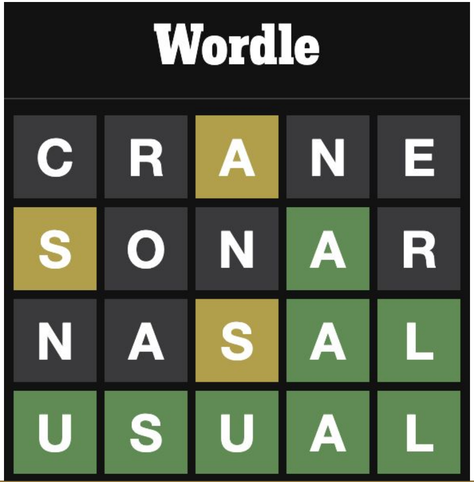
Attempting to Solve Wordle Through the Use of Artificial Intelligence Processes With Varying Reward Functions

Joseph Kwilman, Michael Ruberto
December 15, 2022

Introduction

- Aimed to create an agent to optimally solve a game of Wordle
- Rules of Wordle
  - A random five letter English word is selected. The agent does not know the word.
  - The agent has six attempts to guess the word
  - After each guess, every letter is marked either GRAY (Not in word), YELLOW (In word, different position), or GREEN (Correct position)



Background

- Wordle, like all guessing games, involves a degree of randomness
    - Being in one state and guessing a certain word won't always lead you to the same state
- What AI algorithms that we've studied could be applied to this type of problem:
    - Markov Decision Processes (MDPs)
        - Well suited for modeling states and actions with stochastic transitions
    - Hidden Markov Models (HMMs)
        - Can handle probability distributions for possible answers that can change as the game progresses

Approach – Fewest Words Agent

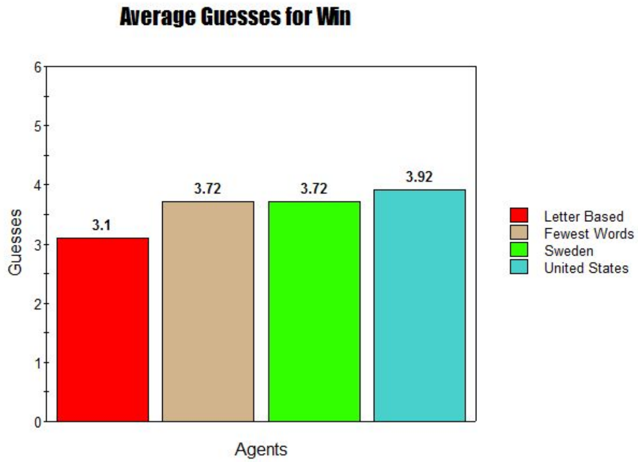
- MDP Based
  - State: (# Guesses Remaining, List of Possible Remaining Solutions)
  - Action: Guess a Word
  - Gamma: 0.9
  - Reward Function: # Total Valid Words - # Possible Remaining Solutions
    - Incentivizes agent to eliminate as many words as it can
  - Transition Function
    - Each action leads to at most 243 next states
      - 5 letter guess, 3 possible colors for each letter. $3^5 = 243$
    - P(State) = # Remaining Words in Specific State / Total Number of Words Across All States

Approach – Letter Based Agent

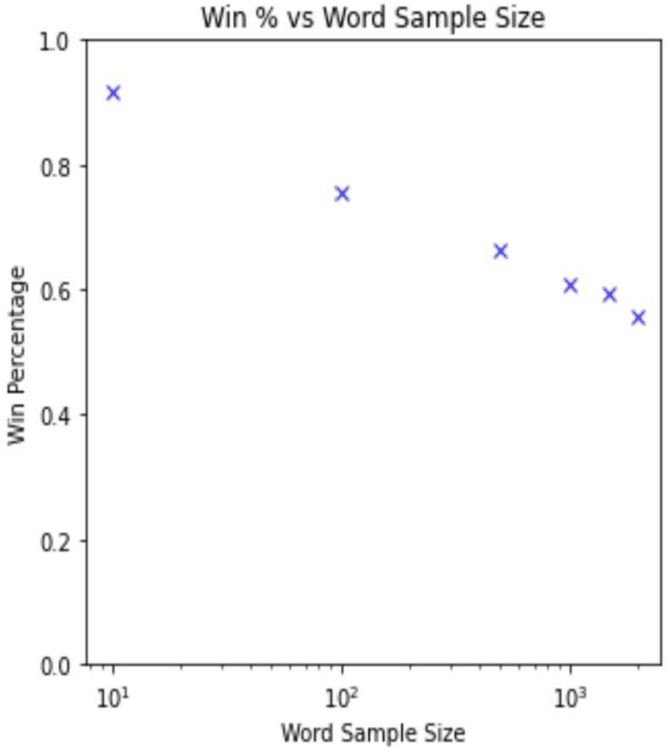
- Inspired by Forward Filtering in Hidden Markov Models
- Probability distribution of every possible word
- Updated at each observation
  - Bad Letters: letters known to not be in the word
  - Known Letters: letters known to be in the word
  - Confirmed Letters: $[x_0, x_1, x_2, x_3, x_4]$ for confirmed letters and their position
- $P(\textit{word} \mid \textit{all observations}) \propto P(\textit{new observation} \mid \textit{word})\, P(\textit{word} \mid \textit{previous observations})$
- Normalize the distribution
- Next guess is whatever word is most likely to be the solution

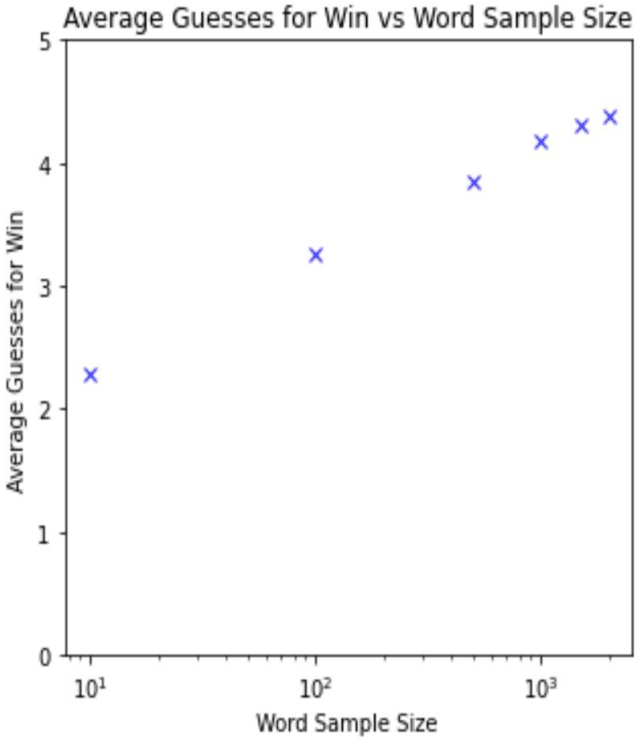
Results - Win Rate and Average Score

- Had to randomly sample 100 words due to Fewest Words Agent's run time
- Fewest Words Agent displayed a superior Win Rate (100% vs 76%)
- Letter Based Agent boasted a better Average Score (3.1 Guesses vs 3.72 Guesses)
- Both Agents performed better than the average American (3.92 Guesses)
- Both Agents tied/beat the Average Score of the best country (Sweden, 3.72 Guesses)



Results - Assess Effectiveness of Letter Based Agent Against Solution Sample Size





Conclusion

- Both Agents are capable of playing Wordle at a high level
  - Fewest Words Agent slightly sacrifices performance for the sake of victory certainty
  - Letter Based Agent plays a bit riskier in exchange for a slightly higher score
  - Both can play the game at a human level
- Increasing the size of the initial set of words predictably impacts performance, but even with the full word list, the agents perform reasonably well

Thank You!

Any Questions?