Reinforcement Learning for Simple Spelling Correction

Ainslee Archibald

Pitzer College

May 6, 2025

Problem Statement

- Goal: Train an RL agent to correct a scrambled 5-letter target word ("hello").
- Task: Starting from a slightly misspelled version, the agent must learn a sequence of actions to reach the correct spelling.
- Motivation: A simplified environment to explore fundamental RL concepts in a discrete action and observation space.

Environment Specifications

- Target Word: "hello" (fixed).
- Observation Space: A vector of 5 integers, each representing a letter's position in the alphabet (1-26).
- Action Space: 10 discrete actions:
  - Actions 0-4: Decrement the letter at the corresponding position.
  - Actions 5-9: Increment the letter at the corresponding position.
- Choice Justification:
  - Discrete and bounded spaces simplify the learning task for initial exploration.
  - Direct manipulation of letter positions provides a clear and interpretable action space.

Environment Dynamics

Observation: Current 5-letter word state (e.g., $[8, 5, 12, 12, 1]$ for "hello").
Actions: Selecting an action modifies one letter in the current word by incrementing or decrementing its alphabetical position (with wrap-around, e.g., 'a' decrements to 'z').
Reward Function:

- $+10$ for reaching the target word "hello".
- $+1$ if the current distance is less than the previous step's distance
- -0.2 if the current distance is farther than the previous step's distance
- -0.01 penalty per step.

Termination Conditions:

- Episode terminates successfully when the agent spells "hello".
- Episode also terminates if a maximum of 5 steps is reached.

Environment Demo (On the Board)

Learning Algorithm: Proximal Policy Optimization (PPO)

- An actor-critic policy gradient algorithm.
- **Actor:** Learns a policy $\pi(a|s)$ that maps states to probability distributions over actions.
- **Critic:** Learns a value function $V(s)$ that estimates the expected future reward from a given state.
- PPO uses a clipped surrogate objective to ensure stable policy updates.
- This helps prevent large policy changes that could destabilize learning.
- The algorithm balances exploration (trying new actions) and exploitation (taking actions that have worked well in the past).

Results

Future Directions

- **More Complex Target Words:** Increasing the length and complexity of the target word.
- **Larger Action Space:** Allowing for actions like swapping letters or inserting/deleting (which would require a different environment and potentially a different algorithm).
- **Dynamic Scrambling:** Introducing more varied and challenging initial scrambled words.
- **Curriculum Learning:** Gradually increasing the difficulty of the scrambling over training.
- **Generalization:** Training on multiple target words and evaluating the agent's ability to learn to spell new words.