Paper summary: DAQN: deep autoencoder and q-network

Marko Guberina 19970821-1256

date

1 Idea in few sentances

2 Explanation of the central concept

3 Methodology

4 Initial rambly notes

Has 19 citations so kinda lame. The papers that cited it have even lower citations. But it seems like it's exactly what we're doing now and it's good to see what a run-of-the-mill paper in the field looks like. The paper is literred with bad grammar at worst and a lot of plainly bad language at best.

4.1 Abstract

Just running RL on real robotics entails requiring more samples and dealing with noise. Sticking a "generative model" (an autoencoder in this case (how is ae generative???)) to initialize the network should reduce the number of trials. Learning was 2.5 times faster on a task on a real robot.

4.2 Introduction

Autoencoders can be pretrained when taking random actions. That's due to the fact that they work well for unsupervised learning. There are some studies which use convolutional autoencoders to pretrain convolutional layers for general supervised learning tasks.

The hypothesis is that pretrainining an autoencoder will yield better performance as it is much easier to train an autoencoder on just observations in an unsupervised manner than it is to collect rewards and optimal actions and use a loss generated on them. In short, unsupervised learning is easier than plain RL so let's leverage that to boost RL.

4.3 Method

The method consists of the following steps:

1. train a deep autoencoder using unsupervised learning
2. delete the decoder layers and add a fully-connected layer on top of encoder layers
3. use that network as a starting point for a policiy train with deep q-learning

In math notation we have:

1. let $\mathcal{X} = \mathbb{R}^n, \mathcal{Y} = \mathbb{Y}^n, \boldsymbol{x} \in \mathcal{X}$
2. encoder: $\phi : \mathcal{X} \to \mathcal{Y}$
3. decoder: $\psi : \mathcal{Y} \to \mathcal{X}$
4. training goal: $\phi^*, \psi^* = \text{argmin}_{\phi,\psi} ||\boldsymbol{x} \circ (\psi + \phi)\boldsymbol{x}||^2$

The samples for the autoencoder are generated by taking random actions in the environment. Once it's been trained, the decoder is removed, randomly initialized FC layers are added to the encoder and this betwork becomes the Q-network train in the standard Q-learning fashion. Namely, the update function of Q is:

$$Q(\boldsymbol{s}_t, \boldsymbol{a}_t) \leftarrow Q(\boldsymbol{s}_t, \boldsymbol{a}_t) + \alpha \left(r_{t+1} + \gamma \max_{\boldsymbol{a}} Q(\boldsymbol{s}_{t+1}, \boldsymbol{a}_t) - Q(\boldsymbol{s}_t, \boldsymbol{a}_t) \right) \tag{1}$$

wher $\max_{\boldsymbol{a}} Q(\boldsymbol{s}_{t+1}, \boldsymbol{a}_t)$ is the maximum estimate action-value for state $\boldsymbol{s}_{t+1}$. The loss of the deep q-network is:

$$L(\theta) = E_{(\boldsymbol{s}_t, \boldsymbol{a}_t, r_{t+1}, \boldsymbol{s}_{t+1}) \sim \mathcal{D}} \left[(y - Q(\boldsymbol{s}_t, \boldsymbol{a}_t; \theta))^2 \right] \tag{2}$$

$$y = \begin{cases} r_{t+1} & \text{terminal} \\ r_{t+1} + \gamma \max_{\boldsymbol{a}} Q(\boldsymbol{s}_{t+1}, \boldsymbol{a}; \theta^-) & \text{non terminal} \end{cases} \tag{3}$$

4.4 Other stuff

4.4.1 Results

Are totally trash. They got practically no benefits when compared to plain deep Q-learning. The time they got a better result was for the "paper-rock-scissors" game in which the autoencoder was actually a classifier. They didn't even use human-baseline scores as a comparisson, but winning ratios in the game, which is very stupid metric because it's quite imprecise when comparing well-performing agents.

Also, how the heck did you choose paper-rock-scissors as a game!?!??!?!? The mixed Nash equilibrium, i.e. the game theoretic solution is to literally play uniformly randomly!! So i guess the agent makes its move after the being presented with an image. But then that's not the paper-rock-scissors game everyone

knows! Absolutely horrible. Who published this? Even more importantly, how did this get 16 citations?!?!?!?

The paper belongs to the traaaaash.