



HAL
open science

Performing Deep Recurrent Double Q-Learning for Atari Games

Felipe Moreno-Vera

► **To cite this version:**

Felipe Moreno-Vera. Performing Deep Recurrent Double Q-Learning for Atari Games. International Conference on Machine Learning, LatinX in AI Workshop, Jun 2019, Long Beach, United States. hal-02217800v1

HAL Id: hal-02217800

<https://hal.science/hal-02217800v1>

Submitted on 1 Aug 2019 (v1), last revised 8 Aug 2019 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Performing Deep Recurrent Double Q-Learning for Atari Games

Felipe Moreno-Vera

Universidad Católica San Pablo, Arequipa - Perú.

Abstract

Currently, many applications in Machine Learning are based on define new models to extract more information about data, In this case Deep Reinforcement Learning with the most common application in video games like Atari, Mario, and others causes an impact in how to computers can learning by himself with only information called rewards obtained from any action. There is a lot of algorithms modeled and implemented based on Deep Recurrent Q-Learning proposed by DeepMind used in AlphaZero and Go. In this document, We proposed Deep Recurrent Double Q-Learning that is an implementation of Deep Reinforcement Learning using Double Q-Learning algorithms and Recurrent Networks like LSTM and DRQN.

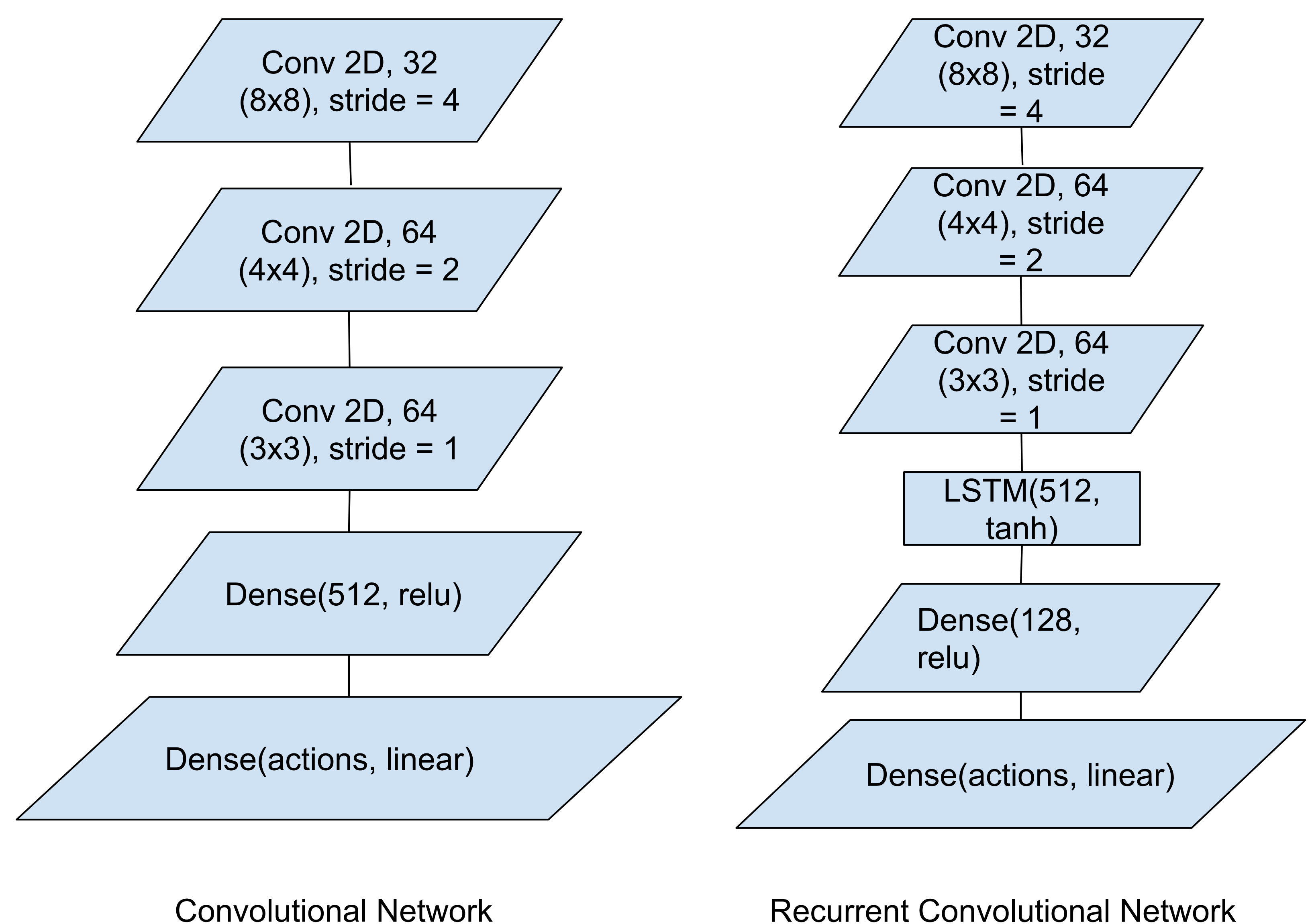
Introduction

Currently, there is an increase the number of application in Reinforcement Learning, specially in Deep Reinforcement Learning with new techniques. One of application of DRL (Deep Reinforcement Learning) is in Games like AlphaZero (Go, Chess, etc) and video games like Mario, Top racer, Atari, etc. Deep Reinforcement Learning is considered like a third model in Machine Learning (with Supervised Learning and Unsupervised Learning) with a different learning model and architecture.

There are several methods of implementing these learning processes, where Q-Learning is a prominent algorithm, the Q value of a pair (state, action) contains the sum of all these possible rewards. The problem is that this sum could be infinite in case there is no terminal state to reach and, in addition, we may not want to give the same weight to immediate rewards as to future rewards, in which case use is made of what is called an accumulated reinforcement with discount: future rewards are multiplied by a factor $\gamma \in [0, 1]$ so that the higher this factor, the more influence future rewards have on the Q value of the pair analyzed. Formally: DQN, DRQN, DDQN.

Methods

We implement the CNN proposed by Chen et al. with some variations in the last layers and using ADAM error. The first attempt was a simple CNN with 3 Conv 2D layers, with the Q-Learning algorithm, we obtain a slow learning process for easy games like SpaceInvaders or Pong and very low accuracy in complicated games like Beam Rider or Enduro. Then, we try modifying using Dense 512 and 128 networks at last layer with linear activation and relu, adding a LSTM layer with activation tanh.



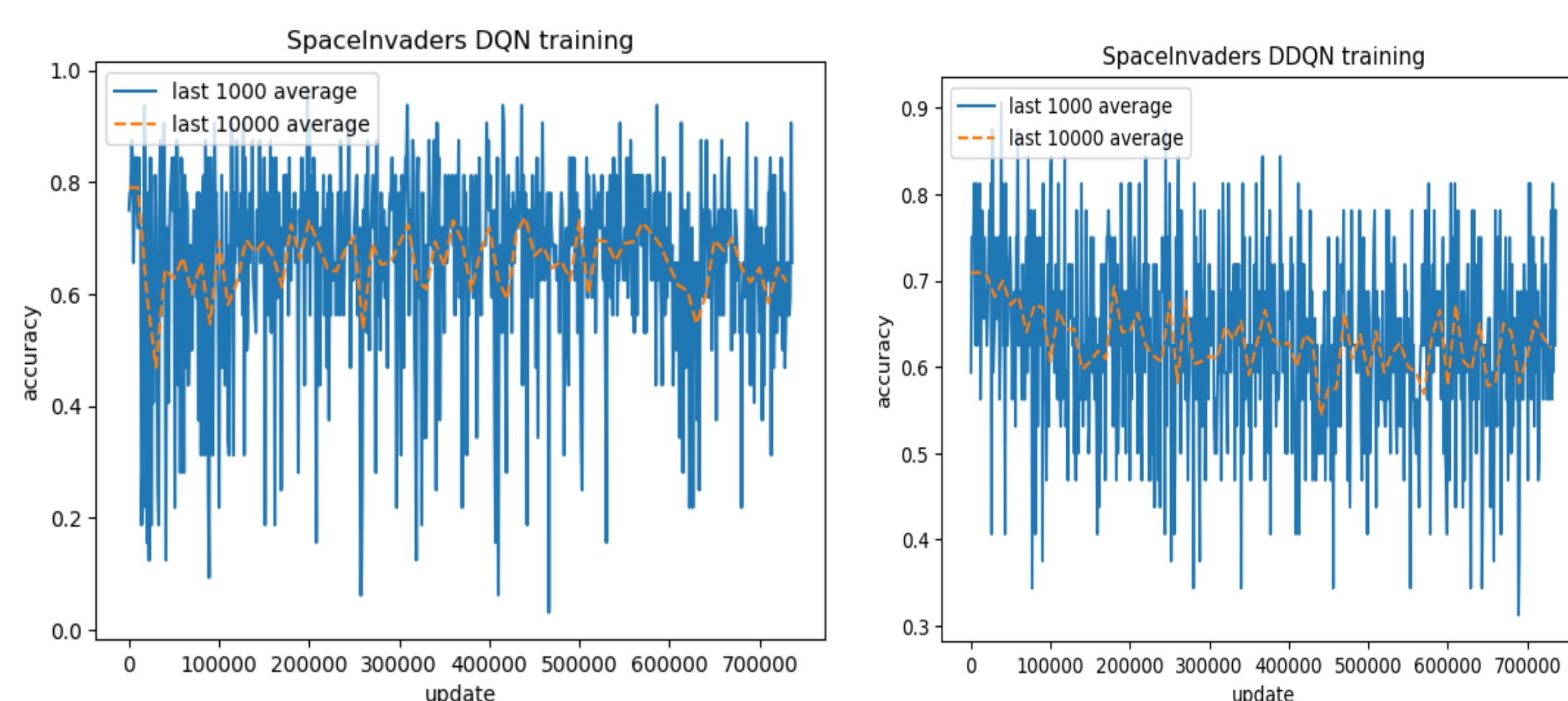
Tables

Models and respective Scores

Model	SpaceInvaders	Enduro	Pong	Beam Rider
DQN	1450	1095	65	349
DRQN	1680	885	39	594
DDQN	2230	1283	44	167
DRDQN	2450	1698	74	876

Results

Our experiments is build over Atari Learning Enviroment (ALE) which serve us as an evaluation platform for our algorithm and allow us to compare with DQN, DDQN and DRQN. After to run our algorithms using 10M (10 millions) episodes, we obtain results for each model in each respective game. We get best scores for the 4 games mentioned above (SpaceInvaders, Enduro, Beam Rider and Pong).



Conclusiones

We Present a model based on DRQN and Double Q-Learning combined to get a better performance in some games, using LSTM and CNN to analyze frames. We notice that each method could be good for an specific Atari game and other similar games but not for all. but can be improved using different CNN and get more information from the frames in each batch iteration.