AAAI Technical Report FS-12-07
Robots Learning Interactively from Human Teachers

Using Causal Models for Learning from Demonstration

Halit Bener Suay, Joseph Beck, Sonia Chernova
Worcester Polytechnic Institute
100 Institute Dr.
Worcester, Massachusettes 01609

Abstract

Most learning from demonstration algorithms are implemented with a certain set of variables that are known to be important for the agent. The agent is hardcoded to use those variables for learning the task (or a set of parameters). In this work we try to understand the causal structure of a demonstrated task in order to find: which variables cause what other variables to change, and which variables are independent from the others. We used a realistic simulator to record a simple pick and place task demonstration data, and recovered different causal models using the data in Tetrad, a computer program that searches for causal and statistical models. Our findings show that it is possible to deduce irrelevant variables to a demonstrated task, using the recovered causal structure.

Graphical models are powerful tools to express one's belief about the underlying model of the observed world (Russell and Norvig 2003; Pearl 2009). They can be used for making probabilistic inference of the state variables, modeling the state transition of the world, monitoring variables, and smoothing the earlier inference with the current evidence in hindsight. All these tasks require a model of the world. Given enough information, one can manually model the world, or use causal structure recovery algorithms to come up with different plausible models that may represent the true model underlying the observed data.

This ability of interpreting the observed (raw) data, and trying to find how the world works, can be useful for Learning from Demonstration for software agents and robots. Learning from Demonstration can be defined very generally as the subfield of computer science / robotics that aims to create agents that can acquire different skills and tasks without the necessity of coding. The teacher can be a human, as well as another agent. There are several different approaches for teaching a task to an agent: teaching with human rewards and guidance (e.g. Interactive Reinforcement Learning), autonomous learning using a reward function (e.g. Reinforcement Learning), explicit approaches where the teacher provides the label for different states (e.g. Confidence Based Autonomy), to name just a few (Argall et al. 2009).

There are several challenges in Learning from Demonstration; for instance dealing with noisy or sometimes even seemingly conflicting demonstrations depends on how well the agent can generalize over the state space. Another challenge is learning the task as fast as possible, with the minimum information available. Cost/benefit based approaches try to tackle this problem by assigning every extra information required a cost, and by estimating the benefit that the agent will get from getting that information. Modeling the learning problem can also be hard. Most machine learning techniques have several parameters that need to be tweaked or learned, which adds another layer to the problem. In the light of a given demonstration, understanding what is relevant to the task is another problem. When it comes to implementation, most Learning from Demonstration techniques are implemented in a way that the state vector of the agent contains only relevant variables. Filtering out the unnecessary variables, i.e. feature selection, is generally done by the researcher who implements the technique.

We believe that causal structure recovery can be helpful for tackling the first and the last problems mentioned. If an agent can understand the causal relationship between several different variables, we may be able to prevent the agent from over-fitting to a certain set of values of those variables. Moreover, the strength of causation between observed variables can also help us filter out the unnecessary variables automatically. If we can filter out unnecessary variables, then we may be able to shape (and re-shape) the state vector of our agent throughout its mission.

In this paper we present a preliminary work where we collected observed data, using a realistic simulator (Gazebo) with a physics engine, for a simple pick and place task demonstrated by a human teacher to a realistically simulated service robot, PR2.

We used Robot Operating System[1] as our main framework. Recorded data then was loaded to a causal structure recovery software (Tetrad-IV)[2], and different causal structures of the underlying world (i.e. the task demonstration) were recovered. We show that: a) in a simple demonstration, this technique can be helpful to distinguish relevant variables from irrelevant variables and b) as we add expert

[1] http://www.ros.org/wiki/
[2] http://www.phil.cmu.edu/projects/tetrad/

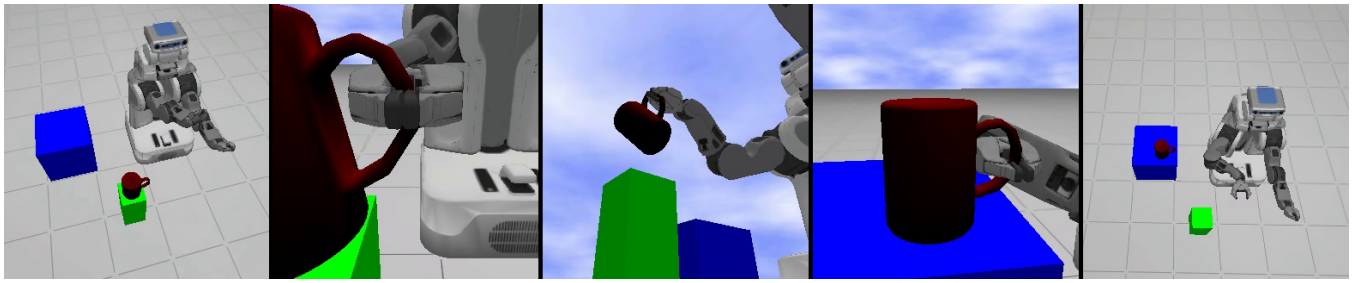

Figure 1: Snapshots of the world during task demonstration.

knowledge into the causal structure search algorithms, recovered models get more useful.

The rest of the paper is organized as follows: Section 2 explains how we recorded our observed data. In Section 3 and 4 we present our findings using the data recorded and the causal structure recovery software. Section 5 gives some ideas that would be interesting to pursue in future work, and finally Section 6 concludes the paper.

Experimental Setup

Observing the world accurately and obtaining data about, for example, several different objects' locations can be quite challenging depending on the environment. To keep this first step as simple as possible we gathered our observation data from a realistic 3D simulator, Gazebo. In the world that we have observed, there were two box-shaped tables, a coffee cup, and a service robot, PR2. Box-shaped tables were static objects that are fixed throughout the session. In this world, an expert human teacher (first author of the paper) demonstrated a simple pick and place task by teleoperating the robot using the computer's keyboard. The task was to pick the coffee cup from the green table and place it on the blue table. The teacher used only the robot's right gripper. Initially the right gripper started from a home position and after placing the cup on the table, the teacher returned the gripper back to somewhere near this initial position. We started recording the data from the first move (i.e. home position of the gripper) until the last move (i.e. back to home position) of the demonstrator. This session included moving the right arm to the coffee cup, opening the gripper, taking the grasp position, closing the gripper, lifting the cup, bringing the cup on the blue table, putting the cup on the table, opening the gripper, moving the gripper away from the cup and taking the gripper back to its home position. Some snapshots of the world and the demonstration are shown in Fig. 1.

We recorded two kinds of data: data that was relevant with the demonstration (cups position, end-effector's position, grippers' state i.e. distance between fingers and the pressure sensor data) and data that was not relevant with the demonstration (both tables' center of gravity locations). Since those two points were static in the world, and there was no data about the size of the boxes, they did not have any informative value for the task. In fact, we could have recorded just any random two (or more) values, however since those two points had a semantic meaning (locations of the tables) it is plausible that they are in the task demonstration. Length of the demonstration was around 5 minutes and all data was observed at a frequency of 2 Hz.

Experimental Results

Recorded data (625 sequential observations in total) were saved to a csv file for causal structure recovery. For recovering different causal structures, we used Tetrad, freely available software that creates, simulates data from, estimates, tests, predicts with, and searches for causal and statistical models. Our data was temporal (that is, every observation belonged to a time slice), and for Tetrad to handle the temporal data we had to copy every j+1th observation next to every jth observation.

We tested the causal model search algorithm under three conditions: with no additional expert knowledge, with only temporal information added (i.e. the information that $t_0 \rightarrow t_1$); and with both temporal information added and the fact that gripper position causes the gripper pressure to change and the fact that the end-effector's position causes the coffee cup's position to change (i.e. gripper_posgripper_effort and endeffector_poscup_pos). The algorithm outputs a graph where the nodes are variables and the edges are causal relations. The meanings of the edges are explained in detail in the following subsections.

The PC Algorithm

The PC algorithm (Spirtes and Glymour 1991) is a pattern search algorithm which assumes that:

- The underlying causal structure of the input data is acyclic, and that no two variables are caused by the same latent (unmeasured) variable.
- The input data set is either entirely continuous or entirely discrete.
- If the data set is continuous (which is, in our case), the causal relation between the two variables is linear, and that the distribution of each variable is Normal.
- The sample is ideally i.i.d.
- No relationship between variables in the data is deterministic.

In the PC algorithm, one can interpret the output as follows:

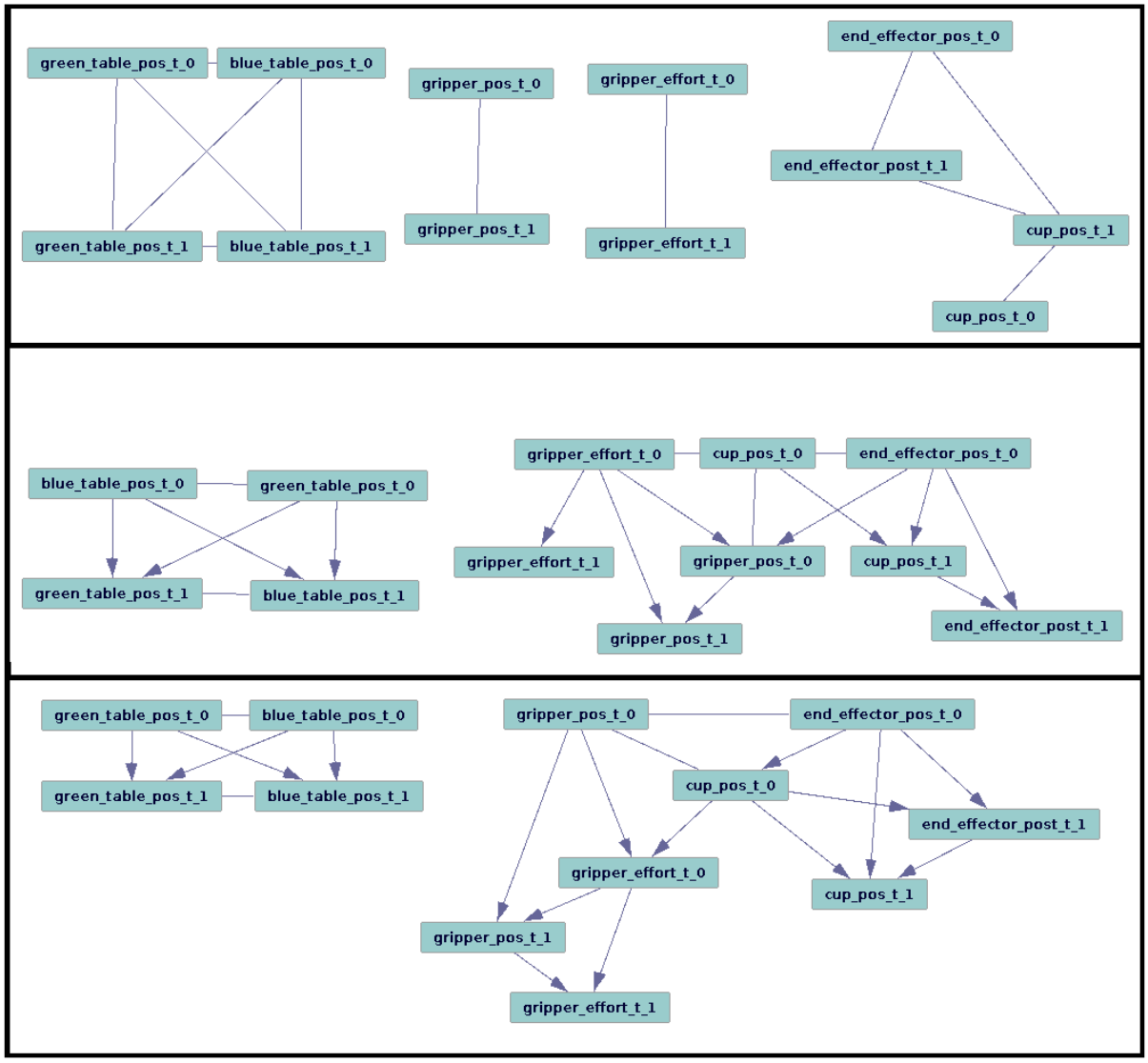

Figure 2: Results returned by the PC Algorithm.

- A directed edge $X \rightarrow Y$ means that X is deduced as a direct cause of Y.
- An undirected edge between X and Y means that the algorithm cannot tell if X causes Y or if Y causes X (in which case additional information may resolve the issue).
- The absence of an edge between any pair of nodes means they are independent.

The results of the search algorithm are shown in Fig. 2, where, the three figures below respectively with the order of no additional data, temporal data added (called Tiers) and temporal data plus two other facts previously mentioned are added. When all three results are compared there are a few points to highlight.

First of all, we can see that as we add knowledge in the algorithm, more causal relations are recovered. In the first result gripper position and effort (i.e. pressure data) seem to be independent, whereas right after adding the tiers, the algorithm returns a model where gripper_effort is a direct cause of gripper_position. The other thing to highlight is, although with additional knowledge, gripper, cup and end-effector data are returned to be causally related, green_table and blue_table variables are always independent from the rest of the variables. This is useful, especially from the LfD perspective, because when after a task demonstration we obtain a result such as shown in the last figure, we can, for instance, use this information about independence of variables to ask for clarification to the teacher. Missing links are especially important, as they suggest no causation.

One interpretation of the causal model suggested on the right hand side of the last figure can be as the following:

- gripper_pos and end_effector_pos are causally related.

This makes sense because we opened / closed the gripper at different positions of the end_effector.

- end_effector_pos causes cup_pos to change, which causes end_effector position to change, which causes cup_pos to change in turn. This also makes sense, because we keep moving the cup until it gets to a certain location. If it is not at that location yet, we keep moving it (hence cup_pos_t_0 end_effector_pos_t_1).
- gripper_pos and cup_pos are causally related. As we moved the cup from start to finish, gripper was closed. However right before grasping, and right after leaving the cup the distance between the fingers changed due to opening/closing action. The directionality is not recovered but the algorithm found the causation between these variables.
- gripper_pos cause gripper_effort to change. This is a knowledge that we manually added in the algorithm.
- cup_pos causes gripper_effort to change. We believe that this is again due to closing and opening the gripper in two different locations of the world (once for grasping the cup, once for releasing the cup).

Future Work

This work is just the very first step in understanding how one can use causal structures for Learning from Demonstration. There is clearly a lot of future work to incorporate causal structures in an implemented LfD method. One incremental step could be to use a filtering based on causation weights (or strengths) to filter out the edges with low weights.

Observing the difference between successful and failed demonstrations can tell us how the recovered causal structures change with a given search algorithm. When there is not enough knowledge (or data) to recover some relations, asking a human teacher reasonable questions for further clarification could be another future work. Recovering a causal structure with multiple successful demonstrations means more data. If the underlying model of the world is not suitable, no matter how much data we have, we cannot recover all the causal structure (Pearl 2009). Investigating the effect of this phenomenon on task demonstrations could be interesting.

Conclusion

In this paper we present a small first step towards using causal models for Learning from Demonstration methods. Intuitively we can understand that for a task demonstration, we need objective(s) and actions. The actions are usually performed towards an objective and they cause the variables in the world to transition from one state to another towards the goal (in case of a successful demonstration).

Here we recorded a simple pick and place task demonstration data using a realistic simulator with a teleoperated service robot, PR2. Using the demonstration data, we tried to recover a causal structure for the demonstration. Causal models we obtained told us which variables have a causal link to what other variables. Although it is not possible to determine the goal of the task with causal models, we show that irrelevant variables for the task (table locations in our case) can be found to be independent from the rest of the variables. This could be useful for feature selection after a task demonstration.

Another conclusion we would like to make is the importance of the additional knowledge given to a search algorithm. Our findings show that, just the temporal information makes a big difference in terms of the directionality of the causal links. Tetrad assumes that if A happens after B, A cannot be the cause of B. Querying a human expert for even more additional information is a possibility, however in general, agents are desired to generalize over a task with the minimum number of demonstrations and set of knowledge possible.

We show that causal model recovery has some useful properties for Learning from Demonstration. How to leverage a causal model is left as a future work.

References

Argall, B. D.; Chernova, S.; Veloso, M.; and Browning, B. 2009. A survey of robot learning from demonstration. *Robot. Auton. Syst.* 57:469–483.

Pearl, J. 2009. *Causality: Models, Reasoning and Inference*. Cambridge University Press.

Russell, S., and Norvig, P. 2003. *Artificial Intelligence: A Modern Approach*. New Jersey: Prentice Hall.

Spirtes, P., and Glymour, C. 1991. An algorithm for fast recovery of sparse causal graph. *Computer Review* 9:62–72.