Autonomous Selection of Inter-Task Mappings in Transfer Learning (extended abstract)

Anestis Fachantidis
Department of Informatics
Aristotle University of Thessaloniki

Ioannis Partalas
Laboratoire LIG
Université Joseph Fourier

Matthew E. Taylor
School of Electrical Engineering and Computer Science
Washington State University

Ioannis Vlahavas
Department of Informatics
Aristotle University of Thessaloniki

Introduction

In recent years, a variety of *transfer learning* (TL) methods have been developed in the context of *reinforcement learning* (RL) tasks. Typically, when an RL agent leverages TL, it uses knowledge acquired in one or more (*source*) tasks to speed up its learning in a more complex (*target*) task.

When transferring knowledge between reinforcement learning agents with different state representations or actions, past knowledge must be efficiently mapped between the two tasks so that it assists learning. The majority of the existing approaches such as that of TIMBREL (Taylor, Jong, and Stone 2008) use a single pre-defined *inter-task mapping* (Taylor, Stone, and Liu 2007) given by a domain expert. To overcome these limitations and allow autonomous transfer learning we propose a generic method for the automated online selection of inter-task mappings in transfer learning procedures.

An important insight comes from the *multi-task* TL problem, where knowledge from multiple source tasks can be transferred to a single target task. Specifically Lazaric *et al.* (2008) propose an algorithm that implements multi-task transfer learning based on two concepts, those of *compliance* and *relevance*. These two probabilistic measures can assist an agent on choosing when and what knowledge to transfer by determining the most similar source task to the target task (compliance) and also on deciding which are the most similar source task instances to transfer (relevance). Our proposed method differs significantly since it is used in a different setting, where we consider using multiple inter-task mappings in a single-task transfer learning problem. This abstract proposes a novel algorithm suitable for that setting, allowing for different state and action variables between tasks.

The main contributions of this work are: i) a novel theoretical viewpoint in which every *multiple mapping* TL problem is equivalent to a *multi-task* TL, problem bridging two previously distinct problem settings and ii) a fully automated method for selecting inter-task mappings that alleviates the problem of predefining a mapping between source and target tasks in TL procedures.

Transferring with Multiple Inter-Task Mappings

.

Based on Lazaric *et al.*'s multi-task transfer method, we propose a reformulation which significantly extends it to use **multiple inter-task mappings, rather than multiple source tasks**.

We consider each inter-task mapping function as a hypothesis, proposed to match the geometry and dynamics between a source task and a target task. Mapping states and actions from a target task to a source task not only transforms the way we view and use the source task, but also the way it behaves and responds to a fixed target task's state-action query (because the later has to be mapped). Thus, every inter-task mapping X_i can be considered as a constructor of a new *virtual source task* S_{X_i}. This naturally **re-formulates the problem of finding the best mapping as a problem of finding the most *compliant* virtual source task**. Additionally, this re-formulation transforms the problem of finding which instances to transfer through a certain mapping, to a problem of sample *relevance*. Based on the notions of compliance and relevance described earlier in this text and using existing notation by Lazaric *et al.* (2008), we re-define the compliance λ_{X_k,τ_i} of a target task transition τ_i and a virtual source task S_{X_k} (constructed by mapping X_k), given a set of $\hat{S}_{X_k}$ source samples:

$$\lambda_{X_k,\tau_i} = \frac{1}{Z^P Z^R}(\sum_{j=1}^{m} \lambda_{ij}^P)(\sum_{j=1}^{m} \lambda_{ij}^R), \quad (1)$$

where $m = |\hat{S}_{X_k}|$ and P and R refer respectively to the transition and reward functions of the target task and $\lambda_{ij}^P, \lambda_{ij}^R$ are the transition and reward compliance's of a single target task tuple i to a single source task tuple j.

The compliance between the target task (not only one instance-tuple of it) and the virtual source task S_{X_k} generated by the mapping X_k is defined as:

$$\Lambda_{X_k} = \frac{1}{t}\sum_{i=1}^{t} \lambda_{X_k,\tau_i} P(s) \quad (2)$$

where $t = |\hat{T}|$, the size of the target task sample set and $P(s)$ is a prior over the source task instances, simply expressing the probability of selecting an instance from a specific virtual source task if you would randomly select from

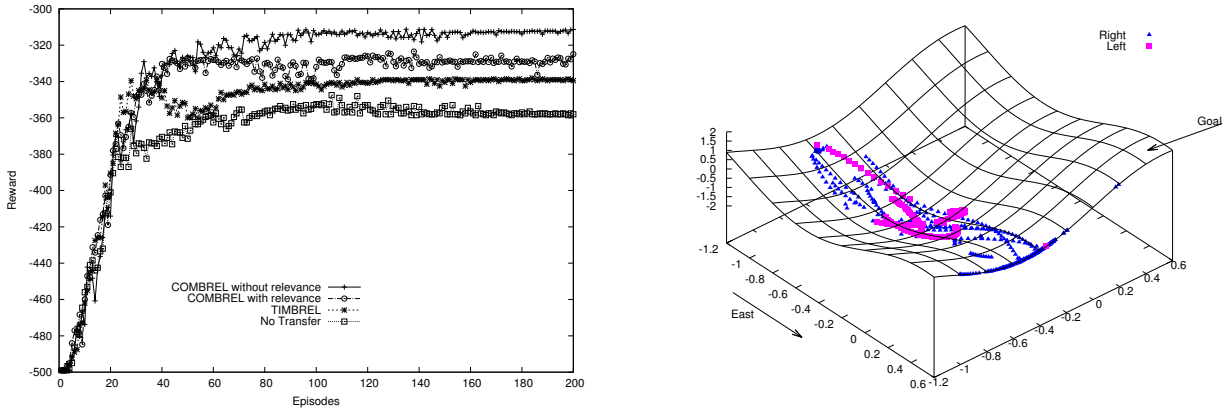

Figure 1: Left: Learning MC4D using 1000 source task (MC2D) instances. Right: Example action mappings for the target task action East in MC4D.

all the instances of all the available mappings. The relevance of a source task instance σ_j is similarly to Lazaric *et al.* (2008), but using the above notation changes.

In order to use compliance and relevance for selecting inter-task mappings, we propose an improved version of the multiple-mappings TL algorithm, *COmpliance aware transfer for Model Based REinforcement Learning* (COMBREL) (Fachantidis et al. 2012). In addition to calculating compliance, this version also uses relevance, and can dynamically select which mapping to use at different states in the target task.

COMBREL first records source task transitions and also generates the set of inter-task mappings that will be used. Then, for each mapping X_i it defines a corresponding virtual source task S_{X_i}. For the first e learning episodes in the target task, a number of target task transitions (instances) are recorded and the algorithm computes the compliance of each of the mappings, X_i to these target task instances to find the most compliant mapping. If the agent's model-based algorithm is unable to approximate a target task state with its current data, it attempts to transfer source task instances from the most compliant virtual source task, prioritizing them by their relevance.

Experiments and Results

Figure 1 (Left) shows the results from the first experiment in the Mountain Car 4D domain. COMBREL uses an exhaustive set of 1960 possible inter-task mappings and results are averaged over 15 trials. The results show the significant benefit of using a multiple-mappings method accompanied with a selection mechanism and also demonstrates the performance boost of the proposed algorithm over the other approaches. The performance difference of COMBREL compared to the other algorithms was statistically significant at a 95% level. COMBREL performs significantly better than TIMBREL, showing that a single and fixed mapping methodology performs worse than a multiple mappings one, even if that single mapping, is the intuitive one. This indicates that in some target task states, the intuitive mapping is not optimal.

Concerning the mappings actually chosen by COMBREL, in the total number of runs, COMBREL constantly ($> 95\%$) finds the correct (e.g., intuitive) state-mapping but frequently changes the action-mapping based on its position and velocity. The reason for this differentiation is that interchanging state variables in a state mapping (i.e., mapping a source task's position to a target task's velocity) results in large errors and thus mappings with very low compliance. However, interchanging actions in action mappings does not result in large errors since different actions can have very similar effects in different states. Figure 1 (right) shows that the mappings that translate action East to action Left are mostly used in the valley and the West portion of the state space. At these points COMBREL finds that the final transition effect of executing action East is more similar to executing the action left at an analogous point in MC2D (e.g. when the car has gained enough speed towards the west direction and its inertia causes an opposite transition effect to that expected when we have zero speed). This example shows that since COMBREL is an instance-based method, it handles mappings based on the actual similarity of the transition effects and not on any expected effect. As actions don't always express the actual action effects, COMBREL in MC4D selects different mappings, even for very nearby positions, depending on the car's speed, gravitational force, etc., thus **finding similarities between the final action outcomes.**

References

Fachantidis, A.; Partalas, I.; Taylor, M.; and Vlahavas, I. 2012. Transfer learning via multiple inter-task mappings. In *Recent Advances in Reinforcement Learning*, volume 7188 of *Lecture Notes in Computer Science*.

Lazaric, A.; Restelli, M.; and Bonarini, A. 2008. Transfer of samples in batch reinforcement learning. In *Proc. of the 25th ICML*, 544–551.

Taylor, M. E.; Jong, N. K.; and Stone, P. 2008. Transferring instances for model-based reinforcement learning. In *Proc. of ECML*, 488–505.

Taylor, M. E.; Stone, P.; and Liu, Y. 2007. Transfer learning via inter-task mappings for temporal difference learning. *Journal of Machine Learning Research* .