Linguistic Priming and Learning Adjacent and Non-Adjacent Dependencies in Serial Reaction Time Tasks

Felix Hao Wang (wang970@usc.edu)[a], Elsi Kaiser (emkaiser@usc.edu)[b]

[a]Department of Psychology, University of Southern California, 3620 McClintock Ave, Los Angeles, CA, 90089
[a]Department of Linguistics, University of Southern California, Los Angeles, CA, 90089

Abstract

Although syntactic priming is well studied and commonly assumed to involve implicit learning, the mechanisms behind this phenomenon are still under debate. We tested whether implicit learning of adjacent and non-adjacent sequences occurs in a non-linguistic, finger sequence task (Serial Reaction Time task), and if so, whether these implicitly-learned dependencies can cause syntactic priming in the linguistic domain. We followed the logic that exposure to statistical patterns in the SRT task may influence language users' relative clause (RC) attachment biases, and trained participants on SRT sequences with adjacent or non-adjacent dependencies. Participants then wrote completions to relative clause fragments in a situation where they could opt for adjacent or non-adjacent linguistic structures. Participants successfully learned the adjacent and non-adjacent dependency implicitly during the SRT task, but, strikingly, their RC continuations did not exhibit priming effects. Implications for theories of syntactic priming and its relations to implicit learning are discussed.

Keywords: implicit learning; syntactic priming; relative clause attachment bias; non-adjacent dependencies

Introduction

Although the phenomenon of syntactic priming has been well studied in the literature (e.g., Bock, 1986; Pickering & Branigan 1998), the exact processes behind priming are still unclear. A burgeoning literature on syntactic priming has taken implicit learning as the one of the mechanisms for syntactic priming (e.g., Bock & Griffin, 2000; Chang, Dell & Bock 2006). In this paper, we build on insights from research on implicit learning research to look at adult sentence processing regarding the representation of abstract dependencies in language and other cognitive domains (Scheepers & Sturt, 2014; Menon & Kaiser, 2014; Van de Cavey & Hartsuiker 2016). We explore whether an abstract relation represented through statistical regularities from a finger sequence task (Serial Reaction Time Task, or SRT) can prime the attachment biases of relative clauses. Specifically, do adjacent and non-adjacent prediction relations derived from statistics prime the low versus high attachment preference during the production of English relative clauses?

The notions of adjacency and non-adjacency are relevant to the domain of syntax with regard to the representation of relative clauses (RCs). In English, sentences with the structure "NP1 of NP2 who" (e.g., *Jessica visited the doctor of the supermodels who…*) are structurally ambiguous as to which NP the relative clause attaches to. A relative clause can potentially attach to either one of the NPs. When the intended meaning is "the doctor lived in Los Angeles" (e.g. *Jessica visited the doctor of the supermodels who <u>lives</u> in Los Angeles)*, the relative clause attaches to the first NP (NP1), a structure called high attachment. When the relative clause attaches back to the lower NP (NP2) (e.g. *Jessica visited the doctor of the supermodels who <u>live</u> in Los Angeles*), this is a low attachment structure. Thus, relative clauses may modify the adjacent (local) noun (low attachment) or the non-adjacent noun (high attachment). Although such constructions may be ambiguous as to the intended attachment site, it is often the case that linguistic cues (e.g. *lives* vs. *live*) signal whether a speaker intended a low or a high attachment.

According to Scheepers 2003, the syntactic distinction between high and low in relative clause attachment bias is only a matter of syntactic sequencing: The syntactic rules used to generate high and low attachment relative clauses are the same. The only distinction between the two is that, in low attachment, the relative clause modifies the noun immediately preceding it, and in high attachment, it modifies the noun that non-adjacently precedes it. In our opinion, this provides a good opportunity for priming to be probed on an abstract level. Our previous work investigated the relationship between artificial language learning and structural priming, and suggested that relative clause attachments can be primed from statistical learning of artificial languages (Wang, Menon & Kaiser, under review). We found that participants who learned an artificial language with non-adjacent dependencies are more likely to complete relative clauses with high attachment, compared to a baseline group as well as compared to participants who were exposed to an artificial language with adjacent dependencies. Van de Cavey & Hartsuiker (2016), among others, have also provided evidence that an array of different experimental tasks (e.g., sentence comprehension, music listening, and math solving) can also prime relative sentence completion.

However, the question regarding the underlying mechanism for syntactic priming is not fully answered by existing work. Given the nature of relative clause attachment, it is not clear whether existing proposals regarding the mechanisms of syntactic priming can explain results from recent syntactic priming work, especially data from studies that successfully used non-linguistic stimuli to induce syntactic priming (e.g. musical sequences or math equations). These kinds of data suggest that learning and processing sequences of numbers, musical notes and other information involves *domain-general implicit learning*, and that this kind of domain-general implicit learning is involved in syntactic priming. This predicts that other kinds of implicit learning should also trigger syntactic priming. However, are there limits to what can trigger syntactic priming? If other kinds of domain-general implicit learning do *not* result in syntactic priming, we may need to reexamine current models of syntactic priming.

To investigate these issues, we used a Serial Reaction Time Task (SRT task, Howard & Howard, 1997) to induce implicit learning. Our goal is to induce domain-general implicit learning of sequences. In an SRT task, participants generally find it very difficult to articulate *explicit* rules for the probabilistic finger sequences they are implicitly learning, whereas other tasks (such as math tasks and music tasks) involve a mixture of explicit and implicit learning that is difficult to untangle. This makes SRT tasks one of the best tasks for our goal of investigating *implicit learning*, given the long history of using SRT to explore implicit learning.

Our study tests whether structural representations arising from distributional information – in this case, finger sequences – can prime relative clause completions. If relative clause attachment biases come from representations that are completely separate from domain-general sequence processing representations, representations constructed based on finger sequences will not result in any changes in the completion of relative clauses. On the other hand, if relative clause attachment biases come from representations that are shared with domain-general sequential representations, the relative clause bias is predicted to change as a result of learning sequential statistics and sequence processing in general.

Experiment

In this experiment, we test whether implicit learning of finger sequences that involve adjacent or non-adjacent dependencies influences whether participants produce completions for incomplete relative clauses fragments that involve adjacent or non-adjacent syntactic dependencies.

Methods

Participants. Seventy-two undergraduates participated.

Stimuli. There are two tasks in the experiment: an SRT task and a sentence completion task. In the SRT task, we used a three circles and a dog image to indicate a position (Figure 1). The dog image changed location depending on which position was being represented. Figure 1 is an example of position 2 (since the dog is in the second position from the left). There are four possible positions. Participants' task was to press a key indicating what the position of the dog is. Participants saw multiple such screens in succession, with varying locations of the dog image (e.g. in position 2 on one screen, in position 1 on the next screen, and so on).

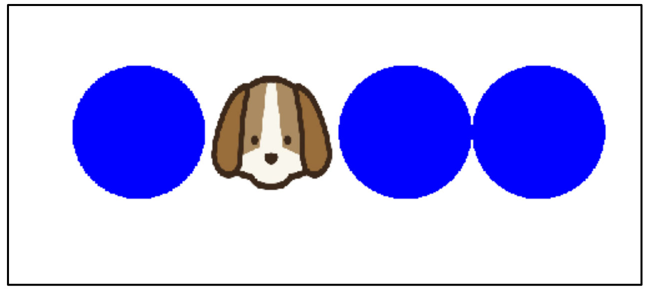

Figure 1. Interface during the SRT task. This example image depicts position 2 (since the dog is in the second position from the left).

In the sentence completion task, we created sentence fragments for participants to complete in the testing phase. There are two kinds of sentence fragments: targets and fillers. There were 30 target sentence fragments. All target sentence fragments have the structure shown in Example (1): In order to complete the fragments, participants write a relative clause modifying a complex noun phrase made of two noun phrases. As shown in Example 1, NP1 (the doctor) and NP2 (the supermodels) are connected by the preposition 'of' and are followed by the relative pronoun 'who', providing the possibility that the continuation can modify either the NP1 or NP2.

(1) John meets [the doctor]$_{NP1}$ of [the supermodels]$_{NP2}$ [who …]$_{RC}$.

In target items, the subject of sentence was always a proper name (e.g. *John*, equal numbers of male and female names), and the two NPs that make up the object (e.g. *the doctor of the supermodels)* were definite, animate nouns preceded by the definite article. NPs were controlled for number: all sentence fragments had NP1 as singular and NP2 as plural (e.g. *the doctor of the supermodels*)[1]. The differences in number makes coding of attachment easier because number marking on the verb can potentially disambiguate the attachment (e.g. *…was happy* vs. *…were happy*). For the same reason, the verbs in the main clause were in the present tense (in 3rd singular form). All verbs in the target fragments (e.g. *counted*) were non-implicit causality verbs, chosen in order to avoid verb semantics bias. Fillers were non-ambiguous English sentence fragments of similar length.

Design. The experiment consisted of sixteen blocks. Blocks 1-10, 12 and 14 consisted of Serial Reaction Time Tasks. Block 11, 13 and 15 were sentence completion tasks. Block 16 was an explicit assessment of the knowledge regarding the patterns presented in SRT task.

[1] We used singular-plural NPs only because our earlier work on this topic indicates that variance from priming came mostly with the singular/plural items (Wang, Menon & Kaiser, under review).

Serial Reaction Time Task Design. In the Serial Reaction Time task, we used a between-subjects task where half of the participants (n=36) were trained on adjacent dependencies and the other half (n=36) were trained non non-adjacent dependencies. A total of 2304 trials (i.e., screens like Figure 2, but with the dog image in varying locations) were created for both adjacent and non-adjacent dependency conditions, with 192 trials in each of the 12 blocks (described below). Between each block, participants were given information about whether the next block is an SRT block or a sentence completion block, as well as a chance to take a self-paced break.

On each trial within an SRT block, participants responded to positional information on the screen (represented by the position of the dog image), and pressed one of four keys on the keyboard (z, x, n or m) that corresponds with the position of the dog on that particular trial. After responding to the position correctly, there is a 120ms blank screen, followed by the next trial with the dog image at the position for the next trial. If the response is incorrect (if the participant pressed any key on the keyboard other than the intended key), the dog display (e.g. Figure 1) remains on the screen until the correct key is pressed. The sequence of the positions is determined by the condition that a participant was in (adjacent vs. non-adjacent dependency group).

In the *adjacent dependency condition*, 4 triplets of sequences were concatenated: R41, R13, R21, R34, where the letter R represents a random position in a display, and 1, 2, 3, and 4 represent the four positions from left to right. Note that there could be repetitions of the same position, and participants would have pressed the same key twice in that instance. The sequence was constructed such that each of the 4 triplets occurred an equal number of times, and within all occurrences of each triplet, all 4 positions occurred equal number of times. This sequence generation procedure ensures that the only dependencies are the intended *adjacent dependencies* (the dog head in position 4 on one display predicts that the dog head will be in position 1 on the next display, 1 predicts 3 on the next display, 2 predicts 1 and 3 predicts 4). For example, a dog head in position 1 can be preceded by a dog head in any of the four positions, but the probability that it was in position 4 is 3 times as the probability that it was in position 1, 2 or 3, which makes the transitional probability of 1 given 4 50% while the others are at 16.67%. Adjacency matrices were calculated (predictions of the current item given the immediate previous item, the item previous to that, and so forth). Up to the 5^{th} order adjacency matrices were examined to make sure that no higher order prediction is possible from this sequence. An example sequence can be viewed in Example (2). In essence, participants in this group were exposed to *adjacent dependencies* across trials, since position 4 on one display predicts position 1 on the next display, 1 predicts 3 on the next display, and so on.

In the *non-adjacent dependency condition*, 4 triplets of sequences were concatenated: 4R1, 1R3, 2R1, 3R4, where the letter R represents a random position. Similar measures were taken as in the adjacent dependency condition to ensure that the only predictive relationship exists in the second order. Thus, participants in this group were exposed to non-adjacent dependencies across trials (e.g. a dog head in position 4 predicts a dog head in position 1 not on the immediately following display but on the display after that; 4R1).

The trials in the SRT were determined by a predetermined sequence, each containing relevant statistics. Two sequences, one each for the two conditions were generated (adjacent and non-adjacent). All subjects were tested on the same sequence by the condition they were in. These two sequences only differed in terms of whether the R position is in the first position of the triplet or the second position of the triplet, demonstrated in example (2) and (3):

(2) [Adjacent] 3 4 2 1 1 3 4 4 2 4 4 2 2 2 1 …
(3) [Non-adjacent] 4 3 2 1 1 3 4 4 2 4 4 2 2 2 1 …

Sentence Completion Task Design. Each participant completed 30 target sentence fragments and 48 filler sentence fragments split into three blocks (10 target and 16 filler in blocks 11, 13 & 15), in a randomized order. The targets and fillers were held constant across participants. The randomized order made sure that no two consecutive sentence fragments are target sentence fragments.

Procedure. Participants were informed that they will be doing two kinds of tasks. Then the study proceeded first with the instructions for the SRT. The instruction emphasized to use the index and middle fingers to press the four keys on the keyboard, and a demonstration phase was included to make sure participants understand the process.

After the demonstration of the SRT task, participants saw the instructions for the sentence completion task. The participants' task was to write a completion for sentence fragments (e.g. *John meets the doctor of the supermodels who*). They could write whatever first came to mind. We analyzed the target completions for whether the relative clause modifies NP1 (e.g. *who supports vaccination*) or NP2 (e.g. *who like to travel*). RCs that modify NP1 are high attachments and RCs that modify NP2 are low attachments.

In the 16th block, participants were tested on their explicit knowledge regarding the positions of the dog image. Along with the interface in Figure 2, participants were told: *"Earlier on, you saw one dog per screen. Now, you will see three rows of circles. You should imagine that these are three different screens. The question is, where will the dog on the last screen, i.e., in the third row, appear?"*

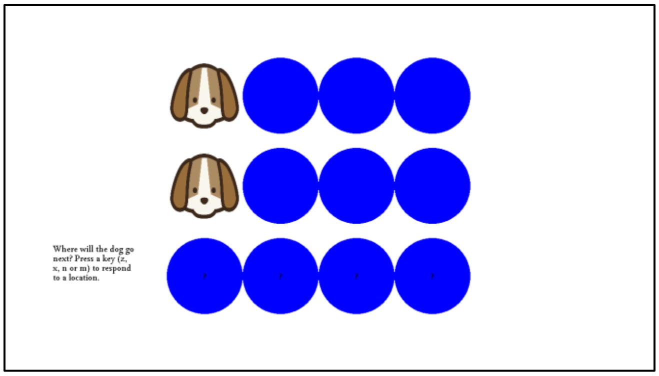

Figure 2. Interface for explicit knowledge test.

In other words, participants were asked to make judgments on where the dog head will appear given two preceding positions. This block tests whether participants have *explicit* knowledge regarding the location patterns. In this block, a total of 48 trials were given to each participant, with 3 repetitions of 16 different questions (full combinations of positions in the first and second row).

Results

We obtained three kinds of data: (i) data from the SRT task, including blocks of 1-10, 12 and 14; (ii) data from the sentence completion task, including blocks 11, 13 and 15; and (iii) the explicit knowledge task for block 16.

Serial Reaction Time Task Results For each trial in the SRT section, there is a reaction time associated with that trial. Each trial can be classified as a predictable trial or an unpredictable trial. Trials are predictable when the previous trial is predictive of the current trial. For example, in the adjacent condition, all trials with the dog head position 1 are predictable if the trial immediate before it had the dog head in position 4. As another example, in the non-adjacent dependency condition, all trials with the dog head in position 1 are predictable if two trials before it the position of the dog head was 4.

For the *adjacent dependency condition*, predictable trials were responded to faster than unpredictable trials. The mean RT for predictable trials is 415ms and the mean RT for unpredictable trials is 443ms. We used a mixed-effect linear regression to examine this difference statistically. Using subjects as a random effect and predictability of trials and block as fixed effects, we found that RT is faster for predictable trials (β= -27.86, z=-16.09, p<0.001). Block also significantly predicts RTs. For ease of modelling, we coded block as a continuous variable with block number 12 and 14 replaced as 11 & 12. There is a significant effect of block (β=-5.459, z=-21.75, p<0.001). The more the participants practice/the more blocks they go through, the fast the RTs become. In sum, participants showed learning of the adjacent statistics in general (Figure 3).

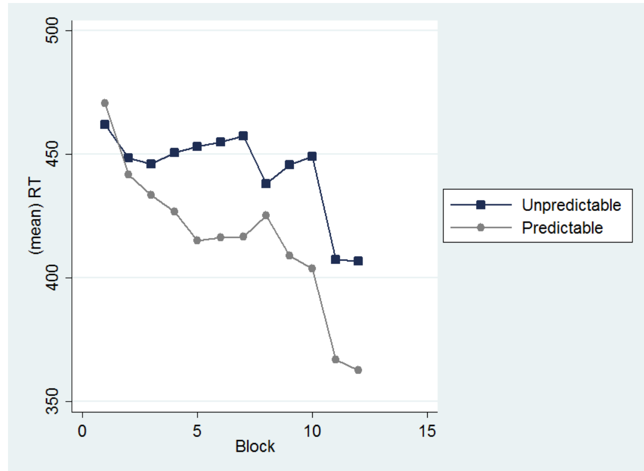

Figure 3. Mean of RT data from adjacent dependency condition. Blocks 11 and 12 in the figure represent data from experiment block 12 and 14, respectively.

For the *non-adjacent dependency condition*, predictable trials were also responded to faster than unpredictable trials. The mean RT for predictable trials is 413ms and the mean RT for unpredictable trials is 419ms. We used a mixed-effect linear regression to examine this difference statistically. Using subjects as a random effect and predictability of trials and block as fixed effects, we found that RT is faster for predictable trials (β= -5.13, z= -3.70, p<0.001). Block also significantly predicts RTs. For ease of modelling, we coded block as a continuous variable with block number 12 and 14 replaced as 11 & 12. The effect of block is significant (β=-5.61, z= -27.96, p<0.001). Similar to what we found for the adjacent group, participants become faster with practice. In sum, participants showed learning of the non-adjacent statistics in general as well, though the effect size may not be as large (Figure 4).

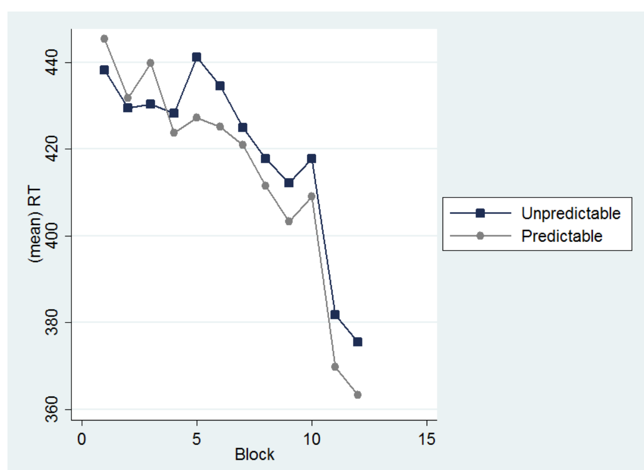

Figure 4. Mean of RT data from non-adjacent dependency condition. Blocks 11 and 12 in the figure represent data from experiment block 12 and 14, respectively.

Turning to the *sentence completion data*, we coded all the target sentence completion trials to see how participants completed the relative clause fragments. The coding of the sentences resulted in three types: high attachment (HA), low attachment (LA), and ambiguous. Coding was done with mostly syntactic considerations, given that the two NPs in our sentences differ in number. The main clue for coding comes from the verb in the relative clause that modifies the "who". More specifically, the verb in the relative clause in most sentences showed overt morphological agreement with the relevant NP. If verb number did not disambiguate (e.g. *went, asked*), semantic cues were used (e.g. Emily worked with the mother of the children *who just got tenure* => high attachment, vs. Chris counted the fans of the singer *who just finished the encore* => low attachment). If both verb marking and semantic cues were unclear, the sentence was coded as ambiguous. Alternatively, if the sentence completion was done without making the "who" the subject of the relative clause, or the sentence failed to make sense for coding purposes, it was coded as ambiguous as well.

To prepare for the mixed-effect logistic regression, high attachment sentences were coded as 1 and low attachment sentences as 0, and ambiguous was treated as missing. In the logistic regression, condition (adjacent/non-adjacent) was entered as the fixed effect, and subjects were entered as the random effect. We found no evidence of priming (β=0.031, z=0.38, p=0.704, ns). A graph of the proportions of high and low attachment completions as function of SRT group (adjacent vs. non-adjacent dependencies in SRT task) is in Figure 5. We find no signs of non-adjacent dependencies (NAD) in the SRT task priming (boosting the rate of) high attachments, compared to adjacent dependencies (ADJ).

Lastly, we examined the responses to the *explicit judgment questions* in block 16. We found that participants were 22.3% correct in their answers in the adjacent dependency condition, and 19.3% correct in the non-adjacent dependency condition. Both of these levels are statistically below chance (25% in a 4-alternative forced choice), suggesting that participants have no *explicit* knowledge of predictive relationships in the SRT task. Thus, participants have not formed explicit knowledge regarding how previous trials in the sequences are predictive of upcoming element in both the adjacent and non-adjacent condition – although their RT patterns provide evidence of *implicit* learning.

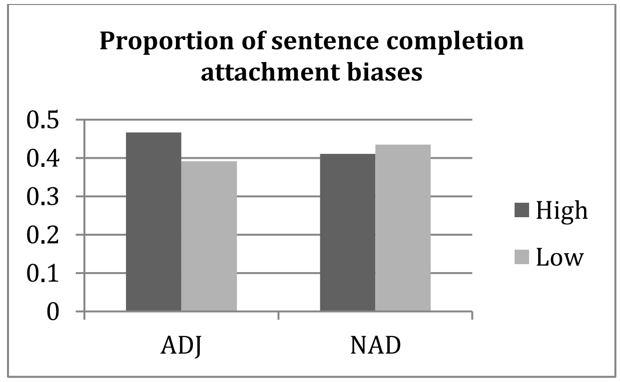

Figure 3. Proportions of high and low attachment completions as function of SRT group (adjacent vs. non-adjacent dependencies)

Discussion

We found evidence for implicit learning but no evidence of explicit learning in the SRT task. We also found no evidence of syntactic priming even when implicit learning of adjacent/non-adjacent statistics was successfully induced.

It is worth taking a closer look at the SRT task used in this experiment. The sequences in this experiment contained only one predictive relationship for each trial. That is, the only predictive relation in the adjacent condition is the current trial given the immediately previous one, and the current trial given the trial before the last in the non-adjacent dependency condition. The most common SRT task that induces representation of non-adjacency is the ASRT task (Alternating Serial Reaction Time task, Howard & Howard, 1997). In the ASRT task, participants are exposed to patterns such as 1R2R3R4R. Predictive relations exist between elements of distance 2, 4, 6, and so on. Crucially, given the configuration of ASRT sequences, it is not clear which previous trial was used for prediction for predictable trials. All we know is that non-adjacent dependencies from previous trials showed facilitated processing. This feature makes ASRT sequences unsuitable for the current relative clause priming work, because the contrast between adjacent to non-adjacent dependencies needs to be controlled. The difference between an adjacent dependency sequence and ASRT is not only between the distance between the current item and the previous item predicting it, but also the number of items that can predict the current trial. Our design avoided this difference by controlling the number of non-adjacent dependencies in the non-adjacent dependency condition.

The purpose of our SRT task was to induce implicit learning in a domain-general sense, which was successful. Indeed, we chose to use SRT to engage implicit learning and to minimize learning that exhibits explicit rule-seeking behavior. In other words, using SRT allows us to target domain general implicit learning specifically. In light of this, and given existing views regarding implicit learning in syntactic priming, there are a number of implications for syntactic priming that can be drawn from our study.

To begin with, an important question in this field is how implicit learning may induce syntactic priming. Previous work (Chang, Dell & Bock, 2006) used a connectionist model to specify how the process of implicit learning happens. In that model, it was assumed that reading sentences of a particular structure changed the weights over that structure such that the bias for the structure increased. This was in turn used to explain, in production, why syntactic priming occurs. This model provided a computational account of how implicit learning influenced syntactic behaviors of humans, but it may require further specification. For example, does the implicit learning process need to happen in the language domain in order for syntactic priming to happen? One possibility is that, in order for syntactic priming to occur, the implicit learning that takes place must be in the linguistic domain. Indeed, existing studies with linguistic materials that engaged implicit learning also successfully induced syntactic priming (Fine & Jaeger 2013; Hartsuiker, Bernolet, Schoonbaert, Speybroeck & Vanderelst, 2008). We recently provided evidence that learning an artificial language with different dependency structures successfully induced syntactic priming (Wang, Menon & Kaiser, under review). Whether the domain of implicit learning matters for syntactic priming needs to be examined more closely.

This also bring up the question of whether implicit learning in a domain-general sense (that is, other than the domain of language) can induce syntactic priming. We found implicit learning of adjacent and non-adjacent SRT finger sequences but no syntactic priming of adjacent and non-adjacent RC attachments. How do these results fit with existing data suggesting that non-linguistic representations in math and music can cause syntactic priming?

Existing evidence with non-linguistic priming (Van de Cavey & Hartsuiker, 2016; Scheepers et al., 2011; Scheepers & Sturt, 2014) may have involved both explicit rule representations in the non-linguistic domains. A subset of the experimental tasks used in these studies (arithmetic operations processing, for example), explicitly involves the use of rule-like positional operations. Thus, it may be that these findings cannot be explained by implicit learning and instead involve some other domain-general mechanisms. For example, if some kind of explicit learning drives syntactic priming, future work needs to be done to examine how this takes place. One possibility is that priming happens on a metacognitive level: Is syntactic priming in these studies the unintended result of participants trying to figure out the aims/goals of experiments, such that priming is present only when participants feel like the experiment is intending them to produce sentences with certain kinds of dependencies? Or is syntactic priming in these explicit manipulations a matter of unconscious, linguistic phenomenon, which, arguably, priming is supposed to be?

Although we do not offer conclusive answers to these questions here, our findings highlight the importance of investigating the effects implicit and explicit learning on syntactic priming (or lack thereof). We found that a finger sequence task with SRT which successfully induced implicit learning was unable to prime relative clause attachment biases and also failed to result in explicit learning of SRT sequences. This work provides preliminary evidence that implicit knowledge from a domain-general sequencing task alone does not induce syntactic priming. Future work is needed to examine whether this observation applies to other non-linguistic domains in terms of syntactic priming.

References

Bock, J. K. (1986). Syntactic persistence in language production. *Cognitive psychology*, 18(2), 355-387.

Bock, K., & Griffin, Z. M. (2000). The persistence of structural priming: Transient activation or implicit learning? *Journal of Experimental Psychology: General, 129*, 177-192.

Chang, F., Dell, G. S., & Bock, J. K. (2006). Becoming syntactic. *Psychological Review*, 113, 234–272.

Fine, A. B., & Florian Jaeger, T. (2013). Evidence for implicit learning in syntactic comprehension. *Cognitive Science*, *37*(3), 578-591.

Hartsuiker, R. J., Bernolet, S., Schoonbaert, S., Speybroeck, S., & Vanderelst, D. (2008). Syntactic priming persists while the lexical boost decays: Evidence from written and spoken dialogue. *Journal of Memory and Language*, 58(2), 214-238.

Howard Jr, J. H., & Howard, D. V. (1997). Age differences in implicit learning of higher order dependencies in serial patterns. *Psychology and aging*, 12(4), 634.

Menon, M. & Kaiser, E. (2014). *Priming from Music to Language: Effects of Number Marking on the Priming of Relative Clause Attachment in Spanish*. Poster, AMLaP 2014 at University of Edinburgh, Scotland.

Rohde, H., Levy, R., & Kehler, A. (2011). Anticipating explanations in relative clause processing. *Cognition*, 118(3), 339-358.

Pickering, M. J., & Branigan, H. P. (1998). The representation of verbs: Evidence from syntactic priming in language production. *Journal of Memory and Language*, 39(4), 633-651.

Scheepers, C. (2003). Syntactic priming of relative clause attachments: Persistence of structural configuration in sentence production. *Cognition*, 89(2), 179-205.

Scheepers, C, & Sturt, P. (2014). Bidirectional syntactic priming across cognitive domains: From arithmetic to language and back. *Quarterly Journal of Experimental Psychology*. 67(8):1643-1654.

Van de Cavey, J., & Hartsuiker, R. J. (2016). Is there a domain-general cognitive structuring system? Evidence from structural priming across music, math, action descriptions, and language. *Cognition*, 146, 172-184.

Wang, F. H., Menon, M. & Kaiser, E. (under review). Statistical Structures in Artificial Language Primes Relative Clause Attachment.

Wang, F. H. & Mintz, T. H. (in revision). Characterizing the Difference Between Learning about Adjacent and Non-adjacent Dependencies.