Neural DNF-MT: A Neuro-symbolic Approach for Learning Interpretable and Editable Policies

Kexin Gu Baugh[1], Luke Dickens[2], and Alessandra Russo[1]

[1]Imperial College London [2]University College London

Motivation

- Interpretability of RL policies
  - Important in high-stake decision making
  - Amenable to policy intervention
- Limitation to existing neuro-symbolic RL methods
  - Rule templates/mode biases to restrict search space
  - Human-engineered predicates
  - Pre-trained model to extract predicates

Neural DNF-MT

We propose a neuro-symbolic model called *Neural DNF-MT*.

Contributions

- **Learning interpretable policy**
  - **Differentiable learning**: trained in standard deep actor-critic algorithm + predicate invention without human engineering
  - **Interpretable**: providing logical representations: probabilistic for stochastic policies and bivalent for deterministic policies
- **Ability to intervene on deterministic policy (represented in logic)**
  - **Bidirectional**: inject changes made on the logic representation back to the neural model, without re-training
  - **Faster**: inference with neural DNF-MT is at least 200 times faster than inference in logic

Neural DNF-based Model

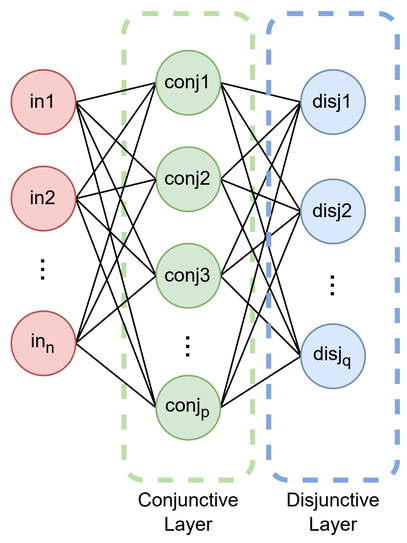
The building block of a neural DNF-based model is a semi-symbolic node [1]:

$$y = f_{\text{activation}}\left(\sum_{i=1}^{I} w_i x_i + \beta\right),$$
$$\text{with } \beta = \delta\left(\max_{i=1}^{I}|w_i| - \sum_{i=1}^{I}|w_i|\right)$$

$w_i \in \mathbb{R}$: trainable weights; $x_i \in [-1, 1]$: inputs
β: bias that enforces semi-symbolic behaviour
$\delta \in [-1, 1]$: controls the node's characteristics, which induces behaviour analogous to a logical conjunction (disjunction) when $\delta = 1(-1)$
$f_{\text{activation}}$: activation function that maps the output to $(-1, 1)$
$y_i \in (-1, 1)$: activation of the semi-symbolic node, which represents its belief in a proposition

Example: A conjunctive node with weights $[6, 0, -6, 0, 0, 6]$ can be translated into a logical form of $conj \leftarrow a\_0, \text{not } a\_2, a\_5$ (zero-indexed).

A neural DNF model can be constructed by a conjunctive layer and a disjunctive layer, both with $\tanh$ activation function:



The Mutex-Tanh Activation

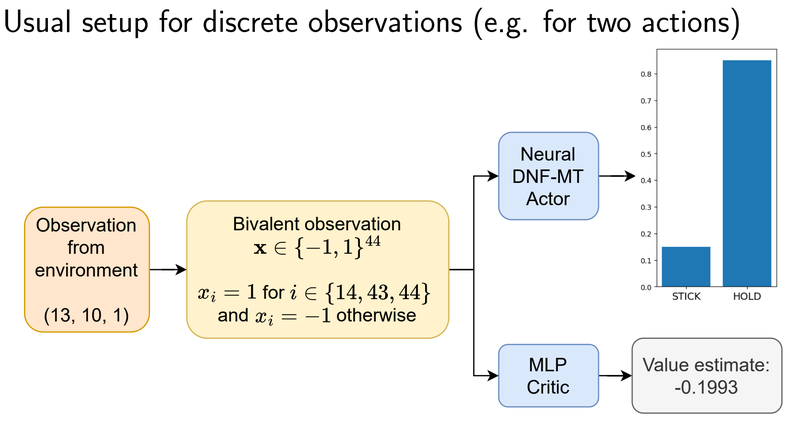
Previous methods [1, 2] used $\tanh$ activation function in the neural DNF-based models. They do not support probabilistic interpretation. But for RL tasks, the model needs to approximate arbitrary probabilities.
We propose a new activation function *Mutex-Tanh*:

$$\text{softmax}(\mathbf{d})_k = e^{d_k} / \sum_i^N e^{d_i}$$
$$\text{mutex-tanh}(\mathbf{d})_k = 2 \cdot \text{softmax}(\mathbf{d})_k - 1$$

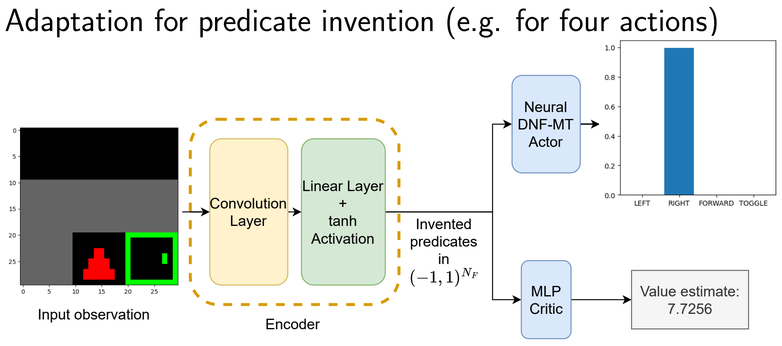
Mutex-tanh activation is used only at the disjunctive layer, while the conjunctive layer still uses $\tanh$.

Neural DNF-MT in Actor-critic PPO

Usual setup for discrete observations (e.g. for two actions)



Adaptation for predicate invention (e.g. for four actions)

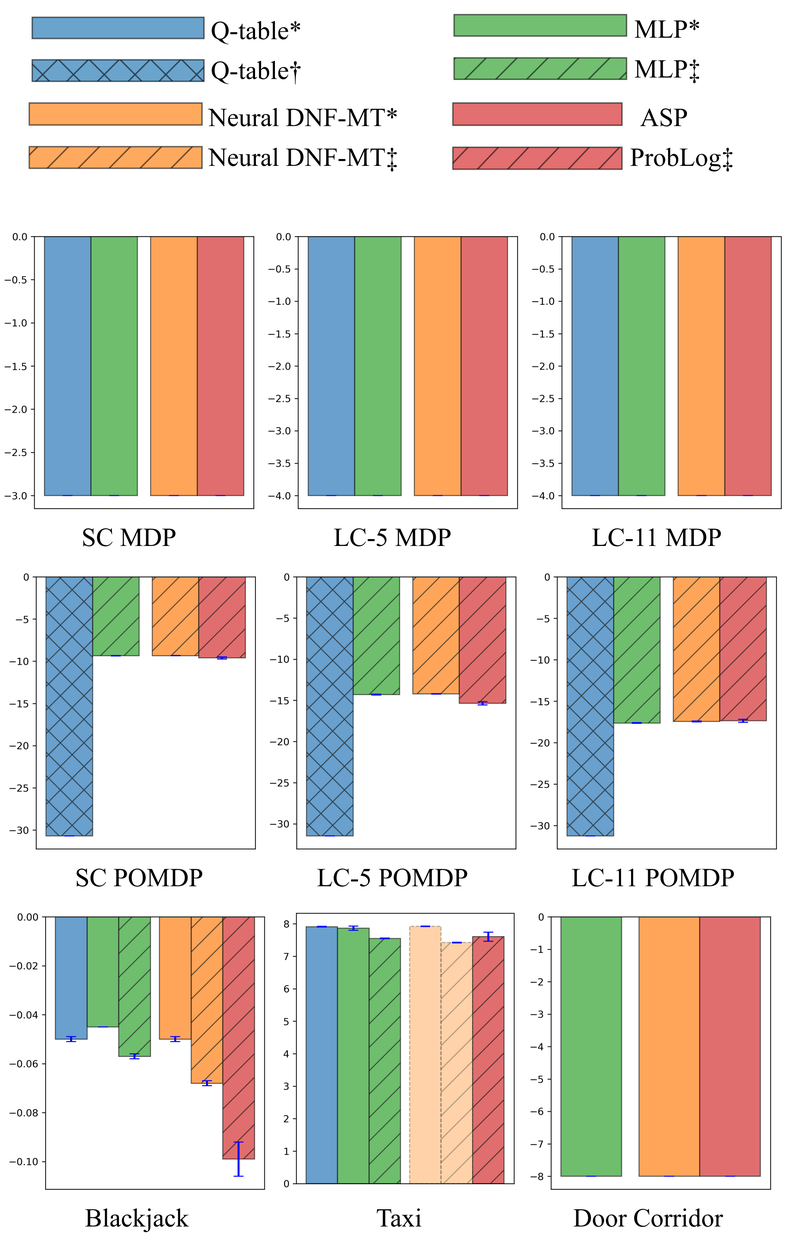


Using an *encoder* to extract meaningful symbolic features

Experiment Overview

- Comparable results to standard RL baselines (Q-learning & neural actor-critic PPO)
- In some environments, there is a trade-off between interpretability and performance

*: argmax action selection, †: ϵ-greedy sampling, ‡: actor's distribution sampling



Paper

Code

[1] Nuri Cingillioglu and Alessandra Russo. "pix2rule: End-to-end Neuro-symbolic Rule Learning". In: *Proceedings of the 15th International Workshop on Neural-Symbolic Learning and Reasoning (NeSy 2021) as part of the 1st International Joint Conference on Learning & Reasoning (IJCLR 2021)*. Ed. by Artur d'Avila Garcez and Ernesto Jiménez-Ruiz. CEUR Workshop Proceedings, 2021. url: https://ceur-ws.org/Vol-2986/paper3.pdf.

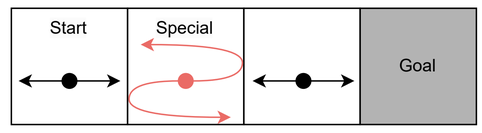
[2] Kexin Gu Baugh, Nuri Cingillioglu, and Alessandra Russo. "Neuro-symbolic Rule Learning in Real-world Classification Tasks". In: *Proceedings of the AAAI 2023 Spring Symposium on Challenges Requiring the Combination of Machine Learning and Knowledge Engineering (AAAI-MAKE 2023)* (Hyatt Regency, San Francisco Airport, California, USA). Ed. by Andreas Martin et al. Vol. Vol-3433. CEUR Workshop Proceedings, 2023. url: https://ceur-ws.org/Vol-3433/paper12.pdf.

Policy Interpretation & Intervention

Switcheroo Corridors

- Two actions: going left and going right
- Reward: -1 per step
- Special state *reverses* the agent's action
- Two types of observations
  - State number (which state the agent is in): *MDP* & *deterministic* policy
  - Wall status (left and right wall): *POMDP* & *stochastic* policy (no memory)

Small Corridor (SC): 4 states, start at 0, goal at 3, special at 1



ASP rules of a neural DNF-MT actor in SC MDP

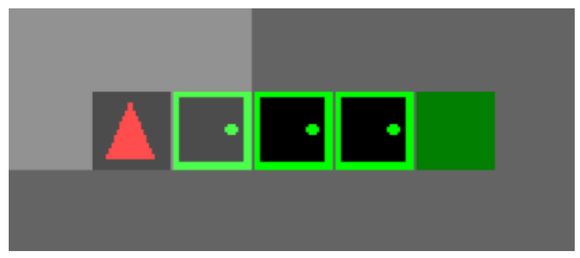
```
action(left) :- in_s_1.    action(right) :- not in_s_1.
```

ProbLog rules of a neural DNF-MT actor in SC POMDP

```
0.041::action(left) ; 0.959::action(right) :-
  left_wall_present, \+ right_wall_present.
0.581::action(left) ; 0.419::action(right) :- \+
  left_wall_present, \+ right_wall_present.
```

Door Corridors

- Based on Key Corridor in Minigrid, but no keys, only togglable doors
- Four actions: turn left, turn right, move forward, and toggle
- Agent observes a 3x3 grid around itself



ASP rules of a neural DNF-MT actor in Door Corridor

```
action(turn_right) :- a_5, a_8.
action(forward) :- a_2.
action(toggle) :- a_3.
% Definitions of each invented predicate a_i:
a_2 :- top_right_corner_wall.
a_3 :- one_step_ahead_closed_door.
a_5 :- not curr_location_open_door,
       not one_step_ahead_closed_door.
a_8 :- two_step_ahead_unseen.
```

Actions for Door Corridor: right, toggle, forward, toggle, forward, toggle, forward, forward

Policy Modification in Modified Door Corridor

We change how the agent interacts with the goal state: instead of stepping into the goal (Door Corridor), it needs to toggle it (DC-T).
Instead of re-training, we adapt the existing policy:

ASP rules of a neural DNF-MT actor in DC-T

```
action(turn_right) :- a_5, a_8.
action(forward) :- not a_1, a_2.
action(toggle) :- a_3.
action(toggle) :- a_1, not a_3, a_12.
```

Actions for DC-T: right, toggle, forward, toggle, forward, toggle, forward, *toggle*

Discussion

- **Performance**: Comparable to standard RL baselines (Q-learning & neural actor-critic PPO)
- **Interpretability**: Two forms of interpretability in probabilistic and bivalent logic
- **Inference**: Support both neural + symbolic inference, and neural is faster
- **Policy intervention**: Edit bivalent logic program, port back to neural model
- **Future work**: Can be helpful if there's background knowledge: provide a hot-start for training

aamas2025 DETROIT SPIKE IMPERIAL