Is Transfer Inductive

Chris Thornton
Cognitive and Computing Sciences
University of Sussex
Brighton
BN1 9QH
UK

Email: Chris.Thornton@cogs.susx.ac.uk
WWW: http://www.cogs.susx.ac.uk
Tel: (44)1273 678856

November 27, 1996

Abstract

Work is currently underway to devise learning methods which are better able to transfer knowledge from one task to another. The process of knowledge transfer is usually viewed as logically separate from the inductive procedures of ordinary learning. However, this paper argues that this 'seperatist' view leads to a number of conceptual di culties. It offers a task analysis which situates the transfer process *inside* a generalised inductive protocol. It argues that transfer should be viewed as a subprocess within induction and not as an independent procedure for transporting knowledge between learning trials.

1 Introduction

Where learning tasks are closely related, it seems reasonable to expect a learner to be able to improve its performance on a particular learning task by reapplying knowledge gained on some previous learning task. The learner should, we feel, be able to *transfer* knowledge from one task to another. Unfortunately, popular

are exactly reverse of what we want: the acquisition of new knowledge appears to catastrophically interfere with existing knowledge [4].

Many workers are engaged in the attempt to realise the benefits of knowledge tranfer within learning [cf. 5, 6, 7, 8].[2] However, there seems to be some residual fuzziness in our thinking about the relationship between transfer and learning. In particular, different assumptions are made about the way in which these two processes interact.

In some cases the role of learning is simply rote storage (i.e., memorisation) of presented data. However, in most cases learning involves going beyond presented data, i.e., it involves some form of induction. Where the goal of learning is some form of behaviour then producing high performance means doing the right thing at the right time. But we can, of course, always see this as a kind of induction simply by treating the motor commands to be learned as the 'target outputs' in a conventional induction problem.

If we accept the idea that learning can usually be viewed as some sort of inductive process, we have to ask how transfer fits in. A common view is that transfer is an operation which takes place between learning tasks. This suggests that the process is somehow independent and separated from normal inductive activity. On the other hand, transfer seems pointless unless it contributes in some way to learning (i.e., inductive) performance. This seems to imply that we should view transfer as being a part of an higher-level inductive process.

There are thus conceptual problems to deal with whether we treat transfer as separate from induction or as closely integrated with it. To try to resolve these I present a task analysis of induction [9]. This differs from some theoretical treatments of learning (e.g., COLT treatments such as [10]) since it concentrates exclusively on properties of the induction problem and ignores possible solutions altogether. Interestingly, it leads to a view of induction which gives a clear role to a transfer process[3] and also allows us to formulate a criterion for deciding when and if such transfer has occurred. The paper thus provides theoretical ammunition for those who take the view that transfer should be treated as an aspect of induction rather than a separate

2 A task analysis of induction

Imagine we have a body of data D, as shown in Table 1. Each datum in D (i.e., each row) is made up of the values of variables x_1, x_2, x_3, x_4 and x_5. One of the values of x_3 is missing (see the '?' in the x_3 column). Can we use the other data to predict this missing value? In other words, can we empirically *induce* the missing value from the data which are provided?

x_1	x_2	x_3	x_4	x_5
c	d	f	a	b
a	b	h	d	b
e	c	h	d	e
c	b	f	a	e
a	c	f	d	e
c	c	?	a	e
b	c	f	a	e
b	d	h	d	e
e	d	f	a	c
a	c	h	d	c
c	d	h	a	c

Table 1: Sample induction problem.

If we observe that every possible value of the relevant variable has the same probability then we clearly cannot make any prediction at all. If all values do *not* have the same probability then we will rationally predict the missing value to be the one which has the highest observed probability. However, there are several ways in which we can work out 'observed probabilities'. First, we can look at the unconditional probability of seeing a particular value v of x_i.

$$P(x_i = v)$$

In the present case this is not productive since both possible values of x_3 have the same unconditional probability. This is just the chance value of 0.5, i.e.,

$$P(x_i = v) = \frac{1}{|V|}$$

where V is the set of all possible values of x_i.

Second, we can look at the probability of seeing a particular value conditional on explicit instantiations of the other values, i.e.,

$$P(x_i = v_a | x_j = v_b ...)$$

where v_a and v

practice it may be hard to allocate a particular method to a particular category.

A small number of cases *can* be conclusively classified within the scheme. The ID3 method [16], now more often used in its updated manifestation as C4.5 [17] is a case in point. ID3 takes a training set of sample input/output pairs from an input/output mapping, and constructs a decision tree (for generating outputs) by recursively partitioning the training set until every pair in a given partition has the same output value.

At each stage of the process, a new partitioning is constructed by dividing up the cases in an existing partition according to which value they have on the variable whose values are most strongly associated (within the partition) with specific output values. This has the effect of maximising the output-value uniformity of new partitions and thus minimising (subject to horizon effects)

and obvious role to knowledge transfer. Arguably, it *requires* that transfer play a role. Recall that exploitation of relational effects involves the identification of relationships in the data. Since in general the space of possible relationships is infinite this identification necessarily involves a bias. A learner seeking to exploit relationships in the data must always have some particular relationships 'in mind.' Thus the learner uses assumptions regarding the relevance or salience of relationships. These assumptions constitute 'knowledge' which, if it is justified at all, must be justified in terms of prior, relevant experience. Relational learning, then, is either unjustified or based on knowledge transfer.

The implication is worth spelling out. A well justified, relational learning process applied to a sequence tasks *necessarily* engages in knowledge transfer. The process can be viewed in terms of the acquisition of a suitable bias. This

learning. The proposed model thus offers some real benefits for the achievement of a better understanding of transfer. Whether it has any worth for those engaged in the application of transfer in practical contexts remains to be seen.

References

[1] Rumelhart, D., Hinton, G. and Williams, R. (1986). Learning representations by back-propagating errors. *Nature*, *323* (pp. 533-6).

[2] Murre, J. (Forthcoming). Transfer of learning in backpropagation and in related neural network models. In J. Levy, D. Bairaktaris, J. Bullinaria and P. Cairns (Eds.), *Connectionist Models of Memory and Language*. London: UCI Press.

[3] Harvey, I. and Stone, J. (1995). Unicycling helps your french: spontaneous recovery of associations by learning unrelated tasks. CSRP 379, School of Cognitive and Computing Sciences, University of Sussex.

[4] McCloskey, M. and Cohen, N. (1989). Catastrophic interference in connectionist networks: the sequential learning problem. In G.H. Bower (Ed.), *The Psychology of Learning and Motivation*. New York: Academic Press.

[5] Schmidhuber, J. (1996). A theoretical foundation for multi-agent learning and incremental self-improvement in unrestricted environments. In X. Yao (Ed.), *Evolutionary Computation: Theory and Applications*. Singapore: Scientific Publishing Company.

[6] Caruana, R. (1995). Learning many related tasks at the same time with backpropagation. *Advances in Neural Information Processing Systems 7* (Proceedings of NIPS-94) (pp. 657-664).

[7] Martin, J. and Billman, D. (1994). Acquiring and combining overlapping concepts. *Machine Learning*, *16* (pp. 1-37).

[8] Pratt, L. (1994). Experiments on the transfer of knowledge between neural networks. In S. Hanson, G. Drastal and R. Rivest (Eds.), *Computational Learning Theory and Natural Learning Systems, Constraints and Prospects* (pp. 523-560). MIT Press.

[9] Thornton, C. (1995). Measuring the difficulty of specific learning problems. *Connection Science*, *7*, No. 1 (pp. 81-92).

[10] Kearns, M. (1990). *The Computational Complexity of Machine Learning*. The MIT Press.

[11] Shavlik, J. and Dietterich, T. (Eds.) (1990). *Readings in Machine Learning*. San Mateo, California: Morgan Kaufmann.

[12] Michalski, R., Carbonell, J. and Mitchell, T. (Eds.) (1983). *Machine Learning: An Artificial Intelligence Approach.* Palo Alto: Tioga.

[13] Michalski, R., Carbonell, J. and Mitchell, T. (Eds.) (1986). *Machine Learning: An Artificial Intelligence Approach: Vol II.* Los Altos: Morgan Kaufmann.

[14] Thornton, C. (1994). Statistical biases in backpropagation learning. *Proceedings of the International Conference on Artificial Neural Networks* (pp. 709-712). Sorrento, Italy.

[15] Dietterich, T., London, B., Clarkson, K. and Dromey, G. (1982). Learning and inductive inference. In P. Cohen and E. Feigenbaum (Eds.), *The Handbook of Artificial Intelligence: Vol III.* Los Altos: Kaufmann.

[16] Quinlan, J. (1986). Induction of decision trees. *Machine Learning, 1* (pp. 81-106).

[17] Quinlan, J. (1993). *C4.5: Programs for Machine Learning.* San Mateo, California: Morgan Kaufmann.

[18] Breiman, L., Friedman, J., Olshen, R. and Stone, C. (1984). *Classification and Regression Trees.* Wadsworth.

[19] Rumelhart, D. and Zipser, D. (1986). Feature discovery by competitive learning. In D. Rumelhart, J. McClelland and the PDP Research Group (Eds.), *Parallel Distributed Processing: Explorations in the Microstructures of Cognition. Vol I* (pp. 151-193). Cambridge, Mass.: MIT Press.

[20] Kohonen, T. (1984). *Self-organization and Associative Memory.* Berlin: Springer-Verlag.

[21] Diday, E. and Simon, J. (1980). Clustering analysis. In K. Fu (Ed.), *Digital Pattern Recognition.* Commu7ln. 151-193). Cambrihbrih Ca12Td980).19.9203T[.6(an0Td[(RtJ
andMachineVVVH.11990Td([(o,)-1004)100

[25] Langley, P., Simon, H., Bradshaw, G. and Zytkow, J. (1987). *Scientific Discovery: Computational Explorations of the Creative Processes.* Cambridge, Mass.: MIT Press.

[26] Wolff, J. (1978). Grammar discovery as data compression. *Proceedings of*