Master 2 Internship Presentation

Creation of a 3D Molecule Generation Model Targeting a Protein of Interest

Victor GERTNER

IPP/TSP
Iktos

02/04/2024 to 30/09/2024

Company Internship Tutors: Vincent Bouttier / Hamza Tajmouati

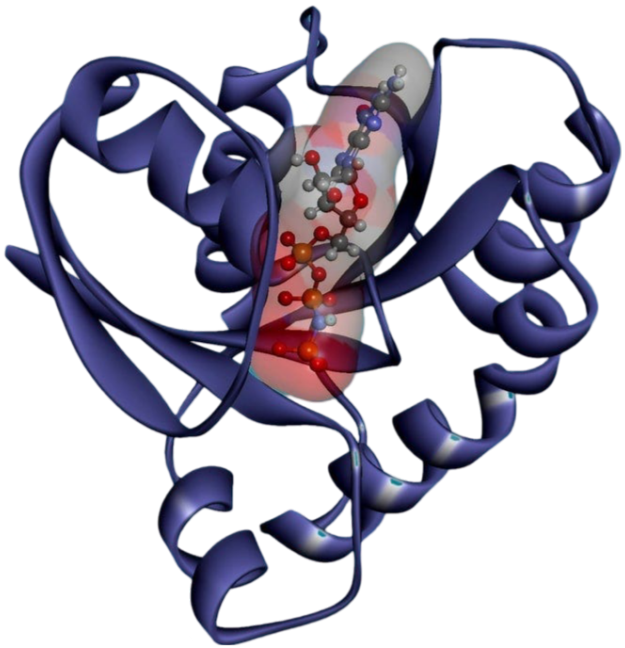
School Internship Tutor: Sholom Schechtman

Table of Contents

Introduction

Itkos

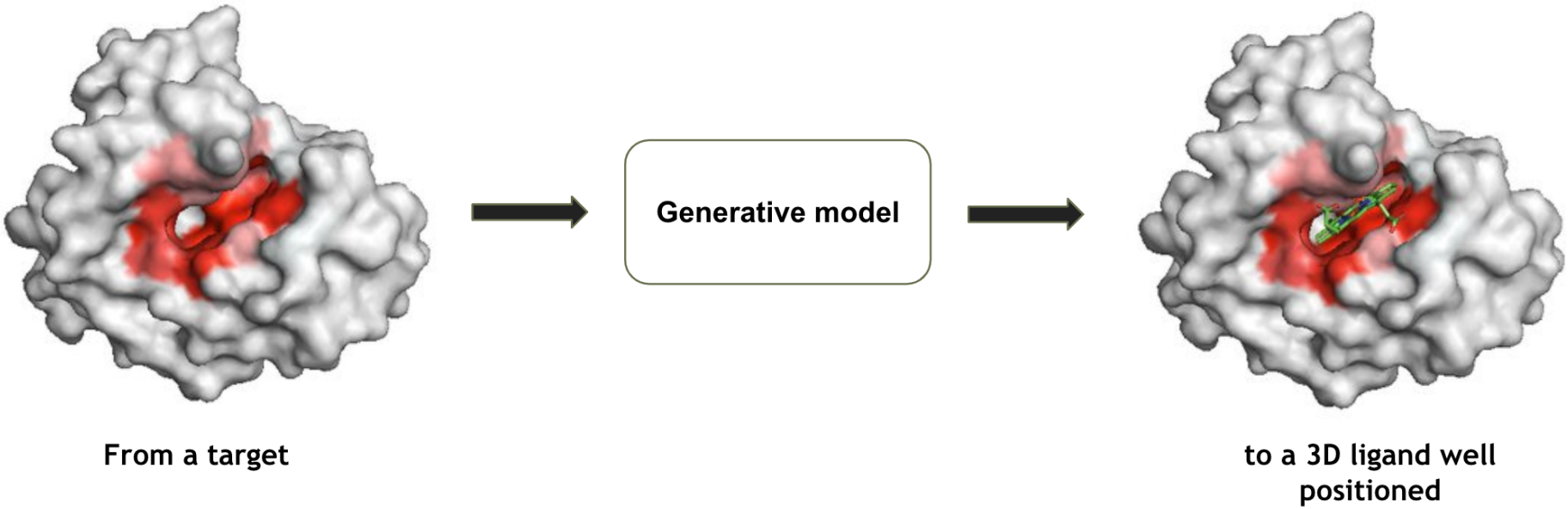
Proteins, Ligands...



Proteins are large, complex molecules made of amino acids, essential for cell structure, function, and regulation. They catalyze reactions, transport molecules, and respond to signals.

Ligands are molecules that bind to proteins to activate or inhibit their functions. They play key roles in processes like cell signaling and enzyme activity.

High View


Generative model
From a target
to a 3D ligand well positioned

Generative Approaches

- **3D Scoring:** Generate molecules in 2D and use a 3D scoring function to indirectly optimize the 3D conformation without explicit 3D generation, as in GenScore.

Generative Approaches

- **3D Scoring:** Generate molecules in 2D and use a 3D scoring function to indirectly optimize the 3D conformation without explicit 3D generation, as in GenScore.
- **2D then 3D:** Generate molecules in 2D, then convert them to 3D, as in DiffDock.

Generative Approaches

- **3D Scoring:** Generate molecules in 2D and use a 3D scoring function to indirectly optimize the 3D conformation without explicit 3D generation, as in GenScore.
- **2D then 3D:** Generate molecules in 2D, then convert them to 3D, as in DiffDock.
- **3D Generation:** Directly generate the 3D structure of the molecule, bypassing the 2D representation entirely, such as in DiffSBDD.

The internship

High View

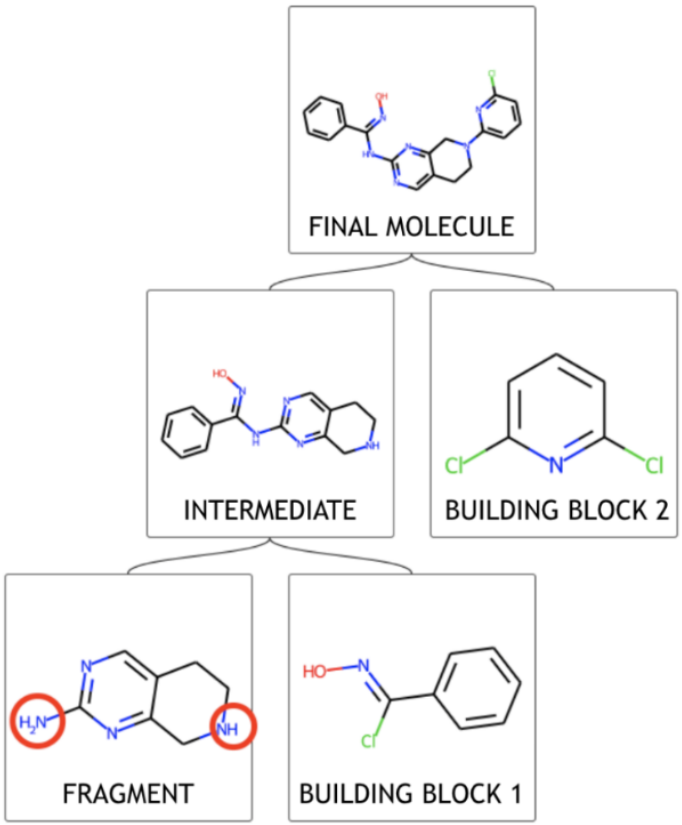
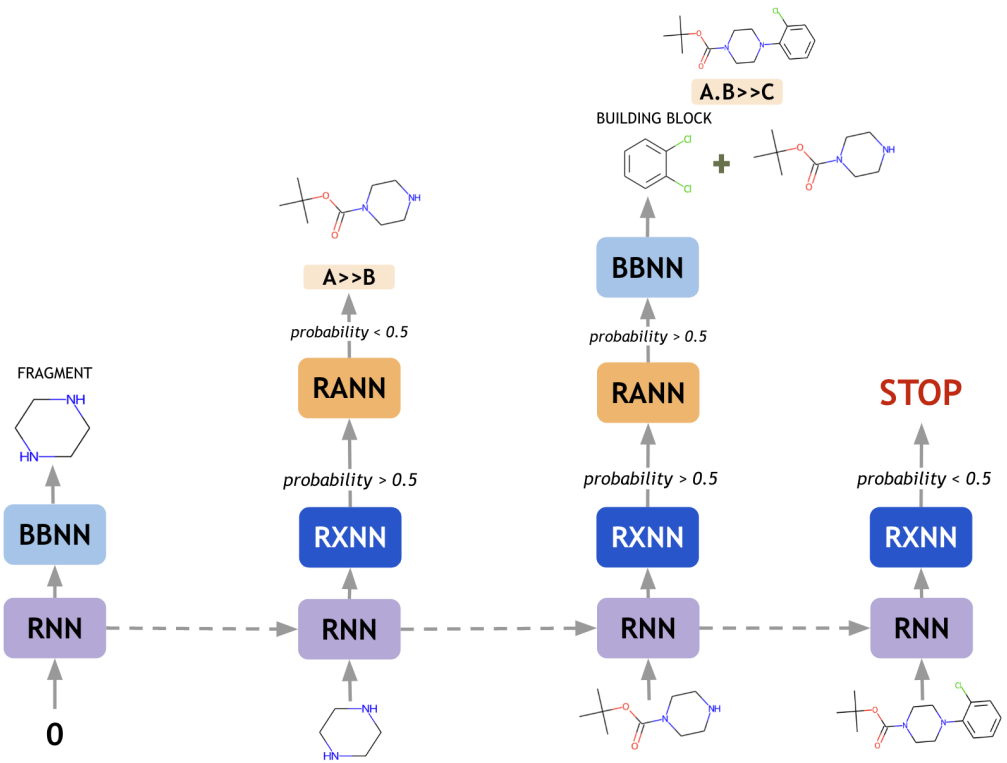


Figure: Growing Optimizer Generation

Iktos has developed the **Growing Optimizer**, a model for reaction-based ligand generation conditioned on a pocket. Currently, it uses building blocks without considering 3D data. **The goal is to enhance this model by incorporating 3D coordinates, optimizing the 3D affinity between the ligand and the pocket.** The task involves using 2D data to predict 3D structures, improving the generation process.

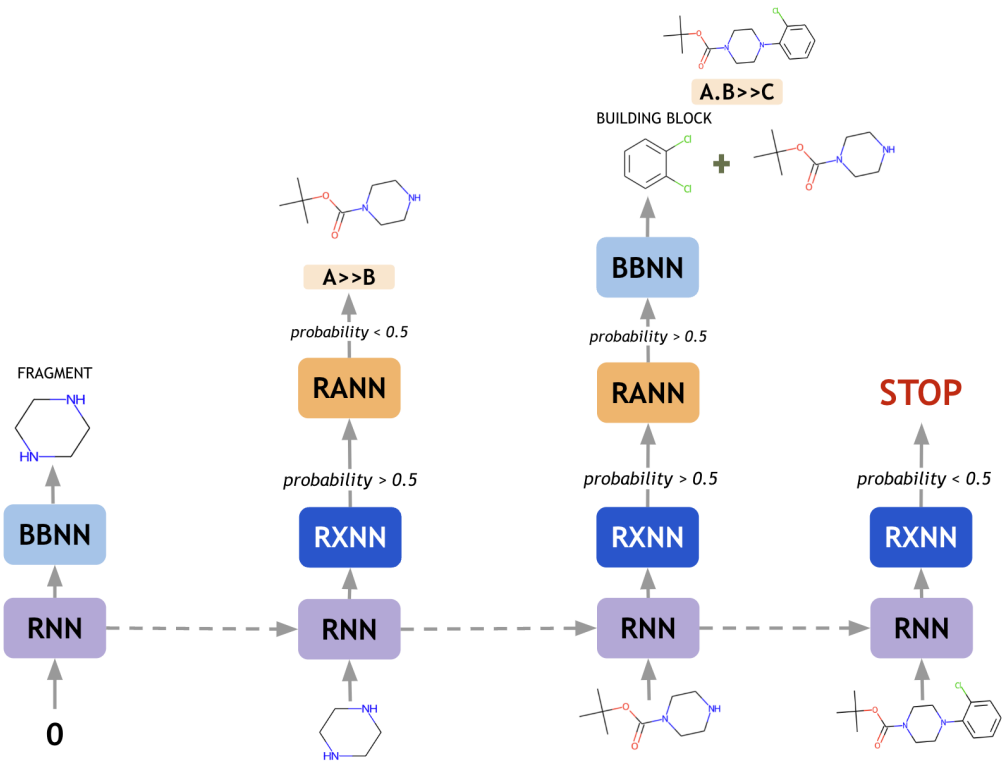
Precise architecture



RNN: Processes past reactions.

Figure: Growing Generator Blocks

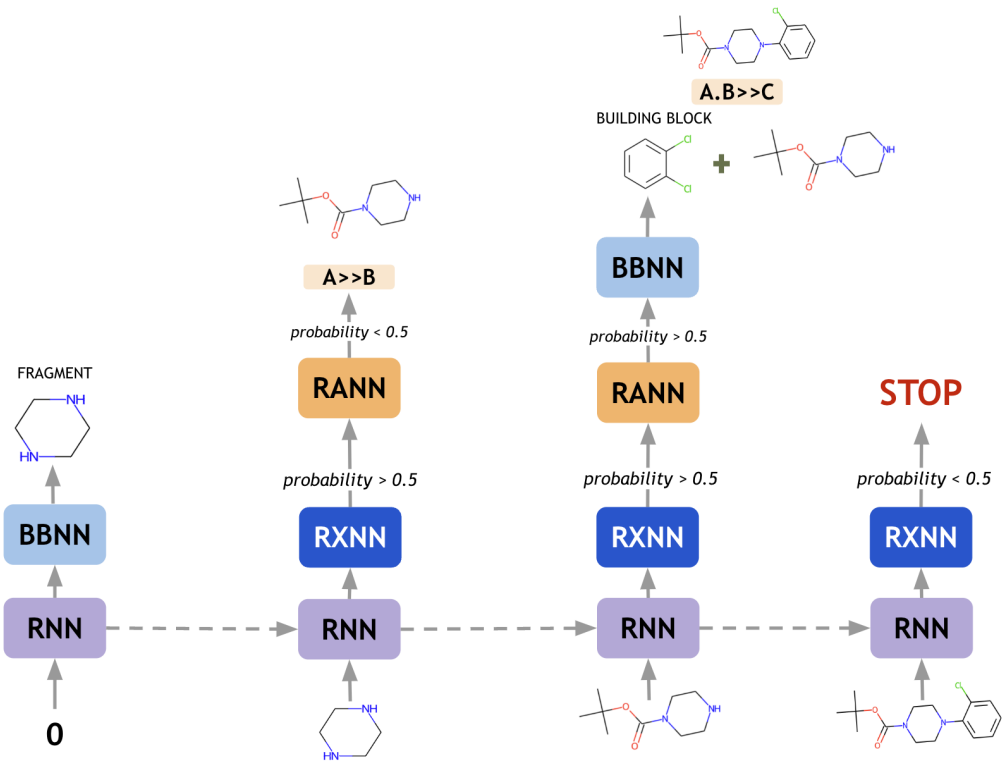
Precise architecture



RNN: Processes past reactions.
RXNN: Decides to stop or continue.

Figure: Growing Generator Blocks

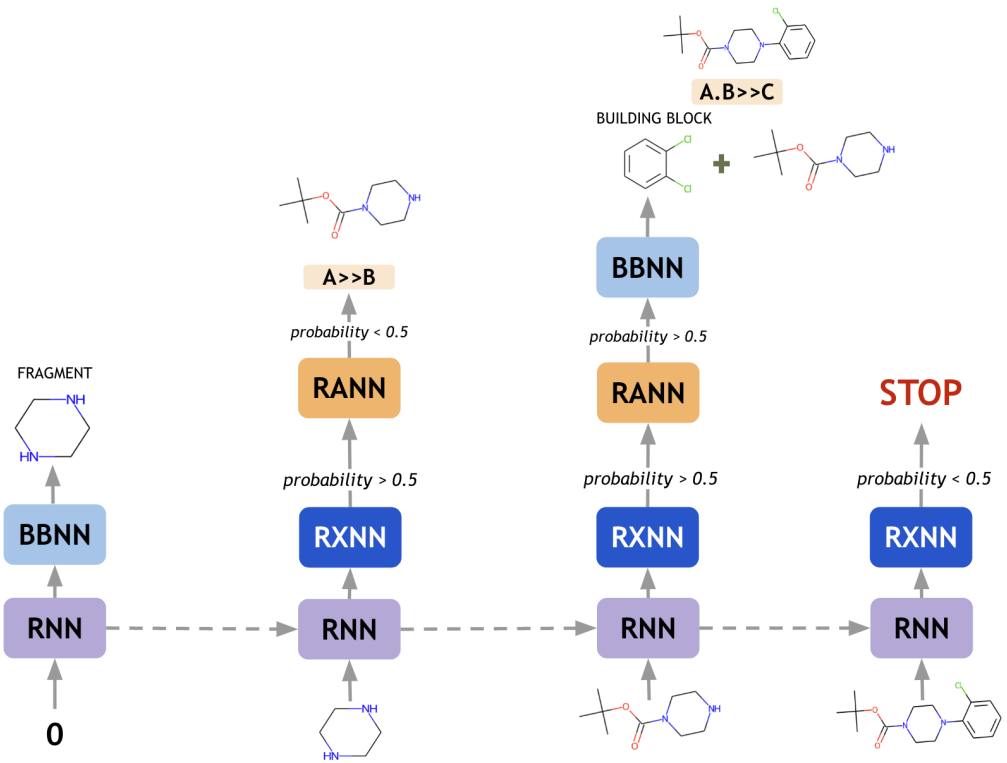
Precise architecture



RNN: Processes past reactions.
RXNN: Decides to stop or continue.
RANN: Selects reaction type.

Figure: Growing Generator Blocks

Precise architecture



RNN: Processes past reactions.
RXNN: Decides to stop or continue.
RANN: Selects reaction type.
BBNN: Chooses building blocks.

Figure: Growing Generator Blocks

The main idea

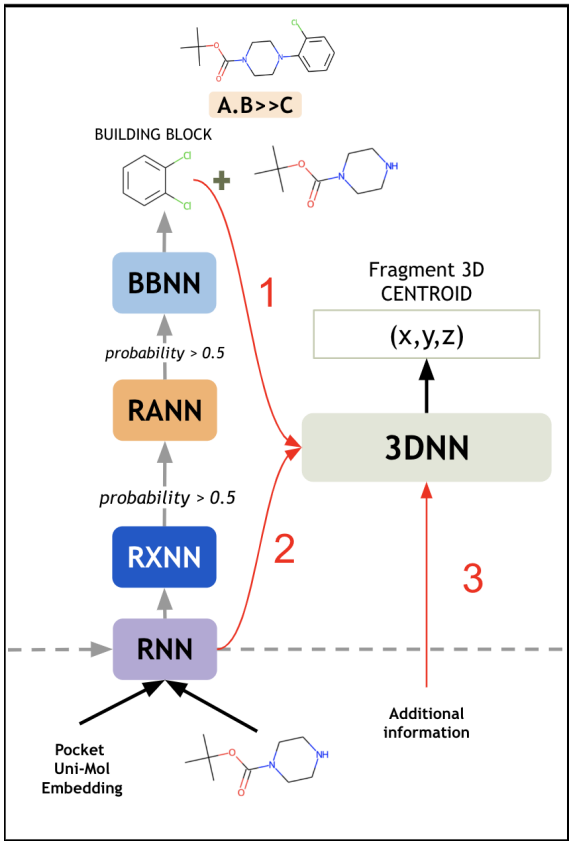


Figure: 3DNN Imbrication into the growing optimizer

The data - 2D

Growing tree: Full iteration of the growing optimizer → Access to the embedding of the building blocks, representation of the protein...

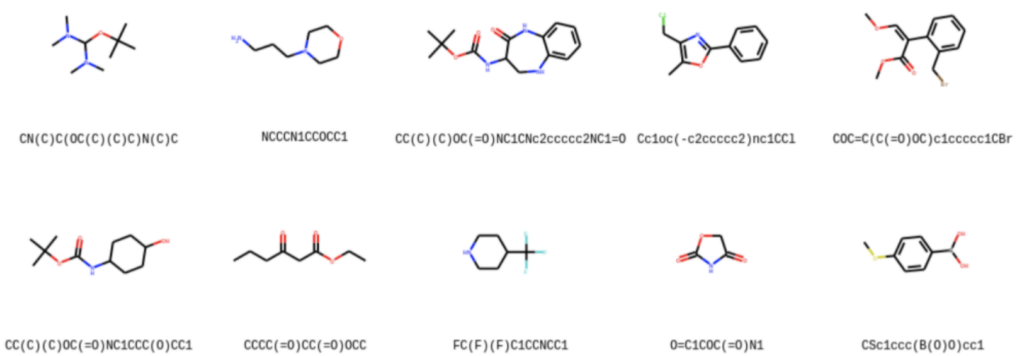


Figure: Example of molecules from the pool of building block and their associated SMILES notation

In total, there are **107,552 growing trees**, representing **251,201** $A \cdot B \rightarrow C$ **reactions**.

The data - 3D

On the 3D module side, the 3D coordinates of the complexes come from the publicly available database **PDBbind**.

Hence, for each growing tree : **Building Blocks ID/Embedding, Pocket Representation, 3D coordinates** (either of the final ligand, or splited in fragments for atoms that remain in the final molecule) are provided and can be used by both the 2D module or the 3D module.

Molecule representation

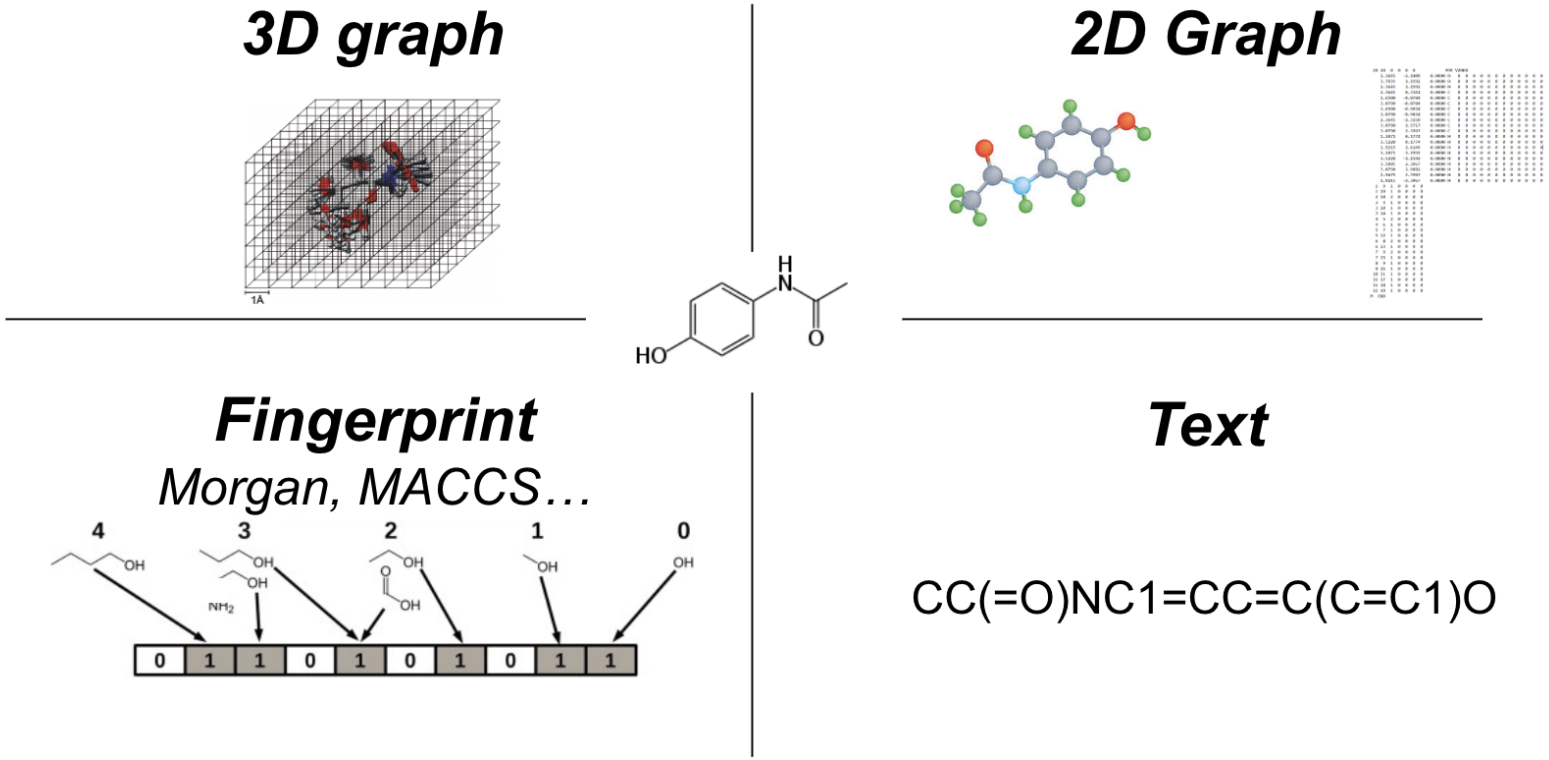


Figure: Different representation of molecules

Molecule representation - 2

A molecular graph $G = (V, E)$ is defined where:

- V is the set of nodes (atoms), where each node i has:
  - Node feature $s_i \in \mathbb{R}^d$
  - Coordinate $x^i \in \mathbb{R}^3$, representing the 3D position
- E is the set of edges (bonds), with a_{ij} representing the edge feature between nodes i and j.

Centroid prediction

The task

The task focuses on predicting the centroids of fragments. The centroid is defined as:

$$\left(\frac{1}{n} \sum_{i=1}^{n} x_i, \frac{1}{n} \sum_{i=1}^{n} y_i, \frac{1}{n} \sum_{i=1}^{n} z_i \right)$$

During the process $A \cdot B \rightarrow C$, the model predicts the centroid of fragment B.

Initial Approach

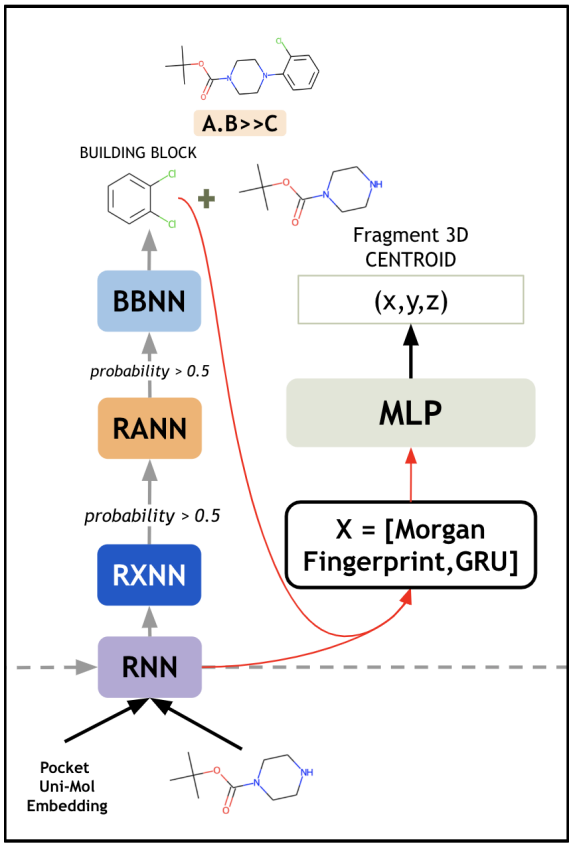


Figure: Initial Approach using an MLP

An initial design by Iktos used an MLP. It took as input the concatenation of the RNN output and the Morgan fingerprint, predicting the centroid of the new building block.

Equivariant Graph Neural Network

The model is defined as follows:

$$m_{ij} = \phi_e\left(s_i, s_j, \|x^i - x^j\|^2, a_{ij}\right) \tag{1}$$

Equivariant Graph Neural Network

The model is defined as follows:

$$m_{ij} = \phi_e\left(s_i, s_j, \|x^i - x^j\|^2, a_{ij}\right) \tag{1}$$

$$x^i_{\text{new}} = x^i + C\sum_{j\neq i}\left(x^i - x^j\right)\phi_x\left(m_{ij}\right) \tag{2}$$

Equivariant Graph Neural Network

The model is defined as follows:

$$m_{ij} = \phi_e\left(s_i, s_j, \|x^i - x^j\|^2, a_{ij}\right) \tag{1}$$

$$x^i_{\text{new}} = x^i + C\sum_{j\neq i}\left(x^i - x^j\right)\phi_x\left(m_{ij}\right) \tag{2}$$

$$m_i = \sum_{j\neq i} m_{ij} \tag{3}$$

Equivariant Graph Neural Network

The model is defined as follows:

$$m_{ij} = \phi_e\left(s_i, s_j, \|x^i - x^j\|^2, a_{ij}\right) \tag{1}$$

$$x^i_{\text{new}} = x^i + C\sum_{j\neq i}\left(x^i - x^j\right)\phi_x\left(m_{ij}\right) \tag{2}$$

$$m_i = \sum_{j\neq i} m_{ij} \tag{3}$$

$$s_i^{\text{new}} = \phi_h\left(s_i, m_i\right) \tag{4}$$

EGNN for molecule generation

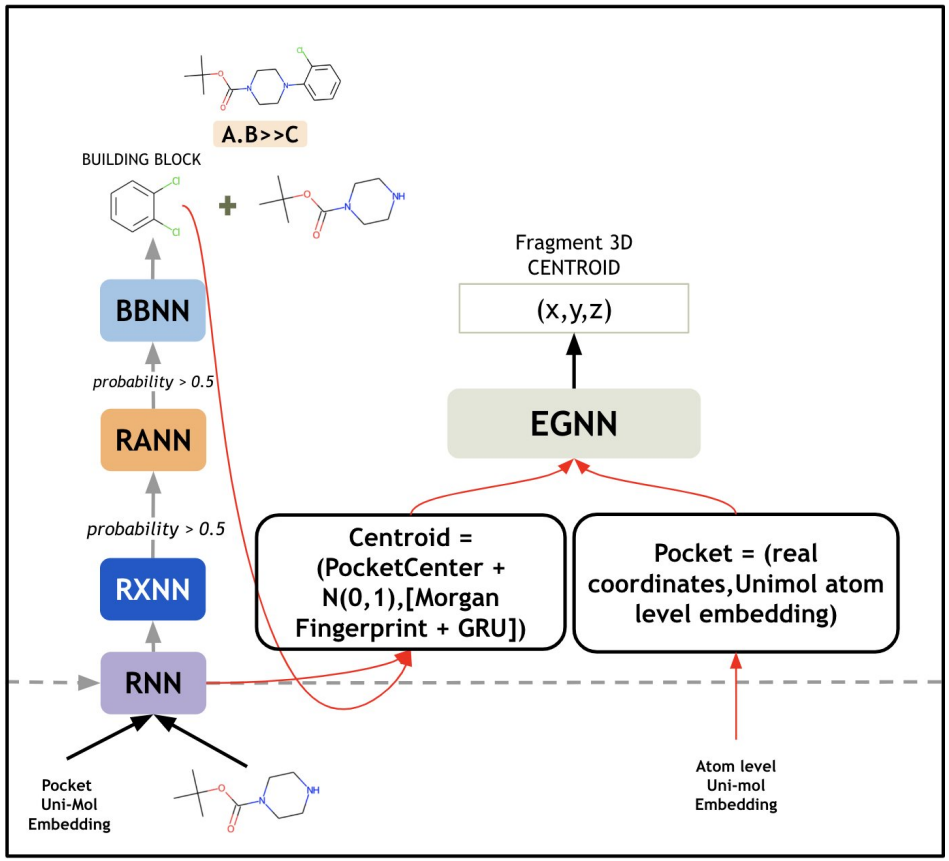


Figure: Integration of the EGNN in the Growing Optimizer inspired from DiffSBDD

Tweaks from DiffSBDD

- **Pocket Representation**: The pocket is only represented once through the whole generative process: **Memory gain!**

Tweaks from DiffSBDD

- **Pocket Representation**: The pocket is only represented once through the whole generative process: **Memory gain!**
- **Objective**: The objective will be to predict (x,y,z)/5: **More stability!**

Tweaks from DiffSBDD

- **Pocket Representation**: The pocket is only represented once through the whole generative process: **Memory gain!**
- **Objective**: The objective will be to predict (x,y,z)/5: **More stability!**
- **Autoregressive generation**: The centroid at step i-1, i-2,.., 1 are used for the centroid i: **Better Prediction!**

Results - 1

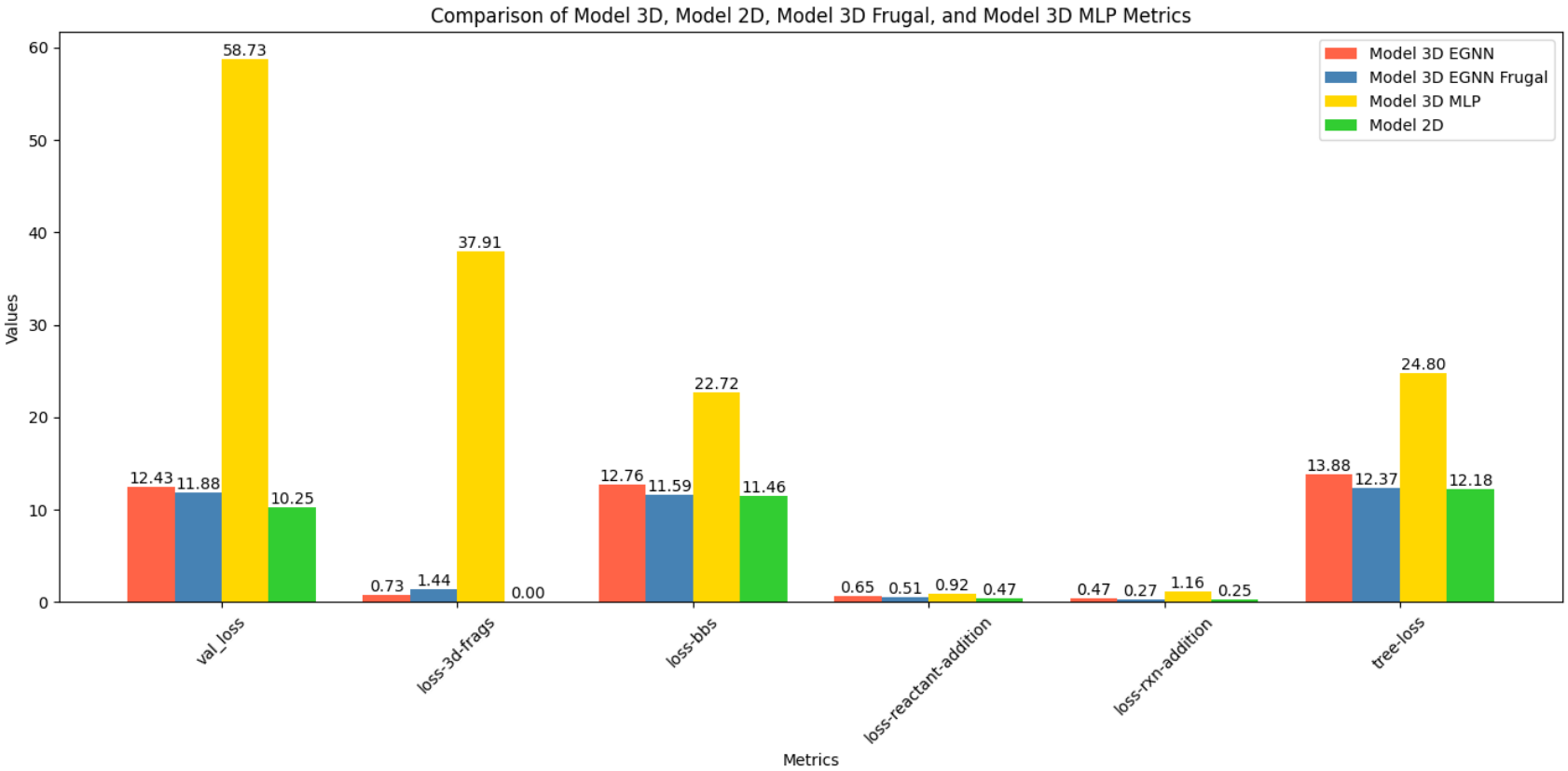


Figure: Assessment of EGNN centroid performance

Results - 2

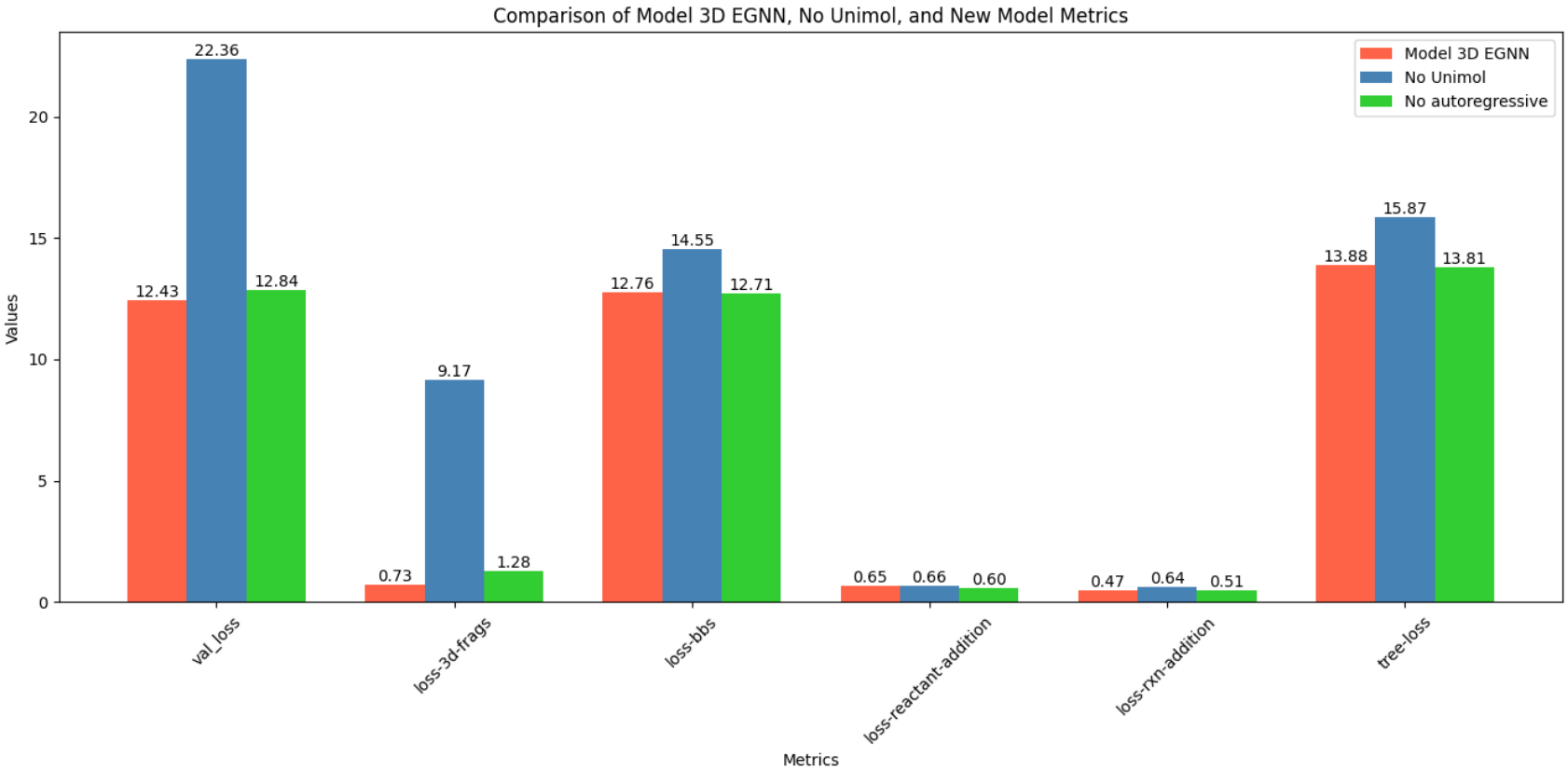


Figure: Abblation Studies

Full Conformation prediction

Distance Matrix and Coordinate Computation

The distance matrix $D \in \mathbb{R}^{(n+m)\times(n+m)}$ is set as follows:

$$D_{i,j} = \begin{cases} \|s_i - s_j\| & \text{if } i,j \leq n, \\ \text{MLP}_d(h_i^{(0)}, h_j^{(0)}) & \text{if } i \leq n, j > n, \\ \|r_i - r_j\| & \text{if } i,j > n. \end{cases} \tag{5}$$

Given a known pocket, all $\|s_i - s_j\|$ are computed. For the ligand, an MLP predicts pairwise distances. The coordinates are calculated via eigenvalue decomposition of the Gram matrix :

$$\tilde{D}_{i,j} = 0.5\left(D_{i,1}^2 + D_{1,j}^2 - D_{i,j}^2\right), \quad \tilde{D} = USU^\top \tag{6}$$

$$\tilde{r}_i = [X_{i,1}, X_{i,2}, X_{i,3}], \quad X = U\sqrt{S} \tag{7}$$

To align coordinates, the Kabsch algorithm computes the rigid transformation:

$$r_i = R\tilde{r}_i + t, \quad i > n \tag{8}$$

FLAG + Uni-Mol

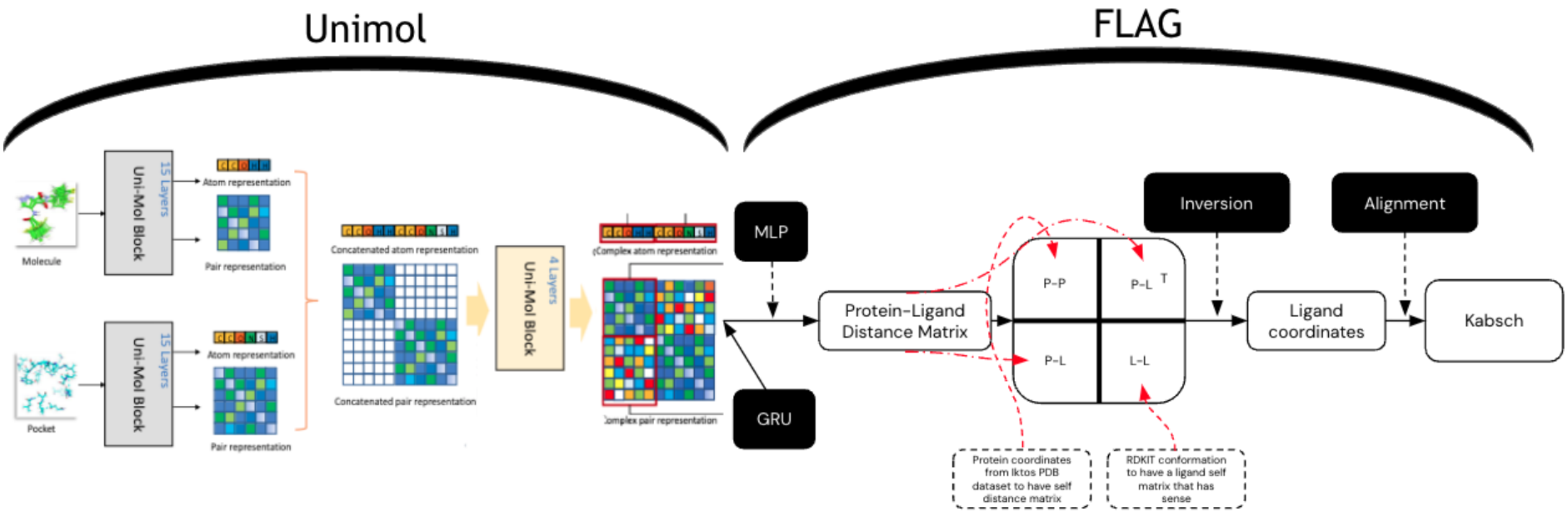


Figure: Merging FLAG and Uni-Mol

Issues !

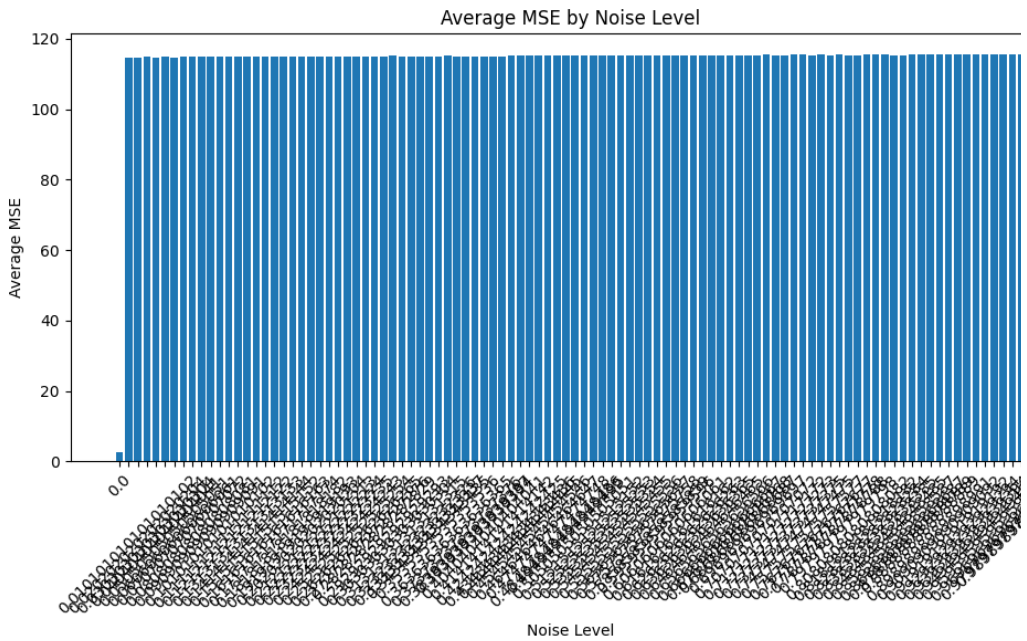


Figure: MSE with Flag Inversion

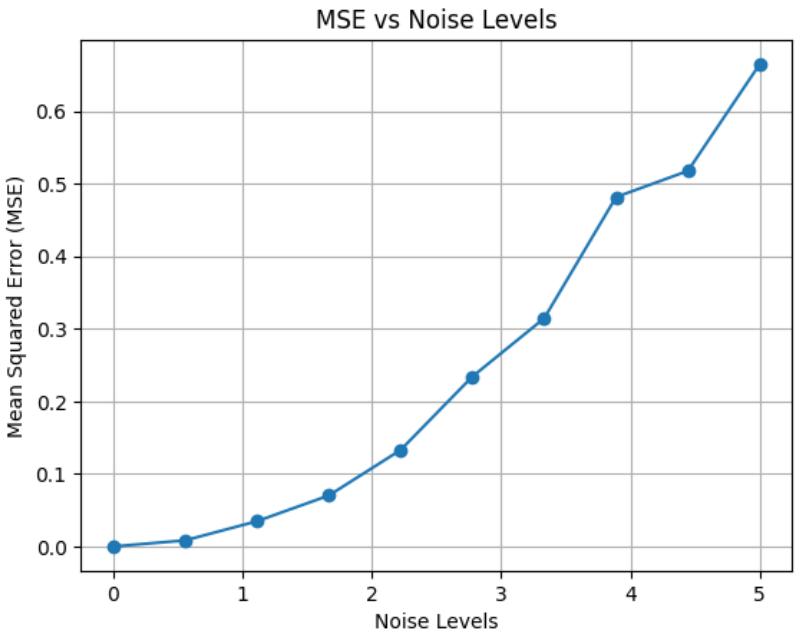


Figure: MSE with Optimization Inversion

Figure: Comparison of Mean Squared Error (MSE) with respect to the noise of the added noise using different inversion methods.

Results

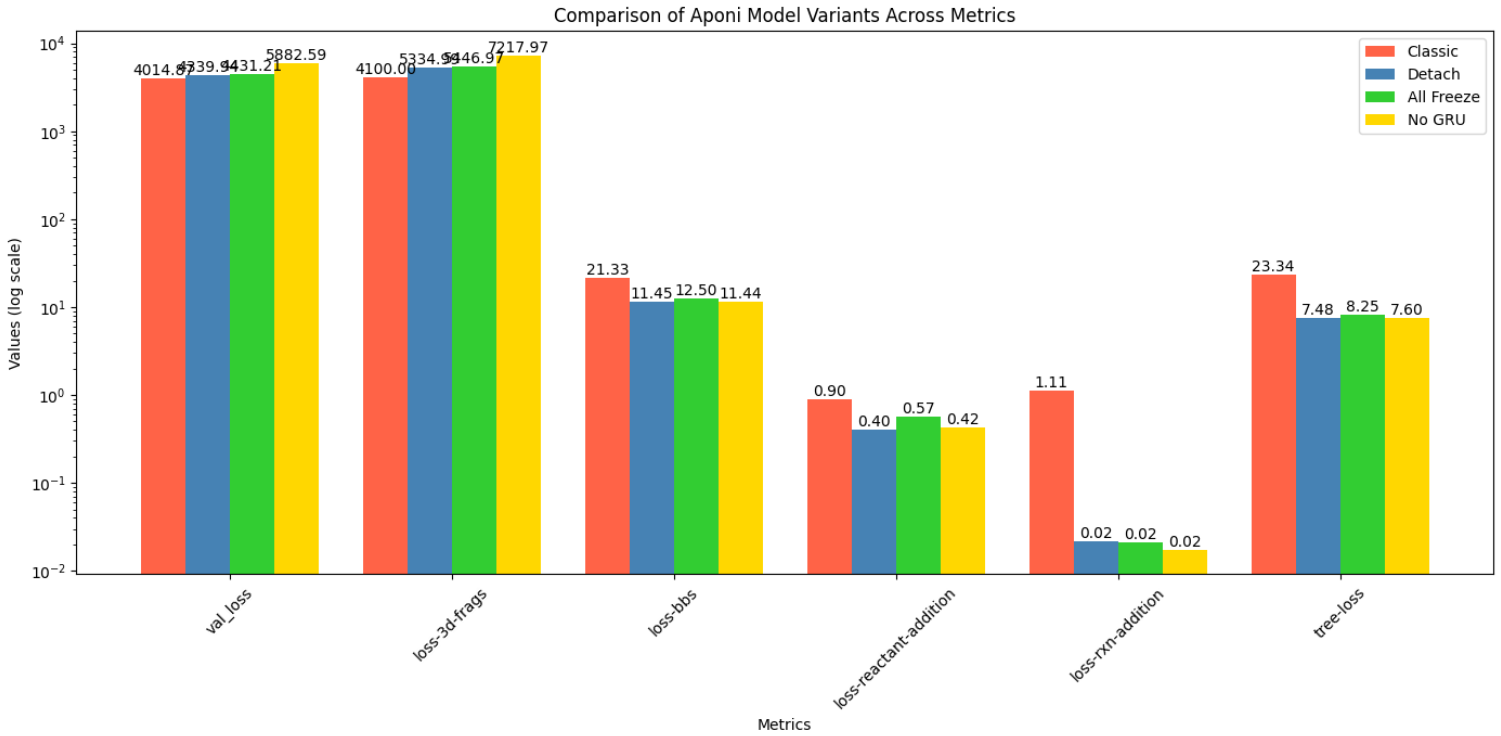


Figure: Results and comparison of the importance of the GRU for Aponi (log scale)

Visualisation

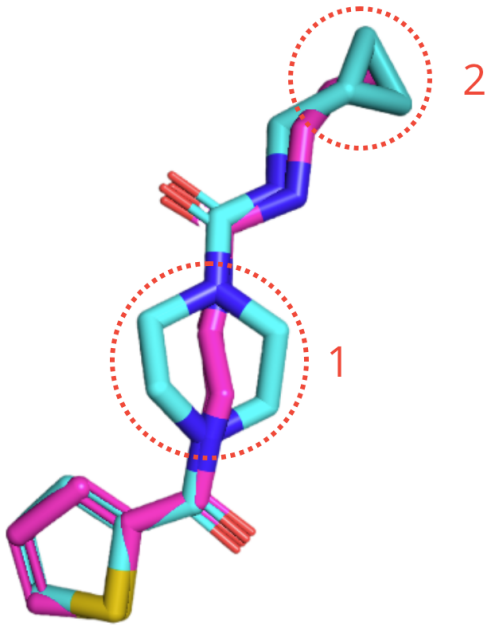


Figure: Inverted ligand visualization based on its predicted distance matrix. Blue indicates the true position, while pink represents the predicted position

EGNN for full conformation

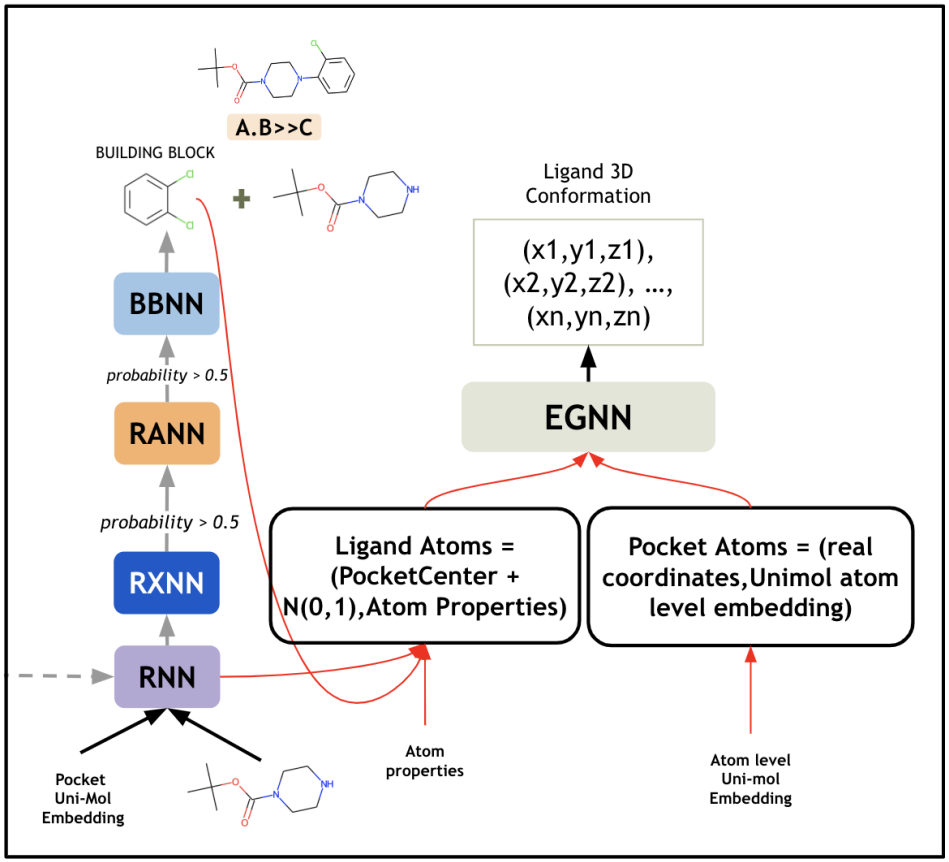


Figure: EGNN Approach

Results

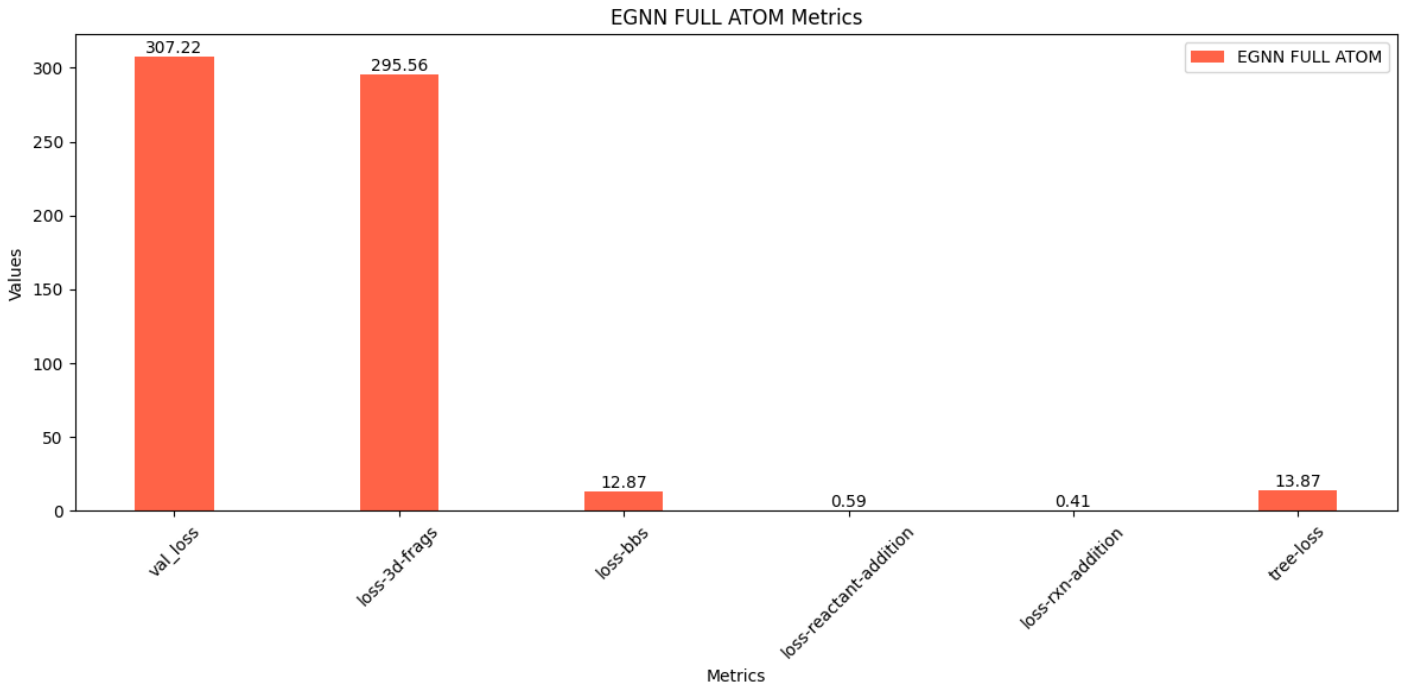


Figure: EGNN Full atom metrics

Visualisation

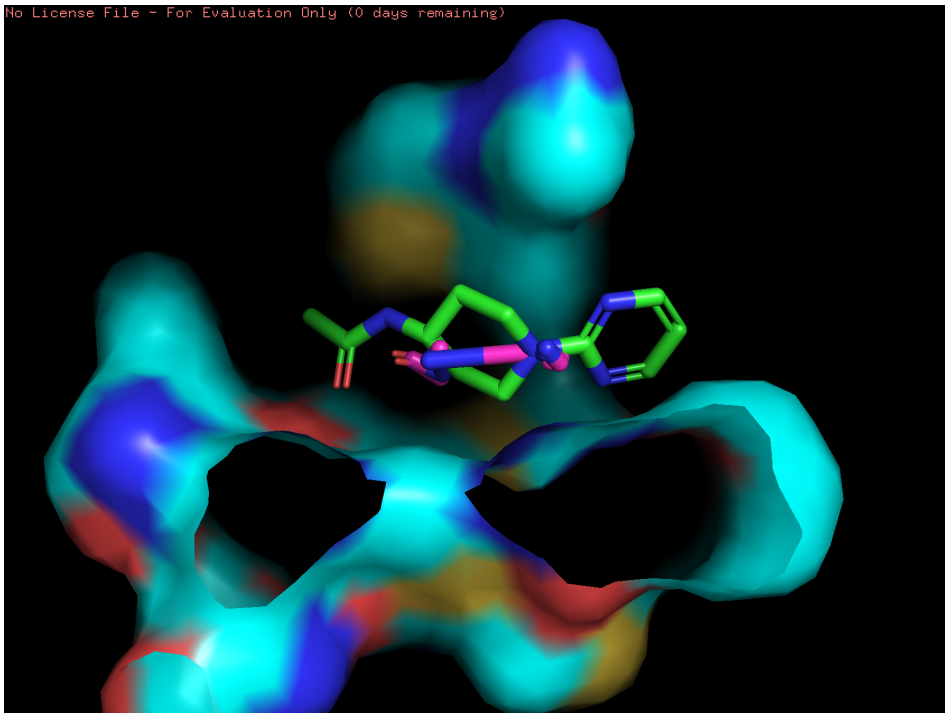


Figure: Comparison of the groundtruth ligand conformation (Green) and the generated one (Pink)

Conclusion

Results Overview

Centroid Prediction:

- **EGNN model** significantly outperformed MLP baseline
- Challenges: Batch size limitations (16) and distilling 3D information into the 2D module
- Ablation studies confirmed the value of autoregressive generation and Uni-Mol embeddings

Results Overview

Full Conformation Generation:

- **FLAG-inspired module**: Promising, but issues with atom interactions lead to false conformations
- Refinement approach unsuccessful for full conformation generation
- **Diffusion models** (e.g., DiffSBDD) and **Flow Matching** hold potential for future development

Key Learnings from the Internship

- Gained hands-on experience with state-of-the-art 3D structure generation techniques
- Learned to adapt complex models for real-world applications and constraints
- Collaborated with chemists and teams for performance evaluation
- Enhanced understanding of model benchmarking and ablation studies
- Learned to produce good and usable code

Thank You!

Thank you for your attention!

Special thanks to my tutor Vincent Bouttier

Appendix: Losses

- **val_loss**: The overall aggregation of losses over the validation dataset.
- **loss_3D_frag**: Part of the loss accounting for 3D predictions.
  - Uses Mean Squared Error (MSE): $\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{y}_i - \hat{\mathbf{y}}_i)^2$, where $\mathbf{y}_i$ and $\hat{\mathbf{y}}_i$ are the true and predicted 3D coordinates.
- **loss_bbs**: Related to predicting the correct building block.
- **loss_reactant_addition**: This loss concerns choosing the correct reaction.
- **loss_rxn_addition**: This loss determines whether to continue the generative process.
- **tree_loss**: Aggregates all of the above losses, grouped for each growing tree, and averages them.

Appendix : Training (Part 1)

Training is made on 4 NVIDIA T4 and 48 VCPU.
Total training time: 50 Epochs

Machine	GPU	vCPU	Cost (USD)
g4dn.xlarge	1	4	0.587
g4dn.4xlarge	1	16	1.342
g4dn.8xlarge	1	32	2.426
g4dn.12xlarge	4	48	4.362

Appendix : Training (Part 2)

Machine	Time (h/epoch)	Training Cost (USD)	Training Time (days)
g4dn.xlarge	7	53.417	3.79
g4dn.4xlarge	5	87.23	2.71
g4dn.8xlarge	3	94.614	1.63
g4dn.12xlarge	1	56.706	0.54

Appendix: Inversion

$$\min_{L} \sum_{i=1}^{N_P} \sum_{j=1}^{N_L} \left(\|P_i - L_j\| - D_{ij} \right)^2 \tag{9}$$

Explanation:
The objective function minimizes the squared differences between distances of points P_i and L_j and target distances D_{ij}. The optimization problem is not convex for the following reasons:

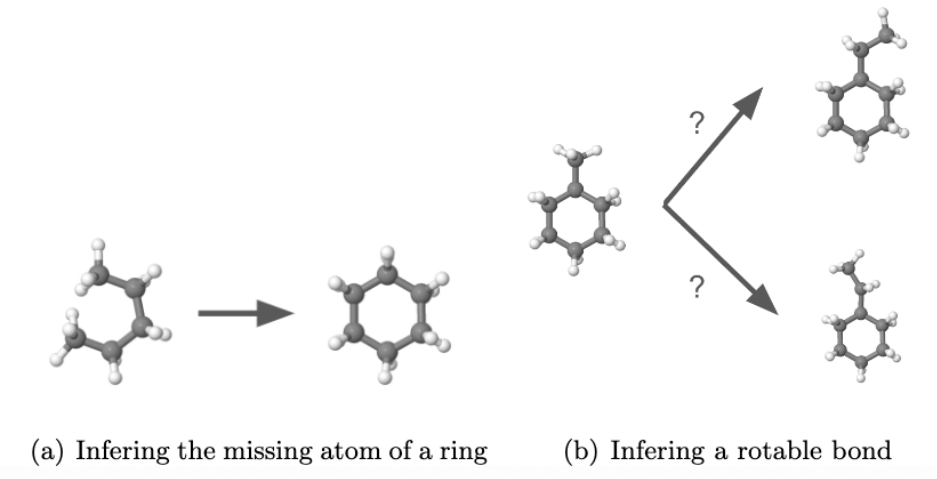
- **Optimization over** L: The minimization is constrained to L, leading to potential non-convexity in the feasible region.
- **Non-Linear Distances**: The Euclidean distance $\|P_i - L_j\|$ is non-linear, which can produce multiple local minima.
- **Quadratic Form**: The squared differences can introduce curvature that may not be convex, depending on point distributions.

Appendix : Type of generators

[-¿] (0,0) – (12.5,0) node[anchor=north] Chemical Plausibility;
(1,0.1) – (1,-0.1) node[anchor=north] Low; (6,0.1) – (6,-0.1)
node[anchor=north] Medium; (10,0.1) – (10,-0.1) node[anchor=north]
High;
[blue] at (1.5,0.2) [circle,fill,inner sep=1.5pt] ; [blue] at (1.5,0.5)
SMILES-based;
[red] at (4,0.2) [circle,fill,inner sep=1.5pt] ; [red] at (4,0.5) Atom-based;
[green] at (7,0.2) [circle,fill,inner sep=1.5pt] ; [green] at (7,0.5)
Fragment-based;
[purple] at (10,0.2) [circle,fill,inner sep=1.5pt] ; [purple] at (10,0.5)
Reaction-based;

Figure: Approaches with Respect to Chemical Plausibility

Appendix: Dataset



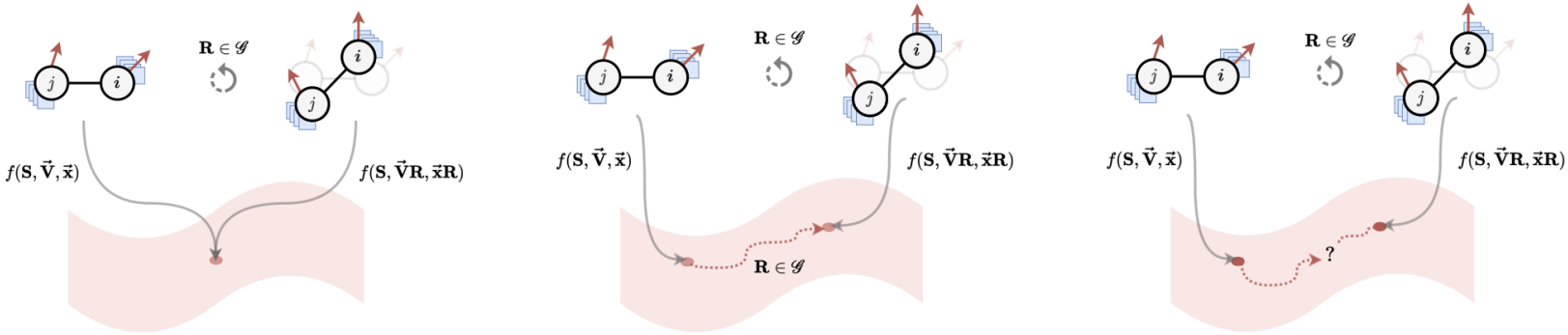
(a) Infering the missing atom of a ring (b) Infering a rotable bond

Figure: the two main atom inference the problem faces

Appendix: Ligand Full Atom representation

atomic number, chirality, degree, formal charge, implicit valence, the number of connected hydrogens, the number of radical electrons, hybridization type, whether it is part of an aromatic ring, the number of rings it is in, and whether it is in a ring of size 3, 4, 5, 6, 7, or 8.

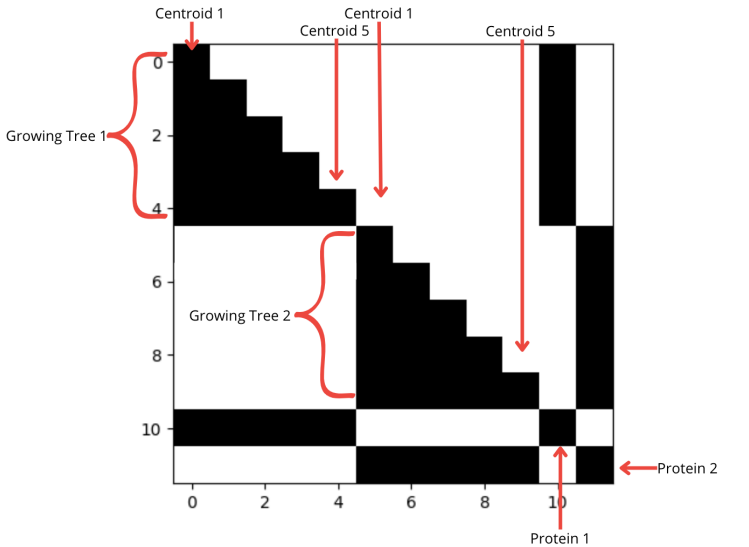
Appendix : Euclidean considerations



(a) $\mathscr{G}$-invariant function (b) $\mathscr{G}$-equivariant function (c) $\mathscr{G}$-unconstrained function

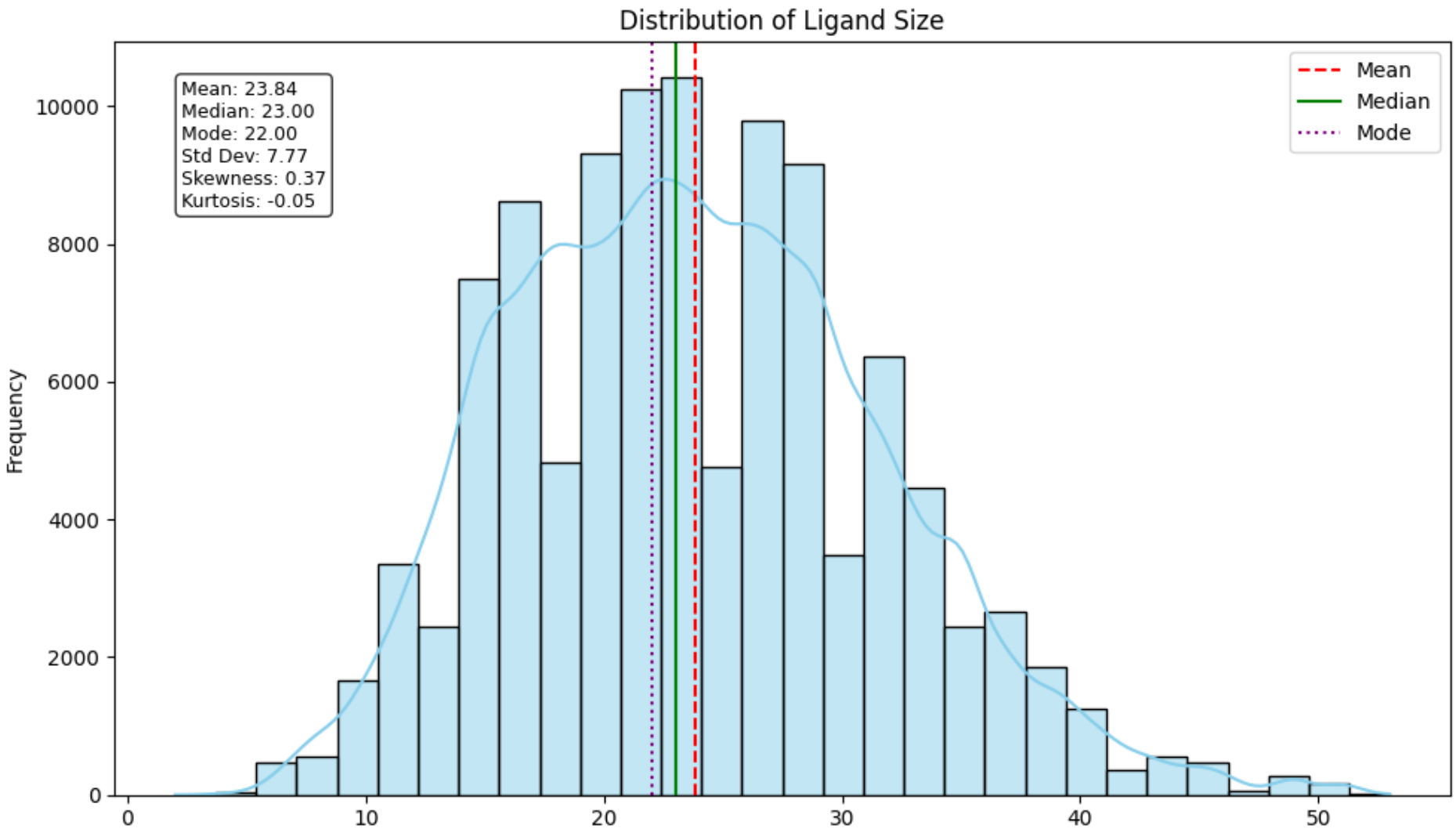
Figure: Invariant, equivariant, and unconstrained functions.

Appendix: Adjacency Matrix



Row 0 represent the first fragment of a growing tree, it's self connected, hence the black square at (0,0) but it's also connected to the protein hence the black square at (0,10). Row 4 represent the last fragment of a growing tree, it's still self connected, but it's also connected to all past centroids, hence the black squares at (4,0),(4,1),(4,2),(4,3). Row 10 represent the protein associated with the first growing tree, it's connected to all the fragments of its fragments: (10,0), (10,1), (10,2), (10,3), (10,4).

Appendix : Distribution of Ligands



Appendix : Distribution of Proteins