Memory & Cognition
2008, 36 (6), 1122-1131
doi: 10.3758/MC.36.6.1122

Effects of grammar complexity on artificial grammar learning

Esther van den Bos and Fenna H. Poletiek
Leiden University, Leiden, The Netherlands

The present study identified two aspects of complexity that have been manipulated in the implicit learning literature and investigated how they affect implicit and explicit learning of artificial grammars. Ten finite state grammars were used to vary complexity. The results indicated that dependency length is more relevant to the complexity of a structure than is the number of associations that have to be learned. Although implicit learning led to better performance on a grammaticality judgment test than did explicit learning, it was negatively affected by increasing complexity: Performance decreased as there was an increase in the number of previous letters that had to be taken into account to determine whether or not the next letter was a grammatical continuation. In particular, the results suggested that implicit learning of higher order dependencies is hampered by the presence of longer dependencies. Knowledge of first-order dependencies was acquired regardless of complexity and learning mode.

When people are presented with exemplars of a simple structure, they may intentionally try to grasp the regularities and become aware of the knowledge they acquire. In addition to this explicit way of learning, Reber (1976, 1989) proposed that structures can also be learned implicitly, without any intention to learn and without complete awareness of the acquired knowledge. These two ways of structure learning are associated with different levels of complexity. For one thing, a minimum level of complexity of the stimuli is required to observe implicit learning (Reber, 1976). This was demonstrated by an experiment in which participants had to obtain target output values by providing input values to unknown equations. The participants acquired explicit knowledge of simple equations, relating each output value to one input value, whereas they acquired implicit knowledge of complex equations, which related each output value to two input values (Lee, 1995).

In addition, Reber (1976) suggested that implicit learning is especially suited to complex structures, whereas explicit learning works well with simple structures but would be hampered by increasing complexity. In the view of Hayes and Broadbent (1988), explicit learning is restricted to simple regularities because the process relies on working memory. Since working memory capacity is limited, explicit learning would involve active selection of a small amount of relevant information. Regularities involving a large number of variables could be acquired only by implicit learning, which would unselectively store the frequency of co-occurrence of all the elements present.

In line with these suggestions, Reber (1976) demonstrated that a finite state grammar could be learned better implicitly than explicitly. In the induction phase of this artificial grammar learning (AGL) experiment, the implicit learning group was instructed to memorize letter strings without being informed that they were generated by a grammar. The explicit learning group received the additional instruction to look for rules underlying the strings. Subsequently, when all the participants had been informed about the existence of the grammar, they were instructed to judge whether or not new strings were grammatical. The participants who had memorized the letter strings were correct more often than the participants who had looked for rules. In addition, Mathews et al. (1989) showed that although a finite state grammar could be learned by performing a memorize task, a rule search task produced better results with a biconditional grammar. Johnstone and Shanks (2001) replicated this finding and suggested that the different results for the two types of grammar are due to differences in complexity. They argued that biconditional grammars are less complex than finite state grammars, since they consist of a smaller number of rules.

Apart from the general observation that implicit learning is better suited to complex structures, whereas explicit learning is more efficient for simple materials, however, the effects of complexity on structure learning have received little attention. For example, the ranges of complexity on which implicit and explicit learning can operate have not been specified. Therefore, it is currently impossible to predict whether or not looking for rules would be an advisable strategy for a given structure. In a review of his work, Reber (1993) mentioned that variations in the complexity of finite state grammars above the level required to observe implicit learning have little effect on performance. To the best of our knowledge, however, no

E. van den Bos, ejv28@cornell.edu

systematic manipulation of the complexity of finite state grammars has been reported in detail.

A few studies have manipulated the complexity of the structure the participants were exposed to in the serial reaction time (SRT) paradigm (Nissen & Bullemer, 1987). In this task, participants have to press a button corresponding to the location at which a stimulus appears on a screen. Structure learning is evidenced by a decrease in reaction times for stimuli that follow a structured sequence. However, the results of studies manipulating the complexity of this structure have been inconsistent. Some have shown that structure learning is hampered by increasing complexity when participants look for rules, whereas it is unaffected when participants just respond to the stimulus (Fletcher et al., 2005; Reed & Johnson, 1998). Other studies have shown a negative effect of increasing complexity without instructing participants to look for rules (Soetens, Melis, & Notebaert, 2004; Stadler, 1992).

These contrasting findings are probably due to differences in the definitions and manipulations of complexity that have been used. They highlight the need for an analysis of the characteristics that contribute to a structure's complexity. The aim of the present study was to provide such an analysis by comparing the measures of complexity that have been used in the implicit learning literature. Subsequently, two of the measures were used to investigate implicit and explicit AGL over a wide range of complexity.

Measures of Complexity

As was noted above, previous studies on implicit learning have used various ways to measure the complexity of the structure that had to be learned. A relatively simple measure of the complexity of artificial grammars, proposed by Johnstone and Shanks (2001), is the number of rules a grammar consists of. This number is equal to the number of letters that can be added to a string at each transition between internal nodes of the grammar plus the number of letters that can terminate a string at each end node (Johnstone & Shanks, 1999). If the end node of the grammar is considered to be equivalent with the initial node, this comes down to the number of links in the grammar. The number of rules has also been used by Reed (in Reed & Johnson, 1998) to vary the complexity of the regularities that had to be learned in an SRT task.

At another level of description, Perruchet and Vinter (1998) noted that learning may be influenced by the number of associations that have to be acquired. For example, a structure that can be described using a few bigrams (i.e., two-letter chunks) repeatedly is simpler than a structure that can only be described by many unique bigrams. The number of bigrams required to describe a structure can therefore be viewed as a second measure of complexity.

A third measure can be found in the SRT paradigm, where the complexity of a structure is often related to its predictability (e.g., Cohen, Ivry, & Keele, 1990; Reed & Johnson, 1994; Soetens et al., 2004; Stadler, 1992). In the simplest case, a stimulus at a certain location on the screen is always followed by a stimulus at one other location. Knowing the location of the present stimulus is then sufficient to predict where the next will appear. The structure becomes more complex when the next stimulus can appear at several locations, depending on the location of the stimulus before the present one. In this way, the number of previous elements required to predict the next can be used to measure the complexity of a structure.

Finally, Bollt and Jones (2000) proposed topological entropy (TE) as a specific measure of the complexity of finite state grammars. A detailed explanation of the computation of TE is provided in Appendix A. In short, the complexity of an artificial grammar is defined as the growth rate of the number of unique strings of a given length that the grammar generates as string length goes to infinity (Bollt & Jones, 2000). The authors explain that, for most grammars, the number of different strings that can be generated grows exponentially as strings of greater length are considered. TE is a measure of the exponent with which the number of possible strings increases. Put simply, the complexity of a grammar increases according to this measure as it generates a greater number of different strings of any given length.

Although the measures discussed above are sensitive to different aspects of complexity, they are highly intercorrelated, because each measure is sensitive to different effects of adding links to a grammar. Adding a link by definition increases the number of rules, and it can also allow for the generation of additional unique strings, increasing TE. At the same time, the new link may create a new bigram, increasing the number of bigrams required to describe the structure. Alternatively, the new link may cause an existing bigram to occur at another location in the grammar. This may increase the number of previous elements that have to be taken into account to predict which letters can succeed the bigram. (See Appendix A for an additional explanation of the correlation between TE and number of previous elements.) Table 1 illustrates the intercorrelatedness of the measures. The correlations are based on the scores of the 10 artificial grammars used in this study on each of the four measures of complexity (see Appendix B).

The partial correlations in Table 1, indicating the correlation between two measures when the contributions of the other two measures have been removed, reveal that the four measures of complexity, in fact, measure two different aspects. On the one hand, the number of rules and the number of bigrams seem to measure the number of associations that have to be learned. The number of elements required to predict the next and TE, on the other hand, seem to measure the length of the dependencies that have to be learned. The number of bigrams and TE are more fine-grained than their counterparts. Since they are probably the most sensitive, these two measures will be used in the present study to investigate whether or not these aspects of complexity affect AGL.

A Closer Look at Dependency Length

Several studies have suggested that dependencies are more difficult to learn as their length increases. For example, when the location of the stimulus in an SRT task could be predicted on the basis of one previous location

Table 1
Correlations and Partial Correlations Between Four Measures of Complexity

	NR		NB		NE		TE	
	r	partial *r*	*r*	partial *r*	*r*	partial *r*	*r*	partial *r*
NR	1.000	1.000	.994**	.926**	.520	−.471	.718*	.418
NB			1.000	1.000	.570	.126	.757*	−.058
NE					1.000	1.000	.965**	.993**
TE							1.000	1.000

Note—NR, number of rules; NB, number of bigrams; NE, number of elements required to predict the next; TE, topological entropy. $^{*}p < .05$. $^{**}p < .01$.

(first-order dependency), participants' reaction times decreased twice as much as when at least two previous locations (second-order dependency) had to be taken into account (because all first-order conditional probabilities were equal; Soetens et al., 2004). When the participants were exposed to a sequence in which one previous location provided some information but prediction could be improved by taking more previous locations into account, sensitivity to first-order dependencies preceded sensitivity to second- and third-order dependencies (Cleeremans & McClelland, 1991). Similarly, an AGL study showed that participants acquired knowledge of higher order dependencies when they were presented with a large number of exemplars, whereas they based their grammaticality judgments on knowledge of first- and second-order dependencies after presentation with a small number (Meulemans & Van der Linden, 1997).

Since second- and higher order dependencies are more difficult to learn than first-order dependencies, it seems probable that their acquisition would suffer more from increasing complexity. If TE turns out to be an influential aspect of complexity, such a finding would provide an indication of how complexity affects AGL. Namely, it would suggest that dependencies of a given length (e.g., second order) become more difficult to learn as the length of the dependencies present in the stimuli increases (e.g., from second order to third and fourth order). Therefore, this study will also investigate the effects of increasing complexity on the acquisition of first- and second-order dependencies.

This question is largely independent of the controversial issue of how participants represent their knowledge of an artificial grammar. In a recent review, Pothos (2007) identified rule learning, fragment learning, and whole-item learning as the main theoretical accounts. A comprehensive analysis of the characteristics of letter strings that affect participants' grammaticality judgments provided evidence for both fragment and whole-item learning (Lotz & Kinder, 2006). Participants were sensitive to the frequency of occurrence of fragments in the induction phase (mainly those beginning and ending the strings; anchor chunk strength), to similarity to individual training strings (edit distance), and to previously seen patterns of letter repetitions within the string (global and local repetition structure). Although such results may be influenced by the complexity of the grammar, we will not address that question here.[1]

In summary, the present study had two aims. First, we investigated how performance on an AGL task under *memorize* and *look for rules* instructions is related to two aspects of complexity of the grammar: the number of associations and the length of the dependencies that have to be learned. Second, we tested the prediction that, if AGL is negatively affected by increasing complexity, the effect will be stronger for second-order than for first-order dependencies.

METHOD

Participants

Sixty-one undergraduate students at Leiden University (20 of them male, 41 female; 18–41 years of age) participated in the experiment. They received either course credit or €4.50 for their participation. The data from 1 participant had to be discarded because, as a nonnative speaker of Dutch, he turned out to be unable to understand the instructions.

Design

The experiment consisted of an induction phase and a test phase. In both phases, letter strings were presented in random order. There were three independent variables. First, the instruction for the induction phase was varied between participants. One half of the participants were instructed to memorize the letter strings. The other half were informed that the strings had been generated according to certain rules and received the instruction to search for these rules. A second between-participants variable was the complexity of the grammar, as measured by its number of bigrams and its TE. Ten artificial grammars were used to vary complexity. This large number was chosen because the research questions were directed at a continuous relationship between grammar complexity and performance, rather than at a difference in performance on two particular grammars. Each of the 10 artificial grammars was studied by 3 participants under each instruction, providing 30 data points per group for the regression analyses. Finally, type of test string was varied as a within-participants factor to examine the length of the dependencies that the participants acquired. The dependent variable was the mean proportion of test strings correctly classified as grammatical or ungrammatical.

Materials

The stimuli in this experiment were letter strings generated by 10 finite state grammars. All the grammars used the same letters—J, M, N, P, Q, R, S, T, W, X, and Z—and consisted of the same 11 states. However, these states were connected in different ways and by a varying number of links to produce different scores on the complexity measures. The number of bigrams varied from 30 to 47, and TE ranged from 0.55 to 2.58. This is a relatively wide range that may exceed the one Reber (1993) referred to. For comparison, the number of bigrams and the TE of some of Reber's grammars were (respectively) 14 and 0.48 (Reber, 1967, 1976), 16 and 0.60 (Reber, Kassin, Lewis, & Cantor, 1980), and 21 and 1.52 (Reber & Allen, 1978). Graphs and scores on the complexity measures for each of the grammars used in the present study are provided in Appendix B.

A computer program generated a set of 120 unique letter strings for each grammar. Each string was generated by moving along the arrows from the initial state (0) to the end state (0) of the graph, while adding the corresponding letter to the string. The length of the strings varied between 5 and 11 letters. Of each set, 60 strings were assigned to the induction phase and 50 to the test phase, balanced over the paths of the grammar. Five of the remaining strings were used on practice trials in the test phase; the rest were discarded. The practice stimuli for the induction phase consisted of number strings unrelated to the grammars.

The 50 strings assigned to the test phase were subdivided into two subsets. One subset consisted of unaltered grammatical strings, whereas the other consisted of strings that were made ungrammatical by switching two adjacent letters, excluding the first and last. Switching letters could result in a violation of a first-order dependency: a bigram that cannot occur, according to the grammar. Alternatively, it could result in a violation of a second-order dependency: a trigram that cannot occur, according to the grammar. Switching two inner letters of a string always affects three transitions, thus producing one of four possible combinations of violations: three first-order violations, two first- and one second-order violation, one first- and two second-order violations, or three second-order violations. These combinations were introduced into the ungrammatical test sets according to their probability of occurrence in the complete set of 120 strings generated by each grammar. Since the probability of three second-order violations was very low in most grammars, only three types of ungrammatical strings were discerned in this study: strings with three first-order violations, strings with two first- and one second-order violation, and strings with two or more second-order violations. Differences in performance on these types of strings would indicate differences in the extent to which knowledge of first- and second-order dependencies had been acquired.

All the stimuli were displayed on a computer monitor as black text (Arial 18 point, bold) against a white background. The participants were seated in front of the computer monitor at a distance of about 50 cm. They reacted by pressing keys on a keyboard.

Procedure

The participants were tested individually in a dimly lit test booth. At the beginning of the experiment, they were informed that it would consist of two parts and would involve studying letter strings for a subsequent test. Then they received their specific instructions for the induction phase. All the participants were presented with 5 practice trials and 60 experimental trials. Each trial started with a fixation cross appearing in the middle of the screen. After 1 sec, the cross was replaced by a letter string centered at the fixation point. The string was displayed for 5 sec. Then the participants with the *memorize* instruction were prompted to reproduce the string. The participants with the *look for rules* instruction were prompted to enter a part of the string that they thought was important. After pressing the Enter key, the participants were again presented with the original letter string for 2 sec so that they could check their answer. Finally, the screen turned blank for 1 sec before the next trial began.

In the second part of the experiment, all the participants were informed that the letter strings presented in the first part had been generated according to a complex set of rules. They were instructed to judge whether or not new letter strings followed the same rules. The participants were required to press the "j" key if they thought that a string followed the rules and the "n" key if they thought that a string did not follow the rules. In addition, they were required to indicate their confidence in each judgment on a scale from 1 (*very little*) to 5 (*very much*) by pressing one of the number keys on the keyboard.

The participants were presented with 5 practice trials, followed by 50 experimental trials. Each trial began with a fixation cross appearing in the middle of the screen. After 1 sec, the cross was replaced by a letter string centered at the fixation point. When the participant pressed the "j" or "n" key, the screen turned blank for 1 sec. Subsequently, the confidence scale was presented until the participant pressed a number from 1 to 5. A final blank screen separated two consecutive trials by 1 sec. When the participants had completed all the trials, they were thanked for their participation. The experiment took about 30 min.

RESULTS

Over all the trials, both the participants who had memorized letter strings and the participants who had looked for the underlying rules scored significantly above chance on the grammaticality judgment test (see Table 2). This indicates that both groups had acquired knowledge of the grammar. However, an independent samples t test showed that the proportion correct was higher after memorizing than after looking for rules [$t(58) = 3.126, p = .003$].

To examine whether the instruction to memorize letter strings had induced implicit learning, the *guessing criterion* (Dienes, Altmann, Kwan, & Goode, 1995) was used. According to this criterion, people possess implicit knowledge if they perform significantly above chance when they claim to be guessing. The criterion provides a meaningful distinction between implicit and explicit knowledge, since it has been shown to distinguish between knowledge that can be acquired under conditions of divided attention and knowledge that cannot (Dienes et al., 1995).

The proportion of correct grammaticality judgments was computed for trials on which the participants had rated (very) little confidence in their judgment (ratings 1 and 2). Scores that were based on fewer than three trials were excluded from the analysis. One-sample t tests showed that the proportion correct was significantly above chance for the participants who had memorized letter strings [$M = .627$, $SD = .169$; $t(25) = 3.847$, $p = .001$] but failed to reach significance for the participants who had looked for rules [$M = .554$, $SD = .158$; $t(26) = 1.775$, $p = .088$].

Table 2
Mean Proportions Correct, Standard Deviations, and One-Sample *t* Tests for Different Types of Test Strings by Instruction in the Induction Phase

	Memorize			Look for Rules		
	M	SD	t	M	SD	t
All strings	.612	.070	8.8^{**}	.555	.070	4.3^{**}
Grammatical	.672	.102	9.3^{**}	.587	.148	3.2^{**}
Three first-order violations	.673	.161	5.9^{**}	.612	.187	3.3^{**}
Two first-, one second-order violation	.488	.170	−0.4	.507	.146	0.2
Two or more second-order violations	.520	.220	0.4	.411	.257	1.7

Note—One-sample t tests compared the proportion of correct grammaticality judgments with chance (.50). There were 23 degrees of freedom for the strings with two or more second-order violations and 29 degrees of freedom for all other comparisons. $^{**}p < .01$.

Effects of Complexity on Performance in AGL

To determine whether a grammar's complexity affects how well artificial grammars are learned under the instruction to memorize letter strings and under the instruction to look for rules, separate linear regression analyses were performed using the enter method. The dependent variable was the proportion of correct classifications in the test phase (regardless of confidence); the predictors were the two measures of complexity: number of bigrams and TE.

For the group of participants who had been instructed to memorize strings in the induction phase, all three of the possible models reached significance. The model including only the number of bigrams accounted for 14.1% of the variance in the proportion of correct classifications [$F(1,28) = 4.601$, $p = .041$]. The model including only TE accounted for 19.8% of the variance [$F(1,28) = 6.931$, $p = .014$], and the model including both predictors accounted for 20.2% of the variance [$F(2,27) = 3.415$, $p = .048$]. Adding the number of bigrams was no significant improvement to the model including only TE, however [$F_{change}(1,27) = 0.118$, $p = .734$], indicating that TE was, by itself, the best predictor of proportion correct in the test phase. The negative relationship is illustrated by Figure 1.

For the group instructed to look for rules underlying the strings, none of the regression models was significant. Although there was as much variance in the proportion of correct classifications in this condition as in the memorize condition [Levene's test: $F(1,58) = 0.553$, $p = .460$], it could not be explained by the number of bigrams [$F(1,28) = 1.257$, $p = .272$], TE [$F(1,28) = 0.461$, $p = .503$], or a combination of these predictors [$F(2,27) = 0.636$, $p = .537$].

Effects of Complexity on the Acquisition of First- and Second-Order Dependencies

To determine whether complexity affects the length of the dependencies acquired in AGL, separate linear regression analyses were performed for each type of test string (grammatical, three first-order violations, two first- and one second-order violation, and two or more second-order violations). Grammars A and D (see Appendix B) were not included in the latter analysis, since they provided fewer than three test strings containing two or more second-order violations. The dependent variable was the proportion of correct classifications in the test phase (regardless of confidence); number of bigrams and TE were the predictors.

For the memorize group, none of the regression models was significant for the grammatical strings (smallest $p = .407$), the strings with three first-order violations (smallest $p = .395$), or the strings with two first- and one second-order violation (smallest $p = .115$). Overall, performance was above chance for grammatical strings and strings containing only first-order violations, but not for strings containing two first- and one second-order violation (see Table 2). For the strings with two or more second-order violations, however, performance was affected by complexity. The regression model including only TE reached

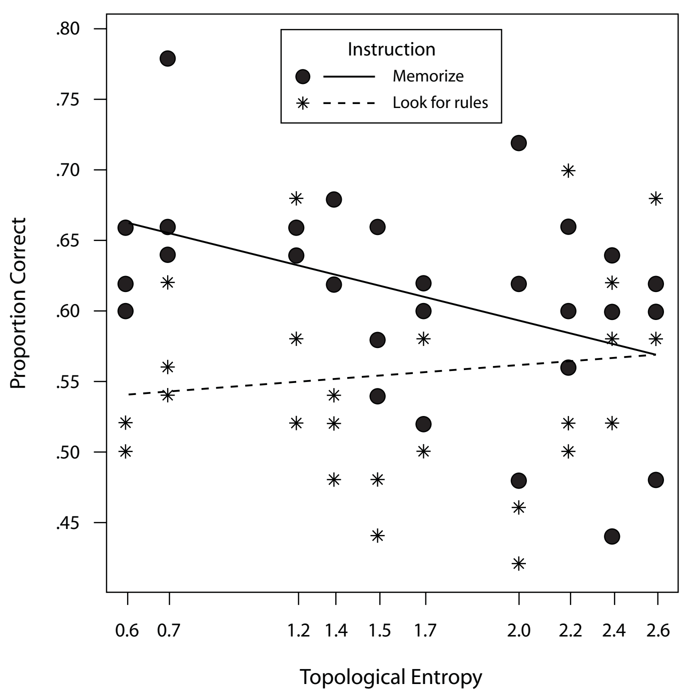


Figure 1. Predicted and observed proportions of correct grammaticality judgments by instruction in the induction phase as a function of topological entropy.

significance [$F(1,22) = 5.885$, $p = .024$], accounting for 21.1% of the variance in the proportion of correct classifications. The relationship was negative, which suggests that the participants in the memorize condition recognized strings containing several illegal trigrams as ungrammatical when they worked with simple grammars but lost this ability as complexity increased.

For the participants who had looked for rules underlying the strings in the induction phase, none of the regression models was significant for the grammatical strings (smallest $p = .797$), the strings with two first- and one second-order violation (smallest $p = .096$), or the strings with two or more second-order violations (smallest $p = .732$). Overall, performance was above chance for grammatical strings, but not for strings with two first- and one second-order violation or strings with two or more second-order violations (see Table 2). For strings containing three first-order violations, however, all the regression models were significant. The model including only the number of bigrams accounted for 22.6% of the variance [$F(1,28) = 8.196$, $p = .008$]; the model including only TE accounted for 25.6% of the variance [$F(1,28) = 9.620$, $p = .004$]; and the model including both predictors accounted for 27.6% of the variance [$F(2,27) = 5.148$, $p = .013$]. However, adding the number of bigrams was no significant improvement to the model including only TE [$F_{\text{change}}(1,27) = 0.759$, $p = .391$]. Interestingly, the relationship between TE and proportion correct was positive, indicating that more strings with three first-order violations were recognized as ungrammatical by the participants who had looked for rules as complexity increased.

DISCUSSION

The first aim of this study was to examine how implicit and explicit AGL were influenced by two aspects of the complexity of the grammars. When the participants had looked for rules underlying the letter strings in the induction phase, their performance on a grammaticality judgment test was not affected by the complexity of the grammar, irrespective of whether it was measured by its number of bigrams or by its TE. On the face of it, this is at odds with the results of previous studies, which have indicated that explicit learning is more efficient than implicit learning for simple materials. However, the simple structures used in those studies were less complex than the simplest grammar in the present study (dependencies involving one element [Lee, 1995] vs. two; 4 rules [Johnstone & Shanks, 2001; Mathews et al., 1989] vs. 20). This suggests that it would have been possible to observe an effect of complexity on explicit learning if the present study had included even simpler grammars.

For the range examined here, performance after looking for rules was significantly above chance, but lower than after memorizing. The latter finding is in line with Reber's (1976) claim that complex materials can be learned better implicitly than explicitly. However, the results also showed that performance after memorizing decreased with increasing complexity of the grammar, as measured by both its number of bigrams and its TE. On the most complex grammar, memorizing did not lead to more correct grammaticality judgments than did looking for rules.

Which Aspect of Complexity Affects Implicit Learning?

Importantly, the present study also identifies an aspect of complexity that makes implicit learning of artificial grammars increasingly difficult. Namely, the regression analyses indicated that TE was the best predictor of performance. Since TE is sensitive to the length of the dependencies that have to be learned, this result suggests that implicit learning of artificial grammars is hampered most when letter chunks of increasing size have to be taken into account to determine whether or not the sequence is grammatical. This situation arises when a link is added to the grammar that generates a letter chunk that could already be generated by an existing link. The number of associations that has to be acquired seems to be a less influential aspect of the complexity of artificial grammars.

Interestingly, these findings also clarify the contrasting results that have been obtained with the SRT paradigm. When complexity was manipulated by increasing the number of previous locations that had to be taken into account to predict the next, sequence learning was shown to be hampered by increasing complexity (Soetens et al., 2004; Stadler, 1992). When this effect was not observed (unless participants were instructed to look for rules), complexity had been manipulated in other ways. In the study by Fletcher et al. (2005), complexity was increased by interspersing a predictable sequence with random locations. In the study by Reed (in Reed & Johnson, 1998), the location of the stimulus depended on the background colors of the locations, and complexity was manipulated by varying the number of rules governing this relationship. This provides converging evidence that the length of the dependencies that have to be acquired is responsible for the negative effect of complexity on implicit learning.

The Length of the Acquired Dependencies

A second question addressed in this study is whether complexity influences the length of the dependencies acquired in AGL. After looking for rules, the participants were able to recognize the grammaticality of grammatical test strings and the ungrammaticality of test strings in which only first-order dependencies were violated. Surprisingly, performance on the latter type of test string was enhanced by increasing complexity. This may have been due to the participants' strategies in explicit learning. On the one hand, participants may restrict themselves to looking for simple first-order dependencies when they are required to deal with high complexity. Grammars of lesser complexity, on the other hand, may invite participants to try to discover higher order dependencies. So, although looking for the rules of an artificial grammar may produce a constant level of performance, this may reflect sensitivity to dependencies of different lengths for grammars of varying complexity.

After memorizing letter strings in the induction phase, test strings that contained only first-order violations were recognized as ungrammatical, irrespective of the complexity of the grammar. However, performance on the grammaticality judgment test was negatively affected by increasing complexity for test strings containing several violations of second-order dependencies. Since most of those test strings also contained a violation of a first-order dependency (i.e., an illegal bigram), there are two possible interpretations for this finding. One is that it becomes more difficult to recognize a single illegal bigram as more bigrams are permitted by the grammar. According to this interpretation, participants would not have learned second-order dependencies for any of the grammars. However, this interpretation implies that performance would be related to the number of bigrams, whereas the results showed that only TE could predict performance on these test strings.

The second interpretation is the possibility discussed in the introduction, that it becomes more difficult to learn second-order dependencies as increasingly longer dependencies are present in the letter strings. This interpretation fits in with a proposal that sequence learning can be accomplished by a multidimensional and a unidimensional system (Keele, Ivry, Mayr, Hazeltine, & Heuer, 2003). Higher order dependencies are learned by the multidimensional system, which can form associations between elements of various dimensions (e.g., shape, position, duration, pitch) as long as they are attended. When unpredictable elements are attended, however, learning in the multidimensional system is disrupted. It should be noted that bigrams and trigrams occur in an increasingly unpredictable context of surrounding letters as the length of the dependencies present in an artificial grammar increases. Therefore, the finding that second-order dependencies are hampered by increasing complexity may be explained by a disruption of the multidimensional system. First-order dependencies, in contrast, are acquired by the unidimensional system, which operates independently of attention (Keele et al., 2003).

Learning Complex Structures

The account of Keele et al. (2003) also suggests that the negative effect of complexity on the acquisition of second-order dependencies could be overcome by providing concurrent information in a different dimension. For example, one could add grouping cues, to divide the letter strings into smaller fragments, or supplement the strings with a reference field. Both initial constraints on fragment length (Elman, 1993; Kersten & Earles, 2001; Newport, 1990) and semantic constraints (Nagata, 1977; Rohde & Plaut, 1999) have been shown to facilitate the acquisition of complex structures, such as natural grammars. It would be interesting to examine whether these factors could also eliminate the negative effect of complexity for the artificial grammars used in the present study.

With regard to explicit learning, performance may be enhanced by providing participants with more specific information concerning the rules that they have to look for (van den Bos & Poletiek, 2008). Moreover, for both learning modes, more extended exposure to grammatical strings in the induction phase (Meulemans & Van der Linden, 1997) may enable participants to learn second-order dependencies and to maintain a relatively high performance on a grammaticality judgment task despite increasing complexity of the grammar. We do not mean to suggest that learning of complex structures is impossible, but further research is needed to identify factors that may enhance it.

The present study indicates that with limited exposure to grammatical strings under standard conditions, implicit learning is negatively affected by increasing complexity of the grammars. The results suggest that the aspect of complexity that is most relevant to structure learning is the length of the dependencies that have to be acquired, rather than the number of associations. In particular, the presence of longer dependencies seems to interfere with the acquisition of second-order dependencies. After the memorizing of letter strings, performance on a grammaticality judgment test suffered as more and more previous letters had to be taken into account to determine the grammaticality of the next. Nevertheless, memorizing letter strings was more efficient than looking for the underlying rules in the tested range.

AUTHOR NOTE

E.v.d.B is now at Cornell University. While revising the manuscript, she was supported by a grant from the Niels Stensen Stichting. We thank Inti Pelupessy for discussing the measures of complexity and Gezinus Wolters, Pierre Perruchet, and several anonymous reviewers for helpful comments on a previous version of this article. Correspondence concerning this article should be addressed to E. van den Bos or F. Poletiek, Cognitive Psychology, Leiden University, P.O. Box 9555, 2300/RB Leiden, the Netherlands (e-mail: ejv28@cornell.edu or poletiek@fsw.leidenuniv.nl).

REFERENCES

Bollt, E. M., & Jones, M. A. (2000). The complexity of artificial grammars. *Nonlinear Dynamics, Psychology, & Life Sciences*, **4**, 153-168.

Chomsky, N., & Miller, G. A. (1958). Finite state languages. *Information & Control*, **1**, 91-112.

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

Cohen, A., Ivry, R. I., & Keele, S. W. (1990). Attention and structure in sequence learning. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, **16**, 17-30.

Dienes, Z., Altmann, G. T. M., Kwan, L., & Goode, A. (1995). Unconscious knowledge of artificial grammars is applied strategically. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, **21**, 1322-1338.

Elman, J. L. (1993). Learning and development in neural networks: The importance of starting small. *Cognition*, **48**, 71-99.

Fletcher, P. C., Zafiris, O., Frith, C. D., Honey, R. A. E., Corlett, P. R., Zilles, K., & Fink, G. R. (2005). On the benefits of not trying: Brain activity and connectivity reflecting the interactions of explicit and implicit sequence learning. *Cerebral Cortex*, **15**, 1002-1015.

Hayes, N. A., & Broadbent, D. E. (1988). Two modes of learning for interactive tasks. *Cognition*, **28**, 249-276.

Johnstone, T., & Shanks, D. R. (1999). Two mechanisms in implicit artificial grammar learning? Comment on Meulemans and Van der Linden (1997). *Journal of Experimental Psychology: Learning, Memory, & Cognition*, **25**, 524-531.

Johnstone, T., & Shanks, D. R. (2001). Abstractionist and processing accounts of implicit learning. *Cognitive Psychology*, **42**, 61-112.

Keele, S. W., Ivry, R., Mayr, U., Hazeltine, E., & Heuer, H. (2003). The cognitive and neural architecture of sequence representation. *Psychological Review*, **110**, 316-339.

Kersten, A. W., & Earles, J. L. (2001). Less really is more for adults learning a miniature artificial language. *Journal of Memory & Language*, **44**, 250-273.

LEE, Y.-S. (1995). Effects of learning contexts on implicit and explicit learning. *Memory & Cognition*, **23**, 723-734.

LOTZ, A., & KINDER, A. (2006). Transfer in artificial grammar learning: The role of repetition information. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, **32**, 707-715.

MATHEWS, R. C., BUSS, R. R., STANLEY, W. B., BLANCHARD-FIELDS, F., CHO, J. R., & DRUHAN, B. (1989). Role of implicit and explicit processes in learning from examples: A synergistic effect. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, **15**, 1083-1100.

MEULEMANS, T., & VAN DER LINDEN, M. (1997). Associative chunk strength in artificial grammar learning. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, **23**, 1007-1028.

NAGATA, H. (1977). The interaction between syntactic complexity and semantic reference in acquisition of a miniature artificial language. *Japanese Psychological Research*, **19**, 90-96.

NEWPORT, E. L. (1990). Maturational constraints on language learning. *Cognitive Science*, **14**, 11-28.

NISSEN, M. J., & BULLEMER, P. (1987). Attentional requirements of learning: Evidence from performance measures. *Cognitive Psychology*, **19**, 1-32.

PERRUCHET, P., & VINTER, A. (1998). Learning and development: The implicit knowledge assumption reconsidered. In M. A. Stadler & P. A. Frensch (Eds.), *Handbook of implicit learning* (pp. 495-531). Thousand Oaks, CA: Sage.

POTHOS, E. M. (2007). Theories of artificial grammar learning. *Psychological Bulletin*, **133**, 227-244.

REBER, A. S. (1967). Implicit learning of artificial grammars. *Journal of Verbal Learning & Verbal Behavior*, **6**, 855-863.

REBER, A. S. (1976). Implicit learning of synthetic languages: The role of instructional set. *Journal of Experimental Psychology: Human Learning & Memory*, **2**, 88-94.

REBER, A. S. (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*. London: Oxford University Press.

REBER, A. S., & ALLEN, R. (1978). Analogy and abstraction strategies in synthetic grammar learning: A functionalist interpretation. *Cognition*, **6**, 189-221.

REBER, A. S., KASSIN, S. M., LEWIS, S., & CANTOR, G. (1980). On the relationship between implicit and explicit modes in the learning of a complex rule structure. *Journal of Experimental Psychology: Human Learning & Memory*, **6**, 492-502.

REED, J. [M.], & JOHNSON, P. [J.] (1994). Assessing implicit learning with indirect tests: Determining what is learned about sequence structure. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, **20**, 585-594.

REED, J. M., & JOHNSON, P. J. (1998). Implicit learning: Methodological issues and evidence of unique characteristics. In M. A. Stadler & P. A. Frensch (Eds.), *Handbook of implicit learning* (pp. 261-294). Thousand Oaks, CA: Sage.

ROHDE, D. L. T., & PLAUT, D. C. (1999). Language acquisition in the absence of explicit negative evidence: How important is starting small? *Cognition*, **72**, 67-109.

SOETENS, E., MELIS, A., & NOTEBAERT, W. (2004). Sequence learning and sequential effects. *Psychological Research*, **69**, 124-137.

STADLER, M. A. (1992). Statistical structure and implicit serial learning. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, **18**, 318-327.

VAN DEN BOS, E., & POLETIEK, F. H. (2008). Intentional artificial grammar learning: When does it work? *European Journal of Cognitive Psychology*, **20**, 793-806.

NOTE

1. In the present study, grammatical and ungrammatical test strings did not differ in their local or global repetition structure for any of the grammars. Anchor chunk strength was significantly higher for grammatical than for ungrammatical test strings ($p < .002$ for each grammar), and edit distance was significantly lower for grammatical than for ungrammatical test strings ($p < .001$ for each grammar). So, for all the grammars, the participants could potentially base their grammaticality judgments on knowledge of anchor chunks or knowledge of specific strings presented in the induction phase. Importantly, however, there was no significant interaction between grammaticality and grammar for either anchor chunk strength [$F(9,480) = 0.082$] or edit distance [$F(9,480) = 1.545$, $p = .129$]. Therefore, the effects of complexity reported in the present article were not confounded by any of these measures.

APPENDIX A
The Computation of Topological Entropy

The TE measure of complexity of finite state grammars developed by Bollt and Jones (2000) is based on two assumptions. First, the grammar is assumed to be unambiguous: Each arrow leaving a certain state has to generate a different symbol. Second, the measure is developed for infinite strings, but for practical purposes, concatenations of finite strings can be used. The computation of TE is an extension of a procedure previously proposed by Chomsky and Miller (1958) to compute the informational capacity of each symbol generated by an artificial grammar.

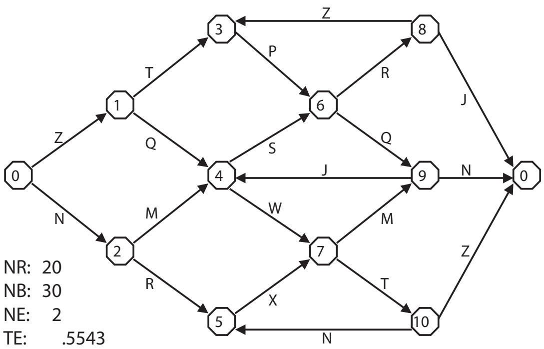
According to this basic procedure, the grammar, usually depicted in a graph, has to be translated into a transition matrix. In Chomsky and Miller's (1958) approach, the elements of the matrix stand for the number of paths of length 1 that provide a transition from the state assigned to a column of the matrix to the state assigned to a row. Subsequently, the largest nonnegative eigenvalue of the matrix is computed, and a logarithm is taken. TE is defined as the natural logarithm of the largest nonnegative eigenvalue of a grammar's transition matrix (Bollt & Jones, 2000).

However, in contrast to Chomsky and Miller (1958), who assumed that the current state of the grammar is always known, Bollt and Jones (2000) assumed that the current state of the grammar has to be inferred from the string. In their approach, the elements of the matrix stand for transitions between the grammar's symbols, instead of transitions between states. Both approaches lead to the same matrix if all the arrows in the graph are associated with a unique symbol. In this case, the current state of the grammar is directly given by the symbol, and therefore, it can be unambiguously established which symbols are legal successors.

However, if a symbol is associated with more than one link in the grammar, it is ambiguous by which symbols it may be followed. In such cases, according to Bollt and Jones (2000), the current state of the grammar has to be inferred from the current symbol and one or more symbols preceding it. Therefore, they proposed to lift the transition matrix. Instead of transitions between symbols, the lifted matrix consists of transitions between groups of symbols, the size of which equals the minimum number of symbols needed to unambiguously predict the legal successors of each element. Because of this lifting technique and the assumptions underlying it, TE is sensitive to the presence of long-distance dependencies and is correlated with the number of elements required to predict the next.

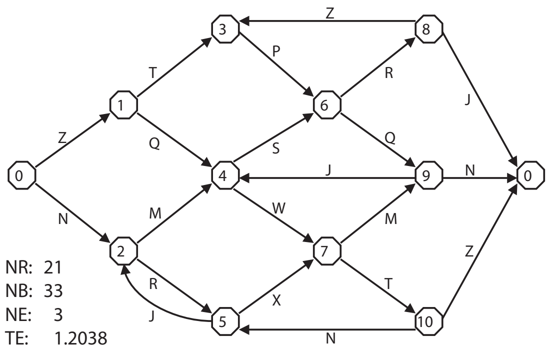
APPENDIX B
Ten Grammars and Their Scores on Four Complexity Measures

Grammar A



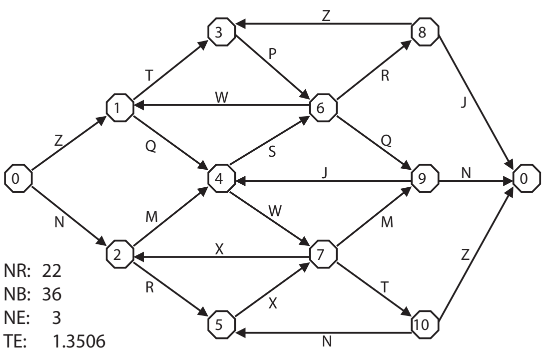
NR: 20
NB: 30
NE: 2
TE: .5543

Grammar B



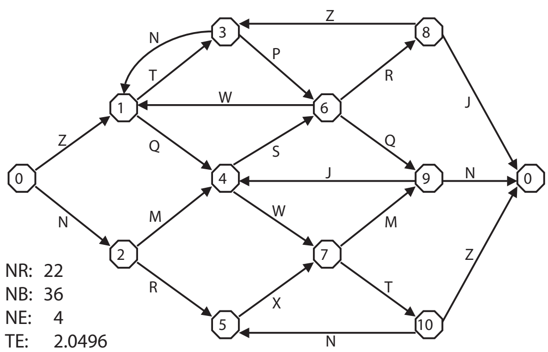
NR: 21
NB: 33
NE: 3
TE: 1.2038

Grammar C



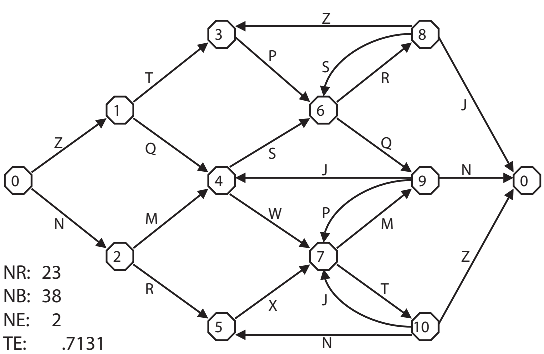
NR: 22
NB: 36
NE: 3
TE: 1.3506

Grammar D



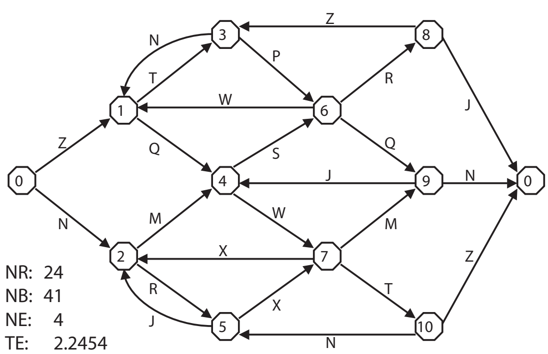
NR: 22
NB: 36
NE: 4
TE: 2.0496

Grammar E



NR: 23
NB: 38
NE: 2
TE: .7131

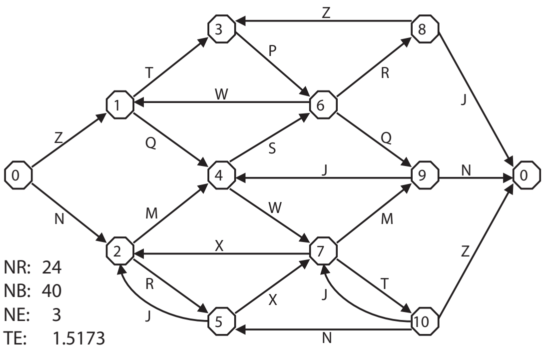
Grammar F



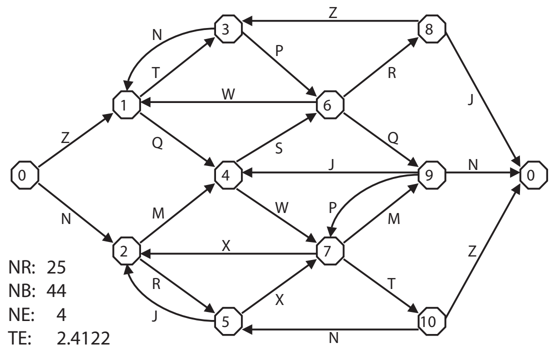
NR: 24
NB: 41
NE: 4
TE: 2.2454

APPENDIX B (Continued)

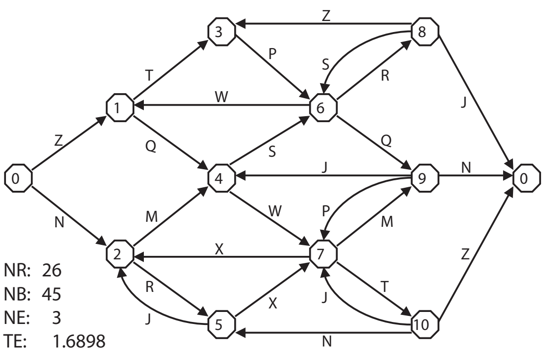
Grammar G



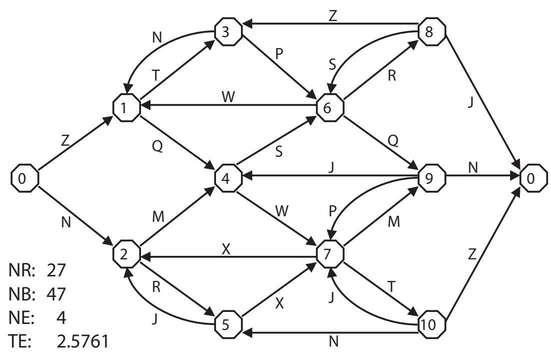
Grammar H



Grammar I



Grammar J



Note—NR, number of rules; NB, number of bigrams; NE, number of elements required to predict the next; TE, topological entropy.

(Manuscript received October 4, 2007;
revision accepted for publication March 31, 2008.)