A Robot That Improves Its Ability To Learn

Joseph O'Sullivan Sebastian Thrun
School of Computer Science
Carnegie Mellon University
Pittsburgh PA 15213

Abstract

The use of machine learning is attractive in the design of autonomous robots, since it enables robots to adapt to the unforeseen. However, one of the key bottlenecks of current machine learning algorithms is the enormous sample complexity, which appears to prohibit their usage in all but the simplest of robotic domains. To make machine learning more practical in such domains, more powerful algorithms are needed that can generalize more accurately from less training data.

This paper investigates the feasibility of learning algorithms that gradually improve over the lifetime of the robot. When faced with a novel thing to learn, knowledge acquired in previous learning tasks improves the ability of the robot to generalize, hence reduces the sample complexity. In this paper, we report results of applying a particular algorithm to mobile robot perception problems, which was originally proposed by Suddarth [15]. The learning tasks considered involve the recognition of persons, objects and locations. We illustrate that having previously learned related tasks allows a robot to learn a novel task from significantly fewer training examples than if it was learning from scratch. Based on these results, we argue that self-adjusting learning strategies are superior to conventional learning algorithms in many robotic domains.

1 Introduction

Over the last few years, machine learning algorithms have been successfully applied to various robotic domains. For example, Pomerleau successfully trained an vehicle to drive autonomously with speeds of up to 55 mph [12]. Christiansen proposed an approach that allowed learning of sequences manipulations in the presence of uncertainty [6]. Despite these initial successes, one of the crucial bottlenecks of current machine learning algorithms is sampling complexity. To learn something complex, usually a vast number of training examples is required. This is not prohibitive in cases where training data are inexpensive to obtain. However, most robotic domains can not afford the luxury of repeated trials.

The need for training data has long been recognized as one of the crucial limitations of current machine learning approaches. Research is ongoing into creating systems which learn as much as possible from each example. In such systems, the possible hypothesizes which fit the observed examples are vastly underconstrained, and so some form of bias is introduced to obtain strong generalization to novel data. Such biases can come from situations such as a human domain expert providing constraints which the learned system must obey [3, 9, 12]. While such methods are generally superior to less biased methods, they strongly rely on the existence of a human expert to set up correct biases.

An alternative way, which will be explored in this paper, is to gradually *learn* bias, so that a robot generalizes better when facing the n-th thing to learn than when facing the first. The theory of learning biases to guide subsequent learning is a focus of current research in the machine learning community. Some basic mechanisms which have been found to allow separate tasks to contribute biases are:

- Features created for predicting one task can be reused for novel tasks [15, 13, 2, 5].
- Features shared across multiple tasks are more robust to noise in the training data[5].
- Features shared across multiple tasks are more likely to capture true regularities of a domain[2, 5].
- Features useful to multiple tasks can be easier to learn in one task T than another T', due to T' using features in a more complex way[1, 15].

In this paper, we build on these results and show how a mobile robot can effectively improve its learning abilities gradually over multiple tasks, by exploiting domain knowledge implicit in previous tasks. As a result, the methods proposed here allow a robot to gradually reduce its need for new training data as it becomes more experienced. The particular learning problem studied here arises from map-based navigation in indoor office environments [10, 4]. Plans that are routinely built and executed by our robot, Xavier, often fail, because some of the robot's paths are blocked by obstacles (such as closed doors). While it is easy to recognize plan failure after the fact (e.g., collisions with obstacles), it is difficult to anticipate them before they occur. The learning task we take as an example is therefore one of recognizing that a map-based navigation plan will fail. The scientific question underlying this research is: *Can this task be learned from less training data, if the robot previously learned other, related things, such as recognizing a particular person, object, or location?*

In the remainder of this paper, we first describe the algorithms used in our research in Section 2. Section 3 describes the robot and the particular learning problems we studied. Section 4 illustrates that, by transferring knowledge from previous learning tasks, our mobile robot generalizes more accurately from less training data, illustrating that it adjusted its internal bias to better match its learning tasks. Finally, the paper is concluded by a list of open questions that arise from this research. Based on our experience we discuss the wider applicability of the approach to robot learning, the mechanisms that made this result possible and the weaknesses we perceive in this implementation.

2 Using Related Tasks

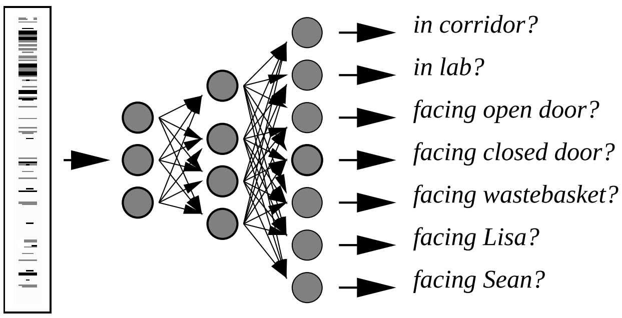


Figure 1: Training a network on multiple tasks.

Backpropagation[14], probably the most popular artificial neural network learning method, applies gradient descent on the weights of a multi-layer network. This effectively minimizes the squared error between the outputs of a neural network and the target values for these outputs as provided from the training data. It is said to be a "tabula rasa" learner since it does not use any external knowledge besides the training data, and its bias is by no means specific to the domain it is being applied to.

Given sufficient data, Backpropagation can usually train a neural network to learn a simple task such as "Is Xavier facing a closed door?". However, Backpropagation and other "tabula rasa" learning algorithms scale poorly to complex domains if training data is scarce.

In mobile robot domains often more than particular task has to be learned. For example, an indoor service robot may face tasks such as mail delivery, maintenance, personal services, and find-and-fetch tasks, which involves recognizing many objects, places, landmarks, and so on. Suddarth and others [15, 1, 5, 2, 17] have shown that a "weak bias" learner such as Backpropagation can benefit from biases available from related tasks by training a single Backpropagation network on multiple tasks as in Figure 1. It has been demonstrated that by using this framework, each of the above mechanisms has the opportunity to come into play [5, 2].

In this paper, we attempt to apply these techniques to relevant perceptual mobile robot problems. Inspired by the work listed above, a diverse range of possible tasks were chosen for this study:

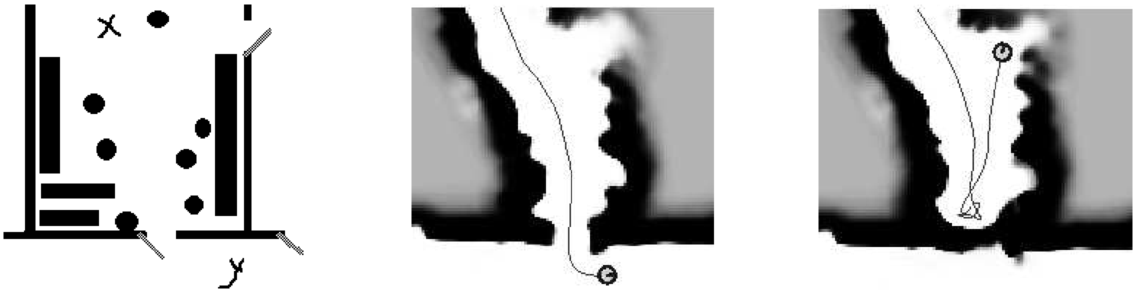
People recognizer: Several members of the laboratory walked in front of the robot as it roamed the lab, staying within its field of view during data collection. As such, we defined 4 novel tasks:

- Is Xavier facing Lisa?
- Is Xavier facing Sean?
- Is Xavier facing Joseph?
- Is Xavier facing Sebastian?

Location recognizer: A common task for mobile robots is to recognize landmarks in the world. To mimic such an effect, Xavier roamed some locations in the environment, defining 2 further tasks:

- Is Xavier in the laboratory?
- Is Xavier in the corridor?

Object recognizer: Finally, we chose some tasks similar to the original goal, of recognizing the nature of an object in the field of view:



(a) Lab Diagram (b) Open Door. (c) Closed Door.

Figure 2: The problem tackled: in a lab (a) containing chairs, tables, and two possibly open exits, Xavier is to get from start X to goal Y. When the door is open (b), the path is straightforward. When the door is closed (c), detecting the closed door requires multiple trials before the occupancy grid accumulates sufficient evidence to believe the door closed and to cause a new path to be planned.

- Is Xavier facing an open door?
- Is Xavier facing a wastebasket?

While it is not necessarily the case that a robot would initially have data available for any one of these tasks, we expect long living robots to have gathered data for a similar broad range of tasks. If we can show that the performance of the learned system is better by learning these tasks in addition to the novel task of "Is Xavier facing a closed door", we will have demonstrated that a long living robot can expect a learning speedup as the number of tasks to which it has been exposed increases.

3 Problem Setting

Figure 3 shows the mobile robot, Xavier, used in the tasks reported here. Xavier was developed at Carnegie Mellon University[11] and is built on a 4 wheeled omnidirectional base, 24 inches in diameter. The sensors we use are the color camera on top of the robot, and a ring of sonar sensors visible approximately one third of the way down the robot torso.

The learning problem described in turn is actually an important problem for Xavier's current indoor navigation system. In day to day use, this robot is frequently required to exit our laboratory at CMU, shown in Figure 2(a). There are two exits from this lab: one is wide and opens to the main building corridor, the other opens to a side chamber and a longer route to the corridor. However, at a particular point in time, one of these doors may be closed, thereby causing the initial plan to fail. The only way for the robot to find out is to travel to them and to test using information derived from its sonar sensors. Sonar

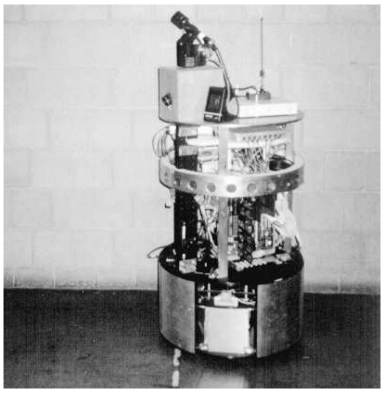


Figure 3: The robot used in this study, Xavier of Carnegie Mellon University's Learning Robot Laboratory.

readings are accumulated in an occupancy grid[10], and expected doorway locations are checked for blockages in this grid. A sample exit from the laboratory is depicted in Figure 2(b). Here, Xavier must test the doorway repeatedly before being confident that it is shut, and creating a new plan. The learning problem is straightforward: Can Xavier used its sensors to anticipate the failure of its plan, inferred by a closed door, before it reaches its door and changes its internal occupancy maps? And, if so, how efficiently can it do it, i.e., how many training examples does this learning task require?

4 Experimentation

To make convergence of the Backpropagation training feasible, camera images were subsampled to a

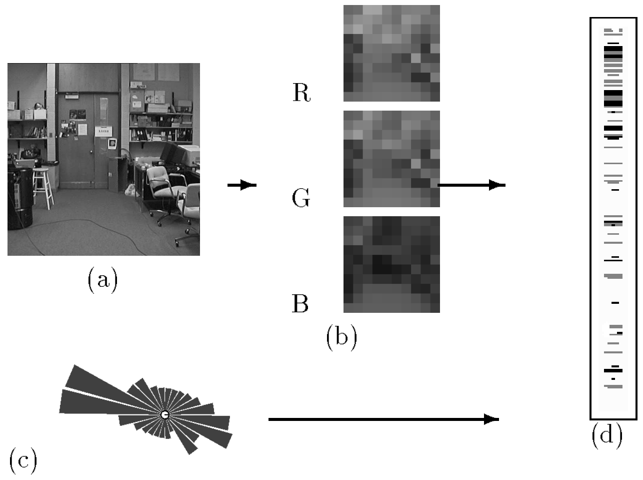


Figure 4: The original view of the world (a) from the camera is down-sampled to a reduced RGB image (b) in order to make experimentation practical. This image and 24 sonar readings (c) yield the final input representation (d). The darkness of the individual lines in the vector diagram (d) indicates the magnitude of the corresponding vector value.

10×10 red-green-blue camera picture and combined with the 24 sonar readings taken at the same instance of time (Figure 4). The resultant vector of size 324 is the state representation used as input to the neural networks throughout this paper. Backpropagation can learn a task such as "Is Xavier facing a closed door?" by assigning an output unit to this task. For all positive training examples of that class, the assigned output unit is set to 0.9, otherwise, it is set to 0.1.

We wished to test the hypothesis that the task "Is Xavier facing a closed door" can be learned from less training data, if Xavier previously learns other, related things, such as recognizing a particular person, object, or location. There are three components to this. First, there is the process of data collection, which causes each state representation to be labelled as a positive example of a single task. The second component, the training procedure, is repeated for the single task by itself as well as for the single task having made available information from all the other tasks. Thirdly, there is an evaluation component, under which we investigate whether Xavier indeed has improved its ability to learn.

4.1 Data Collection

For each of the 9 tasks suggested from Section 2, approximately 2 minutes worth of labelled positive examples were manually gathered. Since ideally we wish Xavier to learn from a few seconds worth of data, this is deemed to be sufficient. This data collections yielded approximately 360 labelled positive training examples of each task.

4.2 Training

The data from each task were randomly subsampled, with X positive examples of each task used for training. X defaulted to 60. One hundred further examples randomly selected from all tasks were utilized for cross-validation during training and the remaining examples were held back for evaluation.

There are certain implementation issues associated with any use of Backpropagation. Here, an offset sigmoid function is used in the threshold units to avoid the "stuck unit problem"[7], and a weak weight decay term was used to help generalization [8].

The Backpropagation algorithm was applied to learn the output task/tasks from the training data. Following each training epoch, the network's performance on the cross-validation set was measured. In the early phase of training, Backpropagation usually reduces the error rate of the network on both the training data and the cross validation data. Often, the network eventually begins to overfit the training data, causing the error rate on the unseen cross-validation set to rise. The network weights at the epoch of best performance of the network over the cross-validation set is considered to be a good approximation to the best performance on any unseen data, and these weights are the weights returned by the algorithm.

4.3 Evaluation

There were two components to the evaluation. The first, an ad hoc demonstration, evaluated the performance of the learned system applied to the original problem of map-based navigation. The second, the more rigorous component, evaluated the performance of the trained networks on the unseen evaluation set.

Figure 5 plots the results of applying the trained networks to the unseen evaluation dataset. Shown are two curves, plotting generalization accuracy on the unseen test set as a function of the number of training examples. Each experiment was repeated 100 times, using different sets for training and testing. Curve # 1 shows the results of standard Backpropagation, which is the performance that one expects were the recognition of closed doors the *first* task the robot learned, and curve # 2 shows the corresponding results using the additional learning tasks to bias generalization. These results reflect the improved generalization rates. Although the robot never learned a concept that directly related to the distinction of closed doors in its past, it nonetheless can exploit its previous learning experience to improve its generalization abilities. These findings demonstrate our hypothesis, that a novel task can be learned from less training data, if Xavier previously learns other, related things.

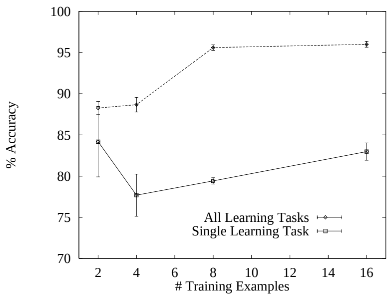


Figure 5: Percentage Accuracy at detecting closed doors on an Evaluation set, plotted with 95% confidence intervals. For a small number of examples, using multiple tasks demonstrates a significantly more accurate performance.

When integrating the new recognition software into our existing navigation software, we recognize that cannot expect 100% accuracy from the learned system. Therefore, readings were combined from three subsequent locations to give an improved (conservative) estimator of the status of the door. The newly learned behavior is visible in Figure 6, and should be compared with the handcoded behavior in Figure 2(c). Instead of recognizing the closed door after changing the occupancy grid (which is time-consuming, since evidence of the door being open has to be overcome by at least as many sensor readings as those that indicated the open door), Xavier now can anticipate the upcoming plan failure and re-plan earlier. In our experience with trained networks, when combining multiple different positions, no false positives have been reported (in over 20 attempts at room exit). This means that the worst case performance of our system degrades to that of the original hand-coded behavior. It never was seen to perform worse, as would be demonstrated by using the slower exit when the main entrance was open. We note that this behavior is identical under training a single and multiple tasks, when sufficient examples are available. However, given the results presented in Figure 5, we suspect that at least one order of magnitude more training examples are required to achieve the same level of performance exhibited by our new approach.

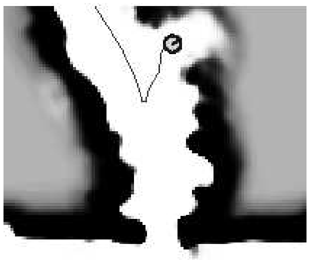


Figure 6: The newly learned behavior for detecting closed doors enables a more efficient plan for exiting the laboratory, avoiding the dependence on sonar evidence accumulation.

5 Discussion

This paper presents results of applying machine learning techniques to a perceptual problem arising in mobile robot control. A key belief underlying our research is that in order to be useful for robotics, machine learning algorithms must generalize accurately even from extremely small datasets. To achieve this goal, this paper advocates learning methods that gradually improve their ability to generalize accurately, over the lifetime of a robot. We demonstrated, using a learning approach that was originally proposed in [15], that a mobile robot exhibits drastically improved learning capabilities when faced with the n-th thing to learn. Our empirical study involved the recognition of closed doors.

It was encouraging that by selecting a diverse selection of tasks, we improved learning curves for a novel task. However, in a previous work, greater improvements were noted when the related tasks were more homogeneous[16]. These leads towards a number of questions:

- What representations were created?
- How well did these representations get used? Were they used, or if not, what components of the extra data contribute toward a learning speedup?
- Most importantly, what framework would describe the effects of task relevance and would en-

able an autonomous system to choose which tasks to combine?

From a practical point of view, the use of Backpropagation is limiting. Without using a dedicated hardware implementation, the convergence time for training is longer than the data collection time. As such, the reduction in the number of training examples offered from using multiple tasks is insignificant. There exist a broad range of machine learning algorithms which have performance characteristics equivalent to that of neural networks, and yet offer low iteration training. It remains to be seen whether the mechanisms described in the Introduction can be captured by these techniques.

Acknowledgments

This research is sponsored in part by the National Science Foundation under award IRI-9313367, and by the Wright Laboratory, Aeronautical Systems Center, Air Force Material Command, USAF, and the Advanced Research Projects Agency (ARPA) under grant number F33615-93-1-1330.

References

[1] Y. S. Abu-Mostafa. Hints. *Neural Computation*, 7:637–671, 1995.

[2] J. Baxter. *Learning Internal Representations.* PhD thesis, The Flinders University of South Australia, 1994.

[3] S. W. Bennett and G. F. DeJong. Real-world robotics: Learning to plan for robust execution. *Machine Learning*, to appear.

[4] J. Buhmann, W. Burgard, A. Cremers, D. Fox, T. Hofmann, F. Schneider, J. Strikos, and S. Thrun. The mobile robot rhino. *AI Magazine*, 15(1):31–38, 1995.

[5] R. Caruana. Learning Many Related Tasks At the Same Time With Backpropagation. In *Advances in Neural Information Processing Systems 6*. Morgan Kaufmann, December 1994.

[6] A. D. Christiansen. *Automatic Acquisition of Task Theories for Robotic Manipulation.* PhD thesis, Carnegie Mellon University, March 1992.

[7] S. E. Fahlman. An empirical study of learning speed in back-propagation networks. Technical Report CMU-CS-88-162, School of Computer Science,Carnegie Mellon University, Sept 1988.

[8] A. Krogh and J. A. Hertz. A Simple Weight Decay Can Improve Generalization. In J. E. Moody, S. J. Hanson, and R. P. Lipmann, editors, *Advances in Neural Information Processing Systems 4*, pages 950–957. Morgan Kaufmann, December 1992.

[9] T. M. Mitchell. Becoming increasingly reactive. In *Proceedings of the Eight National Conference on Artificial Intelligence*, pages 1051–1059. AAAI Press/MIT Press, 1990.

[10] H. P. Moravec. Sensor fusion in certainty grids for mobile robots. *AI Magazine*, 8(2):61–74, 1988.

[11] J. O'Sullivan and K. Haigh. Xavier Manual. Carnegie Mellon University, School of Computer Science, `http://www.cs.cmu.edu/~Xavier`, March 1994.

[12] D. Pomerleau. *Neural Network Perception for Mobile Robot Guidance.* Kluwer Academic Publishers, 1993.

[13] L. Y. Pratt. Experiments on the Transfer of Knowledge between Neural Networks. In *Computational Learning Theory and Natural Learning Systems, Constraints and Prospects*, chapter 14, pages 523–560. MIT Press, 1994.

[14] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. In D. E. Rumelhart and J. L. McClelland, editors, *Parallel Distributed Processing. Vol I+II.* MIT Press, 1986.

[15] S. C. Suddarth and A. D. C. Holden. Symbolic-neural systems and the use of hints for developing complex systems. *Int. J. Man-Machine Studies*, 35:291–311, 1991.

[16] S. Thrun and T. Mitchell. Learning one more thing. In *Proc. of the Fourteenth International Joint Conference on Artifical Intelligence*, San Mateo, CA, 1995. Morgan Kaufmann.

[17] S. Thrun and T. M. Mitchell. Lifelong robot learning. In L. Steels, editor, *Proc. NATO ASI on The Biology and Technology of Intelligent Autonomous Agents*, March 1993. Also appeared as Technical Report IAI-TR-93-7, University of Bonn, Dept. of Computer Science III.