# Teach Your Robot Your Language! Trainable Neural Parser for Modelling Human Sentence Processing: Examples for 15 Languages

Xavier Hinaut[1,2] and Johannes Twiefel[2]

***Abstract*— We present a Recurrent Neural Network (RNN) that performs thematic role assignment and can be used for Human-Robot Interaction (HRI). The RNN is trained to map sentence structures to meanings (e.g. predicates). Previously, we have shown that the model is able to generalize on English and French corpora. In this study, we investigate its ability to adapt to various languages originating from Asia or Europe. We show that it can successfully learn to parse sentences related to home scenarios in fifteen languages: English, German, French, Spanish, Catalan, Basque, Portuguese, Italian, Bulgarian, Turkish, Persian, Hindi, Marathi, Malay and Mandarin Chinese. Moreover, in the corpora we have deliberately included variable complex sentences in order to explore the flexibility of the predicate-like output representations. This demonstrates that (1) the learning principle of our model is not limited to a particular language (or particular sentence structures), but more generic in nature, and (2) it can deal with various kind of representations (not only predicates), which enables users to adapt it to their own needs. As the model is inspired from neuroscience and language acquisition theories, this generic and language independent aspect makes it a good candidate for modelling human sentence processing. It is especially relevant when this model is implemented in grounded multimodal robotic architectures.**

## I. INTRODUCTION

Communicating with the present day robots is a challenging task for humans. It involves either learning a programming language or using complex interfaces. The most sophisticated robots, that recognise commands given orally, are often limited to a pre-programmed set of stereotypical sentences such as "Give me cup". In this study, we propose an approach that allows one to use natural language when interacting with robots in languages originating from Asia and Europe. Our architecture enables the users to train a sentence parser easily on potentially many different contexts, including home scenarios (e.g. grasping remote objects, cleaning furnitures, ...) [1]. It can also enable them to directly teach new sentences to the robot [2].

*We gratefully acknowledge support by a Marie Curie Intra European Fellowship within the 7th European Community Framework Programme: EchoRob project (PIEF-GA-2013-627156). We also thank support for the project "LingoRob - Learning Language in Developmental Robots" by Campus France PHC Procope Project 37857TF and by DAAD Project 57317821 in the Förderprogramm "Projektbezogener Personenaustausch Frankreich".

[1]X. Hinaut is with Inria Bordeaux Sud-Ouest, Talence, France; LaBRI, UMR 5800, CNRS, Bordeaux INP, Université de Bordeaux, Talence, France; and Institut des Maladies Neurodégénératives, UMR 5293, CNRS, Université de Bordeaux, Bordeaux, France. xavier.hinaut@inria.fr

[2]X. Hinaut and J. Twiefel are with Knowledge Technology Group, Department Informatik, Universität Hamburg, 22527 Hamburg, Germany. surname@informatik.uni-hamburg.de

Current approaches typically involve developing methods specifically aimed at — or only tested on — the English language. Specific parsers then need to be designed for other languages, often adapting the algorithms and methods to their specificities. In order to overcome these limitations we propose a simple and flexible way of training a sentence parser for an arbitrary language, without requiring any new programming effort. This enables one to easily create parsers even for languages which do not have a significant corpus available. We believe that fast trainable parsers that are simple and flexible are a key component to make home robotics more efficient and provide social adaptation to the users in particular contexts.

In the current paper, we present a previously developed neural model [3], how it works, what its main properties are and briefly mention an available ROS module that could be used within robotic architectures. Then, we demonstrate that this model is able to learn more flexible representations than what was previously thought. In particular, these representations could be easily defined by users and adapted to various languages. This is done using home scenario corpora with sentences and representations of various complexities. More importantly, we show that this is not limited to a particular language or language family, but works for a variety of European and Asian languages: English, German, French, Spanish, Catalan, Basque, Portuguese, Italian, Bulgarian, Turkish, Persian, Hindi, Marathi, Malay and Mandarin Chinese (traditional and simplified). Note that the all corpora have been written by the respective language experts and not using online or other automatic translator. Afterwards, we discuss the results and give examples of particular linguistic cases that are challenging to learn. Finally, we elaborate why the model is not limited to learn predicate-like representations, but should be able to learn multimodal grounded ones.

## II. MODEL OF HUMAN SENTENCE PROCESSING FOR HRI

### A. Motivations

In this subsection, we explain why we choose the *Echo State Network* (ESN) architecture as the core part of the neural parser: from biological inspiration and language acquisition theories, to its advantages for Human-Robot Interaction (HRI). We propose it as one of the tools useful to study how robot grounded representations could be manipulated in syntactic sequence of symbols, regardless of the language used.

Firstly, since brain processes involved in language acquisition and sentence comprehension are still poorly understood, modelling is the key to make further steps in understand these processes further. Our neural parser aims at modelling brain processes of sentence comprehension while using language acquisition theories [3]. Reservoir Computing paradigm (such as ESNs) is considered to be a biologically plausible model of "canonical circuits" (i.e. generic pieces of cortex) and originates from computational neuroscience [4][5]. For instance, compared to more classical methods of RNN training, Reservoir Computing (RC) learning mechanism does not unfold time, which makes it closer to learning processes in the brain than Back-Propagation Through Time (BPTT) [6]. More recently, it was also used in neuroscience studies to decode the neural activity of primate prefrontal cortex [7], probably because its non-linear computations in high-dimensional space are in some way similar to the dynamics of prefrontal cortex.

Secondly, we believe that the HRI applications need modules that can be trained quickly (e.g. via one-shot offline learning), executed in tens of milliseconds, and can process inputs in an incremental fashion [8][9] (word by word). Additionally, our neural parser is interesting in the context of HRI because there is no need to predefine a parser for each language; just a training corpus is needed. It is able to process natural language sentences instead of stereotypical sentences (like "put cup left"); and it generalizes to unknown sentence structures (that are not present in the training data set).

Thirdly, as we aim to model cognitive language processing, particularly in robotic architectures, we want the model – and the symbols it manipulates – to be able to interact with multimodal and grounded representations [10][11] and enable them to emerge [12] "through" it. The model has several features that could facilitate this: online processing (process word by word the sentence) [3], anytime algorithm (ability to have a partial or complete answer before the end of a sentence [3]), ability to train it in a fully incremental fashion (time step by time step [13]), ability to process distributed representation of words instead of localist representations (unpublished work). This link with multimodal and grounded representations will be further detailed in the discussion. Finally, some studies investigated the ability of such networks to be embodied: a physical version of Reservoir Computing (the framework containing ESNs) was shown to be able to perform embodied or morphological computation [14][15].

### *B. Reservoir Sentence Processing Model*

How do children learn language? In particular, how do they associate the structure of a sentence to its meaning? This question is linked to a more general issue: how does the brain associate sequences of symbols to internal symbolic or sub-symbolic representations? Generic neural architectures are needed to help the modelling such general processes. Echo State Networks (ESNs) [16] are not hand-crafted for a particular task, but on the contrary can be used for a broad range of applications. ESNs are neural networks with a random recurrent layer and a single linear output layer (called "read-out") modified by online or offline learning. ESNs are not primarily aimed at symbolic processing, but as a general architecture, they permit such application. Much research has been done on language processing with neural networks [17][18][19] and in particular with ESNs [20][21]. The tasks used were diverse, from predicting the next symbol (i.e. word) in a sentence to speech processing. In this paper, the task we perform is thematic role assignment.

Mapping the surface form (sequence of words) onto the deep structure (meaning) of a sentence is not an easy task since making word associations is not sufficient. For instance, a system relying only on the order of semantic words (cat, scratch, dog) to assign thematic roles is not enough for the following simple sentences, because even if cat and dog appear in the same order they have different roles: "The cat scratched the dog" and "The cat was scratched by the dog". It was shown that infants are able to quickly extract and generalize abstract rules [22], and we want to take advantage of this core ability without going through all the steps of abstractions, but rather start at a certain level of abstraction to build "meta-abstractions" on top. Thus, to be able to learn such a mapping (from sequence of words to meaning) with a good generalization ability, we rely on abstractions of sentences rather than sentences themselves.

In order to teach the model to extract the meaning of a sentence, we base our approach on the notion of *construction*: the mapping between a sentence's form and its meaning. The hypothesis that children learn such constructions comes from developmental theories of language acquisition [23][24][25]. It assumes children construct abstract linguistic categories and schemas during language acquisition. Constructions are an intermediate level of meaning between the smaller constituents of a sentence and the full sentence itself. For instance, a simple construction that a child could learn is "Give me *X*." when he/she wants to obtain any object *X* out of reach. Constructions could be of variable size and complexity from morpheme (e.g. *anti-*, *pre-*, *-ing*) to passive sentences ("The X was Y by the Z") [24]. Based on the cue competition hypothesis of Bates and MacWhinney [26], we make the assumption that most of the mapping between a given sentence and its meaning can rely on the order of words, and particularly on the pattern of function words and morphemes [27].

The reservoir sentence processing model was used in previous experiments studying it properties to model human sentence processing [3] and its application to HRI [2]. The model learns the mapping of the semantic words (i.e. content words like nouns, verbs, adjectives, adverbs) of a sentence onto different slots (the thematic roles: e.g. action, location) of a basic event structure (e.g. *action(object, location)*). In other words, the task to find the correct predicates for a sentence is equivalent to finding all the correct roles of the SW of this sentence. As depicted in Fig. 1, the system processes a sentence as input and generates corresponding predicates. Words that are not Semantic Words (SW) are Function Words (FW), which means that they have a particular syntactic

function in a sentence (eg: "Give the ball *to* the dog": here "to" indicates the recipient of an action).

In other words, most of the time it is possible to find the role of a SW without knowing that word or any other SW in the sentence (see [27][3] for more details). This is one of the reasons why it works well with many languages. As we can see in Figure 1, before being fed to the ESN, sentences are transformed into a sentence structure (or *grammatical construction* [23]). The SWs (i.e. the words that have to be assigned a thematic role) are replaced by the *SW* item before entering the ESN. The processing of the grammatical construction is sequential (one word at a time) and the final estimation of the thematic roles for each SW is read-out at the end of the sentence. A current estimation of the predicted roles is however available at each time step.

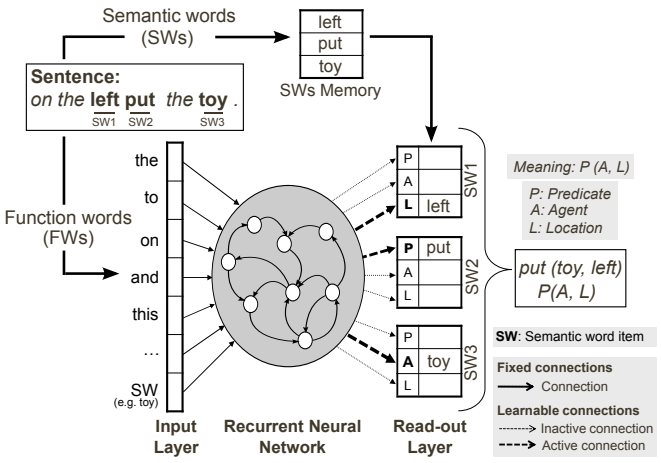


Fig. 1. Sentences are converted to a sentence structure by replacing semantic words by a SW marker. The ESN is given the sentence structure word by word. Each word activates a different input unit. During training, the connections to the readout layer are modified to learn the mapping between the sentence structure and the arguments of the predicates. When a sentence is tested, the most active units are bound with the SW kept in the SWs memory to form the resulting predicate. (Adapt. from [2].)

### C. Advantages of processing constructions

By processing *grammatical constructions* [23] and not sentences *per se*, the model is able to bind a virtually unlimited number of sentences to these sentence structures. Based only on a small training corpus (a few tens of sentences), this enables the model to process future sentences with currently unknown semantic words if the sentence structures are similar. Considering a robot that could learn several new semantic words each day is an interesting advantage: the robot could learn the SW independently of the sentence. For instance, it learns some sentences on the 1st day; on the 2nd day it learns to associate visual features with an object's name (i.e. grounding the given name); on the 3rd day it is able to understand new sentences with these SW as long as they have appeared in similar context than other SW. Otherwise it still can learn sentences about new contexts.

### D. Usage of the ROS Module

The proposed model was encapsulated in a ROS module [28]. It is coded in the Python language and uses the *rospy* library. The source code, implemented as a ROS module, is available at `github.com/neuronalX/EchoRob` On this repository, one can find the different versions of the models developed and more corpora are available.

When running the program, the reservoir model is trained using a text file and a ROS service is initialized. The training text file contains sentences along with their corresponding predicates which make it easily editable. If predicates represent multiple actions that have to be performed in a particular order, the predicates have to be specified in chronological order:

- **`bring coffee me,`** `clean table ;` **`bring me the coffee`** `before you clean the table`
- **`bring coffee me,`** `cleaning table ;` `before cleaning the table` **`bring me the coffee`**

This predicate representation enables one to easily integrate this model into a robotic architecture because motor primitives can be represented in this way. It enables the users to define which kind of predicate representation they want to use. By convention the first word is the predicate (i.e. which plays the role of a kind of function in the programming sense), and the following words are the arguments (any number of arguments could be given).

Once initialized, a request could be sent to the ROS service: it accepts a sentence (text string) as input and returns an array of predicates in real time. With an ESN of 100 units, training the neural parser to learn 200 sentences takes about one second on a laptop computer. Testing a sentence is of the order of 10 ms.

### E. Previous results

The ability of the model to scale to larger copora depends on the similarity of the sentence structures in the corpus. If sentence structures are different variants of similar expressions, then there is no need to use a high number of units in the reservoir. For instance in [3] we have shown that a 5,000 units reservoir could generalize up to 85% on unkown sentence structures on a corpus of size 90,000.

In Hinaut et al. (2015) [29], it has been shown that the model can learn to process sentences with out-of-vocabulary words [30] (i.e. words that are not in the speech recognizer's vocabulary). Moreover, we demonstrated that it can generalize to unknown grammatical constructions in both French and English at the same time. To illustrate how the robot interaction works, a video is available at `youtu.be/FpYDco3ZgkU` [31][32].

In [32] we have integrated this ROS module in a robotic architecture with speech and visual object recognition. Once this ROS module is integrated in such a system, it could be employed to process various hypotheses generated by the speech recognizer: e.g. for one utterance it generates the 10 most probable sequence of words, then the ROS module returns the retrieved predicates for each hypothesis, finally allowing a semantic analyser or world simulator to choose the predicates with the highest likelihood. Preliminary

work has shown that the model could be trained in a fully incremental fashion [13]: we plan to add this feature to the ROS module in the future.

One may argue that the main drawback of the architecture is the fact that it only relies on function words to interpret a sentence: this may cause some ambiguities when an identical sentence structure corresponds to different meanings (see [3] for more details). However, relying on function words is also its strength since it enables to generalize to new inputs with only tens of sentence structures in the training corpus [3][2]. We believe this is not a limitation of the architecture itself since some current unpublished experiments show that real semantic words (distributed representations using word2vec [33]) could be used instead of SW markers (i.e. fillers).

## III. Methods

### A. Echo State Networks

The neural parser is based on an ESN [16] – a particular kind of recurrent neural network (RNN) – with leaky neurons: inputs are projected to a random recurrent layer and a linear output layer (called "read-out") is modified by learning (which can also be done in an online fashion).

Compared to other RNNs, the input layer and the recurrent layer (called "reservoir") do not need to be trained. For other RNNs, the structure of the recurrent layer evolves in most cases by gradient descent algorithms like Backpropagation-Through-Time [6], which is not biologically plausible and is adapted iteratively to be able to hold a representaion of the input sequence. In contrast, the random weights of the ESN's reservoir are not trained, but adapted to possess the echo state property [16] or at least suitable dynamics (e.g. "edge of chaos") to generalize, which includes a non-linear transformation of the input that can be learned by a linear classifier. The weights are adapted by scaling the weights based on the maximum absolute eigenvalue (also called spectral radius), which is a hyperparameter specific to the task. The states of the reservoir are linearly separable and can be mapped to the output layer by a computationally cheap linear regression, as no gradient descent is necessary. The weights of the input layer can be scaled by the input scaling hyperparameter, which also depends on the nature of the inputs. The size of the reservoir is another task-specific hyperparameter, which can be fixed for hyperparameter search, as the other hyperparameters scale with a larger reservoir [34].

The units of the recurrent neural network have a *leak rate* ($\alpha$) hyper-parameter which corresponds to the inverse of a time constant. These equations define the update of the ESN:

$$\mathbf{x}(t+1) = (1-\alpha)\mathbf{x}(t) + \alpha f(\mathbf{W^{in}}\mathbf{u}(t+1) + \mathbf{W}\mathbf{x}(t)) \quad (1)$$

$$\mathbf{y}(t) = \mathbf{W^{out}}\mathbf{x}(t) \quad (2)$$

where $\mathbf{u}(t)$, $\mathbf{x}(t)$ and $\mathbf{y}(t)$ are the input, the reservoir (i.e. hidden) state and the read-out (i.e. output) states respectively at time $t$. $\alpha$ is the *leak rate*. $\mathbf{W^{in}}$ , $\mathbf{W}$ and $\mathbf{W^{out}}$ are the input, the reservoir, and the read-out matrices respectively. $f$ is the *tanh* activation function. After the collection of all reservoir states, the following equation defines how the read-out (i.e. output) weights are trained:

$$\mathbf{W^{out}} = \mathbf{Y^{d}}[1;\mathbf{X}]^{+} \quad (3)$$

where $\mathbf{Y^{d}}$ is the concatenation of the desired outputs, $\mathbf{X}$ is the concatenation of the reservoir states (over all time steps for all trained sentences) and $\mathbf{M}^{+}$ is the Moore-Penrose pseudo-inverse of matrix $\mathbf{M}$.

### B. Preprocessing

For some languages like Spanish, Basque, Turkish, etc., to understand a sentence one may need to extract some word suffixes (or prefixes) from semantic words. In Spanish for instance, "sirve -lo" is usually written as one word ("sirvelo") even if its translation to English is composed of two words ("pour it"; "it" being a cup). This suffix extraction may not be mandatory for all the languages (e.g. French, English).

As we want the model to process sentence structures (e.g. "The X was X-ed by the X."[1]) instead of true sentences (e.g. "The dog was scratched by the cat."), we need to replace all the content words (i.e. semantic words – SW) by a common marker (*X* was used in the example here). This preprocessing can include or not the extraction of prefixes and suffixes (e.g. *-ed*) depending on the language. In this study, we extracted suffixes for a given language only when it seemed useful for the model to learn[2]. This preprocessing to distinguish SW from FW can be done in two ways, one may be more convenient than the other depending on the application:

- define a list of FWs (i.e. all the words that are not in the meaning representation part of the construction) and consider any other word as SW;
- define a list of SWs (i.e. words that are in meaning representations), and consider any other word as FW.

In the current experiments, we used the latter by defining as semantic words (SW), all the words that were in the meaning parts of each corpus (i.e. the predicates). Then all the remaining words, including grammatical markers (suffixes beginning with "-") and commas (for German and Bulgarian), were defined as function words (FW). In a few cases, for some languages we had to slightly modify the list of FW[3] as we did not ask corpus translators to optimize their translation for the model – we wanted the corpora to be close to what any future user of the system could do. Of course, the boundary between FW and SW is not well defined and depending on the context one may want to include it in the meaning part or not. As we state in the Discussion (the use of word "on" as both SW and FW) this could be an advantage because it shows the system can learn these particular cases.

[1]In this study suffixes were not extracted for English corpus. This sentence is just given an example for a matter of understanding.

[2]This extraction of suffixes was done manually given the smallness of the corpus and because exploring automatic extraction of suffixes for all these different languages is out of the scope of this study.

[3]The list of FW used for each corpus is available at `github.com/neuronalX/EchoRob/blob/master/corpora/2017_TCDS_15languages__function-words-FW-used.py`

As explained is subsection II-C, this preprocessing is interesting for HRI because it enables the parser to not rely on semantic words, which enables the robot system to use newly learned semantic words with previously learned data. Currently, there was no need to use part-of-speech (POS) tagging, but it may help to add this complementary information in the input representation of words.

### C. Training the Neural Parser

First, an instance of ESN is created by generating the random weights for the input connections and the recurrent connections. Then, the neural model was trained independently for each corpus. The training of the read-out (i.e. output) connections was done using one shot linear regression with bias. The training is done on the full corpus (21 sentence-meaning pairs) and tested on the same corpus. We use the same corpus for training and testing because we are not interested in the generalization abilities, but on the learnability of each corpus; the question is: Is it possible that a relatively small reservoir finds a way to map inputs (sentence structures) and outputs (predicate-like meanings) for all languages?

Hyper-parameters that can be used for this task are as follows: *spectral radius*: 1, *input scaling*: 0.75, *leak rate*: 0.167. We did not optimise these hyper-parameters to try to have optimal performance with the lowest number of neurons. We simply used values taken from previous similar experiments [2][3][32]: these hyper-parameters are robust to modifications. The number of neurons in the reservoir (i.e. the recurrent layer) was either 50 or 75 depending on the corpus: we did not search for the minimum number of neurons in detail, we just changed it by a 25 unit step.

## IV. Home Corpora in Multiple Languages

### A. Introduction

In this section, we give the full corpus for English (21 meaning-sentence pairs), the corresponding corpus with constructions giving an idea of how the reservoir model "perceives" the inputs and outputs, and some particular sentences of interest for fifteen other corpora[4]). Some languages are written in their alphabet of origin and some are using Roman alphabet. All corpora are available at `github.com/neuronalX/EchoRob/blob/master/corpora/2017_TCDS_15languages.txt`.

Each corpus is organised as follows. Each line contains a sentence together with its corresponding meaning representation: left part of the semi-column is the meaning, right part is the sentence:

1) open door ; open the door
2) take coffee, pour coffee into mug ; take the coffee and pour it into the mug

One can see that the way to write predicates is fairly intuitive and can be done without prior knowledge of a predefined output structure. The words in the left part (i.e. meaning) may not be a particular form of a verb (noun with plural, verb with conjugation, ...), this is because each these words has to be written the same way as in the right part (i.e. sentence) for the preprocessing to know which word is refereed to.

[4]There are sixteen corpora in total, corresponding to fifteen different languages, and two versions of Mandarin (modern and traditional).

### B. English

1) open door ; open the door
2) answer phone ; answer the phone
3) water plants ; water the plants
4) clear table ; clear the table
5) take coffee, pour coffee into mug ; take the coffee and pour it into the mug
6) clean table, put mug on table ; clean the table and put the mug on it
7) put mug on left ; put the mug on the left
8) get phone, bring phone me ; get the phone and bring it to me
9) go to bathroom ; go to the bathroom
10) make tea me ; make me some tea
11) tell joke me ; tell me a joke
12) make sandwhich me, sandwhich tomatoes ; make me a sandwhich with tomatoes
13) bring newspaper me, newspaper on table, table in kitchen ; bring me the newspaper which is on the table in the kitchen
14) bring dress me, dress blue, dress in closet ; bring me the blue dress that is in the closet
15) bring pen me, pen blue, pen beside cup, cup red ; bring me the blue pen beside the red cup
16) dance for me, clean floor, clean same time ; dance for me and clean the floor at the same time
17) switch off light ; switch off the light
18) find bottle, bottle water ; find the bottle of water
19) search object, object small pink ; search for the small pink object
20) search recipe internet, recipe tiramisu ; search the recipe of tiramisu on internet
21) check if, if husband home, husband my, if five ; check if my husband is at home at five

### C. English constructions

In the following, we show what the English corpus become after preprocessing; i.e. the sentence structures on the right parts are input to the reservoir model. To make them more human-friendly we wrote for instance "W the X and M it N the O", but actually the true input given to the neural network is "SW the SW and SW it SW the SW": there is no difference between the semantic words inputs, they are all represented as a SW marker. All the capital variables are arbitrary names: W, X, Y, ... In order to make them more readable, their are organised as groups always in the same order (and at the same position in the predicate-slot), in order to represent the four possible predicate-like parts of the meaning (i.e. right-hand of each line): [W X Y Z], [M N O P], [I J K L], [A B C D]. Sometimes a semantic word is used in several predicates (for a same given construction), it will then have

the same letter in all these slots. For instance, this is the case for construction 8, where X is at position 2 of the 1st predicate and at position 2 of the 2nd predicate: "W X, M X O ; W the X and M it to O". Given these "precautions" one can easily see some patterns appearing in the corpus.

1) W X ; W the X
2) W X ; W the X
3) W X ; W the X
4) W X ; W the X
5) W X, M X O P ; W the X and M it O the P
6) W X, M N O X ; W the X and M the N O it
7) W X Y Z ; W the X Y the Z
8) W X, M X O ; W the X and M it * O
9) W X Y ; W X the Y
10) W X Y ; W Y some X
11) W X Y ; W Y a X
12) W X Y, X N ; W Y a X with N
13) W X Y, X N O, O J K ; W Y the X which is N the O J the K
14) W X Y, X N, X J K ; W Y the N X that is J the K
15) W X Y, X N, X J K, K B ; W Y the N X J the B K
16) W X Y, M N, M J K ; W X Y and M the N at the J K
17) W X Y ; W X the Y
18) W X, X N ; W the X of N
19) W X, X N O ; W * the N O X
20) W X Y, X N ; W the X of N * Y
21) W X, X N O, N J, X B ; W X J N is at O at B

One can see the following symbol in the corpus "*" which needs an explanation. Some words could be considered both as semantic and function word by the language experts whom wrote a given corpus: in practice this means that the word would sometime appear on the meaning (left) part, and sometimes not. Consequently these words need to be always considered as semantic word by the model, in order to let it bind a thematic role to them. Thus even if the word do not appear in the left-hand part, it needs to be a SW marker in the right-hand part[5]. In this case we noted this word with "*" symbol.

### D. Examples from the Corpora

The examples provided in this subsection (sentence 2, 6, 8, 13, 19 and 21) are available (by order) in the following languages:

1) English
2) German
3) Spanish
4) French
5) Italian
6) Catalan
7) Simplified Mandarin
8) Traditional Mandarin
9) Malay
10) Turkish
11) Marathi
12) Hindi
13) Basque
14) Persian
15) Portuguese
16) Bulgarian

[5]That is why the words "for", "on" and "to" are not in the close class word list in the supplementary data.

*a) Sentence 2:* This sentence is a good example for a simple mapping. By learning this mapping, the model is now able to generate predicates for all kind of these sentences, which are often used in simple HRI scenarios, like "open the door", "water the plants" or "clear the table". It is not necessary for the model to know the words "open", "door", "water", "plants", "clear", or "table".

1) answer phone ; answer the phone
2) geh an Telefon ; geh an das Telefon
3) contesta teléfono ; contesta el teléfono
4) réponds téléphone ; réponds au téléphone
5) rispondi telefono ; rispondi al telefono
6) contesta telèfon ; contesta el telèfon
7) 接 电话 ; 接 那 -通 电话
8) 接 電話 ; 接 那 -通 電話
9) jawab telefon ; jawab -kan panggilan telefon itu
10) cevap ver telefon ; telefon -a cevap ver
11) uchal phone ; phone uchal
12) dho javaab phon ; phon ka javaab dho
13) telefono erantzun ; telefono -a erantzun
14) javab telfon ; telfon ra javab bede
15) atenda telefone ; atenda o telefone
16) вдигни телефон ; вдигни телефон -а

*b) Sentence 6:* In this example, a little more complex mapping with two predicates is given. As it can be seen, the original order of the tasks to be performed is preserved by the order of the predicates. Moreover, the possibility of using FWs as SWs is shown, like the word "on", "auf", "sobre", "dessus", ...

1) clean table, put mug on table ; clean the table and put the mug on it
2) reinige Tisch, stell Tasse auf Tisch ; reinige den Tisch und stell die Tasse auf ihn
3) limpia mesa, pon taza sobre mesa ; limpia la mesa y pon la taza sobre ella
4) nettoie table, met tasse dessus table ; nettoie la table et met la tasse dessus
5) pulisci tavolo, metti tazza sopra tavolo ; pulisci il tavolo e metti -ci sopra la tazza
6) neteja taula, posa tassa sobra taula ; neteja la taula i posa la tassa a sobra
7) 清理 桌子, 放 杯子 上 桌子 ; 清理 那 -张 桌子 再 把 那 -个 杯子 放 在 桌子 上
8) 清理 桌子, 放 杯子 上 桌子 ; 清理 那 -張 桌子 再 把 那 -個 杯子 放 在 桌子 上
9) kemas meja, letak cawan ke atas meja; kemas -kan meja itu dan letak -kan cawan itu ke atas nya
10) temizle masa, koy fincan üstünde masa ; masa -yi temizle -yip fincan -i üstünde koy
11) kar saf table, thev pela tya ; table saf kar ani pela tya

-wr thev
12) saaf karo mez, rakho mez mug ; mez saaf karo aur us par mug rakho
13) mahai garbitu, kiker ezarri mahai gainean ; mahai -a garbitu eta kiker -a ezarri gainean
14) tamiz miz, gzar livan rooy miz ; miz ra tamiz kon va livan ra rooy -e an bo- gzar
15) limpe mesa, coloque caneca nela mesa ; limpe a mesa e coloque a caneca nela
16) почисти маса, сложи чаша на маса ; почисти маса -та и сложи чаша -та на нея

*c) Sentence 8:* This sentence illustrates the problem of personal pronouns like “me”, “mir”, “-me”, “moi”, ... which are often suffixes of other words in several languages.

1) get phone, bring phone me ; get the phone and bring it to me
2) nimm Telefon, bring Telefon mir ; nimm das Telefon und bring es zu mir
3) toma teléfono, trae teléfono -me ; toma el teléfono y trae -me -lo
4) prends téléphone, amène téléphone moi ; prends le téléphone et amène le moi
5) prendi telefono, da telefono -mme ; prendi il telefono e da -mme -lo
6) pren telèfon, porta telèfon ‘m ; pren el telèfon i porta ‘m -ho
7) 取 手机, 拿 手机 我 ; 取 手机 再 把 它 拿 给 我
8) 取 手機, 拿 手機 我 ; 取 手機 再 把 它 拿 給 我
9) dapat telefon, ambil telefon saya; dapat -kan telefon itu dan ambil -kan -nya untuk saya
10) al telefon, getir telefon bana ; telefon -u al -ip bana getir
11) ghe phone, aan phone mazajaval ; phone ghe ani mazajaval aan
12) lo phon, lay aao mere phon ; phon lo aur mere paas lay aao
13) telefono hartu, neri ekarri ; telefono -a hartu eta neri ekarri
14) gir telfon, avar telfon man ; telfon ra be- gir va an ra baray -e man bi- avar
15) pegue telefone, traga telefone -me ; pegue e traga -me o telefone
16) вземи телефон, донеси телефон ми ; вземи телефон -а и ми го донеси

*d) Sentence 13:* Here, three predicates can be extracted along with personal pronouns. Note that the predicates are linked by one another by one word, making a simple branching tree structure (newspaper -on- table -in- kitchen). Additionally, it demonstrates one aspect of the flexibility of the system: nouns could be used as localisation predicates, indicating where these objects are.

1) bring newspaper me, newspaper on table, table in kitchen ; bring me the newspaper which is on the table in the kitchen
2) bring Zeitung mir, Zeitung auf Tisch ; bring mir die Zeitung , die auf dem Tisch ist
3) trae periódico -me, periódico sobre mesa, mesa en cocina ; trae -me el periódico que está sobre la mesa en la cocina
4) apporte journal moi, journal sur table, table dans cuisine ; apporte moi le journal qui est sur la table dans la cuisine
5) prendi giornale -mi, giornale trova sul tavolo, tavolo in cucina ; prendi -mi il giornale che si trova sul tavolo in cucina
6) porta periòdic ‘m, periòdic sobre taula, taula en cuina ; porta ‘m el periòdic que està sobre la taula en la cuina
7) 取 报纸 我, 报纸 上 桌子, 桌子 里 厨房 ; 给 我 取 在 厨房 里 桌子 上 的 报纸
8) 取 報紙 我, 報紙 上 桌子, 桌子 裡 廚房 ; 給 我 取 在 廚房 裡 桌子 上 的 報紙
9) ambil surat khabar saya, surat khabar atas meja, meja dalam dapur; ambil -kan saya surat khabar itu yang berada di atas meja di dalam dapur
10) getir gazete bana, gazete üstünde masa, masa -da mutfak ; mutfak -da olan masa -nin üstünde -ki gazete bana getir
11) aan wartapatra, wartapatra table, table svayampakgruh; jo wartapatra svayampakgruh -at table -wr ahe to aan
12) lavo mujhay akhbaar, mez akhbaar, rasoyi ghar mez;mujhay rasoyi ghar may mez par akhbaar lavo
13) egunkari ekar diezadazu, egunkari mahai gainean, mahai sukalde ; sukalde -ko mahai gainean den egunkari -a ekar diezadazu
14) avar roozname man, roozname rooy miz, miz dar ashpazkhane; roozname -yi ra ke rooy -e miz dar ashpazkhane ast ra baray -e man bi- avar
15) traga jornal -me, jornal na mesa, mesa na cozinha ; traga -me o jornal que está na mesa na cozinha
16) донеси вестник ми, вестник на маса, маса в кухня ; донеси ми вестник -а , който е на маса -та в кухня -та

*e) Sentence 19:* This sentence demonstrates another aspect of the flexibility of these predicates. Entities like “object” can be predicate with its arguments playing the role of describers or modifiers (“small”, “pink”).

1) search object, object small pink ; search for the small pink object
2) such Objekt, Objekt kleine pinke ; such das kleine pinke Objekt
3) busca objeto, objeto rosado pequeño ; busca el objeto rosado pequeño
4) cherche objet, objet petit rose ; cherche le petit objet rose
5) cerca oggetto, oggetto piccolo rosa ; cerca il piccolo oggetto rosa
6) cerca objecte, objecte petit rosat ; cerca l’ objecte petit i rosat
7) 寻找 物体, 物体 小件 粉红色 ; 寻找 那 -个 小件 的 粉红色 物体
8) 尋找 物體, 物體 小件 粉紅色 ; 尋找 那 -個 小件 的 粉紅色 物體

9) cari objek, objek kecil merah jambu; cari -kan objek kecil yang berwarna merah jambu itu
10) al şey, şey pembe küçük ; küçük pembe şey -i al
11) shodh vastu, vastu chhoti gulabi ; chhoti gulabi vastu shodh
12) doondo vasthu, chotay vasthu, gulaabi vasthu;chotay gulaabi vasthu doondo
13) gauz bilatu, gauz txiki arros ; gauz -a txiki eta arros -a bilatu
14) gard shey, shey koochak soorati ; donbal -e shey -e koochak -e soorati be- gard
15) procure objeto, objeto pequeno rosa ; procure pelo pequeno objeto rosa
16) намери предмет, предмет малк розов ; намери малк -ия розов предмет

*f) Sentence 21:* This most challenging meaning contains four predicates with a complex tree kind of structure: "if" appears two times as a predicate in non-consecutive slots, separated by a "modifier" predicate.

1) check if, if husband home, husband my, if five ; check if my husband is at home at five
2) überprüfe ob, ob Mann Hause, Mann mein, ob fünf ; überprüfe , ob mein Mann um fünf zu Hause ist
3) chequea si, si marido casa, marido mi, si cinco ; chequea si mi marido está en casa a las cinco
4) vérifie si, si mari maison, mari mon, si heures cinq ; vérifie si mon mari est à la maison à cinq heures
5) controlla se, se marito casa, marito mio, se cinque ; controlla se mio marito é a casa alle cinque
6) comprova si, si marit casa, marit meu, si cinc ; comprova si el meu marit està a casa a les cinc
7) 检查 是不是, 是不是 丈夫 家, 丈夫 我的, 是不是 五点钟 ; 检查 我的 丈夫 是不是 五点钟 有 在 家
8) 檢查 是不是, 是不是 丈夫 家, 丈夫 我的, 是不是 五點鐘 ; 檢查 我的 丈夫 是不是 五點鐘 有 在 家
9) memeriksa jikalau, jikalau suami rumah, suami saya, jikalau lima; memeriksa -kan jikalau suami saya berada di rumah pada pukul lima
10) kontrol et, koca ev, koca -m, saat beş ; koca -m ev -de saat beş -te olacagina kontrol et
11) tapas jr, jr pati ghari, pati maze, pach wajta ; jr maze pati pach wajta ghari astil ka he tapas
12) jaanch agar, agar ghar pathi, mera pathi, agar paanch bajay;jaanch agar mera pathi paanch bajay ghar par hain
13) egiaztatu, senar etxean bada, senar nere, bost bada ; egiaztatu nere senar -a etxean bada bost -etan
14) barrasi agar, agar shohar khane, shohar man, agar panj ; barrasi kon agar shohar -e man saat -e panj dakhel -e khane bashad
15) verifique se, se marido casa, marido meu, se cinco ; verifique se meu marido estará em casa às cinco
16) провери дали, дали съпруг вкъщи, съпруг мой, дали пет ; провери дали мой съпруг е вкъщи в пет

## V. Results

### A. Learning corpora of various languages

In the following, we show that the network is able to learn to parse correctly the sentences related to home scenarios. In particular, similar networks (with the same hyper-parameters) can learn to map sentences to predicates in sixteen corpora. The hyper-parameters were taken from previous experiments and were not optimized for these new corpora. This demonstrates the robustness of this neural parser. Moreover, it shows that, even with all the particularities of each language (discussed in subsection IV-D), a generic neural model is able to learn each of them without being adapted and despite having a random recurrent network as it core part.

This offers one more clue to reject Chomsky's suggestion that children are born with a preprogrammed Universal Grammar in their brain which would allow them to acquire language just by tuning the parameters of a particular language based on a limited exposure [35].

An ESN of 75 neurons is able to learn a set of 21 sentence-predicate pairs in any of these languages (i.e. reproduce exactly the same predicates for each sentence), but Hindi corpora. For Hindi we obtain one error with 75 neurons[6]. Conversely, a network of 50 units is able to learn any of English, French, Catalan, Persian or Portuguese corpus. The difference of units needed is not surprising since some language specificities may be more difficult than others to learn. This is due to the fact that the dataset is tiny and highly variable.

We did not investigate generalization concerning these corpora, even if we already demonstrated that the same network could generalize on both English and French at the same time [29]. However, we focused on the flexibility of the meaning representations, and in particular in robot home scenarios. Furthermore, the corpora are of tiny size (only 21 sentences for each language) with a high variability in their structure. Consequently, each sentence is nearly unique in its structure (and its use of function words), which does not enable generalization as shown in previous studies.

In the following, we give some remarks regarding languages specificities to show why it should be difficult to have a generic system that could learn all of them.

### B. General remarks

All sentences in different languages correspond to the translation from English sentences, but there are some specificities in each language which make the predicates not a direct translation word by word[7]. For instance, in the German sentence "geh an Telefon ; geh an das Telefon" ("answer phone ; answer the phone"), the "an" is not present in the English predicate because it is a "verb particle" (a German specificity). In a similar way, for the Spanish sentence "toma

[6]For the sentence-predicate pair #2. "dho javaab phon;phon ka javaab dho".

[7]As one can see in the corpora, one amazing word which is very similar across languages is "tomato". This is probably because tomatoes come from South America, thus many languages have borrowed the word *tōmatl* from the Nahuatl language, an Uto-Aztecan language.

el telefono y trae -me -lo" ("get the phone and bring it to me"): the direct translation of "to me" would be "a mi", but it is more natural in Spanish to attach "-me" to the verb. When *grammatical markers* such as "-me" are used in the meaning representation, we use "-" to indicate it is not a word by itself but a prefix or suffix, and the direction indicates to which word this marker should be attached: i.e. "prefix- word -suffix".

Some actions or verbs appear in two words: this is for instance the case in English for "switch off" (which is usually one word in other languages) and in Marathi for "kar saf" (*clean*). In such cases, one of the part of the verb was considered as the 1st argument in the meaning structure: "switch *off* light", here *off* is the first argument, and "light" is shifted as the 2nd argument. This does not prevent the neural model to learn this particular case and demonstrate that flexible meaning representations can be learned. In Marathi, the meaning is "kar saf table" and the sentence was "table saf kar": the order of the words composing the verb was different. This is due to the coherency of word order in this language.

Some additional features could be added to allow a more precise way of coding the predicates. For instance when two words should be assigned the same role – consider the action "switch off" –, a more complex coding of predicates should be used. As we will see in the Results section, such particularity does not prevent the model to learn all the English corpus. In future versions of the parser, we could add a particular sign like "_" linking the two words that would indicate that these two words should be learnt to be having the same role.

### C. Comparing English and German constructions

In subsection IV-C, one can see that some English constructions are equivalent for different sentence-meaning pairs. It is obviously the case two times in the corpus: for sentence-meanings pairs 1, 2, 3 and 4: "W X ; W the X"; and sentence-meaning pairs 9 and 17: "W X Y ; W X the Y". This means that the model only needs to learn one of the following sentences to be able to "generalize" to the other sentences:

1) open door ; open the door
2) answer phone ; answer the phone
3) water plants ; water the plants
4) clear table ; clear the table

Conversely this is not the case for the equivalent German sentences:

1) öffne Tür ; öffne die Tür
2) geh an Telefon ; geh an das Telefon
3) gieße Pflanzen ; gieße die Pflanzen
4) leere Tisch ; leere den Tisch

Which produce the following constructions:

1) W X ; W die X
2) W X Y ; W X das Y
3) W X ; W die X
4) W X ; W den X

In this German subsample of the corpus, learning sentence 1 or 3 would be enough to "generalize" to the other. We see that even if two languages are close in the family of languages, they are some differences in theirs sentence structures.

### D. Specific remarks about Hindi

In the Hindi corpus, while using a different word order in the predicates than English, the coherency is kept. As we see, a user creating a training corpus in its own language could use a particular word order for the predicates. It should be noted that, English and Hindi do not use the same general word order: English is a "SVO" language and Hindi is a "SOV" language[8]. This is one reason why the word order in predicates for Hindi seemed more adequate in the above mentioned way.

In the following, we illustrate the difference in the definition of predicates between English and Hindi. Considering the word order of predicates in English as a reference, Hindi word order of predicate where 1st, 3rd, 2nd arguments. Here are some examples of English predicate word order:

- sentence #10: make me some tea
- predicate #10: make tea me {*verb direct-object indirect-object*}
- sentence #18: find the bottle of water
- predicate #18: find bottle, bottle water {*verb direct-object, direct-object object-complement*}

In comparison, here are the corresponding examples of Hindi predicate word order:

- sentence #10: mujhay thoda chai banao
- predicate #10: banao mujhay chai {*verb indirect-object direct-object*}
- sentence #18: paani ka bothal doondo
- predicate #18: doondo bothal, paani bothal {*verb direct-object, object-complement direct-object*}

As one can see, in sentence #10 the order of direct-object and indirect-object are reversed in Hindi compared to English. Similarly, in sentence #18 direct-object and object-complement are reversed.

### E. Flexibility of the meaning representation learnt

Previously we reported that the system could learn chronological ordering. We showed that for sequential motor action, it could organize the predicate in chronological order [2]. An example is shown in the subsection II-D about the ROS module.

In the corpora proposed in this study, the meaning representations used were very variable and are of different nature. We have seen in the sentence examples given that predicates could be of different forms: action, entity localisation, entity modifier, conditional ("if"), ... Moreover, some details (i.e. some arguments in a predicate) could be omitted when not

[8]In linguistics, people usually refer to the main word order in a language by attributing one of the 6 possible word orders of the *Subject*, the *Verb* and the *Object* (SVO, SOV, VSO, VOS, OSV, OVS). English, Mandarin, Malay or Latin languages are SVO for instance.

necessary: this doesn't prevent the neural model from learning. The ability of the neural model to learn such different representations, in several different languages, shows that no particular output seems to be required. For instance, for the Hindi corpus, an organisation of the elements in the predicates was following a VOS (verb-object-subject) structure instead of a VSO structure like in English corpus (see subsection V-D on Hindi corpus for more details). It seems that as long as the output representation is coherent it can be learnt by the neural parser.

### *F. Ambiguous constructions and conflicts*

Some sentences could be ambiguous: they have several possible interpretations. This is the case with the sentence "The girl is looking at the man with a telescope". The telescope could be held by the girl or by the man. This ambiguity means that there are two possible syntactic trees that could be found by a parser. A similar kind of ambiguity can happen with constructions, when a sentence structure can have two possible meanings. In [3] we have shown that the neural parser is able to output the two possible answers at the same time.

There are some other cases which are difficult (or even impossible to learn) by the current system because of some exceptions that conflict with other sentence structures. For instance, this sentence-meaning pair in Hindi (and its corresponding pair in English):

- #2. dho javaab phon ; phon ka javaab dho
- (answer phone ; answer the phone)

is kept as it is because the use of "dho" here is not the common way. Usually "dho" is used like this:

- #3. dho paani paudhon;paudhon ko paani dho
- (water plants ; water the plants)

Due to this exception, this prevents the learning of another sentence-meaning pair, as a collateral side effect:

- #18. doondo bothal, paani bothal ; paani ka bothal doondo
- (find bottle, bottle water ; find the bottle of water)

Here is another special case of sentence-meaning pair which could have impaired the learning, but it hasn't:

- #17. bujhaa dho batthi ; batthi bujhaa dho
- (switch off light ; switch off the light)

These examples demonstrate that the system is able to learn some exceptions but not the ones that conflict with the regular sentence-meaning pairs.

### *G. A truly generic neural parser?*

When talking about languages with people at an open air, you may have heard "Chinese language has no grammar.". Even if this may have been claimed by some linguists, we were previously wondering if our neural parser would be ever able to parse such a language. In fact, as we have shown in this study, Mandarin has enough grammatical markers to allow the model to learn the sentence-meaning pairs.

We believe that it is a particularly interesting result, because the system was not adapted or particularly tuned to process specifically the Mandarin language. As it was not tuned to process the other 14 languages. The current study thus demonstrates the capability of this neural parser to be truly generic across languages and reinforce the language learning assumptions it is based on.

## VI. Discussion

We proposed an approach that allows people to use natural language when interacting with robots. Our architecture enables to train a sentence parser easily on home scenarios: no parsing grammar has to be defined a priori, it only relies on the particular corpus used for training. Moreover, we showed that this neural parser is not limited to one language, but was able to learn corpora in 15 different languages of different parts of the world and of different linguistic families. This demonstrates that this neural parser is not tuned for a particular language, and that it is a flexible tool for studies in Human-Robot Interaction across the world.

In future work, we plan to merge this neural parser to a new speech recognition system that would employ the syntactic reanalysis model of Twiefel et al. [36] in order to overcome the issues of non grammatical sentences produced by speech recognizers. This model also uses the concept of structures of sentence.

The way to write predicates (left hand of each line) is fairly intuitive and can be done without prior knowledge of a predefined structure like that of Robot Control Language (RCL) commands [37], which imposes a tree representation structure which is too focused on a particular robotic task. In other words, we propose to let the users define which meaning/predicate representation they want to use. For instance, some words like "on" can be used in the meaning representation when needed (e.g. "clean table, put mug on table ; clean the table and put the mug on it"), and can be discarded when there is no ambiguity and thus not necessary (e.g. "search recipe internet, recipe tiramisu ; search the recipe of tiramisu on internet"). Accordingly, one can adapt the meaning representation to any kind of "series of slots" – or distributed representations – as soon as they are consistent with each other. This neural model is not committed to predicates, and our architecture could be used to learn other type of representations.

A way to obtain distributed coherent representations would be to use multimodal, grounded, and embodied representations. For instance, a robot performing some actions could use its proprioception representation of the structure of the action as a teacher for the output of the neural parser. Conversely, it could use its visual representations of the properties of an object (e.g. *small* and *pink*) as inputs instead of the words "small" and "pink" of sentence 19.

To frame it in a more neurobiological framework, in Hinaut et al. (2013) [3] we proposed that the connections from the reservoir to the outputs could model the projections from a subpart of Broca's area to the input of the basal ganglia (i.e. the striatum). As the basal ganglia receive projections from nearly all cortical areas, it is a good place to share multimodal representations. Moreover, the reservoir

neural activity is itself using distributed "mixed-selectivity" representations [7][38]. Additionally, even if the model is rather simple – the core part is a random recurrent network – it is able to learn sentence-meaning associations in fifteen different languages. Not many models have this feature but it is an important (if not compulsory) one for models aiming at modelling sentence processing and language acquisition [39][25]. Some future experiments could show if this high flexibility of output, would enable the model to adapt across time to changing output structures: this could help in modelling developmental language acquisition. We believe that such a model could help understand how, not one language, but how any language could be acquired, along with their multimodal, grounded and embodied representations, with a loosely predefined architecture.

This claim on grounding may seem too strong, that is why we would like to provide some intuitive speculations of why it would work. First, the model can be adapted to different kinds of outputs like collection of"slot-structures" inspired from predicates. Actually, one could define these "predicate-like" structures in such a way that it corresponds to a tree structure when explored in a "depth-first" way: each non-terminal node would correspond to the first argument/slot of each predicate. We believe that using such predicate-like structures allows the users to flexibly define their own structured outputs, and adapt them to their cognitive system. Secondly, the training in the reservoir computing framework is based on a similar idea as that of Support Vector Machines: the inputs are projected into a non-linear and high-dimensional space from which one hopes to find a linear separation to a given problem. In the current paper, we are using symbolic outputs, but intuitively one can see that such ESN could learn as well the task by finding a linear relation with distributed outputs that are orthogonal with each other. Additionally, one can assume that it should be fairly the same for non-orthogonal distributed outputs, as the reservoir (i.e. the recurrent layer) in itself has correlated unit activations. Since such multimodal, embodied or grounded outputs have a particular latent structure, the reservoirs states should be rich enough to enable a linear mapping to be found. This is even more true if equivalent multimodal, embodied or grounded states are provided as input to the reservoir.

## ACKNOWLEDGMENT

We thank Ikram Chraibi Kaadoud, Pramod Kaushik, Louis Devers, and Iris Wieser for their ideas of sentences for the home corpora. We also greatly thank Francisco Cruz, Paul Guérin, Olatz Pampliega, Miguel López Cuiña, Thalita Firmo Drumond, Bhargav Teja Nallapu, Antoaneta Genova, Océane Plassart, William Schueller, Giulia Fois, Mohammad Ali Zamani, Chandrakant Bothe, Hwei Geok Ng for their translations in different languages of the 21 sentences-meaning pairs. We thank Remya Sankar, Bhargav Teja Nallapu and Fabien Benureau for their (very) useful feedback.

## REFERENCES

[1] F. Cruz, J. Twiefel, S. Magg, C. Weber, and S. Wermter. Interactive reinforcement learning through speech guidance in a domestic scenario. In *Proc. of IJCNN*, pages 1–8. IEEE, 2015.

[2] 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*, 8, 2014.

[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*, 8(2):e52946, 2013.

[4] P. Dominey. Complex sensory-motor sequence learning based on recurrent state representation and reinforcement learning. *Biological cybernetics*, 73(3):265–274, 1995.

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

[6] P. Werbos. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560, 1990.

[7] M. Rigotti, O. Barak, M. Warden, X. Wang, N. Daw, E. Miller, and S. Fusi. The importance of mixed selectivity in complex cognitive tasks. *Nature*, 497(7451):585–590, 2013.

[8] Timothy Brick and Matthias Scheutz. Incremental natural language processing for hri. In *Human-Robot Interaction (HRI), 2007 2nd ACM/IEEE International Conference on*, pages 263–270. IEEE, 2007.

[9] David Schlangen and Gabriel Skantze. A general, abstract model of incremental dialogue processing. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 710–718. Association for Computational Linguistics, 2009.

[10] S. Harnad. The symbol grounding problem. *Physica D: Nonlinear Phenomena*, 42(1-3):335–346, 1990.

[11] F. van der Velde. Communication, concepts and grounding. *Neural networks*, 62:112–117, 2015.

[12] T. Taniguchi, T. Nagai, T. Nakamura, N. Iwahashi, T. Ogata, and H. Asoh. Symbol emergence in robotics: a survey. *Advanced Robotics*, 30(11-12):706–728, 2016.

[13] X. Hinaut and S. Wermter. An incremental approach to language acquisition: Thematic role assignment with echo state networks. In *Proc. of ICANN 2014*, pages 33–40, 2014.

[14] K. Caluwaerts, M. D'Haene, D. Verstraeten, and B. Schrauwen. Locomotion without a brain: physical reservoir computing in tensegrity structures. *Artificial life*, 19(1):35–66, 2013.

[15] J. Burms, K. Caluwaerts, and J. Dambre. Reward-modulated hebbian plasticity as leverage for partially embodied control in compliant robotics. *Frontiers in neurorobotics*, 9:9, 2015.

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

[17] J. Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

[18] R. Miikkulainen. Subsymbolic case-role analysis of sentences with embedded clauses. *Cognitive Science*, 20(1):47–73, 1996.

[19] T. Mikolov, S. Kombrink, L. Burget, J. Černockỳ, and S. Khudanpur. Extensions of recurrent neural network language model. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 5528–5531. IEEE, 2011.

[20] S. Frank. Strong systematicity in sentence processing by an echo state network. In *International Conference on Artificial Neural Networks*, pages 505–514. Springer, 2006.

[21] M. Tong, A. Bickett, E. Christiansen, and G. Cottrell. Learning grammatical structure with echo state networks. *Neural networks*, 20(3):424–432, 2007.

[22] G. Marcus, S. Vijayan, S. Rao, and P. Vishton. Rule learning by seven-month-old infants. *Science*, 283(5398):77–80, 1999.

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

[24] A. Goldberg. Constructions: a new theoretical approach to language. *Trends in cognitive sciences*, 7(5):219–224, 2003.

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

[26] E. Bates and B. MacWhinney. Competition, variation, and language learning. *Mechanisms of language acquisition*, pages 157–193, 1987.

[27] P. Dominey, M. Hoen, and T. Inui. A neurolinguistic model of grammatical construction processing. *Journal of Cognitive Neuroscience*, 18(12):2088–2107, 2006.

[28] X. Hinaut, J. Twiefel, and S. Wermter. Recurrent neural network for syntax learning with flexible predicates for robotic architectures. In *Proc. of the IEEE Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*. IEEE, 2016.

[29] 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.

[30] D. Jurafsky and J. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition.* Pearson International, 2nd edition, 2009.

[31] X. Hinaut, J. Twiefel, M. Borghetti Soares, P. Barros, L. Mici, and S. Wermter. Humanoidly speaking – learning about the world and language with a humanoid friendly robot. In *IJCAI Video competition, Buenos Aires, Argentina*. https://youtu.be/FpYDco3ZgkU, 2015.

[32] J. Twiefel, X. Hinaut, M. Borghetti, E. Strahl, and S. Wermter. Using Natural Language Feedback in a Neuro-inspired Integrated Multimodal Robotic Architecture. In *Proc. of RO-MAN*, New York City, USA, 2016.

[33] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[34] M. Lukoševičius. A practical guide to applying echo state networks. In *Neural Networks: Tricks of the Trade*, pages 659–686. Springer, 2012.

[35] Chomsky N. *Aspects of the Theory of Syntax.* MIT Press., 1965.

[36] J. Twiefel, X. Hinaut, and S. Wermter. Syntactic reanalysis in language models for speech recognition. In *Proc. of the IEEE Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*. IEEE, 2017.

[37] K. Dukes. Semeval-2014 task 6: Supervised semantic parsing of robotic spatial commands. *SemEval 2014*, page 45, 2014.

[38] P. Enel, E. Procyk, R. Quilodran, and P. Dominey. Reservoir computing properties of neural dynamics in prefrontal cortex. *PLoS Comput Biol*, 12(6):e1004967, 2016.

[39] F. Chang. Symbolically speaking: A connectionist model of sentence production. *Cognitive science*, 26(5):609–651, 2002.