The ArgNet

Taking inspiration from the explicitly relational network (PrediNet) of https://arxiv.org/abs/1905.10307 (deepmind + imperial college), I tried to design an explicitly argumentative network (ArgNet) capable of explicitly learning an argumentation framework, extracting non monotonic relations between rules and entities.
An argumentation framework is a graph defining a hierarchy of rules, so that there are rules that attack others.
The ArgNet builds on the top of the PrediNet, by defining attack relations between the predicates explicitly learned by PrediNet.
More in details: in ArgNet an attack relation is implemented as a mask to apply to the set of learned predicates. The attack mask is learned by a relational layer (that is put on top of the relational layer of the PrediNet), using as comparator a "relu" instead of a "subtract".

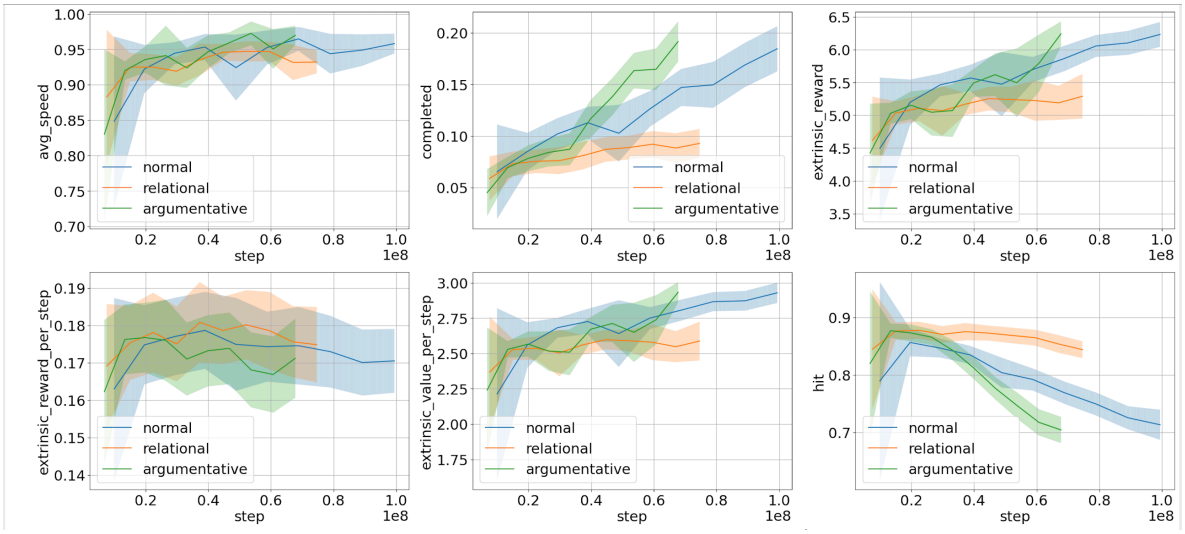
An ArgNet is meant to be helpful in all those cases in which some kind of non-monotonic reasoning is required for solving a task.
In order to verify the effectiveness of the aforementioned ArgNet, we tested it on a toy environment: CescoDrive.
CescoDrive is characterised by a continuous action-space and state-space, in which a car has to learn to follow a lane and to avoid obstacles by controlling acceleration and steering angle.
The rules of "following the lane" and "avoiding obstacles" are enforced by the reward function:

- The more the agent is far from the lane, the lower the reward.
- When the agent hits an obstacles it gets a high negative reward and the game terminates.

Considering that obstacles may randomly appear on the lane, we might expect a sort of hierarchy of rules in which the rule of "following the lane" is defeated by the rule of "avoiding obstacles", in order to maximise the cumulative return of the agent.
This kind of hierarchy can be seen as a rudimental and minimal argumentation framework.

Said that, we would expect that an agent equipped with ArgNet would learn faster to tackle CescoDrive, because it would be equipped with the necessary inductive bias to tackle non-monotonic reasoning.
We ran some experiment to check this out. In the following image we

show the results of such experiments.



The relational network (PrediNet) is the same of https://arxiv.org/abs/1905.10307, they shared the source code.
The normal network is a neural network designed by OpenAI, shown to be effective on many Atari games.

As expected, results clearly show that the agent learns faster to avoid obstacles (see the “obstacle hit” plot and the “track completion” plot), by using an argumentative network that explicitly learns a hierarchy of rules. That is: it learns that the rule of “avoiding obstacles” attacks the rule of “following the lane”.