Bot Development for Playing Pokémon Battles by Using Data Analysis

Data Science Research Center, Faculty of Science, Chiang Mai University, Chiang Mai, Thailand 50200

Authors : Paratthakon Ainun, Phuri Ounjanum and Manassanan Chuenpiriyanon

Advisors : Dr. Thapanapong Rukkanchanunt and Associate Professor Dr. Jakramate Bootkrajang

FACULTY OF SCIENCE
CHIANG MAI UNIVERSITY

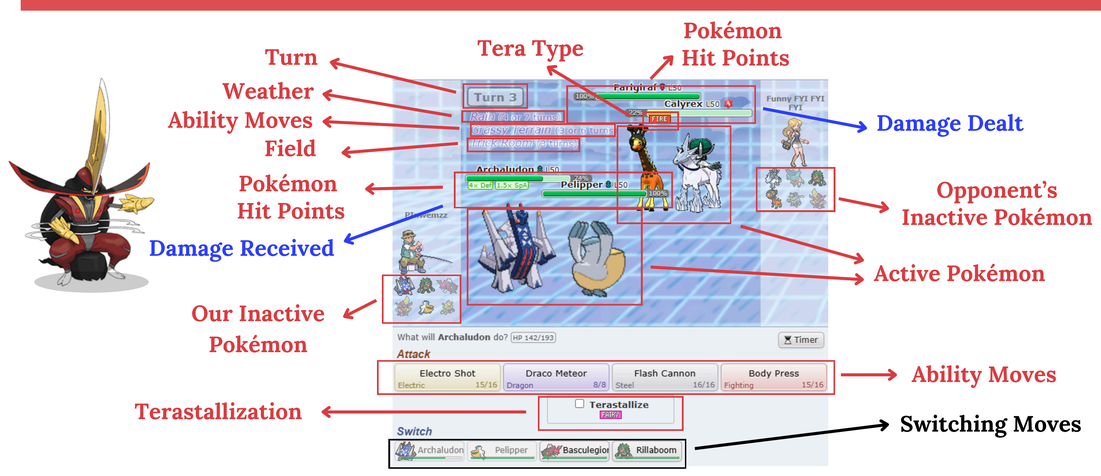
Abstract

This research aims to develop a model using Reinforcement Learning (RL) technique to play Pokémon Battles, which are highly complex, especially under the VGC 2024 Regulation H competition rules. These rules are used in the Pokémon World Championships 2024 and involve a large state space and randomness in various situations that occur during the battle. This research applies a few algorithms from RL to the model, in order to discover the best model that can learn from different environments during the battle and decide to make appropriate decisions in each situation. The model is designed to focus only on choosing actions in the battle phase and the team selection process. The performance test of the model is conducted by comparing it with 3 types of bots: RandomPlayer, which selects actions randomly; MaxdamagePlayer, which chooses the highest damage move; SmartBot, which selects actions according to the set conditions. The evaluation is based on the model's win rate when competing with these three types of bots. The results of this research are expected to help understand the potential of RL in developing the model for Pokémon Battles and can be applied to decision-making systems in other strategic games.

Introduction

Pokémon Scarlet and Violet is a role-playing game, where your role is a Pokémon trainer, and the main purpose of the training is to make your Pokémon stronger and ready for the battle incoming. In the battle, your Pokémon will take action by your order only. This research uses the competitive rules for the battle, which is training the Pokémon is not essential; all Pokémon is auto-leveled to level 50. The research's aim is to make a bot think like a Pokémon trainer. The bot learns through an idea called Reinforcement Learning (RL). RL is designed based on how human and animals learn how to behave to an environment, by receiving some feedback after taking an action to the environment. In the dynamic environment –the same state the bot stays, the same action the bot takes, will not always give the same output– like Pokémon Double Battles environment, this research tries to find whether it is possible that the bot can detect the winning pattern.

Battle Rule



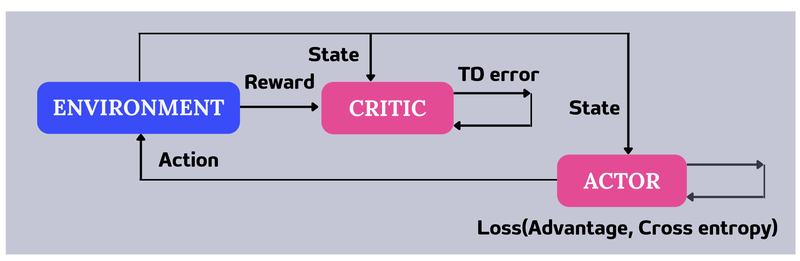
Before the battle begins, players can see all six Pokémon on the opponent's team. Each player then selects four Pokémon to use in battle. At the start of each turn, players choose their actions. The game then determines the order of execution based on the speed of each Pokémon. However, certain actions–such as switching, Terastallization (Tera), and Protect–always occur first. To win the battle, a player must eliminate all four of the opponent's selected Pokémon.

Scope of the Research

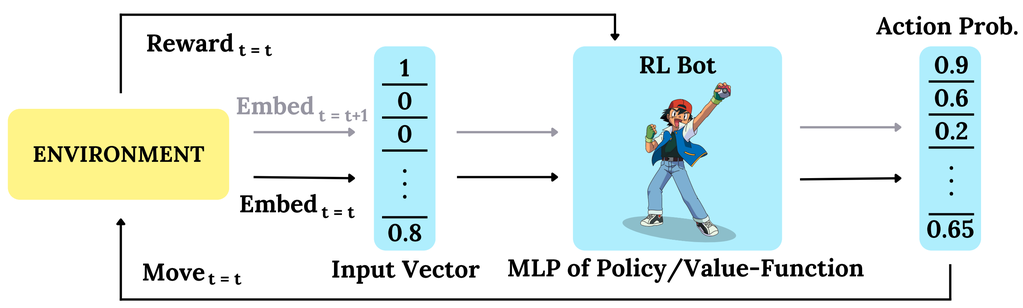
- **Team Selection :** The bot selects Pokémon randomly in the training process to find the best strategy and will use that team throughout the testing.
- **Training Strategy :** The different uses of defining hidden layers, reward and activation functions is employed in the strategy. Adjustment of hyperparameters is also included.
- **Opponents of the RL bot :** Three types of bots is provided for training and evaluation of our RL bot which are : RandomPlayer, the bot chooses action randomly;
  MaxDamagePlayer, the bot chooses the abilities that have the maximum attack damage; and
  SmartBot, the bot's actions follow the specific strategy for its team to get advantage at the first round, and choose the most damage move to attack opponent Pokémon only.

About the A2C Algorithm

The Advantage Actor-Critic (A2C) algorithm takes advantage of both the policy and the estimate of the value function. The actor is responsible in the policy part and the critic in the value- function part. The actor encourages the network to increase the probability of actions lead to higher rewards and minimize its loss. The term "advantage" refers to the advantage function which is applied in the loss. The critic estimates the state-value function, which represents the expected return (total reward) from a given state and following the current policy; the critic's loss follows the mean squared error formula. Its simple architecture can be represented as a picture below.



Learning Method

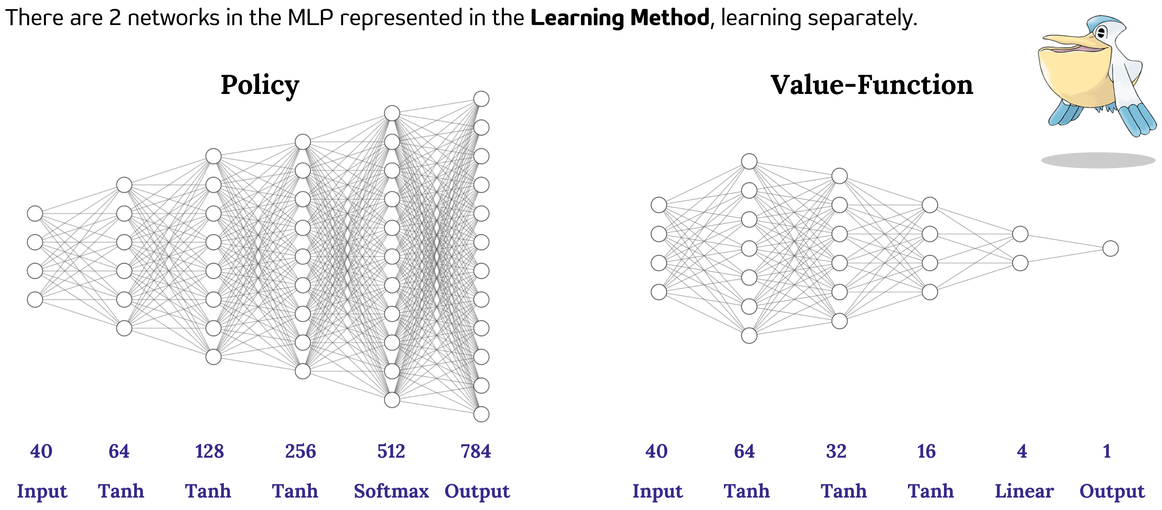


The environment is first converted into a vector representation, which is then processed by a Multilayer Perceptron (MLP) to interpret the game state. The MLP generates a probability distribution over possible actions, and an action is selected based on this distribution. The chosen action is then translated into a corresponding in-game move for that turn.

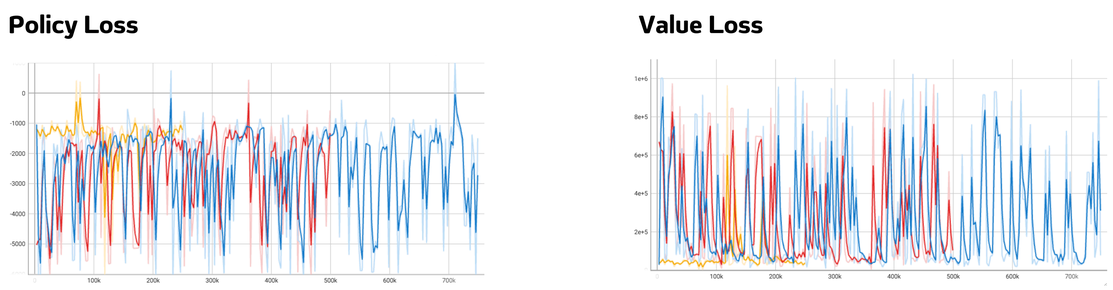
After executing the move, a reward is assigned based on the outcome. This reward is used to adjust the model's policy, helping it learn to make better decisions in future turns. Through continuous training, the model refines its action selection strategy to maximize long-term rewards, improving its decision-making over time.

Policy and Value-Function Network

There are 2 networks in the MLP represented in the **Learning Method**, learning separately.



Result



Win Rate

time-steps train \ opponent	RandomPlayer	MaxDamagePlayer	SmartBot
250,000	88 %	31 %	0 %
500,000	90 %	36 %	2 %
750,000	94 %	42 %	16 %

Conclusion

According to the result, the model can handle the battle against RandomPlayer well but still struggle against Smartbot. The Authors make assumption that the environment design is not effective as it should, for example, the input should also include the Pokémon stats so low number of input dimensions lead the network to learn inefficiently. However, the model demonstrates the ability to learn and adapt by selecting the most effective strategy based on the encountered matchup.

Tools

Pokémon Showdown! BETA · node.js · poke-env · TensorBoard

Reference

[1] Sagar, S., Narayanan, V., Binu, D., Selby, N., and Thomas, S. E. (2023). *Advantage Actor-Critic Reinforcement Learning with Technical Indicators for Stock Trading Decisions.*

DSRC Data Science Research Center, Faculty of Science, Chiang Mai University · CMU Chiang Mai University · Be Fun Fun Functional Fun Damental To the Frontier

Winning Battle Replays

RandomPlayer · MaxDamagePlayer · SmartBot