Does language have a logical structure?

Thomas Seiller
CNRS, LIPN (Paris 13 Univ.)

Rencontre NORMES, Paris
https://www.seiller.org/Normes24.pdf

Logic and Language

Disclaimer

I will not discuss the history of work trying to relate logical systems and language: we will come at this from a quite different perspective.

- There is a complex structure to natural languages.
- The recent introduction of Large Language Models (LLM) witnesses that (at least part of) the structure of natural language can be extracted from the distributional content of a corpus.
- There has been some recent work trying to understand this structure (in particular by Danae-Bradley, Gastaldi, and Terilla).
- This talk is a kind of "mid-term report" on a collaboration with S. Jarvis, J.-L. Gastaldi, L. Pellissier, J. Terilla where we uncover a relation between this structure and some work from linear logic.
- We use this relation to get a better understanding of the structure, but it is only the first step in a much larger project: there is still a lot of work ahead!

Background

Word embeddings

Deep neural network: a function $f : \mathbf{R}^{n_0} \to \mathbf{R}^{n_k}$ explicitly expressed as a composition

$$\mathbf{R}^{n_0} \xrightarrow{f_1} \mathbf{R}^{n_1} \xrightarrow{f_2} \mathbf{R}^{n_2} \xrightarrow{f_3} \dots \xrightarrow{f_{k-1}} \mathbf{R}^{n_{k-1}} \xrightarrow{f_k} \mathbf{R}^{n_k}$$

where

$$f_i(x) = a(M_i x + b_i)$$

where M_i are $n_i \times n_i$ matrices, $b_i \in \mathbf{R}^{n_i}$ are *biases*, a is a (nonlinear) *activation* function, and g is an output function.

- Quick realisation: significant increase of performance when the first layer (i.e. f_1) of a model trained for a linguistic task is used as first layer in another model aimed at a different linguistic task.
- People started training these *word embeddings* separately.

Semantic information in word embeddings

- Word embedding capture semantic information. E.g. the vector for `Berlin` minus the vector for `Germany` is numerically very near the vector for `Paris` minus the vector for `France`.

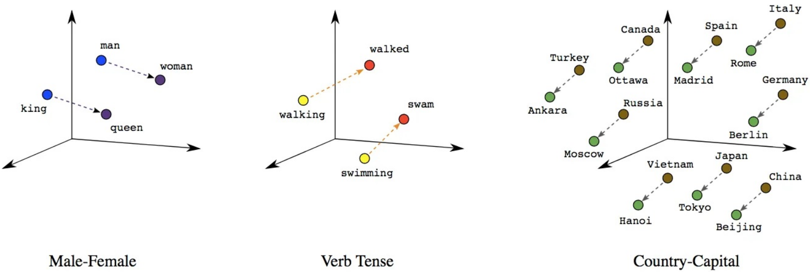


Figure: Image from developers.google.com

- The algorithm has been carefully studied and shown to perform an implicit factorization of a matrix comprised of information about how words are used in language.

In summary, the math story of word embeddings goes like this:

1. Start from a set D of vocabulary words.
2. Move to the free vector space $\mathbf{R}^{n_0}$ with $n_0 = \mathrm{Card}(D)$.
3. Consider a $n_0 \times n_0$ matrix M consisting of rough statistical data about how words go with other words in a corpus of text.
4. The columns of M (or more precisely the columns of a low-rank factorization of M), then interact with the vector space structure to reveal otherwise hidden syntactic and semantic information in the set D of words.

- there is an exact solution for the low-rank factorization of a matrix M using the truncated singular value decomposition (SVD). This roughly corresponds to writing down the matrix that maps eigenvectors of MM^* to corresponding (i.e. same eigenvalue) eigenvectors of M^*M.
- Note that M assumes nothing about the structure that D might possess: it is purely a witness of how D has been used in a particular corpus.

Category theory perspective

This can be reformulated in terms of categories;

- The matrix is replaced by a profunctor $M : \mathbf{C}^{\mathrm{op}} \times \mathbf{D} \to \mathbf{Set}$.
- By standard manipulations (currying and lifting to presheaves), f gives rise to two functors defining an adjunction:

$$\mathbf{Set}^{\mathbf{C}^{\mathrm{op}}} \underset{M_*}{\overset{M^*}{\rightleftarrows}} (\mathbf{Set}^{\mathbf{D}})^{\mathrm{op}} \qquad (M^* \dashv M_*)$$

- The equivalent of eigenvectors are called *nuclei* of the adjunction, they are pairs of objects (A,B) such that $M^*(A) = B$ and $M_*(B) = A$.

Back to language

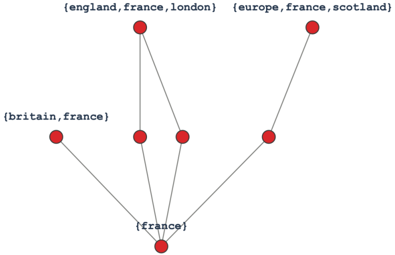
Now consider that instead of a **Set**-valued profunctor, we start with

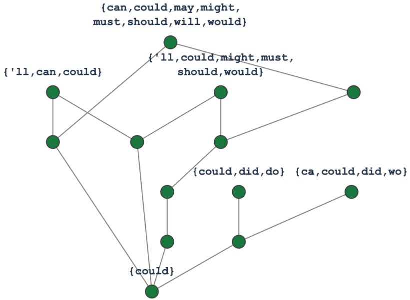
$$M : \mathbf{X}^{\mathrm{op}} \times \mathbf{X} \to \mathbf{2},$$

where $\mathbf{2}$ is the category $0 \leftarrow 1$, and $\mathbf{X}, \mathbf{Y}$ are finite set (seen as discrete categories).

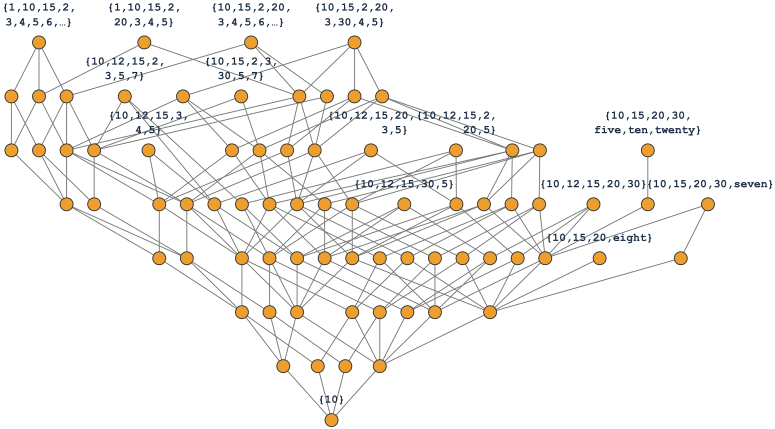
- The profunctor M is simply a binary relation;
- The maps M_* and M^* are defined on *subsets* of $\mathbf{C}$ and $\mathbf{D}$. More precisely, if $A \subset \mathbf{X}$ and $B \subset \mathbf{Y}$:
$$M^*(A) = \{b \in \mathbf{Y} \mid \forall a \in A, M(a,b) = 1\}$$
$$M_*(B) = \{a \in \mathbf{X} \mid \forall b \in B, M(a,b) = 1\}$$
- The *nuclei* of the adjunction, i.e. pairs of subsets (A,B) such that $M^*(A) = B$ and $M_*(B) = A$ are *formal concepts*.
- From the corpus of Wikipedia articles, Danae-Bradley, Gastaldi and Terilla computed formal concepts. The following illustrations are taken from their article in the journal in the AMS.

Back to language: the lattice of formal concepts





Back to language: the lattice of formal concepts



Where is the structure?

- We now want to understand the structure of nuclei (formal concepts)
- But only little is known: the set of nuclei has a natural lattice structure (shown in the previous slides).

This is the topic of the (second half of the) current talk: I will exhibit some of the structure of the nuclei.

Logic and Structure

First approach: real-valued presheaves

The measure used in these is based on what is called *pointwise mutual information*, or *pmi*. It is defined as follows:

$$\mathrm{pmi}(w_1, w_2) = \frac{p(w_1 \cdot w_2)}{p(w_1)p(w_2)},$$

where $p(\cdot)$ denotes the probability that two words appear next to each other. More precisely, one considers;

$$[\![w_1, w_2]\!]_m = -\log(\mathrm{pmi}(w_1, w_2)).$$

This measure is used in word embeddings, and it can be computed easily from the corpus ρ: $p(w) =$ number of occurrences of w in ρ / size of ρ.

- This naturally leads to consider the real numbers $\mathbf{R}$ (with the usual order) instead of the category $\mathbf{2}$ in the previous construction. Start with $M : \mathbf{X} \times \mathbf{Y} \to \mathbf{R}$, and try to understand the nuclei of the induced adjunction.
- A nuclei (A, B) can be understood as a pair of vectors: $A \in \mathbf{R}^{\mathbf{X}}$, and $B \in \mathbf{R}^{\mathbf{Y}}$.

PMI and the trefoil property

One key realisation was that pmi satisfies the *trefoil property* (also called the *2-cocycle property*) with respect to the concatenation of words:

$$\mathrm{pmi}(w_1 \cdot w_2, w_3).\mathrm{pmi}(w_1, w_2) = \mathrm{pmi}(w_1, w_2 \cdot w_3)\mathrm{pmi}(w_2, w_3)$$

This translates as the following property of the measure:

$$[\![w_1 \cdot w_2, w_3]\!]_m + [\![w_1, w_2]\!]_m = [\![w_1, w_2 \cdot w_3]\!]_m + [\![w_2, w_3]\!]_m.$$

This property has appeared in models of linear logic (interaction graphs) based on formal concepts!

Second approach: linear realisability

The trefoil property suggests another approach based on linear logic: we have a set D (of words, or simply characters) together with an associative operation (concatenation), and a measurement $D \times D \to \mathbf{R}$ satisfying the trefoil property. We can therefore define a model of multiplicative linear logic (details: HdR).

- Consider the set of pairs (a, α) of a word a and a real number α.
- Extend the measurement on pairs: $[\![(a,\alpha),(b,\beta)]\!]_m = \alpha + \beta - [\![a,b]\!]_m$.
- Define a binary relation: $(a,\alpha) \perp (b,\beta) \Leftrightarrow \alpha + \beta - [\![a,b]\!]_m \leq 0$.
- Define a *type* as a set A such that there exists B with

$$A = {}^{\perp}B = \{(a,\alpha) \mid \forall (b,\beta) \in B, (a,\alpha) \perp (b,\beta)\}.$$

- Define the following construction on types:

$$A \to B = \{(c,\gamma) \mid \forall (a,\alpha) \in A, (c,\gamma) \cdot (a,\alpha) \in B$$

- This defines a model of Multiplicative Linear Logic!

Relating the constructions

One can notice that with this definition, the types are downward closed:

- if $(a, \alpha) \in A$, and $\lambda > 0$, then $(a, \alpha - \lambda) \in A$.

They can therefore be understood as vectors: $A : D \to \mathbf{R} \cup \{-\infty\}$, where

$$A(a) = \sup\{\alpha \in \mathbf{R} \mid (a, \alpha) \in A\}.$$

In fact:

Theorem

Types in the model of MLL just described are exactly the nuclei of the adjunction obtained between $\mathbf{R}$-enriched presheaves.

A mutual enrichment: composition

This provides an interesting new structure on the set of nuclei, and suggests that some extension of linear logic can describe the structure of language emerging from a corpus.
We give one example of how linear logic describes somehow complex structural properties of the set of nuclei.

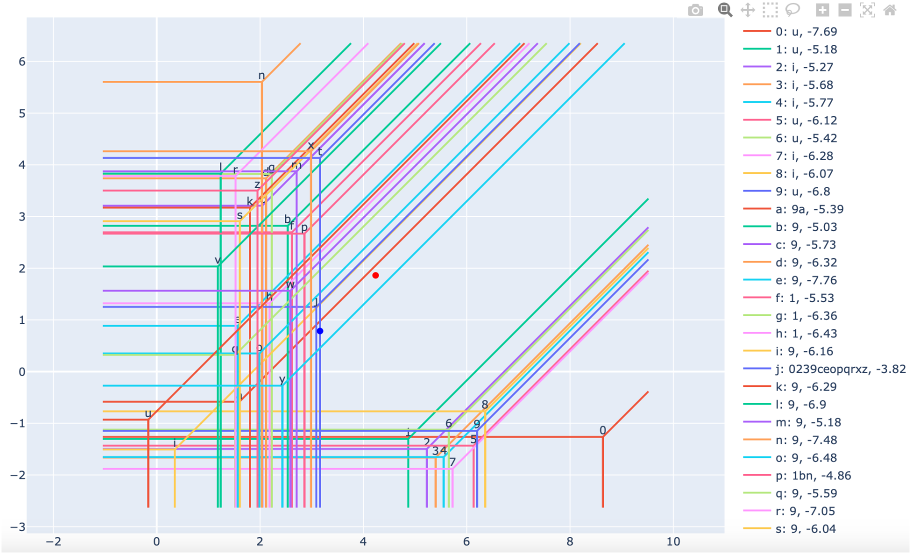
- Remember we started from a measure $D \times D \rightarrow \mathbf{R}$. But in general, D can be considered as Σ^* where Σ is a base alphabet (of characters). In particular, it "contains" measures:
$$\Sigma \times \Sigma \times \cdots \times \Sigma \rightarrow \mathbf{R}.$$
- Consider the simple case of a trivalent measure $\Sigma \times \Sigma \times \Sigma \rightarrow \mathbf{R}$. Then it can be considered as a binary relation between Σ and $\Sigma \times \Sigma$. As such, one can consider a nuclei $(A, A^{\perp})$, where $A^{\perp}$ is a presheaf over $\Sigma \times \Sigma$.
- But $A^{\perp}$ can then be considered as a measure on $\Sigma \times \Sigma$ from which one can define nuclei. There is thus a complex structure of *derived nuclei* to be understood.
- It can be shown to relate to the linear logic structure: if (B, C) is such a nuclei, then $C = A \multimap B$.

A mutual enrichment: tropical geometry

The presheaves approach brings in a different, more geometric, perspective.

- the map M^*M_* is an actual projection;
- the set of nuclei is a tropical space which we can try to understand.

Types over characters as a tropical space



Cells, DNA, and wall crossing formulas

The set of all types is composed of cells of different dimensions (here ≤ 36) embedded in a 36 dimensional space. We can define, given a vector $A \in \mathbf{R}^{\Sigma}$, it *dna*:

$$\mathrm{dna}(A) : \begin{cases} \Sigma & \rightarrow & 2^{\Sigma} \\ a & \mapsto & \{b_0 \in \Sigma \mid \min_b \{M(a,b) - A(a)\} = M(a,b_0) - A(a)\} \end{cases}$$

Then we have the following theorems.

Theorem

A vector A is a type if and only if $\bigcup_{a \in \Sigma} \mathrm{dna}(A) = \Sigma$.

If we write $\sharp A = \sum_{a \in \Sigma} \mathrm{Card}(\mathrm{dna}(A)(a))$,

Theorem

A cell of dimension $36 - d$ *consists of types such that* $\sharp A = 36 + d$.

Finally, moving from one cell to an adjacent cell consists in adding or removing one element in $\mathrm{dna}(A)(a)$ for some $a \in \Sigma$.

Tropical low rank approximations

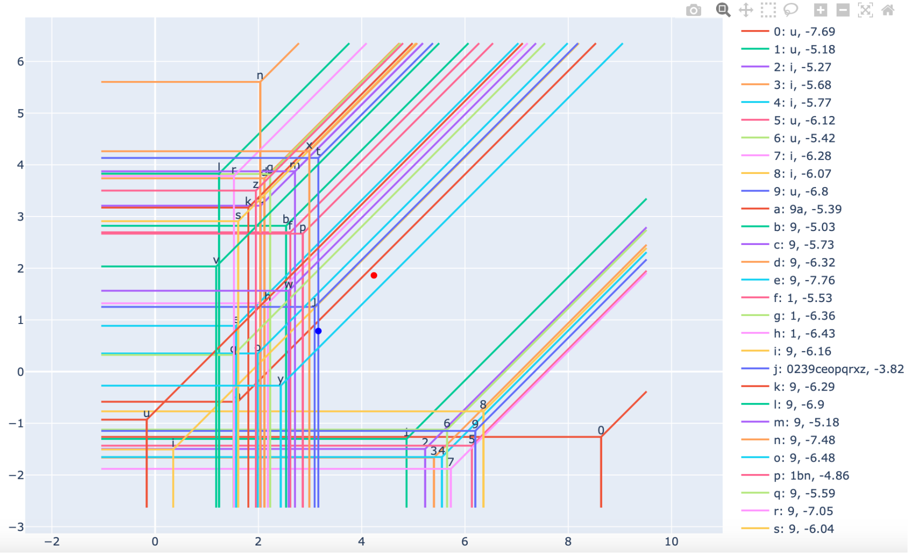
- We would like to consider a tropical version of low-rank approximation;
- Does not really exist in the literature (but lively research topic);
- In our special case, it may be easier to define though.

- In order to properly represent the 36 dimensional space on a 2D figure, we had to consider some reduction;
- we reduce the 36×36 matrix to a 3×36 matrix by iterating the following reduction from a (n,k) matrix to a $(n,k-1)$ matrix:
  - compute the ℓ^1 distance between all "tropical stars"
  - reduce the matrix by removing the two columns corresponding to the closest crosses, and adding the pointwise min of those.
- in this way, we get an approximation whose basis vectors (in the 3-dimensional side) corresponds to sets of elements in Σ.

Dimension reduction: vectors

```
['05', '19', '234678', 'a', 'b', 'c', 'dys', 'e', 'fg', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 't', 'u', 'v', 'w', 'x', 'z']
['05', '19', '234678', 'a', 'b', 'c', 'dysfg', 'e', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 't', 'u', 'v', 'w', 'x', 'z']
['05', '19', '234678', 'a', 'b', 'c', 'dysfgk', 'e', 'h', 'i', 'j', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 't', 'u', 'v', 'w', 'x', 'z']
['05', '19', '234678', 'a', 'b', 'c', 'dysfgkt', 'e', 'h', 'i', 'j', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 'u', 'v', 'w', 'x', 'z']
['05', '19234678', 'a', 'b', 'c', 'dysfgkt', 'e', 'h', 'i', 'j', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 'u', 'v', 'w', 'x', 'z']
['05', '19234678', 'a', 'b', 'c', 'dysfgkt', 'e', 'h', 'i', 'j', 'l', 'm', 'nr', 'o', 'p', 'q', 'u', 'v', 'w', 'x', 'z']
['05', '19234678', 'a', 'b', 'c', 'dysfgktnr', 'e', 'h', 'i', 'j', 'l', 'm', 'o', 'p', 'q', 'u', 'v', 'w', 'x', 'z']
['05', '19234678', 'a', 'b', 'c', 'dysfgktnrm', 'e', 'h', 'i', 'j', 'l', 'o', 'p', 'q', 'u', 'v', 'w', 'x', 'z']
['05', '19234678', 'a', 'b', 'c', 'dysfgktnrmh', 'e', 'i', 'j', 'l', 'o', 'p', 'q', 'u', 'v', 'w', 'x', 'z']
['05', '19234678', 'a', 'b', 'c', 'dysfgktnrmhw', 'e', 'i', 'j', 'l', 'o', 'p', 'q', 'u', 'v', 'x', 'z']
['05', '19234678', 'a', 'b', 'c', 'dysfgktnrmhw', 'eo', 'i', 'j', 'l', 'p', 'q', 'u', 'v', 'x', 'z']
['05', '19234678', 'aeo', 'b', 'c', 'dysfgktnrmhw', 'i', 'j', 'l', 'p', 'q', 'u', 'v', 'x', 'z']
['05', '19234678', 'aeo', 'b', 'c', 'dysfgktnrmhw', 'iu', 'j', 'l', 'p', 'q', 'v', 'x', 'z']
['05', '19234678', 'aeo', 'b', 'c', 'dysfgktnrmhw', 'iu', 'j', 'l', 'p', 'q', 'v', 'xz']
['05', '19234678', 'aeo', 'b', 'c', 'dysfgktnrmhw', 'iu', 'j', 'l', 'pxz', 'q', 'v']
['05', '19234678', 'aeo', 'b', 'c', 'dysfgktnrmhwpxz', 'iu', 'j', 'l', 'q', 'v']
['05', '19234678', 'aeo', 'b', 'c', 'dysfgktnrmhwpxz', 'iu', 'jv', 'l', 'q']
['0519234678', 'aeo', 'b', 'c', 'dysfgktnrmhwpxz', 'iu', 'jv', 'l', 'q']
['0519234678', 'aeo', 'b', 'cdysfgktnrmhwpxz', 'iu', 'jv', 'l', 'q']
['0519234678', 'aeo', 'b', 'cdysfgktnrmhwpxzjv', 'iu', 'l', 'q']
['0519234678', 'aeo', 'bl', 'cdysfgktnrmhwpxzjv', 'iu', 'q']
['0519234678', 'aeo', 'blcdysfgktnrmhwpxzjv', 'iu', 'q']
['0519234678', 'aeoiu', 'blcdysfgktnrmhwpxzjv', 'q']
['0519234678', 'aeoiu', 'blcdysfgktnrmhwpxzjvq']
```

Types over characters as a tropical space



Does language have a logical structure?

Thomas Seiller
CNRS, LIPN (Paris 13 Univ.)

Rencontre NORMES, Paris
https://www.seiller.org/Normes24.pdf