A Symbolic Model of the Nonconscious Acquisition of Information*

Charles X. Ling[†] **Marin Marinov**[‡]

Abstract

The paper presents counter evidence against Smolensky's theory that human intuitive/nonconscious cognitive processes can only be accurately explained in terms of subsymbolic computations carried out in artificial neural networks. We present *symbolic* learning models of two well studied, complicated cognitive tasks involving nonconscious acquisition of information: learning production rules and artificial finite state grammars. Our results demonstrate that intuitive learning does not imply subsymbolic computation, and that the already well-established, perceived correlation between "conscious" and "symbolic" on the one hand, and between "nonconscious" and "subsymbolic" on the other, does not exist.

Preprint. *Cognitive Science*, 18(4), 1994 (in press).

---

*The authors thank gratefully Herb Simon, Dave Touretzky, Billy Schmidt, Paul Thagard, and Danny Silver for their detailed and useful comments on the earlier draft of the paper. Charles Ling acknowledges the support from NSERC Research Grant.

†Department of Computer Science, The University of Western Ontario, London, Ontario, Canada N6A 5B7. Corresponding author. Email: ling@csd.uwo.ca.

‡Department of Philosophy, Sofia University, 15 Ruski Bd., Sofia-1000, Bulgaria.

1 Introduction

The advent of connectionism has sparked a far reaching debate on the nature of human cognitive architecture. Fodor and Pylyshyn (1988) argued that connectionism cannot offer an alternative to the classical Turing/von Neumann cognitive architecture; at best, connectionist systems can serve as possible models for implementing the classical architecture in artificial neural networks (ANNs). Smolensky (1988) has challenged Fodor and Pylyshyn's criticisms, and with his "On the proper treatment of connectionism", has tried to establish connectionism as a new paradigm in cognitive science (also see (Smolensky, 1987, 1991)). He proposes the theory that human cognition could be explained with the help of two processors — the "conscious rule interpreter" and the "intuitive processor". He admits that certain cognitive phenomena can be approximately explained within the symbolic paradigm, but the explanatory power of the symbolic paradigm is limited only to tasks that require conscious rule application. Smolensky refers to the mechanism for behaviour that is not conscious rule application as the *intuitive processor*. According to Smolensky's *Intuitive Processor Hypothesis*, "[The intuitive processor] is presumably responsible for all of animal behaviour and a huge portion of human behaviour: Perception, practiced motor behaviour, fluent linguistic behaviour, intuition in problem solving and game playing — in short, practically all skilled performance" (Smolensky, 1988, page 5). Smolensky claims that the intuitive processor operates on consciously inaccessible subconcepts and is performing subsymbolic computations which can be modeled accurately by artificial neural networks, and that the symbolic paradigm will give only an imprecise and approximate explanation of this vast majority of nonconscious or intuitive cognitive processing.

While Smolensky's hypothesis (that human skilled performance can only be accurately modeled by subsymbolic, rather than symbolic processes) badly requires supporting evidence, we provide counter evidence in this paper. We show that symbolic methods *can* accurately model typical and widely studied complicated intuitive learning processes, which are traditionally believed to be amenable only to connectionist modelling. There seems to exist, in the field of cognitive science, a perceived correlation between conscious cognitive tasks and the symbolic paradigm, and between nonconscious cognitive tasks and the subsymbolic paradigm. For example, Cleeremans and McClelland (1991) implemented a Simple Recurrent Network (SRN) to model the intuitive learning of finite state grammars. They believe that the connectionist model is highly appropriate for modeling the implicit learning phenomena because the acquired knowledge is stored in its connections, which is not comprehensible and accessible to humans.

In this paper, we argue that intuitive learning of rules should better be modeled by symbolic rule-learning programs, and we present a complete symbolic model of intuitive learning of production rules that replicates the actual human results reported by Lewicki, Czyzewska, and Hoffman (1987). The results of our simulations support our long-standing point of view (also see (Ling & Marinov, 1993; Schmidt & Ling, 1994)) that different algorithms underlie different cognitive processes, and at the same time, that one and the same algorithm can be responsible for conscious and nonconscious processes alike.

2 Nonconscious Acquisition of Production Rules

Theoreticians from Helmholtz to Chomsky and Marr have postulated that cognitive processes might be the result of the unconscious processing of complicated rules. It is a striking fact that we cannot articulate most of the rules that govern our use of language but we are perfectly capable of distinguishing between a sentence of our language that sounds "appropriate" and a sentence that sounds "odd". In many cases people can object or accept a line of reasoning without being able to articulate the logical rules underlying their decision. Yet, relatively little is known about the human ability to learn complicated rules unconsciously. Several recent studies of the phenomenon of nonconscious acquisition of information suggest that humans have the capacity to learn fairly complicated rules in a short time without any conscious awareness that they have actually learned them (Kihlstrom, 1987; Lewicki, 1986; Lewicki et al., 1987; Lewicki, Hill, & Bizot, 1988; Lewicki, Hill, & Czyzewska, 1992; Reber, 1989; Stadler, 1989). However, these studies were aimed mainly at establishing the pervasiveness of the phenomenon of nonconscious acquisition of information and have not been complemented by a serious attempt to uncover the cognitive mechanisms that underlie this remarkable human ability. We will try to fill the gap existing in the experimental literature and to model some of these cognitive mechanisms.

2.1 Lewicki, Czyzewska, and Hoffman's Experiment

Overview.

Lewicki et al.'s experiment was based on a matrix-scanning paradigm (Lewicki, 1986).

The subjects' task was to react to the appearance of a target, the digit 6, on a computer terminal by pressing one of four keys corresponding to the quadrant of the target. Each block of trials consisted of six simple identification trials followed by one matrix scanning trial. In the *simple identification trials* the screen was divided by one horizontal and one vertical line into four equal regions (see figure 1 (a)); when the target 6 appeared in one of the four regions, the subjects had to press a key corresponding to the region in which the target appeared. In the *matrix scanning trials*, the target appeared as one digit in a matrix of 36 digits. See figure 1 (b) for an example of one of the matrix scanning trials.

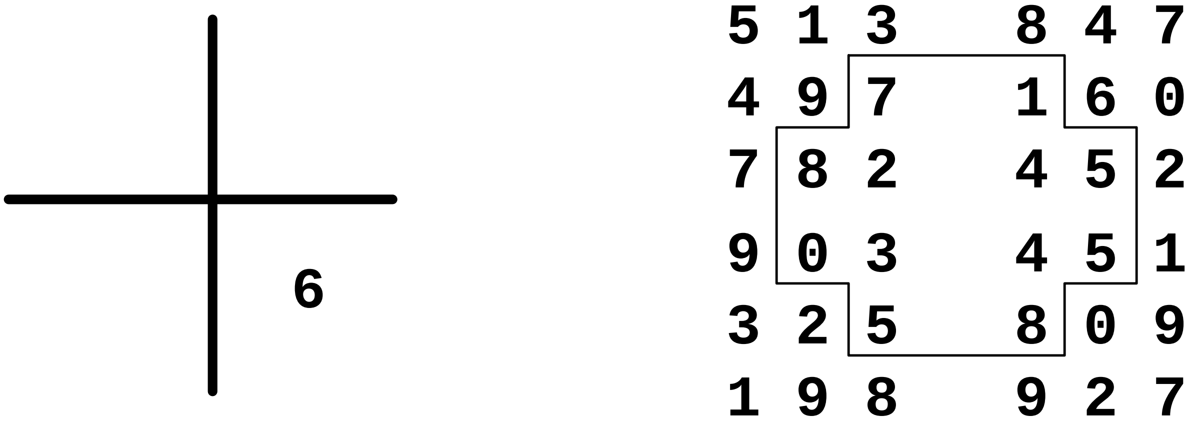


Figure 1: (a) Simple identification trials (b) Matrix-scanning trials.

Unlike the simple trials in which the target appeared close to the fixation point, the matrix-scanning trials were designed so that the search for the target required the subjects to move their eyes (i.e., the target never appeared in the foveal area), and therefore involved a decision where to look first. These permissible locations for the target are outside the lines in Figure 1(b). The reaction times for the matrix-scanning trials were recorded for each subject throughout the experiment. Three subjects were tested in 4,608 blocks of trials (one block = six simple trials followed by one matrix scanning trial). All blocks of trials were divided into 48 segments of 96 consecutive blocks. That is:

1 block = 6 simple trials + 1 matrix-scanning trial

1 segment = 96 consecutive blocks

The whole experiment = 48 segments = 4,608 blocks

The response times for each subject in the matrix scanning trials were averaged for each segment (i.e. over 96 matrix-scanning trials).

Data.

The data for the experiment were generated according to a quite complicated disjunction of propositional rules (see Table 1). In Table 1 each row represents an IF-THEN clause and the table as a whole represents a disjunction of 24 IF-THEN clauses. For example the first row of the table encodes the following rule:

IF

in Simple Trial #1 the target is in the Lower Left Quadrant, and
in Simple Trial #3 the target is in the Upper Left Quadrant, and
in Simple Trial #4 the target is in the Lower Right Quadrant, and
in Simple Trial #6 the target is in the Upper Right Quadrant

THEN

in the Matrix-Scanning Trial the target will be in the Upper Left Quadrant.

Using these basic 24 IF-THEN rules a sequence of 4,608 blocks of trials (6 simple followed by 1 matrix scanning trial) was randomly generated for each of the three subjects. The subjects were not told that the 2nd and 5th positions of the target in Simple trials were randomly chosen and hence do not determine the position of the target in the matrix-scanning trial. The subjects did not know anything about the purpose of the experiment or that the stimuli were created according to rules. The only task that they had to perform is to record the position of the target each time by pressing a key as quick as they can.

Aims.

The aims of the experiment were twofold. The first aim of the experiment was to find out whether the subjects can somehow infer the rules underlying the seemingly unrelated stimuli. If the subjects were able to guess all or even some of the rules, then the times it takes to identify the target during the matrix scanning trials would decrease — if the subjects guess correctly the quadrant in which the target will appear, they need not embark on an exhaustive search involving eye movements. So, if the subjects are guessing correctly the position of the target the overall reaction time should decrease.

This strategy will work only if we had some means of distinguishing the actual learning of the underlying rules from the unspecific learning for skill improvement that occurs during

Table 1: Rules for the determining of the target in the Matrix-Scanning Trials.

IF (Simple Trials)						THEN
#1	#2	#3	#4	#5	#6	Matrix Scan Trial
3	R	1	4	R	2	1
3	R	4	2	R	1	1
2	R	4	3	R	1	1
4	R	1	2	R	3	1
1	R	2	4	R	3	1
2	R	3	1	R	4	1
2	R	4	1	R	3	2
1	R	3	4	R	2	2
1	R	2	3	R	4	2
4	R	3	1	R	2	2
4	R	2	3	R	1	2
3	R	1	2	R	4	2
4	R	1	3	R	2	3
1	R	3	2	R	4	3
4	R	2	1	R	3	3
3	R	2	1	R	4	3
2	R	3	4	R	1	3
1	R	4	3	R	2	3
3	R	2	4	R	1	4
3	R	4	1	R	2	4
1	R	4	2	R	3	4
2	R	1	3	R	4	4
2	R	1	4	R	3	4
4	R	3	2	R	1	4

1 - upper left; 2 - upper right; 3 - lower left; 4 - lower right; and R - random location.

any such experiments involving a repetitive task (see Section 3.4). That is why Lewicki et al. decided to reverse the rules for the generation of the data at the 43rd segment. All blocks of trials after the 43rd segment were generated by rules which had the same IF part as the original 24 but the THEN part was pairwise reversed — upper left was substituted for lower left and vice versa; and upper right was substituted for lower right and vice versa. If the subjects were successful in inferring the correct rules up to the 43rd segment, at the 43rd segment and beyond their formerly correct guesses would lead them to look for the target in the wrong place. The response time should slow dramatically because it takes longer to recover from the wrong guess and to embark on a search for the target that involves eye movements.

The second aim of the experiment was to find out through post-experimental interviews whether the subjects were aware that they had learned any rules or whether they even suspected that any regularities existed in the data.

2.2 Experimental Results

The results from the experiment are presented in Figure 2.

As can be seen from the figure, the individual response times differ for the three subjects but they all follow a common pattern. They all gradually decrease up to segment 42. Then in segment 43 they all register a significant increase — due to the change in the rules generating the data. Then the response time remains relatively high or begins to decrease gradually. The differences in the average mean response latency for segments 38-42 and 43-47 was shown by Lewicki et al. to be statistically significant (ANOVA test).

The results of this experiment clearly indicate that the subjects were able to infer (at least a significant part of) the appearance of the target during the experiment. That this is no trivial task can be seen if we reflect on the number of possible hypotheses that the subjects have to rule out — since in each block there are six simple trials with four possible outcomes each, there are altogether 4,096 possible six-condition-rules to choose from. Finding out that only 24 four-condition rules in Table 1 are actually relevant for the prediction of the target during the matrix-scanning trials is indeed, a daunting task. Moreover, none of the subjects were told to expect any regularities in the data; they had no way of knowing that the outcomes of the simple trials somehow control the outcomes of the last matrix-scanning trial in each block, not to say anything about the independence

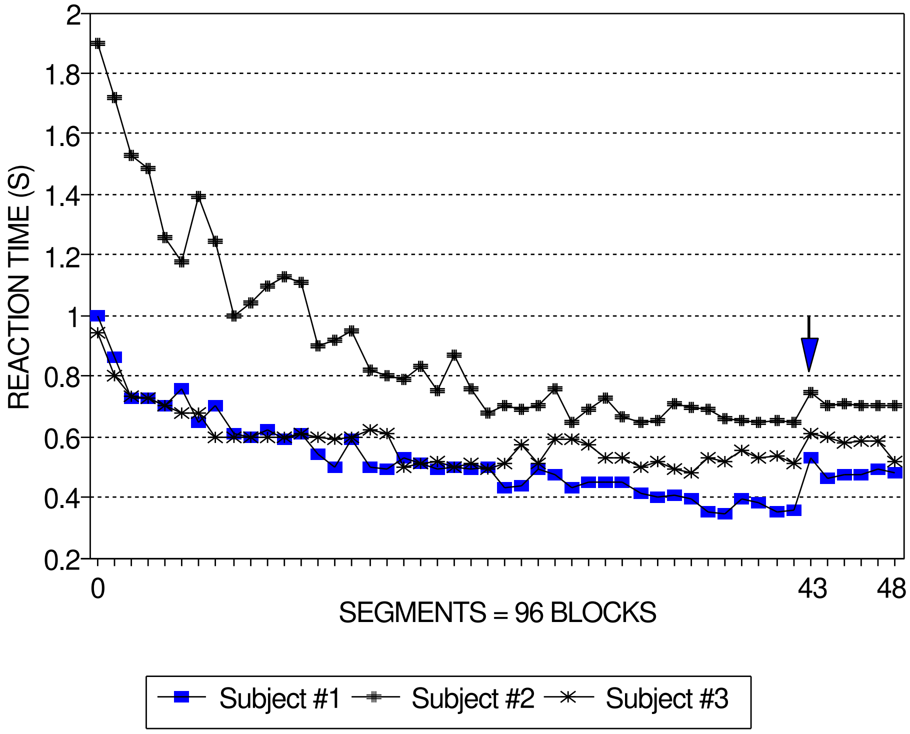


Figure 2: Results of Lewicki et al (1987) experiments.

restrictions on each block of trials known by the experimenters but not by the subjects. Yet, despite these facts, the subjects' performance was clearly influenced by the change of rules after the 43rd segment, indicating that they have somehow learned to predict the appearance of the target in the matrix-scanning trials prior to that.

Another very interesting result of this experiment was a further confirmation that this kind of highly intelligent processing of information was apparently completely nonconscious. The subjects were unable to recall learning any rules for predicting the target location during the first 42 segments. The subjects reported that they did not notice any regularities in the data. They also were completely unaware that they were learning from these regularities to guess the correct position of the target!

It is natural to interpret this experiment as evidence that humans are capable of learning successfully complicated rules in an intuitive (nonconscious) fashion, and that they are able to consult what they have learnt to predict future instances without conscious awareness of doing so.

3 A Symbolic Model of Nonconscious Acquisition of Production Rules

As we saw, there exists overwhelming evidence that humans can learn to predict outcomes controlled by extremely complicated production rules from examples without any conscious awareness that such learning is taking place. Whether the human brain contains a specific "intuitive processor" as Smolensky has hypothesized or not, it is plausible that some cognitive mechanism (or class of such mechanisms) might be responsible for this startling human learning ability. Our results will demonstrate that such possible cognitive processors need not be connectionist processors, and the opaque representation of knowledge in weights of ANNs does not give them warrant or special power of nonconscious learning. Instead, we present a symbolic process that models accurately the cognitive mechanisms underlying the nonconscious or intuitive acquisition of information. In fact, we will use a symbolic learning algorithm to construct a complete model that succeeds in replicating the actual human results (the reaction time) reported by Lewicki et al. We will argue that such a rule-based symbolic model is more psychologically realistic than the ANN model for this task.

The learning task that the human subjects had to face in Lewicki et al.'s experiment can be categorized as supervised learning from examples — each block of six trials followed by the presentation of the target during the matrix-scanning trial can be regarded as one training example, and each prediction of the position of the target during a matrix-scanning trial after learning from such training examples can be regarded as testing examples. Several symbolic learning algorithms (such as ID3 (Quinlan, 1986, 1993), CN2 (Clark & Niblett, 1989), AQ (Michalski, Mozetic, Hong, & Lavrac, 1986)) have been well studied. We decided to use the most popular one of these algorithms — ID3 — in our modeling of the nonconscious/intuitive learning.

3.1 The ID3 Symbolic Learning System

ID3 (Quinlan, 1986) is a descendant from Hunt, Marin and Stone's CLS (Hunt, Marin, & Stone, 1966) and falls into the class of TDIDT (top-down induction of decision trees) learning systems. C4.5 (Quinlan, 1993), an improved implementation of ID3, is used in our modeling. Both ID3 and pattern-association neural networks belong to supervised learning systems that use propositional representation. In addition, neither of them is application specific.

To see how the decision-tree building algorithm of ID3 works, it is useful to look at a concrete example (from (Quinlan, 1986)). Assume the weather of the day is described by the temperature and humidity and so on, and the decision is whether the day is suitable or not for playing golf (P or N). That is, each day can be described with the help of the following attributes:

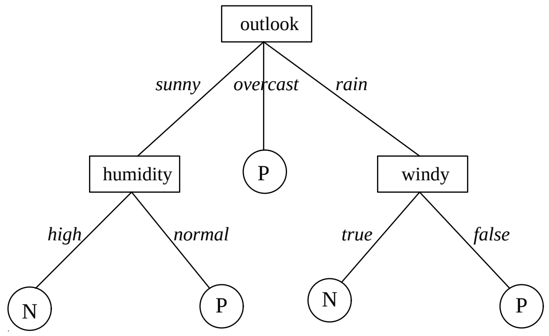
- outlook: with values = {sunny, overcast, rain}.
- temperature: with values = {cool, mild, hot}.
- humidity: with values = {high, normal}.
- windy: with values = {true, false}.

Suppose we provide ID3 with the set of training examples in Table 2. First, ID3 builds a decision tree that classifies all examples in the training set, and then this tree can be used to predict future instances not included in the training set. ID3 builds decision trees by employing a simple divide and conquer strategy. This strategy is recursively applied in building subtrees, until all remaining examples in the training set belong to a single concept (class); then the terminal leaf (circles in the figures of decision trees) is labeled as that concept.

The divide and conquer strategy works as follows: ID3 first picks an attribute based on the information gain of the attributes (see below) — in this case the attribute outlook — and checks its values. As it turns out, all objects with value overcast for the attribute outlook belong to the play golf class, so ID3 closes this branch and assign P to a leaf. In case a value does not classify the subset of examples falling into this branch as belonging to a single class, the same procedure is applied to the subset recursively, looking at another attribute and its values and building a subtree for the subset. The algorithm halts when all subsets of objects are classified into their corresponding single classes. In this way ID3 builds the following simple decision tree to classify all of the training examples in Table 2.

Table 2: A small training set.

Attributes				Class
Outlook	Temperature	Humidity	Windy	
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	P
rain	mild	high	true	N



But how does ID3 choose which attribute to start building the subtree with? For example, if ID3 chose the first node to be the attribute humidity, a larger and much more complex decision tree would be constructed. ID3 uses a heuristic principle according to the Occam's Razor Principle: it uses the information gain criterion to choose the *most discriminative* or *most informative* attribute as the root of a (sub)tree, producing a *small* decision tree. Choosing most discriminative attributes also reduces the entropy or randomness in the remaining data the most. For a set A of k examples, if p_1 of them belong to the class P, and p_2 of them belong to the class N ($p_1 + p_2 = k$), then the *entropy* (i.e., randomness) of the set is:

$$E(A) = \frac{p_1}{k}\log_2\frac{k}{p_1} + \frac{p_2}{k}\log_2\frac{k}{p_2}$$

Clearly $E(A)$ reaches the minimum value 0 when $p_1 = k$ or $p_2 = k$ (i.e. this is a single class and therefore no more classification is needed). Likewise, $E(A)$ reaches the maximum value 1 when $p_1 = p_2 = k/2$ (i.e. one cannot make any better prediction than the 50/50 chance). The formula can be extended to cases with more than two classes easily.

If an attribute X has values $x_1, x_2, ...x_v$ and is chosen as a root of the (sub)tree, the set

A will be split into v subsets $B_1, B_2, ..., B_v$ according to the values of the attribute X. The entropy of the resulting decision tree $E(X)$ (with X as a root) is a weighted sum of the entropies of v subsets. That is:

$$E(X) = \frac{|B_1|}{k}E(B_1) + ... + \frac{|B_v|}{k}E(B_v)$$

To select the most informative attribute, ID3 chooses an attribute X among the remaining attributes such that the resulting entropy $E(X)$ is minimized, or the information gain $E(A) - E(X)$ is maximized[1]. This results in trees that generally branch out from the more informative to the less informative attributes, thus simplifying the overall tree structure. Since the heuristic is applied at one level at a time, it does not guarantee to yield the smallest decision tree. However, the use of this heuristic results in building *small* decision trees which in general have good generalization power. Also, ID3 scales up well; it is polynomial with respect to the total number of attributes, the number of attribute values, and total tree size.

It is easy to see that the decision tree beside Table 2 can immediately be converted into a set of production rules just by tracing all of its branches from the root to the terminal leafs — the terminal leaf is the consequent and the conjunction of all decision nodes composes the body of the rule. The rules produced from the tree beside Table 2 have the same predictive accuracy as the decision tree itself:

If outlook = overcast, then play golf.
If outlook = sunny and humidity = normal, then play golf.
If outlook = rain and windy = false, then play golf.
If outlook = sunny and humidity = high, then do not play golf.
If outlook = rain and windy = true, then do not play golf.

To summarise, ID3 aims at producing small decision trees which can be converted into small sets of production rules. These rules are in the form of the disjunction of the conjunction of the attributes, where disjunction is formed by having several conjunctive rules with the same consequence. Note that ID3 only classifies one outcome from a set of attributes. Ling and Marinov (1993) extended ID3 to SPA, a general purpose Symbolic Pattern Associator that learns to predict from the input pattern (a set of attributes) to the output pattern (a set of outcomes).

[1]To balance different numbers of values among attributes, ID3 actually uses a gain ratio instead of the gain directly.

Hunt et al. (1966) performed a series of experiments comparing their CLS programs — antecedents of ID3 — with human subjects on various classes of concepts. They found that the relative difficulty of concepts learned by humans and by CLS-1 was in general the same: small conjunctive concepts were relatively easy to learn, inclusive disjunction and implication were the next easiest, and exclusive disjunctions, which require many conjunctive rules to represent, were the most difficult. These results show that CLS is quite psychologically realistic. On the other hand, the standard Perceptron (ANNs without hidden units) learning algorithm learns well the class of linearly separable concepts, many of which are difficult to learn by humans. Admittedly, humans easily learn some linearly separable functions (such as "at least m out of n" function) that ID3 cannot generalize well. Therefore, it seems that the variety of concepts that humans can learn well lies somewhere in between ID3 and ANN's learning abilities.

3.2 The Intuitive Learning Component of the Model

At an abstract level, the ID3 learning algorithm is ideally suited for rule learning tasks of the intuitive learning component in Lewicki et al.'s experiment. However, in order to provide a psychologically realistic model of *the reaction time* in their experiment, we need to take account of additional factors not directly connected with learning the rules to predict the position of the target. In particular, we hypothesize that the reaction time measures of the human subjects throughout the experiment depend on three different factors[2].

1. *The Intuitive Learning Component.* The successful predicting of the position of the target as a result of intuitive learning during previous segments.
2. *The Exhaustive Search Component.* The exhaustive search (involving eye movements) of the target in case the prediction in the intuitive learning is wrong.

[2]Unfortunately, Lewicki et al. (1987) did not test the predictive accuracy in the matrix-scanning trials by human subjects; otherwise, the intuitive learning component itself would be sufficient to verify our modeling. On the other hand, it becomes more interesting to model the reaction time by adding two more components. Note that to model the reaction time with ANN models, one has to consider the components of exhaustive search and unspecific learning as well. For example, Cleeremans and McClelland (1991) used a simple linear equation to approximate the reaction time from the predictive accuracy of a recurrent neural network.

3. *The Unspecific Learning Component.* The unspecific learning for skill improvement (such as reaction of fingers in key stroke) that occurs during such experiments.

We used the rules listed in Table 1 to create the data set for our machine learning experiment. The procedures for creating the data were exactly the same as if we were generating a new set for a human subject for the Lewicki et al. experiment — the 24 rules used in human learning experiment were used to generate 48 segments of 96 consecutive blocks each = 4,608 blocks altogether. Each block had the following general form:

$v_1, v_2, v_3, v_4, v_5, v_6 \rightarrow c$

where v_i represents the position of the target in the six simple trials, and c represents the position of the target in the following matrix-scanning trials, and $v_i = c = \{1, 2, 3, 4\}$ where 1 stands for upper left, 2 for upper right, 3 for lower left, and 4 for lower right. Therefore, each block is effectively a training example for ID3.

Initially, we conducted the simulation in the following way: at each iteration, ID3 (we actually used an improved implementation of ID3 called C4.5 (Quinlan, 1993)) was given a new segment of trials (i.e., 96 testing examples), and it learned to predict the position of the target in the matrix-scanning trial based on its current decision tree. ID3's prediction was checked against the actual outcome and its success/failure on each example was recorded. Then, a new decision tree was built with the correct values for the matrix-scanning trial of the current segment, plus all of the previous examples. The process iterated with new segments of trials, until all 42 segments were exhausted. We expected that as with the human subjects, the initial guesses of ID3 would not be better than random. But as more and more training examples with their correct outcomes were being "seen" by ID3 and its decision trees built in accordance with the correct outcomes, ID3's performance would improve dramatically.

However, there were two problems in our initial plan. Pilot experiments showed that ID3 learned too quickly and too well compared with human subjects. With only 1 segment of 96 randomly-generated training examples, ID3 often learned perfect rules for predicting the position of the matrix-scanning trial. The major reason that human subjects cannot learn as fast and as accurately as ID3 does, we believe, is that they cannot remember and use many previous blocks of trials for learning, especially in the experiments where trials are presented at very fast pace. Also without forgetting the previous blocks, data from 43rd segment are contradictory to the overwhelmingly large number of previous ones, the prediction from the 43rd segment would be overridden by the data before 43rd segment. Thus we would not see the drop in the reaction of human subjects. As a psychologically realistic assumption we assumed that the recent blocks of trials are more likely to be retained in short-term memory and that the outcomes of past blocks of trials are gradually forgotten or decayed during the experiment (cf. (Gleitman, 1986)). We adopted the following "forgetfulness" formula:

Data up to segment$_{i+1}$ = (data up to segment$_i$) $\times$ c + 1/4 $\times$ (the current segment)

This means that only a portion of examples (chosen randomly) in the current segment are retained in memory, and the earlier ones (chosen randomly) are gradually forgotten with a "decay factor" c ($0 < c < 1$) as new information comes in. We chose $c = 0.80$ in our experiments.

The second problem is that, for each iteration, the tree was built from scratch with the currently available examples, rather than updated from the previously-built trees. This problem can be avoided easily by using incremental versions of ID3 — ID4 (Schlimmer & Fisher, 1986) or ID5R (Utgoff, 1989). Indeed, Utgoff (1989) showed that ID5R produces the *same* decision tree as the original ID3 on the same training data sets. The basic idea of ID5R is that, each time when a new training example is presented, it checks to see if the tree can classify the new examples while remaining "balanced" (most discriminative attributes as roots of all subtrees). If not, the tree will be expanded with an additional testing attribute, and then restructured to remain "balanced". Each decision node keeps classification counts for unused attributes, which are used and updated during tree updating without re-examining the training examples. The restructuring process pulls up the desired attribute (the most discriminative one) to the top node of the subtree, and is applied to its subtrees recursively. In our experiment, the removal of (1-c) percent of examples in the current decision tree due to decaying may also require to restructure the tree. Since the same decision trees can be induced incrementally, for the purpose of convenience, we continued to use the original ID3 in our experiments.

We ran many experiments (since the previous data are removed randomly, each run has a different training set) and the outcomes are very stable: the accuracy rate of ID3 on predicting new unseen blocks increases gradually and stabilizes. Learning happens mainly in two forms: first, the sizes of the available training sets for ID3 become larger as segments of data are presented. Second, the decision tree is updated and becomes a more accurate predictor, since early decision trees are built from small training sets and are quite different from the trees built with more data. In the beginning of the 43rd segment when all four rules are reversed, the error rate of ID3 rises sharply to 100%. It begins to drop in the subsequent segments as ID3 "forgets" data from the previous segments and learns to predict the outcomes of the blocks between 43rd and 48th segments from the new data. Note that since the conditions of the rules remain valid, the structure of the decision tree is largely unchanged except the labeling of the leaves. See Table 3 for the result of a typical run of the simulation.

Table 3: ID3 performance on the Lewicki et al.'s rule set.

Segment	E	Segment	E	Segment	E	Segment	E
1	51.2%	13	0.0%	25	7.5%	37	4.7%
2	42.1%	14	3.4%	26	7.1%	38	3.4%
3	40.0%	15	8.0%	27	11.5%	39	2.2%
4	32.1%	16	5.0%	28	3.6%	40	4.2%
5	30.2%	17	10.3%	29	3.5%	41	8.3%
6	21.2%	18	4.2%	30	3.8%	42	3.5%
7	20.2%	19	7.5%	31	7.8%	**43**	**100.0%**
8	15.2%	20	3.4%	32	4.2%	44	82.8%
9	9.1%	21	8.0%	33	2.0%	45	58.1%
10	10.4%	22	3.5%	34	3.8%	46	30.7%
11	9.0%	23	0.0%	35	0.0%	47	20.5%
12	4.5%	24	12.2%	36	0.0%	48	8.4%

Note: E = Average Error Rate on predicting unseen matrix-scanning trials.

If we look at the structure of the decision tree of ID3 with low predictive error rates (such as at segment 13), we can see that the algorithm discovers the original production rules that generated the data. See Figure 3 for a complete branch of part of the decision tree. In addition, the decision tree can be automatically converted into a set of simplified production rules (Quinlan, 1993):

IF $v_1 = 3$, $v_3 = 1$, $v_4 = 4$ THEN $c = 1$
IF $v_1 = 1$, $v_3 = 2$, $v_4 = 4$ THEN $c = 1$
IF $v_1 = 2$, $v_3 = 4$, $v_4 = 3$ THEN $c = 1$
IF $v_1 = 2$, $v_3 = 3$, $v_4 = 1$ THEN $c = 1$
IF $v_1 = 3$, $v_3 = 4$, $v_4 = 2$ THEN $c = 1$
IF $v_1 = 4$, $v_3 = 1$, $v_4 = 2$ THEN $c = 1$
IF $v_3 = 2$, $v_4 = 3$ THEN $c = 2$
IF $v_1 = 3$, $v_3 = 1$, $v_4 = 2$ THEN $c = 2$

IF $v_1 = 4,\ v_3 = 3,\ v_4 = 1$ THEN $c = 2$
IF $v_1 = 2,\ v_3 = 4,\ v_4 = 1$ THEN $c = 2$
IF $v_1 = 1,\ v_3 = 3,\ v_4 = 4$ THEN $c = 2$
IF $v_3 = 2,\ v_4 = 1$ THEN $c = 3$
IF $v_4 = 3,\ v_6 = 2$ THEN $c = 3$
IF $v_1 = 1,\ v_3 = 3,\ v_4 = 2$ THEN $c = 3$
IF $v_1 = 2,\ v_3 = 3,\ v_4 = 4$ THEN $c = 3$
IF $v_1 = 2,\ v_3 = 1$ THEN $c = 4$
IF $v_1 = 3,\ v_3 = 2,\ v_4 = 4$ THEN $c = 4$
IF $v_1 = 1,\ v_3 = 4,\ v_4 = 2$ THEN $c = 4$
IF $v_1 = 3,\ v_3 = 4,\ v_4 = 1$ THEN $c = 4$
IF $v_1 = 4,\ v_3 = 3,\ v_4 = 2$ THEN $c = 4$

where 1 = upper left; 2 = upper right; 3 = lower left; and 4 = lower right.

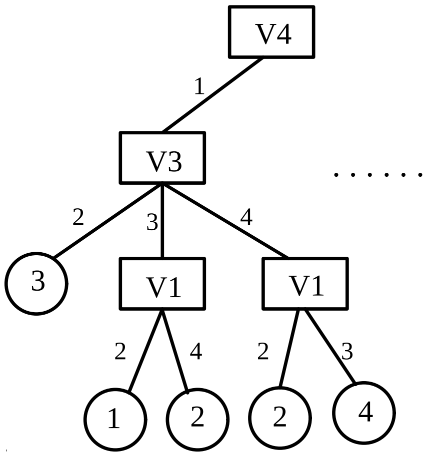


Figure 3: Portion of the decision tree produced by ID3 by segment #13.

If we analyze the decision tree and production rules produced, we find some very interesting results:

1. ID3 has quickly discovered that the 2nd and 5th positions of the target in the simple trials are irrelevant, since the values of v_2 and v_5 do not appear in any of the rules above. That is because they do not in any way affect the outcome of the matrix-scanning trials, and thus, they are not discriminative attributes.

2. Lewicki et al. (1987) assumed that all four simple trials (i.e., v_1, v_3, v_4 and v_6) are relevant to the prediction of the target in the matrix-scanning trials. However, among the four "relevant" simple trials, one simple trial is actually redundant! The rules discovered by ID3 have at most three conditions. For example, the 1st, 3rd and 4th simple trials are sufficient to predict the upper-left position of the target. There are no two blocks in which the 1st, 3rd, and 4th simple trials are the same but the target for the matrix-scanning trials are different.

3. Four rules produced by ID3 are even more general and simpler — they have only two conditions. More redundant conditions in the original rule set used in Lewicki et al.'s experiment are removed. Each 2-condition rule represents two 3-condition rules.

These facts indicate clearly that ID3 is a very good generalizer for this learning problem. With less than 25% of the total possible distinctive training examples, ID3 has already discovered the redundant information, produced a general and compact set of rules, and presented the acquired knowledge in the simple, comprehensible production rules. Therefore, we conclude that the symbolic learning algorithm is sufficient and very capable of modeling the underlying intuitive learning of the nonconscious acquisition of the production rules.

3.3 The Exhaustive Search Component

Even though ID3 can successfully serve as the core of the intuitive learning component, this component alone cannot predict the actual reaction time of the human subjects in Lewicki et al's experiment. This is because within the intuitive learning component we have no means of accounting for the time spent by the subjects for exhaustive search if the prediction is incorrect, and for the time gained due to unspecific learning for skill improvement that occurs during the actual experiment (see Section 3.4).

The exhaustive search component accounts for the consciously controlled exhaustive search. If a guess during a matrix scanning trial goes wrong, i.e., if the target is not in the expected quadrant, the subject has to initiate an exhaustive search which will always terminate with success but which involves a longer seek time. Clearly, the more errors that are made by the subject, the more often he/she has to initiate an exhaustive search. Obviously, if a subject's intuitive guesses are correct, his/her overall reaction time will be his/her guessing time. If, on the other hand, a subject's guesses are incorrect, the overall reaction time will be the sum of both the guessing time and of the exhaustive search time. The total time is simply the weighted sum of these two components, as captured by Equation 1:

$$T = t_1(1-E) + t_2E \tag{1}$$

where t_1 is a constant representing the subject's average guess time, and t_2 is the subject's average guessing plus the exhaustive search time; E is the average error rate of the prediction, representing the probability of incorrect guesses. Clearly, we have $t_2 > 2t_1$.

The intuitive learning and exhaustive search components together model the general trend exhibited by human subjects throughout Lewicki et al. experiment — initial poor performance, followed by improved performance which is sharply reversed later at the beginning of the 43rd segment when the rules generating the data are changed. However, simulating just this general trend is insufficient for a complete psychologically realistic model: the unspecific learning component for skill improvement has to be included too. Otherwise we could not explain why the drop in reaction time in the first 42 segments is much larger than the increase after the 43rd segment.

3.4 The Unspecific Learning Component

So far, in our model we have assumed that the exhaustive search component requires a constant seek time. There is overwhelming evidence, however, of the facilitating effect on similar repetitive search tasks due to *unspecific learning*. Based on evidence from many psychological experiments, Newell and Rosenbloom (1981) have conjectured the existence of a *Power Law of Practice* that characterizes the improvement of human performance during practice. According to this law, when human performance is measured in terms of the time needed to perform a task, it improves as a power-law function of the number of times the task has been performed, or formally

$$T = B\frac{1}{i^{\alpha}} \tag{2}$$

where T is time required to perform the task, i is the number of trials, and B and α are individually-dependent constants.

This law has been shown to hold across a wide range of experiments covering perceptual motor skills (Snoddy, 1926; Crossman, 1959), perception (Kolers, 1975; Niesser, Novick, & Lazar, 1963), motor behaviour (Card, English, & Burr, 1978), elementary decisions (Siebel, 1963), and problem solving (Neves & Anderson, 1981; Newell & Rosenbloom, 1981). In addition, in another nonconscious learning experiment involving the prediction of stimuli generated completely randomly, Cleeremans and McClelland (1991) show that the reaction time of human subjects also decreases. The curve matches broadly with the Power Law, and can be used to determine the parameters in it.

3.5 Simulation of the Human Results with the Complete Model

In order to explain the results of Lewicki et al.'s experiment we have to account for the three components influencing the reaction time — the guessing component, the exhaustive search component, and the unspecific learning component. Equation 3 best describes the interrelationship between the three components:

$$T = t_1(1 - E_i) + t_2 E_i + B\frac{1}{i^{\alpha}} \tag{3}$$

where E_i is the average error rate for the i-th segment; $t_1(1 - E_i)$ is the expected average guessing time at the i-th segment; $t_2 E_i$ is the expected average exhaustive search time at the i-th segment; and B/i^{α} is the unspecific learning facilitation measure at the i-th segment. t_1, t_2, B and α are subject specific constants and i is the segment number.

How can we test the empirical adequacy of our complete model? Ideally, if we were given the values of the *subject-specific* constants t_1, t_2, B and α for each of the subjects in Lewicki et al.'s experiment, and if by relying only on the accuracy rate recorded during the machine learning experiment with ID3 we were able to predict their individual reaction times, we would have clear evidence of the psychological reality of our complete model and of our model of intuitive learning in particular. Unfortunately, we do not know the exact values of t_1, t_2, B, and α for each of the subjects in Lewicki et al's experiment. However, we do know a possible range of psychologically realistic values and constraints for these constants which could correspond to different (possible) human subjects.

$t_1 = 100 - 300$ ms
$t_2 = 200 - 600$ ms
$t_2 > 2t_1$
$B = 500 - 2000$ ms
$\alpha = 0.1 - 0.5$

If we succeed in predicting the reaction times for the three subjects by choosing suitable constant values from these psychologically realistic bounds, we would have achieved our goal to demonstrate that our model can provide an accurate explanation of the mechanism of intuitive learning responsible for the subjects's performance during the experiment. We found the values of some of these constants for each of the three subjects by fitting several key turning points in their performance curve, and doing a parameter search for the rest. Figures 4, 5, and 6 show that there exist three sets of personal constants that together with the learning results from our machine learning experiment succeed in predicting almost exactly the reaction times of the three Subjects from Lewicki et al.'s experiment. Figure 7, on the other hand, demonstrates the reaction times for a range of possible "subjects" whose personal constants were drawn within the constraints and ranges specified above.

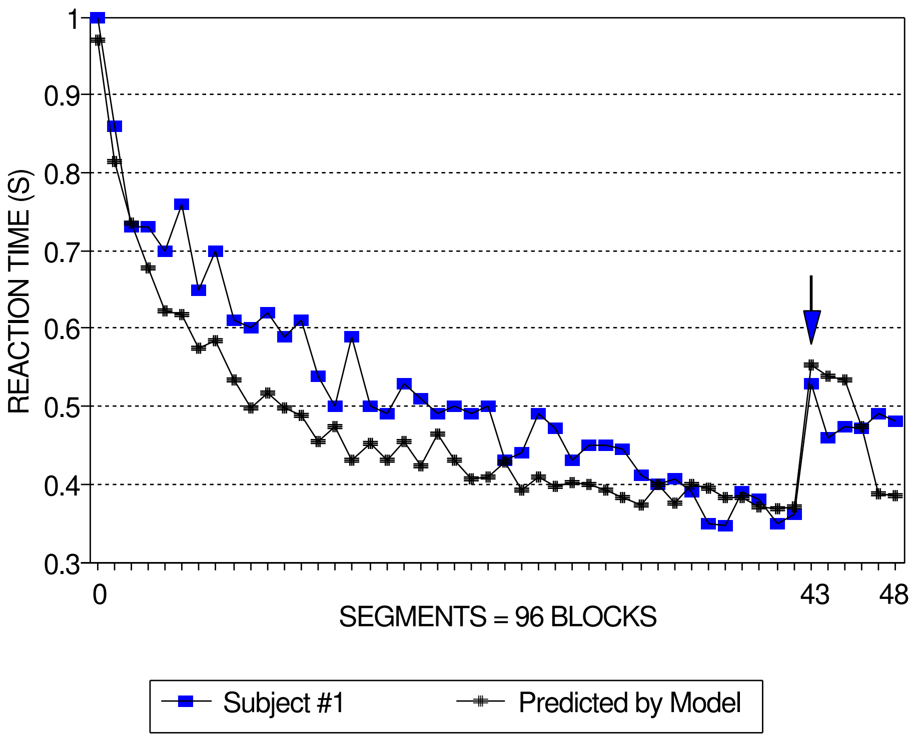


Figure 4: Simulation of human subject #1.

$t_1 = 150; t_2 = 350; B = 700; \alpha = 0.33.$

To summarize, our complete model predicts the reaction times of the three subjects as a function of the accuracy rate of the underlying propositional learning algorithm ID3. This indicates that decision tree learning algorithms like ID3 can serve as models of the

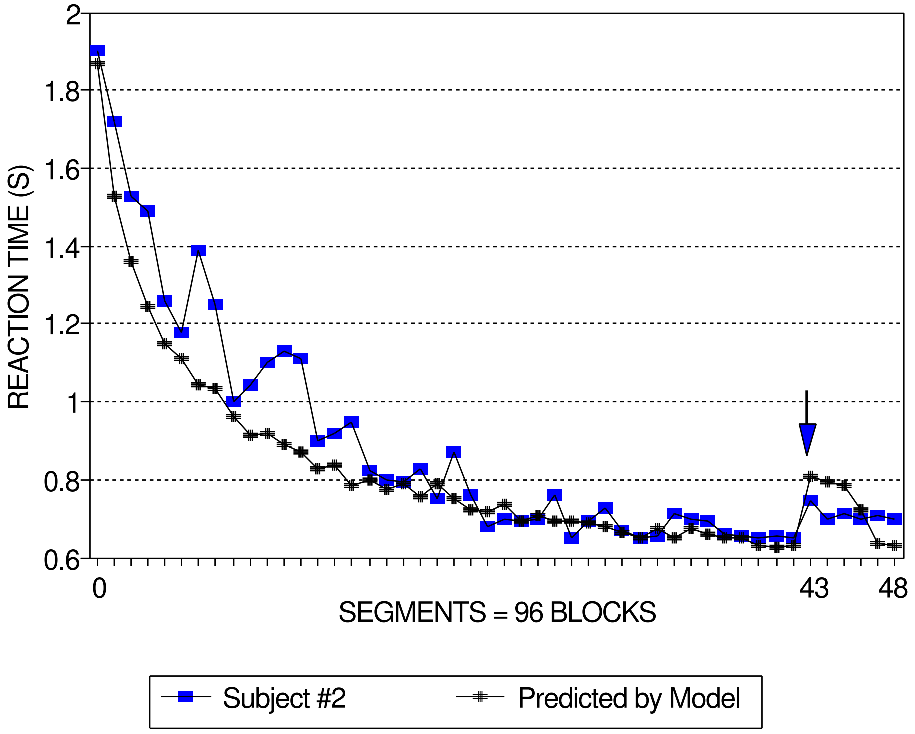


Figure 5: Simulation of human subject #2.

$t_1 = 150; t_2 = 350; B = 1600; \alpha = 0.33.$

intuitive or nonconscious acquisition of information in similar propositional-level learning tasks.

4 Symbolic Models for Intuitive Learning of Artificial Finite State Grammars

To further support our central claim that nonconscious learning does not imply subsymbolic computation, we briefly discuss the applicability of the symbolic learning system ID3 in another domain where nonconscious knowledge acquisition has been extensively studied (cf. (Reber, 1989)): learning finite state grammars. Cleeremans and McClelland (1991)

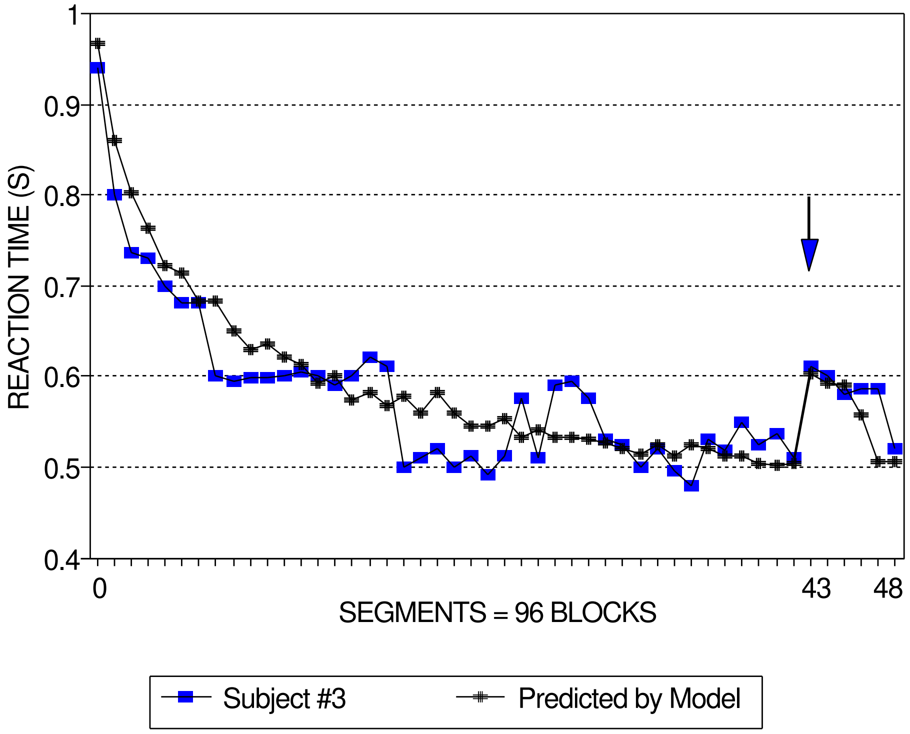


Figure 6: Simulation of human subject #3.

$t_1 = 100; t_2 = 210; B = 800; \alpha = 0.19.$

investigated and implemented a Simple Recurrent Network (SRN) to model the learning of the finite state grammar shown in Figure 8. They believe that the connectionist model is "highly appropriate for modeling implicit learning phenomena" because "all the knowledge of the system is stored in its connections", and "this knowledge may only be expressed through performance" (Cleeremans & McClelland, 1991, page 237) . Indeed, the trained ANNs with large number of weights are like black-boxes, which cannot be expressed easily in any comprehensible way (such as by production rules). We will show in this section that ID3 is perfectly able to model this task using the data generated from the same grammar.

Instead of demonstrating in details how a complete model can be constructed to replicate human performance as we did in the previous section, we only outline briefly the modeling of the intuitive learning component. A simplified version of the first experiment of Cleeremans and McClelland (1991) is described as follows: A string of six letters {T, S, X, V, P, Q} is generated using a fixed finite state grammar in Figure 8. The start state is 0. Whenever there is more than one arc that leads to the next state, one of them is chosen randomly. With the chosen transaction to the next state, the letter on the corresponding arc is generated. This process iterates. Each of the six letters corresponds to one of the six vertical positions on the screen, where a bright dot will flash momentarily. The subjects are asked to push one of the six buttons on the keyboard to react to the appearance of the dot. The reaction time is recorded. The entire experiment consists of 20 sessions, each consists of 20 blocks of 150 trials (stimuli). It is observed that the reaction time decreases when the subjects are exposed to more and more such sequences of stimuli or events.

There are several major differences between sequences generated from the production rules in the previous section and from the finite state grammar in this section. First, only the reaction time of the the seventh stimulus generated from the production rules is plotted; while the reaction time of every stimulus in a long sequence generated from the finite state grammar is recorded. Second, the seventh stimulus generated from the production rules is deterministic based on the previous six stimuli; while most stimuli in the sequence generated from the artificial finite state grammar are *nondeterministic* no matter how long the previous stimuli are remembered. Therefore, the learning error can never be close to zero; the best behavior is the one that matches with the conditional probability based on a certain number of previous stimuli.

To apply ID3 on this task, we first generated a string of the six letters according to the same finite state grammar, and then transferred the string to a set of classified training examples of a certain number of attributes (i.e., window size). Table 4 illustrates the sets of training examples based on the window size of two and three converted from a given sequence of stimuli. Applying ID3 on these training examples, decision trees/rules are built to predict the position of stimuli in the next session. Error rate in prediction is recorded, and the examples in the testing session are added into the training set. A similar "forgetness" formula is used so only recent stimuli are maintained in the set of training examples. Table 5 contains the average error rate of predicting positions of the stimuli in the testing session based on two and three previous stimuli. The curves for the error rate are shown in Figure 9. Clearly, the one using three previous stimuli has slightly lower overall error rate than the one using only two previous stimuli. From the error rate of the prediction, it is not difficult to construct a complete model as we did in Section 3.2.

To eliminate the need to predetermine the length of preceding letters (the window size)

Table 4: A sequence of stimuli and the two sets of training examples for ID3.

A sequence of stimuli: **P T V T V T V X P S ...**						
Training examples based on previous 2 stimuli			Training examples based on previous 3 stimuli			
Attributes		Class	Attributes			Class
P	**T**	**V**	**P**	**T**	**V**	**T**
T	**V**	**T**	**T**	**V**	**T**	**V**
V	**T**	**V**	**V**	**T**	**V**	**T**
T	**V**	**T**	**T**	**V**	**T**	**V**
V	**T**	**V**	**V**	**T**	**V**	**X**
T	**V**	**X**	**T**	**V**	**X**	**P**
V	**X**	**P**	**V**	**X**	**P**	**S**
X	**P**	**S**				

for predicting the current one, we can simply build a decision tree based on the previous n letters with a sufficiently large n. Since the training examples are quite noisy due to the nondeterministic property of the stimuli, a pruned (and small) decision tree (Quinlan, 1986) that avoids overfitting the training examples actually predicts better than a large tree. Experiments with $n = 10$ show that with a pruning level $c = 1\%$ (Quinlan, 1993), C4.5 produces a decision tree with 37 total nodes that reaches the best predictive accuracy. See Figure 10 for one actual pruned decision tree produced. Examining the resulting decision tree, we find that the most recent stimulus (S1) is used as the root of the tree, and with the 2nd preceding stimulus (S2) as the second level decision nodes. Clearly, ID3 has learned the obvious fact that the most recent stimulus is the most important factor, and the one before it the second most important factor for predicting the current stimulus. In addition, the decision tree only uses up to the 5th previous stimulus (S5). Therefore, we find immediately the optimal window size needed for this task is 5. In the recurrent network (Cleeremans & McClelland, 1991), although there is no need to explicitly specify the window size (always 1), one has to tweak the number of hidden units, which encode how far back the stimuli that the network needs to remember. Our method, however, determines the window size automatically without trial-and-error. As we also discuss in the Conclusion Section, the explicit representation that allows us to determine very easily the window size is one of the major advantages of symbolic learning algorithms.

Table 5: ID3 performance on the finite state grammar.

	Error rate in prediction	
Session	With 2 previous stimuli	with 3 previous stimuli
1	88.7%	88.7%
2	62.0%	64.7%
3	57.3%	34.7%
4	60.7%	46.7%
5	46.7%	44.7%
6	43.3%	45.3%
7	48.0%	43.3%
8	44.0%	46.7%
9	50.7%	40.7%
10	46.0%	42.0%
11	44.0%	39.3%
12	42.7%	46.7%
13	36.7%	37.3%
14	47.3%	42.0%
15	45.3%	47.3%
16	40.0%	40.7%
17	47.3%	44.7%
18	41.3%	38.7%
19	42.3%	43.3%
20	41.7%	40.3%

5 Discussions and the Conclusion

Although our symbolic simulations replicate the non-conscious learning of human subjects very well, we do not deny the possibility of replicating some of these tasks using artificial neural networks — it is quite conceivable that one can tweak and train a network to produce learning behaviors similar to those in Tables 3 and 5. Note that to model the reaction time with ANN models, one has to consider the components of exhaustive search and unspecific learning as we did in our symbolic modeling. Cleeremans and McClelland (1991) used a simple linear equation to approximate the reaction time from the predictive accuracy of the recurrent network. However, the fact that human subjects fail to articulate consciously what they have learned does *not* imply that results of the underlying learning mechanism have to be uncomprehensible as in artificial neural networks. It could be the case that the underlying learning mechanism is symbolic, and it controls the performance directly, and therefore, it is not directly accessible to human conscious thinking.

The theory that there is a single processor responsible for all nonconscious processes is also doubtful. We believe that different algorithms underlie different cognitive processes and at the same time it will not be surprising to find one and the same algorithm responsible for conscious and nonconscious processes alike. How well a learning algorithm models a cognitive process depends very much on how well the *bias* of the learning algorithm fits the underlying regularities of the task. ID3 attempts to learn compact decision trees and production rules, so it will model domains regulated by small decision trees or compact production rules better than artificial neural networks, as this is the case in learning production rules discussed in Section 3. ANN systems, on the other hand, model processes with continuous variables better, in general, than symbolic ones. Therefore, they may be more suitable for tasks such as motor control, image processing, and speech recognition.

There are other advantages of symbolic algorithms in modeling various cognitive tasks. Symbolic models have clear syntactic and semantic structures. Therefore it is relatively easy to set parameters of the learning program to capture the salient features of the cognitive process. The results from the learning programs are easy to comprehend, and can be represented explicitly in production rules. This allows for further processes and integrations of the acquired knowledge with other explicit knowledge and modules (Ling & Marinov, 1993; Ling, 1994). On the other hand, setting the right parameters (there are many of them) in artificial neural networks is often the main task to make the learning behavior right. These parameters include the network structure, the initial weights, the number and number of layers of hidden units, the learning rate, when to stop training, etc. However, there is no general guideline on how to set these parameters. In addition, the trained networks are usually difficult to understand. It is also hard for us to imagine how the acquired knowledge stored in weights can be integrated into higher level modules.

In sum, we have demonstrated that there exist symbolic models of intuitive learning tasks which involve nonconscious acquisition and processing of information previously believed to be amenable only to connectionist modelling. Our results question the perceived association between symbolic rule processing with conscious mental processing on the one hand, and the computations of ANNs with intuitive/nonconscious cognitive processes on the other.

Reference

Card, S., English, W., & Burr, B. (1978). Evaluation of mouse, rate controlled isometric joystick, step keys, and text keys for text selection on a crt. *Ergonomics*, *21*, 601–613.

Clark, P., & Niblett, T. (1989). The CN2 induction algorithm. *Machine Learning*, *3*(4), 261–283.

Cleeremans, A., & McClelland, J. (1991). Learning the structure of event sequences. *Journal of Experimental Psychology: General*, *120*(3), 235–253.

Crossman, E. (1959). A theory of the acquisition of speed-skill. *Ergonomics*, *2*, 153–166.

Fodor, J., & Pylyshyn, Z. (1988). Connectionism and cognitive architecture: A critical analysis. In Pinker, S., & Mehler, J. (Eds.), *Connections and Symbols*, pp. 3 – 71. Cambridge, MA: MIT Press.

Gleitman, H. (1986). *Psychology*. W.W. Norton & Company.

Hunt, E., Marin, J., & Stone, P. (1966). *Experiments in Induction*. New York, NY: Academic Press.

Kihlstrom, J. (1987). The cognitive unconscious. *Science*, *237*, 1445–1452.

Kolers, P. (1975). Memorial consequences of automatized encoding. *Journal of Experimental Psychology: Human Learning and Memory*, *6*, 689–701.

Lewicki, P. (1986). *Nonconscious Social Information Processing*. Academy Press: New York.

Lewicki, P., Czyzewska, M., & Hoffman, H. (1987). Unconscious acquisition of complex procedural knowledge. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *13*, 523–530.

Lewicki, P., Hill, T., & Bizot, E. (1988). Acquisition of procedural knowledge about a pattern of stimuli that cannot be articulated. *Cognitive Psychology*, *20*, 24–37.

Lewicki, P., Hill, T., & Czyzewska, M. (1992). Nonconscious acquisition of information. *American Psychologist*, 796–801.

Ling, C. X., & Marinov, M. (1993). Answering the connectionist challenge: a symbolic model of learning the past tense of English verbs. *Cognition*, *49*(3), 235–290. Anonymous ftp: /pub/ling/papers/cognition-verb.ps.Z on ftp.csd.uwo.ca.

Ling, C. X. (1994). Predicting irregular past tenses. In *Proceedings of The Sixteenth Annual Conference of the Cognitive Science Society*, pp. 577 – 582. Anonymous ftp: /pub/ling/papers/cogsci94-verb.ps.Z on ftp.csd.uwo.ca.

Michalski, R., Mozetic, I., Hong, J., & Lavrac, N. (1986). The AQ15 inductive learning system: An overview and experiments. In *Proceedings of the American Association for Artificial Intelligence Conference (AAAI-86)*.

Neves, D., & Anderson, J. (1981). Knowledge compilation: Mechanisms for the automatisation of cognitive skills. In Anderson, J. (Ed.), *Cognitive Skills and Their Acquisition*. Erlbaum, Hillsdale, NJ.

Newell, A., & Rosenbloom, P. (1981). Mechanisms of skill acquisition and the law of practice. In Anderson, J. (Ed.), *Cognitive Skills and Their Acquisition*. Erlbaum, Hillsdale, NJ.

Niesser, U., Novick, R., & Lazar, R. (1963). Searching for ten targets simultaneously. *Perceptual and Motor Skills*, *17*, 427–432.

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

Quinlan, J. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann: San Mateo, CA.

Reber, A. (1989). Implicit learning and tacit knowledge. *Journal of Experimental Psychology: General*, *118*, 219–235.

Schlimmer, J., & Fisher, D. (1986). A case study of incremental concept induction. In *Proceedings of the Fifth National Conference on Artificial Intelligence*, pp. 496 – 501. Morgan Kaufmann.

Schmidt, W. C., & Ling, C. X. (1994). A C4.5 model of balance scale development. Submitted to *Machine Learning*. Anonymous ftp: /pub/ling/papers/tr-bs*.ps.Z (* = 1, 2, 3) on ftp.csd.uwo.ca.

Siebel, R. (1963). Discrimination reaction time for a 1,023 alternative task. *Journal of Experimental Psychology*, *66*, 215–226.

Smolensky, P. (1987). The constituent structure of connectionist mental states: A reply to fodor and pylyshyn. *The Southern Journal of Philosophy*, *26*, 137–161.

Smolensky, P. (1988). On the proper treatment of connectionism. *Behavioral and Brain Sciences*, *11*, 1 – 74.

Smolensky, P. (1991). Tensor product variable binding and the representation of symbolic structures in connectionist systems. In Hinton, G. (Ed.), *Connectionist Symbolic Processing*, pp. 159 – 216. Cambridge, MA: MIT Press.

Snoddy, G. (1926). Learning and stability. *Journal of Applied Psychology*, *10*, 1–36.

Stadler, M. (1989). On learning complex procedural knowledge. *Journal of Experimental Psychology: General*, *15*, 1061–1069.

Utgoff, P. E. (1989). Incremental induction of decision trees. *Machine Learning*, *4*, 161 – 186.

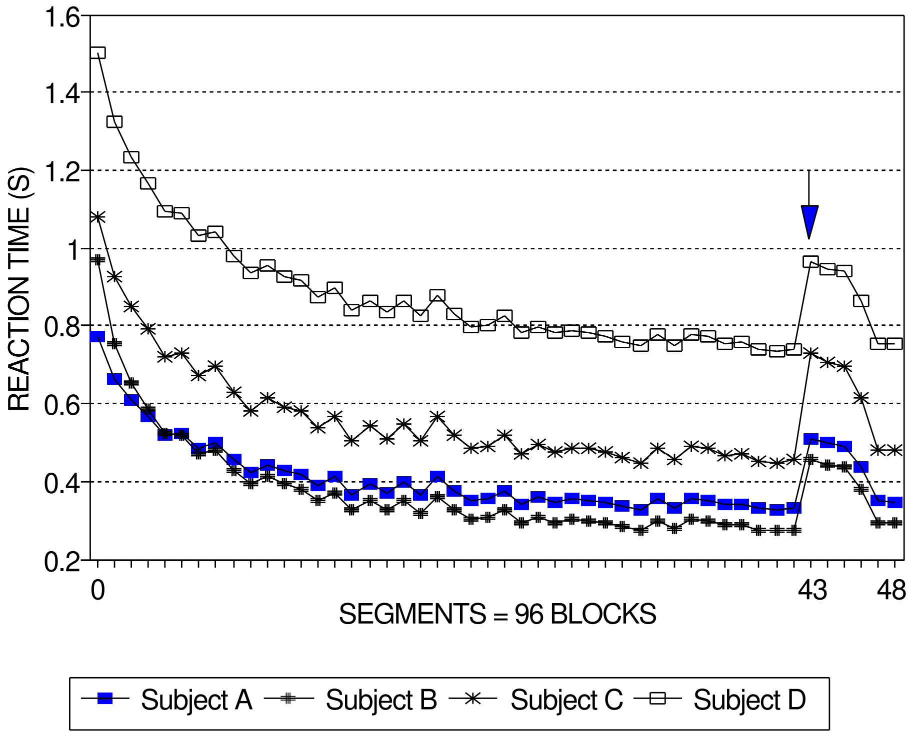


Figure 7: Simulation of four possible subjects.

(A): $t_1 = 150; t_2 = 350; B = 500; \alpha = 0.3.$

(B): $t_1 = 150; t_2 = 300; B = 700; \alpha = 0.5.$

(C): $t_1 = 200; t_2 = 500; B = 700; \alpha = 0.3.$

(D): $t_1 = 150; t_2 = 400; B = 1200; \alpha = 0.2.$

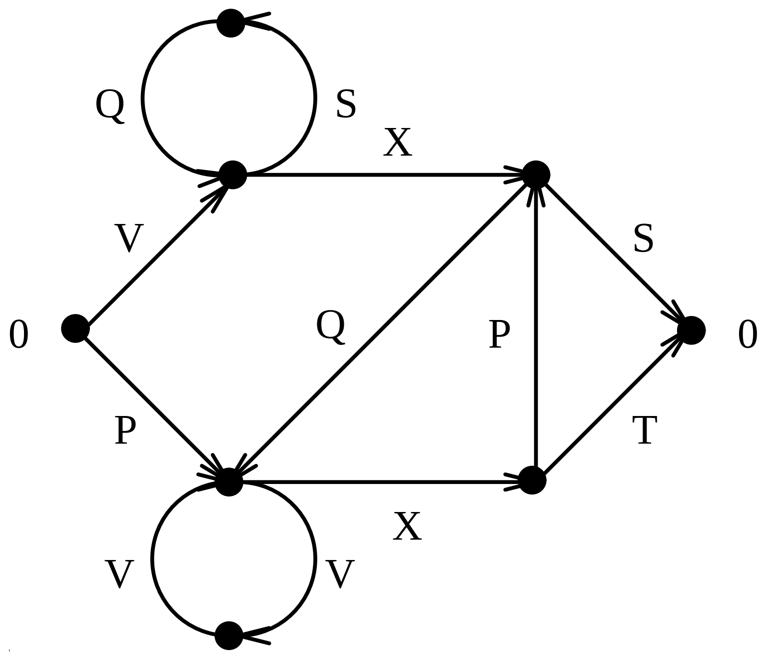


Figure 8: The finite state grammar used to generate the stimulus sequence.

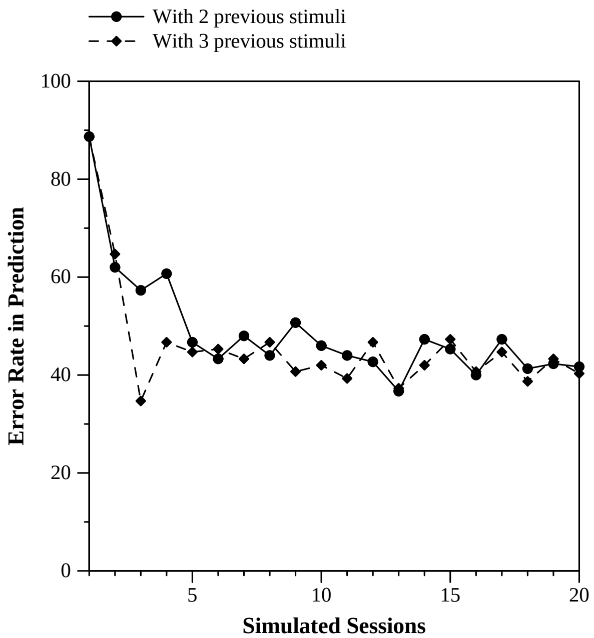


Figure 9: Error rate of prediction from training examples with 2 and 3 previous stimuli.

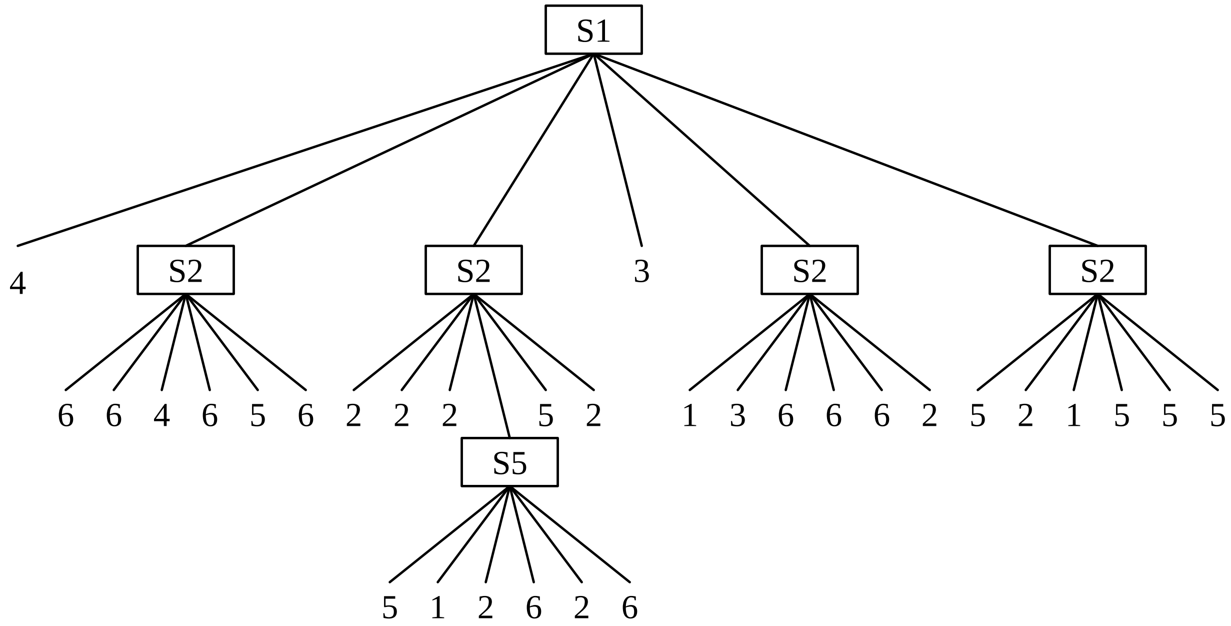


Figure 10: The optimal pruned decision tree produced with window size 10 uses only 5 previous stimuli. Note that the branches of all decision nodes (S1, S2 and S5) are labeled 1 to 6 from left to right.