Temporal Watermarks for Deep Reinforcement Learning Models

Prakash K. Naikade
prna00001@stud.uni-saarland.de

1 Summary

According to the paper, existing white-box watermarking solutions uses parameter-embedding methods which requires the model owner to have full access to the parameters during verification, and becomes useless in the case where the target model is a black-box to external users. To solve this problem this paper proposes a novel temporal-based black-box watermarking methodology for deterministic and stochastic Deep Reinforcement Learning (DRL) models and tasks. To overcome the problem of changed model behaviour induced by watermark inference, the paper introduced three new concepts, (1) new algorithm to identify $damage - free\ states$, from which the DRL system can still be safe and reliable when there is a deviation of action probability, (2) a new $reward\ function$ to efficiently implant the necessary watermark behaviours into the model for both deterministic and stochastic reinforcement learning algorithms, (3) a $statistic\ tests$ to verify the action probability distributions of the damage-free watermark states. The paper also developed a $damage\ metric$ for creating watermarks that are safe to integrate in the target model. Experiments shows generalized nature of proposed watermarking scheme.

1.1 Overview of Temporal Watermarking

Proposed methodology tries to fulfill four laid-down requirements, $Functionality\ preserving$ (watermarked model should exhibit the competitive performance compared with original model), $State\ preserving$ (good watermarking scheme should use the watermark states sampled from the same state space), $Damage\ free$ (backdoors should not change the prediction results on the watermark samples) and $Robustness$ (embedded watermarks can not be removed by model transformations).

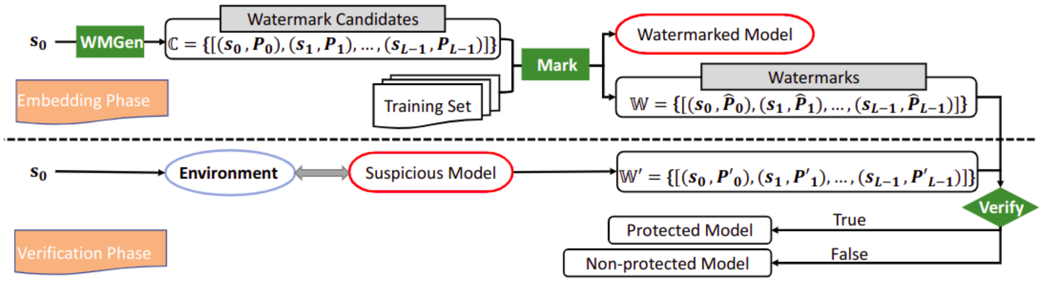
s_0
WMGen
Watermark Candidates
$\mathbb{C} = \{[(s_0, P_0), (s_1, P_1), \ldots, (s_{L-1}, P_{L-1})]\}$
Embedding Phase
Training Set
Mark
Watermarked Model
Watermarks
$\mathbb{W} = \{[(s_0, \widehat{P}_0), (s_1, \widehat{P}_1), \ldots, (s_{L-1}, \widehat{P}_{L-1})]\}$
s_0
Environment
Suspicious Model
$\mathbb{W}' = \{[(s_0, P'_0), (s_1, P'_1), \ldots, (s_{L-1}, P'_{L-1})]\}$
Verify
True
Protected Model
False
Non-protected Model
Verification Phase

Figure 1: Temporal watermarking methodology

The paper proposes three new algorithms to achieve temporal watermarking and satisfy above mentioned requirements,
(1) **WMGen** algorithm looks for damage-free states in order to create a dataset of watermark candidates that are safe and don't interfere with the DRL policy. If the agent can choose any legal actions

Preprint. Under review.

at state s to perform the task correctly then s is damage-free. On the other hand, A significant action probability distribution (APD) variance, indicates that the agent tends to choose a specific action with strong will at state s, implying that s is crucial for the task and could result in a crash if other actions are chosen instead.
(2) **Mark** algorithm embed the identified set of watermark candidates into the target DRL model. **Mark** forces the model to predict different actions on the damage-free states in these watermark candidates. To achieve this, the proposed new reward function is used that adds an incentive reward on the original one over the damage-free states.
(3) **Verify** algorithm uses statistic tests to compare the probability distributions of state-APD pairs to identify the presence of watermarks.
Proposed workflow of watermarking scheme is as follows,
WMGen is called by the model owner during the embedding phase to generate a dataset of watermark candidates. He then employs **Mark** to train a watermarked model and generate the final watermark sequences $\mathbb{W}$. He queries a suspicious model with the states of each watermark sequence during the verification phase and obtains the runtime results $\mathbb{W}^{'}$. The owner can check if the suspect model is the watermarked or not by comparing $\mathbb{W}$ and $\mathbb{W}^{'}$ using **Verify**.

2 Strengths

The paper proposes general watermarking scheme for both deterministic and stochastic policy as compared to Behzadan and William, 2019; which is unique and strong step in this area of research. As this watermarking scheme avoids using spatial triggers, perturbations, or out-of-distribution states in different environments as watermark states, it relies solely on black box access, which makes the model uniquely distinguishable and preserves the original model performance. Also, this watermarking scheme shows robustness against various model transformations.

3 Weaknesses

Sometimes it is difficult to obtain full information about states and actions in some complex environments where this scheme will struggle to generate watermarked model. Also the performance of watermarking embedding and verification is sensitive to many hyperparameters.

4 Possible Improvements

This watermarking scheme should be evaluated in more DRL tasks specifically more complex stochastic environments. Also it should be tested against more model transformation techniques to support the claims of robustness.

5 Possible Extensions

Developing more efficient and sophisticated attacks to remove DRL watermarks in order to develope robust watermarking scheme could be important extension of this research area.