Title
An Inquiry Into the Function of Implicit Knowledge and its Role in Problem Solving

Journal
Proceedings of the Annual Meeting of the Cognitive Science Society, 22(22)

Authors
Nokes, Timothy J.
Ohlsson, Stellan

Publication Date
2000

Peer reviewed

An Inquiry Into the Function of Implicit Knowledge and its Role in Problem Solving

Timothy J. Nokes (tnokes@uic.edu)
Stellan Ohlsson (stellan@uic.edu)

Department of Psychology
The University of Illinois at Chicago
1007 West Harrison Street (M/C 285)
Chicago, IL 60607, U.S.A.

Abstract

Research on implicit learning has shown that the knowledge generated from memorizing patterned symbol sequences can be used to make familiarity judgements of novel sequences with similar structure. However, the degree to which these knowledge representations can be used for subsequent cognitive processing is not known. In this study, participants memorized either patterned number strings (patterned training) or random number strings (random training) and then solved either a number or letter sequence extrapolation problem. Patterned training participants performed significantly better on number problems than on letter problems, thus implying that patterned training influences performance, but only on near transfer problems.

Function of Implicit Knowledge

To support successful performance on complex, unfamiliar tasks, knowledge must be both abstract and generative. The origin of such knowledge is a central question for cognitive psychologists, developmental psychologists, educators, machine learning researchers and philosophers of science.

Many theoretical proposals conceptualize the acquisition of deep knowledge as a deliberate, effortful and constructive process. For example, one frequently stated hypothesis with roots in both philosophy (Popper, 1972/1959) and psychology (Thorndike, 1898) claims that learners replace or revise their knowledge when the latter is *falsified* by contradictory information; on this view, deep learning is driven by the evaluation of evidence (Gopnick & Meltzoff, 1997; Posner, Strike, Hewson & Gertzog, 1982). The hypothesis of *analogical learning* (e.g., Holyoak & Thagard, 1995) claims that the learner retrieves a possible analog to his or her current problem from memory and discovers their shared structure by constructing a mapping between them. According to the idea of *representational redescription* (Karmiloff-Smith, 1992), the learner reflects on his or her knowledge and, as a consequence, generates a higher-order representation of it. Many other proposed learning mechanisms share this active character (Ram & Leake, 1995).

In contrast, research on implicit learning of artificial grammars (Reber, 1989, 1993) suggests that learning is a passive, inductive process which is independent of any intention to learn and which creates knowledge that cannot be deliberately recalled. In the *training phase* of the standard artificial grammar learning paradigm, the participants memorize letter strings, one by one. The strings have been generated with an artificial grammar and hence embody some very abstract properties, but the participants are not informed of this fact. In the *test phase*, the participants encounter new letter strings which are derivable from the relevant grammar, mixed with distractors which are not. The task is to decide whether the test strings are of the same type as the strings seen during the training phase. A large body of evidence (Stadler & Frensch, 1998) shows that people perform better than chance in the test phase, indicating that they have acquired knowledge of the underlying grammar.

Servan-Schreiber and Anderson (1990) and Perruchet and Gallego (1997) have attempted to explain artificial grammar learning in terms of the learning of substrings. However, Manza and Reber (1997) report a series of six experiments in which the strings encountered in the test phase were expressed in different letters than the strings encountered in the training phase. People perform better than chance in this condition as well, indicating that what is learned is abstract enough to transfer and hence does not consist of knowledge about the relative frequencies of individual substrings. In short, the data imply that what is learned in the artificial grammar learning paradigm is an abstract representation of the relevant grammar.

This finding is counterintuitive, because string memorization is passive, incidental and purely inductive and so stands in contrast to the constructive learning mechanisms hypothesized in other areas of cognitive research. This leads us to inquire into the nature of the knowledge generated by the string memorization procedure. How does that knowledge function in subsequent processing? Can it support problem solving, text comprehension and other higher-order cognitive processes?

To investigate this question, we revised the standard artificial grammar learning paradigm by replacing the string classification task typically used in the test phase with a letter sequence extrapolation problem. Sequence extrapolation problems were first introduced into psychology by Louis L.

Thurstone and they have been studied from a cognitive perspective by Simon (1972), Greeno and Simon (1974), and Kotovsky and Simon (1973). In this type of problem, the problem solver is given a sequence of letters generated in accordance with some pattern and asked to extrapolate it. To solve the problem, he or she must thus first uncover the pattern in the given segment of the letter sequence and then use that pattern to generate the next N letters in the sequence.

The goal of the present study was to determine whether implicit learning of the pattern embedded in a sequence improves the ability to extrapolate that sequence. In the training phase, our participants memorized strings of double-digit numbers generated in accordance with a pattern. In the test phase, they tried to extrapolate a letter or number sequence that followed that same pattern. If string memorization produces an abstract and generative representation of the pattern underlying the strings and if people can access that representation during problem solving, string memorization should improve performance on sequence extrapolation.

To investigate the levels of abstraction we instantiated the extrapolation tasks in both numbers (near transfer) and letters (far transfer). If the knowledge generated from string memorization is encoded in terms of the surface features of the training strings, then that knowledge should not be available for problem solving. In contrast, if the knowledge gained during training is of limited abstraction, then it should be available to solve number problems (near transfer) but not letter problems (far transfer). Finally, if the knowledge gained is completely abstract it should be available to solve both number and letter problems.

Method

Participants Ninety-eight undergraduate students from the University of Illinois at Chicago participated in return for course credit.

Materials The target tasks were two sequence extrapolation problems with a periodicity of six items. The target tasks were instantiated in both numbers (near transfer) and letters (far transfer); see Table 1. To enable the participants to induce the pattern, the given segments were 12 items long. That is, they covered two complete iterations of the underlying pattern. Problems were created specifically for this study with patterns similar to those used by Simon (1972) and Kotovsky and Simon (1973).

For example, pattern 1 in Table 1 can be described as follows: The pattern consists of two groups of two letters, separated by X and ending with Z. Within the first group of two, the second letter is two steps forward in the alphabet from the first. In the second group of two, the first letter is one step forwards from the last letter in the first group, and the second letter is one step backwards from that same letter. The second period has the same internal structure but begins with the letter that is one step forward from the second letter in the first group of two in the previous period.

Table 1. Two sequence extrapolation problems expressed in both letters and numbers.

Symbol Type	Given letter or number sequence & the correct 8-step extrapolation
Problem 1	
Letter	B D X E C Z E G X H F Z H J X K I Z K M
Number	25 27 47 28 26 49 28 30 47 31 29 49 31 33 47 34 32 49 34 36
Problem 2	
Letter	C D B E A M D E C F B N E F D G C O F G
Number	63 64 62 65 61 73 64 65 63 66 62 74 65 66 64 67 63 75 66 67

There were 24 training strings consisting of 12 double-digit numbers, twelve for each problem. The twelve strings associated with a problem followed the same pattern as the given letter or number sequence; see Table 2 for examples. In addition, there were 24 strings of random double-digit numbers used in the control condition. Participants in both number and letter problem solving conditions received the same training.

Table 2. Two training strings for Problem 1.

Example	String
1	13 15 35 16 14 37 16 18 35 19 17 37
2	59 61 81 62 60 83 62 64 81 65 63 83

Each participant received a booklet with two parts. Within each part, there were twelve sheets presenting the strings to be memorized, twelve blank recall sheets, one sheet for assessing the result of the training, one sheet presenting the sequence extrapolation problem, and one blank sheet to assess the participants knowledge of the pattern. Problems were counterbalanced across all conditions.

Design and procedure The participants were randomly assigned to one of four groups created by pairing training (patterned vs. random) with problem-type (letter vs. number): patterned near (n = 26), patterned far (n = 27), random near (n = 21), and random far (n = 24). In the patterned training groups, the participants memorized the strings that conformed to the same patterns as those in the extrapolation problems; see Table 2 for examples. In the random groups, the participants memorized random number sequences. In the near transfer groups, the target problems were number extrapolation problems; see Table 1. In the far transfer groups, the target problems were letter extrapolation problems; see Table 1.

The participants were tested in groups of 25. The procedure consisted of two *cycles.* Each cycle was composed of training followed by problem solving. The participants memorized and recalled twelve strings, one by one. They were given 60 seconds to memorize each string. They were then told to turn the page and write down the string. This procedure was repeated through the twelve training strings. Next, the participants were told to turn the page and solve the sequence extrapolation problem. They were given 5 minutes to solve the problem. They were then asked to turn the page and describe the pattern in the extrapolation sequence as best they could. The second cycle proceeded in the same way. The procedure took approximately 70 minutes.

Results

Training The first question is whether the participants in the patterned training group extracted the pattern embedded in the patterned training strings. If they did, they should perform better on the memorization task than the participants in the random training group. Knowledge of the pattern can be used to reconstruct the number sequence so it should improve recall performance.

The *memory score* for each participant was the number of double-digit numbers correctly recalled in the memorization task. Because there were 12 numbers to memorize, the memory score varied between 0 and 12. Mean memory scores for both patterned and random groups for each pattern are presented in Figure 1.

A 2 (training, patterned vs. random) by 2 (pattern-type, 1vs. 2) mixed analysis of variance revealed that there were main effects for both training and pattern-type. The patterned training group performed significantly better than the random training group, $\underline{F}$ (1, 96) = 88.28, $\underline{MSE}$ = 8.68, $\underline{p} < .05$, indicating that the former benefited from the patterns embedded in the training sequences. As Figure 1 shows, this effect is present for each training pattern. There was also a main effect of pattern-type, $\underline{F}$ (1, 96) = 4.25, $\underline{MSE}$ = 1.48, $\underline{p} <$ .05, indicating that pattern 2 was easier to detect than pattern 1. Finally, type of training interacted significantly with pattern-type, $\underline{F}$ (1, 96) = 5.38, $\underline{MSE}$ = 1.48, $\underline{p} < .05$, indicating that the advantage of the patterned training group was larger for pattern 2 than for pattern 1.

In summary, the data show that the patterned training group performed better on the string memorization task than the random training group. We infer that the participants in the patterned group learned the pattern embedded in the relevant training strings. It is noteworthy that the memorization strings did not share any substrings. Hence, this result contradicts that predicted by the substring hypothesis (e.g., Perruchet & Gallego, 1997).

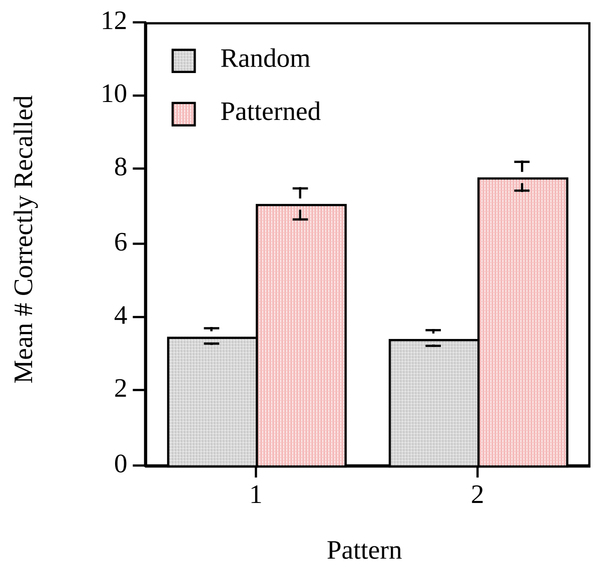

Figure 1. Mean memory scores for both training groups on pattern 1 and 2.

Problem-solving The second question is whether the relevant training group performed better on the problem solving tasks. The *problem solving score* was the number of letters or numbers correctly extrapolated in each problem solving task. Because the participants were asked to continue the sequence to eight places their problem solving scores varied between 0 and 8. Figure 2 shows the mean problem solving scores for both patterned and random groups on each problem.

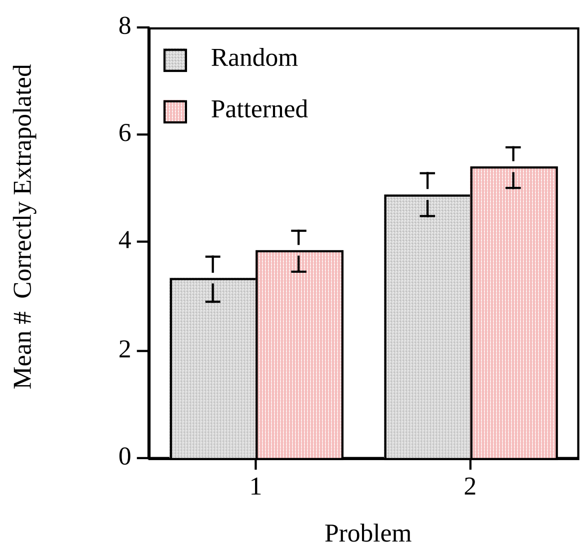

Figure 2. Mean problem solving scores for patterned and random groups on problems 1 and 2.

Although the patterned training group performed better than the random group on each problem, a 2 (treatment, patterned vs. random) by 2 (problem-type, 1 vs. 2) by 2 (transfer, near vs. far) mixed analysis of variance revealed no main effect for treatment condition, $\underline{F}$ (1, 94) = 1.03, $\underline{MSE}$ = 15.33, $\underline{ns}$. However, there was a main effect of transfer, $\underline{F}$ (1, 94) = 10.42, $\underline{MSE}$ = 15.33, $\underline{p}$ < .05, indicating that participants in the near transfer groups performed significantly better than participants in the far transfer groups. There was also a main effect of problem-type, $\underline{F}$ (1, 94) = 16.99, $\underline{MSE}$ = 6.34, $\underline{p}$ < .05, indicating that problem 2 was easier than problem 1. This is consistent with the higher memory performance on pattern 2; see Figure 1.

In addition, the interaction of treatment by transfer was marginally significant, $\underline{F}$ (1, 94) = 3.94, $\underline{MSE}$ = 15.33, $\underline{p}$ = .05, indicating that the advantage for participants in the patterned group was larger on near transfer problems than on far transfer problems. Figure 3 shows the mean problem solving scores for both patterned and random groups as a function of transfer. Main comparisons show that the patterned group performed significantly better than the random group on near transfer problems but not on far transfer problems, $\underline{F}$ (1, 94) = 6.41, $\underline{p}$ < .05, and $\underline{F}$ (1, 94) = .87, $\underline{ns}$ respectively. These results show that participants in the patterned group only benefited from training when solving near transfer problems.

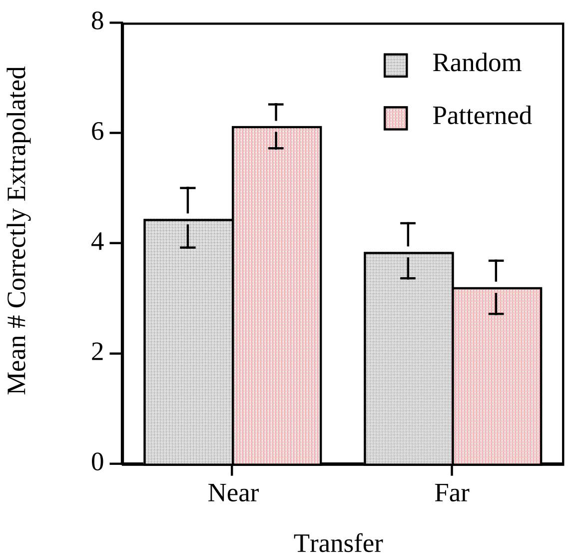

Figure 3. Mean problem-solving score for patterned and random groups on near and far transfer problems.

Individual differences To further investigate the relationship between string memorization and problem solving, we compared mean memory performance for each position in the sequence to the number of participants who correctly solved that position in problem solving. Participants were classified as either high or low memory based on a median spilt of the memory scores for both patterned and random training. Median splits were calculated at each position of the pattern and the number of participants to correctly extrapolate each position was recorded. Table 3 shows the average number of participants to solve any given position correctly for both patterned and random groups as a function of memory.

Table 3. Percentage of subjects to solve any given problem position correctly

Memory	Training Condition	
Performance	Patterned	Random
Low	39%	56%
High	79%*	48%

In addition, chi square tests were calculated at each position of the problem to compare the number of high memory participants to correctly solve a particular position to the number of low memory participants to correctly solve that position. Chi square tests revealed that for patterned training, significantly more high memory participants solved corresponding extrapolations than low memory participants, χ^2 (1, $\underline{N}$ = 53) = 10.43, $\underline{p}$ < .05. Chi square tests also showed that high-low memory groups with random training did not significantly differ in problem solving performance, χ^2 (1, $\underline{N}$ = 45) = .20, $\underline{ns}$.

Similar position by position analyses were conducted comparing participants who solved near transfer problems to those who solved far transfer problems for both training groups. Table 4 shows the average number of participants to solve any given position correctly for both patterned and random groups as a function of transfer.

Table 4. Percentage of subjects to solve any given problem position correctly

	Training Condition	
Problem	Patterned	Random
Near transfer	76%*	55%
Far transfer	41%	48%

Chi square tests revealed that for patterned training, significantly more participants solved near transfer (number) problems than far transfer (letter) problems, χ^2 (1, $\underline{N}$ = 53) = 5.84, $\underline{p}$ < .05. Chi square tests also revealed that participants in the random group did not significantly differ when solving near and far transfer problems, χ^2 (1, $\underline{N}$ = 45) = .64, $\underline{ns}$.

Discussion

As expected, the patterned training group performed significantly better than the random training group on the memorization task. The number strings were equivalent in the two conditions except for the fact that the strings memorized by the patterned group contained a pattern, while the strings memorized by the control group were random. The higher performance of the patterned group is strong evidence that they acquired a schema for the underlying pattern during memorization. This replicates the common result found in implicit learning experiments (Berry, 1997; Reber, 1993; Stadler & Frensch, 1998).

The question asked here is whether the participants could apply this implicitly learned schema in deliberate problem solving. The patterned group was slightly better than the random group on the problem solving tasks, but the difference was small in magnitude (see Figure 2). However, significant differences appear when we take the type of problem and individual differences into account. There was a significant interaction between type of training and type of problem solved. The patterned group performed significantly better than the random group on near transfer problems but not on far transfer problems, indicating the that the knowledge generated from the memorizing the pattern facilitated problem solving, but only when solving problems instantiated in surface features similar to those used in the training sequences.

This conclusion is also supported by the position-by-position analyses. Participants in the patterned training condition who performed above the median on the memorization tasks were consistently more likely to solve any one position during sequence extrapolation than those who performed below the median. This result was true for both problems 1 and 2 (see Table 3). In addition, the number of subjects who correctly solved any one position during extrapolation was consistently larger for participants in the patterned training condition who solved near transfer problems than for those who solved far transfer problems. Again, this result was true for both problems (see Table 4). No effect was observed in the random training conditions.

A plausible explanation for why the participants could not apply what they learned during training to the letter strings is that the relations in the patterns are less obvious on the alphabet than on numbers. For example, to solve pattern 1, the subject needs to realize that the letters E and C are the predecessor and successor, respectively, to D, a fact which is less obvious than the fact that the numbers 26 and 28 have those positions with respect to 27. This explanation implies that memorization of letter strings might produce different results. We are currently conducting studies to explore this implication.

In summary, our results are consistent with the hypothesis that what is acquired by memorizing patterned symbol sequences is a knowledge representation that is potentially generative but of limited abstraction. Such a representation might not be available for recall, conscious inspection or verbalization, but is nevertheless available to other high-level cognitive processes such as problem solving.

Although people do not go through life memorizing symbol strings, they do experience sequences, repetitions, and recurring events. Everyday tasks like starting a car has an intrinsic sequential structure: A person has to insert the key before he or she can turn it; he or she must be inside the car in order to insert the key; he or she must open the door in order to get inside the car; and so on. In symbolic domains, sequential patterns of various kinds are perhaps even more prevalent. An example is the set of computer commands for accomplishing an elementary task such as a writing and sending an email message. Sequential patterns are consequences of the fundamental fact that actions have preconditions.

Given the importance and prevalence of sequential patterns, it is plausible that human beings have evolved cognitive mechanisms for identifying and encoding them. The output of this mechanism are what cognitive scientists often call schemas (Marshall, 1995). The data presented in this paper are consistent with the hypotheses that this mechanism operates even when the learner is not deliberately trying to extract a schema. We find this conclusion compatible with everyday experience: We doubt that human beings walk around and deliberately attempt to find patterns in experience; they find those patterns anyway.

If this conclusion is supported in future studies, the problem for cognitive theory is to elucidate the mechanism by which a schema that is not available for deliberate recall nevertheless influences problem solving, decision making, conceptual change and other cognitive processes. Hybrid models that combine symbolic representations with subsymbolic operations on activation levels (e.g., Anderson & Lebiere, 1998) seem the right kind of model, but the precise specification of such a model has to await replication and elaboration of the empirical observations reported in this paper.

References

Anderson, J. R., & Lebiere, C. (1998). *The atomic components of thought*. Mahwah, New Jersey: Lawrence Erlbaum.

Berry, D., (Ed.), (1997). *How implicit is implicit learning?* Oxford, UK: Oxford University Press.

Gopnik, A., & Meltzoff, A. N. (1997). *Words, thoughts, and theories*. Cambridge, MA: MIT Press.

Greeno, J. G., & Simon H. A. (1974). Processes for sequence production. *Psychological Review*, *81*, 187-198.

Holyoak, K., & Thagard, P. (1995). *Mental leaps*. Cambridge, MA: MIT Press.

Karmiloff-Smith, A. (1992). *Beyond modularity*. Cambridge, MA: MIT Press.

Kotovsky, K., & Simon, H. (1973). Empirical tests of a theory of human acquisition of concepts for sequential patterns. *Cognitive Psychology*, *4*, 399-424.

Manza, L. & Reber, A. S. (1997) Representing artificial grammars: Transfer across stimulus forms and modalities. In D. Berry (Ed.), *How implicit is implicit learning?* (pp. 73-106). Oxford, UK: Oxford University Press.

Marshall, S. (1995). *Schemas in problem solving*. Cambridge, UK: Cambridge University Press.

Perruchet, P. & Gallego, J. (1997) A subjective unit formation account of implicit learning. In Berry, D. C., (Ed.), *How implicit is implicit learning?* (pp. 124-161). Oxford, UK: Oxford University Press.

Popper, K. (1972/1959). *The logic of scientific discovery*. London, UK: Hutchinson. [Orig. *Logik der Forschung*, Vienna, 1935.]

Posner, G., Strike, K. A., Hewson, P. W., & Gertzog, W. A. (1982). Accommodation of a scientific conception: Toward a theory of conceptual change. *Science Education*, *66*, 211-227.

Ram, A., & Leake, D. (Eds.), (1995). *Goal-driven learning*. Cambridge, MA: MIT Press.

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

Reber, A. S. (1993) *Implicit learning and tacit knowledge: An essay on the cognitive unconscious*. New York: Oxford University Press.

Servan-Schreiber, E., & Anderson, J. R. (1990). Learning artificial grammars with competitive chunking. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *16*, 592-608.

Simon, H. (1972). Complexity and the representation of patterned sequences of symbols. *Psychological Review*, *79*, 369-382.

Stadler, M., & Frensch, P, (Eds.), (1998). *Handbook of implicit learning*. Thousand Oaks, CA: SAGE.

Thorndike, E. (1898). *Animal intelligence*. Unpublished doctoral dissertation. New York: Columbia University.