On Logic-Based Explanations for Graph Neural Networks:

From Post-hoc Explanations to Self-Explainable Models

Alessio Ragno

INSTITUT NATIONAL
DES SCIENCES
APPLIQUÉES
LYON

INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

Who am I?

Postdoc @ INSA Lyon:

- Jan 2025 (Nov 2024) - present
- Supervisors: Prof. Celine Robardet, Prof. Marc Plantevit
- WAIT4 Project:
    - *Caractérisation des activités animales et de leurs interactions sociales à l'aide de réseaux de neurones sur graphes interprétables.*

PhD in Artificial Intelligence @ Sapienza University of Rome:

- Nov 2021 - Jan 2025
- Topology-based Explanations for Neural Networks

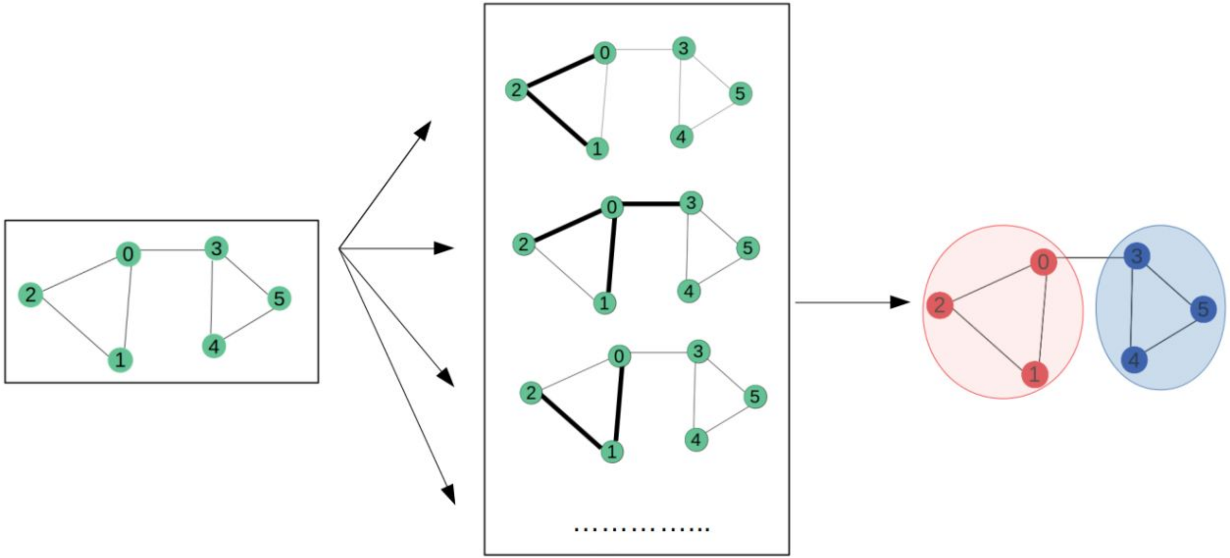
Master’s Degree in Artificial Intelligence and Robotics @ Sapienza University of Rome:

- Sep 2019 - May 2021

INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

Table of Contents

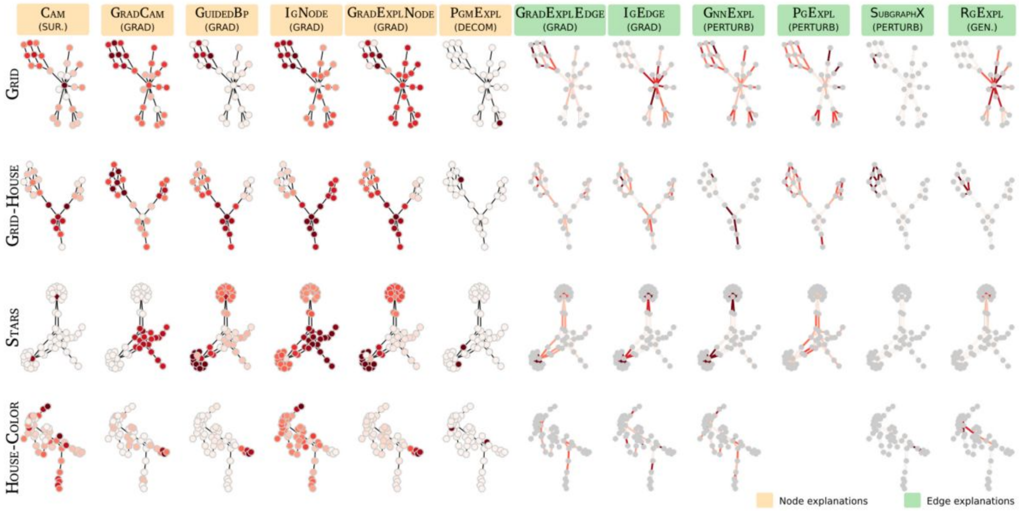
Graph Neural Networks



XAI for GNNs

Post-hoc:

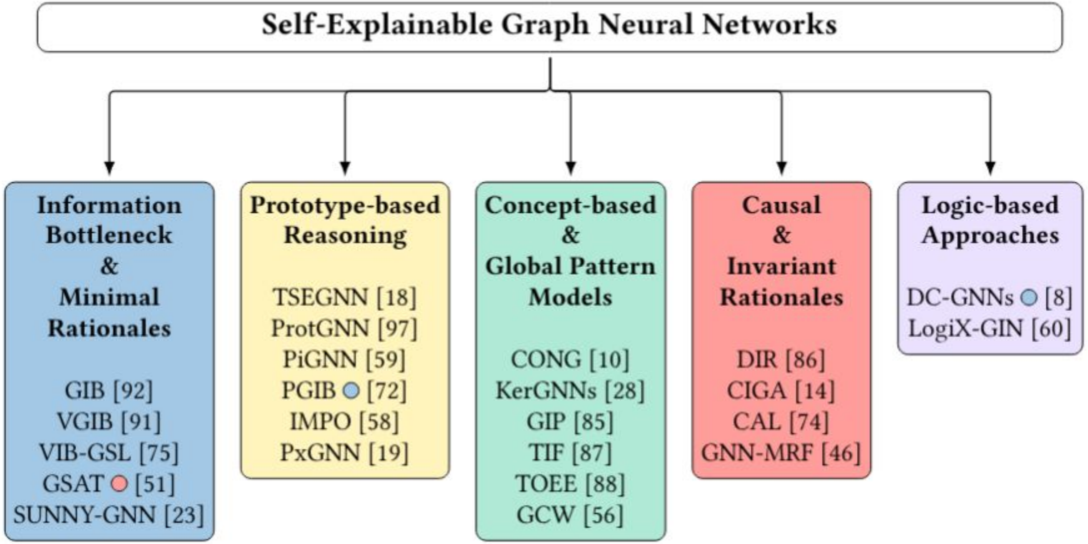
- gradient-based approaches
  (Li et al. 2021, Kasanishi 2021)
- perturbation-based approaches
  (Ying et al. 2019, Luo et al. 2020)
- optimization-based approaches
  (Yuan et al. 2021)
- pattern-based
  (Veyrin-Forrer et al. 2022, Kamal et al. 2023)



Explaining the Explainers in Graph Neural Networks: a Comparative Study (Longa et al., 2025, ACM Comp. Surv.)

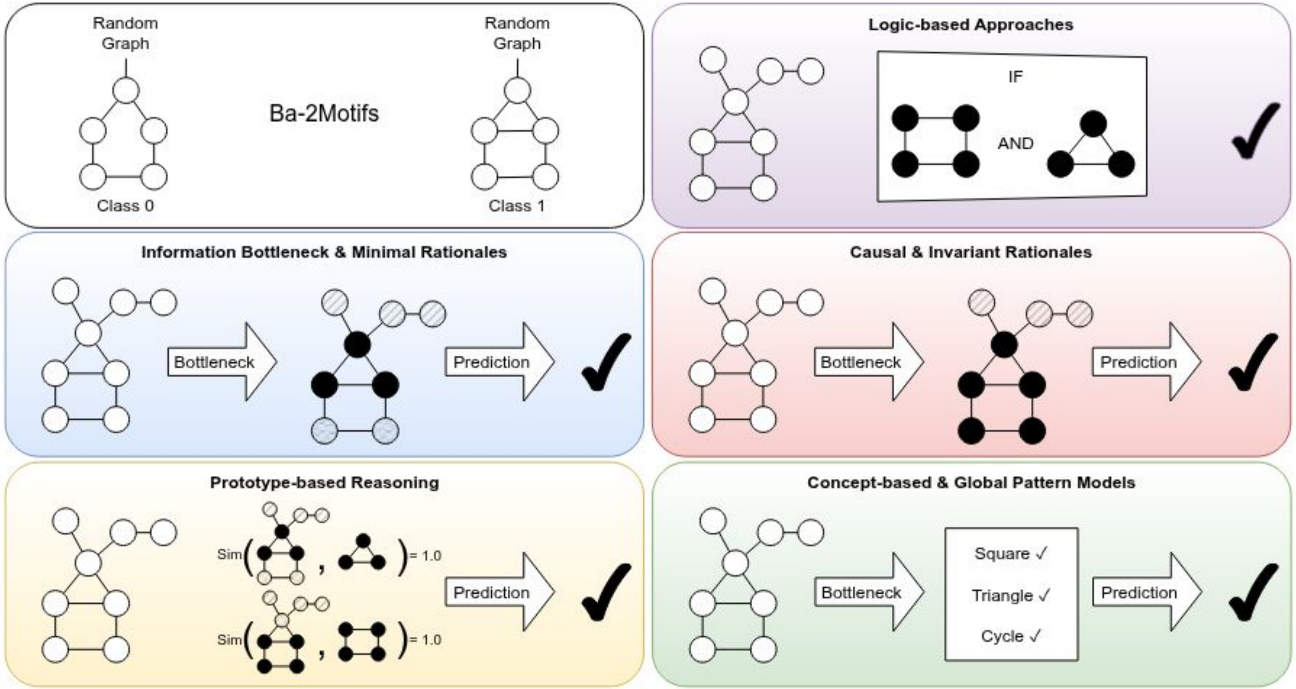
INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

Self-Explainable GNNs



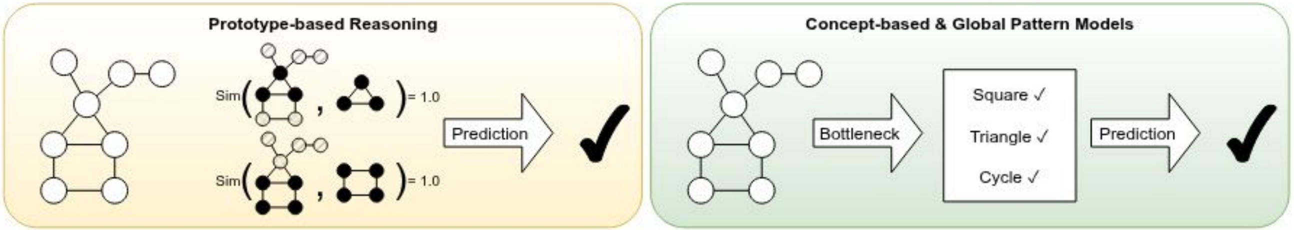
(Hopefully there will be a survey before too long xD)

Self-Explainable GNNs



Self-Explainable GNNs

In the beginning… there was nothing.

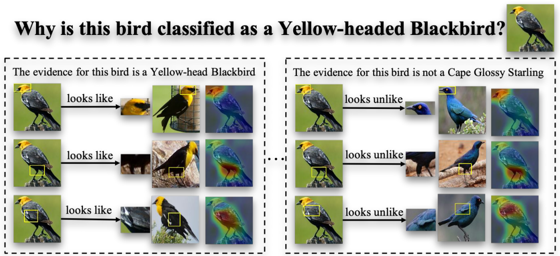


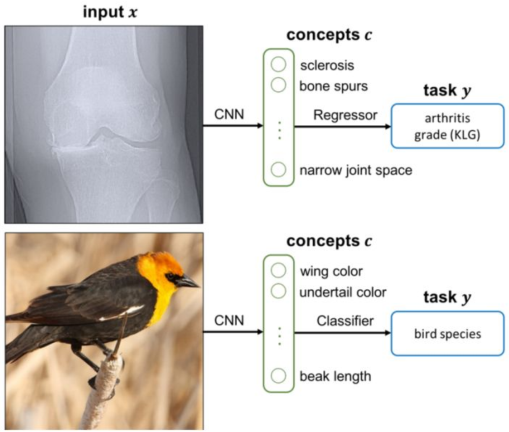
Self-Explainable Neural Networks

Explainable by-design models developed with interpretable reasoning process.

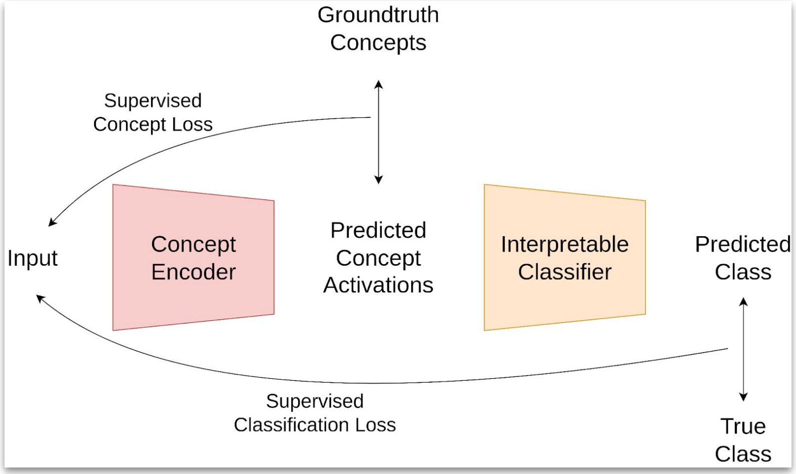
To achieve this result, researchers have developed:

- Prototype-based Neural Networks (C. Chen, 2018; Wang, 2021; S.F. Stefenon, 2022)
- Concept-based Neural Networks (Z. Chen, 2020; P.W. Koh, 2020)

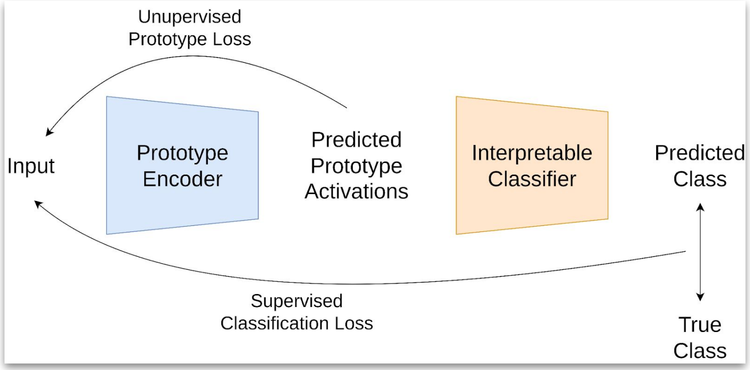




Self-Explainable Neural Networks



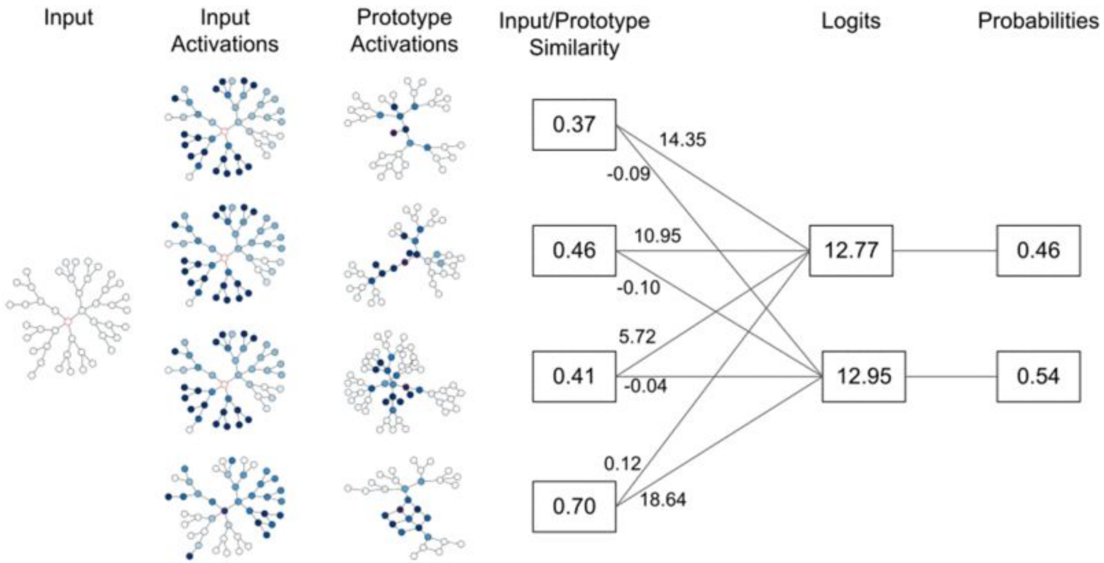
Concept-based



Prototype-based

INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

Self-Explainable GNNs

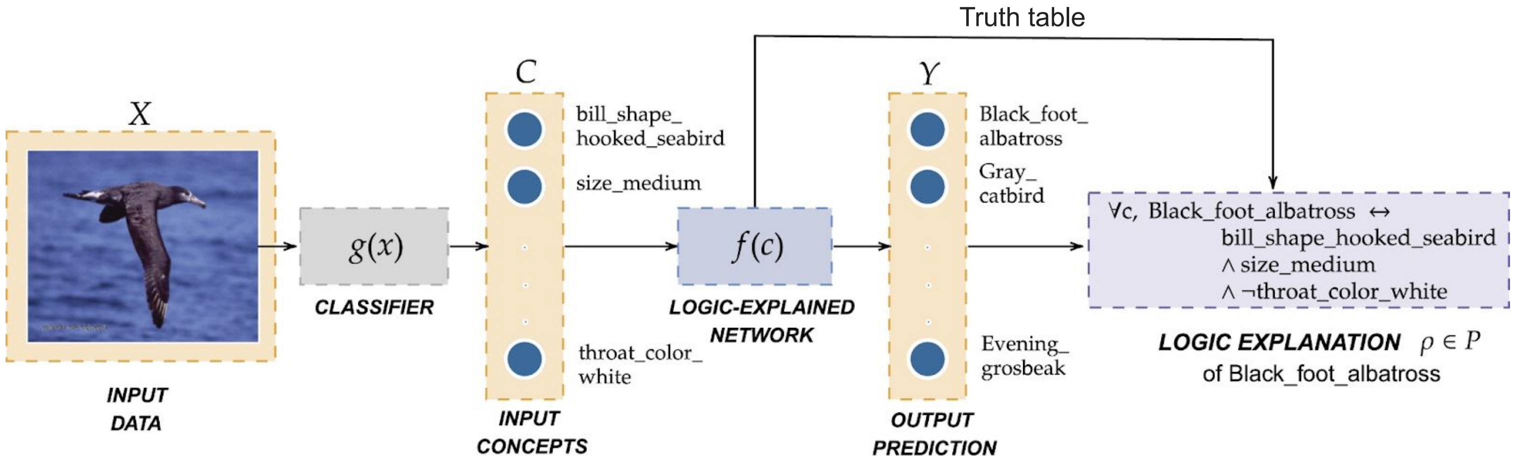


Prototype-based Interpretable Graph Neural Networks (Ragno et al., 2022, IEEE TAI)

INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

Table of Contents

Logic-Explained Networks



Logic Explained Networks (Ciravegna et al., 2023, Artificial Intelligence)

Entropy-based logic explanations of neural networks (Barbiero et al., 2022, AAAI)

Motivating Experiment: Logic Induction

Ground Truth Rule:

$$y = 1 \iff (x_1 > 0.5 \wedge x_2 > 0.2) \vee (x_1 > 0.5 \wedge x_3 > 0.7)$$

Model	Accuracy (%)	Rules Accuracy (%)	Fidelity (%)	Complexity	Best Model's Rule
ENTROPY	94.68 ± 0.33	59.48 ± 0.56	74.62 ± 8.25	1.00 ± 0.00	$x_1 > 0.50$
ψ	89.98 ± 0.36	65.45 ± 0.60	64.74 ± 3.26	1.73 ± 0.13	$x_1 > 0.50$
RELU	$\mathbf{99.12 \pm 1.00}$	93.12 ± 0.14	92.92 ± 0.29	1.00 ± 0.00	$x_1 > 0.50$
μ	$\mathbf{98.38 \pm 0.41}$	92.49 ± 0.23	92.11 ± 0.34	2.93 ± 0.48	$x_1 > 0.50$
DR-NET	96.72 ± 0.37	92.70 ± 0.35	90.85 ± 0.53	133.13 ± 2.07	Explanation too long

Transparent Explainable Logic Layers

$$y = \sigma(w_1x_1 + w_2x_2 + w_3x_3 + b)$$

$$y_d = 1 \quad \leftrightarrow \quad y > 0.5 \quad \leftrightarrow \quad w_1x_1 + w_2x_2 + w_3x_3 + b > 0$$

Transparent Explainable Logic Layers

x_i is binary
$w_i >= 0$

$$y = \sigma(w_1x_1 + w_2x_2 + w_3x_3 + b)$$

$$y_d = 1 \quad \leftrightarrow \quad y > 0.5 \quad \leftrightarrow \quad w_1x_1 + w_2x_2 + w_3x_3 + b > 0$$

Transparent Explainable Logic Layers

x_i is binary
$w_i >= 0$

$$y = \sigma(w_1x_1 + w_2x_2 + w_3x_3 + b)$$

$$y_d = 1 \quad \leftrightarrow \quad y > 0.5 \quad \leftrightarrow \quad w_1x_1 + w_2x_2 + w_3x_3 + b > 0$$

$$y = \sigma(5x_1 + 2x_2 + 2x_3 - 2.5)$$

Transparent Explainable Logic Layers

x_i is binary
$w_i >= 0$

$$y = \sigma(w_1x_1 + w_2x_2 + w_3x_3 + b)$$

$$y_d = 1 \quad \leftrightarrow \quad y > 0.5 \quad \leftrightarrow \quad w_1x_1 + w_2x_2 + w_3x_3 + b > 0$$

$$y = \sigma(5x_1 + 2x_2 + 2x_3 - 2.5)$$

es. $x = [1\ 0\ 1] \rightarrow y = \sigma(5 + 0 + 2 - 2.5) > 0.5 \rightarrow y_d = 1$

Transparent Explainable Logic Layers

x_i is binary
$w_i >= 0$

$$y = s(w_1x_1 + w_2x_2 + w_3x_3 + b)$$

$$y_d = 1 \quad \leftrightarrow \quad y > 0.5 \quad \leftrightarrow \quad w_1x_1 + w_2x_2 + w_3x_3 + b > 0$$

$$y = \sigma(5x_1 + 2x_2 + 2x_3 - 2.5)$$

es. $x = [1\ 0\ 1] \rightarrow y = \sigma(5 + 0 + 2 - 2.5) > 0.5 \rightarrow y_d = 1$

es. $x = [0\ 0\ 1] \rightarrow y = \sigma(0 + 0 + 2 - 2.5) < 0.5 \rightarrow y_d = 0$

Transparent Explainable Logic Layers

x_i is binary
$w_i >= 0$

$$y = \sigma(w_1x_1 + w_2x_2 + w_3x_3 + b)$$

$$y_d = 1 \quad \leftrightarrow \quad y > 0.5 \quad \leftrightarrow \quad w_1x_1 + w_2x_2 + w_3x_3 + b > 0$$

$$y = \sigma(5x_1 + 2x_2 + 2x_3 - 2.5)$$

$$y_d = 1 \leftrightarrow \Sigma_{i \in X} w_i > -b \qquad X = \{i: x_i=1\}$$

we need to identify the subsets of features with corresponding weights that sum up to a value greater than -b

Transparent Explainable Logic Layers

x_i is binary
$w_i >= 0$

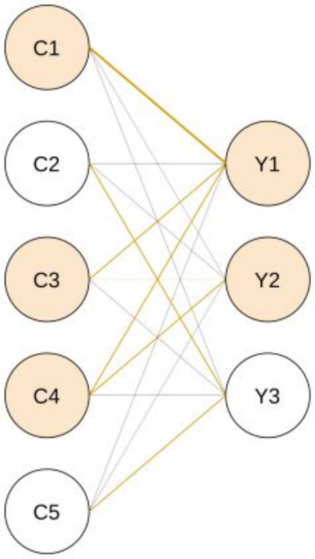
$$y = \sigma(w_1x_1 + w_2x_2 + w_3x_3 + b)$$

$$y_d = 1 \quad \leftrightarrow \quad y > 0.5 \quad \leftrightarrow \quad w_1x_1 + w_2x_2 + w_3x_3 + b > 0$$

$$y = \sigma(5x_1 + 2x_2 + 2x_3 - 2.5)$$

Logic Rule: $x_1 \vee (x_2 \wedge x_3)$

Transparent Explainable Logic Layers



$$\begin{array}{ccc} \mathbf{W} & & \mathbf{-b} \\ \begin{pmatrix} 2 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix} & \overset{?}{>} & \begin{pmatrix} 2.5 \\ 1.5 \\ 0.5 \end{pmatrix} \end{array}$$

Min. Concepts Subsets	Rule
[{C1,C3},{C1,C4}]	(C1 ∧ C3) ∨ (C1 ∧ C4)
[{C3,C4}]	C3 ∧ C4
[{C2},{C5}]	C2 ∨ C5

Transparent Explainable Logic Layers (Ragno et al., 2024, ECAI)

Working with continuous inputs

To work with continuous inputs:

- we add a preprocessing layer with sigmoid activation which thresholds the features

- we introduce a loss over thresholded features that pushes them towards 0 and 1

$$\hat{X} = \sigma(X \odot \exp(W_i) + b_i)$$
$$\text{s.t. } W_i \in \mathbb{R}^I, b_i \in \mathbb{R}^I$$

$$\hat{X} > 0.5 \iff X > -\frac{b_i}{\exp(W_i)}$$

$$\mathcal{L}_E = -\hat{X}log(\hat{X}) - (1 - \hat{X})log(1 - \hat{X})$$

INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

Motivating Experiment: Logic Induction

Ground Truth Rule:

$$y = 1 \iff (x_1 > 0.5 \wedge x_2 > 0.2) \vee (x_1 > 0.5 \wedge x_3 > 0.7)$$

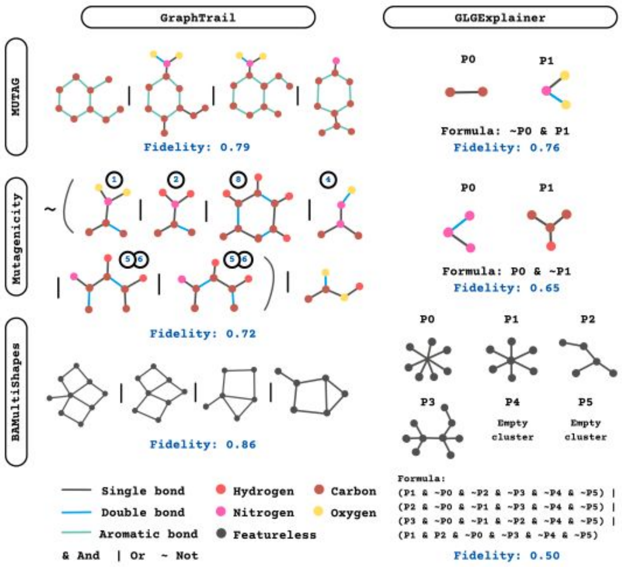
Model	Accuracy (%)	Rules Accuracy (%)	Fidelity (%)	Complexity	Best Model's Rule
TELL	96.84 ± 0.07	$\mathbf{95.70 \pm 0.38}$	$\mathbf{97.79 \pm 0.39}$	3.40 ± 0.17	$(x_1 > 0.48 \wedge x_2 > 0.20) \vee (x_1 > 0.48 \wedge x_3 > 0.70)$
ENTROPY	94.68 ± 0.33	59.48 ± 0.56	74.62 ± 8.25	1.00 ± 0.00	$x_1 > 0.50$
ψ	89.98 ± 0.36	65.45 ± 0.60	64.74 ± 3.26	1.73 ± 0.13	$x_1 > 0.50$
RELU	$\mathbf{99.12 \pm 1.00}$	93.12 ± 0.14	92.92 ± 0.29	1.00 ± 0.00	$x_1 > 0.50$
μ	$\mathbf{98.38 \pm 0.41}$	92.49 ± 0.23	92.11 ± 0.34	2.93 ± 0.48	$x_1 > 0.50$
DR-NET	96.72 ± 0.37	92.70 ± 0.35	90.85 ± 0.53	133.13 ± 2.07	Explanation too long

Classification and Explanation

	Model	Binary Datasets				Concept-Bottleneck			
		MIMIC-II		V-DEM		MNIST E/O		CUB	
		Accuracy (%)	Fidelity (%)	Accuracy (%)	Fidelity (%)	Accuracy (%)	Fidelity (%)	Accuracy (%)	Fidelity
Explainable NN	TELL (ours)	**79.28 ± 0.80**	**96.15 ± 0.62**	**92.50 ± 0.43**	**97.56 ± 0.45**	99.85 ± 0.01	**99.98 ± 0.00**	92.39 ± 0.19	**97.99 ± 0.09**
	ψ	76.57 ± 0.69	58.38 ± 3.60	91.11 ± 0.29	60.30 ± 5.10	99.84 ± 0.01	69.62 ± 2.84	92.22 ± 0.19	75.35 ± 0.92
	RELU	**79.06 ± 0.87**	75.30 ± 2.57	**92.85 ± 0.42**	85.46 ± 3.77	99.85 ± 0.01	93.72 ± 3.88	92.43 ± 0.20	97.45 ± 0.07
	μ	**79.16 ± 0.82**	91.15 ± 1.12	91.82 ± 0.42	**97.49 ± 0.48**	**99.90 ± 0.01**	99.90 ± 0.01	**92.65 ± 0.18**	94.80 ± 0.21
	ENTROPY	**78.89 ± 0.72**	71.47 ± 2.49	90.28 ± 0.60	87.44 ± 5.40	99.84 ± 0.01	41.03 ± 8.24	**92.57 ± 0.19**	97.56 ± 0.08
	DR-NET	74.16 ± 0.60	75.36 ± 1.72	N/A	N/A	85.69 ± 0.06	85.52 ± 0.07	N/A	N/A
ML	DT	76.52 ± 0.75	100.00 ± 0.00	84.91 ± 0.67	100.00 ± 0.00	99.89 ± 0.01	100.00 ± 0.00	80.59 ± 0.79	100.00 ± 0.00
	BRL	77.20 ± 1.10	100.00 ± 0.00	77.20 ± 1.10	100.00 ± 0.00	99.84 ± 0.01	100.00 ± 0.00	91.15 ± 0.45	100.00 ± 0.00

Table of Contents

Logic-based Explanations for GNNs: SOTA



GRAPHTRAIL: Translating GNN Predictions into Human-Interpretable Logical Rules (Ranu et al., 2024, NeurIPS)

Global Explainability of GNNs via Logic Combination of Learned Concepts (Azzolin et al. 2023, ICLR)

Applying TELL paradigm to GNNs

Post-hoc:

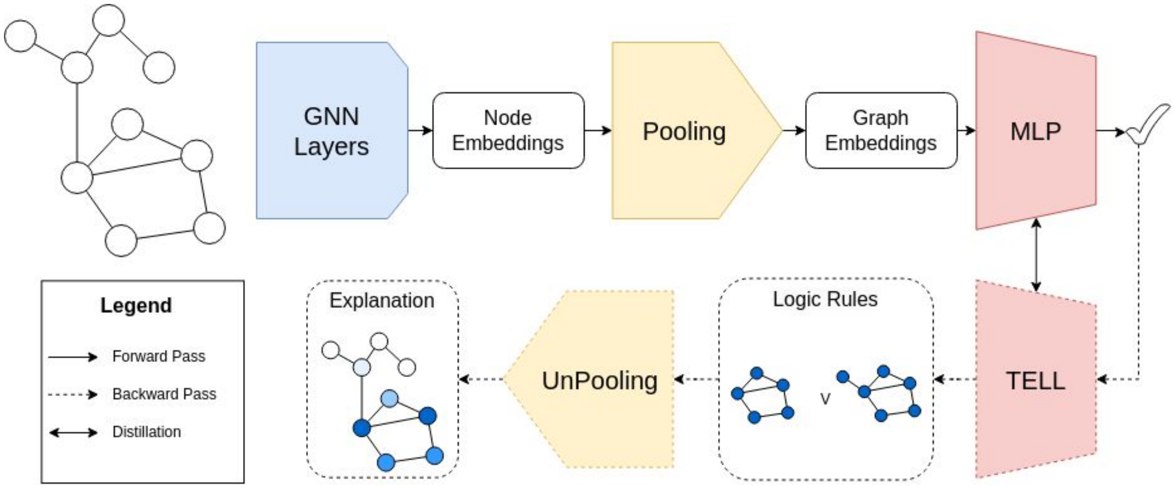
- We use TELL to approximate the GNN prediction and identify the most important nodes

Self-Explainable:

- We create a new GNN Layer upon TELL’s paradigm to create a transparent GNN

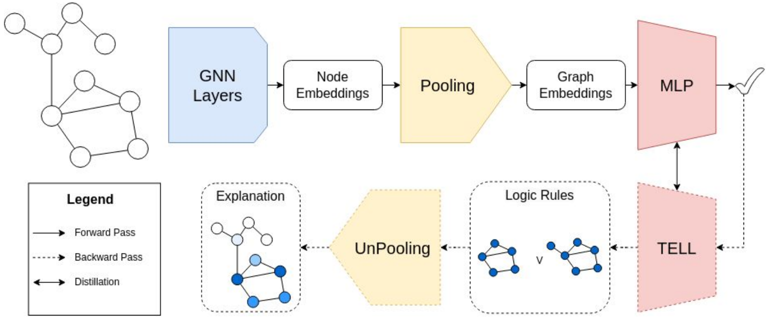
INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

LogiX: Faithful Explanations for GNNs using Logic



Faithful Explanations for GNNs using Logic (Ragno et al., 2025, ECML-PKDD)

LogiX: Rule Extraction



Node Embeddings

	h_1	h_2
1	1	0
2	1	1
3	1	0

Graph Embeddings

sum	h_1	3
	h_2	1
max	h_1	1
	h_2	1
avg	h_1	1
	h_2	.3

TELL Rule

$$\mathrm{sum}(h_1) > 2 \land \max(h_2) > 0.5$$

Rule Contributions

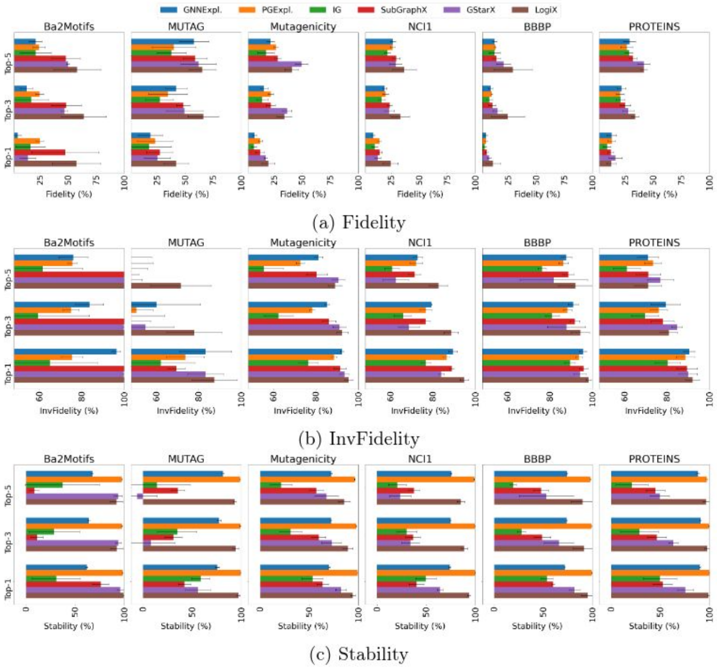
h_1	h_2
.3	0
.3	1
.3	0

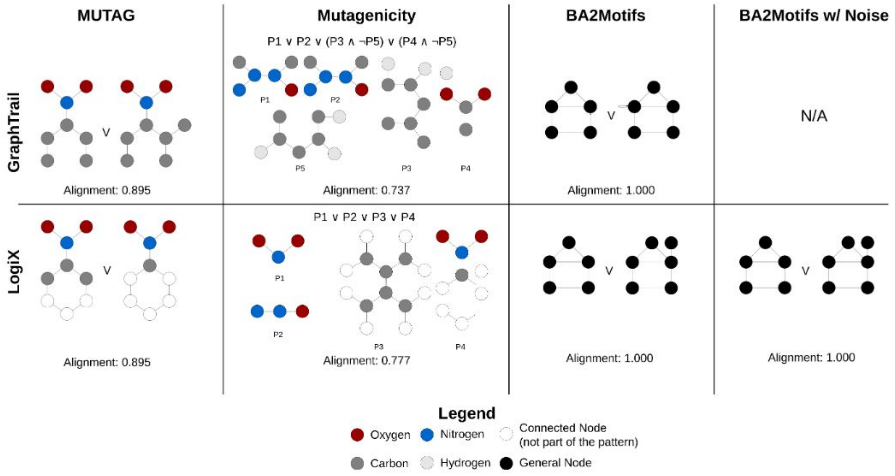
Node Attributions

15%
70%
15%

Faithful Explanations for GNNs using Logic (Ragno et al., 2025, ECML-PKDD)

LogiX: Faithful Explanations for GNNs using Logic





Faithful Explanations for GNNs using Logic (Ragno et al., 2025, ECML-PKDD)

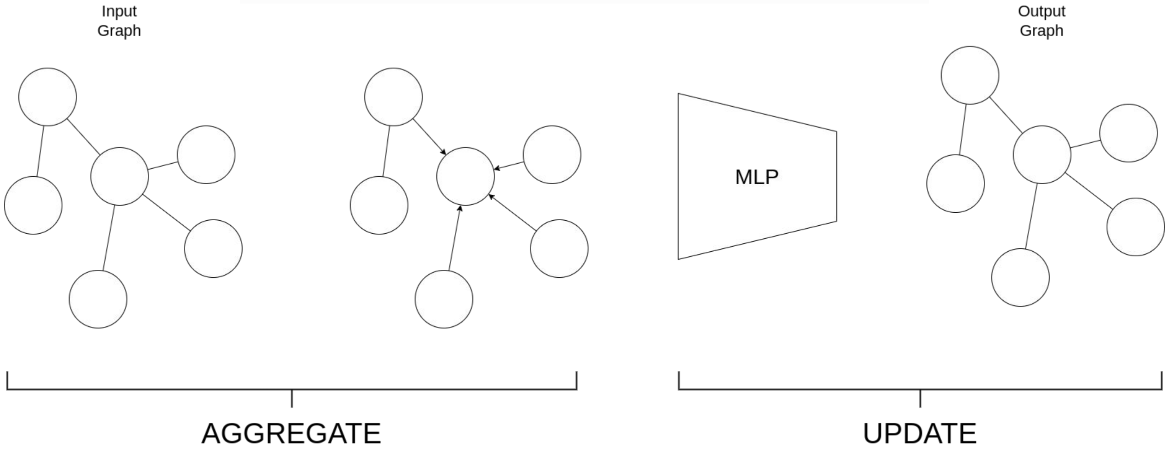
INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

Table of Contents

- Graph Neural Networks & XAI
- Logic-based Explanations with Transparent Explainable Logic Layer (TELL)
- Logic-based Post-hoc Explanations for GNNs
- **Logic-based Self-Explainable GNNs**
- Application of Logic-based Explanations to Drug Discovery

Applying TELL to Graph Neural Networks: GIN

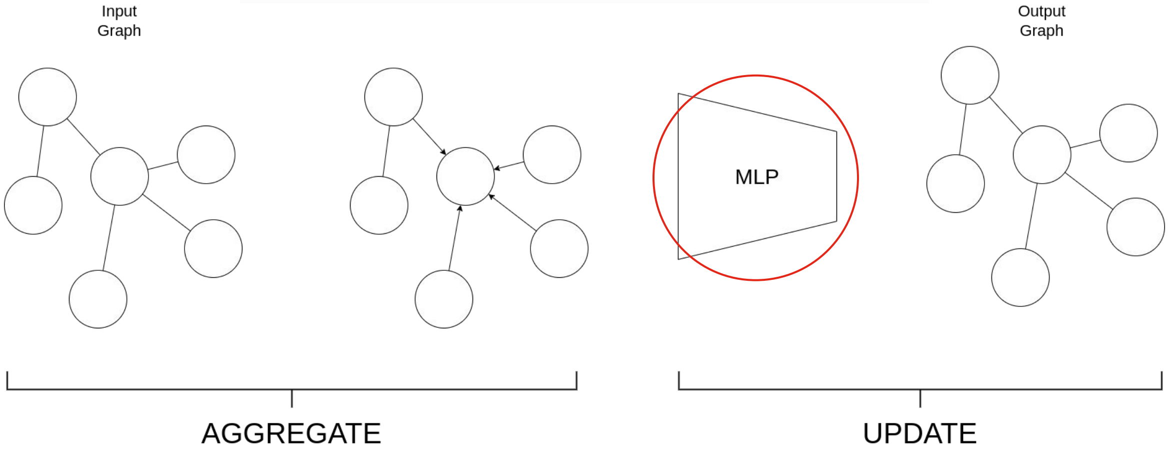
$$\mathbf{x}'_i = h_{\mathbf{\Theta}} \left((1 + \epsilon) \cdot \mathbf{x}_i + \sum_{j \in \mathcal{N}(i)} \mathbf{x}_j \right)$$



On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

Applying TELL to Graph Neural Networks: GIN

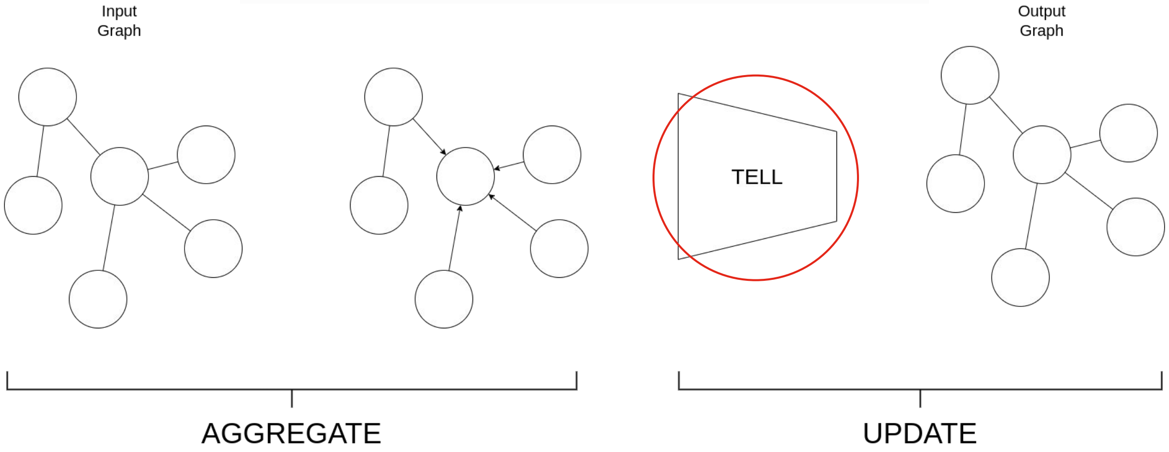
$$\mathbf{x}'_i = h_{\mathbf{\Theta}} \left((1 + \epsilon) \cdot \mathbf{x}_i + \sum_{j \in \mathcal{N}(i)} \mathbf{x}_j \right)$$



On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

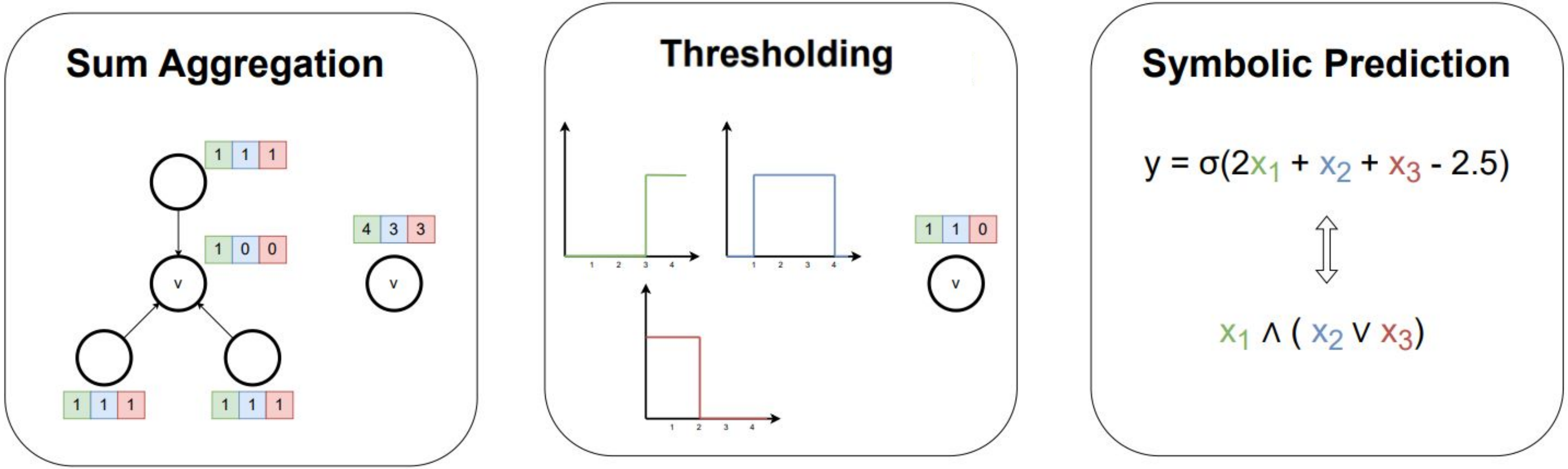
Applying TELL to Graph Neural Networks: GIN

$$\mathbf{x}'_i = h_{\mathbf{\Theta}} \left((1+\epsilon) \cdot \mathbf{x}_i + \sum_{j \in \mathcal{N}(i)} \mathbf{x}_j \right)$$



On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

LogiX-GIN



$$\sum_{v'\in\mathcal{N}(v)\cup\{v\}} \bigcirc \geq 3 \wedge \left(1 \leq \sum_{v'\in\mathcal{N}(v)\cup\{v\}} \bigcirc \leq 4 \vee \sum_{v'\in\mathcal{N}(v)\cup\{v\}} \bigcirc \leq 2\right)$$

On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

Main difficulties

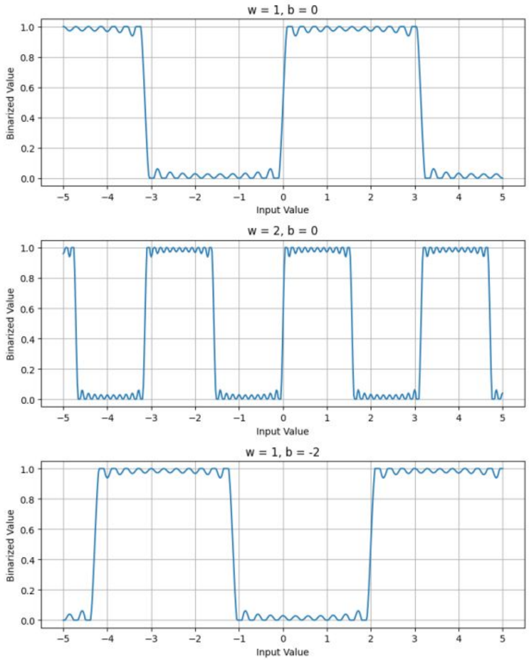
- When aggregating nodes, we apply rules on the number of neighbors which have a determinate pattern. Instead of using a simple threshold, can we learn ranges?

On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

Fourier Thresholding

We want a parametrizable function that learns step-shaped intervals

$$\tilde{a}_v^{(k)} = \beta^{(k)}(a_v^{(k)}) = \text{clamp}\left(\frac{1}{2}\left(\frac{\sum_{i=0}^{\tau} \frac{\sin((2i+1)(a_v^{(k)} \odot \tilde{W}^{(k)} + \tilde{b}^{(k)}))}{2i+1}}{\sum_{i=0}^{\tau} \frac{\sin\left((2i+1)\frac{\pi}{2}\right)}{2i+1}} + 1\right), 0, 1\right)$$



On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

Main difficulties

- ~~When aggregating nodes, we apply rules on the number of neighbors which have a determinate pattern. Instead of using a simple threshold, can we learn ranges?~~
- Sigmoid makes TELL hard to train in deep networks: vanishing gradient

On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

Training by distillation

- We train a model of 5 GIN layers with Gumbel-Sigmoid activation function

On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

Training by distillation

- We train a model of 5 GIN layers with Gumbel-Sigmoid activation function

- We then initialize a model of 5 Logic-GIN layers with the same characteristics

On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

Training by distillation

- We train a model of 5 GIN layers with Gumbel-Sigmoid activation function
- We then initialize a model of 5 Logic-GIN layers with the same characteristics
- We pretrain the logic model to replicate the first one layer-by-layer

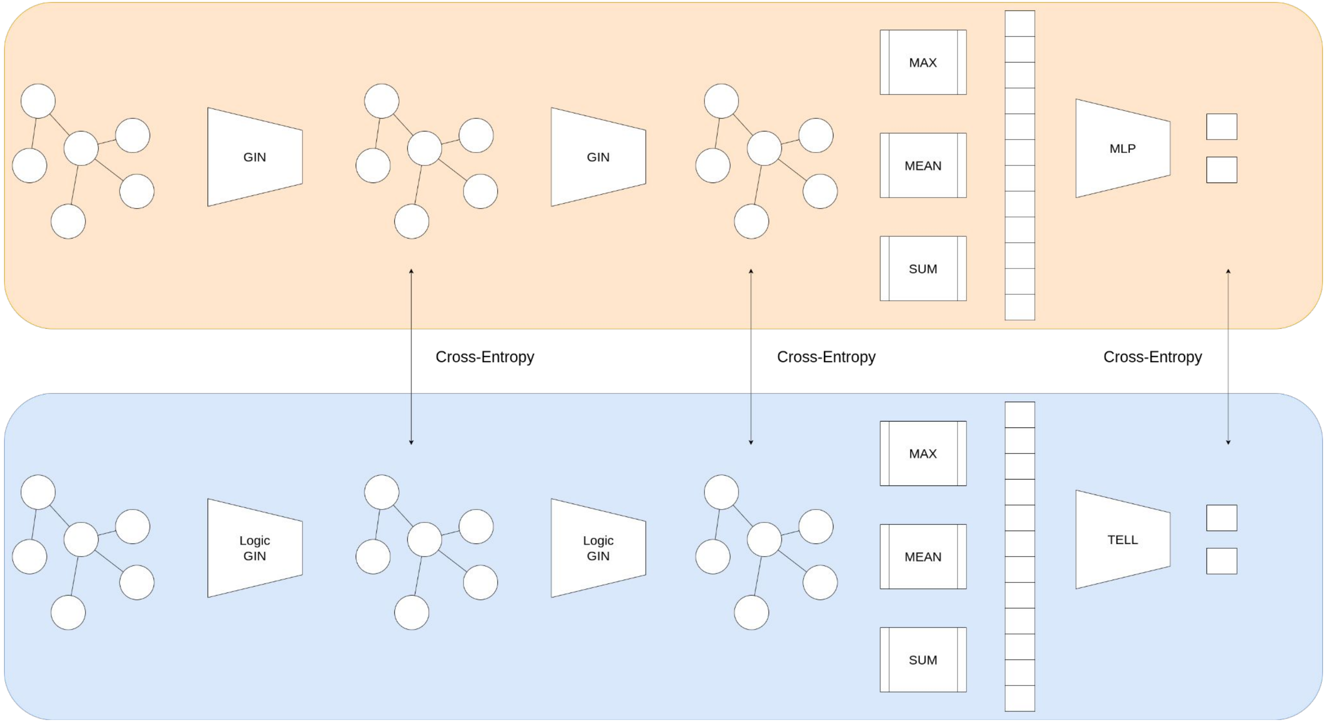
On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

Training by distillation

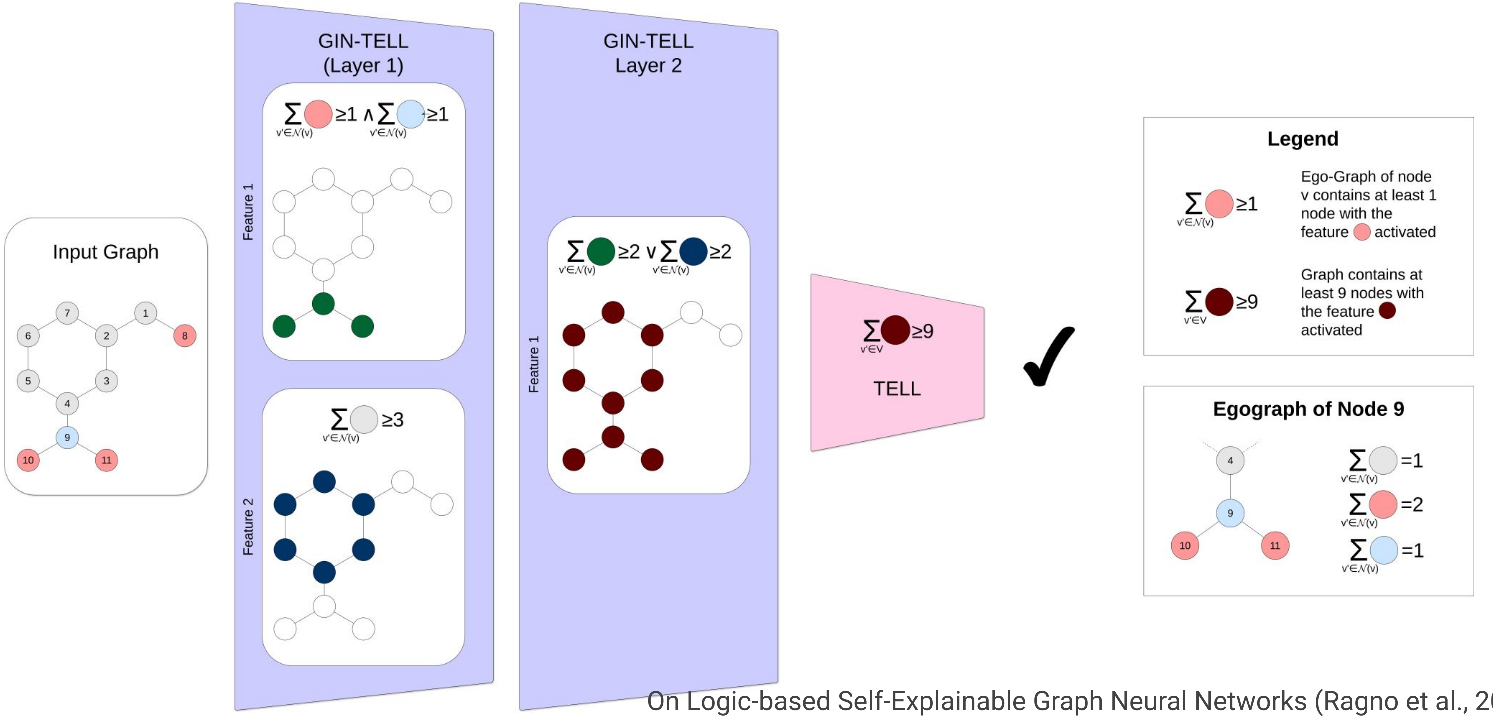
- We train a model of 5 GIN layers with Gumbel-Sigmoid activation function
- We then initialize a model of 5 Logic-GIN layers with the same characteristics
- We pretrain the logic model to replicate the first one layer-by-layer
- We terminate training with classic backpropagation on the classification task

On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

Pre-Training Procedure

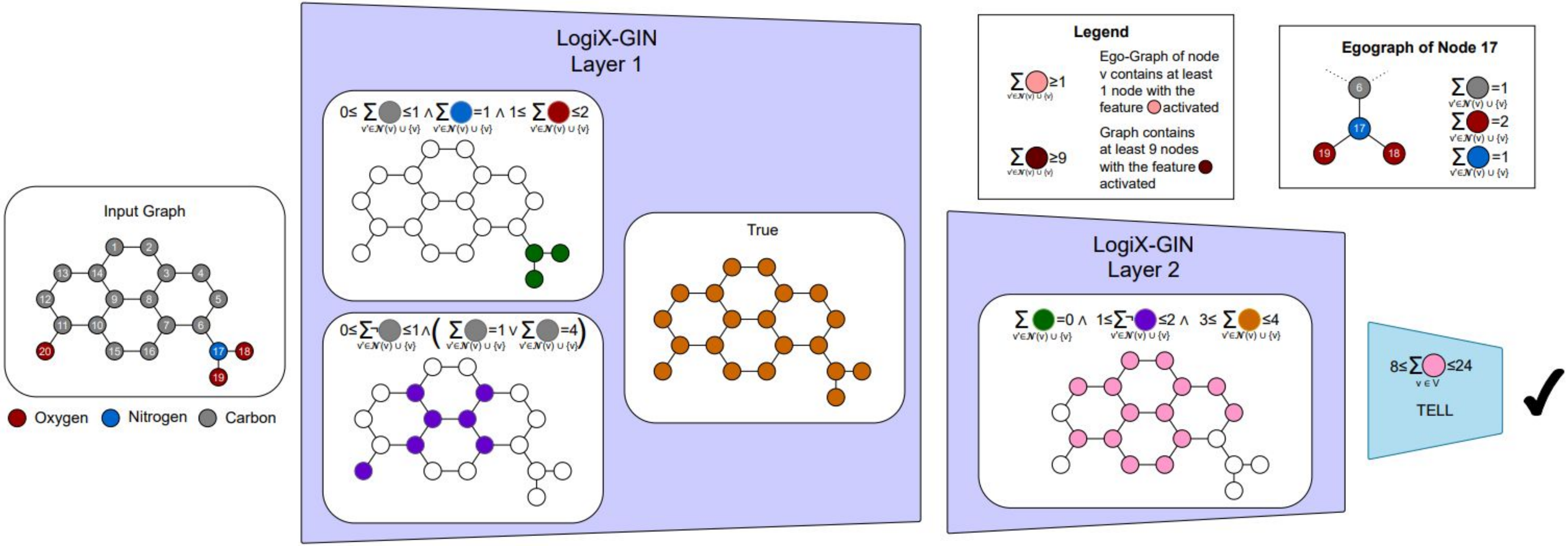


How does Logic GIN work?



On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

An actual explanation...



On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

Graded Modal Logic

Graph Structure : $G = (V, E, \{P_i\}\ i \in I)$

- V : finite set of nodes
- $E \subseteq V \times V$: symmetric edge relation
- $P_i \subseteq V$: unary relations interpreting predicates

Satisfaction Relation $G, v \models \varphi(x)$:

- $G, v \models P(x)$: node v satisfies predicate P
- $G, v \models \neg\psi(x)$: node v does not satisfy $\psi(x)$
- $G, v \models \psi_1(x) \wedge \psi_2(x)$: v satisfies both formulas
- $G, v \models \psi_1(x) \vee \psi_2(x)$: v satisfies at least one formula
- $G, v \models \exists^{\geq N} y\ (E(x, y) \wedge \psi_3(y))$: at least N neighbors satisfy ψ_3

Connection to GNNs : Barceló et al. show GML precisely characterizes node properties computable by certain GNN classes.

On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

Graded Modal Logic

Unary Predicates: Blue(x) and Red(x), denote node colors

Example 1: Atomic predicate

$$\varphi(x) := Blue(x)$$

Formula holds if x is blue.

Example 2: At least two red neighbors

$$\varphi(x) := \exists^{\geq 2} y \, (E(x, y) \land Red(y))$$

Node x has at least two neighbors that are red.

Example 3: Blue node without red neighbors

$$\varphi(x) := Blue(x) \land \neg \exists^{\geq 1} y \, (E(x, y) \land Red(y))$$

Node x is blue and has no red neighbors.

On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

The Circle of Logic

Barcelo et al. demonstrate that we can learn GML with GNNs

GML

GNNs

We demonstrate that we can learn a GNN that yields GML rules

Results

Dataset	Black-box GIN	LogiX-GIN	Acc. Decay	Class. Obj.
BAMultiShapes	100.00 ± 0.00	100.00 ± 0.00	0.00	Graph
BA2Motifs	100.00 ± 0.00	100.00 ± 0.00	0.00	Graph
BBBP	87.95 ± 2.07	85.90 ± 0.99	2.05	Graph
MUTAG	84.74 ± 8.02	82.63 ± 5.58	2.11	Graph
NCI1	81.87 ± 1.39	78.93 ± 1.51	2.94	Graph
PROTEINS	72.59 ± 3.04	72.05 ± 5.77	0.54	Graph
Mutagenicity	82.21 ± 1.87	79.31 ± 1.25	2.90	Graph
BaShapes	97.14 ± 0.95	94.29 ± 2.52	2.85	Node
BaCommunity	81.00 ± 1.59	84.14 ± 4.55	-3.14	Node
TreeGrid	100.00 ± 0.00	98.71 ± 0.87	1.29	Node

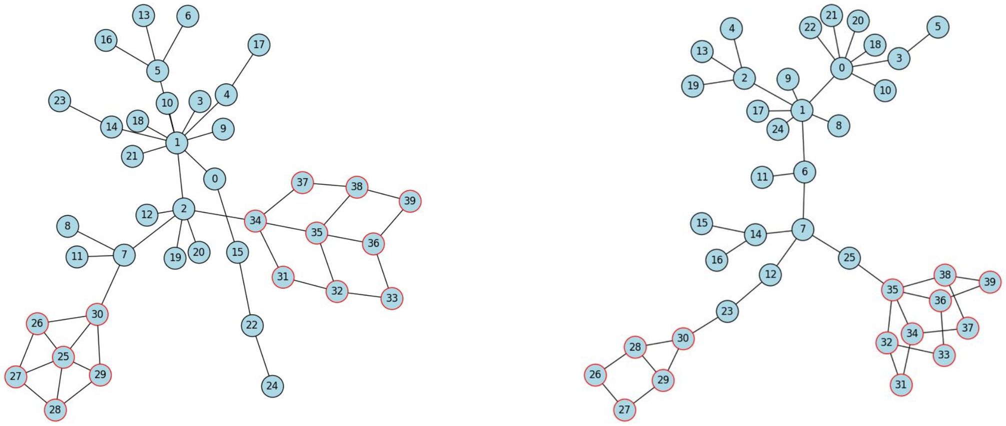
On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

Results

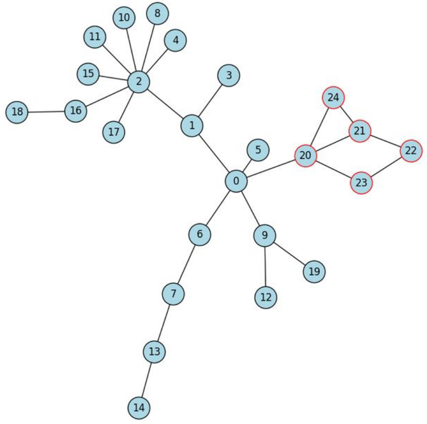
Dataset	LogiX-GIN	PiGNN	GIB	KerGNN	GNAN
Ba2Motifs	100.00 ± 0.00	99.89 ± 0.31	100.00 ± 0.00	98.80 ± 0.75	49.10 ± 0.66
BaMultiShapes	100.00 ± 0.00	85.40 ± 5.08	97.60 ± 2.06	83.20 ± 1.94	49.10 ± 0.66
MUTAG	82.63 ± 5.58	82.51 ± 10.48	90.53 ± 6.14	82.11 ± 9.18	55.79 ± 16.43
Mutagenicity	79.31 ± 1.25	82.39 ± 1.68	80.14 ± 0.98	73.32 ± 2.93	55.36 ± 0.52
NCI1	78.93 ± 1.51	78.54 ± 2.74	78.15 ± 1.32	69.00 ± 1.38	50.80 ± 1.15
PROTEINS	72.05 ± 5.77	70.00 ± 2.44	68.47 ± 5.00	72.97 ± 4.87	57.67 ± 2.62
BBBP	85.90 ± 0.99	83.54 ± 0.37	84.78 ± 2.57	84.12 ± 2.09	22.80 ± 0.86

On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

Can Logic GIN identify relevant subgraphs?



BaMultiShapes



Ba2Motifs

On Logic-based Self-Explainable Graph Neural Networks (Ragno et al., 2025, NeurIPS)

Limitation and Future Work

- Due to GML limitations, LogiX-GIN expressiveness is limited to counting nodes in neighborhoods.
- Training takes a lot of time due to the distillation procedure (this is TELL's fault)

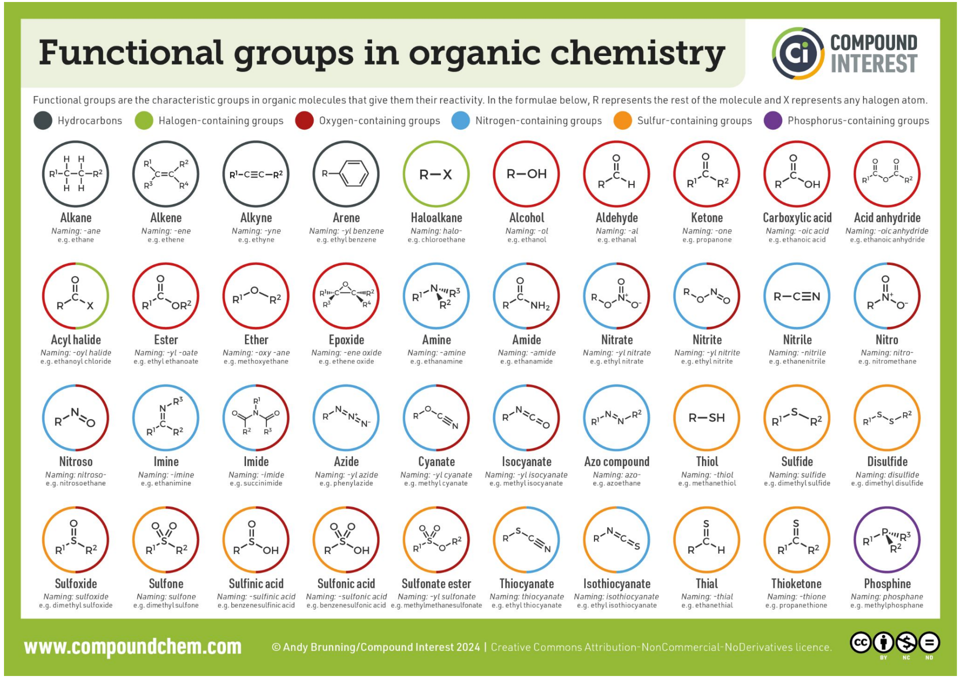
Future Work:

Limitations + Real world domains such as chemistry

Preliminary Results: Higher Expressivity

Dataset	Gumbel-GIN	Gumbel-Me	LogiX-GIN	LogiX-Me
BaMultiShapes	99.90 ± 0.32	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00
Ba2Motifs	75.00 ± 26.48	99.90 ± 0.32	100.00 ± 0.00	100.00 ± 0.00
BBBP	88.54 ± 1.89	86.20 ± 2.47	85.90 ± 0.99	86.44 ± 2.02
MUTAG	85.26 ± 8.88	78.95 ± 9.28	82.63 ± 5.58	85.26 ± 4.40
NCI1	80.95 ± 1.19	80.36 ± 1.55	78.93 ± 1.51	77.57 ± 1.70
PROTEINS	$75.00 \pm nan$	73.39 ± 4.47	72.05 ± 5.77	75.18 ± 6.23
Mutagenicity	81.75 ± 1.75	81.64 ± 1.93	79.31 ± 1.25	78.62 ± 3.02
BaShapes	95.00 ± 1.68	96.86 ± 0.60	94.29 ± 2.52	97.43 ± 1.56
BaCommunity	84.71 ± 2.36	95.07 ± 1.63	84.14 ± 4.55	94.29 ± 1.01
TreeGrid	96.94 ± 1.97	98.71 ± 1.58	98.71 ± 0.04	99.35 ± 0.36

Preliminary Results: Chemistry



	dataset	acc	rocauc	exp	model	grouping_method
0	HIV	96.98 ± 0.38	75.31 ± 2.71	256	GIN	NaN
1	HIV	96.77 ± 0.70	73.85 ± 3.52	64	FG_GIN	functional
2	HIV	97.08 ± 0.35	72.70 ± 2.97	64	FG_GIN	brics
3	HIV	96.86 ± 0.60	75.65 ± 3.07	64	FG_LOGIXGIN	functional
4	HIV	96.93 ± 0.64	73.30 ± 3.48	64	FG_LOGIXGIN	brics
5	BBBP	81.95 ± 2.18	80.89 ± 6.92	256	GIN	NaN
6	BBBP	81.95 ± 4.46	85.27 ± 4.52	64	FG_GIN	functional
7	BBBP	79.61 ± 3.80	81.86 ± 6.24	64	FG_GIN	brics
8	BBBP	83.32 ± 3.49	85.26 ± 4.09	64	FG_LOGIXGIN	functional
9	BBBP	81.27 ± 3.79	82.67 ± 5.83	64	FG_LOGIXGIN	brics
10	Tox21	71.32 ± 10.40	nan ± nan	106	GIN	NaN
11	Tox21	71.05 ± 10.45	nan ± nan	30	FG_GIN	functional
12	Tox21	71.02 ± 10.31	nan ± nan	32	FG_GIN	brics
13	Tox21	71.09 ± 10.47	nan ± nan	36	FG_LOGIXGIN	functional
14	Tox21	70.70 ± 10.38	nan ± nan	31	FG_LOGIXGIN	brics
19	BACE	76.03 ± 2.12	81.67 ± 2.32	256	GIN	NaN
20	BACE	74.17 ± 3.54	82.13 ± 2.99	64	FG_GIN	functional
21	BACE	76.16 ± 2.77	81.31 ± 0.76	64	FG_GIN	brics
22	BACE	74.44 ± 4.04	79.47 ± 4.03	64	FG_LOGIXGIN	functional
23	BACE	75.10 ± 4.10	81.22 ± 1.98	64	FG_LOGIXGIN	brics

Thank You