Compositionality in Recursive Neural Networks

Martha Lewis

ILLC
University of Amsterdam

SYCO3, March 2019

Oxford, UK

Outline

- Compositional distributional semantics
- Pregroup grammars and how to map to vector spaces
- Recursive neural networks (TreeRNNs)
- Mapping pregroup grammars to TreeRNNs
- Implications

Compositional Distributional Semantics

Frege's principle of compositionality

The meaning of a complex expression is determined by the meanings of its parts and the rules used for combining them.

Compositional Distributional Semantics

Distributional hypothesis

Words that occur in similar contexts have similar meanings

[Harris, 1958].

Frege's principle of c… …y

The meaning of a con… …on is determined by the meanings of its parts an… …ed for combining them.

Symbolic Structure

- A pregroup algebra is a partially ordered monoid, where each element p has a left and a right adjoint such that:

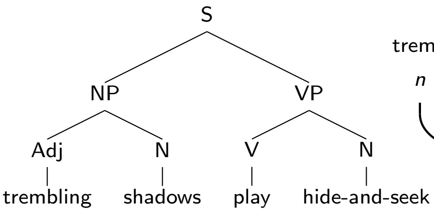
$$p \cdot p^r \leq 1 \leq p^r \cdot p \qquad p^l \cdot p \leq 1 \leq p \cdot p^l$$

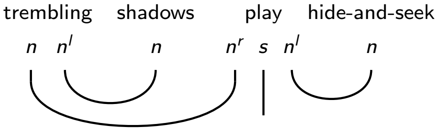
- Elements of the pregroup are basic (atomic) grammatical types, e.g. $\mathcal{B} = \{n, s\}$.
- Atomic grammatical types can be combined to form types of higher order (e.g. $n \cdot n^l$ or $n^r \cdot s \cdot n^l$)
- A sentence $w_1 w_2 \ldots w_n$ (with word w_i to be of type t_i) is grammatical whenever:

$$t_1 \cdot t_2 \cdot \ldots \cdot t_n \leq s$$

Pregroup derivation: example

$$p \cdot p^r \leq 1 \leq p^r \cdot p \qquad p^l \cdot p \leq 1 \leq p \cdot p^l$$

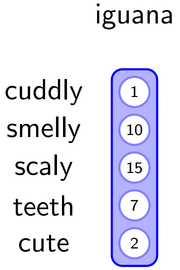


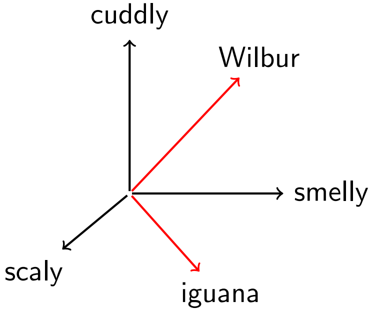


$$\begin{aligned} n \cdot n^l \cdot n \cdot n^r \cdot s \cdot n^l \cdot n &\leq n \cdot 1 \cdot n^r \cdot s \cdot 1 \\ &= n \cdot n^r \cdot s \\ &\leq 1 \cdot s \\ &= s \end{aligned}$$

Distributional Semantics

- Words are represented as vectors
- Entries of the vector represent how often the target word co-occurs with the context word





Similarity is given by cosine distance:

$$sim(v, w) = \cos(\theta_{v,w}) = \frac{\langle v, w \rangle}{||v||||w||}$$

The role of compositionality

Compositional distributional models

We can produce a sentence vector by composing the vectors of the words in that sentence.

$$\overrightarrow{s} = f(\overrightarrow{w_1}, \overrightarrow{w_2}, \ldots, \overrightarrow{w_n})$$

Three generic classes of CDMs:

- *Vector mixture* models [Mitchell and Lapata (2010)]
- *Tensor-based* models [Coecke, Sadrzadeh, Clark (2010); Baroni and Zamparelli (2010)]
- *Neural* models [Socher et al. (2012); Kalchbrenner et al. (2014)]

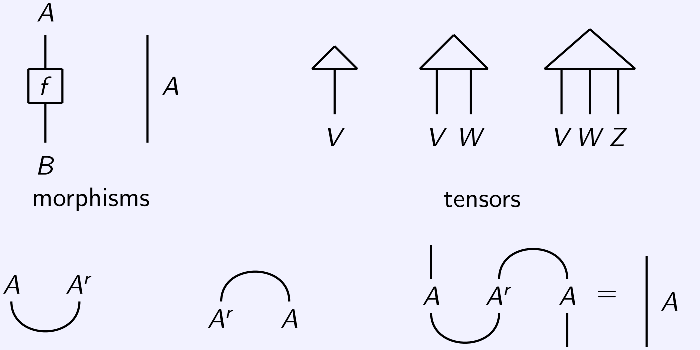
A multi-linear model

The grammatical type of a word defines the vector space in which the word lives:

- Nouns are vectors in N;
- adjectives are linear maps $N \rightarrow N$, i.e elements in $N \otimes N$;
- intransitive verbs are linear maps $N \rightarrow S$, i.e. elements in $N \otimes S$;
- transitive verbs are bi-linear maps $N \otimes N \rightarrow S$, i.e. elements of $N \otimes S \otimes N$;

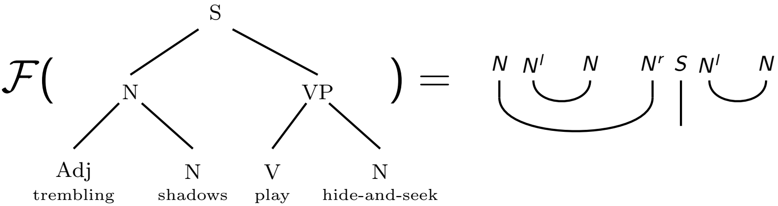
- The composition operation is tensor contraction, i.e. elimination of matching dimensions by application of inner product.

Coecke, Sadrzadeh, Clarke 2010

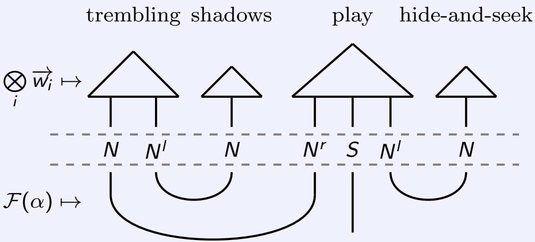


$$(\epsilon_A^r \otimes 1_A) \circ (1_A \otimes \eta_A^r) = 1_A$$

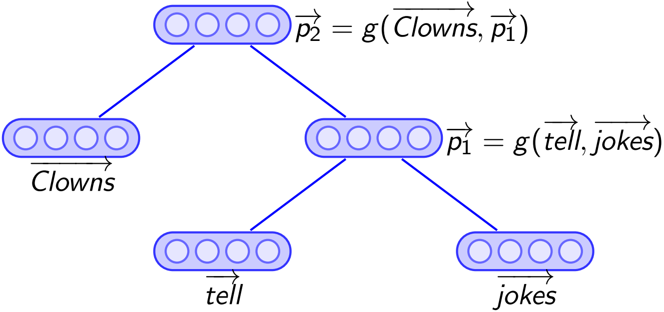
Diagrammatic calculus: example



$$\mathcal{F}(\alpha)(\overrightarrow{trembling} \otimes \overrightarrow{shadows} \otimes \overrightarrow{play} \otimes \overrightarrow{hide\text{-}and\text{-}seek})$$



Recursive Neural Networks



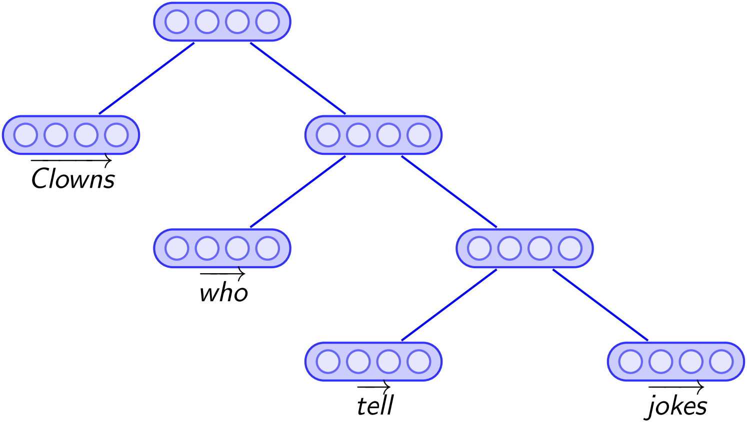
$$g_{RNN} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n :: (\overrightarrow{v_1}, \overrightarrow{v_2}) \mapsto f_1 \left(M \cdot \begin{bmatrix} \overrightarrow{v_1} \\ \overrightarrow{v_2} \end{bmatrix} \right)$$

$$g_{RNTN} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n :: (\overrightarrow{v_1}, \overrightarrow{v_2}) \mapsto g_{RNN}(\overrightarrow{v_1}, \overrightarrow{v_2}) + f_2 \left(\overrightarrow{v_1}^\top \cdot T \cdot \overrightarrow{v_2} \right)$$

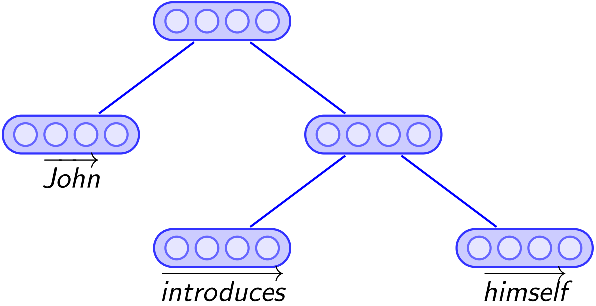
How compositional is this?

- Successful
- Some element of grammatical structure
- The compositionality function has to do everything
- Does that help us understand what's going on?

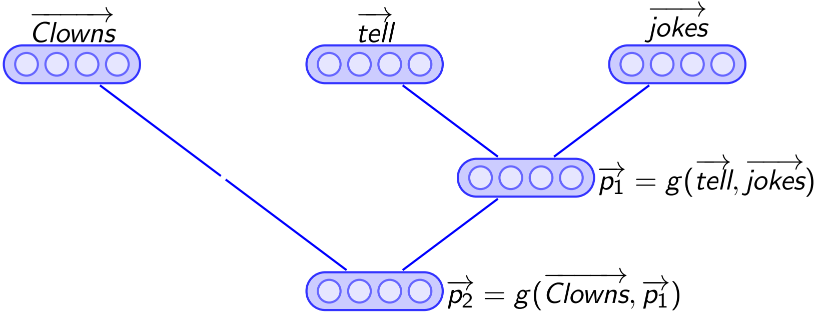
Information-routing words



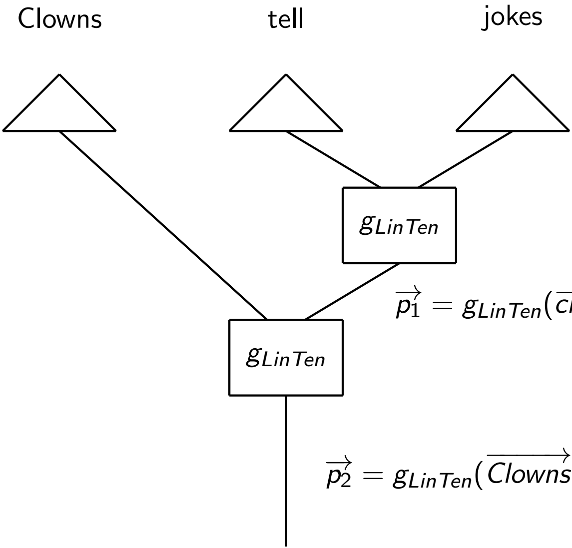
Information-routing words



Can we map pregroup grammar onto TreeRNNs?



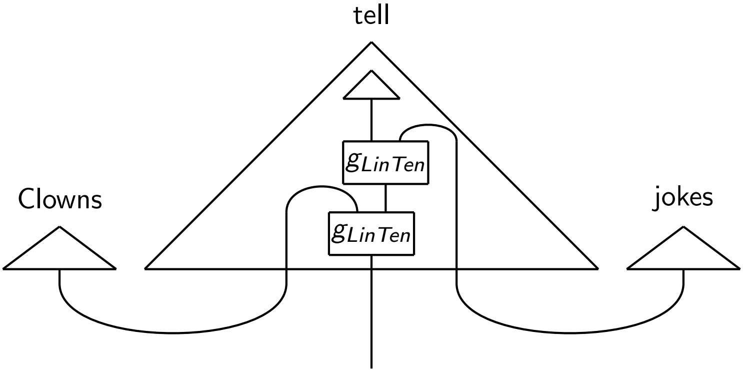
Can we map pregroup grammar onto TreeRNNs?



$$\overrightarrow{p_1} = g_{LinTen}(\overrightarrow{cross}, \overrightarrow{roads})$$

$$\overrightarrow{p_2} = g_{LinTen}(\overrightarrow{Clowns}, \overrightarrow{p_1})$$

Can we map pregroup grammar onto TreeRNNs?

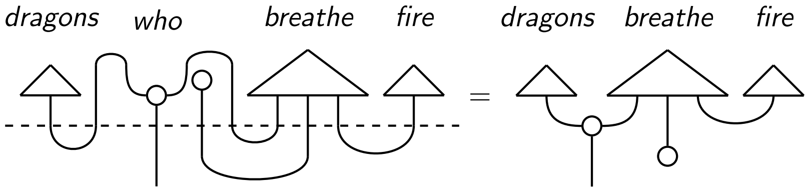


Why?

- Opens up more possibilities to use tools from formal semantics in computational linguistics.
- We can immediately see possibilities for building alternative networks - perhaps different compositionality functions for different parts of speech
- Decomposing the tensors for functional words into repeated applications of a compositionality function gives options for learning representations.

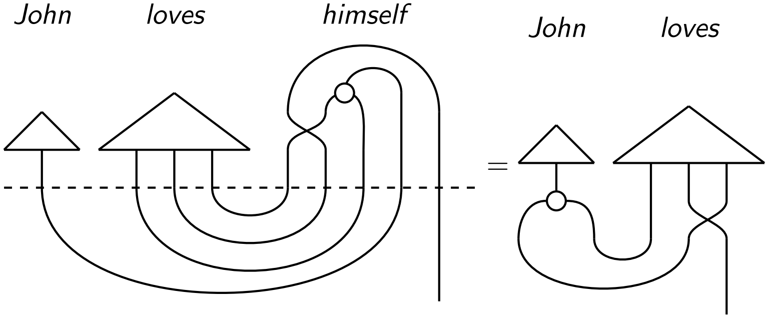
Why?

who : $n^r n s^l s$



Why?

himself : $n s^r n^{rr} n^r s$



Experiments?

Not yet. But there are a number of avenues for exploration

- Examining performance of this kind of model with standard categorical compositional distributional models
- Different compositionality functions for different word types
- Testing the performance of TreeRNNs with formally analyzed information-routing words.
- Investigating the effects of switching between word types.
- Investigating meanings of logical words and quantifiers.
- Extending the analysis to other types of recurrent neural network such as long short-term memory networks or gated recurrent units.

Summary

- We have shown how to interpret a simplification of recursive neural networks within a formal semantics framework
- We can then analyze 'information routing' words such as pronouns as specific functions rather than as vectors
- This also provides a simplification of tensor-based vector composition architectures, reducing the number of high order tensors to be learnt, and making representations more flexible and reusable.
- Plenty of work to do on both the experimental and the theoretical side!

Thanks!

NWO Veni grant ‘Metaphorical Meanings for Artificial Agents’

Category-Theoretic Background

- The category of pregroups **Preg** and the category of finite dimensional vector spaces **FdVect** are both *compact closed*
- This means that they share a structure, namely:
  - Both have a tensor product $\otimes$ with a unit 1
  - Both have adjoints A^r, A^l
  - Both have special morphisms

$$\epsilon^r : A \otimes A^r \to 1, \quad \epsilon^l : A^l \otimes A \to 1$$

$$\eta^r : 1 \to A^r \otimes A, \quad \eta^l : 1 \to A \otimes A^l$$

  - These morphisms interact in a certain way.
- In **Preg**:

$$p \cdot p^r \leq 1 \leq p^r \cdot p \qquad p^l \cdot p \leq 1 \leq p \cdot p^l$$

A functor from syntax to semantics

We define a functor $\mathcal{F}$: **Preg** $\rightarrow$ **FdVect** such that:

$$
\begin{aligned}
\mathcal{F}(p) &= P \quad \forall p \in \mathcal{B} \\
\mathcal{F}(1) &= \mathbb{R} \\
\mathcal{F}(p \cdot q) &= \mathcal{F}(p) \otimes \mathcal{F}(q) \\
\mathcal{F}(p^r) = \mathcal{F}(p^l) &= \mathcal{F}(p) \\
\mathcal{F}(p \leq q) &= \mathcal{F}(p) \rightarrow \mathcal{F}(q) \\
\mathcal{F}(\epsilon^r) = \mathcal{F}(\epsilon^l) &= \text{inner product in } \mathbf{FdVect} \\
\mathcal{F}(\eta^r) = \mathcal{F}(\eta^l) &= \text{identity maps in } \mathbf{FdVect}
\end{aligned}
$$

[Kartsaklis, Sadrzadeh, Pulman and Coecke, 2016]

References I