May the Flop be with you - Deep RL based Texas Hold'em PokerBot

Srivatsan Srinivasan, Sebastien Baur, Donghun Lee

CS281-Advanced Machine Learning, SEAS, Harvard University

Objective

To build a self-trained intelligent Texas Hold'em PokerBot that is capable of

- Learning hand strength using neural nets(NN).
- Choosing appropriate training experiences.
- Evolving its game via NN fictitious self-play.
- Beating baseline and advanced AI opponents.

Key Concepts

Convolutional and Fully-Connected Neural Nets, MDP, Double Deep-Q Networks, Policy Networks, Prioritized Experience Replay, Nash Equilibrium, Neural Fictitious Self-Play.

Introduction

No-Limit Texas Hold'em Poker presents an interesting template for probabilistic modeling and deep RL as it involves sequential decisions, imperfect information, adversarial behavior and humongous state action space where simulating possible cases for supervised learning is prohibitive. Hence, we use fictitious self-play [1]., which is proven to achieve near-optimal Nash Equilibrium, using Double DQNs and policy networks with prioritized experience replay buffers so that agent plays against its own clones to learn optimal behavior. Also, we employ state space abstraction through a NN hidden feature layer that infers the win probability of given hand and board.

Problem Definitions

- Hand(H) - 13 * 4
- Plays(P) - Actions in four rounds(PF,F,T,R).
- Board(B) - 4(PF,F,T,R) * 13 * 4
- Game Variables(G) - Pot, Dealer, Stacks
- State - (H,B,G,P)
- Action - (Call, Fold, Check, All-In, Raise-Bin).
- Rewards - Δ ($) with each action.
- Setup - Initial : $100 each, SB : $1, BB : $2

Card Featurizer

A model that implictly represents **hand strength** (Prob(Win) with current cards and board). State-Space Abstraction - Eg: Hands (2H, 3S) and (4H,5S) with a board of (KH,KD,KC) are treated the same.

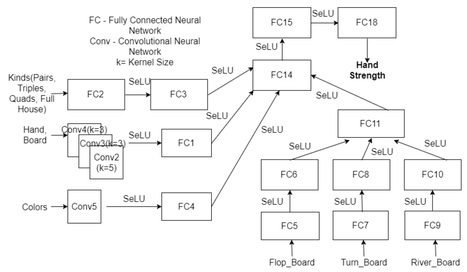

Figure 1: Card Featurizer Network

Model Architecture

Here, we have a shared network architecture that predicts Q-values and π values using card features and game states.

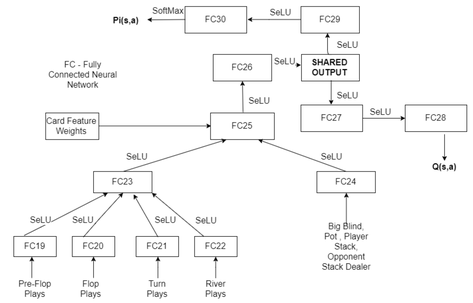

Figure 2: π and Q network architecture

Important Result

Hand/Board State abstraction using Neural Network is very effective in deep RL for poker. NFSP is a viable AI solution in imperfect information games that could learn purely through continuous self-play.

Neural Fictitious Self-Play(NFSP)

Init : Game transitions memory(using PER) and best-response memory M_{RL}, M_{SL} ; Q-network, π network. Repeat the following until termination.

- Choose action with η probability from β or π.
- Populate M_{RL} using each (s,a,r,s,a') and M_{SL} using (s,a) when action is chosen from β.
- Train Q-network(off-policy RL) to yield ϵ-greedy strategy β and average policy network - π learned from prior best responses.

Why choose from both networks?

Classic Off-Policy Learning would involve playing π and learning Q. This would limit the experiences that π is trained on. Here, We train π on experiences generated from our best-response behavior β while also doing off-policy Q-Learning. We sample our next action from both networks probabilistically in order to create a mixture of exploration and exploitation.

Model Results

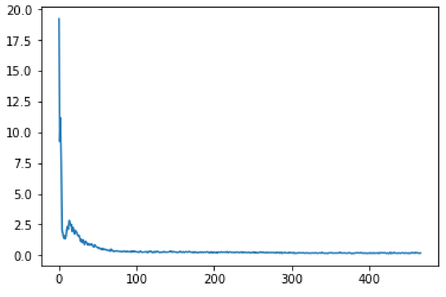

Figure 3: Test loss on card featurizer

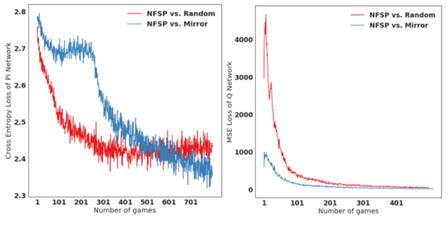

Figure 4: MSE Loss on Q and π Networks

Experiments

- NFSP vs Random
- NFSP vs Mirror
- NFSP vs DDQN(WIP)
- NFSP vs NFSP variants(WIP)

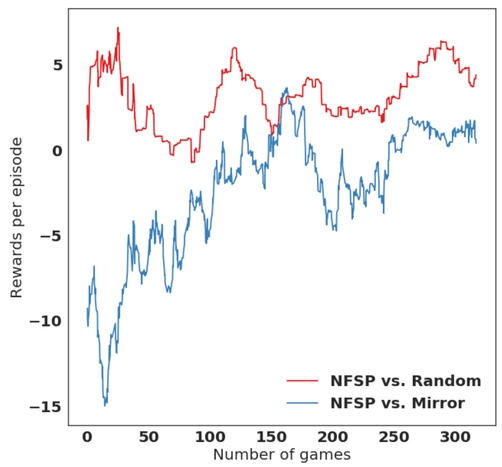

Figure 5: Experiment Results - Rewards

Observed Shortcomings

- Games against random agent mostly end up as folds due to random opponent actions for several games - Need to incorporate opponent game style.
- Rank based ER buffer does not sample extreme actions(showdowns,all-ins) that frequently - Need to convert it to proportionality based ER.

Future Work

- Hyperparameter tuning and initialization.
- Opponent Modeling and Game Theory ideas.
- Simulations against commercial AI.

Key Reference

[1] J. Einrich and D. Silver.
Deep reinforcement learning from self-play in imperfect-information games.
arXiv, (1603.01121v2).