon Cheon, and Ekin Dogus Cubuk. Scaling deep learning for materials discovery. *Nature*, 624(7990):80–85, 2023.

[9] Bojun Liu, Yangzhi Ma, Ao Luo, Li Li, and Dong Liu. Voxel-based Point Cloud Geometry Compression with Space-to-Channel Context, 2025. URL https://arxiv.org/abs/2503.18283.

[10] Maria Boulougouri, Pierre Vandergheynst, and Daniel Probst. Molecular set representation learning. *Nature Machine Intelligence*, 6(7):754–763, July 2024. ISSN 2522-5839. doi: 10.1038/s42256-024-00856-0.

[11] Huiwen Wang. Prediction of protein–ligand binding affinity via deep learning models. *Briefings in Bioinformatics*, 25(2):bbae081, 03 2024. ISSN 1477-4054. doi: 10.1093/bib/bbae081.

[12] Yeji Wang, Shuo Wu, Yanwen Duan, and Yong Huang. A point cloud-based deep learning strategy for protein–ligand binding affinity prediction. *Briefings in Bioinformatics*, 23(1): bbab474, 11 2021. ISSN 1477-4054. doi: 10.1093/bib/bbab474.

[13] Zifan Shi, Sida Peng, Yinghao Xu, Andreas Geiger, Yiyi Liao, and Yujun Shen. Deep Generative Models on 3D Representations: A Survey, 2023. URL https://arxiv.org/abs/2210.15663.

[14] Vincent Sitzmann, Julien N. P. Martel, Alexander W. Bergman, David B. Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NeurIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

[15] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: representing scenes as neural radiance fields for view synthesis. *Commun. ACM*, 65(1):99–106, December 2021. ISSN 0001-0782. doi: 10.1145/3503250.

[16] Sosuke Kobayashi, Eiichi Matsumoto, and Vincent Sitzmann. Decomposing NeRF for editing via feature field distillation. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NeurIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

[17] Itai Lang, Fei Xu, Dale Decatur, Sudarshan Babu, and Rana Hanocka. iSeg: Interactive 3D Segmentation via Interactive Attention. In *SIGGRAPH Asia 2024 Conference Papers*, SA '24, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400711312. doi: 10.1145/3680528.3687605.

[18] Jiaqi Guan, Wesley Wei Qian, Xingang Peng, Yufeng Su, Jian Peng, and Jianzhu Ma. 3D Equivariant Diffusion for Target-Aware Molecule Generation and Affinity Prediction. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=kJqXEPXMsE0.

[19] Pavol Drotár, Arian Rokkum Jamasb, Ben Day, Cătălina Cangea, and Pietro Liò. Structure-aware generation of drug-like molecules, 2021. URL https://arxiv.org/abs/2111.04107.

[20] Sudarshan Babu, Richard Liu, Avery Zhou, Michael Maire, Greg Shakhnarovich, and Rana Hanocka. HyperFields: Towards Zero-Shot Generation of NeRFs from Text, 2024. URL https://openreview.net/forum?id=84Hk01tFKq.

[21] Sudarshan Babu, Pedro Savarese, and Michael Maire. HyperNetwork Designs for Improved Classification and Robust Meta-Learning, 2020. URL http://people.cs.uchicago.edu/~sudarshan/pdf/HyperNet_MAML.pdf.

[22] M. Przewięźlikowski, P. Przybysz, J. Tabor, M. Zięba, and P. Spurek. HyperMAML: Few-Shot Adaptation of Deep Models with Hypernetworks, 2024. URL https://arxiv.org/abs/2205.15745.

[23] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NeurIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

[24] Oriel Frigo, Rémy Brossard, and David Dehaene. Graph Context Encoder: Graph Feature Inpainting for Graph Generation and Self-supervised Pretraining, 2021. URL https://arxiv.org/abs/2106.10124.

[25] Thomas E. Hadfield, Fergus Imrie, Andy Merritt, Kristian Birchall, and Charlotte M. Deane. Incorporating Target-Specific Pharmacophoric Information into Deep Generative Models for Fragment Elaboration. *Journal of Chemical Information and Modeling*, 62(10):2280–2292, 2022. doi: 10.1021/acs.jcim.1c01311. PMID: 35499971.

[26] Maxime Langevin, Hervé Minoux, Maximilien Levesque, and Marc Bianciotto. Scaffold-Constrained Molecular Generation. *Journal of Chemical Information and Modeling*, 60(12): 5637–5646, 2020. doi: 10.1021/acs.jcim.0c01015. PMID: 33301333.

[27] Krzysztof Maziarz, Henry Richard Jackson-Flux, Pashmina Cameron, Finton Sirockin, Nadine Schneider, Nikolaus Stiefl, Marwin Segler, and Marc Brockschmidt. Learning to Extend Molecular Scaffolds with Structural Motifs. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=ZTsoE8G3GG.

[28] Xiao Zhang, Ruoxi Jiang, William Gao, Rebecca Willett, and Michael Maire. Residual Connections Harm Generative Representation Learning, 2025. URL https://arxiv.org/abs/2404.10947.

[29] Xiao Zhang and Michael Maire. Structural Adversarial Objectives for Self-Supervised Representation Learning, 2023. URL https://arxiv.org/abs/2310.00357.

[30] Xiao Zhang, David Yunis, and Michael Maire. Deciphering 'What' and 'Where' Visual Pathways from Spectral Clustering of Layer-Distributed Neural Representations, 2024. URL https://arxiv.org/abs/2312.06716.

[31] Xingyi Yang and Xinchao Wang. Diffusion Model as Representation Learner. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 18892–18903, 2023. doi: 10.1109/ICCV51070.2023.01736.

[32] Tianhong Li, Huiwen Chang, Shlok Kumar Mishra, Han Zhang, Dina Katabi, and Dilip Krishnan. MAGE: MAsked Generative Encoder to Unify Representation Learning and Image Synthesis. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2142–2152, 2023. doi: 10.1109/CVPR52729.2023.00213.

[33] David Weininger. SMILES, a Chemical Language and Information System. 1. Introduction to Methodology and Encoding Rules. *Journal of Chemical Information and Computer Sciences*, 28(1):31–36, 1988. doi: 10.1021/ci00057a005.

[34] David Weininger, Arthur Weininger, and Joseph L. Weininger. SMILES. 2. Algorithm for Generation of Unique SMILES Notation. *Journal of Chemical Information and Computer Sciences*, 29(2):97–101, 1989. doi: 10.1021/ci00062a008.

[35] David Weininger. SMILES. 3. DEPICT. Graphical Depiction of Chemical Structures. *Journal of Chemical Information and Computer Sciences*, 30(3):237–243, 1990. doi: 10.1021/ci00067a005.

[36] Mario Krenn, Florian Häse, AkshatKumar Nigam, Pascal Friederich, and Alan Aspuru-Guzik. Self-referencing Embedded Strings (SELFIES): A 100% Robust Molecular String Representation. *Machine Learning: Science and Technology*, 1(4):045024, October 2020. doi: 10.1088/2632-2153/aba947.

[37] Mario Krenn, Qianxiang Ai, Senja Barthel, Nessa Carson, Angelo Frei, Nathan C. Frey, Pascal Friederich, Théophile Gaudin, Alberto Alexander Gayle, Kevin Maik Jablonka, Rafael F. Lameiro, Dominik Lemm, Alston Lo, Seyed Mohamad Moosavi, José Manuel Nápoles-Duarte, AkshatKumar Nigam, Robert Pollice, Kohulan Rajan, Ulrich Schatzschneider, Philippe Schwaller, Marta Skreta, Berend Smit, Felix Strieth-Kalthoff, Chong Sun, Gary Tom, Guido Falk von Rudorff, Andrew Wang, Andrew D. White, Adamo Young, Rose Yu, and Alán Aspuru-Guzik. SELFIES and the Future of Molecular String Representations. *Patterns*, 3(10):100588, 2022. ISSN 2666-3899. doi: https://doi.org/10.1016/j.patter.2022.100588.

[38] Stephen R Heller, Alan McNaught, Igor Pletnev, Stephen Stein, and Dmitrii Tchekhovskoi. InChI, the IUPAC International Chemical Identifier. *Journal of Cheminformatics*, 7, May 2015. ISSN 1758-2946.

[39] Emiel Hoogeboom, Víctor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant Diffusion for Molecule Generation in 3D. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 8867–8887. PMLR, July 2022.

[40] Minkai Xu, Alexander Powers, Ron Dror, Stefano Ermon, and Jure Leskovec. Geometric Latent Diffusion Models for 3D Molecule Generation, May 2023. arXiv:2305.01140.

[41] Clement Vignac, Nagham Osman, Laura Toni, and Pascal Frossard. MiDi: Mixed Graph and 3D Denoising Diffusion for Molecule Generation, June 2023. arXiv:2302.09048.

[42] Pedro O. Pinheiro, Joshua Rackers, Joseph Kleinhenz, Michael Maser, Omar Mahmood, Andrew Martin Watkins, Stephen Ra, Vishnu Sresht, and Saeed Saremi. 3D molecule generation by denoising voxel grids. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NeurIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

[43] Toshiki Ochiai, Tensei Inukai, Manato Akiyama, Kairi Furui, Masahito Ohue, Nobuaki Matsumori, Shinsuke Inuki, Motonari Uesugi, Toshiaki Sunazuka, Kazuya Kikuchi, Hideaki Kakeya, and Yasubumi Sakakibara. Variational autoencoder-based chemical latent space for large molecular structures with 3D complexity. *Communications Chemistry*, 6(1), November 2023. ISSN 2399-3669. doi: 10.1038/s42004-023-01054-6.

[44] Jaechang Lim, Seongok Ryu, Jin Woo Kim, and Woo Youn Kim. Molecular generative model based on conditional variational autoencoder for de novo molecular design. *Journal of Cheminformatics*, 10(1), July 2018. ISSN 1758-2946. doi: 10.1186/s13321-018-0286-7.

[45] Ryan J Richards and Austen M Groener. Conditional β-VAE for De Novo Molecular Generation, 2022. URL https://arxiv.org/abs/2205.01592.

[46] Minkai Xu, Shitong Luo, Yoshua Bengio, Jian Peng, and Jian Tang. Learning Neural Generative Dynamics for Molecular Conformation Generation, 2021. URL https://arxiv.org/abs/2102.10240.

[47] Yiheng Zhu, Zhenqiu Ouyang, Ben Liao, Jialu Wu, Yixuan Wu, Chang-Yu Hsieh, Tingjun Hou, and Jian Wu. MolHF: A Hierarchical Normalizing Flow for Molecular Graph Generation, 2023. URL https://arxiv.org/abs/2305.08457.

[48] Eyal Rozenberg and Daniel Freedman. Semi-Equivariant Continuous Normalizing Flows for Target-Aware Molecule Generation, 2022. URL https://arxiv.org/abs/2211.04754.

[49] Bruno Macedo, Inês Ribeiro Vaz, and Tiago Taveira Gomes. MedGAN: optimized generative adversarial network with graph convolutional networks for novel molecule design. *Scientific Reports*, 14(1), January 2024. ISSN 2045-2322. doi: 10.1038/s41598-023-50834-6.

[50] Andrew E. Blanchard, Christopher Stanley, and Debsindhu Bhowmik. Using GANs with adaptive training data to search for new molecules. *Journal of Cheminformatics*, 13(1), February 2021. ISSN 1758-2946. doi: 10.1186/s13321-021-00494-3.

[51] Ziqiao Zhang, Fei Li, Jihong Guan, Zhenzhou Kong, Liming Shi, and Shuigeng Zhou. *GANs for Molecule Generation in Drug Design and Discovery*, page 233–273. Springer International Publishing, 2022. ISBN 9783030913908. doi: 10.1007/978-3-030-91390-8_11.

[52] Arne Schneuing, Charles Harris, Yuanqi Du, Kieran Didi, Arian Jamasb, Ilia Igashov, Weitao Du, Carla Gomes, Tom L. Blundell, Pietro Lio, Max Welling, Michael Bronstein, and Bruno Correia. Structure-based drug design with equivariant diffusion models. *Nature Computational Science*, 4(12):899–909, December 2024. ISSN 2662-8457. doi: 10.1038/s43588-024-00737-x.

[53] Clemens Isert, Kenneth Atz, and Gisbert Schneider. Structure-based drug design with geometric deep learning. *Current Opinion in Structural Biology*, 79:102548, 2023. ISSN 0959-440X. doi: https://doi.org/10.1016/j.sbi.2023.102548.

[54] Gregor N. C. Simm and José Miguel Hernández-Lobato. A generative model for molecular distance geometry. In *Proceedings of the 37th International Conference on Machine Learning*, ICML 2020. JMLR.org, 2020.

[55] Chence Shi, Shitong Luo, Minkai Xu, and Jian Tang. Learning Gradient Fields for Molecular Conformation Generation. In Marina Meilă and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 9558–9568. PMLR, July 2021.

[56] Minkai Xu, Lantao Yu, Yang Song, Chence Shi, Stefano Ermon, and Jian Tang. GeoDiff: a Geometric Diffusion Model for Molecular Conformation Generation, 2022. URL https://arxiv.org/abs/2203.02923.

[57] Matthieu Kirchmeyer, Pedro O. Pinheiro, and Saeed Saremi. Score-based 3D molecule generation with neural fields, 2025. URL https://arxiv.org/abs/2501.08508.

[58] Yuyang Wang, Ahmed A. Elhag, Navdeep Jaitly, Joshua M. Susskind, and Miguel Angel Bautista. Swallowing the bitter pill: Simplified scalable conformer generation, 2024. URL https://arxiv.org/abs/2311.17932.

[59] Chao Hu, Song Li, Chenxing Yang, Jun Chen, Yi Xiong, Guisheng Fan, Hao Liu, and Liang Hong. ScaffoldGVAE: scaffold generation and hopping of drug molecules via a variational autoencoder based on multi-view graph neural networks. *Journal of Cheminformatics*, 15(1), October 2023. ISSN 1758-2946. doi: 10.1186/s13321-023-00766-0. URL http://dx.doi.org/10.1186/s13321-023-00766-0.

[60] Zhirui Liao, Lei Xie, Hiroshi Mamitsuka, and Shanfeng Zhu. Sc2Mol: a scaffold-based two-step molecule generator with variational autoencoder and transformer. *Bioinformatics*, 39 (1):btac814, 12 2022. ISSN 1367-4811. doi: 10.1093/bioinformatics/btac814. URL https://doi.org/10.1093/bioinformatics/btac814.

[61] Krzysztof Maziarz, Henry Richard Jackson-Flux, Pashmina Cameron, Finton Sirockin, Nadine Schneider, Nikolaus Stiefl, Marwin Segler, and Marc Brockschmidt. Learning to Extend Molecular Scaffolds with Structural Motifs. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=ZTsoE8G3GG.

[62] David Ha, Andrew Dai, and Quoc V. Le. HyperNetworks, 2016. URL https://arxiv.org/abs/1609.09106.

[63] Yinbo Chen and Xiaolong Wang. Transformers as meta-learners for implicit neural representations. In *European Conference on Computer Vision*, pages 170–187. Springer, 2022.

[64] Prafulla Dhariwal and Alex Nichol. Diffusion models beat GANs on image synthesis. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NeurIPS '21, Red Hook, NY, USA, 2021. Curran Associates Inc. ISBN 9781713845393.

[65] Grace Luo, Lisa Dunlap, Dong Huk Park, Aleksander Holynski, and Trevor Darrell. Diffusion hyperfeatures: searching through time and space for semantic correspondence. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NeurIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

[66] Alexander Quinn Nichol and Prafulla Dhariwal. Improved Denoising Diffusion Probabilistic Models. In Marina Meilă and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8162–8171. PMLR, July 2021.

[67] Mariana Pegrucci Barcelos, Suzane Quintana Gomes, Leonardo Bruno Federico, Isaque Antonio Galindo Francischini, Lorane Izabel da Silva Hage-Melim, Guilherme Martins Silva, and Carlos Henrique Tomich de Paula da Silva. *Lead Optimization in Drug Discovery*, pages 481–500. Springer International Publishing, Cham, 2022. ISBN 978-3-031-07622-0. doi: 10.1007/978-3-031-07622-0_19.

[68] William L. Jorgensen. Efficient Drug Lead Discovery and Optimization. *Accounts of Chemical Research*, 42(6):724–733, 2009. doi: 10.1021/ar800236t. URL https://doi.org/10.1021/ar800236t. PMID: 19317443.

[69] Stephanie Kay Ashenden. Chapter 6 - Lead optimization. In Stephanie Kay Ashenden, editor, *The Era of Artificial Intelligence, Machine Learning, and Data Science in the Pharmaceutical Industry*, pages 103–117. Academic Press, 2021. ISBN 978-0-12-820045-2. doi: https://doi.org/10.1016/B978-0-12-820045-2.00007-6. URL https://www.sciencedirect.com/science/article/pii/B9780128200452000076.

[70] Terry Kenakin. Predicting therapeutic value in the lead optimization phase of drug discovery. *Nature Reviews Drug Discovery*, 2(6):429–438, June 2003. ISSN 1474-1784. doi: 10.1038/nrd1110. URL http://dx.doi.org/10.1038/nrd1110.

[71] Simon Axelrod and Rafael Gómez-Bombarelli. GEOM, Energy-annotated Molecular Conformations for Property Prediction and Molecular Generation. *Scientific Data*, 9(1):185, April 2022. ISSN 2052-4463. doi: 10.1038/s41597-022-01288-4.

[72] Zhenqin Wu, Bharath Ramsundar, Evan N. Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S. Pappu, Karl Leswing, and Vijay Pande. MoleculeNet: A Benchmark for Molecular Machine Learning, March 2018.

[73] Walid Ahmad, Elana Simon, Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. ChemBERTa-2: Towards Chemical Foundation Models, 2022. URL https://arxiv.org/abs/2209.01712.

[74] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019. URL https://arxiv.org/abs/1907.11692.

[75] Yang You, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. Large Batch Optimization for Deep Learning: Training BERT in 76 minutes, 2020. URL https://arxiv.org/abs/1904.00962.

[76] RDKit: Open-source Chemoinformatics, 2025. URL https://www.rdkit.org.

[77] Noel M O' Boyle, Michael Banck, Craig A James, Chris Morley, Tim Vandermeersch, and Geoffrey R Hutchison. Open Babel: An Open Chemical Toolbox. *Journal of Cheminformatics*, 3(1), October 2011. ISSN 1758-2946. doi: 10.1186/1758-2946-3-33.

[78] Colin A. Grambow, Hayley Weir, Christian N. Cunningham, Tommaso Biancalani, and Kangway V. Chuang. CREMP: Conformer-rotamer ensembles of macrocyclic peptides for machine learning. *Scientific Data*, 11(1), August 2024. ISSN 2052-4463. doi: 10.1038/s41597-024-03698-y. URL http://dx.doi.org/10.1038/s41597-024-03698-y.

[79] Janani Durairaj, Yusuf Adeshina, Zhonglin Cao, Xuejin Zhang, Vladas Oleinikovas, Thomas Duignan, Zachary McClure, Xavier Robin, Gabriel Studer, Daniel Kovtun, Emanuele Rossi, Guoqing Zhou, Srimukh Veccham, Clemens Isert, Yuxing Peng, Prabindh Sundareson, Mehmet Akdel, Gabriele Corso, Hannes Stärk, Gerardo Tauriello, Zachary Carpenter, Michael Bronstein, Emine Kucukbenli, Torsten Schwede, and Luca Naef. PLINDER: The protein-ligand interactions dataset and evaluation resource. *bioRxiv*, 2024. doi: 10.1101/2024.07.17.603955.

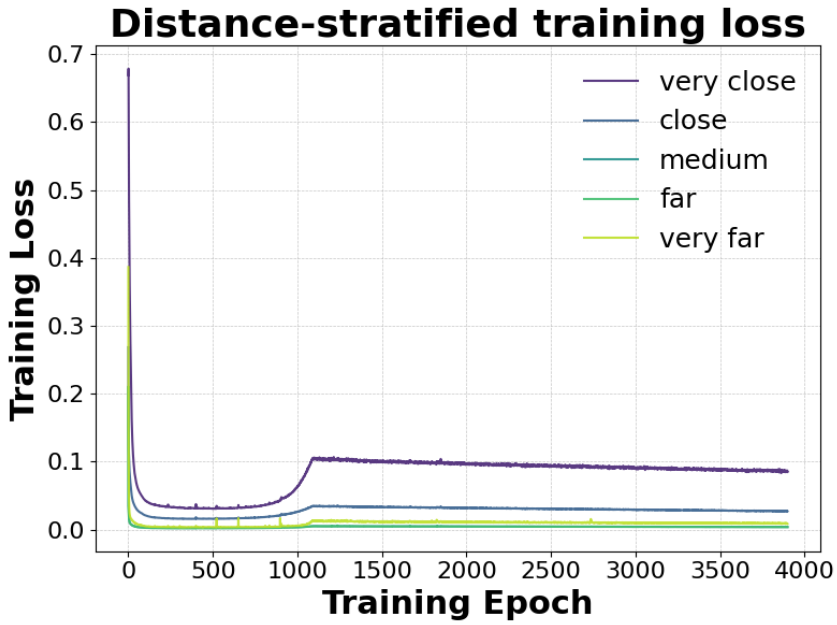

Figure 7: Training curves stratified by the distance from the query point to the nearest atom. Loss is tracked separately for five distance bins: *very close*, *close*, *medium*, *far*, and *very far*. The *very close* regime is the most challenging due to the high-frequency variation near the molecular surface. Loss in each bin initially decreases, then increases due to the progressive diffusion noise schedule.

A Pretraining Results

In Figure 8, we show distance-stratified training curves that break down the training loss based on the proximity of each query point to its nearest atom. Specifically, we divide query points into five distance bins: *very close*, *close*, *medium*, *far*, and *very far*; the percentile bins are given by $[0\% - 2\%], [2\% - 33\%], [33\% - 66\%], [66\% - 98\%], [98\% - 100\%]$. For each bin, we track the training loss separately over time.

These curves reveal that the *very close* category is the most challenging to learn. This regime contains query points nearest the atoms of the molecule, where high-frequency geometric variation is most pronounced. In such regions, the direction to the atom can change abruptly across small distances, introducing sharp changes in the direction field. Accurately modeling these points requires the network to capture fine-grained structural detail. As such, improvements in the *very close* bin are strong indicators that the model is learning the geometry of the molecule with high fidelity.

Observe that the loss in all bins initially decreases but then rises again during training before plateauing. This behavior is due to the training curriculum we employ, in which the diffusion noise level is gradually increased across epochs. Early in training, the model sees easier (low-noise) examples, leading to rapid improvements. As training progresses, higher levels of noise are introduced, making the task more difficult and increasing the overall loss. We describe this curriculum in more detail in Section B.

B Implementation Details

Hypernetwork Implementation Details. As described in Equation 3 and Section 3.4, the hypernetwork is a function H_ϕ that maps a noised MDF $\mathbf{F}_t(Q)$, a set of query points Q, and a diffusion timestep t to the parameters θ of a Molecular Neural Field (MNF) $\mathbf{F}_\theta$.

To encode the input MDF $\mathbf{F}_t(Q)$, we begin by drawing a bounding box around the molecule and partitioning the enclosed volume into a uniform $c \times c \times c$ 3D grid. A fixed number of query points are uniformly sampled within each cell. For each query point $\mathbf{q}_j$, we first apply a learned positional encoding to its 3D coordinate and then concatenate the resulting embedding with the corresponding

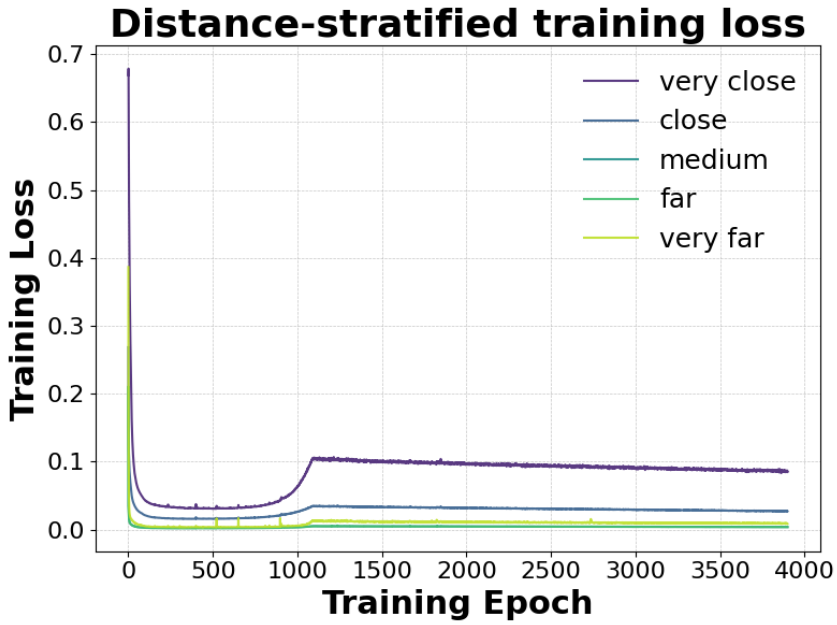

Figure 8: Training curves stratified by the distance from the query point to the nearest atom. Loss is tracked separately for five distance bins: *very close*, *close*, *medium*, *far*, and *very far*. The *very close* regime is the most challenging due to the high-frequency variation near the molecular surface. Loss in each bin initially decreases, then increases due to the progressive diffusion noise schedule.

noised field vector $\mathbf{F}_t(\mathbf{q}_j)$. These encoded query points form the input tokens to the transformer-based hypernetwork.

The diffusion timestep t is embedded using sinusoidal Fourier features, as in [15], and broadcast across all tokens. The transformer processes the sequence using self-attention layers to capture spatial dependencies across grid cells, followed by MLPs. The final token representations are pooled and passed through linear projections to produce the complete set of MLP weights θ for the downstream field $\mathbf{F}_\theta$.

From scalar fields to vector fields. Recall from §3.5 that the hypernetwork takes a noised *vector* field as input, but the loss is computed on an output *vector* field. One challenge with this formulation is that it introduces a mismatch between the model's input and output during generation: the input to the hypernetwork is a direction field, but the output is a scalar field. To resolve this, note that the direction field can be recovered from the distance field as follows:

$$\nabla_{\mathbf{q}} \frac{1}{2} \left(f^{(k)}(\mathbf{q}) \right)^2 = \nabla_{\mathbf{q}} \frac{1}{2} \left\| \mathbf{a}^{(k)}_{\text{nearest}} - \mathbf{q} \right\|^2 = \mathbf{a}^{(k)}_{\text{nearest}} - \mathbf{q} = \mathbf{F}^{(k)}(\mathbf{q}) \tag{7}$$

Here, $\mathbf{a}^{(k)}_{\text{nearest}}$ denotes the nearest atom of type k to point $\mathbf{q}$. This allows us to compute $\mathbf{F}^{(k)}(\mathbf{q})$ from the predicted scalar field using automatic differentiation. During generation, we apply this transformation at each timestep to produce the direction field needed for the next step of the reverse diffusion.

Training curriculum. We train with a curriculum on the noise level. For the first 100 epochs of training, the maximum noise level t that we add is 10 (out of a total of $T = 1000$ noise levels). For every subsequent epoch, the maximum possible noise level is increased by 1, until $T = 1000$ is reached; we find that this accelerates training in practice; we show ablation studies in §I

Computational Complexity. Since our method implicitly models the molecule via the MDF, the computational complexity scales only with the number of query points, not the size of the molecule. During training and inference, we keep the number of query points constant so memory complexity is constant.

Hyperparameters. We use the LAMB optimizer [75] with a learning rate of 1×10^{-2}, a batch size of 5500, and dropout set to 0.1. Both the noise schedule and the learning rate follow a cosine

Figure 9: Overview of how we use HyDiF as a foundational model for property prediction. In practice, we always extract the central activation within the stack of transformer layers, e.g., given 26 transformer blocks, we extract the 14th intermediate activation.

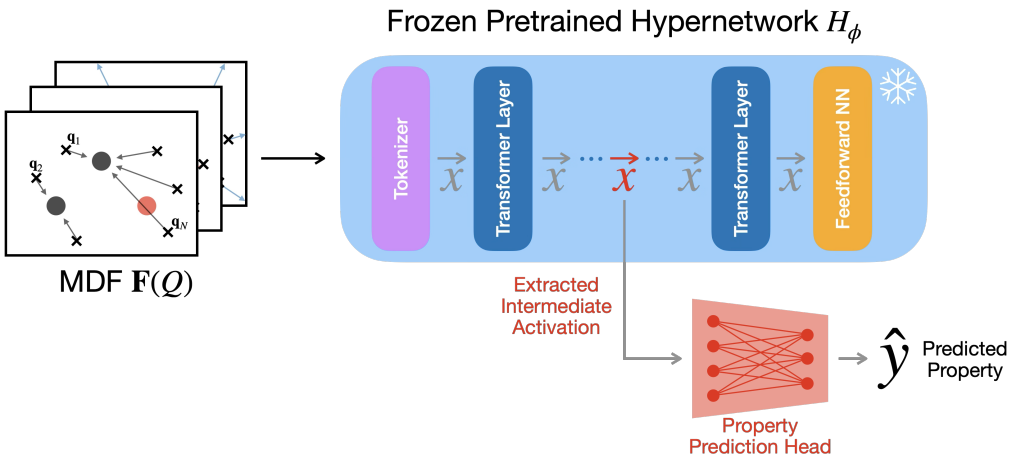

Table 3: Overview of the subset of MoleculeNet tasks that we perform property prediction on.

Task Name	Task Type	Train Size	Description
BBBP	classification	1,631	blood-brain barrier penetration
BACE	classification	1,210	human β-secretase 1 binding
Lipo	regression	3,360	lipophilicity
ESOL	regression	902	water solubility
FreeSolv	regression	513	hydration free energy

decay. The hypernetwork consists of 26 transformer layers and contains approximately 52 million parameters. All models are trained on NVIDIA H100 GPUs for 3500 epochs.

HyDiF as a feature extractor We illustrate how we leverage internal representations of HyDiF as feature representations of molecules for downstream molecular property prediction in Figure 9. We pass an unnoised MDF for the molecule of interest into a frozen pretrained hypernetwork and extract an intermediate activation, which we average over the per-cell tokens. We then pass this activation into a property prediction head, which for us is a two layer MLP with $\tanh$ activations and hidden layer size 512. When training on MoleculeNet tasks, since MoleculeNet datasets represent molecules as SMILES strings rather than conformers, we generate a conformer for each SMILES string using ETKDG.

C Property Prediction Tasks

In Table 3, we provide a brief summary of the MoleculeNet tasks run in §4.2.

D From MDFs to Molecules

While the output of HyDiF is an MDF for a conformer, at inference time it is more desirable to have a SMILES or graph representation of a molecule. Our method of parsing MDFs into molecules involves two steps: (1) recovering atomic point clouds from MNFs and (2) recovering molecular bond information from atomic point clouds.

To recover atomic coordinates from MNFs, we find minima of the generated distance field and use these minima as atomic coordinates. We do this by sampling 4096 random query points in the volume of a molecule, computing the gradient of the predicted field at these query points, and using this gradient information to iteratively find the minima of field until a tolerance is met.

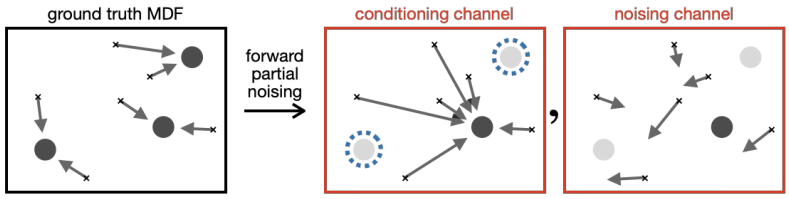

Figure 10: We extend our method to support conditional generation by expanding the input to two channels: the first encodes the field corresponding to the conditional atom, while the second initializes the diffusion process from pure noise.

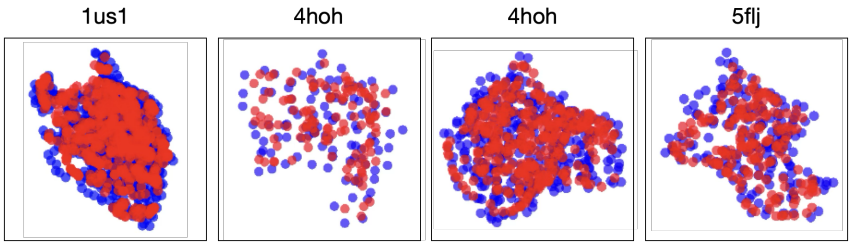

Figure 11: Four uncurated comparisons of ground truth alpha carbon coordinates of proteins from PLINDER (in blue) compared with coordinates reconstructed from denoising a noised MDF (in red). We see that in all four cases, we are able to reconstruct the overall shape of the protein. This illustrates the scalability of HyDiF to large biomolecules. Subplot captions are Protein Data Bank identifiers.

To recover molecular bond information from atomic point clouds, we use a combination of the standard chemoinformatics toolkits RDKit [76] and OpenBabel [77] bond determination algorithms to convert atomic point clouds into RDKit-parsable molecules.

E Molecular Inpainting and Scaffold-Constrained Generation

Our training pipeline for partial noising is identical to that in §3.5, except in addition to a noised MDF, we also include a *conditioning MDF* as an input to the hypernetwork. The conditioning MDF is computed by randomly deleting a subset of atoms from a conformer and recomputing the MDF for the partial molecule. This allows the network to learn to denoise a fully noised MDF *conditioned on* a partial molecule being held constant (see Figure 10).

F Scaling to Larger Biomolecules

HyDiF scales to large biomolecules in a way that current baseline architectures cannot. Most molecular generative models—such as EDM, GeoLDM, and VoxMol—are limited to small molecules with fewer than 100 heavy atoms. This is primarily due to architectural bottlenecks. GNN-based models like EDM and GeoLDM construct fully connected molecular graphs, resulting in $\mathcal{O}(n^2)$ pairwise interactions for n atoms, which becomes computationally expensive as molecular size grows. VoxMol represents molecules using dense 3D voxel grids, which are memory-intensive and do not scale well to larger biomolecules. As shown in [57], VoxMol fails to scale to the CREMP dataset [78], which includes macrocyclic peptides with an average of 74 heavy atoms.

Proteins contain hundreds to thousands of atoms. To evaluate the scalability of HyDiF, we train our diffusion pipeline (see §3.5) on alpha carbon traces from the PLINDER dataset [79], a collection of 3D protein structures. Figure 11 shows reconstructed proteins from the training set, generated by noising and denoising their molecular direction fields. In each case, we are able to recover the overall structure of the protein, demonstrating that HyDiF can model large-scale molecular geometry.

The key insight enabling scalability is that the computational cost of HyDiF grows quadratically with the number of query points, rather than the number of atoms in the molecule. This decouples

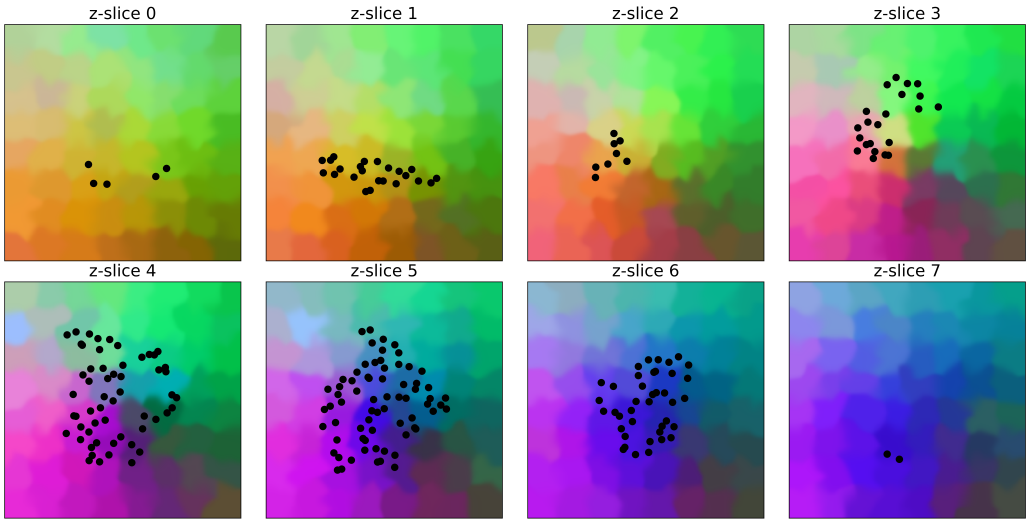

Figure 12: Visualization of spatially localized features for a protein (whose alpha carbons are plotted in black). The volume inhabited by the protein is separated into 8 z-slices. The latent feature corresponding to each query point is projected and scaled into $[0, 1]^3$ and interpreted as an RGB values; individual pixels of our visualization are colored by a weighted interpolation of neighboring query points.

computational complexity from molecular size: we train on large proteins by sampling a sparse set of query points, as the model sees each protein multiple times over training.

G Protein Feature Visualization

In Figure 12, we demonstrate how HyDiF is able to produce spatially localized features for molecules. We input a protein structure through a pretrained HyDiF and extract the intermediate activations from the hypernetwork. We are able to visualize the features at the per-query point resolution, after projecting from a high dimensional latent space to $\mathbb{R}^3$. The features smoothly change gradually in space.

H Algorithms for Training and Generation

Here we present pseudocode for training HyDiF (Algorithm 1) and generating sample conformers after training (Algorithm 2).

Algorithm 1 illustrates the noise scale curriculum we employ in training. For the first 100 epochs, we cap the maximum noise level T at $10/1000$; for each epoch beyond the first 100, we increase the maximum noise level by 1 until we reach $T = 1000$.

Algorithm 2 illustrates how we generate novel molecules from a trained HyDiF. We start by sample Gaussian noise and taking that to be a molecule noised at time step $t = 1000$. We then ask the hypernetwork to denoise it to $t = 0$, then noise the denoised prediction by noise level $t = 999$ and repeat. Since the outputs of the hypernetwork are the weights to an implicit *distance field*, we must first compute the gradients with respect to the query points to convert it back to a *direction field* that we can pass back into the hypernetwork.

I Ablation Studies

We conduct two ablation experiments to better understand key design choices in HyDiF: (1) the impact of using a noise-level curriculum during training, and (2) the effect of the input field type provided to the hypernetwork.

Algorithm 1 Training Loop for HyDiF

$Q \leftarrow$ random query points
$\mathbf{F}(Q) \leftarrow$ ground truth MDF
for $\text{epoch} = 1, 2, \ldots$ **do**
  if $\text{epoch} \leq 100$ **then**
    $T \leftarrow 10$
  else
    $T \leftarrow \max(\text{epoch} - 100 + 10, 1000)$
  end if
  $t \leftarrow$ random uniform sample from $\{0, 1, \ldots, T\}$
  $\alpha_t \leftarrow$ cosine noise schedule
  $\boldsymbol{\epsilon} \leftarrow \mathcal{N}(0, I)$ ▷ sample Gaussian noise
  $\mathbf{F}_t(Q) \leftarrow \sqrt{\overline{\alpha}_t} \cdot \mathbf{F}(Q) + \sqrt{1 - \overline{\alpha}_t} \cdot \boldsymbol{\epsilon}$ ▷ noise the MDF
  $\theta \leftarrow H_\phi(\mathbf{F}_t(Q), Q, t)$ ▷ pass noised MDF into H_ϕ to get predicted MNF parameters
  $\mathcal{L}_{\text{dist}} \leftarrow \frac{1}{|Q|} \sum_{\mathbf{q} \in Q} \sum_{k=1}^{K} \left| f_\theta^{(k)}(\mathbf{q}) - f^{(k)}(\mathbf{q}) \right|$ ▷ compute loss against ground truth distance field
  $\phi \leftarrow$ gradient update ▷ update hypernetwork parameters from $\mathcal{L}_{\text{dist}}$
end for

Algorithm 2 Generation Procedure for HyDiF

$t \leftarrow 1000$
$Q \leftarrow$ random query points
$F_t(Q) \leftarrow$ Gaussian noise
for $t = 1000, 999, \ldots, 1$ **do**
  $\theta \leftarrow H_\phi(F_t(Q), Q, t)$
  $\mathbf{F}_\theta(Q) \leftarrow \nabla_Q f_\theta(Q)$
  $\alpha_{t-1} \leftarrow$ cosine noise schedule
  $\mathbf{F}_{t-1}(Q) \leftarrow \sqrt{\overline{\alpha}_{t-1}} \cdot \mathbf{F}_\theta(Q) + \sqrt{1 - \overline{\alpha}_{t-1}} \cdot \boldsymbol{\epsilon}$
end for

Effect of Curriculum on Noise Scale. In Figure 13, we compare training with and without a noise-level curriculum. In the curriculum setting, the maximum diffusion noise level is gradually increased over the course of training, starting from low noise levels and eventually reaching pure noise. This strategy leads to faster convergence and improved training stability compared to the baseline with a fixed maximum noise level throughout. Notably, we observe a rise in training loss around epoch 800 in the curriculum setting, which coincides with the introduction of highly noised inputs that are intrinsically more difficult to denoise.

Effect of Input Field Type. In Figure 14, we assess the impact of the type of input field—either a distance field or a direction field—provided to the hypernetwork. We find that using a direction field as input leads to lower training loss and more effective learning. We attribute this improvement to the fact that direction fields contain higher-frequency signals compared to distance fields, providing a more expressive representation of local molecular structure.

In both experiments, we report ℓ_1 loss on the subset of query points that are *very close* to the ground truth atomic coordinates—defined as the top 2% of query points with the smallest distance to any atom. These points represent the highest-frequency components of the MDF, and accurately modeling them is critical for capturing precise atomic structure.

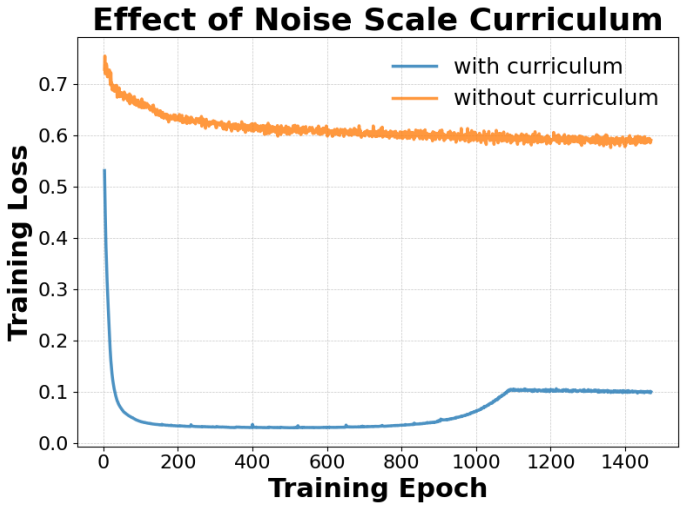

Figure 13: Ablation study illustrating the benefit of using a training curriculum (blue) on the diffusion noise scale versus no curriculum (orange). The curriculum improves convergence and early performance by exposing the model to gradually harder denoising tasks. The rise in loss near epoch 800 reflects the model being trained on highly noised inputs.

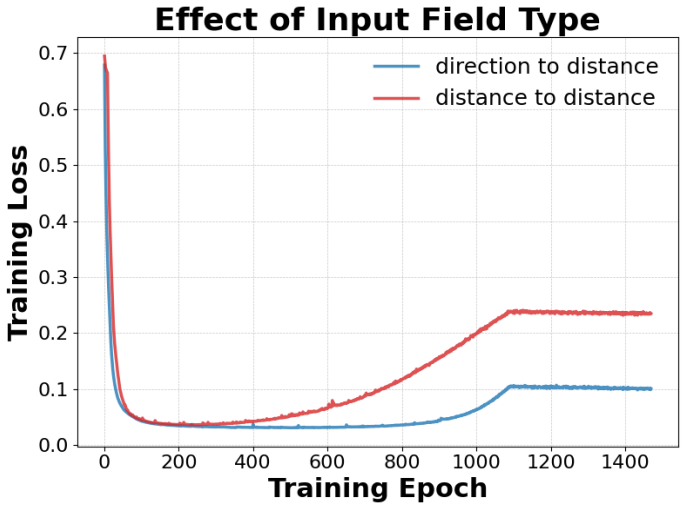

Figure 14: Ablation study comparing the effect of input field type. Using a *direction* field (blue) as input leads to improved training over a *distance* field (red), likely due to the richer high-frequency information present in direction fields.