

Deep Reinforcement Learning: Robotic Arm Control

Shane Reynolds

February 28, 2019

Abstract

1 Introduction

Prior to 2010, state of the art speech to text was achieved by taking an audio sample input which was passed through various independently developed models, such as acoustic models, and language models, which converted the input to text. In 2010, Deep Neural Networks (DNN) started to make improvements on this approach. The DNN approach simply took the raw audio input and returned the text output, replacing the various acoustic, and language models with single network. Speech recognition with DNNs is now considered state of the art. Prior to 2012, the Computer Vision task of taking raw pixel input and labelling the picture as an output was achieved by extracting key points, using SIFT feature computation, and applying deformable part models. In 2012, ImageNet and AlexNet demonstrated performance which surpassed the current state of the art, effectively replacing the image labelling process with a single DNN. Again, prior to 2014, text-to-text machine translation was undertaken with a series of individual models and computation processes. Today, state of the art machine translation is achieved with a single DNN. The common theme in these examples is that input to output transformation, which were comprised of many individual models, been replaced by a single DNN yielding significant increases in system performance.

The field of robotics presents a similar story: robotic sensor input is processed, and actions in the physical world are the system output. Currently, state-of-the-art approaches to achieving this sensor input to action output transformation is done using perception models to localise the robot and map the environment, navigation models to path plan, and control models to move actuators. Thinking of a robotic system in this context presents a compelling argument for trying to replace our approach to perception, navigation, and control using a single DNN. Simple analogies aside, it must be acknowledged that robotics is more challenging than labelling an image. The main reason is that the robot's actions change the nature of the robot's environment. The implication is that we cannot collect static data to learn actions from, rather, the data set needs to be built by the robot exploring and interacting with the environment. One of the best ways that we currently know how to approach this problem is through reinforcement learning.

TALK ABOUT REINFORCEMENT LEARNING

This paper explores deep reinforcement learning (DRL) and its application to a three degree of freedom robotic arm. The reinforcement learning agent is trained to touch a cylindrical object which is spawned in the robot's workspace. The goal comes in two flavours: the first is to touch the cylindrical object with any part of the robot; the second, more difficult, goal is to touch the cylindrical object with the robot arm end effector. Learning takes place over repeated episodes, which are terminated when the robot successfully touches the object, or when the robot arm collides with the ground. A final reward is given when the agent reaches a terminal state. An interim rewards is assigned at each discrete time step. The interim reward function design is the main focus of the paper. The benchmark for

2 Background

Reinforcement Learning (RL) is a branch of machine learning that is concerned with how agents make sequential decisions to maximise some notion of a cumulative reward. Suppose that a robotic agent exists in some environment which is comprised of many discrete states, $s \in S$, such that S denotes the state space. At any discrete point in time the agent can take an action $a \in A$, where A denotes the action space. When the agent takes an action in a given state, the agent receives some reward, denoted with r . If an agent is in a given state, s , and takes an action, a , this will transition the agent to a new state, s' , and yield reward, r , with some given probability. The set of probabilities which govern the environmental dynamics are referred to as transition probabilities. We denote them as follows:

$$P(S_{t+1} = s', R_{t+1} = r | S_t = s, A_t = a) \quad (1)$$

3 Robotic Arm Reinforcement Learning Problem

WHAT IS THE DEFINITION OF THE PROBLEM?

HOW IS THE END OF AN EPISODE OR THE TERMINAL STATES OF THE EPISODE DETERMINED?

WHAT IS THE ARCHITECTURE OF THE NEURAL NETWORK THAT IS BEING USED TO DELIVER THE DRL

4 Robot Arm Simulation

TALK ABOUT THE SIMULATION ENVIRONMENT AND THE GEOMETRICAL CONFIGURATION OF THE ROBOT - KINEMATICS, MOVEMENT

TALK ABOUT OTHER ELEMENTS OF CONFIGURATION THAT ARE USED WITH THE ROBOT - FOR EXAMPLE THE CAMERA

SET UP OF THE DQN AGENT

CONTROL OF THE ROBOT USING JOINT POSITION OR JOINT VELOCITY - WHAT IS THE DIFFERENCE BETWEEN THE TWO? SHOULD THIS BE INCLUDED IN THE DISCUSSION?

5 Collisions

CHECKING OF THE COLLISION BETWEEN THE OBJECT AND THE ROBOT, AND CHECKING THE COLLISION BETWEEN THE ROBOT AND THE GROUND PLANE

6 Reward Function

A SINGLE REWARD IS DEFINED, HOWEVER, THE COLLISION RULES ARE CHANGED WHICH ALTERS THE INTERIM REWARD THAT THE AGENT RECEIVES THROUGHOUT THE EPISODE

7 Hyperparameters

TALK ABOUT THE HYPERPARAMETERS USED FOR EXAMPLE THE CHANGES TO IMAGE DISCRETISATION THAT WAS USED IS 64 BY 64 TO ENSURE THAT MEMORY RESOURCES WERE NOT EXCEEDED.

8 Results

TALK ABOUT HOW LONG IT TOOK FOR THE ROBOT TO REACH THE DESIRED SOLUTION SPACE FOR EACH OF THE TASKS

INCLUDE SCREENSHOTS AND VARIOUS VIDEOS OF EACH OF THE

9 Discussion

TALK ABOUT THE FACT THAT THE SUCCESS OF THE REINFORCEMENT LEARNING TASK LARGELY COMES DOWN TO THE DETERMINATION OF A REWARD FUNCTION THAT ALLOWS THE ROBOT TO CONVERGE ON THE DESIRED BEHAVIOUR.

10 Future Work

TALK ABOUT THE FACT THAT THERE ARE ADDITIONAL CAMERAS AND THAT THE OBJECT COULD BE RANDOMLY PLACED THROUGHOUT THE ENVIRONMENT AND THAT ROBOT BASE COULD BE MOVED ALSO TO TRY TO GET THE ROBOT TO LEARN MORE EXOTIC POLICIES