Supplementary material for: *Which Input Abstraction is Better for a Robot Syntax Acquisition Model? Phonemes, Words or Grammatical Constructions?*

Xavier Hinaut
Inria Bordeaux Sud-Ouest, Talence, France.
LaBRI, UMR 5800, CNRS, Bordeaux INP, Université de Bordeaux, France.
Institut des Maladies Neurodégénératives, UMR 5293, CNRS, Université de Bordeaux, France.
xavier.hinaut@inria.fr

I. Introduction

The main paper is published in the proceedings of the IEEE ICDL-EPIROB 2018 conference, please refer to this citation:

> Xavier Hinaut. Which Input Abstraction is Better for a Robot Syntax Acquisition Model? Phonemes, Words or Grammatical Constructions?. *2018 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, Sep 2018, Tokyo, Japan. https://hal.inria.fr/hal-01889919

A. General aims of the model

The model has several aims: (1) to model human sentence processing, and (2) to enable robots to acquire quickly the ability to parse sentences with few training data. The model is not designed for a particular language: we showed that the same model[1] could generalize on English and French sentences [5]. In a preliminary study[2], we showed that it could be applied to fifteen different languages from different language families from Asia or Europe. Another interesting property is that it generates a dynamic probabilistic estimation of the outputs (thematic roles) during the processing of a sentence, and updates the outputs each time a new word comes in.

B. Broca sentence parsing model (continued)

Previous versions of the model [3]–[5] were applied only on grammatical constructions: i.e. sentences in which the content words (e.g. nouns, verbs, ...) are replaced by a common marker (i.e. a placeholder). The idea behind this was multi-fold:

- it was shown that infants are able to quickly extract and generalize abstract rules in artificial language tasks [8],
- it is inspired from language acquisition theories, and in particular the construction framework [2], [9],
- relying on word placeholders instead of words themselves could enable to generalize faster with smaller data, because sentences could share the same grammatical construction without having the same content words in the placeholder slots.

This last idea is particularly interesting when incorporating the model inside a robot, because it can thus generalize within a few dozen of sentences if variability of sentences structures is limited [4].

C. Long term goals for robots acquiring syntax

This study takes place in a broader perspective with several long term goals: (1) construct a unified architecture that would output the meaning of the sentence given the acoustic raw inputs, thus taking care of all steps from phonemes, word and thematic role recognition; (2) enable robots with cognitive architectures to have more flexibility and adaptability while dealing with natural language processing instead of using black-box speech tools; (3) enable more robust speech recognition for particular robotic applications instead of using pretrained architecture that may not be well suited.

II. Methods

A. Intuitions on ESNs

Unlike other RNNs, the input layer and the recurrent layer (called "reservoir") do not need to be trained. In contrast, the random weights of the ESN's reservoir are scaled to possess the echo state property [6]: i.e. to possess suitable dynamics (e.g. "edge of chaos") that enable generalization. Theses dynamics obtained are a non-linear transformation of the inputs, which include "useful" computations for the given task. The objective is to have reservoir states that are linearly separable and that can be mapped to the output layer using a computationally cheap linear regression.

B. Training and testing (more details)

In the case of sentences that do not contain the maximal number of SWs (*i.e.* 8) we discard the remaining outputs

[1] The same instance of reservoir with the same random weights.

[2] X. Hinaut and J. Twiefel, "Teach your robot your language! trainable neural parser for modelling human sentence processing: Examples for 15 languages," (IN REVISION).

because no SW in the input sentence could be linked to them: e.g. if there are only four SWs in the sentence, we discard outputs concerning SWs 5 to 8. Unit activations of the discarded SW-outputs represent predictions about SWs that will never occur. The output layer is not limited to a particular size: the number of output units depends on the maximum number of semantic words in the sentence (and the number of predicates), for a given corpus. In [3] and unpublished works we successfully made experiments with more than 10 semantic words and more than 4 predicates.

C. Simulating noisy speech recognition

What would happen if we now embed this architecture in a robotic system that have to deal with real speech inputs? All the words of the sentences would probably not be correctly recognized. The state of the art speech-to-text engines claim to perform a WER of 5%: which means that 5 words on a 100 are not correctly recognized in average. In order to simulate these conditions and see how bad recognition of words affects the performances, we augmented the dataset by duplicating each sentence 20 times and randomly replacing 5% of the words by another word in the list of all possible words present in the corpus. This makes an important change in the corpus, because many sentences have two or more words that have been changed. We did not enable the option of replacing infrequent words in the corpus this time, because each sentence would appear several times in the corpus. Here, we performed a 4-fold cross-validation instead of a 10-fold cross-validation in order to save time and because it does not seem to influence the results to have more folds.

D. Hyper-parameters and implementation details

Hyper-parameters and implementation details are available in the supplementary materials. The input W_{in} and recurrent W weight matrices are randomized following these distributions: values are taken with equiprobability in the set $\{-1, 1\}$ for W_{in}, and with a normal distribution with 0 mean and $Wstd$ standard deviation for W. Both matrices have a sparsity of 0.2, i.e. only 20% of the connections are non-zero. After random initialization, the input matrix W_{in} is scaled with a scalar value called *input scaling* (IS), and the absolute maximum eigenvalue of the recurrent matrix W is scaled by the *spectral radius* (SR) value.

We use 500 internal units inside the recurrent layer (i.e. the reservoir). A few hyper-parameters were optimized using the *hyperopt* python toolbox [1], namely the *input scaling* (IS), the *leak rate* α, the *regularization parameter* (ridge), the standard deviation of the generated internal weights $Wstd$ and the threshold θ under which Infrequent Words are replaced. When exploring hyperparameters we average each data point over 10 instances with the same hyperparameters. A set of rounded parameters were then chosen from the parameter space region leading to good performance for the three PHON, WORD, CONST conditions: SR=1, IS=0.6, Wstd=0.1 and α=0.06. The ridge (regularization parameter) and θ were dependent on the experimental conditions.

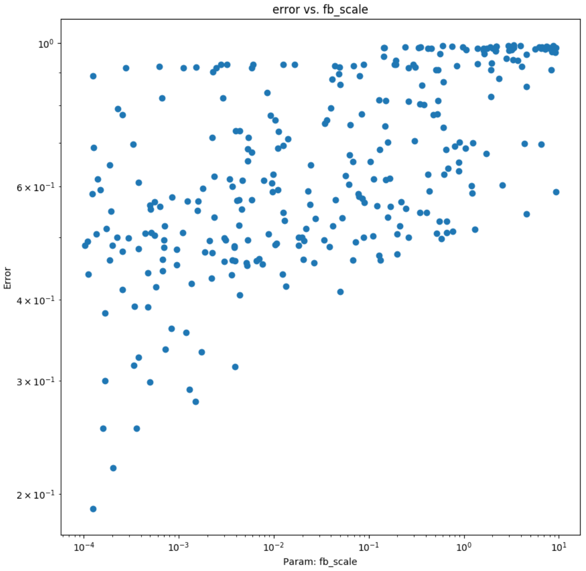

Fig. 1. Error vs. feedback scaling in a 300 trials random exploration of hyperparameters for PHON condition. Best results are obtained for the minimal values of *feedback scaling* (left exploration boarder).

III. Results

A. On the use of feedback connections

We explored whether adding a feedback connection from the output units to the reservoir could improve the performance. We performed learning by applying teacher forcing of the feedback [6], [7] *i.e.* during training the feedback activity given by the readout is given by the teacher signal. We explored the hyper-parameters (HPs), with additional ones specific to feedback: *feedback scaling*, *feedback connection probability* and *feedback activation function*. Activation functions considered (for the HP search) were: linear (identical as readouts), piecewise linear (linear from 0 to 1 and clamping values otherwise) and heavyside changing from 0 to 1 at 0.5. An example of exploration can be seen on Figure 1 for PHON condition. Best results are obtained for the minimal values of *feedback scaling* (left exploration border), a linear activation function and a connection probability of 0.6. For the WORD condition, we experimented with lower values of feedback, and similar results were found: the best results were obtained for minimal values of *feedback scaling* as can be seen on Figure 1. In summary, the use of feedback with low *feedback scaling* seems to have comparable performance to that of without feedback. We would have expected the opposite, that the feedback from the readout units could help to generalize better, especially in the PHON experiment for which information has to be kept for longer periods of time than WORD or CONST conditions, because each word (in WORD and CONST cond.) lasts one time step; instead in PHON cond. it lasts several time steps (i.e. the number of phonemes in the given word).

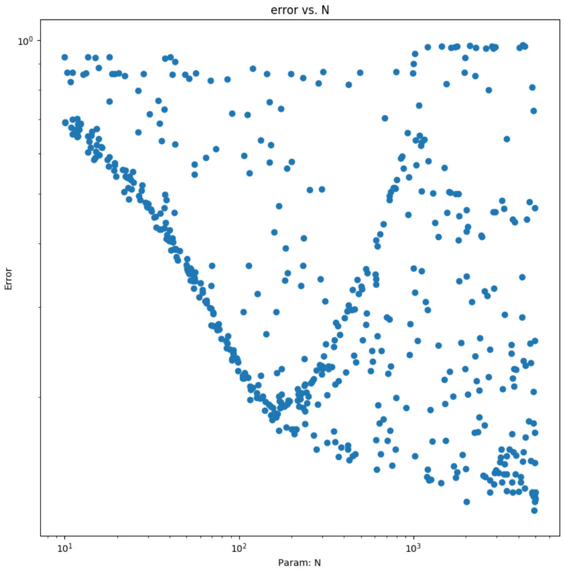

Fig. 2. Error vs. number of units N in the reservoir 550 trials random exploration of hyperparameters for WORD experiment. Only two hyperparameters are varied here: N and *ridge* (regularization parameter). An inverted gaussian shape can be seen, letting think that at its minimum (around 200) is the optimal number of reservoir units. But the group of points on the bottom right lead to lower error rate with much more units (5000). This group reveals the importance of the regularization parameter which enables to lower the error until 12%.

B. Does the performance increase with reservoir size?

Figure 2 shows the error as a function of the number of units in the reservoir for 550 trials random exploration of hyperparameters. Performance increases with the number of units in the reservoir and reaches approximately 88% (± 1.64) for WORD experiment (i.e. 12% error), which is the maximal border of the exploration space for N.

It is interesting to see that even with a small corpus, an increasing number of neurons can be used (jointly with regularization) to increase the performances. Considered in a developmental perspective, this means that a cognitive agent (child or robot) could dedicate an increasing number of its resources (*i.e.* neurons or computational substrates) to a particular task until its performances converge. More experiments would be needed to explore whether a growing reservoir would be able to increase its performances by using online learning, while adapting the regularization parameter (augmenting most probably).

References

[1] J. Bergstra, D. Yamins, and D. D. Cox, "Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms," in *Proceedings of the 12th Python in Science Conference*, 2013, pp. 13–20.

[2] A. Goldberg, *Constructions: A construction grammar approach to argument structure*. University of Chicago Press, 1995.

[3] X. Hinaut and P. Dominey, "Real-time parallel processing of grammatical structure in the fronto-striatal system: a recurrent network simulation study using reservoir computing," *PLoS ONE*, vol. 8, no. 2, p. e52946, 2013.

[4] X. Hinaut, M. Petit, G. Pointeau, and P. Dominey, "Exploring the acquisition and production of grammatical constructions through human-robot interaction with echo state networks," *Frontiers in Neurorobotics*, vol. 8, 2014.

[5] X. Hinaut, J. Twiefel, M. Petit, P. F. Dominey, and S. Wermter, "A recurrent neural network for multiple language acquisition: Starting with english and french," in *NIPS 2015 Workshop on Cognitive Computation: Integrating Neural and Symbolic Approaches*, 2015.

[6] H. Jaeger, "The "echo state" approach to analysing and training recurrent neural networks," *Bonn, Germany: German National Research Center for Information Technology GMD Tech. Report*, vol. 148, p. 34, 2001.

[7] M. Lukoševičius, "A practical guide to applying echo state networks," in *Neural Networks: Tricks of the Trade*. Springer, 2012, pp. 659–686.

[8] G. Marcus, S. Vijayan, S. Rao, and P. Vishton, "Rule learning by seven-month-old infants," *Science*, vol. 283, no. 5398, pp. 77–80, 1999.

[9] M. Tomasello, *Constructing a language: A usage based approach to language acquisition*. Cambridge, MA: Harvard University Press, 2003.