CPSC 481

Professor McCarthy

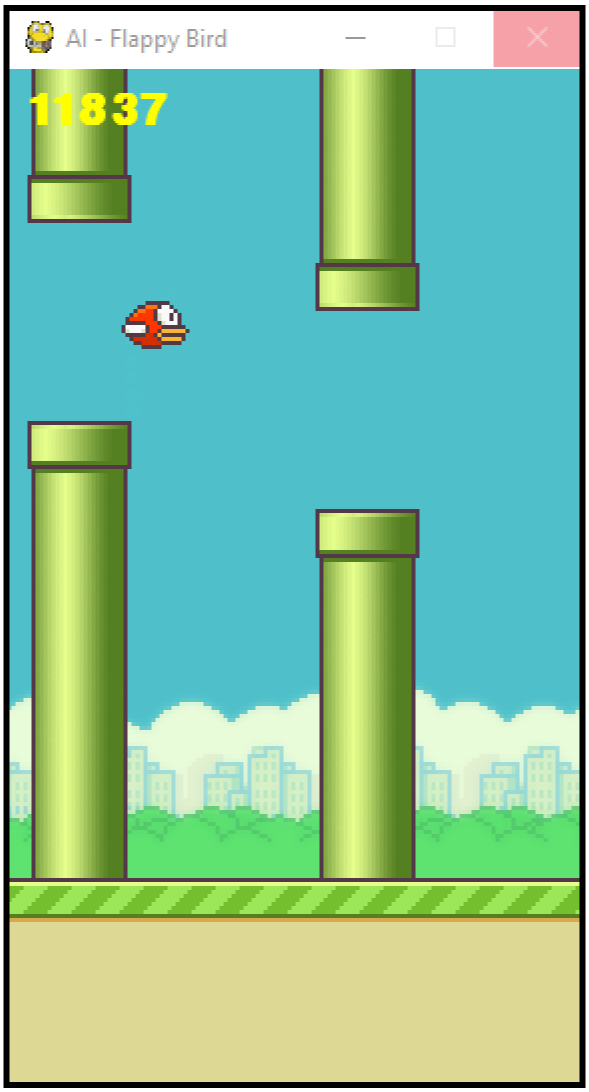
12/18/19

Final Project

Flappy Bird AI

Group Members:

- Kunal Amos Matthews - amos-m2345@csu.fullerton.edu
- Erik Lipsky - eriklipsky@csu.fullerton.edu
- Allen Mrazek - amrazek@csu.fullerton.edu
- Mohammed Kasim Panjri - kasimp@csu.fullerton.edu



Repository: https://github.com/kasim95/Deep_Learning-Flappy_Bird

A Brief History

Flappy Bird isn't just a game, it was a mobile phenomenon. The game was released in 2013 and become the Top Free App of 2014. This game uses familiar 'Mario' theme graphics with a simple goal: to get as far as possible without dying. To play the game, the bird continuously moves right and is pulled down by gravity. The player must tap the screen to make the bird jump and avoid the pipes.

Despite all the mixed success, the game was taken down from Apple's App Store and Google Play in just over a year, several listings appeared shortly after on eBay for the pre-installed game for thousands of dollars. The real reason as to why it was taken down is unknown, but it is likely because it shared too many similarities between Nintendo's *Mario* games.

According to Recordsetter.com, the highest score achieved was 1,940 set by Ben King. Our goal was to beat this record and we did just that. Our model was able to achieve a score of 11,837.

Q Learning Model

Our first attempt to solve the problem used a Deep-Q network based on previous work by DeepMind. A similar model has been successfully employed by DeepMind in 2013 to play a variety of Atari games with no changes to the network architecture or hyperparameters (Mnih). This network took as input a stack of 4 80 by 80 images and learned to play a selection of games at human-level or above. Unfortunately, we were unable to reproduce their success using our own model, even after employing a variety of later techniques designed to improve the efficiency, convergence speed, and stability of deep-Q networks. These included gradient clipping to avoid gradient explosion, normalizing input data, reducing the training rate, employing a pair of networks to aid in network q-value stability, and various other hyperparameter modifications which had little effect. Even after several days of training, and a variety of tweaks to the random moves the network makes to explore better solution, this model was able to achieve only a very poor outcome. The failure of our network to converge ultimately led us to seek an alternate solution.

Regression Model

Careful examination of the previous failures indicated that the convolutional layers of our networks were the cause of its instability. Our best solution removed these layers entirely, replacing them with a dense regression network. This network takes four inputs: bird position on the screen, bird vertical velocity, distance to the next pipe, and height of the pipe. It was trained on several hundred thousand frames of the game. Within several dozen epochs, it could already produce reasonable results. Even with the game's difficulty turned up, within several hours it was capable of playing games consisting of hundreds of pipe traversals. Further adjustments of hyperparameters, and in particular normalizing input data to a small range, resulted in stellar results. After training overnight, we achieved a score of 11,837 with this model until it was manually stopped. The model can, in theory, play the game indefinitely.

Works Cited

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D. & Riedmiller, M. (2013). Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, .