Report on the outcomes of a Short-Term Scientific Mission[1]

Action number: CA22145

Grantee name: Manuel Eberhardinger

Details of the STSM

Title: Explaining the Decision-Making Process of Board Game Agents with Program Synthesis

Start and end date: 15/07/2024 to 26/09/2024

Description of the work carried out during the STSM

Description of the activities carried out during the STSM. Any deviations from the initial working plan shall also be described in this section.

In this STSM, a general framework for learning programmatic representations of board game agents based on imitation learning has been explored. We use a genetic programming (GP) algorithm for learning features of board games based on [1]. The search space of GP is formed by a typed probabilistic grammar from which we can sample programs from. After finding features with GP, we can use these features to generate a dataset for training a decision tree (DT) by executing the feature programs on the board observation and returning a value indicating how strongly the feature is activated.

This tree can be used to generate post-hoc explanations for the moves by highlighting the relevant features for decision making by following the decision path of the DT. In the best case, the DT can also be used to play the game, but currently the focus is more on post-hoc explanations, as training a DT to play the game with programmatic representations is more difficult than generating post-hoc explanations.

This was observed by using the trained DT as a game playing agent for the game Othello. With the current framework, the DT was disqualified most of the time for using incorrect moves. In these experiments against 402 different MCTS agents from various training stages, the DT was disqualified 799 times, lost 2 times and won 3 times. A total of 804 games were evaluated, so that the DT starts as the first player in the first game and the second player in the second game.

Some generated explanations of the DT are shown in the following for a given game board of the Othello game:

[1] This report is submitted by the grantee to the Action MC for approval and for claiming payment of the awarded grant. The Grant Awarding Coordinator coordinates the evaluation of this report on behalf of the Action MC and instructs the GH for payment of the Grant.

```
Rules used to predict sample 0:

decision node 0 : (X_test[0, 50] = 0.0) <= 0.007936508394777775)
(get-board-feature  $1 (scanning-piece-type  LEFT  $1 (get  $1 (cell  7  4 ))(if_object (eq-obj? (get  $1 (cell  1  6 ))(get-game-piece (get
$1 (cell  3  2 )))) Black  Black )))

decision node 224 : (X_test[0, 18] = 0.0) <= 0.016807910054922104)
(get-board-feature  $1 (scanning-piece-type  RIGHT  $1 (get  $1 (cell  1  4 )) White ))

decision node 225 : (X_test[0, 55] = 0.0) <= 0.01613322924822569)
(get-board-feature  $1 (scanning-same-cell-piece  UP  $1 (get  $1 (cell  1  6 ))))

decision node 226 : (X_test[0, -2] = 0.0) > -2.0)
(get-board-feature  $1 (scanning-piece-type  UP-RIGHT  $1 (get  $1 (cell  7  2 )) White ))
```

This decision path includes four tests of activated or non-activated features, but after examining the checked features, it became clear that these features are not useful and only the training data set is well discriminated. Decision node 224 checks the cell (1, 4) and scans the right side of the cell for white pieces, but the activation of this feature is 0.0, i.e., there are no white pieces to the right of (1, 4). All tested features have an activation of 0 in the above explanations and thus no expressive power.

We think that the problem with our approach is that only state features are learned from the data without considering the action, resulting in the DT having difficulty correctly associating the board state with the action, and so shortcuts are often learned that correctly discriminate the training data set but are unable to generalize to data outside the distribution, which is a common problem in machine learning research [2].

To mitigate this problem, we plan to learn state-action features similar to [3]. The main difference is that our framework aims to learn features from data, whereas [3] learns the features in the training process of the MCTS agent.

[1] Manuel Eberhardinger, Florian Rupp, Johannes Maucher, & Setareh Maghsudi (2024, August). Unveiling the Decision-Making Process in Reinforcement Learning with Genetic Programming. In International Conference on Swarm Intelligence (pp. 349-365).

[2] Geirhos, R., Jacobsen, JH., Michaelis, C. *et al.* Shortcut learning in deep neural networks. *Nat Mach Intell* **2**, 665–673 (2020). https://doi.org/10.1038/s42256-020-00257-z

[3] Dennis J. N. J. Soemers, Eric Piette, Matthew Stephenson, & Cameron Browne (2023). Spatial State-Action Features for General Games. Artificial Intelligence, 321, 103937.

Description of the STSM main achievements and planned follow-up activities

Description and assessment of whether the STSM achieved its planned goals and expected outcomes, including specific contribution to Action objective and deliverables, or publications resulting from the STSM. Agreed plans for future follow-up collaborations shall also be described in this section.

The framework is currently still work in progress, but offers a good foundation for improvements if the problems mentioned above can be mitigated. At the moment, the expected outcomes and results have not been achieved, but we believe that the overall result of the STSM is in a reasonable state as we plan to continue working on this framework and turn the results into a publication.

Furthermore, we are in contact with BoardGameArena to obtain data from human players. Since it does not matter to our framework where the data comes from, we can easily use this method to compare demonstrations from human players with demonstrations from AI players. This addresses point (vii) of the GameTable objectives of working group 1:

“to bring the notion of explainability to AI techniques for game playing, and design algorithms able to efficiently compare the strategies and tactics played by each AI agent to the strategies played by humans;” (from WG1: https://gametable.network/index-wg.html)

We have several ideas for follow up work until the next in-person meeting in London:

First, we need to evaluate whether the framework works for simpler games like Tic-Tac-Toe, as Othello seems too difficult at the moment. We also need to re-evaluate whether the domain-specific language (DSL) is actually able to learn the right features and how it can be improved if necessary. A related issue is how to include the action in the DSL so that state-action features are learned.

If the framework can create reasonable explanations for Tic-Tac-Toe, we think it is also possible to create explanations for other games with a DSL tailored to the specific games. Our main goal would be to make this tailored DSL fully learnable from data using a library learning system, which was originally proposed in the working plan of this STSM application.