An introduction to Structural Learning - A new approach in Reinforcement Learning

Alexander Gabriel
Knowledge Engineering Group
Technische Universität Darmstadt
64289 Darmstadt
einalex@mayanna.org

Abstract

The recent combination of Structure Learning[1] and Reinforcement Learning has resulted in a number of promising scientific works that overcome previous problems that Reinforcement Learning techniques face when they are confronted with real world problems. This paper will give a short introduction into both Reinforcement Learning and Structure Learning, explain the combination and its advantages and then explore the combination of the two in a few selected examples. Lastly a selection of papers for further study will be given.

1 Introduction

Modeling reward-punishment learning processes through the use of Reinforcement Learning has been a reason for major progress in understanding animal and human decision making. There is however a drawback in using this approach.

When applied to real world problems, Reinforcement Learning is often confronted with hundreds of actions - for example the number of muscles in the body - or even more states - like the mass of visual, auditory and haptic sensor information. In these scenarios Reinforcement Learning becomes unusably slow [1, 2].

Yet animals and humans cope with this complexity effortless on a daily basis. To explain this discrepancy it has been suggested, that animals and humans might use the underlying hidden structure of the state space and action space to reduce the complexity significantly[3]. This approach was termed Structure Learning.

The following sections will give a short introduction first into Reinforcement Learning and second into Structure Learning and lastly into Parametric Learning. They will list known algorithms and touch on limitations. Section 5 will discuss some examples of Structure Learning in animals and humans.

[1] Another name is Structural Learning

2 Reinforcement Learning

Reinforcement Learning is a computational framework that allows to model unsupervised learning scenarios in which an agent learns how to behave in an environment in such a way as to maximize overall reward.

2.1 A Short Introduction

To that end, Reinforcement Learning makes use of states which may or may not hold rewards or punishments (negative rewards), actions whose outcome may or may not be probabilistic and Markov Decision Processes (MDPs).

The agent moves from one state to the next by use of its actions. The choice of actions is determined by a policy that may change during the learning process (active learning) or can stay constant (passive learning).

If the agent encounters rewards or punishments while following its policy on its way through the state space, those values are used to learn about the environment and the policy. One possibility would be to propagated a reward to predecessor states in an agent's internal model of the state space.

2.2 Basic Assumptions

Reinforcement Learning algorithms learn a policy which maps states or state-action pairs onto actions. This requires the existence of states and knowledge about which state the agent is in as well as the existence of discrete actions the agent can perform. A reward or punishment in at least one state is also needed.

2.3 Different Approaches

There are quite a few different agent designs which are optimized for various special cases or general performance. A few essential designs are listed here as an example[2]:

- reflex agents, that learn the utility of a policy which maps directly from states to actions
- Q-learning agents, that learn the expected utility of taking a specific action in a state
- utility-based agents, that learn the utility of states and use a transition model to chose an action

2.4 Limitations

Reinforcement Learning algorithms work well on simple problems, but become slow when confronted with real world problems. One prominent cause of this is the existence of many states or many actions within the problem. Efforts to circumvent this by approximation of the utility function have had some success, but rely on expert knowledge about the features [1].

Other types of real world problems don't have obvious states, but feature a possibly large number of sensor values from which the current state has to be deduced [2].

One approach that might cover both the complexity issue and the problem of unobvious states is to use the output of Structure Learning methods as input to the Reinforcement Learning algorithm. The next section will give a short explanation of Structure Learning.

[2] Agent design examples taken from [1].

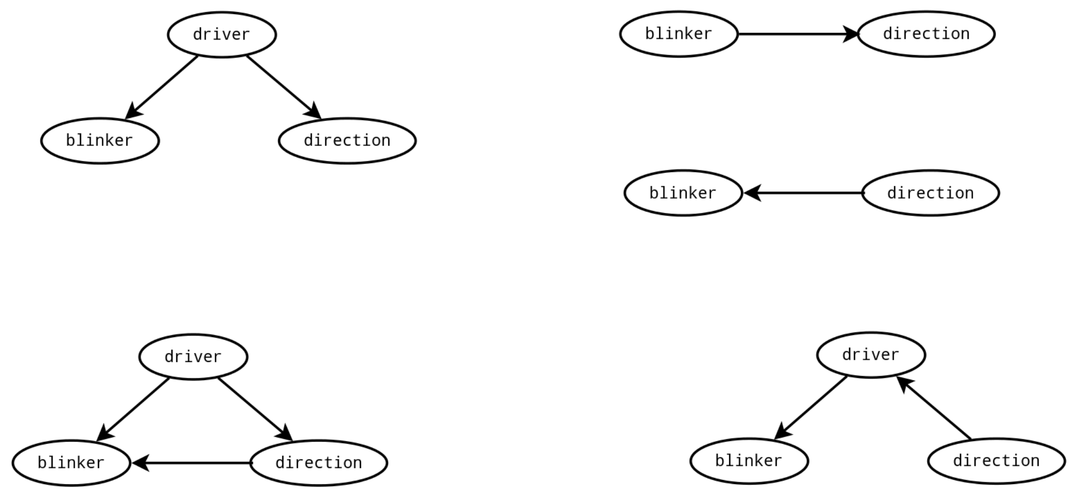

Figure 1: Differently structured Bayesian Networks

3 Structure Learning

Animals and humans learn much faster than artificial agents when confronted with the kind of real world problems mentioned in the last section.

It has been suggested that they make use of the hidden structure behind observations to reduce the size of the state space. Similar processes are probably responsible for the reduction of the action space. This idea, when coined into computational terms, leads to the discipline of Structure Learning.

3.1 A Short Introduction

The task of Structure Learning is to identify the hidden causes and relations that lead to a given set of values. A known structure of causes and relations can be modeled by a Bayesian Network, a directed acyclic graph of random variables as nodes and their conditional dependencies as edges.

Finding causes and relations is therefore analog to finding the equivalent Bayesian Network and can be summarized as finding an answer to the two questions “What are the random variables we need?” and “What are the relations between them?”.

3.2 Different Approaches

Unfortunately in many cases the number of possible structures is infinite and in the general case it is intractable to compute the actual structure. It is however possible to compute an approximation of it. Some algorithms that can be used to do this are:

- (HC) hill climbing [4]
- (CI) conditional independence [4]
- (SEM) the Bayesian Structural EM algorithm [5]
- (MCMC) Markov Chain Monte Carlo [6]

3.3 Application To The Problem At Hand

In real world problems, the agent has generally no direct information of what state the system is in. All it gets is the sensor data caused by the current state of the environment. The relation between cause and effect (state and sensor data) might be approximated by the use of Structure Learning.

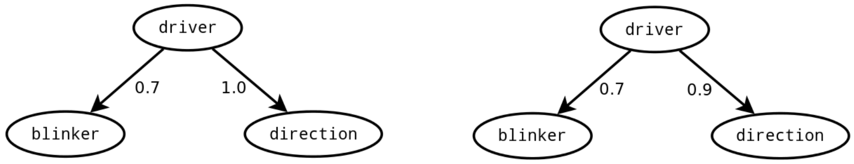

Figure 2: Bayesian Network with different conditional dependencies

A similar approach to the one mentioned in the last paragraph can also be used to model motor control tasks. Here the cause is the motor setting while the effect is the motion of whatever should be moved.

3.4 Advantages of Structure Learning

Structure Learning offers a general approach to reduce the complexity caused by high numbers of states and actions. It does so by identifying hidden lower dimensional subspaces [7]. In addition to that, Structure Learning allows for the identification of states where no obvious choice is present.

When Reinforcement Learning is combined with Structure Learning, the combination may lead to faster problem solving and may even increase the number of scenarios for which Reinforcement Learning is applicable.

Lastly, Structure Learning improves an agent's adaption speed to structurally similar tasks in the future if the agent also features a means to apply Parameter Learning [8].

4 Parameter Learning

While Structure Learning answers the question "What is the correct underlying structure?" and is concerned with the correct set of variables and the existence or non-existence of the conditional dependencies between those variables. Parameter Learning tries to answer the question "Given the correct structure how strong are the relations between the probability variables?".

While the later question has been addressed extensively in psychology and cognitive neuroscience, the former question has remained unsolved [8].

4.1 Basic Assumptions

To compute the strengths of the conditional dependencies, Parametric Learning relies on the existence of a Bayesian Network with nodes and directed edges.

4.2 Different Approaches

Once the structure behind the agents observations or behind a set of tasks is known and we have a Bayesian Network representing the task at hand, it is fairly simple to find the parameters that lead to the best performance or to adjust parameters to a new but structurally similar task. Well known algorithms that can be utilized to these ends are for example:

- (EM) the Expectation Maximization algorithm [9]
- (IPF) the Iterative Proportional Fitting algorithm [10]

5 Examples Of Structure Learning In Animals And Humans

5.1 Animals

The available research on animal learning phenomenons is generally older than that for humans and therefore doesn't look specifically for Structure Learning versus other possible explanations.

5.1.1 Structure Learning in perception

Bauvet and Vauclair showed in [11] that baboons can learn to compare multiple objects based on their edibility. More specifically they can learn to tell if two objects are either both (in)edible or one is edible while the other is not. This is significant because the learned feature is a hidden variable and has to be deduced.

5.2 Humans

The much more recent research on human behavior doesn't have the same drawbacks as the animal research.

5.2.1 Structure Learning on motor tasks

D. Braun et. al. present compelling evidence of Structure Learning in [7]. They use a set of experiments featuring different combinations of randomly varying visuomotor tasks to rule out acceleration of learning by the possible proximity of test and control tasks in parameter space.

5.2.2 Structure Learning on sequential decision making

Acuna and Schrater employ a Multi-armed-Bandit experiment to explore human structure learning on sequential decision making tasks [12]. They highlight that the perceived human behavior over tasks with varying structure is better captured by an optimal behavioral model than by two others each representing a belief in a fixed structure.

6 Conclusion

The discrepancy in performance of learning between artificial agents on the one hand and animals and humans on the other hand make it obvious that Reinforcement Learning alone can't explain animals and human learning behavior.

A growing body of literature suggests that the addition of Structure Learning might be able to bridge the gap in performance and lead to a more accurate model of learning to learn processes.

This work gives a short introduction into the methods used and supplies a starting point to the available research. More in-depth discussions of the topic and an extensive list of literature can be found at [8] and [2] which were also the basis for this summary.

References

[1] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*, volume 1st of *Series in Artificial Intelligence*. Prentice Hall, 3rd editio edition, 2010.

[2] Samuel J Gershman and Yael Niv. Learning latent structure: carving nature at its joints. *Current opinion in neurobiology*, 20(2):251–6, April 2010.

[3] Charles Kemp and Joshua B Tenenbaum. Structured statistical models of inductive reasoning. *Psychological review*, 116(1):20–58, January 2009.

[4] Ivan Stajduhar and Bojana Dalbelo-Basić. Learning Bayesian networks from survival data using weighting censored instances. *Journal of biomedical informatics*, 43(4):613–22, August 2010.

[5] Xavier Boyen, Nir Friedman, and Daphne Koller. Discovering the Hidden Structure of Complex Dynamic Systems. In *Proceedings of the 15th Annual Conference on Uncertainty in Artificial Intelligence*, number August, pages 206–215, 1999.

[6] W. Buntine. A guide to the literature on learning probabilistic networks from data. *IEEE Transactions on Knowledge and Data Engineering*, 8(2):195–210, April 1996.

[7] Daniel a Braun, Ad Aertsen, Daniel M Wolpert, and Carsten Mehring. Motor task variation induces structural learning. *Current biology : CB*, 19(4):352–7, February 2009.

[8] Daniel a Braun, Carsten Mehring, and Daniel M Wolpert. Structure learning in action. *Behavioural brain research*, 206(2):157–65, January 2010.

[9] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1):1–38, 1977.

[10] Radim Jirousek and Stanislav Pleucil. On the effective implementation of the iterative proportional fitting procedure. *Computational Statistics & Data Analysis*, 19(2):177–189, 1995.

[11] Dalila Bovet and Jacques Vauclair. Judgment of conceptual identity in monkeys. *Psychonomic Bulletin & Review*, 8(3):470–475, 2001.

[12] Daniel E Acuña and Paul Schrater. Structure learning in human sequential decision-making. *PLoS Computational Biology*, 6(12):e1001003, January 2010.