What affects Priming Strength?
Simulating Structural Priming Effect with PIPS

Zhenghao Zhou and **Robert Frank**
Yale University
370 Temple Street
New Haven, CT 06511
herbert.zhou@yale.edu and robert.frank@yale.edu

1 Introduction

Over the past thirty years, numerous psycholinguistic studies have shown the *structural priming effect* (also called *syntactic priming*) (Dell and Ferreira, 2016): speakers tend to reuse the syntactic structures they have recently encountered during production (Bock, 1986). For example, speakers tend to produce a double object (DO) structure (e.g. *The student sent the professor a letter*) rather than a prepositional dative (PD) structure (e.g. *The student sent a letter to the professor*) after encountering a DO sentence (e.g. *Alice gave Bob a book*). Structural priming is observed even when no word is shared between prime and target sentences (Pickering and Branigan, 1998).

Two mainstream theories have been proposed to account for structural priming, specifically the factors that affect priming strength. Pickering and Branigan's (1998) lexical activation theory claims that activation of the representations that have been accessed to produce or comprehend a structure persists for a short time, so the representations can be reused on the next relevant opportunity. The lexical activation theory correctly predicts that the structural priming effect is stronger when the word that heads the primed structures is repeated between prime and target sentences, which is known as the *lexical boost effect* (Pickering and Branigan, 1998). For example, if the target sentence is *Alice gave Bob a book*, the structural priming effect is stronger if the prime is *Carl gave Danis a letter* rather than *Alice showed Bob a book*.

Alternatively, the implicit learning theory by Chang et al. (2006) claims that humans implicitly learn probabilistic information about different structures from experience and use such information to predict the form of a prime sentence. Crucially, priming strength is determined by the difference between the predictions from probabilistic information and the actual input. Therefore, the implicit learning theory predicts the *inverse frequency effect* (Jaeger and Snider (2007), Bernolet and Hartsuiker (2010)): less preferred syntactic alternatives (measured by the relative frequency in the learner's experience against those of the counterparts) cause stronger overall priming than more preferred structures. For example, since *give* is biased towards DO in English, a prime sentence with *give* in a PD structure will cause greater priming effect than a prime sentence with *give* in a DO structure.

Recently, Cho et al. (2020) and Smolensky et al. (2022) proposed the Gradient Symbolic Computation (GSC) framework as a general model of human cognitive processing. Brehm et al. (2022) instantiated this framework in a model of the incremental processes involved in language production, resulting in the Parallelism in Producing Syntax (PIPS) model. Brehm et al. (2022) showed that PIPS can effectively simulate the agreement attraction effect (Bock and Miller, 1991) in language production with the preamble-completion paradigm. In the current paper, we show that the same PIPS model can be used to model the strength of structural priming. Specifically, our simulation results suggest that the PIPS model can qualitatively reproduce both the lexical boost effect and the inverse frequency effect observed in humans by varying the model parameters determining the strength and modes of priming, as well as the hyperparameters determining the internal representation of the model.

2 Background: GSC and PIPS

In this section, we briefly introduce relevant features of GSC and PIPS, but refer the reader to the original papers for mathematical and implementational details (Cho et al. (2020), Brehm et al. (2022)). A GSC parser simulates a continuous-time, continuous state cognitive system with a neural network that uses tensor product representation (filler-role bindings) to encode binary tree structures — decomposable vector representations of

Proceedings of the Society for Computation in Linguistics (SCiL) 2023, pages 413-417.
Amherst, Massachusetts, June 15-17, 2023

symbolic structures (Cho et al., 2020). As illustrated in Cho et al. (2018), to encode a tree structure `S[1](A,B)`, we can represent each unique position by a *role* and bind the content (i.e. *filler*) at that position to the role. If we assign roles `r,0,1` to the root, the left child, and the right child, respectively, the filler/role binding of the structure is: `S[1](A,B)` $\equiv$ {`B/1,S[1]/r,A/0`}.

As a stochastic dynamic system, a GSC parser computes a discrete structure gradually by optimizing over a set of grammatical and non-grammatical constraints. The grammatiacl constraints, defined in terms of Harmonic Grammar (Hale and Smolensky, 2006), impose a reward or a penalty on the wellformedness of a gradient symbolic structure. Optimizing over grammatical constraints means finding the structures that best satisfy the constraints of the grammar. To model the structural uncertainty during incremental parsing, the GSC parser yields a conjunctive blend of multiple possible parsing structures simultaneously. It is forced to converge to a final parsing decision within fixed time steps through a commitment policy q. As q increases at each time step, the parser is forced to move closer to *grid states*, at which the bindings of each role to all symbols have activation 0 except one, which has activation 1 (Cho et al., 2018). When q reaches its maximum value, the parser commits to a parsing structure.

PIPS implements a GSC parser for sentence production by co-activating possible parses according to a given preamble (Brehm et al., 2022). Crucially, PIPS represents the similarity between filler vectors (similarly between role vectors) by similarity scores, which is defined as the dot product between the two vectors. Higher similarity scores means greater co-activation between two roles (or fillers). Similarity score is the key component in PIPS that models the structural and lexical similarities among representations.

3 Simulation Procedure

We have adapted the PIPS model for simulating the structural priming results from Pickering and Branigan (1998) in the following way.

3.1 Training

We constructed three probabilistic context free grammars (PCFGs) that generate ditransitive sentences with both DO and PD structures using the 9 dative verbs studied by Pickering and Branigan (1998). To isolate the verb effect, we abstracted away the content of the noun phrases and only included three fillers for noun phrases: $\mathtt{NP}_s$ (subject), $\mathtt{NP}_i$ (indirect object), and $\mathtt{NP}_d$ (direct object). The three PCFGs can therefore each produce exactly 18 sentences: for each verb, either the DO structure ($\mathtt{NP}_s$ $\mathtt{VERB}$ $\mathtt{NP}_i$ $\mathtt{NP}_d$) or the PD structure ($\mathtt{NP}_s$ $\mathtt{VERB}$ $\mathtt{NP}_d$ $\mathtt{to}$ $\mathtt{NP}_i$). The frequency distribution over the 18 sentences is determined by counting the verb-specific occurrences of DO and PD structures in the British National Corpus (Yi et al., 2019), see Appendix A for reference.

Since the absolute frequency of *give* dominates those of the remaining 8 verbs, this caused other verbs to be underrepresented. For example, *give* has 23713 total occurrences while *loan* only has 23, so that *loan* almost vanished in the probabilistic distribution over the 18 sentences with absolute frequency. To mitigate such a dominance effect, we trained three separate models over the three PCFGs with different probabilities over the 18 structures, as they are proportional either to: (i) the absolute frequency of verbs in each structure; (ii) the normalized frequency, such that all verbs are equiprobable yet the relative probability of DO vs. PD for each verb is preserved; or (iii) all 18 sentences are equiprobable. We labeled the three models as **ABS**, **NORM**, and **BASE**, respectively.

As is mentioned in section 2, similarity scores are hyperparameters defined over pairs of fillers. We did a hyperparameter search over three types of similarities on the `VP` fillers: (i) two sentences share the same structure; (ii) two sentences share the same verb; (iii) two sentences share nothing. Each similarity score ranges from 0.2 to 0.7, on par with Brehm et al. (2022). We used the similarity scores (0.7, 0.2, 0.2) for the three types in training since this set of scores yielded the most human-like behavior in terms of structural priming.

3.2 Evaluating Priming Effects

An evaluation trial consists of a priming phase followed by a preamble completion phase.

We simulated priming by activating the relevant bindings (a symbol at a position in tree representations) of a prime sentence to the activation level (i.e. priming weights) at time step 0 (i.e. before the start of production). Such activations decay in the rate of 0.9, together with the preamble input, simulating the memory decay in humans, as is on par with Brehm et al. (2022). We experimented with three modes of priming: (i) only the nonterminal binding for the prime's VP node (which encodes both structural and verb information) (**structure**); (ii) all terminal bindings, corresponding to the words in the prime sentence but no structural information in higher layers of the tree (**words**); (iii) all bindings, both terminal (words) and nonterminal (structure) information of the prime sentence (**whole**). We varied activation values of the prime among 0.05, 0.1, and 0.2.

For the preamble completion phase, we gave the preamble "$\texttt{NP}_s$ `VERB`" to the model with each of the 9 verbs, having primed it with each of the 18 conditions (9 verbs and 2 structures for each verb) in each of the three priming modes with the three weights. Following previous work, we activated the preamble to 0.5. For each of the prime+preamble combinations, we ran the model with 50 production trials and recorded the production proportion of each of the 18 sentences. Productions that were not equal to any of the 18 sentences (i.e. ungrammatical productions) were recorded as *Others*. We consequently obtained the production distribution of each target verb primed with all 18 structure priming conditions, which we used to compute the relative priming strength of each structure+preamble combination. We also ran each model with no priming as the baseline for comparison.

4 Results and Discussion

4.1 Quantifying Priming Effects

To assess the strength of structural priming, we computed for each target verb (given in the preamble) the average deviation between the primed production and the baseline (i.e. production without priming), as shown in (1) and (2) [1].

$$\text{Ratio}_v(\text{DO}) = \frac{\#\text{DO}_v}{\#\text{DO}_v + \#\text{PD}_v} \tag{1}$$

$$\text{Dev}_v^{v'}(\text{DO}) = \text{Ratio}_v(\text{DO})_{\text{by}v'} - \text{Ratio}_v(\text{DO})_{\text{unprimed}} \tag{2}$$

First, we computed the proportion of sentences generated with the correct verb that contained the primed structure. We also did this distinguishing whether the prime sentence contained the target verb or not. We call the first quantity the Structure Priming score as shown in (3), which measures the structural priming effect: within the cases in which the model correctly produced the target verb, priming by a DO sentence should increase the proportion of DO over PD sentences, and vice versa. We call the second quantity the LBE score as shown in (4), which measures the lexical boost effect: when the verb is repeated in prime and target, we expect the model to produce even more sentences of the primed structure, compared to priming by another verb.

$$\text{StrucPriming} = \frac{\sum_{v\in\mathcal{V}} \sum_{s\in\{\text{DO, PD}\}} \text{Dev}_v(\text{s})}{\#\text{conditions}} \tag{3}$$

$$\text{LBE} = \frac{\sum_{v\in\mathcal{V}} \sum_{s\in\{\text{DO, PD}\}} [\text{Dev}_v^v(\text{s}) - \text{Dev}_v^{v'}(\text{s})]}{\#\text{conditions}} \tag{4}$$

Finally, we computed the difference between the deviation primed by the less preferred structure and the deviation primed by the more preferred structure. We call it the IFE score as shown in (5), which measures the inverse frequency effect: if a verb is biased towards PD, then we expect the model to produce more DO sentences when being primed by a sentence with this verb in a DO structure (i.e. the less preferred one) than such a priming boost of producing more PD sentences when primed by a sentence with this verb in a PD structure (i.e. the more preferred one). We only computed IFE score for NORM and ABS models, since there is by definition no structural bias in the BASE distribution.

$$\text{IFE} = \frac{\sum_{v\in\mathcal{V}} \sum_{v'\in\mathcal{V}_{\text{PD}}} [\text{Dev}_v^{v'}(\text{DO}) - \text{Dev}_v^{v'}(\text{PD})]}{\#\text{conditions}} \tag{5}$$

Since all three scores are computed as the deviation from baseline or from the counterpart, we interpret positive values as aligning with the human results.

4.2 Results

As shown in Fig. 1, in all priming settings, the Structural Priming scores and the LBE scores are positive, suggesting that PIPS can qualitatively reproduce human results in structural priming and lexical boost effects. Moreover, we found a strictly increasing relation of both quantities with respect to priming modes and weights. The more the priming

[1] For the sake of space, we don't lay out the formula for the PD counterpart

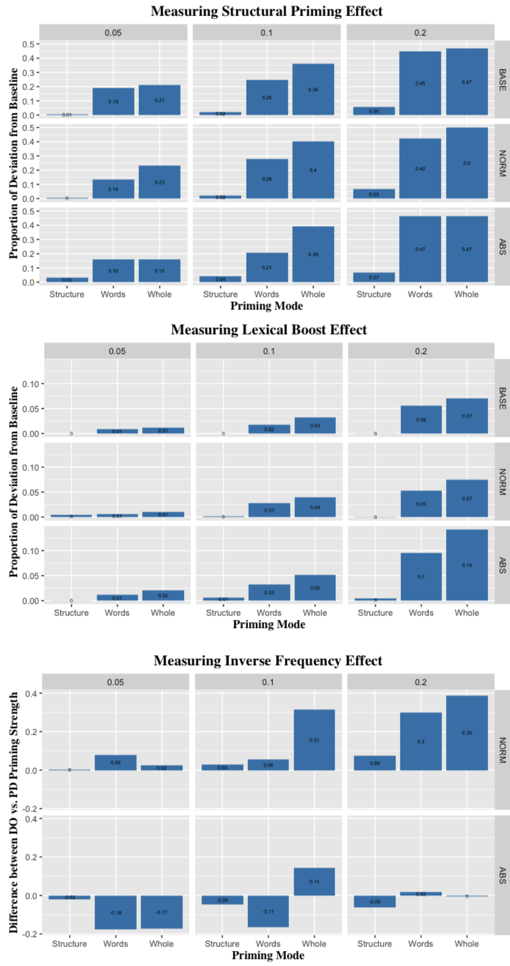

Figure 1: The Structural Priming score is plotted at the top, the LBE score is plotted in the middle, and the IFE score is plotted at the bottom.

bindings are activated, the stronger both effects are. Priming the model on solely structural information is less effective (i.e., yields weaker effects) than on all lexical information, which is less effective than on both types of information. The simulation results align with results in Pickering and Branigan (1998) and correctly reflect our predictions on priming modes and weights.

Turning to the IFE scores, we observe that the scores are only positive in the NORM model, while they are negative in most of the priming settings for the ABS model. Further in the NORM model, we only observe an increasing relation of the inverse frequency effect when the priming weight is 0.1 and 0.2, and we interpret the values with priming weights equals 0.5 as noise. However, no correlation is observed in the BASE and ABS models between priming settings and weights. Therefore, we conclude that the PIPS model with the current hyperparameter setting could only model the inverse frequency effect when the training probabilistic distribution of the 18 sentences are normalized.

4.3 Interpreting the Results

Our results show that the PIPS model is capable of modeling both the structural priming effect and the lexical boost effect. We haven't found a good way of aligning our simulation results with human results directly, since the human baseline production distribution of the 9 verbs isn't presented in Pickering and Branigan (1998).

It remains a question why only the NORM model captures the inverse frequency effect. As noted earlier, the inverse frequency effect has been attested in Jaeger and Snider (2007) and Bernolet and Hartsuiker (2010). Why might we find a difference in the ability of the different models to simulate this effect? One additional difference we note among our models is their ability to learn the data distribution reflected in the PCFGs used for training. We computed the Jensen-Shannon divergence between the probabilistic distribution specified in each PCFG and the production distribution of the two models in the unprimed baseline: $JS(\text{NORM}) = 1.2698, JS(\text{ABS}) = 2.5982$. Since the NORM model learns the target PCFG distribution better than the ABS model, this could be one factor explaining the NORM model's ability of modeling the inverse frequency effect. We leave the question of what parameters and target distributions the PIPS model is sensitive to for future investigation.

5 Conclusion

We have shown that the PIPS model is able to simulate the structural priming effect, the lexical boost effect, and the inverse frequency effect under some conditions. More broadly, we demonstrate the potential of GSC framework to simulate the process of human sentence production. The relation between the model and the two theories, though, is worth discussion. On the surface, it follows a transient activation approach, yet the fact that it could model the inverse frequency effect, as a prediction of implicit learning theory, is interesting. In future work, we will extend this priming simulation to the production of filler-gap dependencies (Momma, 2022) with PIPS.

References

Sarah Bernolet and Robert J. Hartsuiker. 2010. Does verb bias modulate syntactic priming? *Cognition*, 114(3):455–461.

J. Kathryn Bock. 1986. Syntactic persistence in language production. *Cognitive Psychology*, 18(3):355–387.

Kathryn Bock and Carol A Miller. 1991. Broken agreement. *Cognitive Psychology*, 23(1):45–93.

Laurel Brehm, Pyeong Whan Cho, Paul Smolensky, and Matthew A. Goldrick. 2022. PIPS: A Parallel Planning Model of Sentence Production. *Cognitive Science*, 46(2):e13079.

Franklin Chang, Gary S. Dell, and Kathryn Bock. 2006. Becoming syntactic. *Psychological Review*, 113:234–272.

Pyeong Whan Cho, Matthew Goldrick, Richard L. Lewis, and Paul Smolensky. 2018. Dynamic encoding of structural uncertainty in gradient symbols. In *Proceedings of the 8th Workshop on Cognitive Modeling and Computational Linguistics (CMCL 2018)*, pages 19–28, Salt Lake City, Utah. Association for Computational Linguistics.

Pyeong Whan Cho, Matthew Goldrick, and Paul Smolensky. 2020. Parallel parsing in a Gradient Symbolic Computation parser.

Gary S. Dell and Victor S. Ferreira. 2016. Thirty years of structural priming: An introduction to the special issue. *Journal of Memory and Language*, 91:1–4.

John Hale and Paul Smolensky. 2006. Harmonic grammars and harmonic parsers for formal languages. *Smolensky and Legendre*, pages 393–416.

T. Florian Jaeger and Neal Snider. 2007. Implicit Learning and Syntactic Persistence: Surprisal and Cumulativity. *University of Rochester Working Papers in the Language Sciences*, 3:26–44.

Shota Momma. 2022. Producing filler-gap dependencies: Structural priming evidence for two distinct combinatorial processes in production. *Journal of Memory and Language*, 126:104349.

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

Paul Smolensky, R. Thomas McCoy, Roland Fernandez, Matthew Goldrick, and Jianfeng Gao. 2022. Neurocompositional computing: From the Central Paradox of Cognition to a new generation of AI systems.

Eunkyung Yi, Jean-Pierre Koenig, and Douglas Roland. 2019. Semantic similarity to high-frequency verbs affects syntactic frame selection. *Cognitive Linguistics*, 30(3):601–628.

A List of Verbs and Structure Frequencies

Verb	DO Frequency	PD Frequency
give	15311	8402
show	502	571
send	658	3134
lend	177	677
hand	308	659
loan	12	11
offer	752	1203
sell	190	1288
post	1	55

Table 1: The DO vs. PD frequencies of the 9 verbs studied in Pickering and Branigan (1998).