Replication Experiments and Results Analysis of the STARLA Method

This file provides our reproduction of the STARLA method, which is proposed in the paper [1] and its application to test the DRL agents trained for three benchmark RL tasks, including Flappy Bird, Pong, and Catcher. Considering that STARLA requires additional historical episodes to train a ML model to predict the failure probability of generated sequences, we collected about 1500 complete episodes to train the ML classifier. Following this, harnessing the objective function and the genetic algorithm delineated in the paper, coupled with operations like crossover and mutation.

Over 10 independent repeated runs, we obtained 500.6, 176.4, and 127.6 synthesized sequences predicted to be failures. Upon actual execution, it was averagely found that only 245.7, 6.5, and 30.9 of these synthesized sequences manifested actual failures during execution. This number is significantly lower than what DRLFuzz achieved. Therefore, the preliminary experiment indicates that, **under the tasks and DRL models presented in our paper, DRLFuzz performs better than STARLA**.

Considering that STARLA is sensitive to the hyperparameter *d*, we carried out experiments with different values, specifically 0.05, 0.1, 0.5, and 1. In fact, the aforementioned test results represent our selection of the best-performing value for test case generation. The less-than-optimal performance of STARLA in three RL tasks is attributed to the poor predictions of the ML classifier. To illustrate, we provide detailed experimental results for the predictive model across three RL tasks as follows.

Table 1 Prediction of Functional Faults with Random Forest in the Flappy Bird case study

d	Abstract States	Accuracy	Precision	Recall	F1-measure
0.05	72980	57.4%	61.8%	28.5%	0.3800
0.1	32728	59.0%	65.1%	29.9%	0.4019
0.5	2880	81.5%	89.0%	69.3%	0.7783

1	928	81.5%	87.1%	71.4%	0.7846

Table 2 Prediction of Functional Faults with Random Forest in the Pong case study

d	Abstract States	Accuracy	Precision	Recall	F1-measure
0.05	976797	98.6%	0.00%	0.00%	0
0.1	350206	98.6%	0.00%	0.00%	0
0.5	9916	98.6%	0.00%	0.00%	0
1	1823	98.6%	10.0%	1.2%	0.0222

Table 3 Prediction of Functional Faults with Random Forest in the Catcher case study

d	Abstract States	Accuracy	Precision	Recall	F1-measure
0.05	155485	88.1%	99.3%	38.7%	0.5538
0.1	46333	88.3%	99.3%	39.8%	0.5646
0.5	1233	88.9%	99.8%	42.7%	0.5947
1	268	88.2%	99.9%	39.3%	0.5623

The experimental results highlight that **the efficacy of search-based testing approach guided by a prediction-based objective function relies on the performance of the prediction model**. However, well-trained DRL agents often have a low probability of failure. Hence, within the interaction episodes collected, the vast majority do not fail, resulting in a severe class imbalance issue in the training data.

From the above results, it is evident that the performance of the trained classifier may not be accurate enough in the presence of class imbalance, where a well-trained RL agent produces substantially more non-failed test sequences compared to failed ones. Despite the classifier having a high Precision value, its Recall value is quite low. In the context of class-imbalanced classification problems, its overall accuracy for positive

and negative instances (measured by a more comprehensive metric, the F1-measure) is relatively poor. Thus, when it is challenging to accurately predict the failure probability of synthesized sequences, the test cases generated by guiding through a prediction-based objective function might exhibit false-positive issues, meaning that a significant number of sequences predicted as failures by the model do not actually fail.

This observation is also confirmed in the STARLA paper. The paper mentioned that the average episode reward for the agent in the CartPole task was 124, while the optimal episode reward was 200, suggesting a less-than-optimal testing performance of the tested agent. In such a scenario, the historical data obtained from the environment do not present a severe class imbalance issue. However, in our experiment, the agent performs well after extensive training in most scenarios, as reflected in the high average values of the episode rewards. Therefore, the training data exhibits significant class imbalance, making the training of the prediction model highly challenging.

The above presents our experimental analysis using STARLA to test trained agents on the Flappy Bird, Pong, and Catcher tasks. It is worth noting that these are preliminary experimental results, validating some of the limitations mentioned by the authors of the STARLA paper in their discussion section. The code has been uploaded to the GitHub repository, allowing interested readers to replicate and explore STARLA.

Reference:

[1] Zolfagharian A, Abdellatif M, Briand L C, et al. A Search-Based Testing Approach for Deep Reinforcement Learning Agents[J]. IEEE Transactions on Software Engineering, 2023.