Explainable Artificial Intelligence in the Life Sciences

Dr. Andrea Mastropietro

Junior Research Group Leader
Department of Life Science Informatics and Data Science, B-IT
Lamarr Institute for Machine Learning and Artificial Intelligence
Rheinische Friedrich-Wilhelms-Universität Bonn

Specially Appointed Assistant Professor
Data Science Center
Nara Institute of Science and Technology (NAIST)

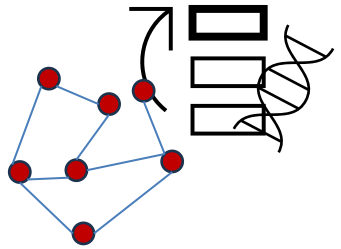
Outline

- Introduction
- Deep learning-based models
  - Explaining graph neural networks for compound activity prediction
  - Learning characteristics of graph neural networks predicting protein-ligand affinities
- Not by neural networks alone…
  - Explainability of support vector machine models
- Conclusions

Introduction

- In recent years, we noted an explosion of **deep learning** models for **life science applications**

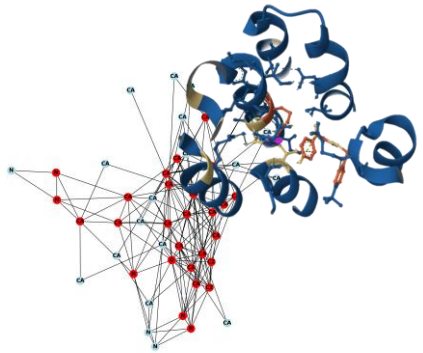
  - Disease gene discovery
  - Compound activity prediction
  - Protein-ligand interaction studies



- Huge amount of data → extremely **accurate predictions**

- Problem: Neural networks are **black-box models**

  - They learn **highly nonlinear functions**
  - Impossible to give a **direct interpretation** → impact trustworthiness



Introduction – Graph neural networks

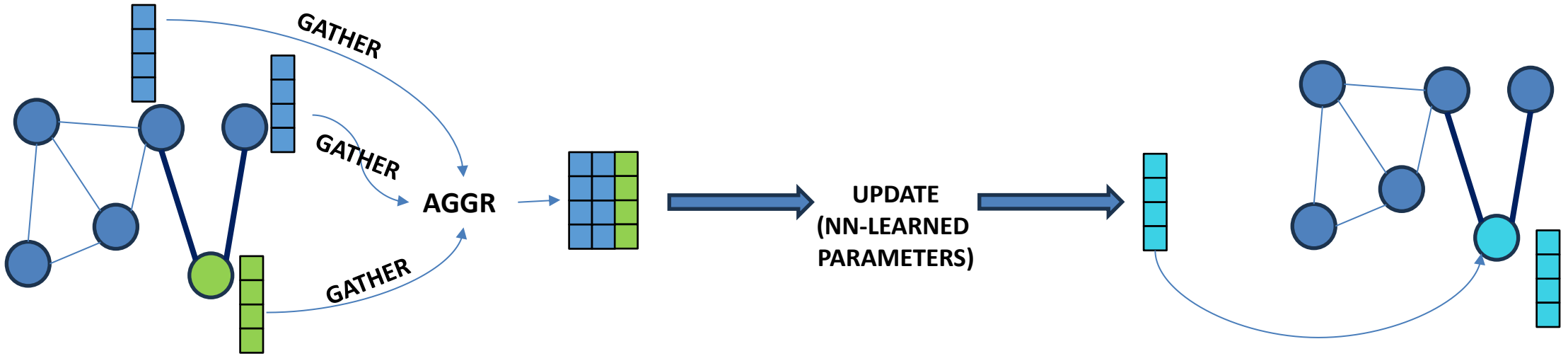
- In chemoinformatics: **network-like nature** of molecular data → massive use of **graph neural networks** (GNNs)

- GNNs learn effective representations by **leveraging graph/network topology**

- Based on the **message passing** concept

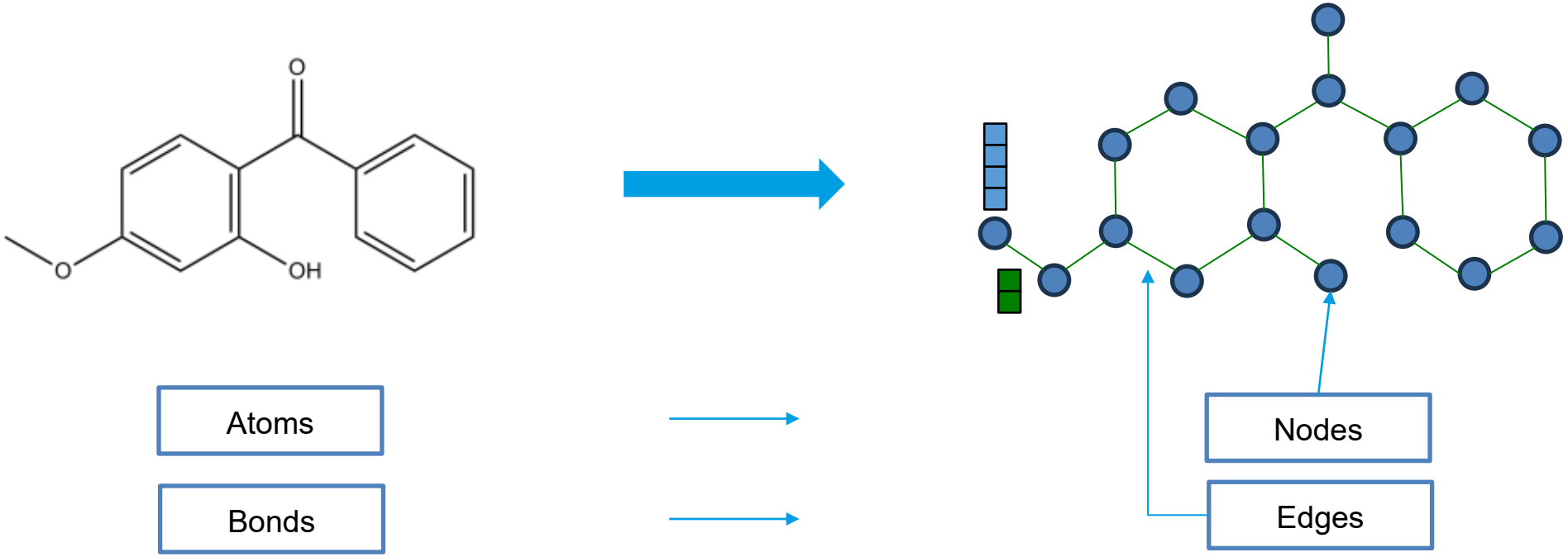
Introduction – Graph neural networks

- Message passing consists of several **main functions**
- **GATHER**: **collects information** from nodes (node feature vectors)
- **AGGR**: **aggregates information** of the current node with **information from neighbor nodes**
- **UPDATE**: generates a **node embedding** by updating the information **using a neural network**



Introduction – Graph neural networks

- Molecules need to be represented as graphs



Node features: atom type, charge, atomic number, …

Edge features: bond type, ring membership, …

Introduction – Explainable artificial intelligence

- GNNs delivered **extremely accurate** predictions in important drug discovery tasks:
  - Compound activity prediction
  - Compound potency prediction (protein-ligand affinity)
- Non-interpretable models → **explainable artificial intelligence** (XAI) comes into play
  - Determine **important features** for the prediction
- Question: **Can we trust** GNNs in life science applications?

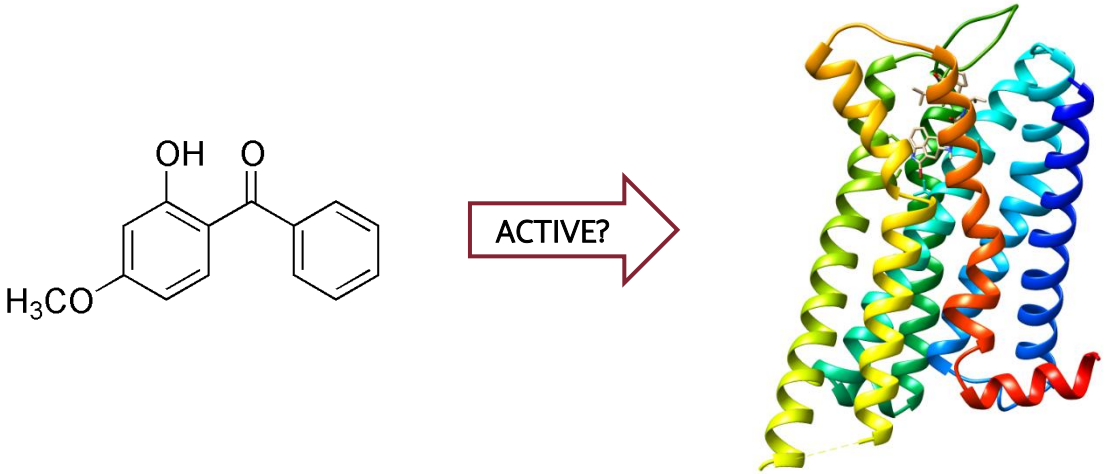
Do GNNs **learn chemical principles,** or do they simply **exploit statistical patterns** in the data?

Explaining graph neural networks for compound activity prediction

GNNs for compound activity prediction

- **Compound activity prediction**: given a chemical compound, it consists in predicting its **activity against a target** (protein, gene, enzyme, ...)

- **Binary classification** in machine learning



GNNs for compound activity prediction

- GNNs deliver **high classification accuracy**
- Question: What is the **explanation** of the prediction **from a chemical point of view**?
- Molecules are characterized by atoms and **bonds** between them, **forming chemical structures**
- **Edges** in graphs are the **means through which information is distributed**

XAI for GNNs in chemoinformatics

Explain a GNN prediction in terms of
important edges (bonds)

- Goal: Find **critical edges** forming **important substructures** responsible for **molecular activity**

- **EdgeSHAPer**[1]: XAI method using **Shapley values** to determine **edge importance**

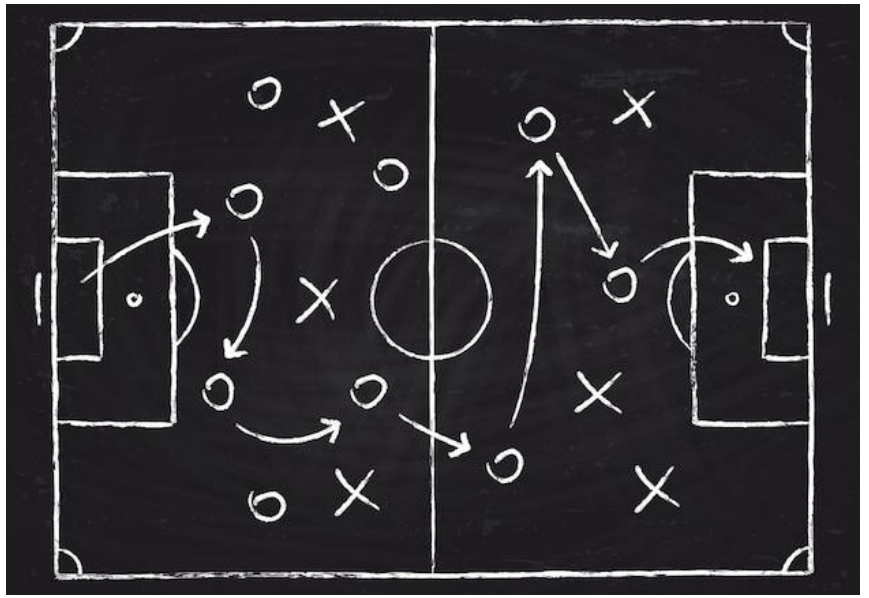
[1] Mastropietro, Andrea, et al. "EdgeSHAPer: Bond-centric Shapley value-based explanation method for graph neural networks." iScience 25.10 (2022)

Overview on Shapley values

- Concept from **cooperative game theory**[2]

- How to fairly **distribute a payout to players** who work in a **coalition** toward a **common goal** (game)

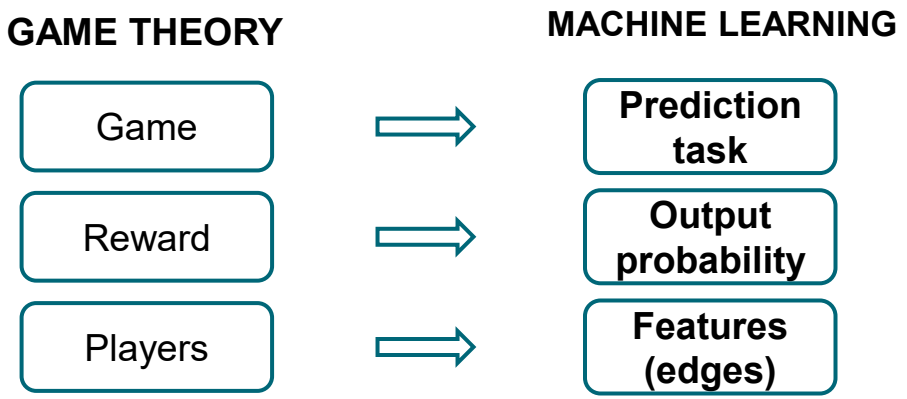
- **Shapley value**: **marginal contribution** of a player across all possible player coalitions



[2] Shapley, Lloyd S. “A value for n-person games.” Contributions to the Theory of Games 2.28 (1953): 307-317

Overview on Shapley values

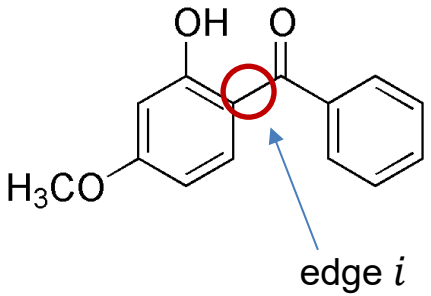
- The Shapley value concept can be **translated to machine learning**
- Consider the **prediction task** as a **collaborative game** played by features
- Shapley values can be used to assess **feature importance**



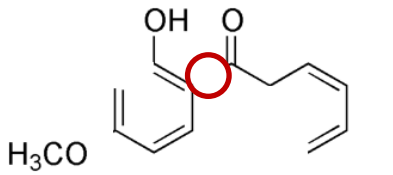
EdgeSHAPer

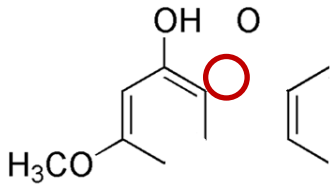
- Not feasible to compute exact Shapley values due to their **combinatorial nature**
- Approximate Shapley values using a **novel Monte Carlo sampling** of edges
- Create an information background using **randomly sampled graphs**

EdgeSHAPer

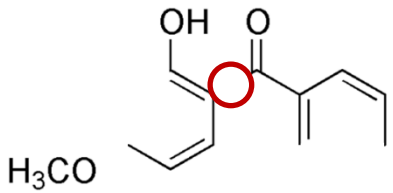


Molecule to explain





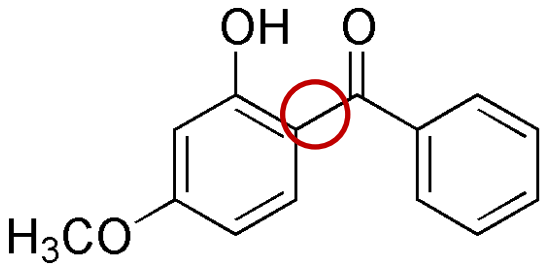
...



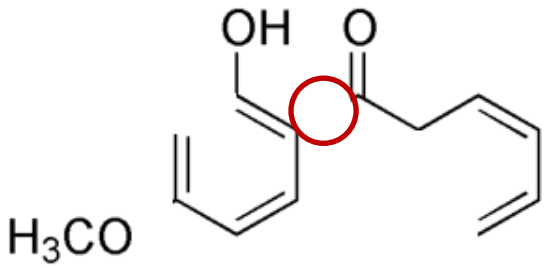
M sampled graphs

Marginal contribution to the prediction of edge i (**Shapley value**)

EdgeSHAPer

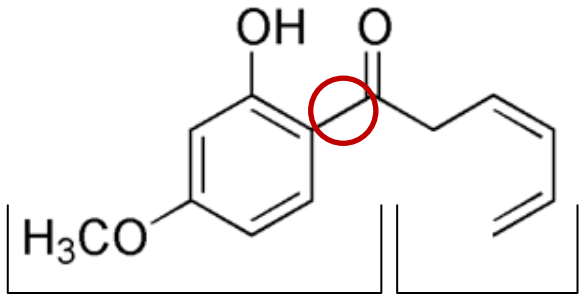


Molecule to explain

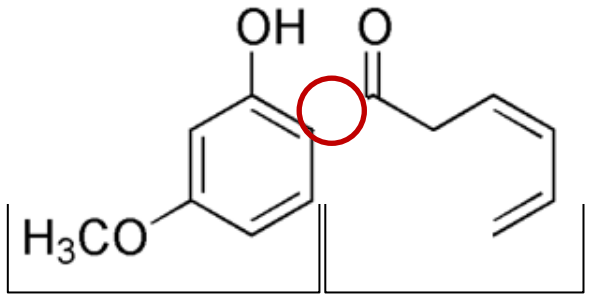


Sampled graph

- Given an edge i, to determine its **contribution** (**Shapley value**), we create two samples:

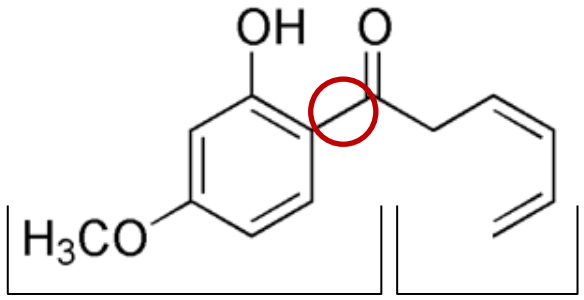


Original graph Sampled graph

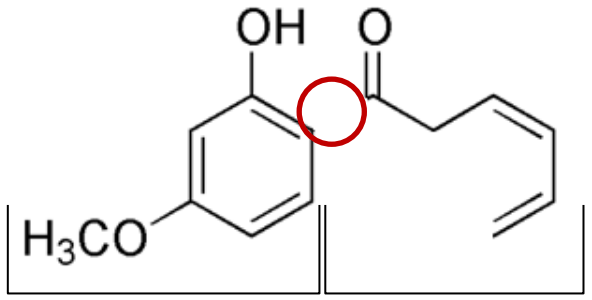


Original graph Sampled graph

EdgeSHAPer



Original graph Sampled graph



Original graph Sampled graph

- We can now compute the **marginal contribution** to the prediction of edge i, by repeating the sampling M times, as the **average difference of the predictions**:

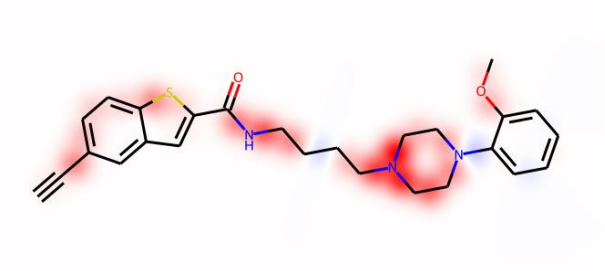
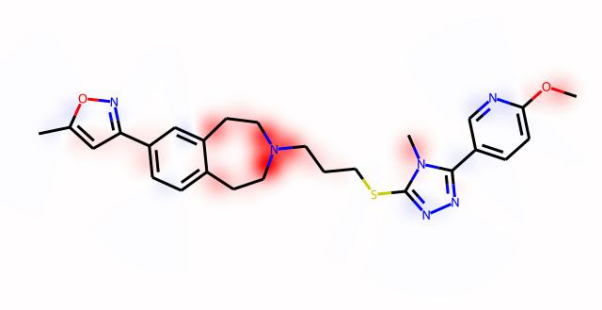
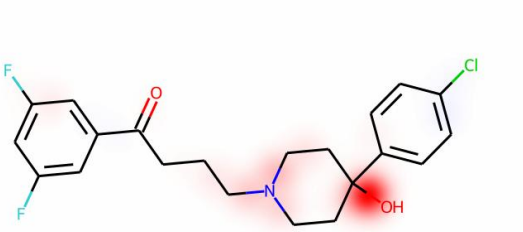
$$\widehat{\phi_i} = \frac{1}{M}\sum_{m=0}^{M-1} f\left(G^m(N, E_+)\right) - f\left(G^m\left(N, E_-\right)\right)$$

Repeat until convergence: **approximation of Shapley values**

Use case

- Task: **prediction of the activity** of molecules against the Dopamine D2 Receptor
- We trained a **graph convolutional network**, obtaining **97% accuracy**
- Question: Does the GNN **learn chemically meaningful information** to arrive at accurate predictions?

Use case

- EdgeSHAPer identifies **small**, **explicative**, and **chemically intuitive** explanations
- Highlighting **critical molecular substructures**
- **Observation:** the GNN learned **patterns in the data** that **may or may not be in line** with chemical knowledge

HIGH ACCURACY ≠ LEARNING CHEMISTRY

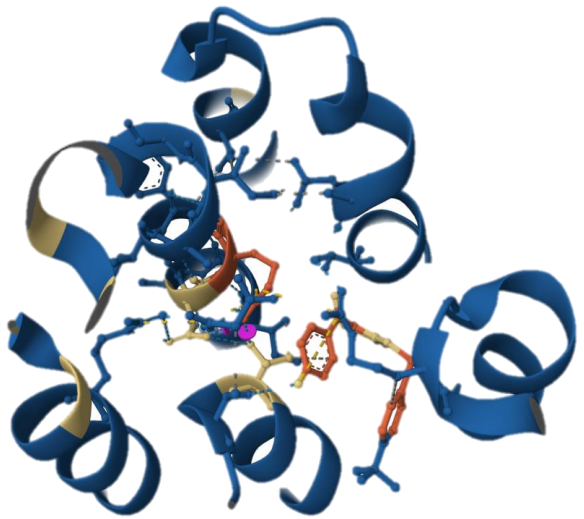
Learning characteristics of graph neural networks predicting protein-ligand affinities

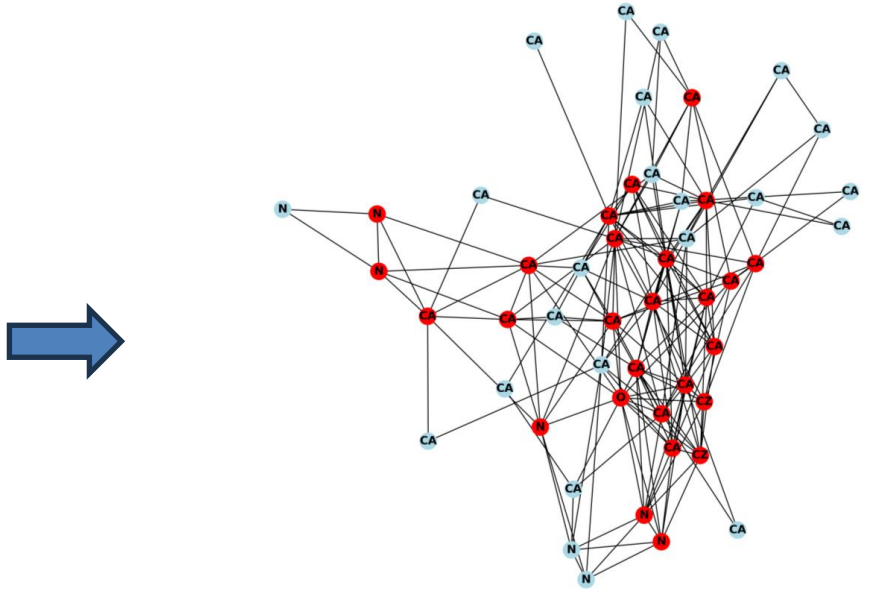
GNNs for protein-ligand interactions

- **Protein-ligand affinity prediction**: another paramount task in machine learning for drug design
- Compound **activity prediction**: active or not? (classification)
- Compound **potency prediction**: determine the **interaction affinity** of compound and protein (regression)
- GNNs have been recently employed, delivering **extremely accurate predictions** leveraging **protein-ligand interaction graphs**

Protein-ligand interaction graphs

- Standard de-facto data representation for potency prediction



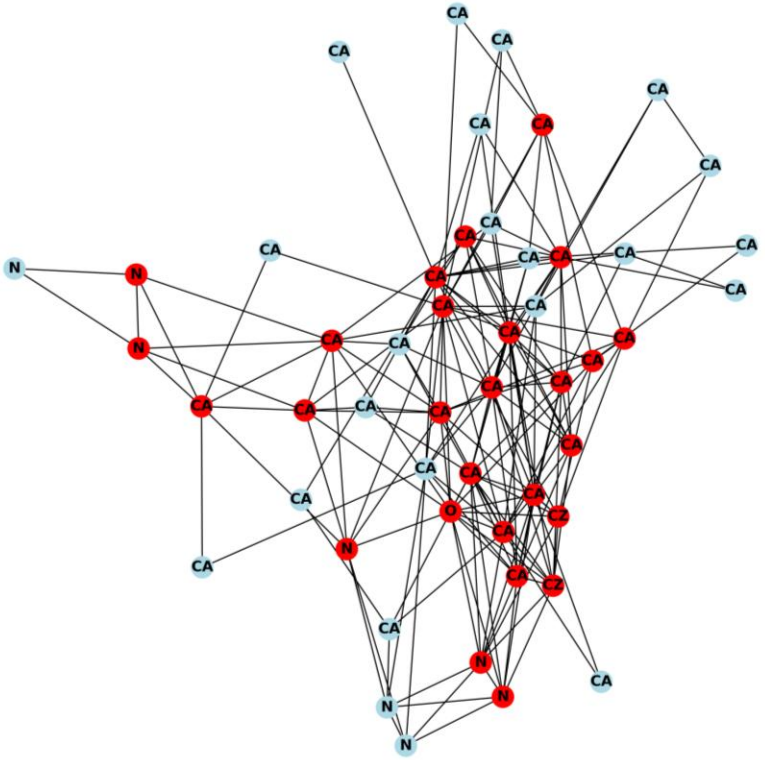


Protein-ligand complex:
Complicated biochemical phenomenon

Protein-ligand interaction graph:
Simplified graph representation

Protein-ligand interaction graphs

- Protein-ligand interaction graphs are formed by:
- **Protein pseudo-atoms**: indicate **positions** of **protein residues**
- **Ligand pseudo-atoms**: indicate **ligand atoms** or **intermediate positions** between interacting atoms
- **Interacting pseudo-atoms** are connected via an **edge**



Protein-ligand interaction graphs

- Popular claim**:** GNNs **learn interactions** and energetic effects just by **simple graph representations**
- Are we sure? We need to **explore what GNNs learn** when trained on interaction graphs
- Goal: Determine their applicability in drug discovery

Real **learning** or simple **memorization?**

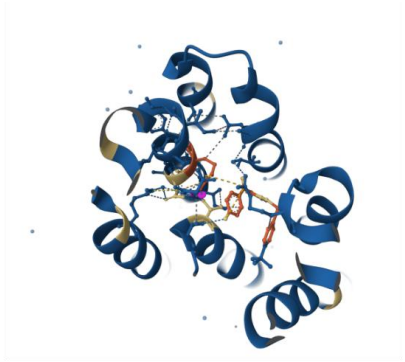
Explaining the predictions

- Goal: understand **which parts** of the interaction graph (**ligand, protein,** or **interaction**) play a major role in the prediction of the potency value
- **Real learning** of interactions would focus on **interaction edges**
- Our approach[3]:

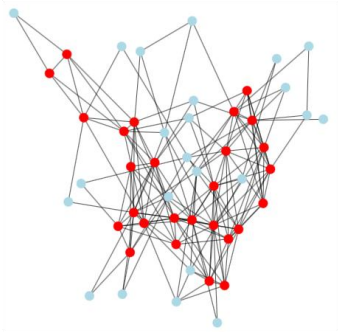
1. Define different **GNN models**
2. Train on **benchmark protein-ligand interaction** graphs
3. **Explain** the predictions **using EdgeSHAPer**

[3] Mastropietro, Andrea, Giuseppe Pasculli, and Jürgen Bajorath. "Learning characteristics of graph neural networks predicting protein–ligand affinities." Nature Machine Intelligence 5.12 (2023): 1427-1436

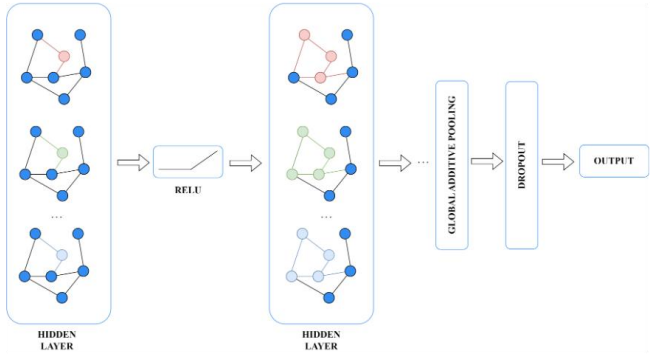
Explanation workflow



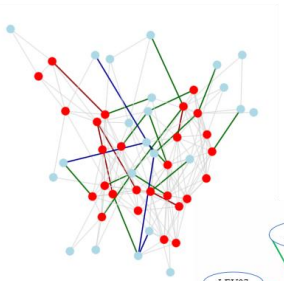
Protein–ligand complex

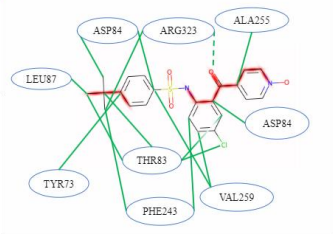


Interaction graph

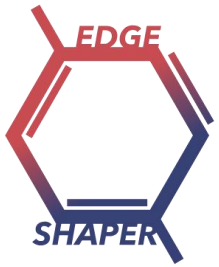


Graph neural network





Important edges



Shapley value-based explainer

PREDICTED POTENCY

TRAINED GNN MODEL

Explaining protein-ligand affinity predictions

- The GNNs reported errors **lower than state-of-the-art** for protein-ligand interaction affinity prediction

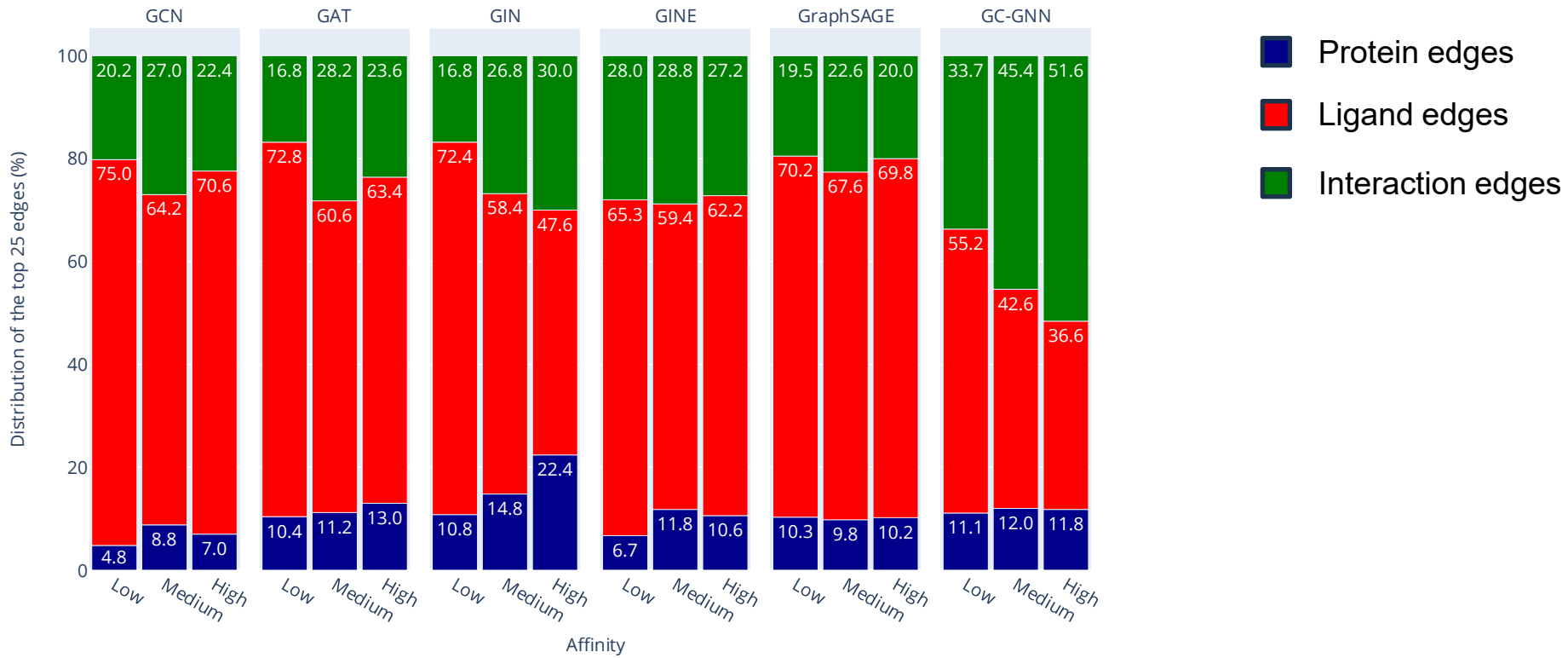
- Accurate models are the **basis for explanation**

Reference literature value

Method	Core set RMSE	Hold-out set RMSE
MPNN	1.605	1.563
GCN	1.397	1.218
GAT	1.321	1.166
GIN	1.318	1.290
GINE	1.398	1.327
GraphSAGE	1.277	1.173
GC-GNN	1.329	1.280

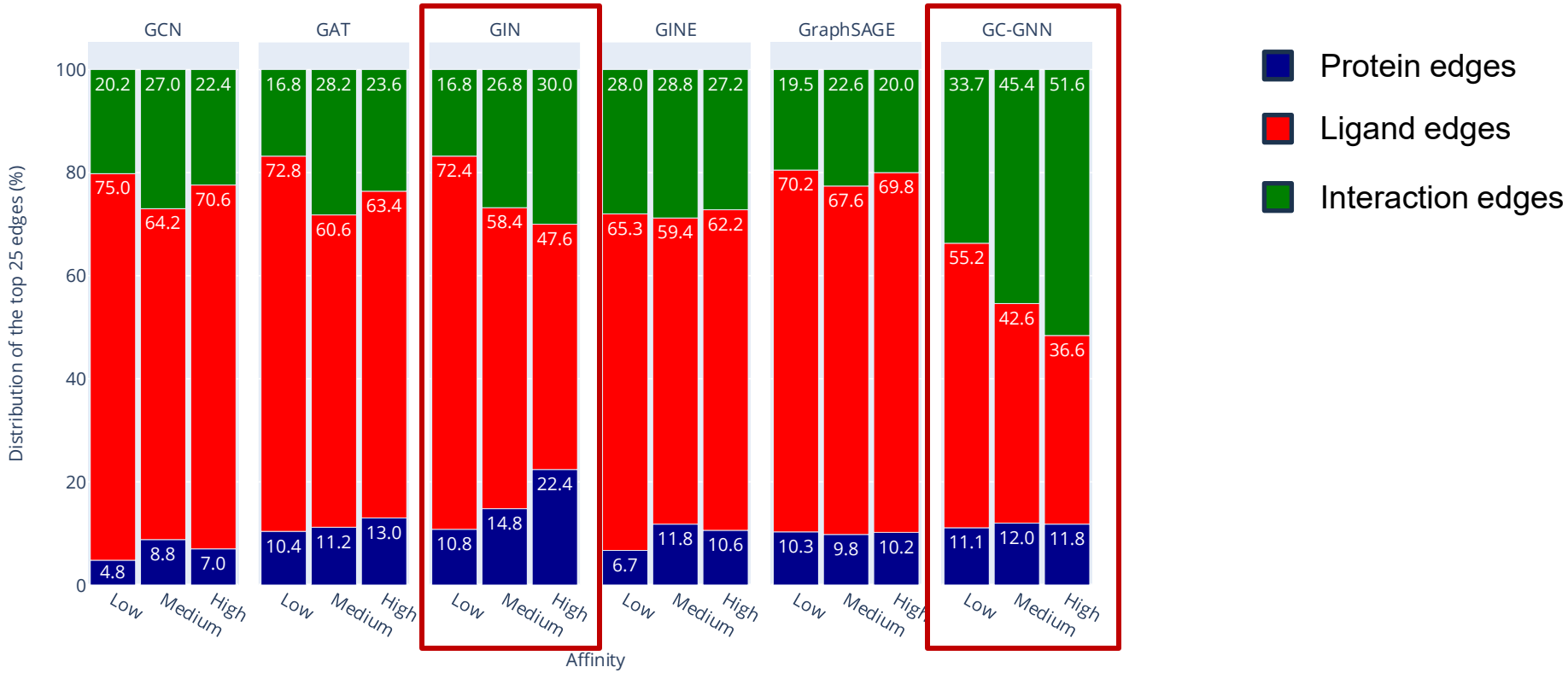
Explaining protein-ligand affinity predictions

- To better study the **learning characteristics** of GNNs, we analyzed the contribution of top edges at different **affinities sub-ranges** (**low**, **medium**, **high affinity**)



Explaining protein-ligand affinity predictions

- To better study the **learning characteristics** of GNNs, we analyzed the contribution of top edges at different **affinities sub-ranges** (**low**, **medium**, **high affinity**)



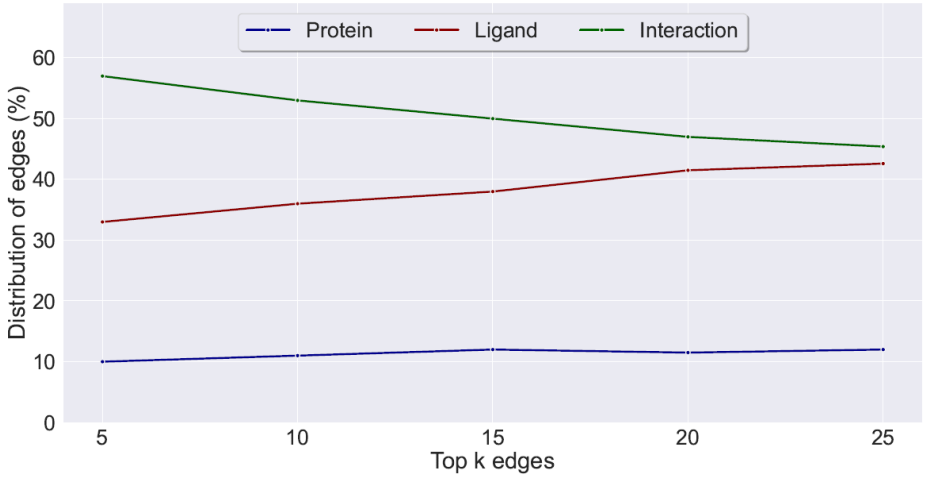
Explaining protein-ligand affinity predictions

- Overall, **ligand edges** drive the predictions, especially for **low-affinity** interactions
  - In line with previous studies[4]
  - **Memorization effect** of **similar ligand** structures in data
  - **Structurally similar ligands** have **similar affinity** with the same (or different) proteins
- However, **interaction edges** are **increasingly prioritized** when predicting potencies of **highly interacting compounds**
- **Highly interacting compounds: interest in drug design**

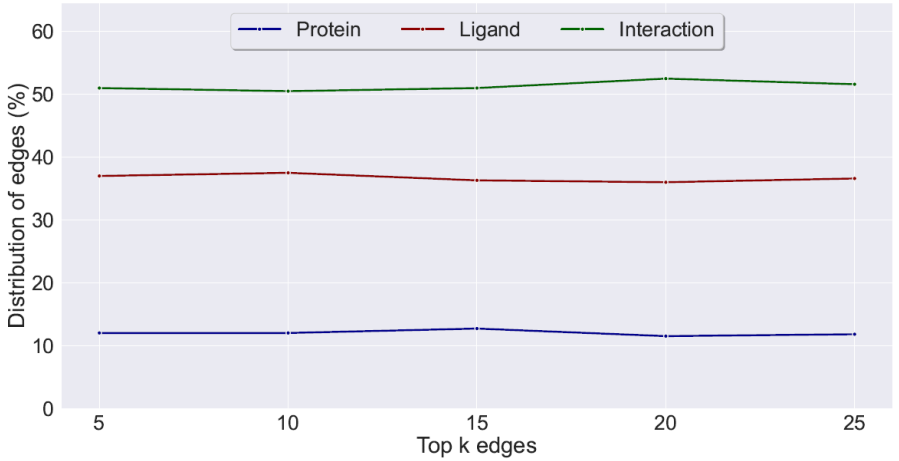
[4] Volkov, Mikhail, et al. "On the frustration to predict binding affinities from protein–ligand structures with deep neural networks." *Journal of medicinal chemistry* 65.11 (2022): 7946-7958.

Explaining protein-ligand affinity predictions

Low affinity

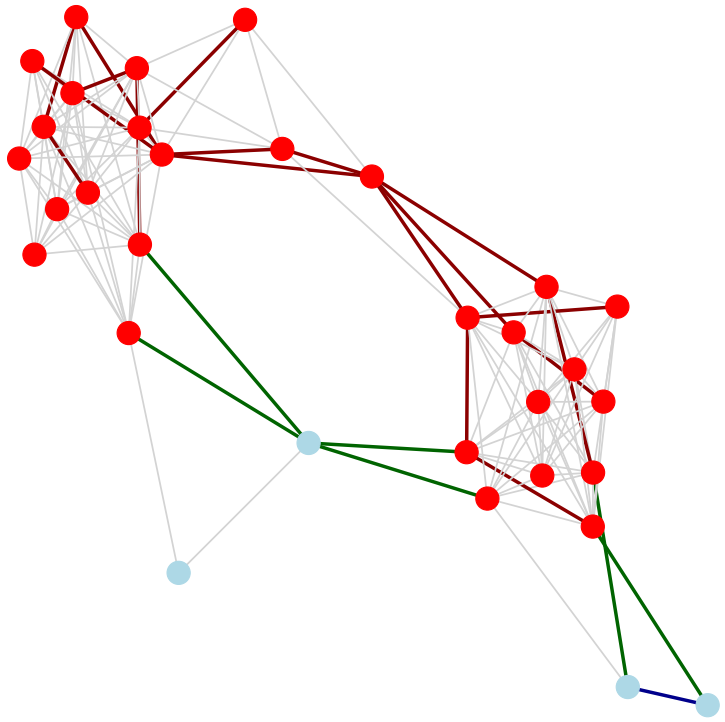


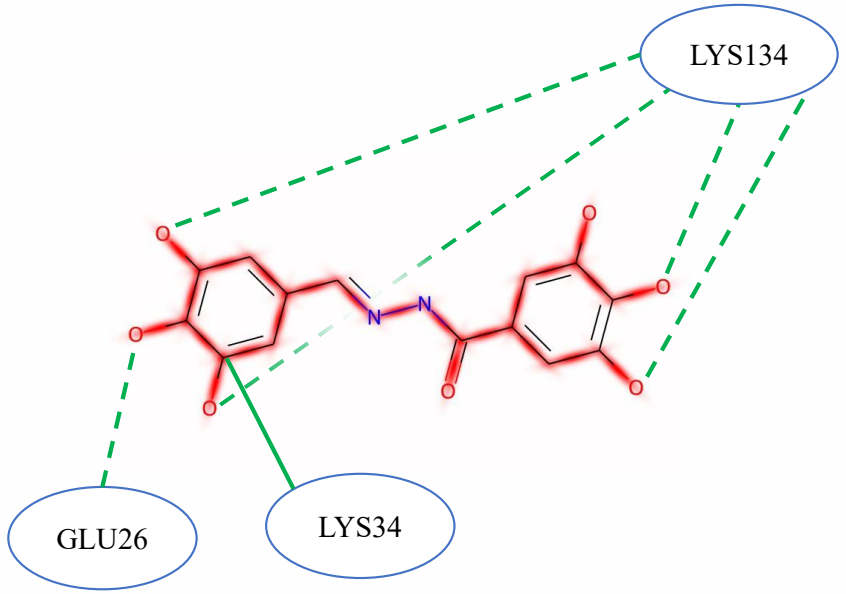
Medium affinity



High affinity

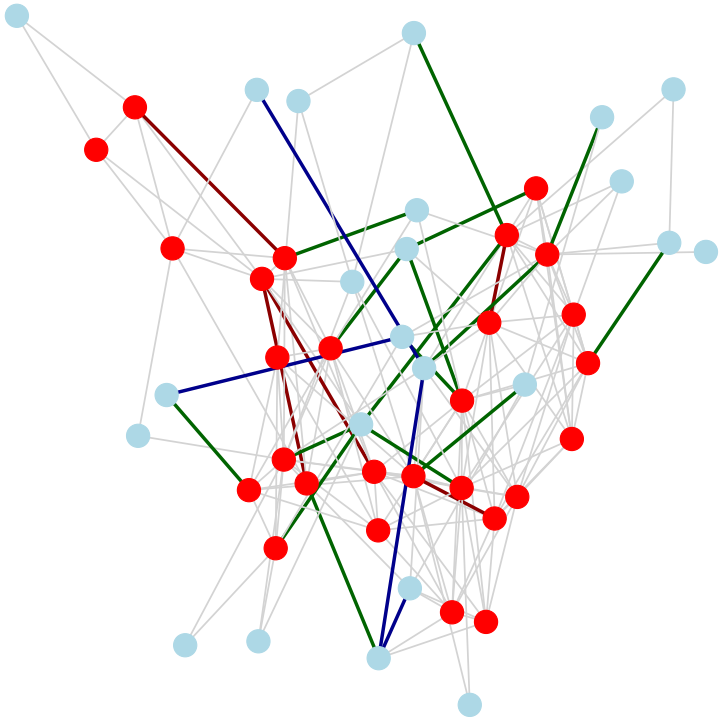
Mapping of important edges

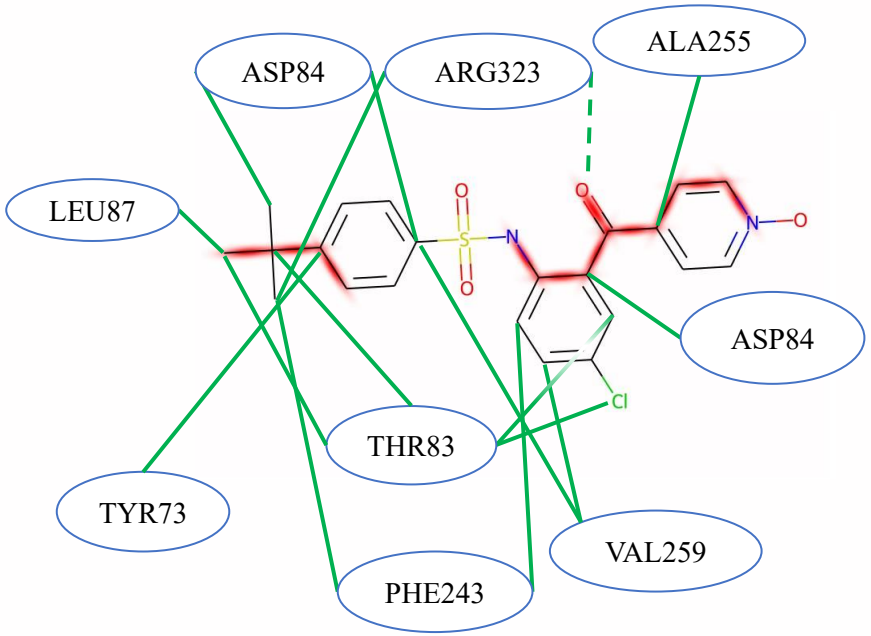




Ligand edge memorization (**red**) drives the prediction of **low-affinity compounds**

Mapping of important edges





Learning of **interaction edges** (**green**) drives the prediction of **high-affinity compounds**

Take-home message

- We run a systematic XAI analysis to reveal if protein-ligand interactions can be learned by GNNs using **simple graph representations of complex biochemical phenomena**

- GNN predictions **do not exclusively depend** on learning protein–ligand interactions:

  - Predominant **ligand memorization**
  - **Interaction** information is leveraged and **learned** to predict **highly potent compounds** (of interest in drug design)

- **Question:** Can we trust GNN predictions in chemoinformatics?

- **Answer:** It depends!

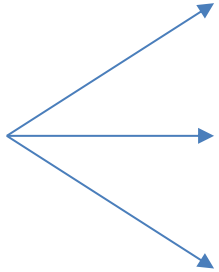
Take-home message

Mind the data

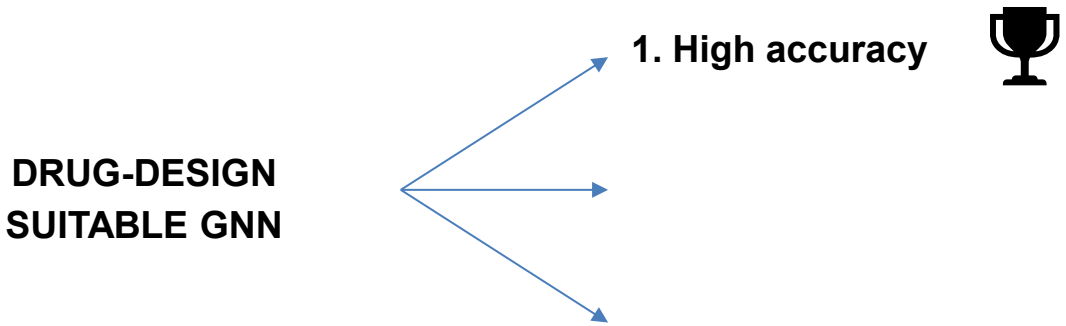
- Interaction information **can be learned** by GNNs
- They need to use graphs that **emphasize interaction patterns,** limiting ligand and protein information
- **Avoid memorization** and **favor learning,** rendering GNNs **suitable for drug design**

Take-home message

DRUG-DESIGN
SUITABLE GNN

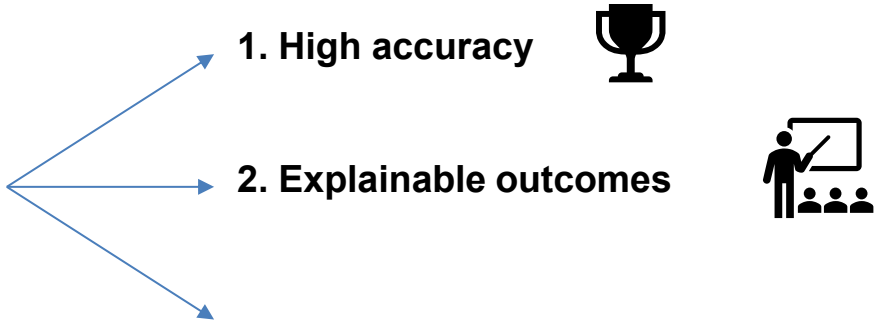


Take-home message



Take-home message

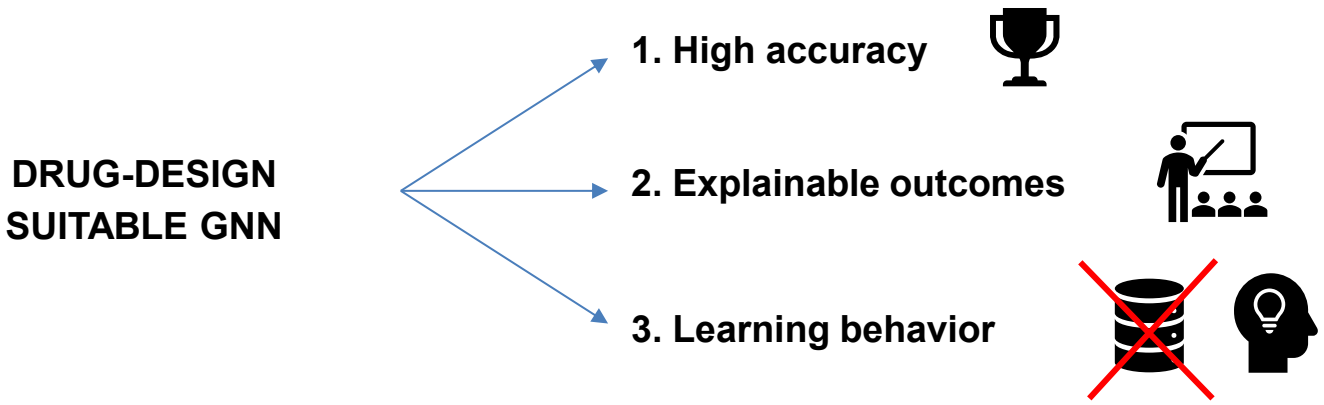
DRUG-DESIGN SUITABLE GNN



1. High accuracy

2. Explainable outcomes

Take-home message



Explaining support vector machine models

Beyond Shapley value approximation

Beyond Shapley value approximation

- EdgeSHAPer and methods in the literature have shown the **efficacy of approximation** of Shapley values
- Question: is Shapley value **approximation always enough**?
- Answer: not always
- **Approximation not effective** when explaining support vector machine (SVM) models for compound activity predictions

Beyond Shapley value approximation

- SVMs **valid and extensively used tools** for chemoinformatics
- They match the accuracy of complex neural network models
- Working simpler models can be a steppingstone for **extension to deep learning**

Make a step back from neural networks

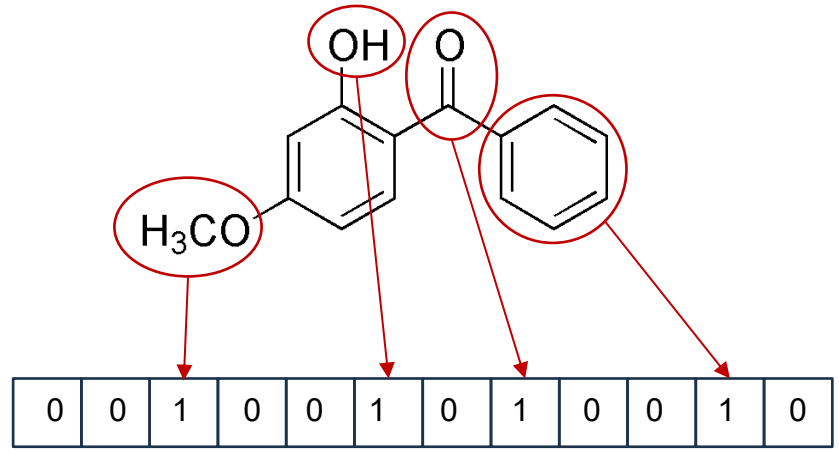
Make a forward: toward **exact Shapley value computation**

Shapley Value-Expressed Radial Basis Function

- Consider SVM with radial basis function kernel (RBF):

$$K(\boldsymbol{x}, \boldsymbol{x}') = e^{-\frac{d(\boldsymbol{x},\boldsymbol{x}')^2}{2\sigma^2}}$$

- Relies on the Euclidean distance



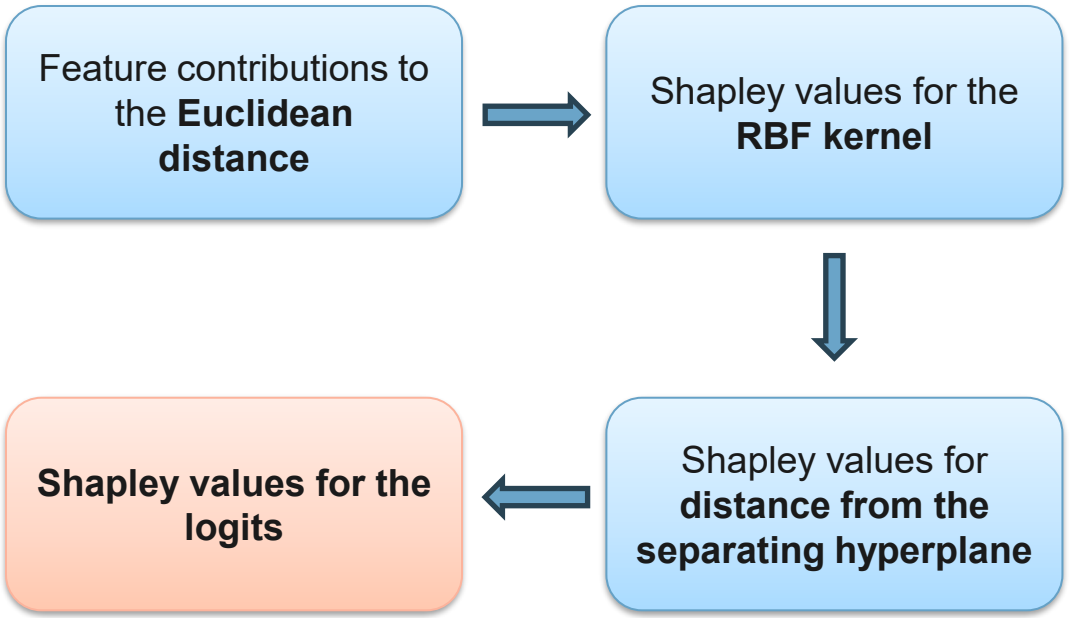
- **Binary molecular fingerprint** (predefined feature descriptors for molecules)

- We propose **SVERAD**[5] (Shapley Value-Expressed Radial Basis Function)
  - Perform **exact Shapley value computation** in quadratic time

[5] Andrea Mastropietro, Christian Feldmann, Jürgen Bajorath, Calculation of exact Shapley values for explaining support vector machine models using the radial basis function kernel, *Scientific Reports*, 13,19561 2023

Overview of SVERAD

- To express feature contribution as Shapley values for SVM:



Feature contribution to Euclidean distance

- Euclidean distance: $d(\boldsymbol{x}, \boldsymbol{x}') = ||\boldsymbol{x} - \boldsymbol{x}'|| = \sqrt{\sum_i (x_i - x_i')^2}$
- Given two feature vectors, our strategy uses only the **number of intersecting** (common) N_i and **symmetric difference features** (present in one instance only) N_d **to compute Shapley values**
- Intersecting features **do not increase the distance:** $(x_i - x_i') = 0$ if $x_i = x_i'$
- Symmetric difference features increase $d(\boldsymbol{x}, \boldsymbol{x}')^2$ by $\Delta_d = (1-0)^2 = (0-1)^2 = 1$

Feature contribution to Euclidean distance

- Euclidean distance: $d(\boldsymbol{x}, \boldsymbol{x}') = ||\boldsymbol{x} - \boldsymbol{x}'|| = \sqrt{\sum_i (x_i - x_i')^2}$
- Given two feature vectors, our strategy uses only the **number of intersecting** (common) N_i and **symmetric difference features** (present in one instance only) N_d **to compute Shapley values**
- Intersecting features **do not increase the distance:** $(x_i - x_i') = 0$ if $x_i = x_i'$
- Symmetric difference features increase $d(\boldsymbol{x}, \boldsymbol{x}')^2$ by $\Delta_d = (1-0)^2 = (0-1)^2 = 1$

$$d(\boldsymbol{x}, \boldsymbol{x}')^2 = N_d \Rightarrow K(\boldsymbol{x}, \boldsymbol{x}') = e^{-\frac{N_d}{2\sigma^2}}$$

Number of symmetric difference features

Fast kernel calculation

Shapley values for the RBF kernel

- We need to compute **change in kernel value** when a **feature is added to a coalition**
- Adding intersecting feature:
  - No effect on the kernel: $\Delta v_{f_+}(N_i, N_d) = e^{-\frac{N_d}{2\sigma^2}} - e^{-\frac{N_d}{2\sigma^2}} = 0$
  - Exception for addition to empty coalition ($v(\emptyset)$=0): $\Delta v_{f_+}(0,0) = 1$
- Adding symmetric difference feature: $\Delta v_{f_-}(N_i, N_d) = e^{-\frac{N_d+1}{2\sigma^2}} - e^{-\frac{N_d}{2\sigma^2}}$
- Features **not present in either instance** (missing features) **do not contribute**
  - In line with Shapley value formalism

Shapley values for the RBF kernel

- Shapley values can be computed as: $\phi_f = \Delta v_f \cdot C_f \cdot \binom{I+D}{1, N_i + N_d, I + D - N_i - N_d - 1}^{-1}$

- Shapley value for **intersecting feature**:

$$\phi_{f_+} = 1 \cdot 1 \cdot \binom{I+D}{1, N_i + N_d, I + D - N_i - N_d - 1}^{-1}$$

- Shapley value for **symmetric difference feature**:

$$\phi_{f_-} = \sum_{N_i=0}^{I} \sum_{N_d=0}^{D-1} \Delta v_{f_-}(N_i, N_d) \cdot C_{f_-}(N_i, N_d) \cdot \binom{I+D}{1, N_i + N_d, I + D - N_i - N_d - 1}^{-1}$$

Sum of Shapley values = kernel value

Shapley values for distance from hyperplane

- Distance from the separating hyperplane: $dist(\boldsymbol{x}) = b + \sum_{n=0}^{N_v-1} y_n w_n K(\boldsymbol{x}, \boldsymbol{V}_n)$

- Express **kernel** as **sum of Shapley values**: $b + \sum_{n=0}^{N_v-1} y_n w_n \sum_{f=0}^{|F|-1} \phi_{f,n} = b + \sum_{f=0}^{|F|-1} \sum_{n=0}^{N_v-1} y_n w_n \phi_{f,n}$

- Shapley values for the **distance from the** separating **hyperplane**:

$$\phi_f = \sum_{n=0}^{N_v-1} y_n w_n \phi_{f,n} \qquad \phi_b = b$$

Shapley values for SVM predictions

- We consider Platt scaling: $logit\big(p(\boldsymbol{x})\big) = \log\left(\frac{1}{e^{A\cdot dist(\boldsymbol{x})+\boldsymbol{B}}}\right) = -A \cdot dist(\boldsymbol{x}) - B$

- Express $dist(\boldsymbol{x})$ as sum of Shapley values

- Obtain **Shapley values for the logits** as linear transformation:

$$logit\big(p(\boldsymbol{x})\big) = -A \cdot \left(\phi_b + \sum_{f=0}^{|F|-1} \phi_f\right) - B = \boxed{-(A \cdot \phi_b + B)} - \sum_{f=0}^{|F|-1} \boxed{A \cdot \phi_f}$$

Quadratic time in the number of input features (worst case ⟶ all features in the symmetric difference)

Application to compound activity prediction

- Task: distinguish active from inactive compounds (adenosine receptor A3 ligands)
- SVM: accuracy of 0.93
- Compare different **exact** Shapley values and SHAP **approximation** approaches

Kernel

	Exact Shapley values	SVERAD	SHAP
Exact Shapley values	1.0 ± 0.0	1.0 ± 0.0	0.72 ± 0.43
SVERAD	1.0 ± 0.0	1.0 ± 0.0	0.72 ± 0.43
SHAP	0.72 ± 0.43	0.72 ± 0.43	1.0 ± 0.0

SVM

	SVM - SVERAD	SVM - KernelSHAP	RF - TreeSHAP	RF - KernelSHAP
SVM - SVERAD	1.000	0.120	-0.040	-0.010
SVM - KernelSHAP	0.120	1.000	0.758	0.750
RF - TreeSHAP	-0.040	0.758	1.000	0.994
RF – KernelSHAP	-0.010	0.750	0.994	1.000

Pearson's *r* correlation coefficient

Need for exact Shapley value calculation for feature contributions

Take-home message

- Approximation of Shapley values is **not always enough**
- SVERAD: **exact and efficient** Shapley value computation for SVMs
- Neural networks do not live alone: SVMs valuable tools in chemoinformatics
- They **match the accuracy** of more advanced deep learning models

Conclusions

- Deep learning is widely used in life science applications with outstanding results
- We need **explainable** or **interpretable** models
- We need to understand the behavior of a model **before deploying** it for tasks like drug design
- In life science and medical applications, it is not enough to know **what** is predicted; we need to know **why** it is predicted

Conclusions

- However, model predictions **should not be overinterpreted as evidence of chemical understanding**
- Models memorize **statistical patterns in training data**
- This is not something “wrong”, we just need to be careful on how we use and interpret findings
- Research directions are wide open **toward more representative data** and **expert knowledge-based models**

b-it LSI
Life
Science
Informatics

LAMARR
INSTITUTE FOR
MACHINE LEARNING
AND ARTIFICIAL
INTELLIGENCE

LiMES
Life & Medical Sciences Institute

Thank you!

Andrea Mastropietro, Giuseppe Pasculli, Christian Feldmann, Raquel Rodríguez-Pérez, Jürgen Bajorath, **EdgeSHAPer: Bond-centric Shapley value-based explanation method for graph neural networks**. *iScience*, 25(10). 2022

Andrea Mastropietro, Giuseppe Pasculli, Jürgen Bajorath, **Learning characteristics of graph neural networks predicting protein–ligand affinities,** *Nature Machine Intelligence*, 5(12), 1427—1436, 2023

Andrea Mastropietro, Christian Feldmann, Jürgen Bajorath, **Calculation of exact Shapley values for explaining support vector machine models using the radial basis function kernel,** *Scientific Reports*, 13,19561 2023

mastro.me
mastropietro@bit.uni-bonn.de

UNIVERSITÄT BONN

Nara Institute of Science and Technology
NAIST®