Does the success of deep neural network language processing mean – finally! – the end of theoretical linguistics?

Paul Smolensky & Jennifer Culbertson
Department of Cognitive Science & Department of Linguistics and English Language
Johns Hopkins University & University of Edinburgh

SIGNLL Conference on Computational Natural Language Learning
31 July 2015 Beijing
Q: Shouldn’t your DNNs for language processing have structured representations (e.g., tensor product representations *)?

* see below

A: Do you want

a. a pretty theory or
b. a system that works?

Obviously:

- a

- We already have 7.3 billion systems that work (as of July 2015).
- The burning question is:
  - How do they work? The goal is understanding.
Why might CoNLLers care about a Cognitive Science sketch?

- Progress in the study of human cognition/language has anticipated corresponding progress* in NLP in some notable cases
  * conceptual, not just technical, progress

  ➤ (Restricted) Boltzmann Machines [Hinton & Sejnowski 83; Smolensky 83]
  ➤ Back-prop [Rumelhart, Hinton & Williams 86]
  ➤ Bayes networks [Pearl’s cognitive theory of causation 88]
  ➤ Tensor methods [Dolan & Dyer 87; Smolensky 87, 90]
  ➤ Maxent
    ✦ general knowledge [Smolensky 83, 86]
    ✦ grammar [Harmonic Grammar: Legendre, Miyata & Smolensky 90]
The death of linguistic theory? A sketch from Cognitive Science++

II. Ling++: how to understand linguistic cognition in terms of structure processing

A. Representations
   1. Ling: Structural roles
   2. Ling++: Gradient structural roles

B. Grammars
   1. Ling: Universal wellformedness constraints, Con
   2. Ling++: SoftCon, Bayesian universal biases
   3. Limiting the power of language learning; innate linguistic knowledge?
      1. Argument from Universals* [not from Poverty of the Stimulus]
      2. New experimental evidence supporting *

I. NN++: Understandably implements Ling++

A. Neural networks with structured distributed vectorial representations
B. Understanding structure computation: (gradient) symbolic functions
C. Understanding encoding of structural wellformedness: Ling++ grammars

Upshot: Upgraded to Ling++, implemented in NN++, linguistics may lead us to deeper understanding of how to compute natural language (learning)
The death of linguistic theory? A sketch from Cognitive Science

II. Ling++: how to understand linguistic cognition in terms of structure processing
   A. Representations
      1. Ling: Structural roles
      2. Ling++: Gradient structural roles
   B. Grammars
      1. Ling: Universal wellformedness constraints, Con
      2. Ling++: SoftCon, Bayesian universal biases
      3. Limiting the power of language learning; innate linguistic knowledge?
         i. Argument from Universals* [not from Poverty of the Stimulus]
         ii. New experimental evidence supporting *
   I. NN++: Understandably implements Ling++
      A. Neural networks with structured distributed vectorial representations
      B. Understanding structure computation: (gradient) symbolic functions
      C. Understanding encoding of structural wellformedness: Ling++ grammars
Linguistic-theoretic semantics: elementary explanatory ingredients

1. Argument Structure:
   syntactic $\leftrightarrow$ semantic roles: $[L [\text{loves } M]] \leftrightarrow \text{loves}(L, M)$

2. Embedding/recursion:
   $\text{said}(J, \text{thinks}(K, \text{loves}(L, M)))$

3. Semantic (LF) operators
   $\neg, \exists, \text{if, } \diamond$
   a. Operator/variable binding $\exists x. P(x)$
   b. Operator scope $\neg P) \& Q \quad \neg(P \& Q)$

4. Compositionality
   $\neg, \& \quad P, Q$

5. $\beta$-reduction in $\lambda$ calculus
   $(\lambda x. B)A = B$ with every $x \to A$

6. Tree adjoining in TAG
The death of linguistic theory? A sketch from Cognitive Science

I. NN++: Implements Ling++
   A. Neural networks with structured distributed vectorial representations
   B. Computing (gradient) symbolic functions
   C. Encoding Ling++ grammars

II. Ling++
   A. Representations
      1. Ling: Structural roles
      2. Ling++: Gradient structural roles
   B. Grammars
      1. Ling: Universal wellformedness constraints, Con
      2. Ling++: SoftCon, Bayesian universal biases
      3. Limiting the power of language learning; innate linguistic knowledge?
         i. Argument from Universals* [not from Poverty of the Stimulus]
         ii. New experimental evidence supporting *

Upshot: Upgraded to Ling++, implemented in NN++, linguistics may lead us to deeper understanding of how to compute natural language (learning)
Figure 2. Fully local realization of symbol strings

This pattern of activation $AB$ realizing the string $AB$ can be contrasted with the pattern $BA$ realizing $BA$ (shown in Figure 2A). The decomposition of the new vector is shown in (13):

\[ A = A_1 + B_2 \]

\[ B = A_2 + C_1 \]

\[ C = r_1 \]

\[ D = r_2 \]

\[ E = A + B + C + D + E \]

\[ AB = A_1 + B_2 \]

\[ BA = A_1 + B_2 \]
Formalizing the Principles I: Representation and Processing

Section 1.2

\( B_2 = (0 0 0 0 0 0 1 0 0 0) \) is the realization of \( B \) in the role of second element.

Figure 2. Fully local realization of symbol strings

This pattern of activation \( AB \) realizing the string \( AB \) can be contrasted with the pattern \( BA \) realizing \( BA \) (shown in Figure 2A). The decomposition of the new vector is shown in (13):

\[
AB = A \otimes r_1 + B \otimes r_2 = \sum_x a_x \otimes r_x
\]

Fully local case (both \( A \) and \( r_1 \) are 1-hot)
Figure 4. Semilocal realization of symbol strings

Figure 4B simply rotates the two subpatterns so that in Figure 4C we may readily see the internal tensor product structure of the constituent vector $A_1$. As in the local case, it is simply the tensor product of the vector $A$ (vertical, left of the box) with a vector $B$.

$AB = A_1 + B_2$

$AB = A_1 + B_2$

$AB = A_1 + B_2$

$AB = A_1 + B_2$
Semi-local representation of letter strings

\[ AB = A \otimes r_1 + B \otimes r_2 = \sum x a_x \otimes r_x \]

Semi-local case (\( r_1 \) but not \( A \) is 1-hot)
Semi-local representation of trees

Recursive NNs:
Implicit role vectors are local

Socher, Manning & Ng 2010
Fully distributed representation of letter strings

\[ AB = A_1 + B_2 \]

Figure 5. Fully distributed realization of symbol strings

Independence assumption

In a connectionist realization of symbolic computation, the activation vectors realizing the atomic symbols, and those realizing the role vectors, are (linearly) independent.
In a connectionist realization of symbolic computation, the activation vectors realizing the atomic symbols, and those realizing the role vectors, are (linearly) independent.

A key idea of this work: distributed role vectors.
Formalizing the Principles I: Representation and Processing

Section 1.2

B.

\[ AB = A_1 + B_2 \]

C.

\[ A_1 = A \otimes r_1 \]

Independence assumption

In a connectionist realization of symbolic computation, the activation vectors realizing the atomic symbols, and those realizing the role vectors, are (linearly) independent.

Fully distributed representation of letter strings

\[ AB = A \otimes r_1 + B \otimes r_2 = \sum_x a_x \otimes r_x \]

Fully distributed case (neither \( r_1 \) nor \( A \) is 1-hot)

Every unit contributes to symbols in both positions.

Distributed role vector: A key idea of this work

\[ B_2 = B \otimes r_2 \]
Arg-struc similarities among verbs’ SYN/SEM-role vectors

All Vs are \( sem = \text{ROT}, \ syn = \) [Intrans Subj ~ Trans Obj] (“unaccusative”) except where noted \( TR \) or [Intrans Subj ~ Trans Subj] (“unergative”)
Reprise — Combinatorial structure in activation patterns

Friday, July 31, 15
Reprise — Combinatorial structure in activation patterns

Frodo + lives = Frodo lives

Macrostructure:
Symbolic Structures
Microstructure:
Unit Activations

c: Cognitive Science
Reprise — Constituent construction

**Filler pattern**

\[ \ast \]

**Role pattern**

\[ = \]

**Filler/role constituent pattern**

\[
\mathbb{M} s = \{f_k/r_k\} \\
\mathbb{m} s = \sum_k f_k \otimes r_k
\]
The death of linguistic theory? A sketch from Cognitive Science++

I. NN++: Implants Ling++
   A. Neural networks with structured distributed vectorial representations
   B. Computing (gradient) symbolic functions
   C. Encoding Ling++ grammars

II. Ling++
   A. Representations
      1. Ling: Structural roles
      2. Ling++: Gradient structural roles
   B. Grammars
      1. Ling: Universal wellformedness constraints, Con
      2. Ling++: SoftCon, Bayesian universal biases
      3. Limiting the power of language learning; innate linguistic knowledge?
         i. Argument from Universals* [not from Poverty of the Stimulus]
         ii. New experimental evidence supporting *

Can we understand what structure processing NN++ can do?
Yes.
The death of linguistic theory? A sketch from Cognitive Science

I.  NN++: Implements Ling++
   A.  Neural networks with structured distributed vectorial representations
   B.  Computing (gradient) symbolic functions

Basic observation: Primitive tree-constructing and tree-constituent-accessing functions are implementable as linear transformations — matrix multiplication.

So a linear network can compute the set of recursive functions that is the closure under composition of the primitive tree operations.

To implement a recursive function defined by primitive recursion equations, solve the corresponding recursion equations for matrices.
The death of linguistic theory? A sketch from Cognitive Science++

I. NN++: Implements Ling++
   A. Neural networks with structured distributed vectorial representations
      B. Computing (gradient) symbolic functions

   1. Ling++: Implement Ling: Structural roles Ling
      i. Limiting the power of language learning; innate linguistic knowledge?
         i. Argument from Universals* [not from Poverty of the Stimulus]
         ii. New experimental evidence supporting *
Basic operation of $\lambda$-calculus [function application]: $(\lambda x.B)A$

$$\Lambda = \lambda x. B$$

$\beta$-reduce$(\Lambda, A) = \left[ 1 + (A - x) \otimes x^+ \right] B$

Next: Substituting in a structure rather than an atom.
Tree Adjoining

Initial tree  \(t\)  

Auxiliary tree  \(a\)  

TA:  

\[
\begin{align*}
& t - A \otimes R_A \\
& + A \otimes [ R_\alpha \otimes R_A ] \\
& + a \otimes R_A \\
& - \alpha \otimes [ R_\alpha \otimes R_A ]
\end{align*}
\]

retain all of \(t\) unaffected by adjoining  

\[
\begin{align*}
a \cdot r_e^+ \equiv A \\
A^+ \cdot t \equiv R_A \\
t \cdot R_A^+ \equiv A \\
\alpha^+ \cdot a \equiv R_\alpha
\end{align*}
\]

root symbol of \(a\) ("A")  

location of \(A\) in \(t\)  

subtree \(A\) in \(t\)  

location of \(\alpha\) in \(a\)
The death of linguistic theory? A sketch from Cognitive Science++

I. NN++: Implements Ling++
   A. Neural networks with structured distributed vectorial representations
   B. Computing (gradient) symbolic functions
   C. Encoding Ling++ grammars

Ling++ grammars are Harmonic Grammars that reward or penalize the co-presence of 1–2 constituents within a single structure.

Key observation: These grammars can be straightforwardly implemented as connection matrices that generate, in the co-presence of the distributed representations of the target constituents, the specified contribution to the Harmony of the network state. This measure of wellformedness is maximized by activation-spreading dynamics in the network.

When appropriate stochastic dynamics are deployed, a Maxent distribution results, the log probability of a state being proportional to its Harmony.
Why do DNNs for language do as well as they do?

Are they approximating the kind of structural representation and processing characteristic of NN++ systems?

If DNNs were constructed to incorporate NN++ structural processing, would they do even better?
The death of linguistic theory? A sketch from Cognitive Science++

I. NN++: Implements Ling++
   A. Neural networks with structured distributed vectorial representations
   B. Computing (gradient) symbolic functions
   C. Encoding Ling++ grammars

II. Ling++
   A. Representations
      1. Ling: Structural roles
      2. Ling++: Gradient structural roles

   i. Argument from Universals* [not from Poverty of the Stimulus]
   ii. New experimental evidence supporting *
The death of linguistic theory? A sketch from Cognitive Science

I. NN++: Implements Ling++
   A. Neural networks with structured distributed vectorial representations
   B. Computing (gradient) symbolic functions
   C. Encoding Ling++ grammars

II. B. Grammars
   1. Ling: Universal wellformedness constraints, Con

   i. Argument from Universals* [not from Poverty of the Stimulus]
   ii. New experimental evidence supporting *
The death of linguistic theory? A sketch from Cognitive Science

I. NN++: Implements Ling++
   A. Neural networks with structured distributed vectorial representations
   B. Computing (gradient) symbolic functions
   C. Encoding Ling++ grammars

II. Ling++
   A. Representations
      1. Ling: Structural roles
      2. Ling++: Gradient structural roles
   B. Grammars
      1. Ling: Universal wellformedness constraints, Con
      2. Ling++: SoftCon, Bayesian universal biases — later
      3. Limiting the power of language learning; innate linguistic knowledge?
         i. Argument from Universals* [not from Poverty of the Stimulus]
         ii. New experimental evidence supporting *
N, Adj word order in the world’s languages

<table>
<thead>
<tr>
<th></th>
<th>N-Adj</th>
<th>Adj-N</th>
</tr>
</thead>
<tbody>
<tr>
<td>Num-N</td>
<td>17%</td>
<td>27%</td>
</tr>
<tr>
<td>N-Num</td>
<td>52%</td>
<td>*4%</td>
</tr>
</tbody>
</table>

*Particular non-harmonic pattern Adj-N, N-Num is disfavored

Substantive learning bias

Experimental results I: Adult group data

Do typological statistics correspond to any active on-line learning bias?

Learners of an artificial language, given two-word nonce-utterance examples (N with either Adj or Num), dominant order in each of 4 conditions = 70%.
Do typological statistics correspond to any active on-line learning bias?

Learners of an artificial language, given two-word nonce-utterance examples (N with either Adj or Num), dominant order in each of 4 conditions = 70%.
6-7-year-old kids:
Strongly adopt a harmonic order in all conditions
[tending to follow the dominant order of their condition’s Adj and N]

Greenberg (1968)’s U20: {Demonstrative Number Adjective Noun}:
Also N Dem Num Adj.
Trained with 2-word utterances, N + modifier, always N-initial.
Would learners favor English order among modifiers, or scope order?

Further evidence: U20

{Demonstrative Number Adjective Noun}:


Also N Dem Num Adj.

Trained with 2-word utterances, N + modifier, always N-initial
The death of linguistic theory? A sketch from Cognitive Science

I. Prior generates a non-uniform distribution over possible grammars in the predicted typology: provides an approach to counter-evidence.

Constraints (which are simply absent from Ling's entailed by explaining statistical typological generalizations ('[Culbertson, Smolensky, Wilson 13].

Learners resist violating universals Ling

B. Grammars

1. Ling: Universal wellformedness constraints, Con

2. Ling: SoftCon, Bayesian universal biases

3. Limiting the power of language learning; innate linguistic knowledge?
   i. Argument from Universals* [not from Poverty of the Stimulus]
   ii. New experimental evidence supporting *
Constraint bias and statistical universal U18 [theoretical]


To generate only harmonic orders (Num-Adj-N, N-Adj-Num), posit:
\[ Con = \{\text{HEAD-R, HEAD-L}\} \]

To admit typologically less-preferred Num-N-Adj, add NUM-L to SoftCon but with a bias against ‘use’ (non-zero weight) of specific-head constraints: Gaussian prior, with ‘strength’ (precision) \( \kappa \), that disfavors \( G_{\{w_c\}} \) the more \( w_{\text{NUM-L}} \) is > 0

To admit marginal Adj-N-Num, add NUM-R to Con, which incurs the specific-head penalty \( \kappa \) plus an additional penalty \( \lambda \); result:

\[
P_{\text{prior}}(G_{\{w_c\}}) \propto e^{-\kappa\left(w_{\text{NUM-L}}^2 + w_{\text{NUM-R}}^2\right)} e^{-\lambda w_{\text{NUM-R}}^2}
\]

Not all constraints in Con are created equal: substantive bias
Approach to explaining statistical typological patterns (‘universals’)

Friday, July 31, 15
In addition to word-order bias, there is the regularization bias:

Prior also favors $>0$ weight values (disprefers completely free variation resulting from all-0 weights)

- Gaussians with means greater than zero

Fitting the total prior distribution’s parameters to the experimental data, Bayes yields these predictions for the experiment:
Real-time learning bias in U18 [experimental]


- Preferred means pull weights further from zero than needed to match the input ⇒ regularization [line at 0.7 is rate in the input].

- Cost of doing so for specific-head constraint weights (NUM-L, NUM-R) impedes regularization.

- So get the most regularization for harmonic orders (conditions 1–2), then condition 3, and least for U18-violating condition 4 [see red ‘Mean’ bars]
The death of linguistic theory? A sketch from Cognitive Science++

I. NN++: Implements Ling++
   A. Neural networks with structured distributed vectorial representations
   B. Computing (gradient) symbolic functions
   C. Encoding Ling++ grammars

II. Ling++
   A. Representations
      1. Ling: Structural roles
      2. Ling++: Gradient structural roles
   B. Grammars
      1. Ling: Universal wellformedness constraints, Con
      2. Ling++: SoftCon, Bayesian universal biases
      3. Limiting the power of language learning; innate linguistic knowledge?
         i. Argument from Universals* [not from Poverty of the Stimulus]
         ii. New experimental evidence supporting *
That's all folks!

Thanks to the organizers and to you for your attention.
The death of linguistic theory? A sketch from Cognitive Science

I. NN\^{++}: Implements Ling\^{++}
   A. Neural networks with structured distributed vectorial representations
   B. Computing (gradient) symbolic functions
   C. Encoding Ling\^{++} grammars

II. Ling\^{++}
   A. Representations
      1. Ling: Structural roles
      2. Ling\^{++}: Gradient structural roles
   B. Grammars
      1. Ling: Universal wellformedness constraints, Con
      2. Ling\^{++}: SoftCon, Bayesian universal biases
      3. Limiting the power of language learning; innate linguistic knowledge?
         i. Argument from Universals* [not from Poverty of the Stimulus]
         ii. New experimental evidence supporting *