
POLITÉCNICA

**UNIVERSIDAD POLITÉCNICA DE MADRID**

**ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA AGRONÓMICA, ALIMENTARIA Y DE BIOSISTEMAS**

**MÁSTER EN BIOLOGÍA COMPUTACIONAL**

**DEPARTAMENTO DE BIOTECNOLOGÍA Y BIOLOGÍA VEGETAL**

**DEPARTAMENTO DE BIOCIENCIAS CELULARES Y MOLECULARES**

# A Graph Convolutional Neural Network Approach to Molecular Dynamics: Analyzing TLR4/MD-2 Interactions with Agonist and Antagonist Ligands

## TRABAJO FIN DE MÁSTER

Autor: Álvaro Arrayás Ruiz

Tutora: Sonsoles Martín Santamaría

Tutor Académico: Krzysztof Wabnik

**Julio 2025**

# INDEX

# LIST OF FIGURES

# KEYWORDS

| | |
|---|---|
| **ARG** | Arginine |
| **CIB** | Centro de Investigaciones Biológicas “Margarita Salas” |
| **CD14** | Cluster of Differentiation 14 |
| **DAMP** | Damage-Associated Molecular Pattern |
| **ED** | Extracellular Domain |
| **FA** | Fatty Acid Chain |
| **FS** | Femtosecond |
| **GAT** | Graph Attention Network |
| **GAFF** | General Amber Force Fields |
| **GCNN** | Graph Convolutional Neural Network |
| **LBP** | LPS-binding Protein |
| **LEU** | Leucine |
| **LPS** | Lipopolysaccharide |
| **LRR** | Leucine-rich Repeats |
| **LYS** | Lysine |
| **MD** | Molecular Dynamics |
| **MD-2** | Myeloid Differentation Factor 2 |
| **MPLA** | Monophosphoryl Lipid A |
| **NPT** | Isothermal-Isobaric Ensemble |
| **NVT** | Canonical Ensemble |
| **NS** | Nanoseconds |
| **PAMP** | Pathogen-Associated Molecular Patterns |
| **PDB** | Protein Data Bank |
| **PHE** | Phenylalanine |
| **PRR** | Pattern Recognition Receptor |
| **PS** | Picoseconds |
| **RELU** | Rectified Linear Unit |
| **RMSD** | Root Mean Square Deviation |
| **SER** | Serine |
| **TIR** | Toll/Interleukin 1 Receptor |
| **TLR4** | Toll-like Receptor 4 |
| **TYR** | Tyrosine |
| **VAL** | Valine |
| **VMD** | Visual Molecular Dynamics |

# SUMMARY

Bridging the gap between experimental biological research and efficient computational modeling of protein-ligand interactions is crucial for accelerating drug discovery. This study addresses this challenge by integrating molecular dynamics (MD) simulations with a graph-based deep learning strategy to predict the functional activity of synthetic FP ligands targeting the TLR4/MD-2 complex, a key component in innate immune signaling.

Three replicas of 150 ns MD simulations were performed for each ligand-receptor complex, involving the FP11 and FP18 and FP12 and FP7 glycolipids, designed for their high structural similarity but opposing agonist/antagonist biological activity correspondingly.

A Graph Convolutional Neural Network (GCNN) was trained on these resulting dynamic simulation data and designed to classify ligand activity on a per-frame basis, incorporating attention mechanisms to improve its ability to differentiate subtle conformational changes of the complex.

Hyperparameter exploration, including training length, data quantity, and particularly atomic input selection, revealed that performance strongly depended on biologically meaningful feature definition. Specifically, residues within the MD-2 binding pocket emerged as the most informative atomic subset, enabling the model to extract patterns tied to functional activity. While the GCNN did not achieve perfect generalization, it consistently outperformed control selections and demonstrated its capacity to learn relevant structural signals.

These promising findings mark a first step for future efforts to build a predictive framework that leverages graph neural networks with MD data. Further validation, such as the integration of dropout layers or the addition to additional simulation data can help to prevent overfitting, while an architectural shift toward Graph Attention Network (GAT) can be done to improve performance as well. This study establishes a strong foundation for applying deep learning to dynamic molecular systems, paving the way for future applications in computational drug design.

# INTRODUCTION

The immune system serves as the primary defense mechanism against pathogens, neutralizing many threats before they can cause harm. This initial barrier plays a crucial role in activating advanced immune responses that prevent infections and maintain homeostasis. A key component of this system is a class of receptors known as pattern recognition receptors (PRRs), which identify microorganisms by recognizing pathogen-associated molecular patterns (PAMPs) and damage-associated molecular patterns (DAMPs). Upon detection, PRRs initiate signaling cascades that trigger an immune response, ensuring a rapid and effective defense against potential threats (Faenza & Blalock, 2022; Taguchi & Mukai, 2019).

Among PRRs, Toll-like receptors (TLRs) are one of the most well-characterized families, consisting of up to ten functional members in humans. TLRs share a common structural organization, featuring an extracellular domain (ED) composed of 16–28 leucine-rich repeats (LRRs) responsible for ligand recognition, and an intracellular toll/interleukin-1 receptor (TIR) domain that mediates downstream signaling (Duan et al., 2022). When activated, the TIR domain recruits TIR-containing adaptor proteins, initiating a signaling cascade that leads to the synthesis of proinflammatory cytokines, illustrated in Figure 1. These inflammatory mediators coordinate immune responses by recruiting immune cells and enhancing pathogen clearance (Fitzgerald & Kagan, 2020).

TLR4 stands out due to its ability to recognize a diverse range of ligands, mainly bacterial lipopolysaccharides (LPS). TLR4 signaling is modulated by myeloid differentiation factor 2 (MD-2), a co-receptor that facilitates ligand binding and receptor dimerization. The TLR4/MD-2 complex formed then is a crucial target in immunomodulation, as its activation can lead to either proinflammatory or anti-inflammatory outcomes depending on the nature of the ligand (H. J. Kim et al., 2023).

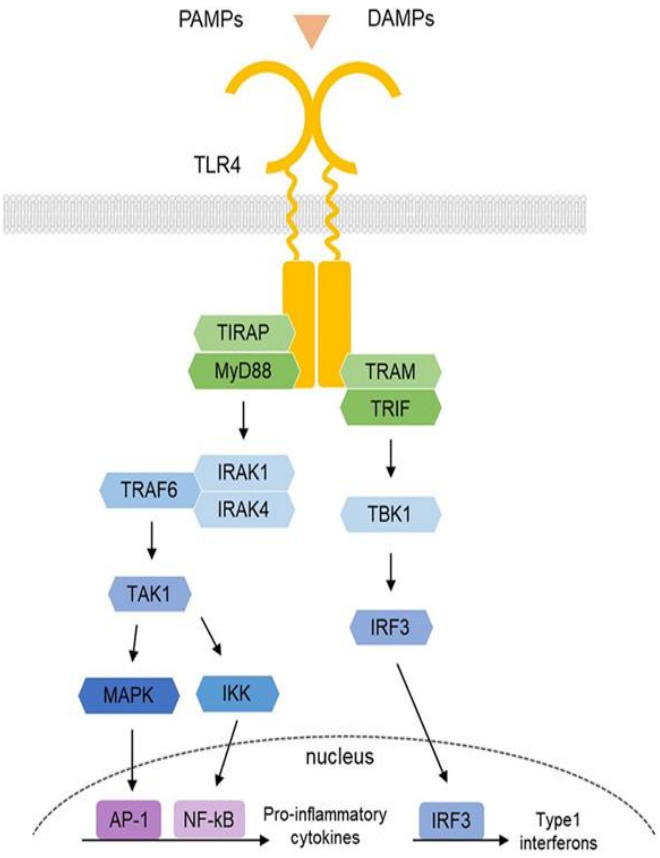


*Figure 1. TLR4 signaling pathways upon activation of the extracellular domain (H. J. Kim et al., 2023).*

Before reaching MD-2, LPS is extracted from the outer membrane of Gram-negative bacteria by LPS-binding protein (LBP), which transfers it to cluster of differentiation 14 (CD14) for its delivery to MD-2 (Kurt-Jones et al., 2000). Then MD-2, which has a β-cup structure with two interaction interfaces, forms a stable heterodimer with TLR4 via salt bridges and hydrogen bonds (Park & Lee, 2013). The dimerization assembly activates, facilitating TLR4/MD-2/LPS homodimerization through non-polar interactions involving hydrophobic residues in the C-terminal region of the adjacent TLR4 (Park et al., 2009). LPS molecule binds to the TLR4/MD-2 system by inserting its fatty acid (FA) chains into the deep and wide hydrophobic pocket of MD-2 and establishing a complex network of polar interactions between the LPS oligosaccharide and TLR4 residues (Matamoros-Recio et al., 2021) as displayed in detail in Figure 2.

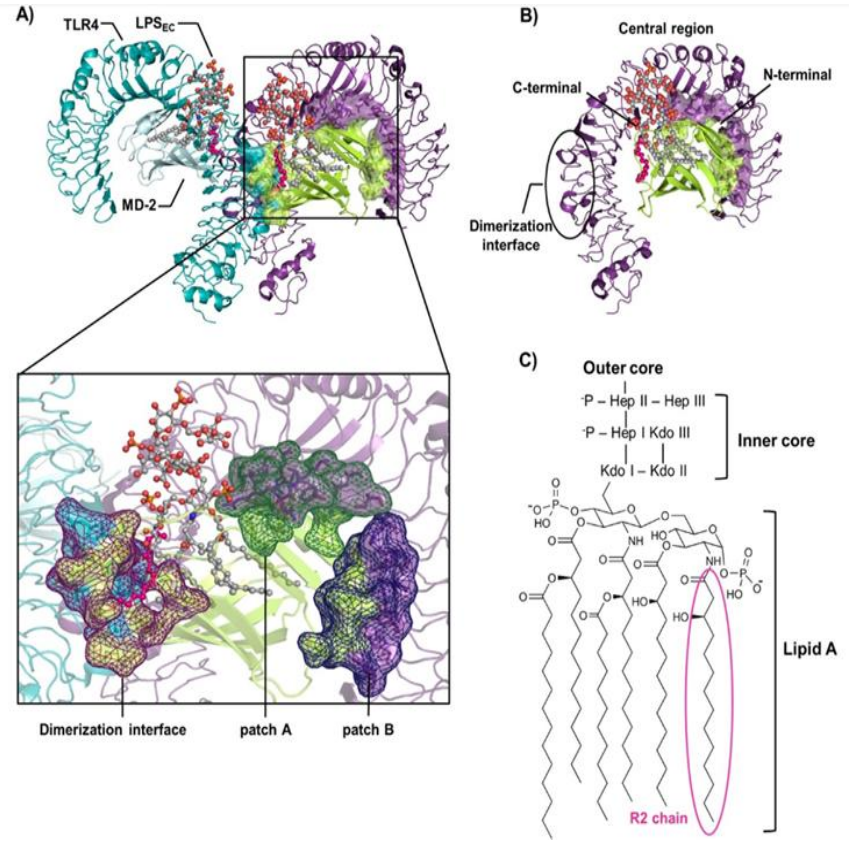


*Figure 2. 3D structure of the TLR4 extracellular domain in complex with agonist LPS (Matamoros-Recio et al., 2021).*

TLR4 has recently emerged as a promising therapeutic target, with the development of both agonist and antagonist ligands offering potential treatments for a wide range of conditions, such as sepsis, autoimmune diseases or neurodegenerative disorders among others (Matamoros-Recio et al., 2021). TLR4 agonists are particularly useful as vaccine adjuvants and in cancer immunotherapy (Facchini et al., 2021). For example, synthetic nontoxic LPS lipid analogs, such as monophosphoryl lipid A (MPLA) derivatives, are key components of vaccines like Fendrix™, for hepatitis B, and Cervarix™, for cervical cancer (Gregg et al., 2017; Lindemann et al., 2017). These molecules enhance immune responses, making them valuable tools in vaccine development and oncology (Facchini et al., 2021).

On the other hand, TLR4 antagonists are being actively studied for their potential to combat excessive inflammation and sepsis (Facchini et al., 2018). LPS mimetics, such as Eritoran, were initially considered promising TLR4 inhibitors, though Eritoran failed Phase III clinical trials, research continues to explore its therapeutic applications (H. M. Kim et al., 2007). Other antagonists include glycolipids, non-LPS-like small molecules, and peptidomimetics, all of which show potential for modulating TLR4 activity (Facchini et al., 2018).

Given the structural complexity of LPS (Dardelle et al., 2024) in both, real life applications and computational modelling, exploring simpler and more accessible LPS analogs that can effectively interact with the TLR4/MD-2 system while maintaining or improving therapeutic efficacy is the main objective and challenge for ongoing research.

Beyond the well-known agonists and antagonists previously mentioned, the CIB research group has successfully developed structurally simpler molecules that function as potent TLR4/MD-2 modulators (Figure 3). These novel compounds, designed to retain key functional properties of LPS while overcoming its limitations, demonstrate promising activity as their simplified structures not only enhance feasibility in drug development but also improve computational modeling accuracy, facilitating the design of optimized therapeutic candidates (Facchini et al., 2018, 2021).

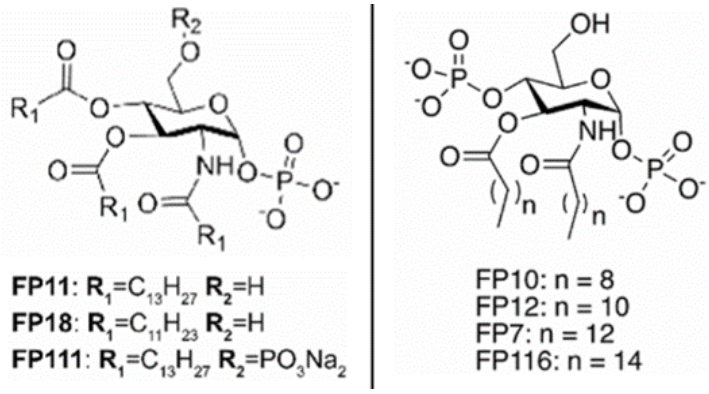


*Figure 3. Chemical structure of TLR4 agonists (left) and antagonists (right) (Facchini et al., 2018, 2021).*

The homologous series of FP glycolipids with fatty acid chains were designed as potential MD-2 ligands, with the aim to make them scalable at an industrial level. The FP11 series and its derivatives (Figure 3, left) were developed to evaluate their capacity to trigger TLR4 activation and act as selective agonists, whereas, the FP12 series and its derivatives (Figure 3, right) were engineered to minimize TLR4 agonist activity, thereby serving as selective antagonists. Specifically, FP11 holds strong potential for development as a next-generation vaccine adjuvant due to its efficacy and ease of synthesis compared to established molecules such as MPLA, while FP12 demonstrates a potent ability to modulate downstream TLR4 signaling without triggering an agonistic response for the treatment of inflammatory diseases (Facchini et al., 2018, 2021).

This study primarily focuses on both FP11 and FP12, as agonist and antagonist candidates respectively, based on their superior binding affinities, favorable pharmacological profiles and the absence of detectable toxicity in vitro and in vivo settings (Facchini et al., 2018, 2021). Despite these encouraging findings, the high structural similarity between FP11 and FP12 poses a challenge in clearly delineating the molecular determinants underlying their distinct biological activities. To further explore these mechanistic differences and expand the scope of the analysis, FP18 agonist compound and FP7 antagonist compound were also included in the study as reference compounds for comparative evaluation.

Predicting interactions between proteins and ligands based on structure emerged in the last decade as promising tools but still remains a difficult task in biology (Yang et al., 2022). Computational techniques to investigate the structural and dynamic features that drive the functional divergence between FP glycolipids are paramount. By analyzing binding modes through data derived from molecular simulations, machine learning models can be employed to identify subtle yet critical differences in their interaction patterns with MD-2 (Marchetti et al., 2021). This strategy is intended to move beyond traditional analysis and toward an interpretable framework that supports the rational design of FP-based compounds with enhanced confidence in their immunomodulatory function.

In this context, graph theory offers a powerful and structured approach to extract meaningful patterns from complex molecular systems (Patel et al., 2024). A graph, in its most basic form, represents a system as a set of nodes, e.g., atoms or residues, and edges, e.g., interactions or spatial proximity, thereby capturing the topological and dynamic relationships inherent in biomolecular structures (Vishveshwara et al., 2002). Moreover, molecular dynamics (MD) simulations serve as a rich source of temporal data, where dynamic conformational changes and interaction networks can be abstracted into graph-based models (Patel et al., 2024).

Graph theoretical methods have proven to be effective in revealing global properties of biomolecular systems, such as connectivity, centrality, modularity, and community structure while maintaining a simple syntax. When integrated with MD simulations, graphs enable the identification of critical residues, interaction hubs, and communication pathways that are not easily accessible through conventional analysis alone (Bougueroua et al., 2023; Patel et al., 2024).

Building on the graph-based representation of molecular dynamics, Graph Convolutional Neural Networks (GCNNs) are a powerful class of machine learning models capable of learning from structured data encoded as graphs (Reiser et al., 2022). Unlike traditional convolutional neural networks (CNNs), which operate on grid-like data such as images, GCNNs generalize the convolution operation to non-Euclidean domains, allowing them to directly model the topological and relational properties of molecular systems (Hofstetter et al., 2022). This makes GCNNs particularly well-suited for the analysis of biomolecular interactions, where spatial and functional connectivity play a critical role.

By encoding simulation frames as graphs, it becomes possible to classify functional states and uncover structural features that drive agonist versus antagonist activity (Li et al., 2022). GCNNs will be employed to learn from molecular graphs derived from MD simulations of TLR4/MD-2/ligand complexes. Each simulation frame is transformed into a graph where nodes represent atoms or residues, and edges encode physicochemical interactions or spatial proximity (Figure 4). Then, training the GCNN to classify whether a given graph corresponds to an agonist or antagonist state, the model can uncover features that are predictive of their divergent functional behaviors, enabling not only classification but also a mechanistic interpretation of the molecular determinants involved in TLR4 modulation.

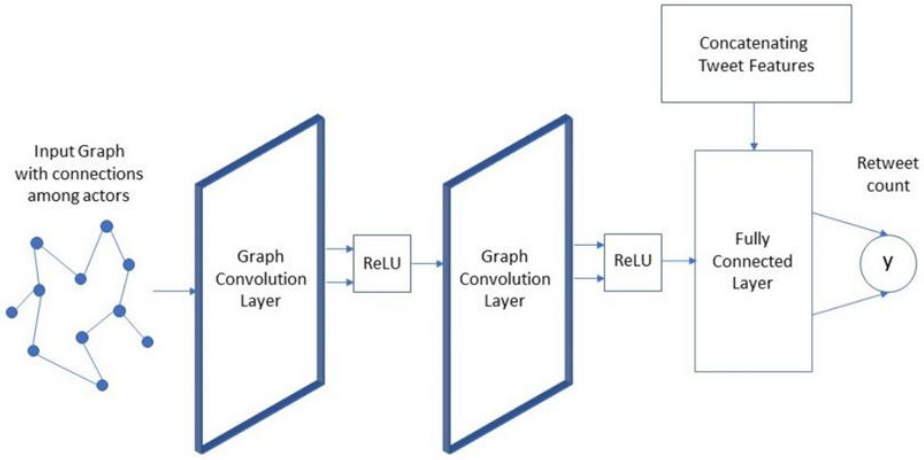


*Figure 4. Architecture of a Graph Convolutional Neural Network social media model (Vijayan, 2018).*

GCNNs have been successfully applied across a broad range of domains beyond structural biology, demonstrating their flexibility in learning from complex relational data (Zhang et al., 2019). For example, a study by applied GCNNs to predict the virality of tweets during political elections, modeling user interactions as graphs and achieving superior performance over traditional approaches (Vijayan, 2018). This type of application highlights the general utility of GCNNs for forecasting outcomes from early structural patterns in dynamic networks, a principle that is similarly leveraged in this work to predict ligand behavior from early molecular features. A visual representation of the GCNN architecture used in that study served as inspiration for the model design adopted (Figure 4).

Overall, combining graph theory with deep learning introduces an interpretable, data-driven approach for elucidating the structural basis of functional divergence in chemically similar compounds. The GCNN model, trained on simulation data, provides a predictive lens through which subtle differences in ligand-receptor interaction dynamics can be understood, ultimately advancing the rational design of immunomodulatory glycolipids with precision.

# OBJECTIVES

- **Perform and Analyze MD simulations** of the complexes between TLR4/MD-2 and agonist and antagonist FP compounds developed by the lab to evaluate binding over time.
- **Build a Graph Convolutional Neural Network** trained on graph-based data from MD simulation capable of learning meaningful molecular features to classify ligand activity.
- **Predict the activity** of individual frames to validate the model accuracy.

# MATERIAL AND METHODS

## 1. Molecular Dynamics (MD)

### Ligand Preparation

The molecular models of the TLR4/MD-2 complex and the set of FP compounds used in this study were previously developed and provided by members of the CIB research group (Facchini et al., 2018, 2021; Pérez-Regidor et al., 2022). To maintain consistency across the data, the antagonist-bound conformation of the TLR4/MD-2 complex was employed as a reference framework regardless of the functional profile (agonist or antagonist) of the ligands.

Given the lack of a complete experimental structure of the human TLR4/MD-2 complex in its antagonist-bound state, a hybrid structural approach was utilized. Specifically, the antagonist conformation of the human MD-2 protein (PDB ID: 2E59) (Ohto et al., 2007) was superimposed onto the MD-2 subunit of the agonist-bound TLR4/MD-2/LPS complex (PDB ID: 3FXI) (Park et al., 2009) using the PyMOL molecular visualization system. The resulting structure was subsequently refined to ensure structural stability and suitability for molecular dynamics simulations (Pérez-Regidor et al., 2022).

Ligand structures were prepared starting from mol2 files generated using Protein Preparation Wizard in Maestro. These were optimized with Gaussian09, partial atomic charges were calculated using quantum mechanical methods from Antechamber (J. Wang et al., 2006) and atom types were assigned according to the General Amber Force Field (GAFF) (J. Wang et al., 2004). Finally, partial charges were refined and all necessary input with topology and coordinates information for each file were generated correctly with the AMBER suite for its use in molecular dynamics simulations.

This first stage of the project was the result of extensive previous optimization and development by the CIB research group, allowing the following computational analyses of this study.

### Topology and Coordinates of the Protein/Ligand System

Before performing molecular simulations, a detailed structural analysis of the TLR4/MD-2 complex was carried out in order to gain a deeper understanding of the bonds between the MD-2 binding pocket and the FP ligands.

As mentioned in the introduction, this study focuses on FP11 and FP18 as main agonists and FP12 and FP7 as main antagonists. Using PyMol for molecular visualization, ligands were analyzed jointly based on their functional activity to determine key binding site residues (Figure 5). This will be helpful to highlight binding zones during simulation to track down and check if the bonds still remain throughout all the simulation to the end just to be considered relevant and the docking stable (Matamoros-Recio et al., 2021).

Among the residues identified for each FP compound, phenylalanine was noted for its potential to engage in $\pi$ stacking interactions with the aromatic moieties of the ligands. Lysine, a consistently positively charged residue under physiological conditions, was observed to form ionic interactions with the phosphate groups present in both agonists and antagonists. Additional residues such as tyrosine, arginine, and leucine were also identified as potentially significant and were selected for further monitoring during the MD simulations.

Regarding the position of the agonist glycolipids FP11 and FP18, shown in Figure 5, their fatty acid chain placements reveal good positioning within the MD-2 complex, with only slight angular deviations between them. Additionally, a trigger phenylalanine residue, known to change its angle orientation depending on the conformational state, generally <60º if agonist activity and >60º if antagonist activity (Matamoros-Recio et al., 2021), is highlighted on the left edge of MD-2. In the case of FP18, this residue displays an antagonist-like orientation, however, this is not a cause for concern. As stated in the *Ligand Preparation* section, the antagonist-bound conformation of the TLR4/MD-2 complex was used as the structural reference for all ligands so it is not anomalous behaviour.

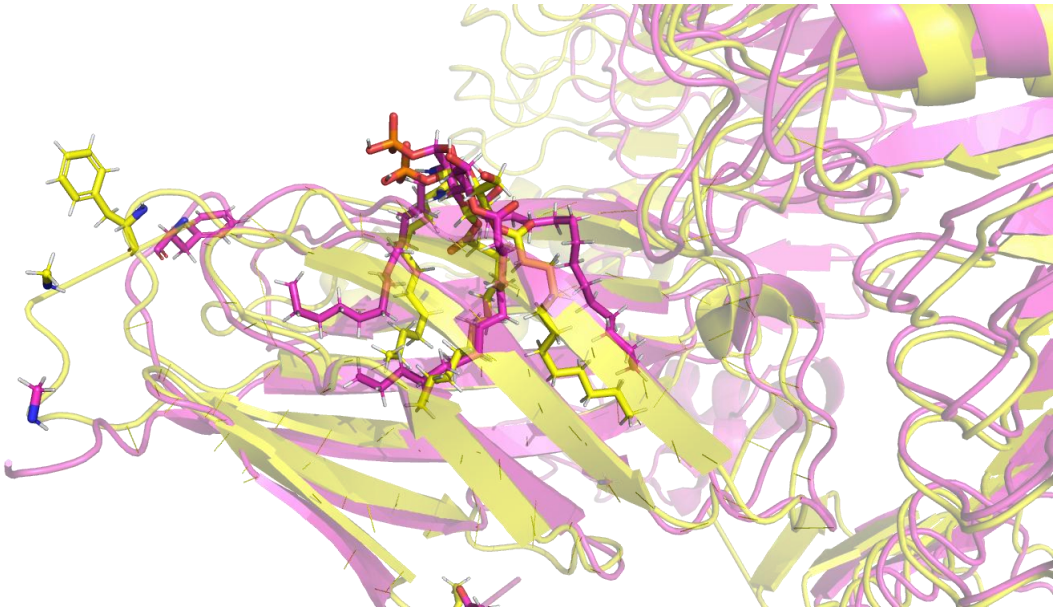

*Figure 5. Binding of the FP11 (pink) and FP18 (yellow) agonist ligands within the MD-2 pocket.*

Similarly, FP12 and FP7, displayed in Figure 6, exhibit comparable fatty acid chain orientations directed toward the insides of the MD-2 pocket, again with minor angular deviations between them. In both cases, the trigger phenylalanine remains in an antagonist-like configuration, which aligns with the initial system setup. These structural observations further informed atom selection strategies and guided subsequent simulation analyses.

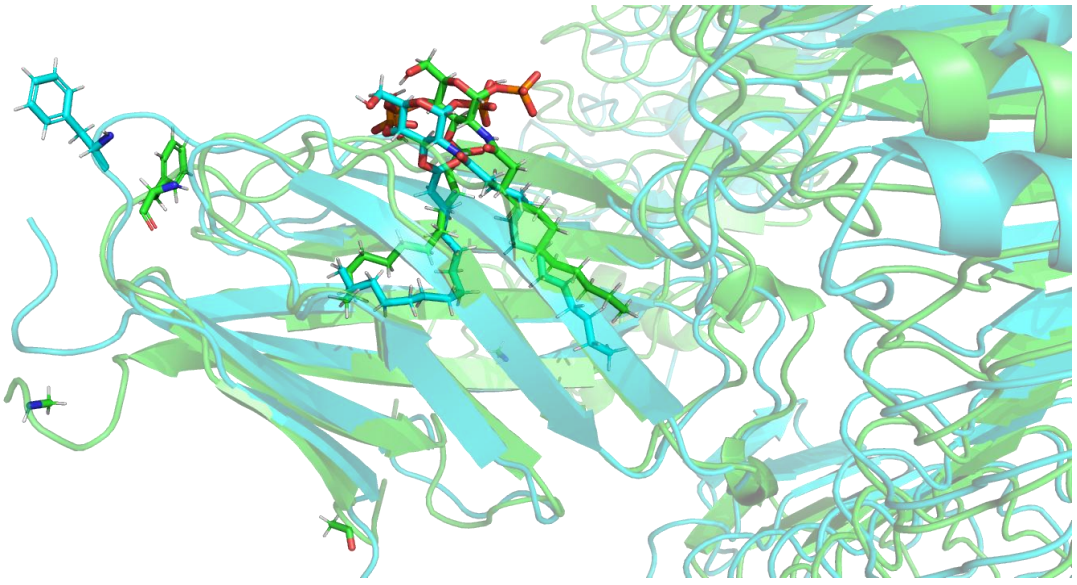

*Figure 6. Binding of the FP12 (green) and FP7 (green) antagonist ligands within the MD-2 pocket.*

By establishing a solid understanding of the molecular interface between receptor and ligand, the extracted trajectory data from the simulations to build the graphs for the neural network could be interpreted with greater confidence in terms of mechanistic relevance. This practice lets also save time later when analyzing simulations to look for significant interactions assuring the binding is occurring.

### AMBER Simulation

For each FP compound three independent simulations were performed as a means to ensure robustness and continuity of the results as well as to provide more input data for the neural network training. All MD simulations were done under AMBER20 molecular package and followed a consistent protocol of minimization, heating and production phases.

The minimization phase began with 8000 steps, including a combination of 1000 steps using the steepest descent algorithm and 7000 steps with the conjugate gradient algorithm. A harmonic potential constraint of 100 $kcal \cdot mol^{-1} \cdot Å^{-2}$ was applied to both the proteins and the ligand during this process. This restraint was progressively reduced to 10, 5, and 2.5 $kcal \cdot mol^{-1} \cdot Å^{-2}$ over three intervals of 600 steps each, using the conjugate gradient algorithm. Therefore, the entire system was uniformly minimized without constraints.

The heating phase was conducted in two stages. First, the system was heated from 0 to 100 K in the canonical ensemble (NVT) using the Langevin thermostat (Nauchitel', 1981) with a 10 $kcal \cdot mol^{-1} \cdot Å^{-2}$ harmonic restraint on the protein and ligand. Then, heating continued from 100 to 300 K under the isothermal-isobaric ensemble (NPT) (Jorgensen et al., 1983) with the same restraint, followed by a 100 ps equilibration run without restraints.

The production phase was performed in the NPT ensemble using the Langevin thermostat with a 2 fs time step. Long-range electrostatic interactions were computed using the Particle Mesh Ewald (PME) method (Essmann et al., 1995). Coordinate trajectories were saved every 2 ps, analyzed using the cpptraj module within AMBER, and visually inspected using the VMD program (Humphrey et al., 1996). RMSD plots for system components were also generated using cpptraj (Roe & Cheatham, 2013).

## 2. Graph Convolutional Neural Network

### Data Representation and Graph Construction

Each frame extracted from the molecular dynamics trajectories was treated as a snapshot of atomic spatial configurations. To transform these frames into graph-based data suitable for deep learning, the MDAnalysis library (Gowers et al., 2016; Michaud-Agrawal et al., 2011) was used to extract atomic positions, residue identities, and atom types. These features served as the basis for node attributes.

In the constructed graph, each atom represents a node, and edges are determined by spatial proximity using a radius graph algorithm from the torch_geometric library. A fixed radius of 5.0 Å was used to establish edges, capturing relevant short-range spatial interactions while maintaining computational efficiency.

Node features were built using two parallel encodings:

- Residue identity, embedded using torch.nn.Embedding with a vocabulary of 24 standard residues (Table 1 in Annex).
- Atom type, embedded using torch.nn.Embedding for 33 distinct atom types (Table 2 in Annex).

These categorical mappings were curated using custom JSON dictionaries (Pezoa et al., 2016) (atom-list.json and residue-list.json) and their inputs were transformed into 128-dimensional vectors before being processed by the neural network.

In Figure 7, a visual representation of the graph structure derived from the molecular system is provided, rendering an example frame using NetworkX package (Aric A. Hagberg, 2008) and optimized with Gephi visualization platform (Bastian et al., 2009.), along with the actual TLR4/MD-2 complex with its structural residues components color-coded according to their functional relevance during the agonist/antagonist induced conformational changes in the simulation.

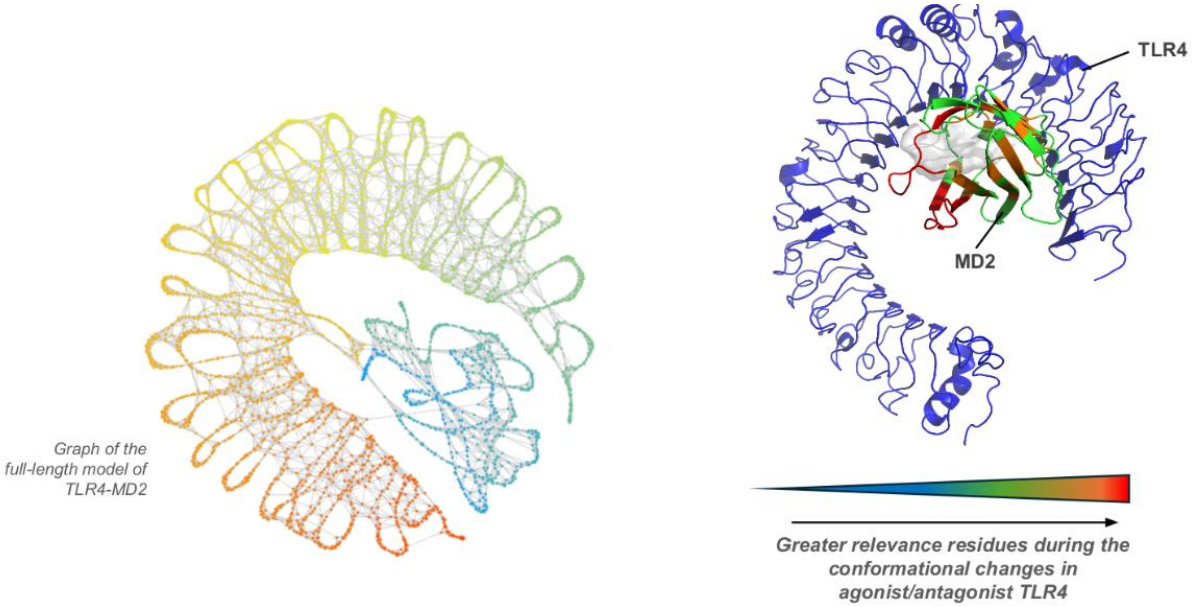


*Figure 7. Graph representation derived from a MD simulation frame (left) compared to the full TLR4/MD-2 complex colored by residue relevance during conformation changes associated with ligand activity (right).*

While this visualized graph is not directly used as input to the GCNN, it serves as an intuitive abstraction of the model's perspective on the molecular data, as the resulting structure closely mirrors the topology of the actual TLR4/MD-2 complex. This resemblance reinforces the suitability of a graph-based representation for this problem because it aligns with the underlying physical and biological architecture of the system.

**Model Architecture**

The Graph Convolutional Neural Network (GCNN) was implemented using the PyTorch (Paszke et al., 2017) and PyTorch Geometric libraries, as it allows to work with this kind of data derived from MD simulation. The framework of the architecture, presented in Figure 8, consists of two parallel branches, each designed to extract complementary chemical features from the atom and residue encodings, which are subsequently merged in a fully connected layer that integrates the learned representations to produce the final activity prediction.

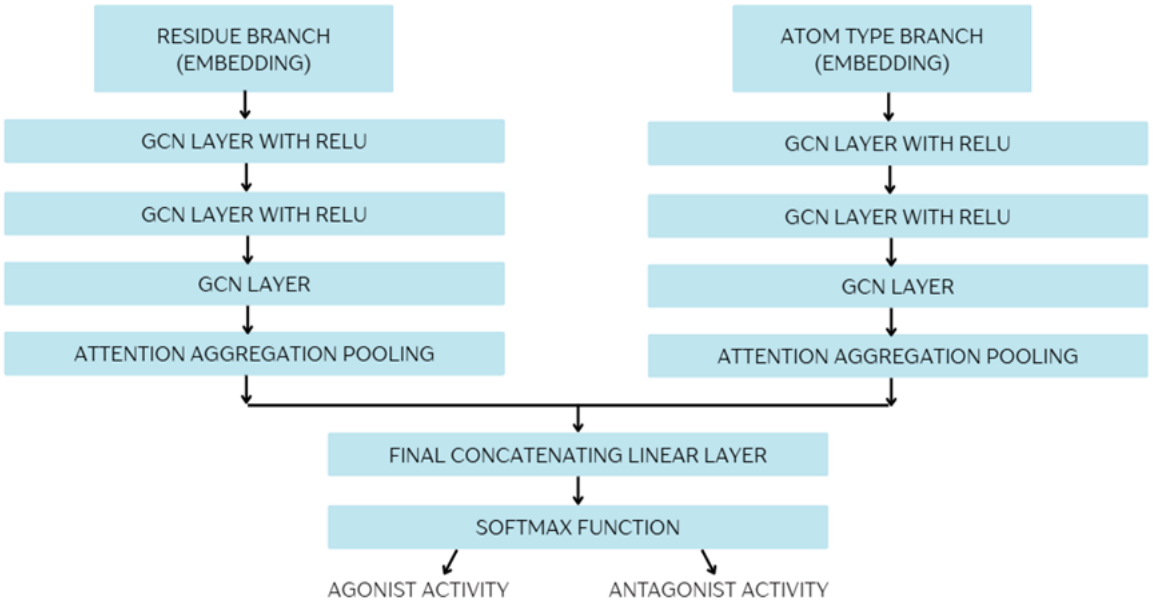


*Figure 8. Architecture of the Graph Convolutional Neural Network developed.*

Residue Branch:

- **Input:** Residue indices are mapped to 128-dimensional vectors using an embedding layer, allowing the model to learn residue-specific representations.
- **Layer 1:** GCNConv (128, 64) + ReLU
- **Layer 2:** GCNConv (64, 32) + ReLU
- **Layer 3:** GCNConv (32, 16)

Atom Type Branch:

- **Input:** Atom types are embedded in a separate 128-dimensional space using a second embedding layer, mapping the different types of atomic interactions.
- **Layer 1:** GCNConv (128, 64) + ReLU
- **Layer 2:** GCNConv (64, 32) + ReLU
- **Layer 3:** GCNConv (32, 16)

The GCNConv function perform neighborhood aggregation in the layer, allowing each node to update its feature representation by integrating information from its spatial neighbors in the graph. This is essential for modeling local interactions such as residue-residue or atom-type interactions relevant to ligand binding (Crampon et al., 2023).

Following each convolutional layer, the ReLU (Rectified Linear Unit) activation function introduces non-linearity into the model, helping it capture complex relationships. ReLU outputs zero for negative inputs and passes through positive values unchanged, which mitigates vanishing gradient issues during training and promotes sparsity (Agarap, 2019).

At the end of each branch, Attentional Aggregation is used in the pooling layer to produce a fixed-size graph-level embedding. The attention mechanism learns importance weights for each node via a small gating MLP, Linear (16, 1) ReLU and Linear (1, 1), enabling the network to focus on the most informative parts of the graph for classification (Y. Wang et al., 2022).

The 16-dimensional pooled outputs from both branches are then concatenated into a single 32-dimensional feature vector. This combined representation is passed through a final fully connected layer to produce raw class scores for binary classification (agonist vs. antagonist).

While the final layer does not explicitly include a softmax activation, this step is handled internally by the CrossEntropyLoss function during training (Mao et al., 2023). Softmax is crucial to interpret the model's outputs as class probabilities, and is responsible for transforming raw logits into meaningful prediction scores during evaluation or inference (Gu et al., 2023).

**Training Procedure**

The training of the GCNN was carried out again using PyTorch Geometric for graph-based operations. The model was optimized using the Adam optimizer with a learning rate of 0.001 and a weight decay of 5e-4 to prevent overfitting. The loss function employed was CrossEntropyLoss, suitable for multi-class classification and internally applying a softmax operation to the model's output logits. Training was performed over multiple epochs, with each epoch comprising a full pass through the training dataset, which was loaded in mini-batches using the DataLoader function for efficient graph batching.

During each training iteration, the model received batches of simulation frames represented as graphs. For each graph, nodes were encoded using the residue or atom type embeddings, and message passing occurred via GCNConv layers with ReLU activations. Following the forward pass, the loss was computed between the predicted logits and true class labels (agonist or antagonist), and gradients were back propagated through the network to update the model parameters.

To monitor generalization, a separate test set was evaluated after each epoch. Accuracy and loss metrics were recorded throughout training to track model convergence and performance. All training and evaluation steps were conducted in evaluation or training mode as appropriate, using .train() and .eval() model switches to control that batch norm behaviors were relevant.

All the results and performance evaluations of the trained model, including classification accuracy and interpretability of learned features, will be discussed in detail in the Results and Discussion section that follows.

# RESULTS AND DISCUSSION

## 1. Molecular Dynamics (MD)

Evaluation of the stability and binding mode of the complexes between the TLR4/MD-2 system and glycolipids, FP11, FP12, FP7 and FP18 (FP compounds), was undertaken by means of molecular dynamics (MD) simulations. FP compounds FP11 and FP18 are TLR4 agonists and FP12 and FP7 are TL4 antagonists and their design, synthesis and biological activities were published by the CIB research group (Facchini et al., 2018, 2021). Each system underwent three independent 150 nanosecond (ns) simulation runs, making a total of 450 ns per ligand. The root-mean-square deviation (RMSD) of both the ligand and the TLR4/MD-2 complex was monitored throughout the simulations.

In all cases, the ligands displayed similar behavior. Although they exhibited considerable fluctuations early in the simulation, the MD-2 receptor also showed deviations a bit more moderate but often showing a coordinated movement pattern with the ligand. This suggests that even with considerable motion, the ligand and receptor remain associated as their fluctuations are synchronized.

RMSD plots, in Figures 9 and 10, were calculated from trajectory frames sampled every 1 ns over the course of the simulations. By measuring the RMSD of each FP compound replicate relative to MD-2, the movement of this ligands can be tracked and analyzed to assess conformational stability and dynamic behavior, as expected in real protein/ligand systems. With the exception of one FP18 replicate that reached values above 7 Å, most trajectories remained below 5 Å, indicating overall structural stability with normal fluctuations. Notably, replicates of agonists tended to exhibit slightly higher RMSD values, which may be attributed to the use of the antagonist-bound TLR4/MD-2 complex as a common reference structure for all ligands. Despite this, both agonist and antagonist simulations showed consistent behavior with no significant deviations, suggesting that the ligands maintained stable binding conformations within the MD-2 pocket. These results suggest that the binding modes observed in docking (Facchini et al., 2018, 2021) were preserved during the MD simulations, supporting the reliability of the predicted interactions.

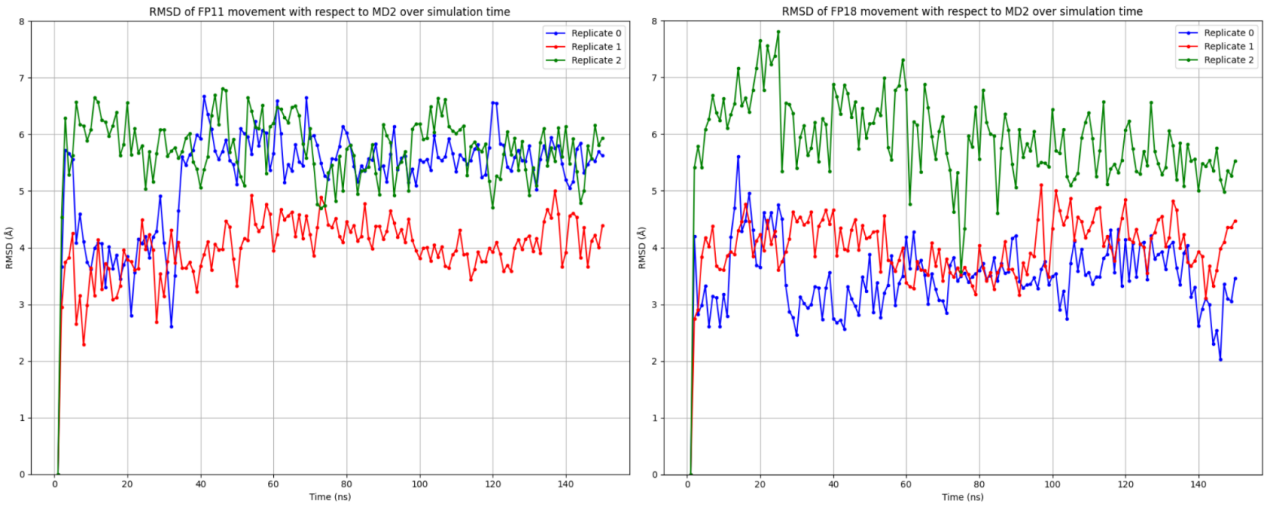


*Figure 9. Root mean square deviation (RMSD) of FP11 (left) and FP18 (right) with respect to MD-2 over a 150 ns. Each plot shows RMSD trajectories from three independent replicates (Replicates 0, 1, and 2) for each agonist.*

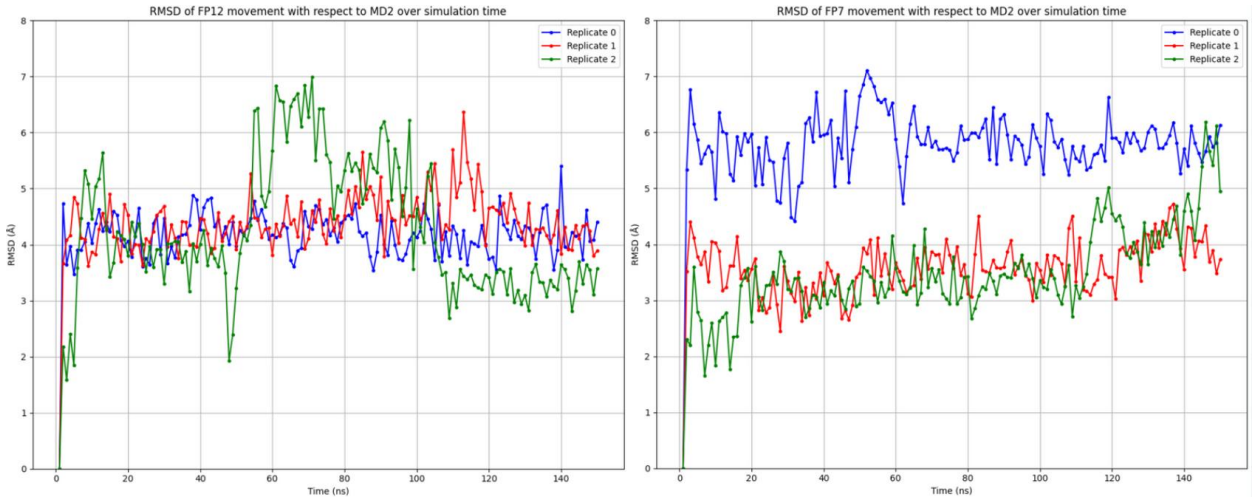


*Figure 10. Root mean square deviation (RMSD) of FP12 (left) and FP7 (right) with respect to MD-2 over a 150 ns. Each plot shows RMSD trajectories from three independent replicates (Replicates 0, 1, and 2) for each antagonist.*

Specific interactions between MD-2 residues and FP compounds were also monitored (Figure 11), highlighting Lys716, which showed stable interactions with the phosphate groups of the ligands during the first 100 ns, with some detachment observed in the final 50 ns. Moreover, Phe720 and Tyr725 remained well positioned throughout most of the simulation while Phe713 and Phe715 remained close, with only minor mid-simulation separations from the fatty acid chains within the MD-2 pocket. Leu655, Phe670, Val729, and Phe741 maintained stable contacts consistently with lipid regions, creating hydrophobic interactions. Arg684 alternated between interactions with the fatty acid chain and the phosphate group, while Ser714 exhibited intermediate stability forming hydrogen bonds without fully detaching.

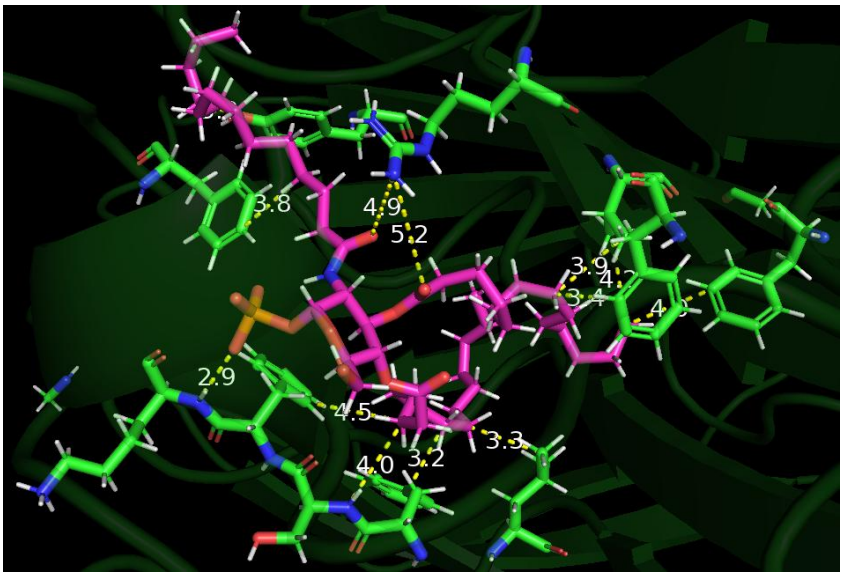


*Figure 11. Distance proximity measured in Å between identified MD-2 key residues (green) and FP11 agonist ligand (pink). FP11 is shown here as a representative example, the same MD-2 residues and ligand interactions were observed across all other FP compou*

The fatty acid chains tended to insert into the hydrophobic pocket of MD-2, while the sugar-phosphate moiety remained near MD-2's polar surface, favoring interactions with lysine and arginine residues as noted. Additionally, a unique observation was made for agonist ligands where sometimes the central fatty acid chain extended outward instead of inserting into the pocket, causing one lateral chain to adopt a constrained conformation, but it did not appear to affect the binding. Building upon these structural and dynamic insights with no unexpected deviations obesrverd, the study went on to the next phase.

## 2. Graph Convolutional Neural Network

Deeping into the patterns of protein/ligand interactions is of great importance for the understanding of agonist/antagonist activity of modulators and further drug design. These processes are intrinsically dynamic, not static, so MD simulations contain subtle information that we wanted to extract. To do so, trajectories from the MD simulations were analyzed by graph convolutional neural network (GCNN). To understand how the final architecture of the GCNN model was determined, it is essential to walk through the series of experimental variations that were tested during training. The model optimization process followed an iterative, trial-and-error methodology in which small, controlled changes were introduced to evaluate their impact on label classification accuracy and loss. These tests were carried out with the dual objective of improving predictive performance while minimizing overfitting.

In this study, one trajectory corresponds to the complete 150 ns MD simulation replication of the TLR4/MD-2 complex with the FP compounds. Each trajectory consists of 1500 frames and this set of frames constitutes the input of the dataset.

### Graph Input Data Visual Representation

At the heart of this approach lies the graph representation of each simulation frame, the data structure capable of capturing the spatial and biochemical relationships between selected atoms in the TLR4/MD-2 complex. As shown in Figure 12, the resulting graph is a dense, web-like network comprising hundreds of nodes and thousands of edges.

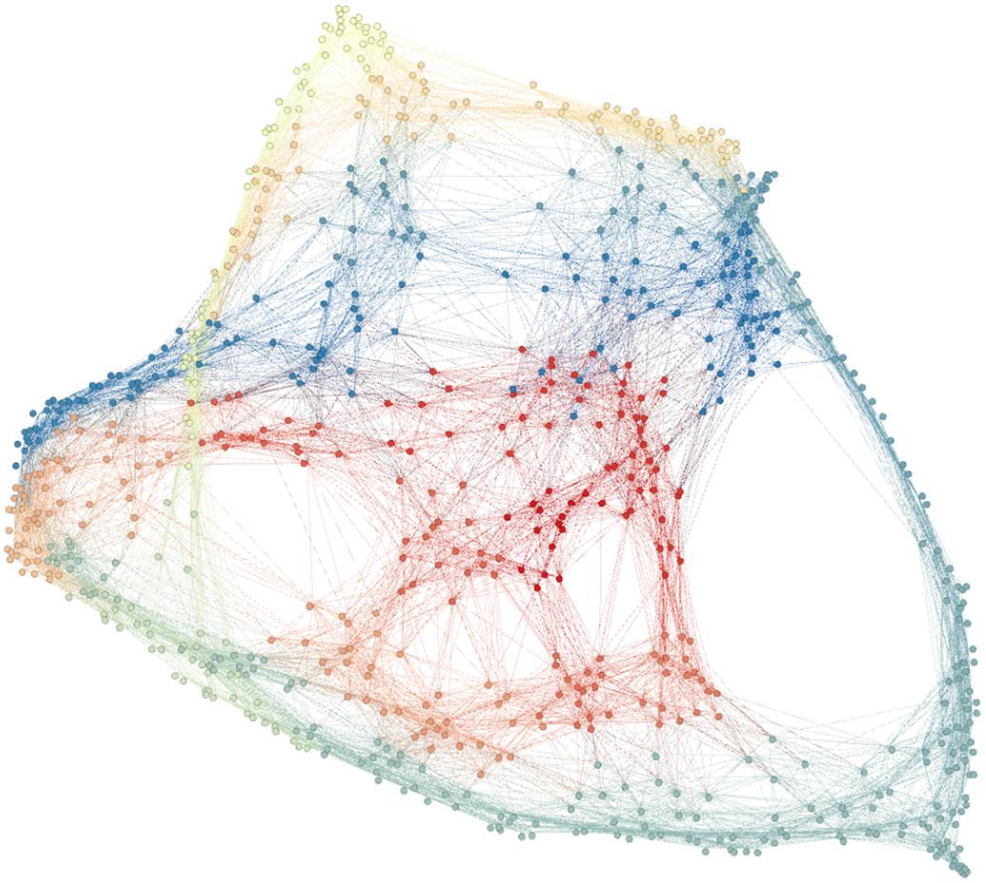

*Figure 12. Graph representation of a specific atom selection of a single frame derived from the MD simulation the GCNN receives as input data. Each node represents an atom and edges reflect spatial proximity within a 5 Å radius. Node color gradients are ap*

Each node encodes residue identity and atom type, embedded into high-dimensional vectors for the neural network. While visually abstract, this graph is a direct representation of the molecular configuration at a specific moment in time of the simulation, enabling the model to compact enriching information to infer biologically relevant patterns from the MD data.

The complexity of these input graphs underscores the challenge tackled in this work. Unlike simpler molecular systems, the TLR4/MD-2 complex undergoes subtle conformational changes tied to ligand function, which are not easily distinguishable by eye or traditional methods. The model objective is to learn these slight shifts from highly entangled graph structures, where only a small fraction of nodes may carry predictive information.

The color-coded node gradients in the figure, used here solely for visualization, highlight the vast search space the GCNN must go through to decode the underlying mechanisms of agonist and antagonist activity. This challenge motivated the extensive architectural tuning and residue selection strategies that follow in the next sections.

**Hyperparameter tuning**

The search for the best parameters for the neural network was challenging and it can be easy to get lost in the vast amount of changes that can be done that could change the outcome of the model for the better. For the GCNN architecture described, these were the best parameter configuration found to be the most robust:

- Batch Size (batch_size = 8): number of frames the model processes at once during training.
- Data Files: number of MD trajectory files used as input in the dataset.
- Selection String: a filter from MDAnalysis package to select specific atoms from the TLR4/MD-2 structure.
- Epochs (epochs = 25): number of times the model goes through the entire dataset during training.
- Learning Rate (lr = 0.0001): parameter that controls how much the model updates during training.
- Radius (r = 5.0): cutoff distance used to define which atoms are considered neighbors in the graph model.

Great effort was put into tuning the model while most conventional changes such as increasing batch size from 8 to 16, altering the learning rate and introducing dynamic scheduling, showed no meaningful improvement. Moreover, epochs went up from 5 in the first attempts to 25, as the epochs went higher occasional spikes of accuracy were spotted, indicating a need for more time for the model to learn. Another essential observation was the amount of data introduced to train the model, especially, when training with fewer MD trajectories per compound, the model tended to overfit so it was decided to train the model with all the data available to prevent this matter. The cutoff distance between atoms to define graph edges in this work was set at 5 Å following a reference paper (Viloria et al., 2017) where stable and relevant Protein Structure Network (PSN) across diverse proteins and simulation conditions were produced.

To sum up, this first phase of extensive model testing led to two key conclusions: standard hyperparameter tuning alone was insufficient to make an impact on the predictive performance and the model exhibited overfitting on small datasets and plateauing on larger ones. These outcomes highlighted the need for more strategic adjustments, particularly in how input data is selected and structured.

To do so, two variables were identified as essential to unlock the potential of the model predictive capacity: the number of MD trajectories and the selection string used to define atomic input features. From this point forward, all the data available to train is used, which is three trajectories per FP compound, up to a total of 18000 frames, 4500 each for FP11, FP12, FP18 and FP7, to ensure a robust dataset.

Additionally, the selection string, emerged as the most decisive factor in model performance Throughout the first phase of hyperparameter tuning commented before, the selection focused on the alpha carbons of the complex first and the full protein backbone, which includes alpha carbon (CA), nitrogen (N), carbonyl carbon (C) and carbonyl oxygen (O), secondly, however, both approaches seemed to introduce too much noise to capture the subtle ligand-induced conformational changes desired.

In order to find which atomic features can make an impact and improve the model ability to generalize on simulation data, it is important to carefully thread the needle that can help capture significant chemical features off the graphs. Four validation strategies, from weakest to strongest, were applied to try and prevent overfitting caused by temporal correlation:

- **Random Split Validation:** this is the standard procedure where all frames are pooled together and randomly split into 80% for training and 20% for testing without considering frame order or trajectory origin.
- **Frame Split Validation:** in this setup, the 1500 frames of each trajectory are segmented into consecutive blocks of 100 frames and stored alternatively between train and test to balance the size of the datasets while maintaining some temporal structure.
- **Manual Soft Split Validation:** here, two full trajectories of each FP compound are used for training, while the remaining third unseen trajectory is fed to testing, simulating real world new replicas for the model to predict.
- **Manual Strong Split Validation:** this is the most stringent approach because the model is trained exclusively on all three trajectories of FP11 and FP12, which have the strongest bioactivity, and tested entirely on all three trajectories of FP18 and FP7. This ultimate validation could prove whether the model has learnt structural patterns that generalize beyond specific ligands and not just memorized from the training set.

These validation techniques form the foundation of the analysis that will be discussed next along with the different atomic selection strings explored, willing to play a critical role in the model potential to capture relevant structural features that could differentiate TLR4/MD-2 agonists from TLR4/MD-2 antagonists. Therefore, to evaluate its impact, four biologically and structurally driven selections were tested:

- **MD-2** Protein **Backbone** (CA, N, C, O).
- **MD-2 Polar Region** previously reported to undergo conformational changes upon ligand binding.
- **MD-2 Specific Residues** within 5 Å of the ligand, identified using PyMOL proximity wizard tool.
- **Negative Control** composed of residues from a distal and functionally irrelevant part of TLR4.

These distinct atomic scopes can help demonstrate how spatial and functional relevance of the input features influence model generalization under the various validation schemes discussed above.

**MD-2 Backbone**

The first atomic selection tested included the full backbone of the MD-2 protein without taking into account the side chains. This representation aimed to capture general backbone movements of the receptor and serves as a broad structural input to assess whether overall protein dynamics encode sufficient information for ligand classification as agonist or antagonist ligand.

Random split validation showed no improvement during the first 11 epochs with just a spike in epoch 9 reaching almost 65%. However, from epoch 13 to the end the model started to learn going from 60% to a stable 80% accuracy in the last 5 epochs (Figure 13).

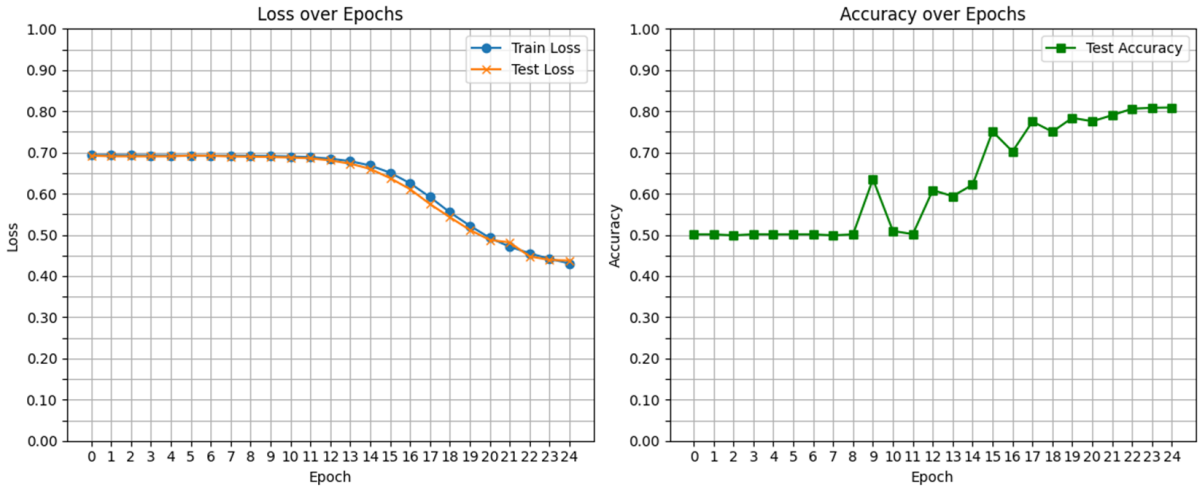


*Figure 13. Evaluation of Metrics of the GCNN performance: Training and Test Loss Function values (left) and Test Accuracy (right) of the model over epochs of the Random Split Validation for the MD-2 Backbone Atom Selection.*

On the other hand, the frame split validation exhibited an irregular behaviour with constant ups and downs in accuracy ranging from 50% to 80% until the last 8 epochs where the model stabilized around 78% accuracy (Figure 14).

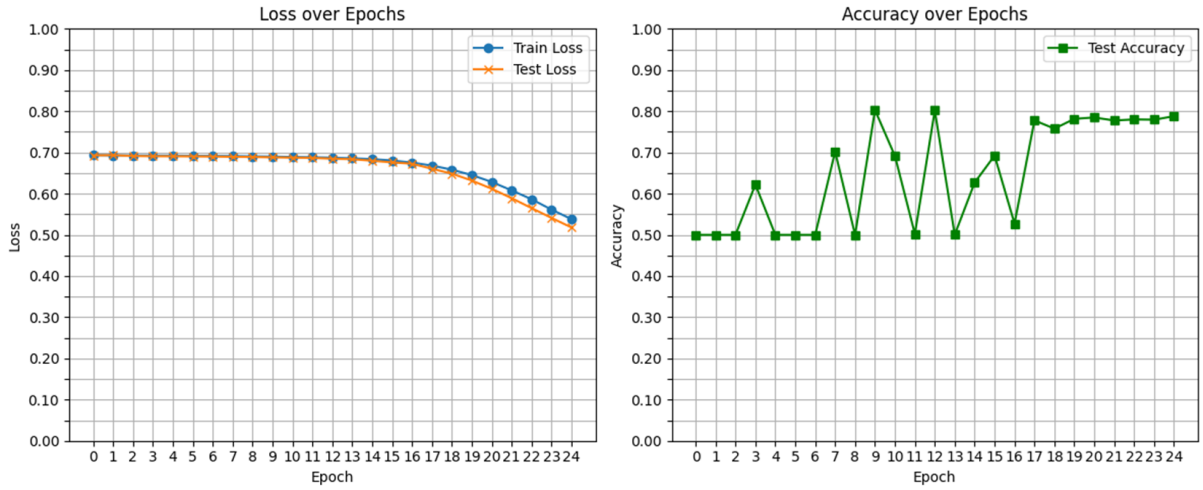


*Figure 14. Evaluation of Metrics of the GCNN performance: Training and Test Loss Function values (left) and Test Accuracy (right) of the model over epochs of the Frame Split Validation for the MD-2 Backbone Atom Selection.*

In both random and frame validations the loss function in training and test displayed similar decreasing behaviour, with a more pronounced downfall in the random split.

For the manual validation soft strategy, the model showed a slow and little improvement. Initial epochs stayed near random accuracy, 50%, starting from epoch 7, the test accuracy began to rise modestly but only to reach 58% by the final epoch, while test loss increased gradually (Figure 15).

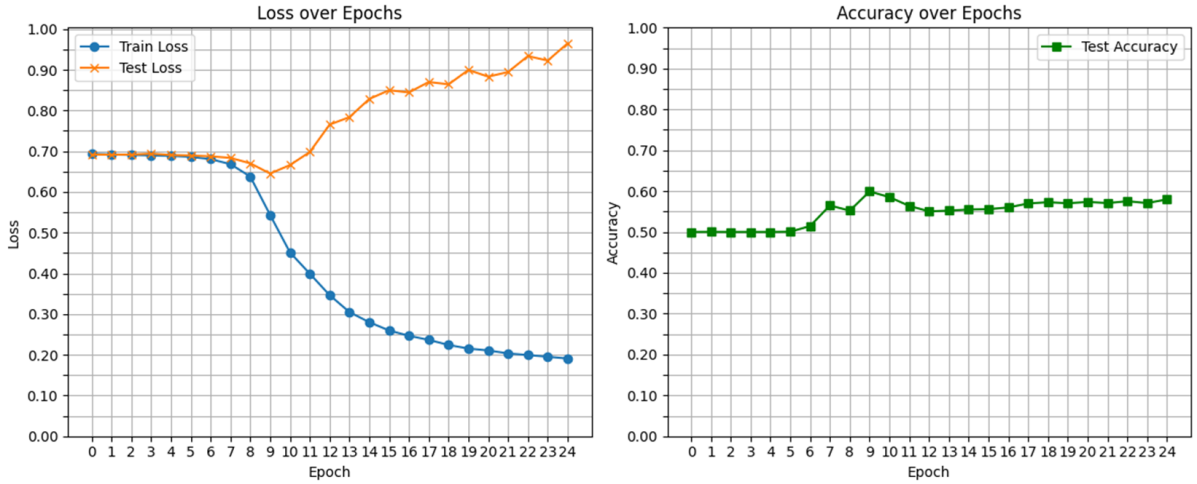


*Figure 15. Evaluation of Metrics of the GCNN performance: Training and Test Loss Function values (left) and Test Accuracy (right) of the model over epochs of the Manual Soft Split Validation for the MD-2 Backbone Atom Selection.*

The strongest manual validation proved to be more challenging. Accuracy remained around 50% throughout all epochs, even decreasing to 38% in epoch 4, showing no sign of improvement, despite training loss decreasing steadily. Meanwhile, test loss surged drastically, surpassing values above 3.5 even (Figure 16).

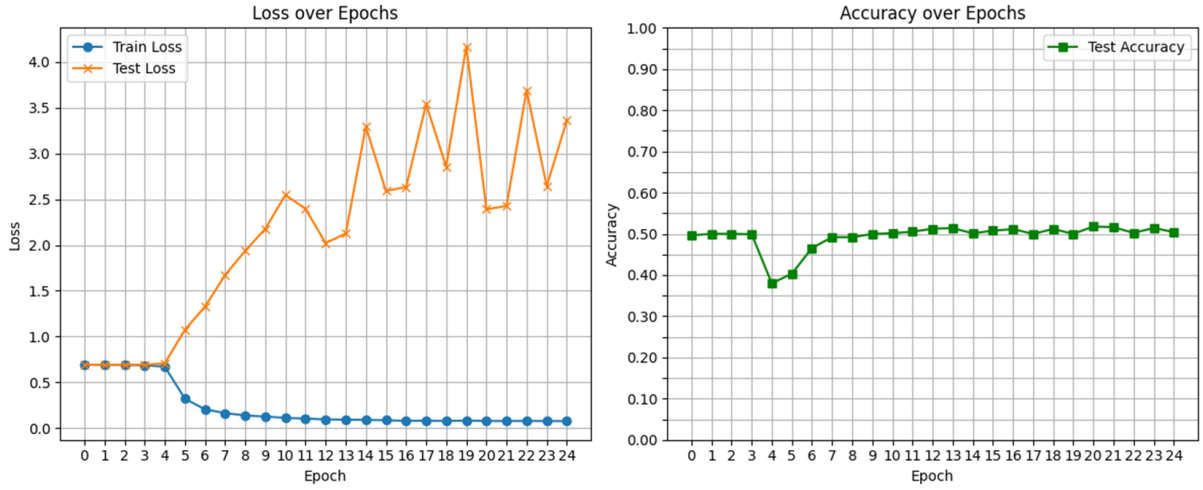


*Figure 16. Evaluation of Metrics of the GCNN performance: Training and Test Loss Function values (left) and Test Accuracy (right) of the model over epochs of the Manual Strong Split Validation for the MD-2 Backbone Atom Selection.*

Taken together, these results demonstrate that while random split and frame-based validation can reach test accuracies up to 80%, this performance is misleading and primarily driven by overfitting to patterns that lack biological relevance. The manual validations, which better reflect real generalization scenarios, reveal that the model fails to distinguish meaningful features when tested on unseen ligand classes or trajectories.

This disparity suggests that the MD-2 backbone selection, despite being a natural first approximation, is too coarse and includes conformationally irrelevant or redundant information. As a result, the model captures superficial trends within individual trajectories rather than learning robust and transferable features that differentiate agonists from antagonists.

**MD-2 Polar Region**

The limitations of backbone-only atomic selection revealed the need for more focused ligand-relevant atomic representations. An approach supported by research activity was the second posture adopted to truly unravel the underlying structural determinants of receptor-ligand interaction.

Despite occasional spikes in test accuracy, the random split showed erratic behaviour as it stagnated in 50% most of the time while train and test loss functions remained constant with no improvement either way (Figure 17).

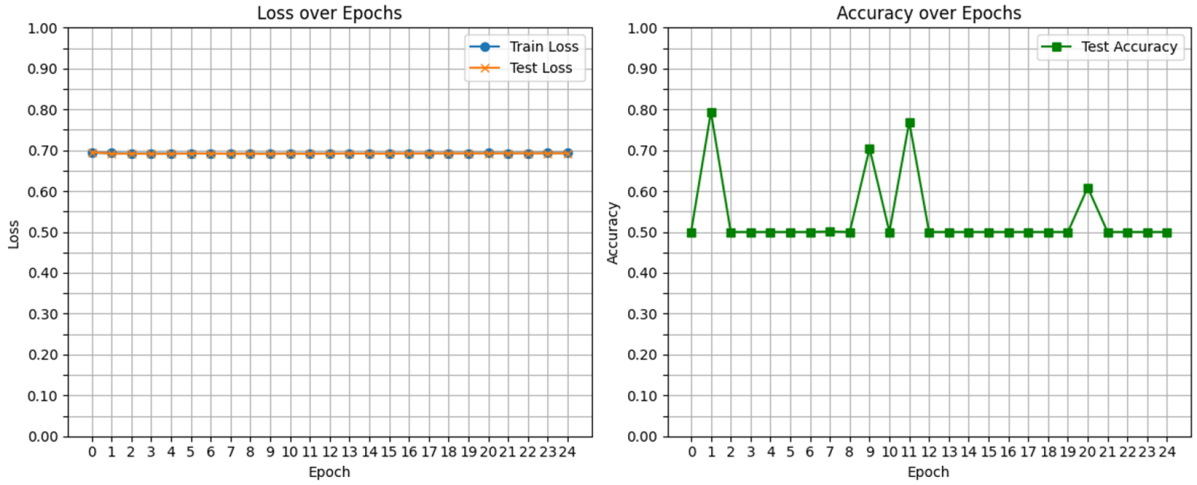


*Figure 17. Evaluation of Metrics of the GCNN performance: Training and Test Loss Function values (left) and Test Accuracy (right) of the model over epochs of the Random Split Validation for the MD-2 Polar Region Atom Selection.*

In contrast, the frame validation results exhibited a fast learning trajectory in predictions. Test accuracy consistently improved from the first epoch, surpassing 85% in the last with both loss functions decreasing at a progressive pace (Figure 18).

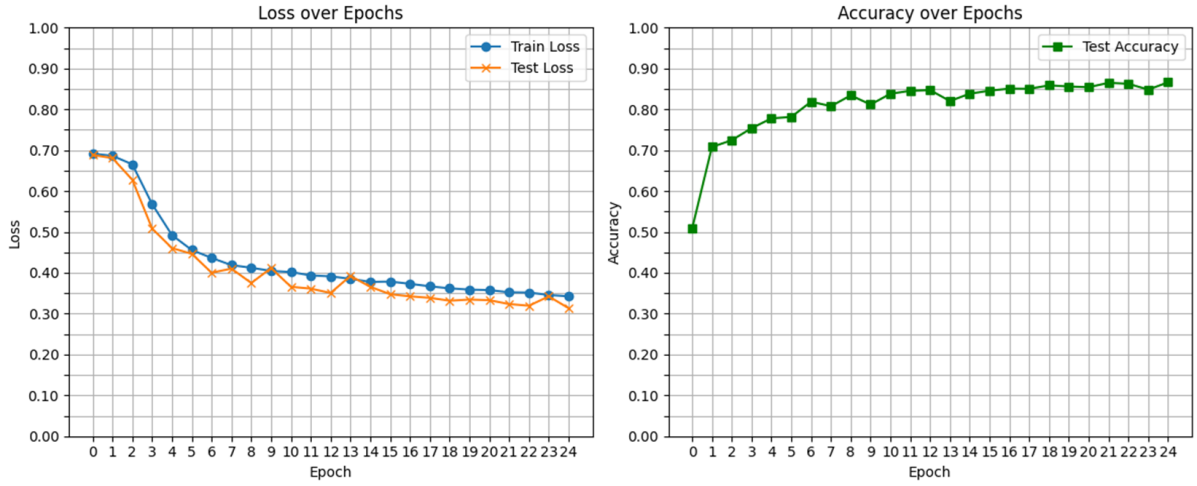


*Figure 18. Evaluation of Metrics of the GCNN performance: Training and Test Loss Function values (left) and Test Accuracy (right) of the model over epochs of the Frame Split Validation for the MD-2 Polar Region Atom Selection.*

In the soft manual split, the model performed poor test accuracy across epochs, lowering accuracy under random chance, fluctuating around 40% and 43%, even when training loss showed a clear decrease but with high test loss values, fluctuating around 40–43%, despite a clear and steady decline in training loss (Figure 19).

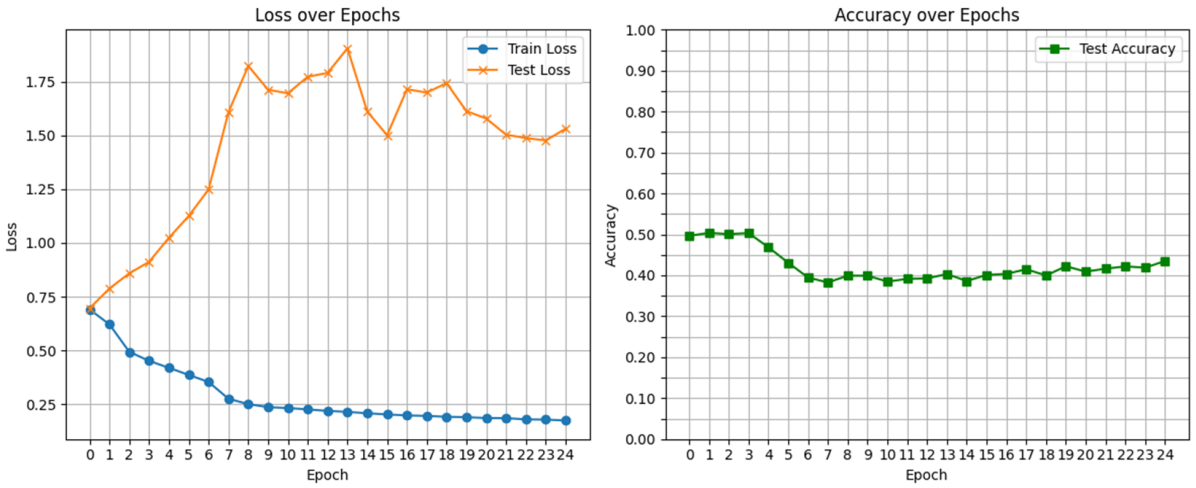


*Figure 19. Evaluation of Metrics of the GCNN performance: Training and Test Loss Function values (left) and Test Accuracy (right) of the model over epochs of the Manual Soft Split Validation for the MD-2 Polar Region Atom Selection.*

The strong manual split performance indicates that results plateau over 50%, not better than random accuracy, and even trending toward 47% in the final epochs. While training loss decreased fast, test loss continued to climb past 1.5 values (Figure 20).

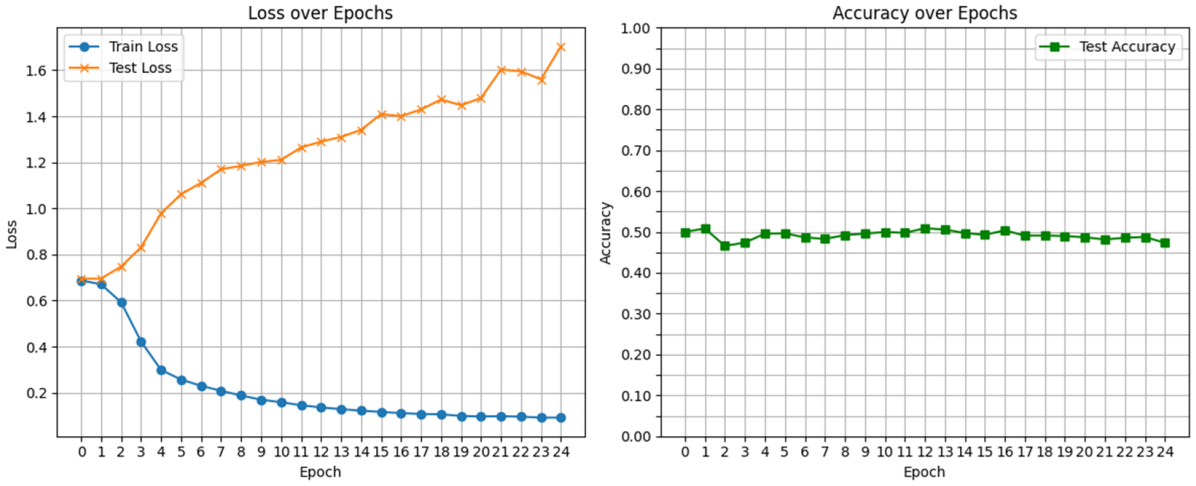


*Figure 20. Evaluation of Metrics of the GCNN performance: Training and Test Loss Function values (left) and Test Accuracy (right) of the model over epochs of the Manual Strong Split Validation for the MD-2 Polar Region Atom Selection.*

Although the focus on relevant residues looked promising, the MD-2 polar region selection generally failed to produce reliable generalization under rigorous evaluation. Random split validation reported irregular behaviour and while frame-based validation initially suggested improvement, with accuracies surpassing 85%, indicating that the model may be capturing time-dependent conformational shifts within ligand-bound trajectories, performance sharply declined under manual validation. Both soft and strong showed the model struggled to generalize beyond specific replicas or ligand classes.

This divergence between validation strategies could mean that the random nature of frame splitting still introduces temporal leakage that undermines meaningful pattern recognition and emphasizes the need for more targeted atomic selections capable of resolving the overfitting issue.

**MD-2 Specific Residues**

After highlighting the limited capacity of the MD-2 polar selection to resolve the broader conformational distinctions necessary for agonist/antagonist ligand classification, attention shifted toward a more personalized and concise system-specific atomic selection. This approach aimed to directly capture the physical interactions driving conformational change. Using the PyMOL proximity tool, the PDB of each of the four FP compounds were analyzed over the stable 100 ns segment of the simulation and residues within 5 Å of the ligand were selected due to their proximity to the object of study. This MD-2 specific residue selection was designed to focus on local perturbations likely missed by the previous selections.

The model using the random split validation grew up rapidly to 70% and then went on until reaching around 90% accuracy by the end. Additionally, training and test loss functions decreased pretty fast too in the same way (Figure 21).

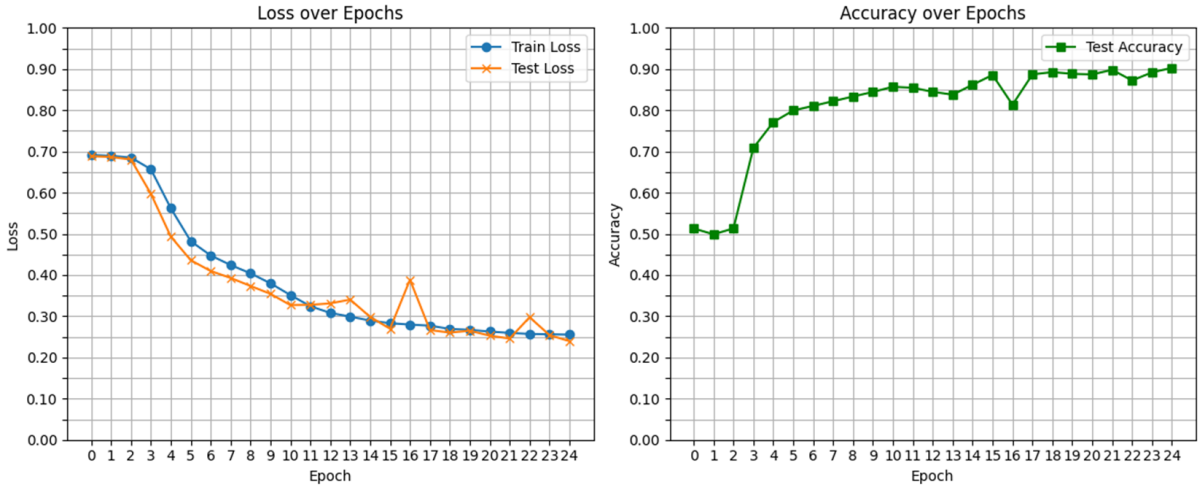


*Figure 21. Evaluation of Metrics of the GCNN performance: Training and Test Loss Function values (left) and Test Accuracy (right) of the model over epochs of the Random Split Validation for the MD-2 Specific Residues Atom Selection.*

When evaluated with frame split validation, the model also demonstrated clear learning progress. Accuracy increased reasonably with few downs reaching final 78% accuracy with corresponding improvements in both loss functions (Figure 22).

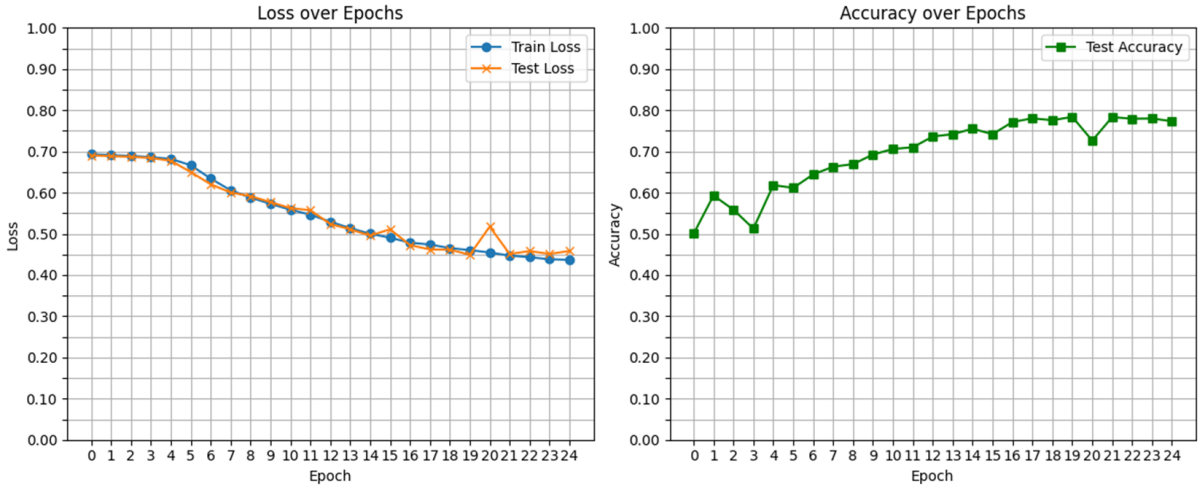


*Figure 22. Evaluation of Metrics of the GCNN performance: Training and Test Loss Function values (left) and Test Accuracy (right) of the model over epochs of the Frame Split Validation for the MD-2 Specific Residues Atom Selection.*

With soft manual validation, the results showed a modest learning pace, where the test accuracy climbed from a baseline of 50% to over 61%, while the train loss reduced and the test loss did not rise up and stabilised in a mid-range (Figure 23).

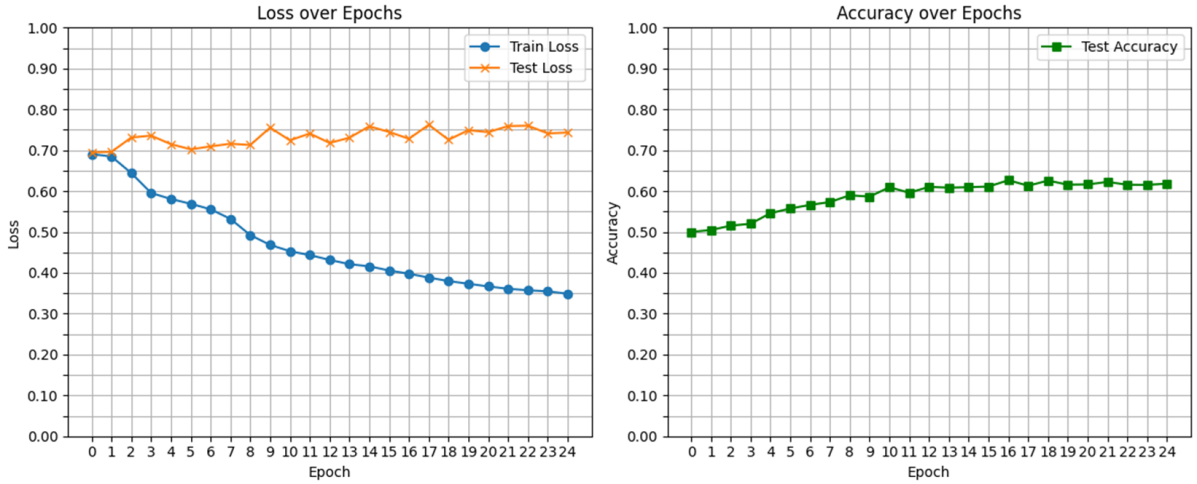


*Figure 23. Evaluation of Metrics of the GCNN performance: Training and Test Loss Function values (left) and Test Accuracy (right) of the model over epochs of the Manual Soft Split Validation for the MD-2 Specific Residues Atom Selection.*

The strong manual validation results revealed a quick jump to 55% accuracy but then fluctuated throughout all epochs around this value. Moreover, while training loss decreased to very low values the test loss function was high exceeding values of 2.0 for most of the time (Figure 24).

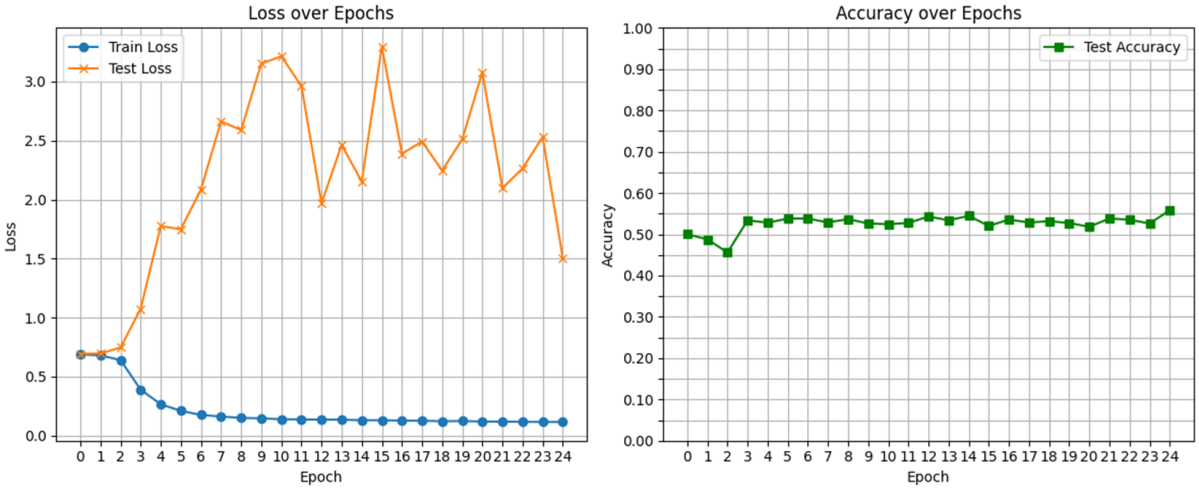


*Figure 24. Evaluation of Metrics of the GCNN performance: Training and Test Loss Function values (left) and Test Accuracy (right) of the model over epochs of the Manual Strong Split Validation for the MD-2 Specific Residues Atom Selection.*

The MD-2 specific residue selection marked a clear improvement over previous atom selection strategies, with all validation types showing enhanced performance over them, particularly in terms of test accuracy. The random split reached accuracy values above 90% and the frame validation up to beyond 78%, suggesting that the model is now capturing more relevant structural signals tied to ligand binding. Even under the more stringent manual validation schemes, the model moved past the 50% accuracy threshold, something not achieved with earlier selections, highlighting that proximity-based residue targeting provides a more informative and biologically grounded input.

It is important to remember that while this atom selection displayed the best prediction performance, overfitting remains the major issue. In the manual validation settings, the gap formed by the trend of training loss rapidly decreasing and reaching very low values whereas the test loss increases indicates that the model struggles to generalize reliably across different ligand contexts. Overfitting is a persistent challenge that needs to be addressed properly and be cautious when interpreting performance results.

**Negative control**

To verify that the model's performance is not simply a product of training dynamics or architecture bias, a negative control experiment was conducted. For this purpose, a random subset of residues was chosen from one of the TLR4 rings, structurally distant from the MD-2 binding pocket and thus not expected to contribute to ligand recognition. This selection served as a baseline to test whether models trained on unrelated atoms can still extract predictive patterns. If performance collapses under these conditions, it supports the idea that the previous targeted selections, particularly the MD-2 specific residues, contain relevant structural information, even when overfitting is present.

In the random split setting, the model showed almost no learning progression. While test accuracy fluctuated between 53% and 56% across all epochs, training and test loss remained flat during the process (Figure 25).

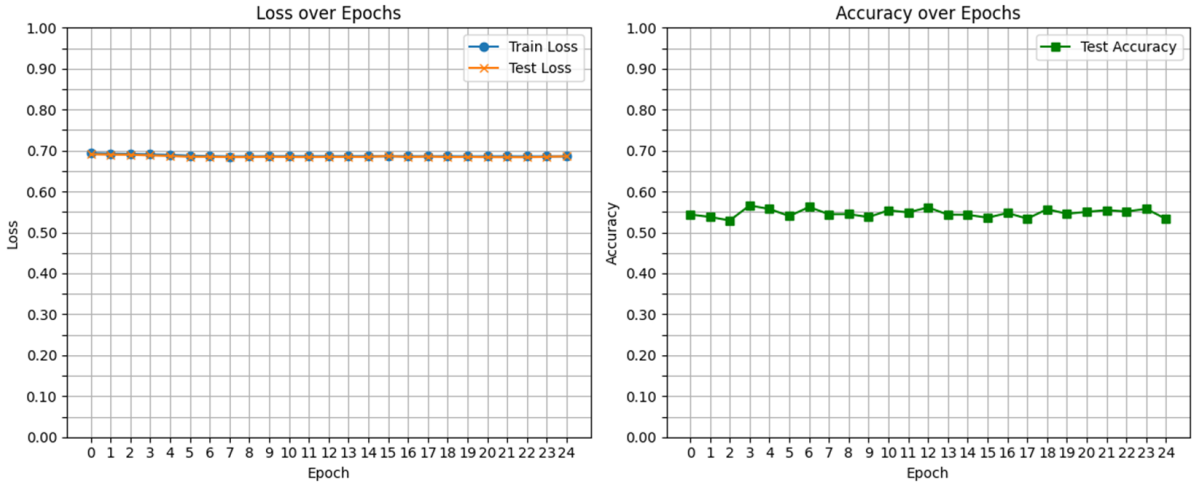


*Figure 25. Evaluation of Metrics of the GCNN performance: Training and Test Loss Function values (left) and Test Accuracy (right) of the model over epochs of the Random Split Validation for the Negative Control Atom Selection.*

The frame-based split further illustrates the absence of useful signals in this random control group. Accuracy constantly had spikes going from 50% to 60%, and both loss functions again remained flat (Figure 26).

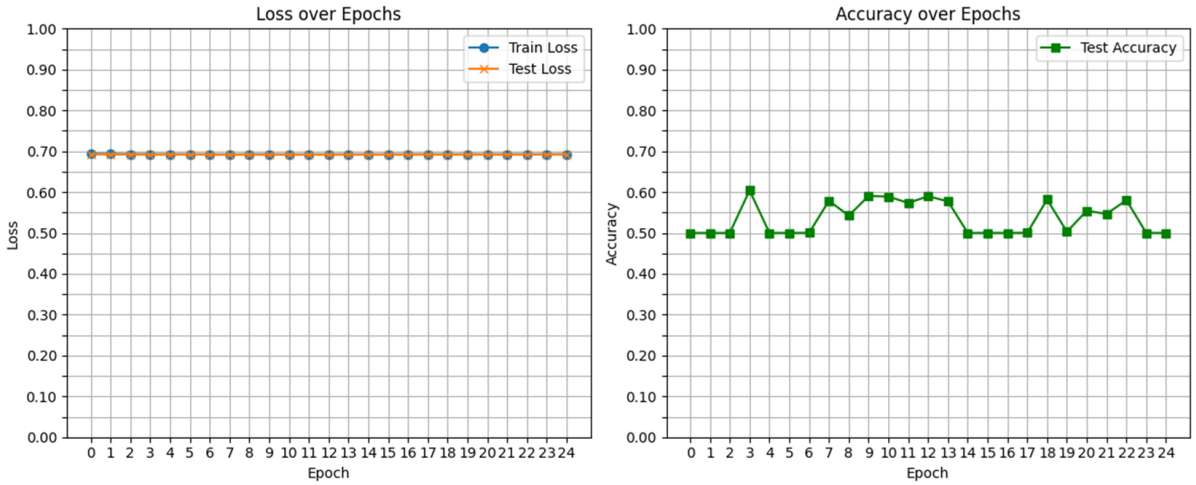


*Figure 26. Evaluation of Metrics of the GCNN performance: Training and Test Loss Function values (left) and Test Accuracy (right) of the model over epochs of the Frame Split Validation for the Negative Control Atom Selection.*

Under the soft manual validation strategy performance decreased even more, down to 30%. Test accuracy rapidly dropped 40% and did not recover, with training loss decreasing a bit and test loss climbing up to 1.0 value (Figure 27).

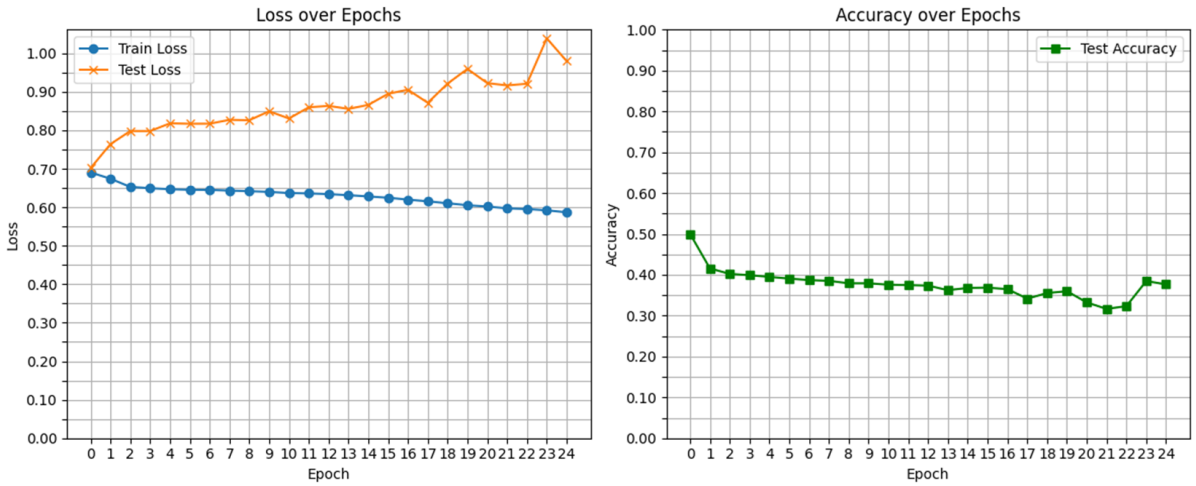


*Figure 27. Evaluation of Metrics of the GCNN performance: Training and Test Loss Function values (left) and Test Accuracy (right) of the model over epochs of the Manual Soft Split Validation for the Negative Control Atom Selection.*

The strongest validation scenario delivers poor predictive performance as expected. Constantly hovering around 50% accuracy with few down spikes to 40% and both loss functions remained flat one more time (Figure 28).

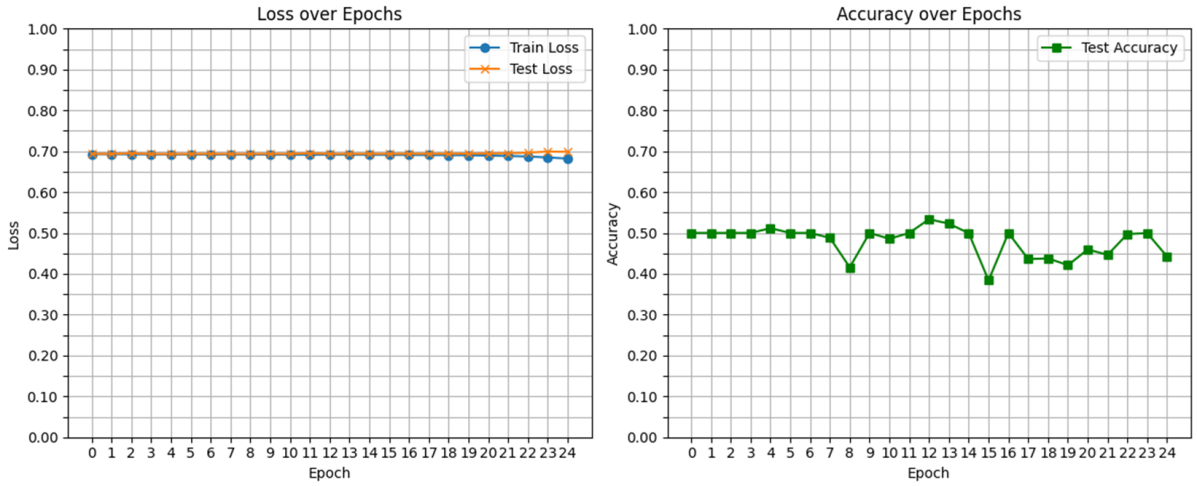


*Figure 28. Evaluation of Metrics of the GCNN performance: Training and Test Loss Function values (left) and Test Accuracy (right) of the model over epochs of the Manual Strong Split Validation for the Negative Control Atom Selection.*

The negative control experiments successfully underscored the crucial role of biologically relevant residue selection in enabling the model to extract meaningful structural information. Across all validation strategies, models consistently failed to learn, exhibiting flat and erratic performance and signs of overfitting to noise. The absence of reaching high accuracy levels and the flatness of loss functions confirmed that the model architecture alone cannot compensate for irrelevant input. On the good hand, it is worth praising that the MD-2 specific residues, despite overfitting, enabled the model to capture non-trivial, ligand-associated patterns, suggesting that the underlying conformational dynamics of the complex are indeed concentrated in that region. These results validate the targeted approach and highlight the binding interface as a key driver of structure-function encoding in this system.

**Latent Space Visualization of Learned Features**

To further assess how well the MD-2-specific model captured meaningful structure-function relationships, a two-dimensional t-SNE projection of the latent space was generated using the output embeddings from the final trained GCNN (Figure 29).

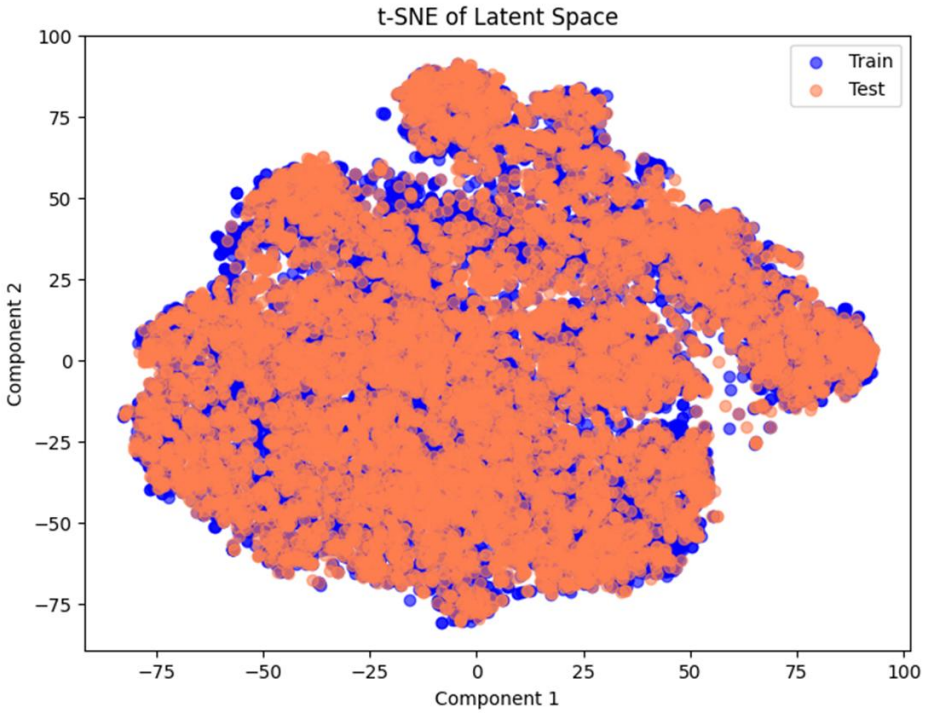


*Figure 29. t-SNE visualization of the latent space learned by the GCNN model trained on MD-2 Specific Residues Atom Selection. Blue points represent training frames and light red points represent test frames.*

In this visualization, each point corresponds to a frame from the MD simulation, with training data shown in blue and test data in light red. The resulting plot reveals a highly populated and continuous latent space, where train and test points are densely intermixed across the full range of both axes. This overlap is an indicator of the model's ability to learn generalizable features: rather than clustering train and test examples into distinct regions, which could reflect memorization, the shared embedding space suggests that the model has learned to organize the data in a consistent way. Despite the presence of overfitting in training metrics, the t-SNE plot supports the interpretation that the MD-2 residue-focused model is not far to capture real information tied to ligand-induced conformational change.

To sum up everything discussed in this section, the impact of MD-2 specific residues selection from PyMOL over the other atom selection techniques was clear and stood out as the best model parameter selection, even in the presence of overfitting. It was able to extract partially generalizable patterns tied to ligand binding, reinforcing the idea that the functional and structural dynamics crucial to discrimination between agonist and antagonist ligands are encoded primarily within the MD-2 pocket region. The model's ability to memorize training data while failing to generalize, especially under stronger validation strategies, highlights the need for additional safeguards and architectural refinement, as overfitting remains a central limitation.

Alternative strategies have been pursued to address this issue. Regularization techniques such as dropout can prevent adaptation of features and improve generalization. In addition, increasing the volume of input data by incorporating more simulation trajectories can help the model learn broader representations and add more diversity with additional FP compounds. Another approach could be the transition from standard graph convolutional networks (GCNNs) to graph attention networks (GATs), as GATs can be more sensitive to subtle and context-dependent node interactions, potentially capturing patterns of conformational change that are vital for functional activity.

Preliminary results from these suggestions have already been tested out and shown improved generalization and stability across validation strategies, even with the strongest, offering a promising path forward for future work. All together, these findings emphasize that while data quality and biological insight are foundational, their full potential is unlocked through personalized and precise refinement of both modeling approaches and architectural design.

# CONCLUSION

This study successfully integrated molecular dynamics simulation data with a graph convolutional neural network (GCNN) framework to classify ligand activity as agonist or antagonist within the TLR4/MD-2/ligand complexes. MD trajectories of 150 ns were employed along with strict validation strategies and carefully curated atom selections to explore how data size, temporal scope, and distinct molecular regions affect the ability of the model to learn biologically relevant patterns. Hyperparameter tuning played a central role in refining its predictive performance, establishing the importance of input design and parameter refining in model performance.

A significant finding was that selecting a handful of cherry picked residues with the PyMOL tool stood out as the single most impactful factor in achieving the best performance out of all trials. This approach, which included all atom types for a set of biologically relevant residues due to their spatial proximity to the ligands of study, enabled the model to better encode subtle conformational changes associated with ligand function, outperforming other configurations even under rigorous validation. This demonstrated that spatial focus near the binding interface is not only informative but essential for reliable prediction.

An important strength, and challenge at the same time, of the model lies in its ability to operate at frame-level resolution, rather than treating the trajectories as a whole, to predict ligand activity. Despite the complexity of the input graph, the model could isolate signal from noise when trained on biologically meaningful subsets. Visualization of the latent space using t-SNE on the best model revealed dense but overlapping distributions between training and test data, indicating a smooth embedding space and partial generalization, validating the choice of atom selection as a solid and biologically significant strategy for interpreting protein-ligand dynamics.

While promising, this work is an early step toward building a predictive framework that combines graph neural networks with MD data. Although overfitting is the main issue, potential improvements are being actively pursued, including dropout regularization, the use of Graph Attention Networks (GATs) for subtler pattern detection or more data size with new compounds and trajectory replicas. There is still much to explore to fully unlock the potential of this approach, however, the current GCNN model already offers a powerful tool for uncovering structure-function relationships from MD simulations, paving the way for future applications in computational drug discovery.

# BIBLIOGRAPHY

All references included in the work were cited with the help of Mendeley Reference Manager Software and are listed below by order of appearance:


[1] *Faenza, I., & Blalock, W. L. (2022). Innate immunity: a balance between disease and adaption to stress. Biomolecules, 12(5), 737.*

[2] *Taguchi, T., & Mukai, K. (2019). Innate immunity signalling and membrane trafficking. Current opinion in cell biology, 59, 1-7.*

[3] *Duan, T., Du, Y., Xing, C., Wang, H. Y., & Wang, R. F. (2022). Toll-like receptor signaling and its role in cell-mediated immunity. Frontiers in immunology, 13, 812774.*

[4] *Fitzgerald, K. A., & Kagan, J. C. (2020). Toll-like receptors and the control of immunity. Cell, 180(6), 1044-1066.*

[5] *Kim, H. J., Kim, H., Lee, J. H., & Hwangbo, C. (2023). Toll-like receptor 4 (TLR4): new insight immune and aging. Immunity & Ageing, 20(1), 67.*

[6] *Kurt-Jones, E. A., Popova, L., Kwinn, L., Haynes, L. M., Jones, L. P., Tripp, R. A., ... & Finberg, R. W. (2000). Pattern recognition receptors TLR4 and CD14 mediate response to respiratory syncytial virus. Nature immunology, 1(5), 398-401.*

[7] *Park, B. S., & Lee, J. O. (2013). Recognition of lipopolysaccharide pattern by TLR4 complexes. Experimental & molecular medicine, 45(12), e66-e66.*

[8] *Park, B. S., Song, D. H., Kim, H. M., Choi, B. S., Lee, H., & Lee, J. O. (2009). The structural basis of lipopolysaccharide recognition by the TLR4–MD-2 complex. Nature, 458(7242), 1191-1195.*

[9] *Matamoros-Recio, A., Franco-Gonzalez, J. F., Perez-Regidor, L., Billod, J. M., Guzman-Caldentey, J., & Martin-Santamaria, S. (2021). Full-Atom Model of the Agonist LPS-Bound Toll-Like Receptor 4 Dimer in a Membrane Environment. Chemistry–A European Journal, 27(62), 15406-15425.*

[10] *Facchini, F. A., Minotti, A., Luraghi, A., Romerio, A., Gotri, N., Matamoros-Recio, A., ... & Peri, F. (2021). Synthetic glycolipids as molecular vaccine adjuvants: mechanism of action in human cells and in vivo activity. Journal of medicinal chemistry, 64(16), 12261-12272.*

[11] *Gregg, K. A., Harberts, E., Gardner, F. M., Pelletier, M. R., Cayatte, C., Yu, L., ... & Ernst, R. K. (2017). Rationally designed TLR4 ligands for vaccine adjuvant discovery. MBio, 8(3), 10-1128.*

[12] *Lindemann, M., Zaslavskaya, M., Fiedler, M., Wilde, B., Heinemann, F. M., Heinold, A., ... & Witzke, O. (2017). Humoral and Cellular Responses to a Single Dose of Fendrix in Renal Transplant Recipients with Non-response to Previous Hepatitis B Vaccination. Scandinavian Journal of Immunology, 85(1), 51-57.*

[13] *Facchini, F. A., Zaffaroni, L., Minotti, A., Rapisarda, S., Calabrese, V., Forcella, M., ... & Peri, F. (2018). Structure–activity relationship in monosaccharide-based toll-like receptor 4 (TLR4) antagonists. Journal of Medicinal Chemistry, 61(7), 2895-2909.*

[14] *Kim, H. M., Park, B. S., Kim, J. I., Kim, S. E., Lee, J., Oh, S. C., ... & Lee, J. O. (2007). Crystal structure of the TLR4-MD-2 complex with bound endotoxin antagonist Eritoran. Cell, 130(5), 906-917.*

[15] *Dardelle, F., Phelip, C., Darabi, M., Kondakova, T., Warnet, X., Combret, E., ... & Caroff, M. (2024). Diversity, Complexity, and Specificity of Bacterial Lipopolysaccharide (LPS) Structures Impacting Their Detection and Quantification. International Journal of Molecular Sciences, 25(7), 3927.*

[16] *Yang, C., Chen, E. A., & Zhang, Y. (2022). Protein–ligand docking in the machine-learning era. Molecules, 27(14), 4568.*

[17] *Marchetti, F., Moroni, E., Pandini, A., & Colombo, G. (2021). Machine learning prediction of allosteric drug activity from molecular dynamics. The journal of physical chemistry letters, 12(15), 3724-3732.*

[18] *Patel, A. C., Sinha, S., & Palermo, G. (2024). Graph theory approaches for molecular dynamics simulations. Quarterly Reviews of Biophysics, 57, e15.*

[19] *Vishveshwara, S., Brinda, K. V., & Kannan, N. (2002). Protein structure: insights from graph theory. Journal of Theoretical and Computational Chemistry, 1(01), 187-211.*
[20] *Bougueroua, S., Bricage, M., Aboulfath, Y., Barth, D., & Gaigeot, M. P. (2023). Algorithmic graph theory, reinforcement learning and game theory in MD simulations: From 3D structures to topological 2D-molecular graphs (2D-MolGraphs) and vice versa. Molecules, 28(7), 2892.*
[21] *Reiser, P., Neubert, M., Eberhard, A., Torresi, L., Zhou, C., Shao, C., ... & Friederich, P. (2022). Graph neural networks for materials science and chemistry. Communications Materials, 3(1), 93.*
[22] *Hofstetter, A., Böselt, L., & Riniker, S. (2022). Graph-convolutional neural networks for (QM) ML/MM molecular dynamics simulations. Physical Chemistry Chemical Physics, 24(37), 22497-22512.*
[23] *Li, C., Liu, J., Chen, J., Yuan, Y., Yu, J., Gou, Q., ... & Pu, X. (2022). An interpretable convolutional neural network framework for analyzing molecular dynamics trajectories: A case study on functional states for G-protein-coupled receptors. Journal of Chemical Information and Modeling, 62(6), 1399-1410.*
[24] *Vijayan, R., & Mohler, G. (2018, October). Forecasting retweet count during elections using graph convolution neural networks. In 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA) (pp. 256-262). IEEE.*
[25] *Zhang, S., Tong, H., Xu, J., & Maciejewski, R. (2019). Graph convolutional networks: a comprehensive review. Computational Social Networks, 6(1), 1-23.*
[26] *Pérez-Regidor, L., Guzmán-Caldentey, J., Oberhauser, N., Punzón, C., Balogh, B., Pedro, J. R., ... & Martín-Santamaría, S. (2022). Small molecules as toll-like receptor 4 modulators drug and in-house computational repurposing. Biomedicines, 10(9), 2326.*
[27] *Ohto, U., Fukase, K., Miyake, K., & Satow, Y. (2007). Crystal structures of human MD-2 and its complex with antiendotoxic lipid IVa. Science, 316(5831), 1632-1634.*
[28] *Wang, J., Wang, W., Kollman, P. A. & Case, D. A. Automatic atom type and bond type perception in molecular mechanical calculations. J. Mol. Graph Model., 25, 247–260 (2006).*
[29] *Wang, J., Wolf, R. M., Caldwell, J. W., Kollman, P. A. & Case, D. A. Development and testing of a general amber force field. J. Comput. Chem., 25, 1157–1174 (2004).*
[30] *Nauchitel', V. V. Energy distribution function for the NVT canonical ensemble. Mol. Phys., 42, 1259–1265 (1981).*
[31] *Jorgensen, W. L., Chandrasekhar, J., Madura, J. D., Impey, R. W. & Klein, M. L. Comparison of simple potential functions for simulating liquid water. J. Chem. Phys., 79, 926–935 (1983).*
[32] *Essmann, U. et al. A smooth particle mesh Ewald method. J. Chem. Phys., 103, 8577–8593 (1995).*
[33] *Humphrey, W., Dalke, A. & Schulten, K. VMD: Visual molecular dynamics. J. Mol. Graph., 14, 33–38 (1996).*
[34] *Roe, D. R. & Cheatham, T. E. PTRAJ and CPPTRAJ: Software for Processing and Analysis of Molecular Dynamics Trajectory Data. J. Chem. Theory Comput., 9, 3084–3095 (2013).*
[35] *Gowers, R. J., Linke, M., Barnoud, J., Reddy, T. J. E., Melo, M. N., Seyler, S. L., ... & Beckstein, O. (2019). MDAnalysis: a Python package for the rapid analysis of molecular dynamics simulations (No. LA-UR-19-29136). Los Alamos National Laboratory (LANL), Los Alamos, NM (United States).*
[36] *Michaud-Agrawal, N., Denning, E. J., Woolf, T. B., & Beckstein, O. (2011). MDAnalysis: a toolkit for the analysis of molecular dynamics simulations. Journal of computational chemistry, 32(10), 2319-2327.*
[37] *Pezoa, F., Reutter, J. L., Suarez, F., Ugarte, M., & Vrgoč, D. (2016, April). Foundations of JSON schema. In Proceedings of the 25th international conference on World Wide Web (pp. 263-273).*
[38] *Hagberg, A., Swart, P. J., & Schult, D. A. (2008). Exploring network structure, dynamics, and function using NetworkX (No. LA-UR-08-05495; LA-UR-08-5495). Los Alamos National Laboratory (LANL), Los Alamos, NM (United States).*
[39] *Bastian, M., Heymann, S., & Jacomy, M. (2009, March). Gephi: an open source software for exploring and manipulating networks. In Proceedings of the international AAAI conference on web and social media (Vol. 3, No. 1, pp. 361-362).*
[40] *Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., ... & Lerer, A. (2017). Automatic differentiation in pytorch.*

[41] *Crampon, K., Giorkallos, A., Vigouroux, X., Baud, S., & Steffenel, L. A. (2023). Heterogeneous graph convolutional neural network for protein-ligand scoring. Exploration of Drug Science, 126-139.*
[42] *Agarap, A. F. (1803). Deep learning using rectified linear units (relu)(2018). arXiv preprint arXiv:1803.08375, 114.*
[43] *Wang, Y., Hu, L., Wu, Y., & Gao, W. (2022). Graph multihead attention pooling with self-supervised learning. Entropy, 24(12), 1745.*
[44] *Mao, A., Mohri, M., & Zhong, Y. (2023). Cross-entropy loss functions: Theoretical analysis and applications. In International conference on Machine learning (pp. 23803-23828). PMLR.*
[45] *Gu, M., Zhang, Y., Wen, Y., Ai, G., Zhang, H., Wang, P., & Wang, G. (2023). A lightweight convolutional neural network hardware implementation for wearable heart rate anomaly detection. Computers in Biology and Medicine, 155, 106623.*
[46] *Viloria, J. S., Allega, M. F., Lambrughi, M., & Papaleo, E. (2017). An optimal distance cutoff for contact-based Protein Structure Networks using side-chain centers of mass. Sci Rep.*

Complementary documentation for the official resources of the computational tools and packages used throughout this work is provided below:

- *The PyMOL Molecular Graphics System, Version 3.0 Schrödinger, LLC. Retrieved from: https://www.pymol.org/*
- *Maestro, Schrödinger, LLC, New York, NY, 2021. Retrieved from: https://www.schrodinger.com/platform/products/maestro/*
- *Gaussian 09, Revision A.02, M. J. Frisch, et al., Gaussian, Inc., Wallingford CT, 2016. Retrieved from: https://gaussian.com/glossary/g09/*
- *Case, D. A., et al. AMBER 2020. University of California, San Francisco (2020). Retrieved from: https://ambermd.org/*
- *Fey, M., & Lenssen, J. E. (2025). PyTorch Geometric documentation. Retrieved from: https://pytorch-geometric.readthedocs.io/en/latest/*
- *Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., et al. (2025). scikit-learn: Machine Learning in Python documentation. Retrieved from: https://scikit-learn.org/stable/*
- *Hunter, J. D., Caswell, T. A., Droettboom, M., Firing, E., et al. (2025). Matplotlib documentation. Retrieved from: https://matplotlib.org/*
- *McKinney, W., Reback, J., & contributors. (2025). pandas documentation. Retrieved from: https://pandas.pydata.org/*
- *Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., et al. (2025). NumPy documentation. Retrieved from: https://numpy.org/*
- *da Costa-Luis, C., & contributors. (2025). tqdm documentation. Retrieved from: https://tqdm.github.io/*

# ANNEX

| ID | Code | Full Name | Type |
|---|---|---|---|
| **0** | ACE | Acetyl group | N-terminal cap |
| **1** | ALA | Alanine | Amino acid |
| **2** | ARG | Arginine | Amino acid |
| **3** | ASN | Asparagine | Amino acid |
| **4** | ASP | Aspartic acid | Amino acid |
| **5** | CYS | Cysteine | Amino acid |
| **6** | CYX | Cystine | Disulfide-bonded cysteine |
| **7** | GLN | Glutamine | Amino acid |
| **8** | GLU | Glutamic acid | Amino acid |
| **9** | GLY | Glycine | Amino acid |
| **10** | HIE | Histidine (ε-protonated) | Histidine variant |
| **11** | ILE | Isoleucine | Amino acid |
| **12** | LEU | Leucine | Amino acid |
| **13** | LYS | Lysine | Amino acid |
| **14** | MET | Methionine | Amino acid |
| **15** | NME | N-Methylamide | C-terminal cap |
| **16** | PHE | Phenylalanine | Amino acid |
| **17** | PRO | Proline | Amino acid |
| **18** | SER | Serine | Amino acid |
| **19** | THR | Threonine | Amino acid |
| **20** | TRP | Tryptophan | Amino acid |
| **21** | TYR | Tyrosine | Amino acid |
| **22** | VAL | Valine | Amino acid |
| **23** | HID | Histidine (δ-protonated) | Histidine variant |

*Table 1. Residue IDs, Annotation Codes, Names and Types for Graph Embedding. These annotations encode molecular information used in the residue branch of the GCNN.*

| ID | Code | Full Name | Type |
|---|---|---|---|
| 0 | 2C | C2 ($sp^2$ carbon) | Carbon |
| 1 | 3C | C3 ($sp^3$ carbon) | Carbon |
| 2 | C | Backbone Carbon (carbonyl C) | Carbon |
| 3 | C* | Heteroatom-bound Carbon | Carbon |
| 4 | C8 | Specialized aromatic/labeling carbon | Carbon |
| 5 | CA | Cα (alpha carbon) | Carbon |
| 6 | CB | Cβ (beta carbon) | Carbon |
| 7 | CC | Carbon in carbonyl | Carbon |
| 8 | CN | Carbon bound to nitrogen | Carbon |
| 9 | CO | Carbon in carboxyl/carbonyl group | Carbon |
| 10 | CR | Aromatic carbon | Carbon |
| 11 | CT | Aliphatic tetrahedral carbon | Carbon |
| 12 | CW | Carbon in cyclic ring | Carbon |
| 13 | CX | Generic $sp^2$ carbon | Carbon |
| 14 | H | Hydrogen (generic) | Hydrogen |
| 15 | H1 | Singly-bonded hydrogen | Hydrogen |
| 16 | H4 | Hydrogen on tertiary carbon | Hydrogen |
| 17 | H5 | Hydrogen on quaternary carbon | Hydrogen |
| 18 | HA | Hα (alpha hydrogen) | Hydrogen |
| 19 | HC | Hydrogen on aliphatic carbon | Hydrogen |
| 20 | HO | Hydroxyl hydrogen | Hydrogen |
| 21 | HP | Hydrogen on phosphate | Hydrogen |
| 22 | HS | Hydrogen on sulfur | Hydrogen |
| 23 | N | Backbone nitrogen (amide) | Nitrogen |
| 24 | N2 | $sp^2$ nitrogen | Nitrogen |
| 25 | N3 | $sp^3$ nitrogen | Nitrogen |
| 26 | NA | Aromatic Nitrogen | Nitrogen |
| 27 | NB | Additional aromatic nitrogen | Nitrogen |
| 28 | O | Backbone oxygen | Oxygen |
| 29 | O2 | $sp^2$ Oxygen | Oxygen |
| 30 | OH | Hydroxyl oxygen | Oxygen |
| 31 | S | Sulfur | Sulfur |
| 32 | SH | Thiol sulfur | Sulfur |

*Table 2. Atom IDs, Annotation Codes, Names and Types for Graph Embedding. These annotations encode molecular information used in the atom type branch of the GCNN.*