Journal of Artificial Intelligence Research 61 (2018) 927-946 Submitted 11/17; published 4/18

Pre-Wiring and *Pre-Training*: What Does a Neural Network Need to Learn Truly General Identity Rules?

Raquel G. Alhama rgalhama@bcbl.eu
Willem Zuidema w.h.zuidema@uva.nl
Institute for Logic, Language and Computation
University of Amsterdam
Science Park 107
1098XG Amsterdam, The Netherlands

Abstract

In an influential paper ("Rule learning by seven-month-old infants"), Marcus, Vijayan, Rao and Vishton claimed that connectionist models cannot account for human success at learning tasks that involved generalization of abstract knowledge such as grammatical rules. This claim triggered a heated debate, centered mostly around variants of the Simple Recurrent Network model. In our work, we revisit this unresolved debate and analyze the underlying issues from a different perspective. We argue that, in order to simulate human-like learning of grammatical rules, a neural network model should not be used as a *tabula rasa*, but rather, the initial wiring of the neural connections and the experience acquired prior to the actual task should be incorporated into the model. We present two methods that aim to provide such initial state: a manipulation of the initial connections of the network in a cognitively plausible manner (concretely, by implementing a "delay-line" memory), and a pre-training algorithm that incrementally challenges the network with novel stimuli. We implement such techniques in an Echo State Network (ESN), and we show that only when combining both techniques the ESN is able to learn truly general identity rules. Finally, we discuss the relation between these cognitively motivated techniques and recent advances in Deep Learning.

1. Introduction

One of the crucial aspects of language is that it allows humans to produce and understand an unlimited number of utterances. This is possible because language is a rule-governed system; for instance, if we know that the English present participle is formed by appending *-ing*, then we readily *generalize* this pattern to novel verbs.

Accounting for how humans learn these abstract patterns, represent them and apply them to novel instances is a central challenge for cognitive science and linguistics. In natural languages there is an abundance of such phenomena, and as a result linguistics has been one of the main battlegrounds for debates between proponents of symbolic and connectionists accounts of cognition. One of the most heated debates concerned the regular and irregular forms of the English past tense. Rumelhart and McClelland (1986) proposed a connectionist model that allegedly accounted for the regular and irregular forms of the past tense. However, this model was fiercely critized by Steven Pinker and colleagues (Pinker & Prince, 1988; Pinker, 2015), who held that rules are essential to account for regular forms, while irregular forms are stored in the lexicon (the 'Words-and-Rules' theory).

A similar debate emerged with the publication of a study by Marcus, Vijayan, Rao, and Vishton (1999), this time centered on experimental results in Artificial Grammar Learning (AGL for short). The authors showed that 7 month old infants generalize to novel instances of simple ABA, ABB or AAB patterns after a short familiarization. Crucially, this outcome could not be reproduced by a Simple Recurrent Network (SRN, Elman, 1990), a result that was interpreted by the authors as evidence in favour of a symbol-manipulating system:

> Such networks can simulate knowledge of grammatical rules only by being trained on all items to which they apply; consequently, such mechanisms cannot account for how humans generalize rules to new items that do not overlap with the items that appear in training. (Marcus et al., 1999, p. 79)

This claim triggered many replies, some of which proposed variations of the original model. However, in this debate the issues of whether neural networks are capable of representing generalizing rules at all, and if and how these rules can be found using backpropagation are often conflated. The latter concerns, among other questions, what initial state of the network would allow for discovering these rules, and in our opinion, it has not received enough attention (but see Seidenberg & Elman, 1999a, and Altmann, 2002). This will be therefore the focus of this paper[1], in which we explore two directions. First, we ask which initial values of the connection weights could encourage generalization while remaining cognitively plausible (*pre-wiring*); second, we investigate the role of previous experience in creating an initial state in the network that would facilitate generalization (*pre-training*). We employ a prewiring and a pretraining technique in an Echo State Network (ESN, Jaeger, 2001), and show that only when combining both techniques the ESN is able to accurately generalize to novel items.

2. Background

In this section we first introduce the experimental study by Marcus et al. (1999), and then we briefly summarize the subsequent debate and the computational models that were proposed.

2.1 Empirical Data

Artificial Grammar Learning is a widely used experimental paradigm for studying the mechanisms underlying language acquisition and pattern recognition. Using manually designed artificial languages, researchers investigate under which conditions adults, infants or non-human animals may discover patterns in language-like input, and use it to identify words in a continuous speech stream (Saffran, Aslin, & Newport, 1996a; Saffran, Newport, & Aslin, 1996b; Aslin, Saffran, & Newport, 1998; Peña, Bonatti, Nespor, & Mehler, 2002), or to infer general rules that allow them to generalize these patterns to novel utterances (Marcus et al., 1999; Gomez & Gerken, 1999; Peña et al., 2002; Endress & Bonatti, 2007; Frost & Monaghan, 2016; Endress, Scholl, & Mehler, 2005; Endress, Dehaene-Lambertz, & Mehler, 2007; Marcus, Fernandes, & Johnson, 2007; Gerken, 2010). In this paper, we focus on the study presented by Marcus et al. (1999), since this study, whilst using very simple stimuli,

[1] This paper is a thoroughly revised and extended version of a previous publication by Alhama and Zuidema (2016).

was subject of a fierce debate on the nature of cognitive representations and the capabilities of neural network models.

Marcus and colleagues investigate the generalization abilities of 7 month old infants by conducting three AGL experiments. In their first experiment, the participants are familiarized to syllable triplets that follow a certain grammar: ABA for a randomly assigned group of infants, and ABB for the other. The stimuli contain 16 different triplets, each repeated 3 times. Those triplets are arranged in a 2-min. auditory speech stream, such that syllables are separated by a pause of 250 ms, and triplets of syllables are separated by 1s.

After the familiarization phase, the infants participate in a test phase in which they are exposed to a randomized set of triplets, both from the consistent and the inconsistent grammar. But, crucially, the test triplets are composed of syllables that did not appear in the familiarization triplets. The looking times of the infants are recorded in order to find out whether they can discriminate between the items of the two grammars.

In both groups of infants, the results show a statistically significant difference between mean looking times to consistent and inconsistent grammars. The authors conclude that infants can discriminate among ABA and ABB grammars.

Table 1: Stimuli used in experiment 2 by Marcus et al. (1999).

	Familiarization				Test
ABA	le di le le je le le li le le we le	wi di wi wi je wi wi li wi wi we wi	ji di ji ji je ji ji li ji ji we ji	de di de de je de de li de de we de	ba po ba ko ga ko
ABB	le di di le je je le li li le we we	wi di di wi je je wi li li wi we we	ji di di ji je je ji li li ji we we	de di di de je je de li li de we we	ba po po ko ga ga
3x triplet (random order)					

However, these stimuli contain a possible source of confound. The familiarization items of each grammar coincide with the test items of the same grammar in some phonetic features, despite the use of different syllables. This could have influenced the results, since infants may have learnt to discriminate the grammars based on those phonetic features, instead of learning the abstract rule. For this reason, this experiment was repeated with a more carefully controlled set of syllables, which we report in table 1. In this second experiment infants also exhibit significantly different responses for consistent and inconsistent grammar. Finally, an additional experiment was performed, in this case using AAB vs. ABB grammars, in order to determine whether the rule learnt before was simply the presence or absence of an immediate repetition. Infants also showed significantly different responses in this experiment.

In the light of these results, the authors concluded that: (i) 7 month-old infants can extract grammar-like rules, (ii) they can do it not based solely on statistical information (as would be evidenced from the additional controls in experiment 2), and (iii) the extracted rule is not merely the presence or absence of an immediate repetition.

2.2 Generalization and Neural Networks

Marcus (1998) argues that certain types of generalizations are unattainable for certain types of neural networks: concretely, those that lie *outside the training space*. The author defines *training space* as the combination of all feature values that network has witnessed during training. If there exist feature values that have never appeared during training, any item displaying that feature value lies outside the training space. For neural networks that are trained with the backpropagation algorithm, generalization to items outside the training space is, according to the author, extremely unlikely to occur due to what he calls *training independence*, which stands for the fact that the algorithm updates the weights of nodes independently of the activations of other nodes in the same layer.

Marcus et al. (1999) provide empirical evidence in support of this idea, by simulating the presented experiment in a Simple Recurrent Network (SRN, Elman, 1990), a neural network architecture that incorporates an additional context layer that maintains an exact copy of the hidden layer and presents it to the network in the subsequent timestep, providing the model with memory in this way. The SRN is trained to predict the next syllable in the familiarization stimuli, and then tested on its ability to predict the final syllable of test items consistent with the familiarization grammar. This model failed to produce the correct predictions, confirming the hypothesis of the researchers.

Some following publications proposed to change the encoding of the input (Christiansen & Curtin, 1999; Christiansen, Conway, & Curtin, 2000; Eimas, 1999; Dienes, Altmann, & Gao, 1999; Altmann & Dienes, 1999; McClelland & Plaut, 1999), the task (Seidenberg & Elman, 1999a, 1999b), the neural network architecture (Shultz, 1999; Sirois, Buckingham, & Shultz, 2000; Shultz & Bale, 2001), or – relevant to our work — to incorporate some form of pre-training (Seidenberg & Elman, 1999a; Altmann, 2002). Many of these models were subject of criticism by Marcus (1999a, 1999b, 1999c, 1999d), who argued that the models either involved some form of symbolic manipulation or did not adequately represent the experiment. About the model presented by Altmann (2002), which involves pre-training similar to the regime we explore in section 5, Marcus (1999e) points out, without giving any details, that, even if the model distinguishes grammatical from ungrammatical stimuli to some degree, it is unclear whether the model can actually learn the underlying general rule or discovers some heuristic that weakly correlates with it. In our work, we employ a neural network architecture that was not previously explored for this task (an Echo State Network, a type of Reservoir Computing network), and we report additional performance measures that tell us more about how general the learned rules are.

3. Simulations with a Simple Recurrent Network

Before presenting our simulations with the ESN model, we report our replication of the original simulations. We implement an SRN as described by Elman (1990), and we train it to predict the next syllable in the input. In line with the previous literature on computational models of this study, we focus on modeling the more carefully controlled experiment 2.

As in Marcus et al., we use distributional encoding of phonetic features (based on those presented in Plunkett & Marchman, 1993). But unlike the original simulations, we do not encode the pause between triplets as an additional symbol; instead, we do not update the weights in the network when it predicts the first syllable of the next triplet (we do this to make our baseline results maximally comparable with the simulations we present in the next sections).

In order to remain close to the test used in the experiments with infants, we test the network on both consistent and inconsistent sequences. We take the predicted vector for the third syllable of each triplet, and we find the closest vector that corresponds to one of the seen syllables (both from training and from test). We then evaluate whether the accuracy for consistent and inconsistent triplets is significantly different (for 16 runs of the model, equivalent to the number of infants in the experiment).

The test set used in the original experiments, as can be seen in Table 1, consists only of two triplet types of each grammar. For this reason, we also evaluate our model with an extended test set that contains 5 additional random novel syllables of each type (A and B), consisting therefore of 25 test triplets. Note that the proportion of significant results in either direction can (and will) be quite different for the small and large test sets (the probability of getting a significant result depends not only on the effect size but also on the sample size).

In order to explore the behaviour of the SRN model in this task, we try 160 different hyperparameter settings for each familiarization grammar, varying: the size of the hidden layer (from 84 to 200), the learning rate (from 0.01 to 0.4) and the number of epochs (from 100 to 200; greater numbers for epochs were detrimental to performance and thus we did not extend the search further) [2]. Although there could potentially exist more successful hyperparameter settings in an unexplored range of values, it is not our goal to exhaustively explore the space of configurations: a model that is fine-tuned to the particularities of a certain task and stimuli is hardly useful as a cognitive model, since it should eventually account for related tasks that humans can solve. We nevertheless consider that we explore a sufficiently large range of hyperparamer settings; after all, if it turned out that only a narrow range of configurations would succeed, the implication would be that the responses in the experiment may be difficult to observe. Instead, we aim for a form of evaluation that i) remains as close as possible to the one used in the experiments with infants, and ii) is useful to illustrate how likely it is for this architecture to produce results like those in the original experiment. Once we find a model that succeeds under this lenient metric, we opt for a stricter metric (see section 6).

We run 16 simulations for each configuration, equivalent to the number of participants in the experiment. As in the original experiment, we apply a Student's T test (with significance level 0.05) to find if responses are significantly different between grammars.

We report the proportion of runs (all unique combinations of parameters × 16 random seeds) that yield a significant difference in each direction on the test set. This gives some insight in how likely it is to obtain a significant result, as in the original experiment, under different hypotheses about the underlying cognitive processes[3].

[2] We found that the values of the three hyperparameters had a significant effect on the accuracy of the predicted syllables in the test: the accuracy is higher for smaller learning rate, smaller number of epochs and larger hidden layer dimension.

[3] The hypotheses we explore in this paper are the ones embodied by the computational models we define (SRN in section 4, ESN in sections 5 and 6), with individual variation in the parameters as specified in the main text. This way of evaluating is motivated by (i) the practical consideration that we had run this great number of simulations anyway to evaluate the effects of different parameters, and (ii) the need to avoid evaluating 'significant differences' with highly optimized parameter settings.

Figure 1 shows these proportion of simulations that yield a significant difference in the predicted syllables in the test. For the responses that are significantly different, we separate those for which the neural network responds better to the consistent grammar (in white) from those in which the inconsistent grammar is favoured (in grey).

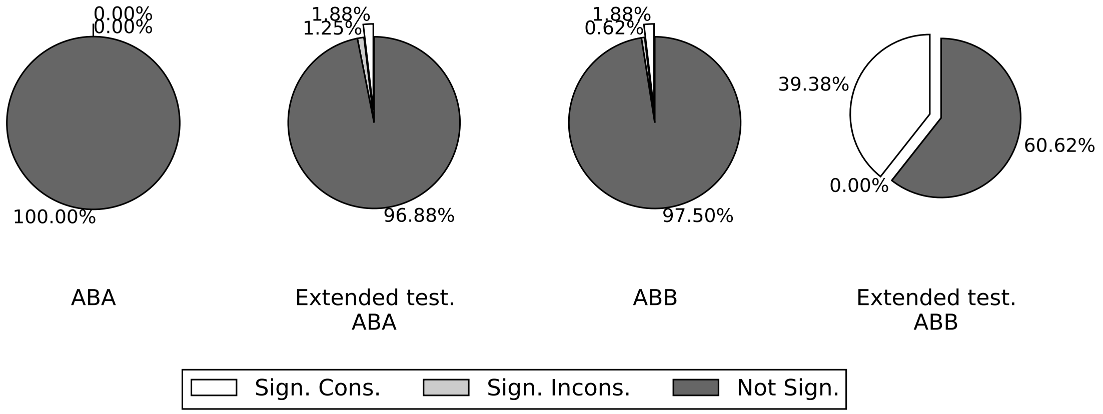


Figure 1: Proportion of parameter settings that yield significant (white), non-significant (dark grey) and inconsistently significant (light grey) responses, based on Student's T test (level of significance at 0.05) over simulations with an SRN.

As shown in the graphic, most of the simulations yield non-significant responses between grammars, in spite of a notable proportion of significant responses in the ABB condition for the extended test, possibly due to the fact that immediate repetitions are easier to learn[4]. We therefore confirm that the SRN does not reproduce the empirical findings.

4. Simulations with an Echo State Network

Recurrent Neural Networks, such as the SRN, can be seen as implementing memory: through gradual changes in synaptic connections, the network learns to exploit temporal regularities for the function it is trained on. An alternative way to learn time-dependent relations is that offered by Reservoir Computing (RC) approaches, such as the Liquid State Machine (Maass, Natschläger, & Markram, 2002) and the model adopted here, the Echo State Network (ESN, Jaeger, 2001; used previously in language learning in Frank & Čerňanský, 2008). In RC models, the weights in the hidden layer (which is dubbed "reservoir") remain untrained, but – if satisfying certain constraints (the so-called "Echo State Property", which depends on the scaling of the weights in the reservoir based on the spectral radius parameter) – the dynamics exhibited by the reservoir "echo" the input sequence: some memory of the input lingers on for some time in the recurrent connections. In other words, the state of the reservoir depends on the fading history of the input, and thus, after a long enough input, the initial state does not determine the final states of the network.

[4] This was also observed in the SRN simulations presented by Altmann (2002).

This property of RC models is interesting for two reasons: first, because it allows us to explore whether a model that remains very close to the SRN but implements a different type of memory would enhance generalization, and second, because the modifications that we apply in section 5 rely on a kind of 'inertia' of the weights across different training iterations and benefit from the fact that the weights in the reservoir are not updated during training.

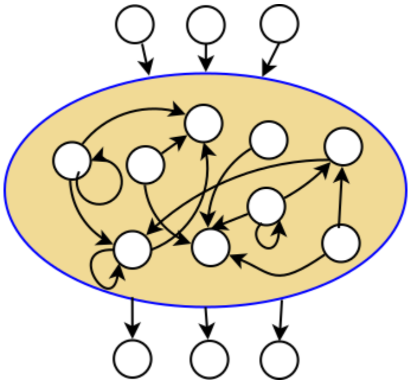


Figure 2: The Echo State Network.

The formalization of the ESN model is as follows. For an input u at time t, the activation x of the nodes in the reservoir is defined as:

$$x(t) = f^{res}(W^{in} \cdot u(t) + W^{res} \cdot x(t-1)) \tag{1}$$

where W^{in} are the input weights, W^{res} are the internal weights of the reservoir, and f is a non-linear function, generally *tanh*.

The activation of the output is defined as:

$$y(t) = f^{out}(W^{out} \cdot x(t)) \tag{2}$$

where W^{out} are the weights that connect the reservoir with the output nodes, and f^{out} is an activation function, which might be different from the function applied inside the reservoir; in fact, it often consists of a simple identity function.

We implement a basic ESN with binary neurons and the *tanh* activation function, and we follow the same procedure described in section 3 to train the network, though we use backpropagation[5]. We try 200 parameter settings for each familiarization grammar, varying the hyperparameters of the model: the number of nodes in the reservoir, the input scaling, the spectral radius, learning rate and epochs.[6] Figure 3 shows the proportion of these runs that yield a significant difference in the predictions.

As can be seen, the results based on the Marcus et al. test set differ greatly from those in our extended test. This confirms our intuition that the amount of test items is crucial for the evaluation. For this reason, we base our analysis of the behaviour of the model on the

[5] We have also run simulations with Ridge Regression. Backpropagation performs no worse for the basic ESN, and allows for the extensions that are presented in the next sections.

[6] We found that the values of the input scaling and the learning rate had a significant effect on the accuracy of the predicted syllables in the test.

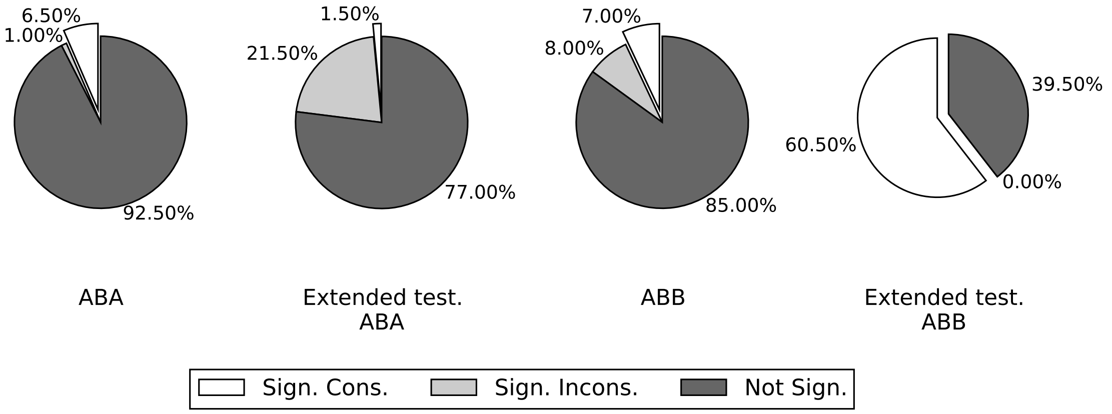


Figure 3: Proportion of parameter settings that yield significant (white), non-significant (dark grey) and inconsistently significant (light grey) responses, for the simulation with the basic ESN.

extended test; however, it should be noted that the amount of test items could have also played a role in the actual experiments with infants (as we discuss later in section 7).

The plots of the extended test condition clearly show an asymmetry between the grammars: more than half of the parameter settings yield significant responses for the ABB, while in the case of ABA, less than a quarter of the simulations are significant, and most of them are actually favouring the inconsistent grammar, which is precisely ABB. As mentioned before, the reason for this asymmetry is probably due to the fact that immediate repetitions are easier to learn, since they are less affected by the decay of the activation function; for now, it suffices to say that the behaviour of the model towards ABA does not suggest that it could be a potential explanation for the experimental results.

5. *Pre-Wiring*: Delay Line Memory

In order to succeed in the prediction task, the model must predict a syllable that is identical to one presented before. In the previous simulations, we relied on the memory that ESNs offer through the recurrent connections and the Echo Property (Jaeger, 2002a). However, other computational principles have been suggested to implement memory in the human brain (Chaudhuri & Fiete, 2016). We now explore one such model of memory: a delay line.

Computationally, a delay line is a mechanism that ensures the preservation of the input by propagating it in a path ("line") that implements a delay. In the brain, delay lines have been proposed as part of the sound localization system (Jeffress, 1948), and they have been identified through intracellular recordings in the barn owl brain (Carr & Konishi, 1988). In a neural network, a delay line can be naturally implemented by organizing a subnetwork of neurons in layers, with "copy connections" (with weights 1 between corresponding nodes and 0 everywhere else) connecting each layer to the next. In this way, the information is kept in the network for as many timesteps as there are copy layers in the network (see Figure 4).

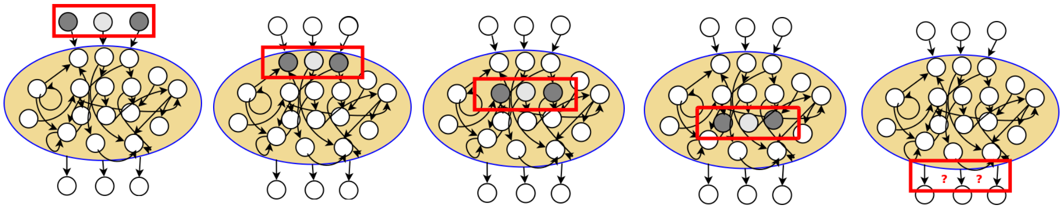


Figure 4: Depiction of five timesteps in the DeLi-ESN. The highlighted nodes show the activations in the delay line (the activation of the rest of the nodes is not illustrated).

We implement a delay line memory in the ESN (creating thus a new model that we call *DeLi-ESN*) by imposing this layer structure in the reservoir. We run 600 combinations of hyperparameters. This number is three times bigger than in the case of the basic ESN because we add a hyperparameter that regulates the addition of some amount of noise to the weights of the reservoir (to experiment with imperfect copies in the delay line). In this way, some models contain a less strict delay line; the greater the noise, the closer the model is to the original ESN.

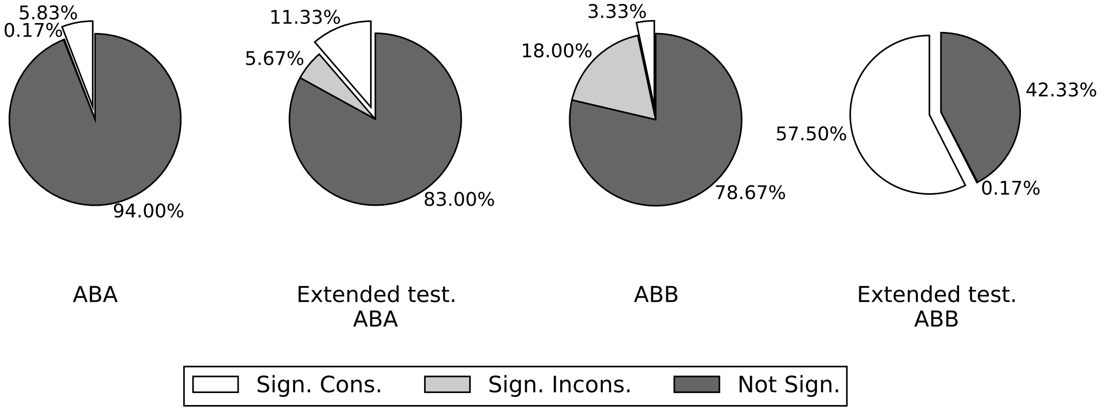


Figure 5: Proportion of parameter settings that yield significant, non-significant and inconsistently significant responses in the tests, for the simulation with DeLi-ESN.

The results, illustrated in Figure 5, show an increased number of significant responses (in favour of the consistent grammar): especially for the extended test of ABA familiarization, which was the least favoured in the previous simulations. Therefore, the addition of the delay line memory indeed helps in the detection of the identity relation. Thus, our study shows the potential of enhancing a neural network model with this explicit form of memory: if the input is available to the model for a few timesteps in the form of activations in the reservoir, the model has more chances to converge to a generalizing solution.

Furthermore, our finding that adding a delay line helps highlighting one of the mechanisms that may be at play in the experiments with infants. A delay-line is a simple computational principle of memory that does not convey unnecessary assumptions about infants' cognition, so it is plausible that this form of memory helped infants in the experiment. The incorporation of neural *pre-wiring* illustrates the need to account for the initial state of the model, and in this case it sufficed to add a very simple structure to the model. Additionally, this finding contributes to the more fundamental question of what a neural network model needs in order to generalize: since the identity rules in the stimuli relate syllables in the input, the network may learn to predict the next syllable based on repeating a previously encountered syllable.

However, although it is nice that the DeLi-ESN reproduces the statistical differences observed in the original experiment, a crucial question remains: does this model *accurately* generalize? In other words, does the model predict the correct syllables when tested with the consistent grammar?

In the case of the study with infants, the data does not allow for investigating whether infants would really predict the next syllable in the consistent test items. But we can perform this analogous study with neural networks by looking into the proportion of correct answers, that is, how often the nearest neighbour of the predicted vector corresponds to the correct syllable. For the DeLi-ESN model, we find that exact prediction accuracy remains low in the DeLi-ESN (rarely above 12% for ABA and above 20% for ABB familiarization), even for the best combination of parameters. Therefore, the DeLi-ESN, despite exhibiting significant differences between grammars, does not produce accurate predictions for either grammar, and thus this technique alone does not suffice for generalizing correctly.

6. *Pre-Training*: Incremental-Novelty Exposure

Although the evaluation metric used until now was useful to remain as close as possible to the original experiments, it does not allow us to determine whether the predictions are accurate. For this reason, we now continue our investigation adopting a stricter measure of success. Since this is a more strict criterion than we used before, we do not repeat the analysis based on statistical significance, which have the additional complication of becoming computationally very expensive when combined with the pre-training procedure.

In the previous section we manipulated the initial state of the model to explore how the biases that infants bring to the lab may help them to generalize to novel items. Likewise, the infants that participated in the original experiment had surely been exposed to human speech before the actual experiment; however, in most computational studies this fact is ignored. We hypothesize that prior experience could have triggered a bias for learning abstract solutions: unlike the familiarization in the experiment, which repeats the same stimuli over time, the input data that infants receive from their environment is likely to contain both familiar and novel items. Thus, infants may have adapted their induction mechanism to account for this type of data. We now propose a method to *pre-train* a neural network to test this hypothesis.

The training regime we propose —which we call Incremental-Novelty Exposure, or INE for short— consists of a chain of complete training and testing steps, which we call 'iterations'. As illustrated in Figure 6, the chain is arranged such that the final parameters of the previous model become the initial state of the model in the next iteration. The test remains constant in each of these iterations; however, the training data is slightly modified from one iteration to the next. The initial training set is designed according to the Marcus et al. stimuli: 4 syllables of type A and 4 syllables of type B are combined into triplets, according to the familiarization grammar (ABA or ABB). In the second iteration, one syllable of type A and one of type B are deleted, and a new syllable of type A and one of type B are incorporated. This set of syllables is again arranged according to the familiarization grammar. In this way, the triplets in each iteration are mostly the same as in the previous iteration, with the exception of a few that incorporate novel syllables.

This method implements the idea of exposing a learner to a larger dataset, but crucially, the training data incorporates novelty in small incremental doses. This is notably different from augmenting the dataset: as argued by Marcus (1998), simply augmenting the training data does not help generalization, since the model still settles for a solution that fits the training set without accounting for the unseen data in the test set. With our procedure, we aim to encourage the model to settle for a generalizing solution by gradually incorporating novelty. In other words, our goal is to 'surprise' the model with new data in each iteration, so that it is pushed towards a solution that fits also the novel input.

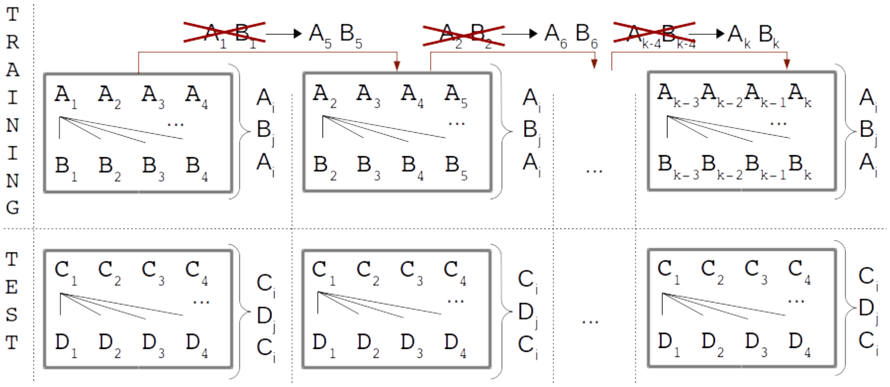


Figure 6: Depiction of the Incremental-Novelty Exposure. Barred syllables are removed from the training set after the training of the corresponding iteration has finished (so they do not remain in the training set of the next iteration).

As before, we simulate 600 different hyperparameter configurations, varying the reservoir size, noise in the delay line, input scaling, spectral radius, learning rate, and number of epochs of each iteration. To show the general trend of the model, we first plot a single representative run[7]. In this way, we avoid the smoothing effect of averaging between runs that start from different initial states – and thus learn at different pace. Figure 7 shows how the accuracy (the proportion of correct predictions over test items) evolves at each stage of the INE procedure. As can be seen in the graphs, the accuracy is really low in the beginning (corresponding to a simulation without *pre-training*) but, with more iterations the model greatly improves, presumably by finding a more general solution.

Now that we have seen how learning takes place on one representative run, we show that these results are robust. Figure 8 shows the mean accuracy across the last quarter of the

[7] The parameters of this run are: reservoir size 200, spectral radius 0.9, reservoir noise scaling 0.00001, learning rate 0.4, number of epochs 100.

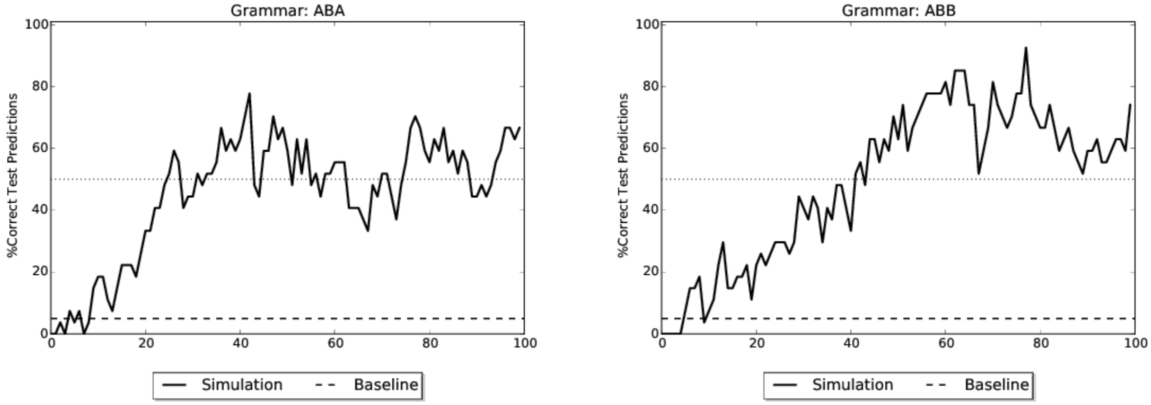


Figure 7: Accuracy (proportion of correct predictions over test items), over 100 iterations with Incremental Novelty Exposure, for the consistent grammar. The ABA familiarization condition is plotted on the left, and the ABB condition on the right. The dashed line indicates the probability of arriving at the correct solution by chance. The dotted line indicates 50% accuracy.

iterations (corresponding to the rightmost part of the plot in figure 7, amounting to a total of 25 iterations), for the consistent test items, again for 16 runs.

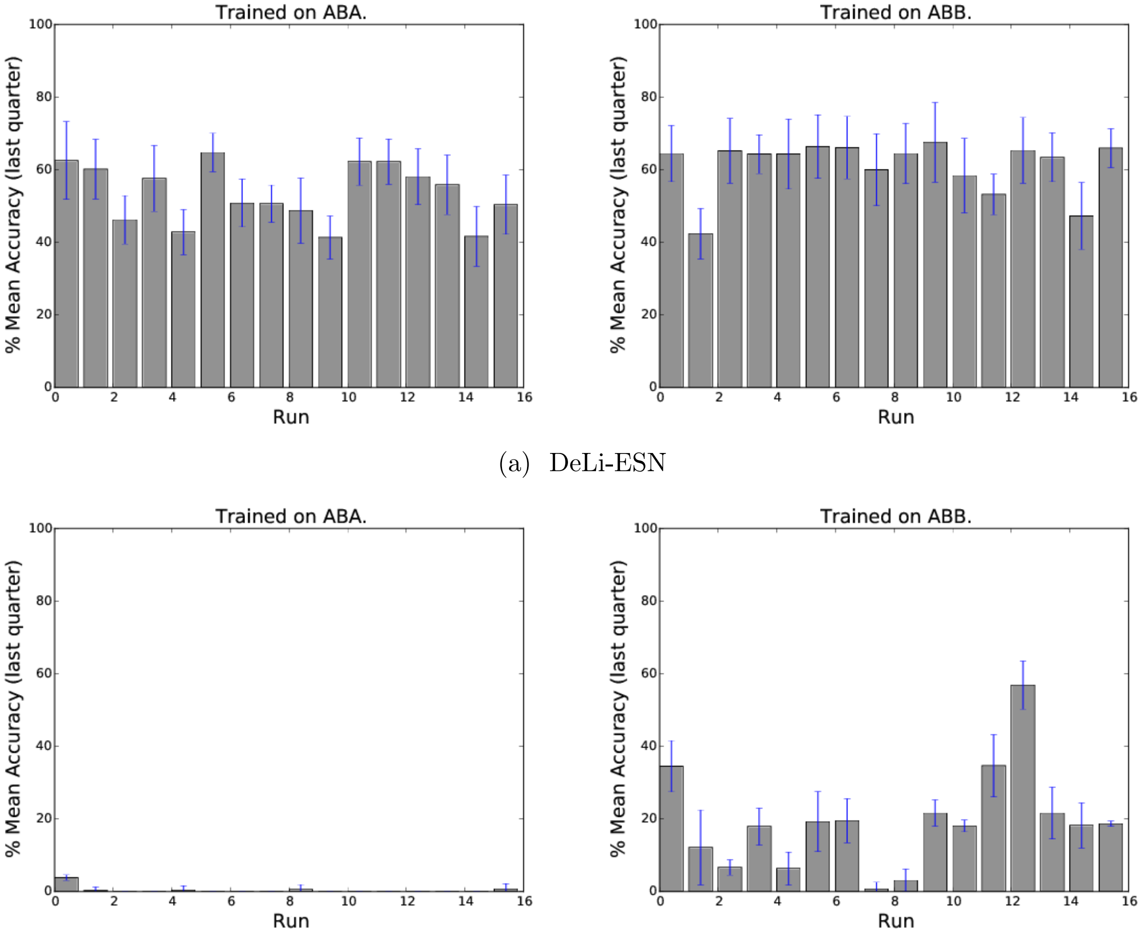
Figure 8a shows that the combination of the DeLi-ESN and the INE drastically boosts the generalization capabilities of the ESN. However, we should identify what is the contribution of the DeLi-ESN. This is reflected in Figure 8b, which shows the mean accuracy (again, for the last quarter of the iterations), for 16 runs of the basic ESN in the INE regime. The effect of the delay line memory is clear: when removed, the accuracy is close to 0 for ABA, and mostly around 20% for ABB.

7. Discussion I: Conclusions from the Study

The Marcus et al. publication conveyed two bold statements: first, that infants spontaneously extract identity rules from a short familiarization, and second, that neural networks are doomed to fail in such simple task. Our study suggests that both the initial optimism about the generalization abilities of infants and the pessimism about neural networks were overstated.

Our work investigates the initial conditions that a neural network model should implement in order to account for the experimental results. First, we have proposed that networks should be *pre-wired* to incorporate a bias towards memorization of the input, which we have implemented as a delay line. With such *pre-wiring*, the model yields a notable proportion of significantly different responses between grammars. But despite this apparent success, the accuracy of the model in syllable prediction is very low.

Therefore, even though successful discrimination between grammars is generally understood as abstraction of the underlying rule, our results show that significantly different responses can be obtained from a model that has not perfectly learnt such a rule. The



(a) DeLi-ESN

(b) Basic ESN

Figure 8: Mean accuracy of 16 simulations, computed over last quarter of a total of 100 iterations of the INE simulation. The DeLi-ESN with INE shows high level of accuracy in both the ABA and ABB conditions (a), much higher than the DeLi-ESN without INE (section 4) and that the DeLi-ESN with INE (b).

corollary is that the generalization abilities of infants may have been overestimated; as a matter of fact, null results in similar experiments also point in that direction (see for instance footnote 1 in Gerken, 2006; also Geambasu & Levelt, personal communication, October 14, 2016).

But can neural networks go beyond grammar discrimination and accurately predict the next syllable according to a generalized rule? Our work shows that this can be achieved when prior experience is incorporated in the model. We have hypothesized that, from all the information available in the environment, it is the gradual exposure to novel items that enhances generalization. This particular hypothesis deviates from related studies in which (i) an SRN was pre-trained to learn the relation of *sameness* between syllables (Seidenberg & Elman, 1999a), and (ii) an SRN was pre-trained with a set of sentences generated from a uniformly sampled vocabulary. Although the data used in our *pre-training* is less realistic than that used by Altmann (2002), our evaluation method is more strict (since we aim to test for accuracy rather than discrimination); for this reason, we have here evaluated our model with a more constrained input, and leave to future work the investigation of whether the same results can be obtained when input data involving gradual novelty is generated from a grammar unrelated to the actual experiment.

Finally, our results also contribute to the symbols vs. associations debate, by showing that a model without symbolic structures can reproduce the results. The delay line may initially appear as providing the model with variables, since the dedicated groups of nodes that encode the input at a certain time may be seen as registers. However, it should be noticed is that these groups of nodes are not isolated, and therefore, the learning algorithm needs to discover this structure in order to make use of it. It is uncontroversial that items are kept in memory for a certain lapse of time, so we do not think that this structure is constitutes the core of the symbolic enterprise. If despite this crucial differences our model is taken to be compatible with the theory of rules-over-variables, our contribution would then result in a unifying model in which both sides in the debate can see their proposals reflected.

8. Discussion II: Relation with Deep Learning Techniques

The study we have presented shows that a very simple task like the one devised by Marcus and colleagues, which appears to be solved by infants at the young age of 7 months, is not achievable by out-of-the box neural network models. Understanding what are the necessary elements for a neural network model to generalize is not only relevant for research in infant cognition, but is also a fundamental question for the field of Artificial Intelligence. Efforts in Deep Learning have provided several techniques to improve the performance of neural network models. We now discuss the relation between our approach and some of the most popular methods in Deep Learning.

8.1 Relation with Recurrent Neural Networks

Our model differs from the SRN model, and from Recurrent Neural Network (RNN) more generally, in its adoption of the *reservoir computing* approach. Concretely this means the weights from the input layer to the hidden layer, and the recurrent weights from the hidden layer to itself, are never updated. Thus, learning only occurs in the output layer.

This has important consequences for how each model implements a memory. In RNNs, input from previous time steps may continue to play a role through the recurrent connections. Since those connections are updated during training, the model must *learn* what to memorize and how to combine it with novel input. In contrast, in reservoir computing models, the reservoir is designed, by enforcing the echo property, to have information from past inputs (and initialization) linger around in the recurrent connections (and this information is asymptotically 'washed out', Jaeger, 2002b). Hence, the architecture does not need to learn to memorize information, but rather may focus on how to use both the memorized and current input to solve a task.

There is an interesting connection between the echo state property in reservoir computing and the now popular 'orthogonal initialization' for recurrent networks. In echo state networks, the echo state property is often achieved by ensuring, as we do in this paper, that the spectral radius (largest eigen value) of the recurrent weight matrix has a value just below 1.0 (although this is neither a sufficient nor a necessary condition for the echo state property to hold; see Jaeger, 2002b). In 'orthogonal initialization' all eigenvalues of that matrix have an absolute value of 1; the largest eigenvalues will thus initially be 1.0, before backpropagation gradually changes the weights and eigen values.

8.2 Relation with Standard Echo State Network

There are two main changes in our model that deviate from the standard ESN. First of all, by *pre-wiring* the reservoir, we depart from the original implementation, in which the reservoir is randomly initialized (although constrained to satisfy the echo state property). This has an effect on the type of memory the model has access to. In standard ESN's, the echo state property ensures that the information of previous input lingers around, but not that this information remains represented in the same way: any (sub)vector representing the information of interest at time t might be inverted, rotated or projected into another subspace at any time t. Instead, the delay line in our model keeps information in the same representation (and projected to a dedicated subspace for each time delay).

Another aspect that deviates from common implementations of ESN is that we train through gradient descent, while generally, ESNs are trained with Ridge Linear Regression. The latter is a simple and fast solution, works over a convex error function, and converges to the global optimum. However, for our 'Incremental Novelty Exposure' in section 6, the global optimum for the current training set is not what we want. Key to the success of that approach is that previous training phases still have an effect on the current. With our use of stochastic gradient descent, our model explores multiple local optima, and thus has a chance to find a solution that generalizes beyond the current datasets.

8.3 *Pre-Wiring*: Relation with LSTMs

The incorporation of a Delay Line extends the memory of the model in a particular way, by forcing the persistence of an explicit representation of the input for a certain time. This technique relates to the Long Short Term Memory (LSTM) networks (Hochreiter & Schmidhuber, 1997), a variant of RNNs that also augments recurrent networks with a 'memory cell'.

In LSTMs, the hidden layer is extended to incorporate a circuitry of *gates* that control the flow of information. The core of such circuitry is the *memory cell*, the component that maintains a persistent copy of the input over time. The gates regulate adding or removing information from the cell state. Thus, the gates are designed to learn which information should be forgotten or retrieved, while the state cell acts like a conveyor belt on which the input is propagated over time.

Both the delay line and the gated memory in LSTMs are incorporated to the networks to ensure persistence of the input by copying it over timesteps. However, the techniques also exhibit notable differences in how the information is propagated: while in LSTMs, the information in the cell state is only modified with linear operations, the delay line incorporates a non-linear activation function (*tanh*) that has the effect of applying a form of decay over the input, such that the input values are closer to zero after every timestep.

These models also exhibit a different approach to forgetting. As mentioned, our model incorporates some form of forgetting by decaying the input over time. This decay is applied uniformly to all the units participating in the representation of a certain input vector. In contrast, in LSTMs the information does not uniformly and constantly deteriorate over time. Instead, the gate circuitry incorporates a *forget* gate that is trained to decide which information of the cell state should decay or be removed. The gate operates over units rather than over vectors, so the features of the input are not uniformly surpressed.

8.4 *Pre-Training*: Relation with Dropout

The Incremental Novelty Exposure training regime of section 6, also finds a counterpart in the current deep learning literature, in the regularization technique of Dropout (Hinton, Srivastava, Krizhevsky, Sutskever, & Salakhutdinov, 2012; Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014). The idea behind Dropout is that, by randomly removing some of the units of the network during training, the model is forced to find solutions that do not rely too much on concrete correlations. This intuition is similar to the one that guides the INE: by constantly adding some novelty to the data, we force the network to not rely too much on accidental correlations. The crucial difference between both techniques is that Dropout modifies the architecture of the network to achieve this purpose, while the INE achieves a similar effect by manipulating the training data.

8.5 *Pre-Training*: Relation with Data Augmentation and Curriculum Learning

A well-known technique in Deep Learning is to enhance the performance of a neural network by adding more datapoints to the training set, a technique known as Data Augmentation. Pretraining can be seen as an instance of data augmentation, but it is actually more constrained: the added datapoints always precede the training data, and they are arranged in a particular way, according to the INE procedure. As explained before, it is precisely the arrangement of the data that helps generalization.

The INE also has some similarities with Curriculum Learning (CL, Bengio, Louradour, Collobert, & Weston, 2009). The intuition behind CL is to enhance learning by pre-training with an ordered sequence of data that goes from simpler to more complex datapoints, based on some function of complexity. Implicitly, this creates smoother objectives at the beginning of the pre-training, which may push the model to regions in parameter space that favour generalization; once in this local optima, the objective can be refined to find a better solution.This is the same intuition that we followed when designing the INE, though the particular strategy we use for *pre-training* is different. Although we also order the datapoints, we do not apply a function of complexity but rather we ‘dose’ the amount of novelty, and we do not change it over time. Thus, the parameter space is not necessarily smoother, but by introducing novelty at a constant rate the model may still be guided to regions of generalizing solutions.

9. Conclusion

This study aimed to explore what a neural network model needs in order to generalize to novel items. We have taken the perspective of investigating the initial state: what is it that infants bring to the lab that allows them to learn more quickly, and how can that be implemented in a neural network model? Our results show that we can approximate this initial state with two techniques: *pre-wiring* the neural network with connections that implement a form of memory, and *pre-training* the model in a way that ensures that the model is presented with novel input in a gradual fashion.

For the field of Artificial Intelligence, a relevant conclusion is that powerful models such as neural network models are not capable of simple generalization if used as *tabula-rasa*. With our work, we have identified some preconditions that are needed for generalization of identity rules, and we have argued that some of the existing techniques in Deep Learning actually implement related intuitions. Thus, we hope that providing a better understanding of the generalization mechanism in humans and neural networks can also inspire future developments in Deep Learning.

Acknowledgments

Both authors were funded by a grant from the Netherlands Organisation for Scientific Research (NWO), Division of Humanities, to Levelt, ten Cate and Zuidema (360-70-450). W.Z. was also funded by the Gravitation Program 'Language in Interaction' of the same organisation (Gravitation Grant 024.001.006). We are grateful to Dieuwke Hupkes and Michael Repplinger for their comments on a previous version of this paper.

References

Alhama, R. G., & Zuidema, W. (2016). Pre-wiring and pre-training: What does a neural network need to learn truly general identity rules?. In Besold, T. R., Bordes, A., d'Avila Garcez, A., & Wayne, G. (Eds.), *Proceedings of the Workshop on Cognitive Computation: Integrating Neural and Symbolic Approaches (CoCo at NIPS2016)*, Vol. 1773, pp. 26–35.

Altmann, G. T. (2002). Learning and development in neural networks–the importance of prior experience. *Cognition*, *85*(2), B43–B50.

Altmann, G. T., & Dienes, Z. (1999). Technical comment on rule learning by seven-month-old infants and neural networks. *Science*, *284*(5416), 875–875.

Aslin, R. N., Saffran, J. R., & Newport, E. L. (1998). Computation of conditional probability statistics by 8-month-old infants. *Psychological science*, *9*(4), 321–324.

Bengio, Y., Louradour, J., Collobert, R., & Weston, J. (2009). Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pp. 41–48. ACM.

Carr, C. E., & Konishi, M. (1988). Axonal delay lines for time measurement in the owl's brainstem. *Proceedings of the National Academy of Sciences*, *85*(21), 8311–8315.

Chaudhuri, R., & Fiete, I. (2016). Computational principles of memory. *Nature neuroscience*, *19*(3), 394–403.

Christiansen, M. H., Conway, C., & Curtin, S. (2000). A connectionist single mechanism account of rule-like behavior in infancy. In *Proceedings of the 22nd annual conference of the cognitive science society*, pp. 83–88.

Christiansen, M. H., & Curtin, S. (1999). Transfer of learning: rule acquisition or statistical learning?. *Trends in Cognitive Sciences*, *3*(8), 289 – 290.

Dienes, Z., Altmann, G., & Gao, S.-J. (1999). Mapping across domains without feedback: A neural network model of transfer of implicit knowledge. *Cognitive Science*, *23*(1), 53–82.

Eimas, P. (1999). Do infants learn grammar with algebra or statistics?. *Science*, *284*(5413), 435.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*(2), 179–211.

Endress, A., & Bonatti, L. (2007). Rapid learning of syllable classes from a perceptually continuous speech stream. *Cognition*, *105*(2), 247–299.

Endress, A. D., Dehaene-Lambertz, G., & Mehler, J. (2007). Perceptual constraints and the learnability of simple grammars. *Cognition*, *105*(3), 577–614.

Endress, A. D., Scholl, B. J., & Mehler, J. (2005). The role of salience in the extraction of algebraic rules. *Journal of Experimental Psychology: General*, *134*(3), 406.

Frank, S. L., & Čerňanský, M. (2008). Generalization and systematicity in echo state networks. In B.C. Love, K. M., & Sloutsky, V. (Eds.), *Proceedings of the 30th Annual Conference of the Cognitive Science Society*, pp. 733–738. Cognitive Science Society.

Frost, R., & Monaghan, P. (2016). Simultaneous segmentation and generalisation of non-adjacent dependencies from continuous speech. *Cognition*, *147*, 70–74.

Gerken, L. (2006). Decisions, decisions: infant language learning when multiple generalizations are possible. *Cognition*, *98*(3), B67 – B74.

Gerken, L. (2010). Infants use rational decision criteria for choosing among models of their input. *Cognition*, *115*(2), 362–366.

Gomez, R. L., & Gerken, L. (1999). Artificial grammar learning by 1-year-olds leads to specific and abstract knowledge. *Cognition*, *70*(2), 109–135.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint*, *1207.0580.*

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735–1780.

Jaeger, H. (2001). The "echo state" approach to analysing and training recurrent neural networks.. Tech. rep., German National Research Center for Information Technology.

Jaeger, H. (2002a). Short term memory in echo state networks. Tech. rep., German National Research Center for Information Technology.

Jaeger, H. (2002b). Tutorial on training recurrent neural networks, covering bppt, rtrl, ekf and the echo state network approach. Gdm report 159, Fraunhofer Institute AIS.

Jeffress, L. A. (1948). A place theory of sound localization.. *Journal of comparative and physiological psychology*, *41*(1), 35.

Maass, W., Natschläger, T., & Markram, H. (2002). Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural computation*, *14*(11), 2531–2560.

Marcus, G. F., Vijayan, S., Rao, S., & Vishton, P. (1999). Rule learning by seven-month-old infants. *Science*, *283*(5398), 77–80.

Marcus, G. F. (1998). Rethinking eliminative connectionism. *Cognitive psychology*, *37*(3), 243–282.

Marcus, G. F. (1999a). Connectionism: with or without rules?: Response to J.L. McClelland and D.C. Plaut (1999). *Trends in Cognitive Sciences*, *3*(5), 168 – 170.

Marcus, G. F. (1999b). Do infants learn grammar with algebra or statistics? Response to Seidenberg and Elman, Negishi and Eimas.. *Science*, *284*, 436–37.

Marcus, G. F. (1999c). Reply to Christiansen and Curtin. *Trends in Cognitive Sciences*, *3*(8), 290 – 291.

Marcus, G. F. (1999d). Reply to Seidenberg and Elman. *Trends in Cognitive Sciences*, *3*(8), 288.

Marcus, G. F. (1999e). Response to Technical Comment on Rule learning by seven-month-old infants and neural networks by Gerry T.M. Altmann and Zoltán Dienes. *Science*, *284*(5416), 875–876.

Marcus, G. F., Fernandes, K. J., & Johnson, S. P. (2007). Infant rule learning facilitated by speech. *Psychological Science*, *18*(5), 387–391.

McClelland, J. L., & Plaut, D. C. (1999). Does generalization in infant learning implicate abstract algebra-like rules?. *Trends in Cognitive Sciences*, *3*(5), 166–168.

Peña, M., Bonatti, L., Nespor, M., & Mehler, J. (2002). Signal-driven computations in speech processing. *Science*, *298*(5593), 604–607.

Pinker, S. (2015). *Words and rules: The ingredients of language*. Basic Books.

Pinker, S., & Prince, A. (1988). On language and connectionism: Analysis of a parallel distributed processing model of language acquisition. *Cognition*, *28*(1-2), 73–193.

Plunkett, K., & Marchman, V. (1993). From rote learning to system building: Acquiring verb morphology in children and connectionist nets. *Cognition*, *48*(1), 21–69.

Rumelhart, D., & McClelland, J. (1986). On learning past tenses of English verbs. In Rumelhart, D., & McClelland, J. (Eds.), *Parallel Distributed Processing, Vol. 2*, pp. 318–362. MIT Press, Cambridge, MA.

Saffran, J. R., Aslin, R. N., & Newport, E. L. (1996a). Statistical learning by 8-month-old infants. *Science*, *274*(5294), 1926–1928.

Saffran, J. R., Newport, E. L., & Aslin, R. N. (1996b). Word segmentation: the role of distributional cues. *Journal of Memory and Language*, *35*(4), 606–621.

Seidenberg, M. S., & Elman, J. L. (1999a). Do infants learn grammar with algebra or statistics?. *Science*, *284*(5413), 433.

Seidenberg, M. S., & Elman, J. L. (1999b). Networks are not ‘hidden rules’. *Trends in Cognitive Sciences*, *3*(8), 288–289.

Shultz, T. R. (1999). Rule learning by habituation can be simulated in neural networks. In *Proceedings of the twenty first annual conference of the Cognitive Science Society*, pp. 665–670.

Shultz, T. R., & Bale, A. C. (2001). Neural network simulation of infant familiarization to artificial sentences: Rule-like behavior without explicit rules and variables. *Infancy*, *2*(4), 501–536.

Sirois, S., Buckingham, D., & Shultz, T. R. (2000). Artificial grammer learning by infants: an auto-associator perspective. *Developmental Science*, *3*(4), 442–456.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*(1), 1929–1958.