Recurrent Babbling:

evaluating the acquisition of grammar from limited input data

Ludovica Pannitto, Aurélie Herbelot

November 19-20 2020 @ CoNLL

CIMeC - University of Trento

Do RNNs learn grammar?

A popular question, relating to **productivity** and **compositionality**[1].

We propose that the evaluation of RNN grammars should be widened to include:

- the effect of the **type of input data** fed to the network
- the **theoretical paradigm** used to analyse its performance

[1] "Linguistic generalization and compositionality in modern artificial neural networks" (Baroni 2020)

Our Setup

- vanilla **char-LSTM** trained on a limited amount of **child-directed language**: CHILDES, subtitles (PG) and Simple English Wikipedia

Our Setup

- vanilla **char-LSTM** trained on a limited amount of **child-directed language**: CHILDES, subtitles (PG) and Simple English Wikipedia
- introduce a methodology to evaluate the **distribution of grammatical items**, focusing on the network's generated output - its *babbling*

Our Setup

- vanilla **char-LSTM** trained on a limited amount of **child-directed language**: CHILDES, subtitles (PG) and Simple English Wikipedia
- introduce a methodology to evaluate the **distribution of grammatical items**, focusing on the network's generated output - its *babbling*
- explore the **interaction** between meaning representations and the abstraction abilities of the network

Our Setup

- vanilla **char-LSTM** trained on a limited amount of **child-directed language**: CHILDES, subtitles (PG) and Simple English Wikipedia
- introduce a methodology to evaluate the **distribution of grammatical items**, focusing on the network's generated output - its *babbling*
- explore the **interaction** between meaning representations and the abstraction abilities of the network

The study is conducted on **English**.

Framework

How much language (Λ) can be learnt from a certain level of computational complexity (C) with a certain type of data (I)?

$$C \times I \xrightarrow{a} \Lambda \tag{1}$$

Framework

How much language (Λ) can be learnt from a certain level of computational complexity (C) with a certain type of data (I)?

$$C \times I \xrightarrow{a} \Lambda \quad (1)$$

All aspects of the equation are of paramount importance in linguistic discussion:

complexity of the learning mechanism C - LSTMs can be seen as domain-general attention and memory mechanisms, without any explicitly hard-coded grammatical knowledge.

Framework

How much language (Λ) can be learnt from a certain level of computational complexity (C) with a certain type of data (I)?

$$C \times I \xrightarrow{a} \Lambda \tag{1}$$

All aspects of the equation are of paramount importance in linguistic discussion:

complexity of the learning mechanism C - LSTMs can be seen as domain-general attention and memory mechanisms, without any explicitly hard-coded grammatical knowledge.

quality and quantity of the stimuli I - stimuli differ in quantity and quality

Framework

How much language (Λ) can be learnt from a certain level of computational complexity (C) with a certain type of data (I)?

$$C \times I \xrightarrow{a} \Lambda \tag{1}$$

All aspects of the equation are of paramount importance in linguistic discussion:

complexity of the learning mechanism C - LSTMs can be seen as domain-general attention and memory mechanisms, without any explicitly hard-coded grammatical knowledge.

quality and quantity of the stimuli I - stimuli differ in quantity and quality

language Λ - status of *lexicon* and *syntax*

Features of λ: *Catenae*

We choose a representation which makes the least possible assumptions on the acquisition process and on the content of the generated language, and is at the same time **flexible** and **computationally tractable**.

[2] "Catenae: Introducing a novel unit of syntactic analysis" (Osborne, Putnam, and Groß 2012)

[3] *Constructions at work: The nature of generalization in language* (Goldberg 2006)

Features of λ: *Catenae*

We choose a representation which makes the least possible assumptions on the acquisition process and on the content of the generated language, and is at the same time **flexible** and **computationally tractable**.

Catenae[2], are characterized as fundamental **meaning-bearing units**, in line with the theoretical tenets of constructionist theories[3], thus being ideal candidates for populating our lexicon (or *Constructicon*).

[2] "Catenae: Introducing a novel unit of syntactic analysis" (Osborne, Putnam, and Groß 2012)
[3] *Constructions at work: The nature of generalization in language* (Goldberg 2006)

Catenae

Definition of *Catena*:

“a word, or a combination of words which is continuous with respect to dominance”

Catenae

Definition of *Catena*:

"a word, or a combination of words which is continuous with respect to dominance"

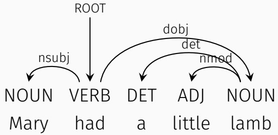

Figure 1: Dependency representation for the sentence: *Mary had a little lamb*

- Mary had lamb
- had a lamb
- little lamb
- Mary had NOUN
- nsubj VERB dobj

Definition of *Catena*:

"a word, or a combination of words which is continuous with respect to dominance"

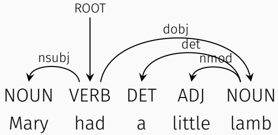

Figure 1: Dependency representation for the sentence: *Mary had a little lamb*

- Mary had lamb
- had a lamb
- little lamb
- Mary had NOUN
- nsubj VERB dobj

The number and composition of *catenae* depends on **how elements are arranged** in the structure of the dependency tree.

Main questions

Q1: To what extent is the network able to generate **new** language?

- We expect the network to reproduce the **statistical regularities** of the input, we further investigate what kind of regularities are acquired and how do the language models differ.

Main questions

Q1: To what extent is the network able to generate **new** language?

- We expect the network to reproduce the **statistical regularities** of the input, we further investigate what kind of regularities are acquired and how do the language models differ.

Q2: On what conditions is the network able to generalize its *grammatical* knowledge?

- We can state that the network has learned some grammar once it is able to use an acquired pattern in a **productive** and **creative** way.
- We expect this generalization ability to evolve during training and the **distributional properties** of patterns to be in relation with the grammatical abilities of the network at various stages of learning.

Main questions

Q1: To what extent is the network able to generate **new** language?

- We expect the network to reproduce the **statistical regularities** of the input, we further investigate what kind of regularities are acquired and how do the language models differ.

Q2: On what conditions is the network able to generalize its *grammatical* knowledge?

- We can state that the network has learned some grammar once it is able to use an acquired pattern in a **productive** and **creative** way.
- We expect this generalization ability to evolve during training and the **distributional properties** of patterns to be in relation with the grammatical abilities of the network at various stages of learning.

Pipeline

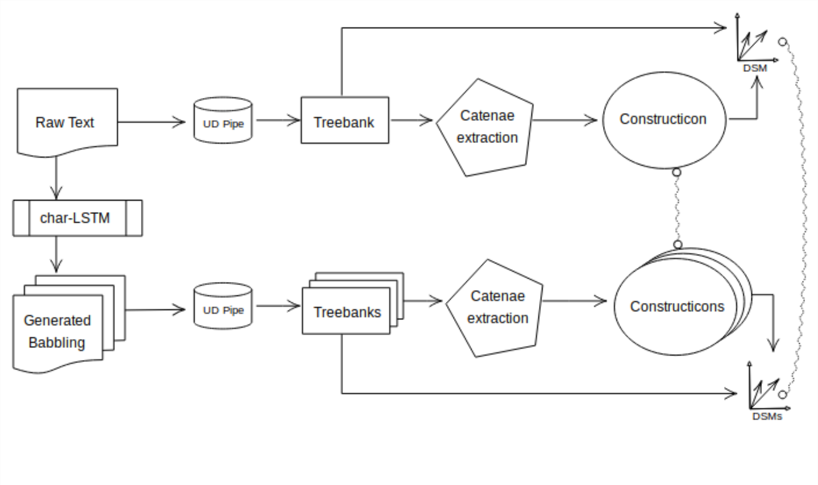

Figure 2: A summary of the work pipeline

Catenae extraction

catena	frequency	mi
largest mi		
@nsubj @root	294.59K	633.93K
_DET _NOUN	189.97K	552.32K
_VERB @obj	190.72K	520.82K
_PRON _VERB	271.44K	503.17K
@nsubj _AUX @root	129.60K	478.86K
smallest mi		
_PRON @nsubj	17.50K	-35.54K
@root @nsubj	27.61K	-34.89K
@nsubj _PRON	11.63K	-30.47K
_VERB @nsubj	12.79K	-26.82K
_AUX _PRON	15.75K	-26.67K

Table 1: Examples of *catenae* extracted from CHILDES. Largest and smallest mutual information are reported, in top and bottom tier of the table respectively.
Part of Speech are prefixed by “_” and syntactic relations are prefixed by “@”

Q1: What do ANNs approximate?

- For each corpus (Input I), we selected the best hyperparameters through Bayesian Optimization and built a language model (Best Model BM)

Q1: What do ANNs approximate?

- For each corpus (Input I), we selected the best hyperparameters through Bayesian Optimization and built a language model (Best Model BM)
- We trained again the LSTM with the found hyperparameters, and saved a model every 5 epochs of training (5, 10, ...45)

Q1: What do ANNs approximate?

- For each corpus (Input I), we selected the best hyperparameters through Bayesian Optimization and built a language model (Best Model BM)
- We trained again the LSTM with the found hyperparameters, and saved a model every 5 epochs of training $(5, 10, ...45)$
- We generated new text from each model, obtaining *babblings* of a size comparable to the input I

Q1: What do ANNs approximate?

- For each corpus (Input I), we selected the best hyperparameters through Bayesian Optimization and built a language model (Best Model BM)
- We trained again the LSTM with the found hyperparameters, and saved a model every 5 epochs of training (5, 10, ...45)
- We generated new text from each model, obtaining *babblings* of a size comparable to the input I
- We processed the input and each *babbling* and extracted *catenae* from them

Q1: What do ANNs approximate?

- For each corpus (**Input I**), we selected the best hyperparameters through Bayesian Optimization and built a language model (**Best Model BM**)
- We trained again the LSTM with the found hyperparameters, and saved a model every 5 epochs of training (5, 10, ...45)
- We generated new text from each model, obtaining *babblings* of a size comparable to the input **I**
- We processed the input and each *babbling* and extracted *catenae* from them

We end up with sets of *catenae* for **the input**, the **best model**, each **babbling stage**.

Q1: What do ANNs approximate?

We evaluated Spearman ρ among the top 10K *catenae* extracted from the input and from each *babbling* stage produced by the LSTM.

Q1: What do ANNs approximate?

We evaluated **Spearman** ρ among the top 10K *catenae* extracted from the input and from each *babbling* stage produced by the LSTM.

Our analysis shows that the network has acquired statistical **regularities at the level of grammatical patterns**, and is able to use them productively to generate **novel** language fragments that **adhere to the same distribution as the input**.

Q1: What do ANNs approximate?

We evaluated **Spearman** ρ among the top 10K *catenae* extracted from the input and from each *babbling* stage produced by the LSTM.

Our analysis shows that the network has acquired statistical **regularities at the level of grammatical patterns**, and is able to use them productively to generate **novel** language fragments that **adhere to the same distribution as the input**.

Catenae extracted from babblings almost perfectly correlate with those extracted from the same input, but correlation values are quite **loose for out-of-domain pairs**.

Q1: What do ANNs approximate?

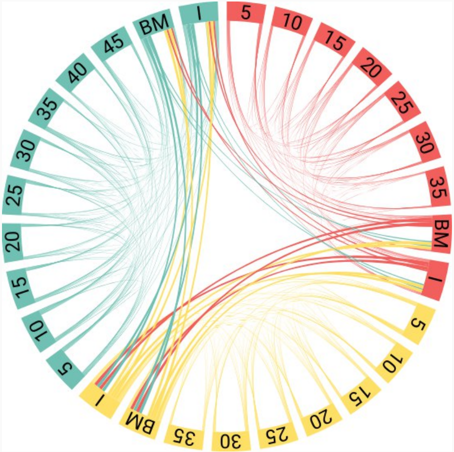

Figure 3: The thickness of the connections is **inversely** proportional to correlation. OpenSubtitles is shown in green on the left of the plot, CHILDES in red in the top right and Simple Wikipedia in yellow at the bottom.

Q2: Meaning and abstraction

The case of *[SBJ V OBJ OBJ2]* [4]

The meaning of the ditransitive pattern emerges from its strong association with ***give*** in child-directed speech: part of the meaning of *give* remains attached to the construction.

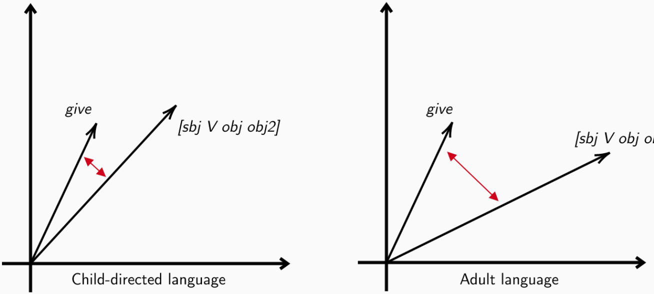

Figure 4: The network is supposed to capture stereotypical instances at early stages of learning and the productivity of the pattern will increase during training

Q2: Meaning and abstraction

- We restrict to *catenae* from CHILDES composed by 2 or 3 elements

Q2: Meaning and abstraction

- We restrict to *catenae* from CHILDES composed by 2 or 3 elements
- We built distributional vector space models for the input and each babbling stage, using the selected *catenae* both as targets and as contexts: two *catenae* are considered to co-occur if they are present in the same sentence

Q2: Meaning and abstraction

- We restrict to *catenae* from CHILDES composed by 2 or 3 elements
- We built distributional vector space models for the input and each babbling stage, using the selected *catenae* both as targets and as contexts: two *catenae* are considered to co-occur if they are present in the same sentence
- We extracted pairs of *catenae* at different level of abstraction: i.e., (*the dog*, *_DET dog*), (*the dog*, *the _NOUN*), (*the dog*, *_DET _NOUN*), (*_DET dog*, *_DET _NOUN*) are all legitimate pairs for our analysis

Q2: Meaning and abstraction

- We restrict to *catenae* from CHILDES composed by 2 or 3 elements
- We built distributional vector space models for the input and each babbling stage, using the selected *catenae* both as targets and as contexts: two *catenae* are considered to co-occur if they are present in the same sentence
- We extracted pairs of *catenae* at different level of abstraction: i.e., (*the dog*, *_DET dog*), (*the dog*, *the _NOUN*), (*the dog*, *_DET _NOUN*), (*_DET dog*, *_DET _NOUN*) are all legitimate pairs for our analysis
- For each pair of *catenae*, we evaluated the **difference in cosine similarity** between the model obtained from the first and last snapshot from training, and compared it to their similarity in the DSM obtained from the input.

Q2: Meaning and abstraction

- We restrict to *catenae* from CHILDES composed by 2 or 3 elements
- We built distributional vector space models for the input and each babbling stage, using the selected *catenae* both as targets and as contexts: two *catenae* are considered to co-occur if they are present in the same sentence
- We extracted pairs of *catenae* at different level of abstraction: i.e., (*the dog*, *_DET dog*), (*the dog*, *the _NOUN*), (*the dog*, *_DET _NOUN*), (*_DET dog*, *_DET _NOUN*) are all legitimate pairs for our analysis
- For each pair of *catenae*, we evaluated the **difference in cosine similarity** between the model obtained from the first and last snapshot from training, and compared it to their similarity in the DSM obtained from the input.

Q2: Meaning and abstraction

Given pairs (cat_1, cat_2) with cat_1 being a less abstract instance of cat_2, our hypothesis is that *catenae* (i.e., cat_2) that underwent the highest shifts during training were those showing **intermediate levels of similarities in the input distributional space**.

Q2: Meaning and abstraction

Given pairs (cat_1, cat_2) with cat_1 being a less abstract instance of cat_2, our hypothesis is that *catenae* (i.e., cat_2) that underwent the highest shifts during training were those showing **intermediate levels of similarities in the input distributional space**.

- pairs with very **high input similarity** are unlikely to exhibit abstraction: the *catena* that is part of the *Constructicon* is the least abstract one, and there is **no need** for the more abstract category
- **low similarity** pairs, on the other hand, may simply contain **unrelated *catenae***

Q2: Meaning and abstraction

cat_1	cat_2	input	5	10	...	30	35	shift
a minute	a _NOUN	0.28	0.71	0.51	...	0.37	0.34	0.37
a minute	a @root	0.13	0.49	0.37	...	0.22	0.20	0.30
you _VERB it	_PRON @root @expl	0.10	0.46	0.28	...	0.17	0.21	0.25
you _VERB you	you _VERB @iobj	0.28	0.68	0.56	...	0.42	0.43	0.25
we can _VERB	_PRON can @root	0.51	0.79	0.74	...	0.61	0.57	0.22

Table 2: Pairs of *catenae* (cat_1, cat_2), their cosine similarity in the space obtained from CHILDES and in the spaces obtained from intermediate *babbling* stages.
The last column shows the difference between cosine similarity at epoch 5 and cosine similarity at epoch 35.

Q2: Meaning and abstraction

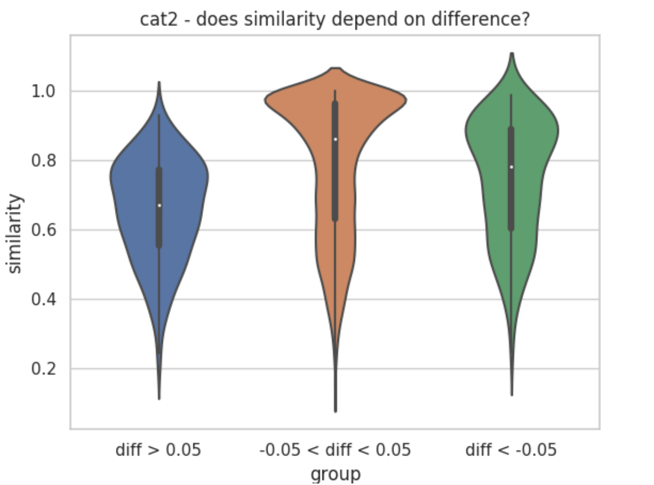

Figure 5: Distribution of average cosine similarities for the three groups of cat_2, showing low, intermediate and high average shifts respectively.

Wrap-up

ANNs approximate the **distribution of constructions** at a quite refined level, even when trained over a bare 3M words from the CHILDES corpus.

[5] "Distributional semantics in linguistic and cognitive research" (Lenci 2008)
[6] *Explain me this: Creativity, competition, and the partial productivity of constructions* (Goldberg 2019)

Wrap-up

ANNs approximate the **distribution of constructions** at a quite refined level, even when trained over a bare 3M words from the CHILDES corpus.

We can follow paths of abstraction by putting our **grammar formalism** in a vector space.

[5] "Distributional semantics in linguistic and cognitive research" (Lenci 2008)
[6] *Explain me this: Creativity, competition, and the partial productivity of constructions* (Goldberg 2019)

Wrap-up

ANNs approximate the **distribution of constructions** at a quite refined level, even when trained over a bare 3M words from the CHILDES corpus.

We can follow paths of abstraction by putting our **grammar formalism** in a vector space.

- no sharp distinction between **lexicon** and **grammar** → different items can therefore be compared, irrespective of their lexical nature
- no assumption about the **stability** of the constructicon → what is relevant for productivity at the earliest stages of learning might become superfluous later on
- all items are **form-meaning** pairs → i.e., constructions
- **distributional semantics** is used both as a quantitative tool and as a usage-based cognitive hypothesis[5] → in line with the view of constructions as "*invitations to form categories*"[6]

[5] "Distributional semantics in linguistic and cognitive research" (Lenci 2008)
[6] *Explain me this: Creativity, competition, and the partial productivity of constructions* (Goldberg 2019)

Thank you!