Assessing Generalization in Connectionist and Rule-based Models Under the Learning Constraint

Thomas R. Shultz (shultz@psych.mcgill.ca)
Department of Psychology; McGill University
Montreal, QC H3C 1B1 Canada

Abstract

Although it is commonly assumed that rule-based models generalize more effectively than do connectionist models, the comparison is often confounded by pitting hand-written rules against learned connections. Three case studies from cognitive development show that, under the constraint that both types of models learn their representations from equivalent examples, generalization is consistently superior in connectionist models.

Generalization Problems

A significant part of the ongoing debate between rule-based and connectionist modeling in psychology has focused on the ability to generalize. A common claim from supporters of the classical, symbolic approach is that rule-based models are superior because they generalize more effectively than do connectionist models (Ling & Marinov, 1993; Pinker, 1997; Marcus, 1998). Generalization is considered important by most modelers because it distinguishes understanding of a problem from mere memorization of solutions.

The generalization ability of rules is often enhanced by the use of variables that can be bound to any number of objects or events. Consider the following rule, written in Common Lisp for a production system program. It generates correct responses on some Piagetian conservation of number problems:

```
((response more ?x ?y)
 (and
  (initially-same-number ?x ?y)
  (or (add1 ?x)
      (subtract1 ?y))))
```

The rule says to conclude that row *x* has more items than row *y* if the two rows initially had the same number of items and if one item was subsequently either added to row *x* or subtracted from row *y*. It has plenty of generality because the variables *x* and *y* can be bound to any rows with any number of items. It could be made even more general by adding a third variable *n*, representing the number of items added or subtracted.

More generally, a rule can be defined as a conditional statement in which conjunctively and disjunctively connected conditions, if verified as true, produce a set of conjunctively connected conclusions. Each condition and conclusion is a proposition that can be stated in predicate-argument form, where arguments can be constants, variables, or other propositions.

Leaving aside the issue of whether people actually generalize as well as such rules do, the claim has commonly been made that connectionist models rarely learn to generalize that well. Indeed, this argument seems to have been accepted by many connectionists (e.g., Anderson, 1995), and is at least partly responsible for the many attempts to improve generalization in neural network learning (e.g., Reed & Marks, 1995).

However, closer inspection reveals a serious confound in this argument. The symbolic rules are often written by hand, or perhaps merely alluded to, while the neural network learns its own connection weights by processing examples. The purpose of this paper is to remove this confound between representation and learning by requiring both types of model to learn their representations from equivalent examples. It is already well known that an alternate method of removing this confound, by hand-designing neural networks to explicitly implement rules and variables, also produces excellent generalization (Shastri, 1995).

The learning constraint proposed here is reminiscent of the developmental tractability constraint proposed by Klahr (1984). In discussing cognitive development, Klahr argued that any two plausible, consecutive developmental states must be integrated in a transition theory that can transform one state into the other. Similarly, and a bit more generally, I propose a constraint that acquired knowledge representations, whether rules or weight vectors, must be learned by a model in order to be considered plausible. Knowledge representations that are instead hypothesized to be produced through biological evolution may be dealt with by hand designing rule-based and connectionist models as noted earlier, or even more ambitiously, by simulated evolution. Covering both inherited and acquired representations is the more general principle that other, non-representational features must be held constant when assessing generalization ability. Otherwise claims about superior generalization ability may be confounded with acquisition issues and possibly other differences.

Choice of Algorithms and Domains

A systematic test of generalization under a learning constraint should eventually involve many algorithms and problem domains. To begin this process, this paper compares one leading connectionist algorithm to one leading rule-learning algorithm in three different domains of cognitive development.

One of the most frequently used connectionist algorithms in cognitive development, and the principal one used in my laboratory, is cascade-correlation (CC). CC creates feed-forward networks by recruiting new hidden units that correlate well with network error and installing them in cascaded layers (Fahlman & Lebiere, 1990). It has been used to simulate a wide variety of cognitive developmental phenomena, including conservation (Shultz, 1998), seriation (Mareschal & Shultz, 1999), the balance scale (Shultz, Mareschal, & Schmidt, 1994), shift learning (Sirois & Shultz, 1998), pronoun acquisition (Oshima-Takane, Takane, & Shultz, 1999), infant familiarization to rule-governed sentences (Shultz & Bale, 2001), and integration of the concepts of velocity, time, and distance for moving objects (Buckingham & Shultz, 2000).

Choosing an equivalent rule-learning algorithm encounters the problem that there are not all that many successful rule-based models of cognitive development, in the sense of implementing developmental transitions. A good case can be made that the largest number of successful rule-based developmental models have been achieved by the C4.5 algorithm (Quinlan, 1993) and its immediate predecessor ID3 (Quinlan, 1986). These include models of English past tense morphology (Ling, 1994; Ling & Marinov, 1993), the balance scale (Schmidt & Ling, 1996), grammar learning (Ling & Marinov, 1994), and reading (Ling & Wang, 1996). There is also a simulation of non-conscious acquisition of rules for visual scanning of a matrix (Ling & Marinov, 1994), and numerous applications in engineering and decision support (Quinlan, 1993). Among alternative symbolic rule-learning algorithms applied to the same phenomenon, the balance scale, C4.5 produced an arguably superior model.

C4.5 learns to classify examples described with features and values by forming a smallish decision tree that can be converted into production rules. It is a greedy (i.e., non-backtracking) algorithm that repeatedly finds the most informative feature with which to classify so far unclassified examples.

There are a number of intriguing similarities between C4.5 and CC. Both algorithms use supervised learning of examples, focus on largest current source of error, gradually construct a solution based on what is already known, and aim for a small solution that generalizes well. In this paper, I report on generalization performance of the CC and C4.5 algorithms on the three problems of conservation acquisition, number comparison, and infant familiarization to sentences in an artificial language.

Conservation Acquisition

A recent CC model of conservation acquisition focused on Piaget's conservation of number problems (Shultz, 1998). In one version of these problems, a child first agrees that two rows have the same number of items, and is then asked which row has more after one of the rows is transformed, for example, by compression. Children below about six years of age typically judge the longer row to have more items, whereas older children correctly judge the rows to remain equal. The vast psychological literature on conservation (over 1000 studies) has produced a number of well-replicated regularities. Among them are acquisition (with a sudden jump in performance), the problem size effect (with better performance and earlier success on small number problems than on large number problems), length bias in pre-conservation children (choosing the longer row as having more), and the screening effect (with young children giving a correct answer to a screened transformation until the screen is removed).

CC networks were trained on 420 examples of number conservation problems of row lengths and densities ranging between 2 and 6, with number of items being the product of length and density. Using inputs coding the length and density of each row, both before and after the transformation, the identity of the transformed row, and the identity of the transformation (addition, subtraction, compression, and elongation), networks learned to judge whether the rows had equal numbers or not, after the transformation. Both equal and unequal initial rows were included. Length and density were coded as real numbers, and the other inputs were coded in a localist binary fashion. There were 100 test problems of the same type, not used in training, to assess generalization performance.

C4.5 was trained with the same examples, learning to classify them into three numerical judgments: one row has more, the other row has more, or both rows have the same. C4.5 was equipped with ability to deal with continuous, as well as qualitative inputs,[1] and to use the option for information gain ratio, which is generally superior to simple information gain (Quinlan, 1993).

Proportions correct on training and test problems, respectively, were 1.0 and .95 for 20 CC networks, and .40 and .35 for 20 C4.5 trees. For both algorithms, generalization performance (on the test problems) was just a bit worse than performance on the training problems; but training and generalization performance was much higher for CC than for C4.5. If the learned

[1] C4.5 finds the gain ratios for each possible cutoff on a continuous feature and then chooses the partition of examples with the highest gain ratio in the usual way.

knowledge representation is inadequate, it does not afford good generalization. This makes a pure test of generalization ability difficult. To control for learning success, proportion correct on the test problems can be divided by proportion correct on the training problems, creating a generalization ratio. This ratio is .95 for CC and .87 for C4.5.

Because a failed model is not by itself very meaningful, I adopted the strategy of changing the input coding to C4.5 until learning was successful and then evaluating what is required to learn in terms of both theoretical plausibility and psychological coverage.

Following the lead of other C4.5 modelers (Schmidt & Ling, 1996), I coded the length and density input in relational, rather than absolute terms. For example, was the first row longer or shorter or the same length as the second row? Although this relational coding produced 100% success on training and test problems, it created knowledge representations that are unlike any that have been reported with children. For example, an English gloss of one of the smaller rules is: *If the first row is longer than the second row before the transformation, and shorter than the second row after the transformation, then the first row has more items.*

Because of this exclusive focus on relative length and density of the rows, there was never any reference to information on the transformation or the identity of transformed row. Nor could the C4.5 models cover any of the various psychological regularities. This is in distinction to both the CC model and children, characterized by a shift from concern with how the rows look to the nature and identity of the transformation. The CC model also covers all of the psychological regularities mentioned: sudden jump in acquisition, problem size effect, length bias, and the screening effect. Thus, although relational input coding can produce perfect learning and generalization in C4.5, it creates implausible knowledge representations and fails to cover the psychological data. In contrast, the CC model can learn and generalize effectively from raw input coding, acquire knowledge representations that are similar to those seen in children, and cover the psychological regularities.

Number Comparison

One of the most basic of numerical skills is that of comparing the size of two numbers. Prominent psychological regularities in number comparison are the min and distance effects. The min effect refers to earlier success and quicker performance the smaller the smaller of the two numbers. The distance effect refers to earlier success and quicker performance the larger the absolute difference between the two numbers.

My simulations focus on pairs created from the integers 0-9. In a study of interpolation, a randomly selected 50 pairs comprised the training set and the remaining 50 pairs comprised the test set. The integers were coded as real numbers, and there were three discrete output classes, including ties. Mean proportion correct on training and test problems, respectively, over 20 runs was 1.0 and 1.0 for CC and .75 and .66 for C4.5. The mean generalization ratio of test correct to train correct was 1.0 for CC and .89 for C4.5. Not only did CC learn the problem and generalize more effectively than did C4.5, but only CC captured the min and distance effects.

Knowledge representation analysis revealed a sensible solution for CC networks that involved positioning a hyper-plane near the diagonal axis designated by $x = y$, where x and y are the two numbers being compared. The fact that this hyper-plane is anchored at the origin and drifts away from the ideal diagonal at the higher values generates the min effect. The soft boundary created by the sigmoid activation function in CC networks produces the distance effect. In contrast, the rules learned by C4.5 made no psychological sense, e.g., *If $x > 5$ and $y > 7$, then $x > y$.*

Another coding trick employed by C4.5 modelers uses the difference between two numbers that are being compared (Schmidt & Ling, 1996). Mean proportions correct on training and test problems, respectively, were .902 and .875 for C4.5 difference coding in the interpolation experiment. This is an improvement, but again there is no coverage of the min and difference effects, and the rules are psychologically inappropriate, e.g., *If difference > 1, and $y > 2$, then $y > x$.*

In a study of extrapolation, the models were trained on pairs of the integers 0-4 and tested on pairs of the integers 5-9. There is no variation in C4.5 performance here because training patterns are not randomly selected for each run. Training and test results are shown in Table 1. Again, the CC algorithm learns and generalizes better than the C4.5 algorithm, whether input coding uses standard raw integers or differences.

Table 1: Proportion correct and generalization ratio for extrapolation.

Algorithm/coding	Train	Test	Ratio
CC	1.00	.99	.99
C4.5/standard	.56	.40	.71
C4.5/difference	.76	.40	.53

In conclusion, C4.5 does not learn or generalize well with either standard or difference coding of input on number comparison problems. It also fails to cover the min and difference effects, and the rules it learns are psychologically implausible. The only apparent way to get C4.5 to learn appropriate number comparison rules and generalize effectively is to build those rule conditions into the input coding, in which case there is nothing to learn. In contrast, CC learns and generalizes well, while covering min and difference effects and generating reasonable knowledge representations, and it does so with raw numerical inputs.

Infant Familiarization to Sentences

The third case study concerns infant familiarization to sentences in an artificial language. A recent paper in this area has been of particular interest because it claimed to have data that could only be accounted for by rules and variables (Marcus, Vijayan, Rao, & Vishton, 1999). That study found that 7-month-olds attend longer to sentences with unfamiliar structures than to sentences with familiar structures. Particular features of the experimental design and some unsuccessful neural network models allowed the authors to conclude that unstructured neural networks cannot simulate these results. Several unstructured connectionist models have since disproved that claim (Shultz & Bale, 2001), but the current focus is on generalization ability of connectionist and rule-based models that learn representations of these sentences.

The present simulations focus in particular on Experiment 1 of Marcus et al. (1999). In this experiment, infants were familiarized to sentences with an ABA pattern, for example, *ga ti ga* or *li na li*. There were 16 of these ABA sentences, created by combining four A words (*ga*, *li*, *ni*, and *ta*) and four B words (*ti*, *na*, *gi*, and *la*). Subsequently, the infants were tested with two novel sentences that were consistent with the ABA pattern (*wo fe wo*, and *de ko de*) and two others that were inconsistent with ABA in that they followed an ABB pattern (*wo fe fe*, and *de ko ko*). There was also a condition in which infants were familiarized instead to sentences with an ABB pattern. Here the novel ABB sentences were consistent and the novel ABA sentences were inconsistent with the familiarized pattern. Infants attended more to inconsistent than to consistent novel sentences, suggesting that they were sensitive to syntactic properties of the sentences.

For consistency, I focus on a particular CC model of these data (Shultz & Bale, 2001). In this model, sentences were coded by real numbers representing the sonority (vowel likeness) of particular consonants or vowels. An encoder version of CC was used, enabling the network to learn to reproduce its inputs on its output units. Deciding on whether a particular sentence is correctly rendered in such networks is somewhat arbitrary. A more natural index of performance on training and test sentences is mean error, which is plotted in Figure 1. Test patterns inside the range of the training patterns were the same as those used with infants. Two additional sets tested extrapolation by using sonority values outside of the training range, by a distance that was either close or far. The greater error to inconsistent sentences corresponds to the attention difference found with infants. The fact that this consistency effect extends to patterns outside of the training range reveals substantial extrapolation ability in these networks. As well, the CC networks exhibited the typical exponential decrease in attention to familiarization stimuli that are found with infants.

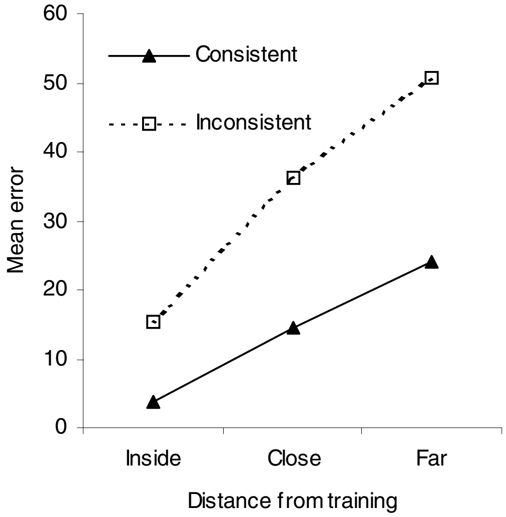

Figure 1: Mean error for CC networks simulating infant interest in consistent and inconsistent test sentences.

I did C4.5 simulations in several different ways to try to achieve successful learning and generalization. The initial attempt involved a literal symbolic encoding of each word in the sentences. For example, the word *ga* was coded as the symbol *ga*. Because there was only one output class when only one type of sentence was used as in the infant experiment (ABA or ABB), the resulting decision tree had only one leaf labeled with the syntactic class. In other words, if exposed only to ABA sentences, then expect more of the same. This is not really a rule and it captures none of the gradual characteristics of familiarization in infants. There is no variation in any of these C4.5 runs of the familiarization problem because each run uses all of the examples, rather than a random selection of examples.

The next C4.5 simulation added the 16 ABB sentences to the examples to be classified, in order to ensure that rules would be learned. This effectively changes the experiment to one of discrimination rather than familiarization. In this case, C4.5 focused only on the third word, concluding that the ABA syntax would be signaled by *ga*, *li*, *ni*, or *ta* as the third word, whereas the ABB syntax would be identified by *ti*, *na*, *gi*, or *la* as the third word. This is a sensible focus because the third word does distinguish the two syntactic types, producing a training success rate of 1.0, but it does not reflect Marcus et al.'s (1999) assumptions about infants comparing the first and third words in each sentence. Moreover, because the test words are novel, this solution does not enable distinction between consistent and inconsistent test sentences. The generalization success rate is 0, as is the generalization ratio.

To obtain successful generalization with this kind of literal symbolic coding in C4.5, it is necessary to code the input relationally, explicitly representing equality of the first and second words, the first and third words, and the second and third words. When the first and third words are the same, then one has an ABA sentence; when the second and third words are the same then one has an ABB sentence. This allows perfect generalization to novel words, but the problem is that C4.5 can learn this relation perfectly with only one example of each pattern because the entire solution is explicitly presented in the inputs. Infants presumably require more examples than that to distinguish these syntactic patterns, reflecting the fact that their inputs are not coded so explicitly and fortuitously.

C4.5 was also trained with discrimination examples coded on sonority values as in the CC model. This model yielded 62.5% of training sentences correct, 0% correct on ABA and ABB test sentences, and a generalization ratio of 0. Moreover, the rules learned by this model were rather odd, e.g., *If C1 < -5, C3 < -5, and C2 > -6, then syntax is ABA*, where *C1* refers to the consonant of the first word, *C3* is the consonant of the third word, etc.

In contrast, the knowledge representations learned by the CC model were psychologically interesting. The hidden units were found to use sonority sums of the consonant and vowel to represent variation in sonority. This was achieved first in the duplicated-word category and next in the single-word category. This hidden unit representation was then decoded with similar weights to outputs representing the duplicate-word category.

Summarizing the results of the familiarization simulations, C4.5 did not show gradual familiarization effects. When the problem was changed to a discrimination problem, C4.5 did not learn the proper rules and did not generalize effectively. With explicit relational coding, C4.5 learns and generalizes perfectly, but it requires only two examples. When trained with sonority codes, C4.5 does not master the training examples, learns inappropriate rules, and does not generalize. In contrast, CC learns and generalizes well, both inside and outside of the range of the training examples, and acquires sensible knowledge representations.

Discussion

When learning of knowledge representations is required, CC reveals a number of advantages over C4.5: familiarizing to a single category, learning both simple (number comparison) and difficult (conservation) problems, finding structural relations that exist implicitly within training examples, learning rule-like functions that are psychologically plausible, covering psychological effects, and generalizing to novel examples, even to the extent of extrapolating outside of the training range. A pure comparison of generalization is difficult because of differences in learning success. However, comparison of generalization ratios that scale test performance by training performance, to control for learning success, consistently showed an advantage for CC over C4.5. This advantage occurred both with identical input coding for the two algorithms and with a variety of coding modifications that made it easier for C4.5 to learn.

Some of the generalization success of CC networks can be traced to the use of analog coding of inputs. In analog codes, the amount of unit activation varies with the intensity of the property being represented. Analog codes are well known to facilitate learning and generalization in artificial neural networks (Jackson, 1997), and exploratory comparative simulations suggest that they were important determinants of the present results. Their use in some of the present simulations can be justified by psychological evidence that people also employ analog representations, for example, of number.

Analog coding is not the entire story, however, because of two considerations. One is that not all of the CC inputs were analog. Some of the inputs to conservation problems that are essential to mature knowledge representations are coded in a discrete binary fashion. A second qualifier is that analog input codes were insufficient to allow successful learning and generalization in C4.5 models, even though C4.5 is equipped to deal with continuous inputs.

For a learning system to generalize effectively, it must of course learn the right sort of knowledge representation. This is why the present results show a close correspondence between success on the training examples and generalization performance. It was typical for performance to be slightly worse on test problems than on training problems, although generalization was considerably worse in some C4.5 runs, as indicated by low generalization ratios.

Because connectionist models generalized better than rule-based models under the learning constraint in three different domains, the argument that rule-based models show superior generalization is highly suspect. However, it is reasonable to ask whether connectionist models invariably generalize better than rule-learning models. Would this finding hold up in different domains and with different learning algorithms? Obviously, more research is needed, but we are now beyond facile comparisons of hand-written or imagined rules to laboriously learned connections.

Choice of algorithm is a key issue because both symbolic and neural algorithms may vary considerably in their ability to learn and generalize. Certainly, CC benefits from its ability to learn difficult problems that are beyond the ability of other neural learning procedures and its tendency to build the smallest network necessary to master the problem on which it is being trained. Likewise, C4.5 benefits from its use of information gain to select the best feature on which to partition unclassified examples. Both algorithms have led the way in their respective class in producing successful simulations of cognitive development. Nonetheless, it is important for other algorithms of each type to be tried. It is possible that other rule-learning algorithms would have better success in finding more abstract and thus more general knowledge representations than C4.5 does. Although C4.5 is adept at learning from examples, it seems unable to represent those examples in anything more abstract than the features used in their input descriptions. This limitation could make learning and generalization difficult.

Finally, it is important to stress that generalization ability should not be taken as the ultimate criterion on which to evaluate different cognitive models. Surely, it is more critical to determine whether a given model generalizes like human subjects do. This is an issue that has not yet been adequately addressed.

Acknowledgments

This research was supported by a grant from the Natural Sciences and Engineering Research Council of Canada. Yoshio Takane, Alan Bale, and Francois Rivest contributed insightful comments on an earlier draft.

References

Anderson, J. A. (1995). *An introduction to neural networks*. Cambridge, MA: MIT Press.

Buckingham, D., & Shultz, T. R. (2000). The developmental course of distance, time, and velocity concepts: A generative connectionist model. *Journal of Cognition and Development, 1*, 305-345.

Fahlman, S. E., & Lebiere, C. (1990). The cascade-correlation learning architecture. In D. S. Touretzky (Ed.), *Advances in neural information processing systems 2* (pp. 524-532). Los Altos, CA: Morgan Kaufmann.

Jackson, T. O. (1997). Data input and output representations. In E. Fiesler & R. Beale (Eds.), *Handbook of neural computation*. Oxford: Oxford University Press.

Klahr, D. (1984). Transition processes in quantitative development. In R. J. Sternberg (Ed.), *Mechanisms of cognitive development*. New York: Freeman.

Ling, C. X. (1994). Learning the past tense of English verbs: The symbolic pattern associator vs. connectionist models. *Journal of Artificial Intelligence Research, 1*, 209-229.

Ling, C. X., & Marinov, M. (1993). Answering the connectionist challenge: A symbolic model of learning the past tenses of English verbs. *Cognition, 49*, 235-290.

Ling, C. X., & Marinov, M. (1994). A symbolic model of the nonconscious acquisition of information. *Cognitive Science, 18*, 595-621.

Ling, C. X., & Wang, H. (1996). *A decision-tree model for reading aloud with automatic alignment and grapheme generation*. Unpublished paper, Department of Computer Science, University of Western Ontario.

Marcus, G. (1998). Rethinking eliminative connectionism. *Cognitive Psychology, 37*, 243-282.

Marcus, G. F., Vijayan, S., Rao, S. B., & Vishton, P. M. (1999). Rule learning by seven-month-old infants. *Science, 283*, 77-80.

Mareschal, D., & Shultz, T. R. (1999). Development of children's seriation: A connectionist approach. *Connection Science*, *11*, 149-186.

Oshima-Takane, Y., Takane, Y., & Shultz, T. R. (1999). The learning of first and second pronouns in English: Network models and analysis. *Journal of Child Language*, *26*, 545-575.

Pinker, S. (1997). *How the mind works*. New York: Norton.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning, 1*, 81-106.

Quinlan, J. R. (1993). *C4.5: Programs for machine learning*. San Mateo, CA: Morgan Kaufmann.

Reed, R., & Marks II, R. J. (1988). Neurosmithing: Improving neural network learning. In M. A. Arbib (Ed.), *The handbook of brain theory and neural networks*. Cambridge, MA: MIT Press.

Schmidt, W. C., & Ling, C. X. (1996). A decision-tree model of balance scale development. *Machine Learning, 24*, 203-229.

Shastri, L. (1995). Structured connectionist models. In M. A. Arbib (Ed.), *The handbook of brain theory and neural networks*. MIT Press, Cambridge, MA.

Shultz, T. R. (1998). A computational analysis of conservation. *Developmental Science*, *1*, 103-126.

Shultz, T. R., & Bale, A. C. (2001, in press). Neural network simulation of infant familiarization to artificial sentences: Rule-like behavior without explicit rules and variables. *Infancy*.

Shultz, T. R., Mareschal, D., & Schmidt, W. C. (1994). Modeling cognitive development on balance scale phenomena. *Machine Learning*, *16*, 57-86.

Sirois, S., & Shultz, T. R. (1998). Neural network modeling of developmental effects in discrimination shifts. *Journal of Experimental Child Psychology, 71*, 235-274.