Deep Learning
Final Project Proposal
Pokémon Battling Agent

Ozaner Hansha

November 11, 2020

Team Members: Ozaner Hansha (oh73)

What to solve: The problem this project will attempt to tackle is that of optimal Pokémon battling. Pokémon is a Japanese game franchise published by Nintendo starting in 1995. A major component of these games is **battling**, in which two players equipped with a team of 6 Pokémon each aim to lower the other opponent's team's health to 0. Battles are turn based, with one Pokémon facing off against another at a time. Each Pokémon has 4 moves which they can use to damage, disable, or otherwise adversely affect their opponent Pokémon. [1]

In the years since Pokémon's initial release, new mechanics, moves, and Pokémon have been introduced with newer released titles in the franchise. This complexity has made Pokémon battling extremely popular. Indeed the competitive scene of Pokémon takes place in Pokémon Showdown [2], an open-source battle simulator that recreates the battling mechanics of the Pokémon game series.

This complexity is an apt testbed for reinforcement learning algorithms, and training data and evaluation metrics are easier to come by due to its popularity. Note that are a variety of different formats (essentially rulesets), which we can battle under. For simplicity we will be attempting to battle using the random gen 8 ruleset. This allows us to forgo the equally difficult task of choosing a team, and focus only on battling.

How to solve: The project will make use of the `poke-env` library [3]. An open source python repository that allows for the creation of battle agents for use/training on Showdown. Our model will be a Deep Q-Network [4], which will be trained in battle against a random agent. The observations our network will make will be consist of a feature vector of pertinent information like the power of the current Pokémon's moves, its HP, the opponent's current Pokémon's HP, status conditions, etc. Rewards will be given for damaging a Pokémon, defeating a Pokémon, and winning the battle. Things that may be changed if feasible include:

- Battling real agents as part of training, much slower but much more informative.
- Including less direct information about the game state in the observation feature vector (e.g. health of switched out Pokémon).
- Including more nuanced rewards (e.g. higher reward for deafeating opponent faster)

Our resulting agent should hopefully be capable of effectively conducting a Pokémon battle.

How to evaluate: We will evaluate our agent by having it compete against 2 other agents:

- A random agent. That is, one whose actions are determined by a uniform probability distribution on possible actions.
- A damage maximization agent. This agent is a simple rule based one that will greedily pick it's action based on whether it deals the most damage.

We will pit all 3 agents against each other and display the win-rate of each match up (row vs column) in a matrix akin to this one:

	RL	Random	Damage
RL	$\approx .5$	?	?
Random	?	$\approx .5$	?
Damage	?	?	$\approx .5$

After a sufficient amount of battles, we'd expect that agents pitted against themselves result in a near 50-50 winrate. The reason we even include these matchups is as a sanity check, and as a marker that the other results are accurate and not just fluctuations.

There has been work on this problem, primarily by undergraduates, using other methods and so a comparison of that work may be in order as well [5] [6].

References

[1] Bulbapedia
https://bulbapedia.bulbagarden.net/wiki/Pokémon_battle

[2] Smogon
pokemon-showdown.
https://github.com/smogon/pokemon-showdown

[3] Haris Sahovic, et al.
poke-env.
https://github.com/hsahovic/poke-env

[4] Mnih, Kavukcuoglu, et al.
Human-level control through deep reinforcement learning
https://web.stanford.edu/class/psych209/Readings/MnihEtAlHassibis15NatureControlDeepRL.pdf

[5] Huang, Lee.
A Self-Play Policy Optimization Approach to Battling Pokemon
https://ieee-cog.org/2019/papers/paper_175.pdf

[6] Rill-García
Reinforcement Learning for a Turn-Based Small Scale Attrition Game
http://ccc.inaoep.mx/ esucar/Clases-mgp/Proyectos/2018/reinforcement-learning-turn%20%281%29.pdf