The Geometry of Syntax (Part 2)

Matilde Marcolli
U. Toronto, Perimeter Institute, Caltech

Utrecht, April 2018
Colloquium Mathematics and Linguistics
A Mathematical Physicist's adventures in Linguistics

- This talk is in two parts:
  - **Today Linguistics**: relations between syntactic parameters via Kanerva networks, coding theory, and via Belkin–Niyogi heat kernel dimensional reduction
  - **Continuation of Math Colloquium**: persistent topology of syntax, algebro-geometric historical linguistics, spin glass models of language evolution

The two parts are largely independent: no need to have seen the previous one (some small amount of repetition if you have seen it)
References for this part:


Questions and Motivation

• Broad question: can we use computational (mathematical) techniques to better understand how the human brain processes language?

• some of the main questions:
  
  ● Language acquisition (poverty of the stimulus): how does the learning brain converge to one grammar?
  
  ● How is language (in particular syntax) stored in the brain?
  
  ● How do languages change and evolve in time? quantitative, predictive modeling?

• focus on the “large scale structure” of language: syntax

• Plan: approach these questions from a mathematical perspective, using tools from geometry and theoretical physics
Syntax and Syntactic Parameters

- we work here within the framework of the Principles and Parameters model (Chomsky, 1981)
  - principles: general rules of grammar
  - parameters: binary variables (on/off switches) that distinguish languages in terms of syntactic structures

- this idea is very appealing for a mathematician: at the level of syntax a language can be described by a set of coordinates given by binary variables

- however, surprisingly no mathematical model of Principles and Parameters formulation of Linguistics has been developed so far
• Example of parameter: head-directionality (head-initial versus head-final) English is head-initial, Japanese is head-final

VP= verb phrase, TP= tense phrase, DP= determiner phrase

• Other examples of parameters:
  - *Subject-side*
  - *Pro-drop*
  - *Null-subject*
Main Problems

- there is no complete classification of syntactic parameters
- there are hundreds of such binary syntactic variables, but not all of them are “true” syntactic parameters (conflations of deep/surface structure)
- Interdependencies between different syntactic parameters are poorly understood: what is a good independent set of variables, a good set of coordinates?
- syntactic parameters are dynamical: they change historically over the course of language change and evolution
- collecting reliable data is hard! (there are thousands of world languages and analyzing them at the level of syntax is much more difficult for linguists than collecting lexical data; few ancient languages have enough written texts)
Databases of syntactic structures of world languages

1. Syntactic Structures of World Languages (SSWL)  
   http://sswl.railsplayground.net/

2. TerraLing  
   http://www.terraling.com/

   http://wals.info/


- First Step: data analysis of syntax of world languages with various mathematical tools (persistent topology, etc.)
- we used the most extensive database currently available: SSWL with 116 “variables” (syntactic “parameters”) and 253 world languages (but... some problems with SSWL)
Problems of SSWL data

- Very non-uniformly mapped across the languages of the database: some are 100% mapped, while for some only very few of the 116 parameters are mapped

- Linguists criticize the choice of binary variable (not all of them should count as “true” parameters)

- the data of Longobardi–Guardiano are more reliable, with 62 languages (mostly Indo-European) and 83 parameters

- linguistic question: can languages that are far away in terms of historical linguistics end up being close in terms of syntactic parameters?

- Guideline for data use: given what is available at present, use SSWL and Longobardi data (two independent set of syntactic features) keeping limitations in mind and comparing structures of the two datasets
Parameters from Modularized Global Parameterization Method


- Determiner Phrase Module:
  - syntactic parameters dealing with person, number, gender (1–6)
  - parameters of definiteness (7–16)
  - parameters of countability (17–24)
  - genitive structure (25–31)
  - adjectival and relative modification (32–14)
  - position and movement of the head noun (42–50)
  - demonstratives and other determiners (51–50 and 6–63)
  - possessive pronouns (56–59)

- more parameters added in the more recent publication of a more extensive list of data from Longobardi and collaborators
What kind of relations exist between syntactic parameters?

- **Entailment relations**: some explicitly known relations where one state of a parameter (or more) can make another parameter undefined.

- **Example**: \( \{p_1, p_2\} = \{\text{Strong Deixis, Strong Anaphoricity}\} \)

\[
\begin{array}{c|cc}
\ell & p_1 & p_2 \\
\hline
\ell_1 & +1 & +1 \\
\ell_2 & -1 & 0 \\
\ell_3 & +1 & +1 \\
\ell_4 & +1 & -1 \\
\end{array}
\]

\( \{\ell_1, \ell_2, \ell_3, \ell_4\} = \{\text{English, Welsh, Russian, Bulgarian}\} \)

**Strong Deixis +1**: governs possible positions of demonstratives in the nominal domain.

**Strong Anaphoricity +1**: obligatory dependence on an antecedent in a local and asymmetric relation to anaphor.
• several entailment relations are recorded in the data of Longobardi–Guardiano
• SSWL database does not record relations between parameters
• relations can be detected through methods of data analysis
• goals: identify a good set of independent variables among syntactic parameters, understand (at least statistically) the “manifold” determined by the relations
• some methods we consider here:
  1. Kanerva networks: sparse distributed memories
  2. coding theory: code parameters, position in the space of codes
  3. heat kernel dimensional reduction: Laplace eigenfunctions
Syntactic Parameters in Kanerva Networks


- Select a subset of SSWL parameters with properties:
  - Completely mapped for a large number of languages in the database
  - Known to have relations, though not of a simple explicit entailment form

- Detect which among these parameters are more or less recoverable from the other ones by testing recoverability in a sparse distributed memory
Preliminary considerations: Frequency of Expression

- Different syntactic parameters have very different frequency of expression among world languages.

- Example: Word Order: SOV, SVO, VSO, VOS, OVS, OSV

<table>
<thead>
<tr>
<th>Word Orders</th>
<th>Percentage</th>
<th>Syntactic Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOV</td>
<td>41.03%</td>
<td>Subject-initial, Specifier-Head</td>
</tr>
<tr>
<td>SVO</td>
<td>35.44%</td>
<td>Specifier-Head</td>
</tr>
<tr>
<td>VSO</td>
<td>6.90%</td>
<td>Subject-medial</td>
</tr>
<tr>
<td>VOS</td>
<td>1.82%</td>
<td>Subject-final</td>
</tr>
<tr>
<td>OVS</td>
<td>0.79%</td>
<td></td>
</tr>
<tr>
<td>OSV</td>
<td>0.29%</td>
<td>Subject-medial, Specifier-Head</td>
</tr>
</tbody>
</table>

Very unevenly distributed across world languages.

- This creates overall effect (using data that record expression of parameters among world languages): needs to be normalized when searching for abstract syntactic relations among parameters.
### Parameters and frequencies (as classified in SSWL)

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Subject-Verb</td>
<td>0.64957267</td>
</tr>
<tr>
<td>02</td>
<td>Verb-Subject</td>
<td>0.31623933</td>
</tr>
<tr>
<td>03</td>
<td>Verb-Object</td>
<td>0.61538464</td>
</tr>
<tr>
<td>04</td>
<td>Object-Verb</td>
<td>0.32478634</td>
</tr>
<tr>
<td>05</td>
<td>Subject-Verb-Object</td>
<td>0.56837606</td>
</tr>
<tr>
<td>06</td>
<td>Subject-Object-Verb</td>
<td>0.30769232</td>
</tr>
<tr>
<td>07</td>
<td>Verb-Subject-Object</td>
<td>0.19230770</td>
</tr>
<tr>
<td>08</td>
<td>Verb-Object-Subject</td>
<td>0.15811966</td>
</tr>
<tr>
<td>09</td>
<td>Object-Subject-Verb</td>
<td>0.12393162</td>
</tr>
<tr>
<td>10</td>
<td>Object-Verb-Subject</td>
<td>0.10683761</td>
</tr>
<tr>
<td>11</td>
<td>Adposition-Noun-Phrase</td>
<td>0.58974361</td>
</tr>
<tr>
<td>12</td>
<td>Noun-Phrase-Adposition</td>
<td>0.29059830</td>
</tr>
<tr>
<td>13</td>
<td>Adjective-Noun</td>
<td>0.41025642</td>
</tr>
<tr>
<td>14</td>
<td>Noun-Adjective</td>
<td>0.52564102</td>
</tr>
<tr>
<td>15</td>
<td>Numeral-Noun</td>
<td>0.48290598</td>
</tr>
<tr>
<td>16</td>
<td>Noun-Numeral</td>
<td>0.38034189</td>
</tr>
<tr>
<td>17</td>
<td>Demonstrative-Noun</td>
<td>0.47435898</td>
</tr>
<tr>
<td>18</td>
<td>Noun-Demonstrative</td>
<td>0.38461539</td>
</tr>
<tr>
<td>19</td>
<td>Possessor-Noun</td>
<td>0.38034189</td>
</tr>
<tr>
<td>20</td>
<td>Noun-Possessor</td>
<td>0.49145299</td>
</tr>
<tr>
<td>A01</td>
<td>Attributive-Adjective-Agreement</td>
<td>0.46581197</td>
</tr>
</tbody>
</table>
Kanerva networks (sparse distributed memories)
• field $\mathbb{F}_2 = \{0, 1\}$, vector space $\mathbb{F}_2^N$ large $N$
• uniform random sample of $2^k$ hard locations with $2^k << 2^N$
• median Hamming distance between hard locations
• Hamming spheres of radius slightly larger than median value (access sphere)

• *writing to network*: storing datum $X \in \mathbb{F}_2^N$, each hard location in access sphere of $X$ gets $i$-th coordinate (initialized at zero) incremented depending on $i$-th entry of $X$

• *reading at a location*: $i$-th entry determined by majority rule of $i$-th entries of all stored data in hard locations within access sphere

Kanerva networks are good at reconstructing corrupted data
Memory items in SDM: most items unrelated but most pairs linked by few intermediaries

illustration from: Ján Kvak, Creating and Recognizing Visual Words Using Sparse Distributed Memory

proposed as a realistic computational model of how information is stored and retrieved in human memory
Procedure

- Kanerva Network with Boolean space $\mathbb{F}_2^{21}$
- 166 data points (fully mapped SSWL languages)
- Kanerva network with access sphere of $n/4$, with $n$ median Hamming distance between points
- optimal: larger $n$ excessive number of hard locations being in the sphere, computationally intractable
- correct data written to the Kanerva network
- known language bit-string, with a single corrupted bit, used as read location
- result of the read compared to original bit-string to test bit recovery
- average Hamming distance resulting from corruption of a given bit (a particular syntactic parameter) computed across all languages
need to identify effects due to syntax from overall frequency effect
Normalize for frequency effect

- the recoverability data obtained combine two effects
  - an overall effect depending on the frequency of expression
  - a finer effect due to actual syntactic relations

- **Procedure** to separate overall frequency effect:
  - for each syntactic parameter subset of languages of fixed size
    chosen randomly with property that half of the languages
    have that parameter expressed
  - ignore those parameters with too few languages for which this
    can be done
  - use a fixed size of 95 languages
  - data of these languages written to Kanerva network and
    recoverability of corrupted individual parameters tested again
  - test run again with random data generated with an
    approximately similar distribution of bits
Overall effect related to relative prevalence of a parameter
More refined effect after normalizing for prevalence (extracting effect of syntactic dependencies)
Additional Remarks

• Overall effect relating recoverability in a Kanerva Network to prevalence of a certain parameter among languages (depends only on frequencies: see in random data with assigned frequencies)

• Additional effects (that deviate from random case) which detect possible dependencies among syntactic parameters: increased recoverability beyond what effect based on frequency

• Possible neuroscience implications? Kanerva Networks as models of human memory (parameter prevalence linked to neuroscience models)

• More refined effects if divided by language families?
Coding Theory to study how syntactic structures differ across the landscape of human languages

- Matilde Marcolli, *Syntactic Parameters and a Coding Theory Perspective on Entropy and Complexity of Language Families*, Entropy 2016, 18(4), 110

- select a group of languages \( L = \{ \ell_1, \ldots, \ell_N \} \)
- with the binary strings of \( n \) syntactic parameters form a code \( C(L) \subset \mathbb{F}_2^n \)
- compute code parameters \( (R(C), \delta(C)) \) code rate and relative minimum distance
- analyze position of \( (R, \delta) \) in space of code parameters
- get information about “syntactic complexity” of \( L \)

- Note: some overlap with my talk “Codes and Complexity” for the Centre for Complex Systems Studies
Codes and code parameters

error correcting codes $\mathcal{C} \subset \mathbb{F}_2^n$

- transmission rate (encoding)

$$R(\mathcal{C}) = \frac{k}{n}, \quad k = \log_2(\#\mathcal{C}) = \log_2(N)$$

for $q$-ary codes in $\mathbb{F}_q^n$ take $k = \log_q(N)$

- relative minimum distance (decoding)

$$\delta(\mathcal{C}) = \frac{d}{n}, \quad d = \min_{\ell_1 \neq \ell_2} d_H(\ell_1, \ell_2)$$

Hamming distance of binary strings of $\ell_1$ and $\ell_2$

- error correcting codes: optimize for maximal $R$ and $\delta$ but constraints that make them inversely correlated

- bounds in the space of code parameters $(R, \delta)$
Bounds on code parameters

• singleton bound: \( R + \delta \leq 1 \)

• Gilbert-Varshamov curve (q-ary codes)

\[
R = 1 - H_q(\delta), \quad H_q(\delta) = \delta \log_q(q - 1) - \delta \log_q \delta - (1 - \delta) \log_q(1 - \delta)
\]

q-ary Shannon entropy: asymptotic behavior of volumes of Hamming balls for large \( n \)

• The Gilbert-Varshamov curve represents the typical behavior of large random codes (Shannon Random Code Ensemble)

• Note: if syntactic parameters really were identically distributed independent random variables, subject to an evolution via a Markov model on a tree (simple assumption of phylogenetic models) then would expect codes from sets of languages to behave like Shannon random codes

• distance from SRCE behavior measures presence of relations that affect distribution of syntactic parameters across languages
The asymptotic bound

- Existence proved by spoiling operations on codes

\[ R = \alpha_q(\delta) \] continuous decreasing function with $\alpha_q(0) = 1$ and $\alpha_q(\delta) = 0$ for $\delta \in \left[ \frac{q-1}{q}, 1 \right]$
Properties of the asymptotic bound

- separates space $[0, 1]^2$ of code parameters into region below asymptotic bound $R = \alpha_q(\delta)$ where code points dense and with infinite multiplicity from region above where code points isolated and with finite multiplicity

- the function $R = \alpha_q(\delta)$ may be non-computable, but only as bad as Kolmogorov complexity (becomes computable given an oracle that orders codes by their Kolmogorov complexity)


Estimates on the asymptotic bound

- Plotkin bound:
  \[ \alpha_q(\delta) = 0, \quad \delta \geq \frac{q - 1}{q} \]

- Singleton bound:
  \[ \alpha_q(\delta) \leq 1 - \delta \]

- Hamming bound:
  \[ \alpha_q(\delta) \leq 1 - H_q\left(\frac{\delta}{2}\right) \]

- Gilbert–Varshamov bound:
  \[ \alpha_q(\delta) \geq 1 - H_q(\delta) \]

- Difficult to construct codes above the asymptotic bound:
  Examples from algebro-geometric codes from curves (but only for \( q \geq 49 \) otherwise entirely below the GV curve)
Application to Linguistics: Syntactic Parameters and Coding


- **idea**: assign a (binary or ternary) code to a family of languages and use position of code parameters with respect to the asymptotic bound to test relatedness and to test difference in behavior of syntactic parameters from independent random variables

- $N = \text{number of syntactic parameters } \Pi = (\Pi_\ell)^N_{\ell=1}$
  - each $\Pi_\ell$ with values in $\mathbb{F}_2 = \{0, 1\}$ (or $\mathbb{F}_3 = \{-1, 0, +1\}$ if include parameters that are not set in certain languages)

- $\mathcal{F} = \{L_k\}^m_{k=1}$ a set of natural languages (language “family”)

- **Code** $C = C(\mathcal{F})$ in $\mathbb{F}^N$ ($\mathbb{F}_2^N$ or $\mathbb{F}_3^N$) with $m$ code words
  - $w_k = \Pi(L_k)$ string of syntactic parameters for the language $L_k$
Interpretation of Code Parameters

- $R = R(C)$ measures ratio between logarithmic size of number of languages in $\mathcal{F}$ and total number of parameters: how $\mathcal{F}$ distributed in the ambient $\mathbb{F}^N$

- $\delta = \delta(C)$ is the minimum, over all pairs of languages $L_i, L_j$ in $\mathcal{F}$ of the relative Hamming distance

$$\delta(C(\mathcal{F})) = \min_{L_i \neq L_j \in \mathcal{F}} \delta_H(L_i, L_j)$$

$$\delta_H(L_i, L_j) = \frac{1}{N} \sum_{\ell=1}^{N} |\Pi_\ell(L_i) - \Pi_\ell(L_j)|$$

- code parameter $\delta$ used in Longobardi’s Parameter Comparison Method for reconstruction of phylogenetic trees
Spoiling operations on binary codes

1. $C_1 = C \star_i a$ associates to a word $c = (a_1, \ldots, a_n)$ of $C$ the word $c \star_i a = (a_1, \ldots, a_{i-1}, a, a_i, \ldots, a_n)$

2. $C_2 = C \star_i$, which is a projection of the code $C$ in the $i$-th direction

3. $C_3 = C(a, i)$ code words with same $i$-th digit equal to $a$

Interpretation of Spoiling Operations

- **first spoiling operation**: effect of including one syntactic parameter in the list which is dependent on the other parameters
- **second spoiling operation**: forgetting one of the syntactic parameters
- **third spoiling operation**: forming subfamilies by considering languages that have a common value of one of the parameters
Simple Example:

- group of three languages $\mathcal{F} = \{\ell_1, \ell_2, \ell_3\}$: Italian, Spanish, French using first group of 6 parameters
- code $C = C(\mathcal{F})$

<table>
<thead>
<tr>
<th></th>
<th>$\ell_1$</th>
<th>$\ell_2$</th>
<th>$\ell_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

- code parameters: $(R = \log_2(3)/6 = 0.2642, \delta = 1/6)$
- code parameters satisfy $R < 1 - H_2(\delta)$: below the Gilbert–Varshamov curve
Spoiling operations in this example:

- **first spoiling operation:**
  first two parameters same value 1, so
  \[ C = C' \ast f_1 = (C'' \ast f_2) \ast f_1 \] with \( f_1 \) and \( f_2 \) constant equal to 1
  and \( C'' \subset \mathbb{F}_2^4 \) without first two letters

- **second spoiling operation:**
  conversely, \( C'' = C' \ast f_2 \) and \( C' = C \ast f_1 \)

- **third spoiling operation:**
  \( C(0, 4) = \{l_1, l_3\} \) and \( C(1, 6) = \{l_2, l_3\} \)
What if languages are not in the same historical family?

Example: \( \mathcal{F} = \{ L_1, L_2, L_3 \} \): Arabic, Wolof, Basque

- excluding parameters that are not set, or are entailed by other parameters, for these languages: left with 25 parameters from original list (number 1–5, 7, 10, 20–21, 25, 27–29, 31–32, 34, 37, 42, 50–53, 55–57)

- code \( C = C(\mathcal{F}) \)

| \( L_1 \) | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| \( L_2 \) | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| \( L_3 \) | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

- code parameters: \( \delta = 0.52 \) and \( R > 0 \) violates Plotkin bound
  \( \Rightarrow \) isolated code above the asymptotic bound
Asymptotic bound and language relatedness

• For binary syntactic parameters: a code $C = C(F)$ violates the Plotkin bound if any pair $L_i \neq L_j$ of languages in $F$ has $\delta_H(L_i, L_j) \geq 1/2$

• $L_i$ and $L_j$ differ in at least half of the parameters: it would not happen in a group of historically related languages

• but what about codes above the asymptotic bound that do not violate the Plotkin bound?

• **Expect**: $C = C(F)$ above the asymptotic bound
  $\Rightarrow F$ not a historical language family
  (quantitative test of historical relatedness)
Why the asymptotic bound?

• Why look at position with respect to asymptotic bound as a test of historical relatedness? because it is the only true “bound” in the space of code parameters across which behavior truly changes
• codes below the asymptotic bound are easily deformable (as long as number of syntactic parameters is large)
• if think of language evolution as a process of parameter change, expect languages that have evolved in the same family to determine codes in this zone of the space of code parameters
• codes $C = C(F)$ above the asymptotic bound should be a clear sign that list of languages in $F$ do not belong to same historical family
• though there can be codes $C = C(F)$ below the asymptotic bound that also don’t come from historically related languages: converse implication does not hold
Code parameters of language sets


- take all sets of two and three languages in the SSWL database and set of parameters completely mapped for languages in the set

- for each pair/triple compute the code parameters of the resulting code and plot where they lie in the space of code parameters
• distribution of code parameters for small sets of languages (pairs or triples) and SSWL data
• in lower region of code parameter space a superposition of two Thomae functions \( f(x) = 1/q \) for \( x = p/q \) coprime, zero on irrationals

and behaves like the case of random codes with fixed \( k = \log_2(N) \)

\[
(\delta = \frac{d}{n}, R = k \cdot \frac{1}{n})
\]
randomly chosen sets of two or three languages tend to populate the lower region of the Thomae function graph

uniformly random sets of three languages
• more interesting what happens in the upper regions of the code parameter space
• take larger sets of randomly selected languages and syntactic parameters in the SSWL database

codes better than algebro-geometric above GV, asymptotic, and Plotkin
Remarks

- construction of binary codes above asymptotic bound through linguistics
- what are the best codes obtained this way? explicit examples with languages that are phylogenetically very distant
- these points are rare compared to typical: find explicitly which languages are involved

The Geometry of Syntax (Part 2)
What kind of dynamics of language change can lead to this type of
distribution of syntactic features (and to codes so high above the
GV line)?

• build a simple model of language change based on interaction
between languages (bilingualism, code switching)
• tendency of parameters to align if high interaction (spin glass model)
• if no relations: independent set of uncoupled Ising models on a
  graph, for each syntactic parameter... convergence to most
  prevalent configuration in initial condition
• crucial role of relations between parameters in giving interesting
dynamics and interesting equilibrium configurations (not all
parameters aligned)
Spin Glass Models of Syntax


• historical examples: Sanskrit flipped some syntactic parameters by influence of Dravidian languages...

• physicist viewpoint: binary variables (up/down spins) that flip by effect of interactions: Spin Glass Model
  – focus on linguistic change caused by language interactions
  – think of syntactic parameters as spin variables
  – spin interaction tends to align (ferromagnet)
  – strength of interaction proportional to bilingualism (MediaLab)
  – role of temperature parameter: probabilistic interpretation of parameters & amount of code-switching in bilingual populations
  – not all parameters are independent: entailment relations
  – Metropolis–Hastings algorithm: simulate evolution
The Ising Model of spin systems on a graph $G$

• **graph**: vertices = languages, edges = language interaction (strength proportional to bilingual population); over each vertex a set of spin variables (syntactic parameters)

• configurations of spins $s : V(G) \to \{\pm 1\}$

• if only one syntactic parameter, would have an Ising model on the graph $G$: configurations $s : V(G) \to \{\pm 1\}$ set the parameter at all the locations on the graph

• variable interaction energies along edges (some pairs of languages interact more than others)

$$H(s) = - \sum_{e \in E(G): \partial(e) = \{v,v'\}} \sum_{i=1}^{N} J_e s_{v,i} s_{v',i}$$

• if all $N$ parameters are independent, then it would be like having $N$ non-interacting copies of same Ising model on the same graph $G$
Example: Single parameter dynamics \textit{Subject-Verb} parameter

Initial configuration: most languages in SSWL have $+1$ for \textit{Subject-Verb}; use interaction energies from MediaLab data
Equilibrium: low temperature all aligned to +1; high temperature:

Temperature: fluctuations in bilingual users between different structures ("code-switching" in Linguistics)
Entailment relations among parameters: toy model example

- Example: \( \{ p_1, p_2 \} = \{ \text{Strong Deixis, Strong Anaphoricity} \} \)

<table>
<thead>
<tr>
<th>( \ell )</th>
<th>( p_1 )</th>
<th>( p_2 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \ell_1 )</td>
<td>+1</td>
<td>+1</td>
</tr>
<tr>
<td>( \ell_2 )</td>
<td>−1</td>
<td>0</td>
</tr>
<tr>
<td>( \ell_3 )</td>
<td>+1</td>
<td>+1</td>
</tr>
<tr>
<td>( \ell_4 )</td>
<td>+1</td>
<td>−1</td>
</tr>
</tbody>
</table>

\( \{ \ell_1, \ell_2, \ell_3, \ell_4 \} = \{ \text{English, Welsh, Russian, Bulgarian} \} \)
Modeling Entailment

- **key idea**: change the Hamiltonian (which determines the dynamics) by adding terms that make the configurations \((p_1 = 1, p_2 = \pm 1)\) and \((p_1 = -1, p_2 = 0)\) energetically favored over others

- variables: \(S_{\ell, p_1} = \exp(\pi i X_{\ell, p_1}) \in \{\pm 1\}, S_{\ell, p_2} \in \{\pm 1, 0\}\) and 
\(Y_{\ell, p_2} = |S_{\ell, p_2}| \in \{0, 1\}\) and Hamiltonian \(H = H_E + H_V\)

\[
H_E = H_{p_1} + H_{p_2} = - \sum_{\ell, \ell' \in \text{languages}} J_{\ell \ell'} \left( \delta_{S_{\ell, p_1}, S'_{\ell', p_1}} + \delta_{S_{\ell, p_2}, S'_{\ell', p_2}} \right)
\]

\[
H_V = \sum_{\ell} H_{V, \ell} = \sum_{\ell} J_{\ell} \delta_{X_{\ell, p_1}, Y_{\ell, p_2}}
\]

if freeze \(p_1\) and evolution for \(p_2\): Potts model with external magnetic field
- two parameters: *temperature* (amount of code switching) and coupling *energy of entailment* (how strongly enforced the entailment relations are)

**Equilibrium configuration**

<table>
<thead>
<tr>
<th>$(p_1, p_2)$</th>
<th>HT/HE</th>
<th>HT/LE</th>
<th>LT/HE</th>
<th>LT/LE</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\ell_1$</td>
<td>$(+1, 0)$</td>
<td>$(+1, -1)$</td>
<td>$(+1, +1)$</td>
<td>$(+1, -1)$</td>
</tr>
<tr>
<td>$\ell_2$</td>
<td>$(+1, -1)$</td>
<td>$(-1, -1)$</td>
<td>$(+1, +1)$</td>
<td>$(+1, -1)$</td>
</tr>
<tr>
<td>$\ell_3$</td>
<td>$(-1, 0)$</td>
<td>$(-1, +1)$</td>
<td>$(+1, +1)$</td>
<td>$(-1, 0)$</td>
</tr>
<tr>
<td>$\ell_4$</td>
<td>$(+1, +1)$</td>
<td>$(-1, -1)$</td>
<td>$(+1, +1)$</td>
<td>$(-1, 0)$</td>
</tr>
</tbody>
</table>
Average value of spin

\[ p_1 \text{ left and } p_2 \text{ right in low entailment energy case} \]
• when consider more realistic models (at least the 28 languages and 63 parameters of Longobardi–Guardiano with all their entailment relations) very slow convergence of the Metropolis–Hastings dynamics even for low temperature

• how to get better information on the dynamics? consider set of languages as codes and an induced dynamics in the space of code parameters
Space of Code Parameters and dynamics of syntactic parameters

- Spin Glass Model dynamics for a set of languages $\mathcal{L}$ induces dynamics on codes $C(\mathcal{L})$ and on code parameters $(R, \delta)$
  - **no entailment** (independent parameters): fixed $R$ and $\delta$ flows towards zero (spoiling code)
  - **entailment**: dynamics can improve code making $\delta$ larger ($R$ fixed)

- for large number of parameters see dynamics more easily on code parameter than with average magnetization of spin glass model
Heat Kernel dimensional reduction


- Geometric methods of dimensional reduction: *Belkin–Niyogi heat kernel method*


- **Question**: low dimensional representations of data sampled from a probability distribution on a manifold

- **Want** more efficient methods than Principal Component Analysis

- **Main Idea**: build a graph with neighborhood information, use Laplacian of graph, obtain low dimensional representation that maintains the local neighborhood information using eigenfunctions of the Laplacian
• **setting**: data points \( x_1, \ldots, x_k \in M \subset \mathbb{R}^\ell \) on a manifold; find points \( y_1, \ldots, y_k \) in a low dimensional \( \mathbb{R}^m \) \((m \ll \ell)\) that **represent** the data points \( x_i \);

• **Step 1 (a): adjacency graph** (\( \epsilon \)-neighborhood): an edge \( e_{ij} \) between \( x_i \) and \( x_j \) if \( \|x_i - x_j\|_{\mathbb{R}^\ell} < \epsilon \);

• **Step 1 (b): adjacency graph** (\( n \) nearest neighborhood): edge \( e_{ij} \) between \( x_i \) and \( x_j \) if \( x_i \) is among the \( n \) nearest neighbors of \( x_j \) or viceversa

• **Step 2: weights on edges**: heat kernel

\[
W_{ij} = \exp\left(-\frac{\|x_i - x_j\|^2}{t}\right)
\]

if edge \( e_{ij} \) and \( W_{ij} = 0 \) otherwise; heat kernel parameter \( t > 0 \)
• Step 3: **Eigenfunctions** for connected graph (or on each component)

\[ L\psi = \lambda D\psi \]

diagonal matrix of weights \( D_{ii} = \sum_j W_{ji} \); Laplacian \( L = D - W \) with \( W = (W_{ij}) \); eigenvalues \( 0 = \lambda_0 \leq \lambda_1 \leq \cdots \leq \lambda_{k-1} \) and \( \psi_j \) eigenfunctions

\[ \psi_i : \{1, \ldots, k\} \to \mathbb{R} \]

defined on set of vertices of graph

• Step 4: **Mapping by Laplace eigenfunctions**

\[ \mathbb{R}^\ell \supset \mathcal{M} \ni x_i \mapsto (\psi_1(i), \ldots, \psi_m(i)) \in \mathbb{R}^m \]

map by first \( m \) eigenfunctions

• Belkin–Niyogi: *optimality* of embedding by Laplace eigenfunctions
Heat Kernel analysis of Syntactic Parameters

- Connectivity in $\epsilon$-neighborhood and nearest-neighbor (difference between SSWL data (json) and Longobardi data (csv))
Graphs with $\epsilon$-neighborhood Longobardi data

Epsilon-Neighbourhood, $\epsilon = 1.000000$

Epsilon-Neighbourhood, $\epsilon = 8.000000$
Graphs with $\epsilon$-neighborhood Longobardi data

Epsilon-Neighbourhood, $\epsilon = 15.000000$
Graphs with $\epsilon$-neighborhood Longobardi data

Epsilon-Neighbourhood, $\epsilon = 22.000000$
Closer look at some of these structures

Examples of parameters in this structure: DMG (def. matching genitives), GCO (gramm. collective number), GST (grammaticalised Genitive) ...
Closer look at some of these structures

The same structure looked at a different $\epsilon$-scale has acquired further connections to other parameters. Most vertices in these structures have high centrality in the graph.
Closer look at some of these structures

Another structure in the Longobardi data involving parameters like EZ2 (non-clausal linker), FGC (gramm. classifier), FGT (gramm. temporality), GSI (grammaticalised inalienability), HMP (NP-heading modifier) ...
Graphs with $\epsilon$-neighborhood SSWL data

Epsilon-Neighbourhood, $\epsilon = 15.000000$
The $\epsilon$-neighborhood construction is better suited to gain connectivity information in the Longobardi data: the SSWL data remain highly disconnected (only small local structures).
In SSWL $\varepsilon$-structures only involve closely related parameters (Neg7,8,9 expressed in some Niger-Congo languages)

More interesting results using the $n$-neighborhood method
Graphs with $n$-neighborhood Longobardi data
Fewer vertices with high centrality: FGM node (gramm. Case)
Graphs with $n$-neighborhood Longobardi data
More high centrality vertices: AST (structured APs), CGB (unbounded sg N), FGP (gramm. person) ...
Graphs with $n$-neighborhood SSWL data

Two main components, lower one highest centrality SubjectVerb

Nearest 1 Connections
Graphs with \( n \)-neighborhood SSWL data components merge, high centrality SubjectVerb, VerbObject

Nearest 2 Connections
Clustering behavior of the $\epsilon$-graphs

- neighborhood $V_i$ of $i$-th node of valence $d_i$

$$K_i = \#\{e \in E(G) : \partial(e) \cap V_i\}$$

measure of clustering in the region $C_i = K_i/(d_i^2)$

- variance of clustering coefficients $C_i$ as function of $\epsilon$
\( \epsilon \) and \( n \)-neighborhood clustering and connectivity

**Epsilon Clustering and Connectivity**

**Neighbor Clustering and Connectivity**
Regions of $\epsilon$-$t$ space

- Graphs depend on $\epsilon$-neighborhood and on $t$-heat kernel variable
- explore $\epsilon$-$t$ space: determine regions where high variance in distribution of each parameter under the heat kernel mapping
- high variance in a parameter suggests it is highly independent (similar to PCA method)
Further Questions

- an in depth linguistic analysis of the meaning of these clustering structures is still needed (ongoing work)
- comparison of the heat kernel technique with other dimensional reduction techniques (PCA etc.)
- more detailed discussion of different regions of the $\epsilon$-$t$ space in the heat kernel model (specific parameters with high independence measure)
- manifold $\mathcal{M}$ reconstruction? Belkin-Niyogi results
Conclusions (for now)

- import a set of different mathematical techniques (phylogenetic algebraic geometry, persistent topology, coding theory, statistical mechanics, geometric models of associative memory) in order to study natural languages as dynamical objects

- longer term goals: create mathematical and computational models of
  1. how languages are acquired?
  2. how languages are stored in the brain?
  3. how languages change and evolve dynamically in time?

For human languages viewed at the level of their syntactic structures
Further Related Work

- Algebro-Geometric Models of Computational Semantics

- Generative Grammars and Renormalization