Deep Transfer for Model-Free Reinforcement Learning Using Autonomous Intertask Mappings and Q-Learning

Anonymous Author(s)
Affiliation
Address
`email`

Abstract

In this paper...

1 Introduction

Transfer Learning (TL) within the Reinforcement Learning (RL) domain can be described as leveraging mastery of one task (a source task) to improve learning speed or asymptotic performance in another task (a target task). The solution to achieving effective transfer depends on the differences between the two task goals, the environments the tasks are posed in, and the agents trying to solve them. The particular problem of deep transfer is the realm of transfer problems where the two tasks may differ in all three of these dimensions.

Reinforcement learning in a realistic setting usually does not allow for either *a priori* knowledge of the environment dynamics or *a priori* knowledge of the relationship between observations, actions, and observed rewards; usually agents must rely solely on experience when learning how to master an RL problem. The method of transfer in RL utilizing actual recorded source task experience is known as instance transfer, and is especially difficult in the deep transfer space. Deep instance transfer can be achieved using a linear or non-linear mapping from the source experience space to the target experience space, and for fully autonomous intelligent transfer to occur this mapping must be learned by the target task agent independent of any explicit human mapping. With an inter-task mapping an agent can translate source task experience into pseudo target task experience and use these pseudo experiences as initial samples in a sample-based RL algorithm such as fitted Q-learning.

2 Reinforcement Learning

3 Transfer Learning for Reinforcement Learning

[introduce TL broadly]

Instance Transfer

Shallow Instance Transfer

Deep Instance Transfer

Submitted to 31st Conference on Neural Information Processing Systems (NIPS 2017). Do not distribute.

4 Restricted Boltzmann Machines

4.1 Bipartite RBMs

4.2 High-Order RBMs

5 Related Work

5.1 Early Work

*See Taylor review - Early Taylor

5.2 Contrasting Methods

Gupta Progressive Nets *See newer review

5.3 Taylor's MASTER Algorithm

[explain method and why it works] [explain limitations, motivating Ammar's work]

5.4 Ammar's TrRBM Method

The main contribution of this paper is to build off the dissertation work of Ammar (CITE) in which he uses a high-order Restricted Boltzmann Machine to learn intertask mappings for instance transfer. Ammar?s method uses a 3-way RBM with one layer for each task?s concatenated instance tuple vectors (s,a,s?) and one hidden layer to modulate interaction between the two visible layers. The visible layer nodes $\mathbf{v}_1, \mathbf{v}_2$ are modelled as Gaussian random variables, $v_1^{(i)} \sim \mathcal{N}(\mu^{(i)}, \sigma), v_2^{(j)} \sim \mathcal{N}(\mu^{(j)}, \sigma)$ and the hidden layer nodes $\mathbf{h}$ take the value of sigmoidal activations. Ammar gives two formulations for the 3-way RBM; the full Transfer Restricted Boltzmann Machine (TrRBM) version with a 3-way weight tensor having elements $\mathcal{W}_{ijk}$, and a factored ?fTrRBM? (Factored Transfer Restricted Boltzmann Machine) version in which the 3-way weight tensor is factored into the product of 3 layer-specific matrixes, . The factored version is motivated be a need to reduce computational complexity from the full version?s $O(n^3)$ to a more manageable complexity of $O(n^2)$. [FIGURE for fTrRBM] [Talk about why the TrRBM can learn a good mapping] [Talk about learning in the TrRBM model i.e. mean of gaussians] [Motivate the extensions i.e. black-box model is unrealistic, sampling method is unrealistic]

6 New Extensions

This section is for showing our formalisms for the extensions we are making

6.1 TrRBM for a Model-Free Setting

6.1.1 Using Source Q-Values Instead of Black-Box Target Task Rewards

6.1.2 Transferring Best- and Worst- Policy Instances

6.1.3 More Realistic Initial Sampling

7 Experimental Setup

7.1 Environments

2D Mountain Cart

3D Mountain Cart

2D Cartpole

3D Cartpole

Acrobot

2D Maze

3D Maze

Breakout

Pong

7.2 Untested Design Choices

Briefly explain all parameter/model choices, why we did not tune/experiment with these, and what effect these might have!

8 Results

Display individual experiments tables comparing modifications against baselines and each other?

Display individual plots showing the same?

9 Limitations

10 Conclusions

References