Language Evolution and Diachrony Generation

Research Project Report

Matthieu Boyer

Laboratoire Lattice

CNRS — ENS-PSL — Université Sorbonne Nouvelle

Under supervision of Mathieu Dehouck

Introduction

Few database available in diachrony :

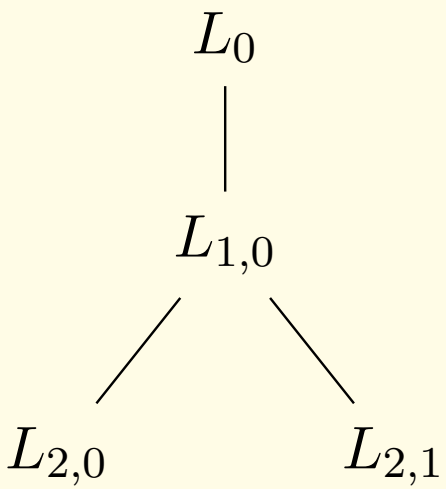
- The Index Diachronica [ind]
- The $\mathcal{E}$vosem [FKD$^+$25]

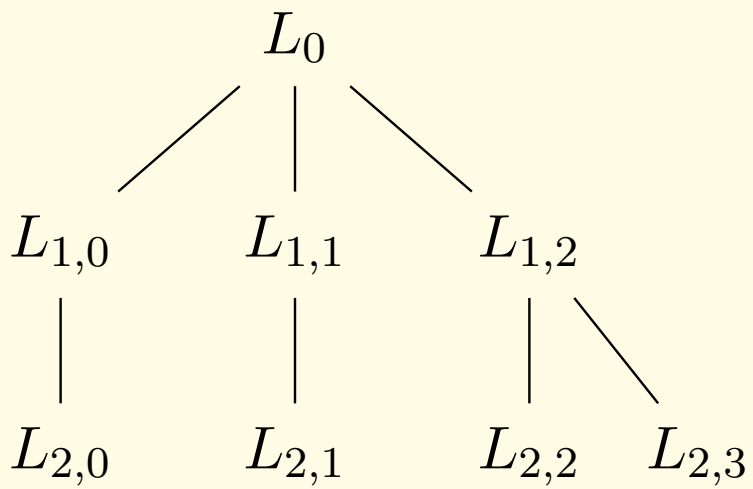
There is a need for less localized data.

Plan

Evolution as Random Trees I

Consider a language L_0, which we will call our *base language*.





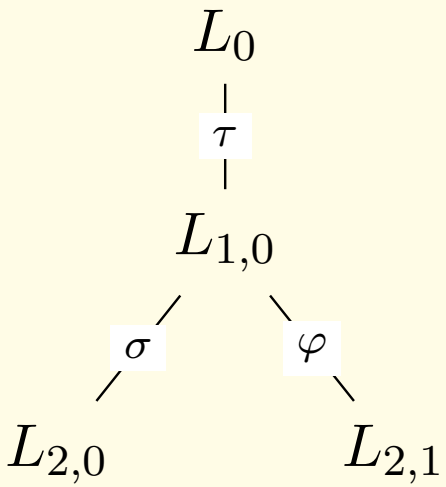
Evolution as Random Trees II

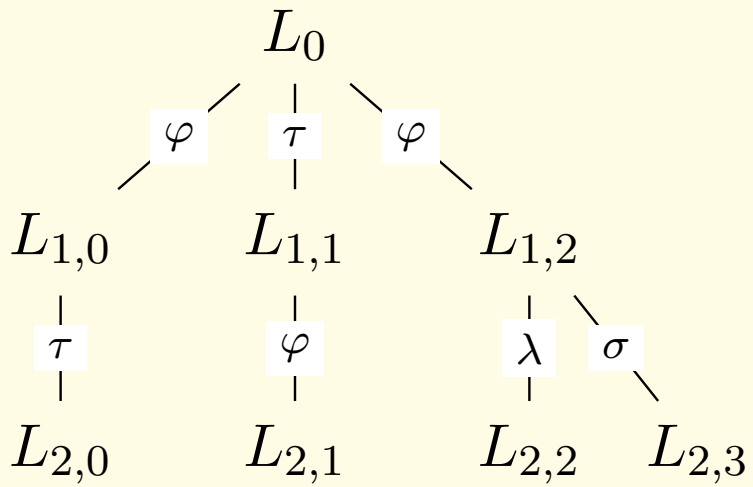
Algorithm One Language Evolution

$leaves \leftarrow \{L_0\}$
$\mathcal{T} \leftarrow \mathrm{Tree}\,(L_0, \varnothing)$
for $n \leq$ `Epochs` **do**
    for $l \in \mathrm{Leaves}\,(\mathcal{T})$ **do**
        $S \leftarrow \mathrm{Evolve}(l)$
        $l \leftarrow \mathrm{Tree}\,(l, S)$ ▷ Here $\mathcal{T}$ is modified in place.
return $\mathcal{T}$

Specification of Evolve I

We want to choose between *evolution types* $(\varphi, \sigma, \tau, \lambda)$ at computation :

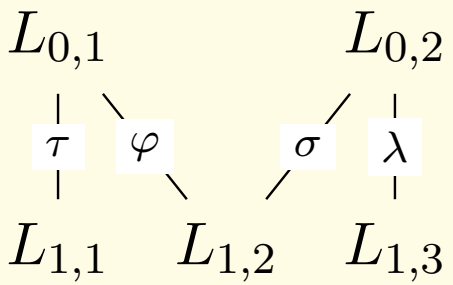




Plan

Collision Hypothesis I

We assume language interacting create evolutions :



Collision Hypothesis II

Algorithm Two Language Evolution

$\mathcal{T} \leftarrow \text{Tree}\,(L_0, \varnothing)$
for $n \leq$ `Epochs` **do**
  for $l \in \text{Leaves}\,(\mathcal{T})$ **do**
    $S \leftarrow \text{Evolve}(l)$
    $\text{Push}\,(\text{Stack}, l \leftarrow l \cup \text{Tree}\,(l, S))$
  for $l \in \text{Leaves}\,(\mathcal{T})$ **do**
    $l^\dagger \leftarrow \mathcal{P}_l\,(\text{Leaves}\,\mathcal{T})$
    $S \leftarrow \text{Collision}\,(l, l^\dagger)$
    $\text{Push}\,(\text{Stack}, l \leftarrow l \cup \text{Tree}\,(l, S))$
  $\text{Apply}(\text{Stack})$
return $\mathcal{T}$

Specification of Collision

Collision should :

- provide a way to choose collision type, for each parent.

Specification of Collision

Collision should :

- provide a way to choose collision type, for each parent.
- take *linguistic* proximity of the parents into account.

Specification of Collision

Collision should :

- provide a way to choose collision type, for each parent.
- take *linguistic* proximity of the parents into account.
- take the probability distribution $\mathcal{P}$ as the strength of the collisions.

A Manifold of Languages I

$\mathcal{P}_l$ models the probability of interaction with l. Defining a *geographical* embedding gives :

$$\mathcal{P}_l \propto \frac{1}{d_l(x)}$$

A Manifold of Languages II

- The simplex, that is, $d_l(x) = 1$ for all l, x.
- $\mathbb{R}^3$, with the ℓ^2 distance.
- The 2-sphere $\S^1$ where each language is a pairλ, φ :

$$d_{(\theta_1,\lambda_1)}(\theta_2, \lambda_2) = \arccos \sin(\varphi_1) \sin(\varphi_2) + \cos(\varphi_1) \cos(\varphi_2) \cos(\lambda_2 - \lambda_1)$$

A Manifold of Languages III

We suppose a language only interacts with languages from the same epoch, for now. We could add a new dimension to the manifold to modelize time.

Moreover, we are not required to use a metric but simply a positive separated function.

Implemented I

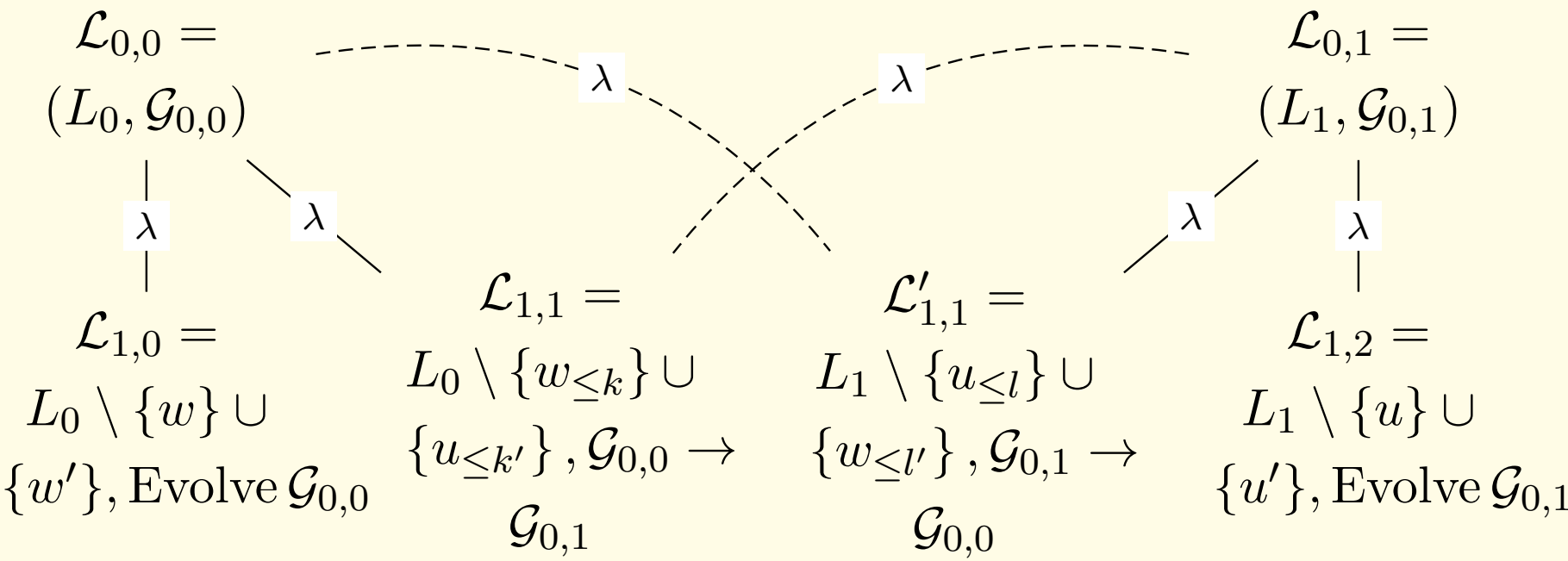
We worked using a modular structure :

- A tree generation module
- A *linguistical* observable
- A *geographical* observable

We have implemented all 3 previously defined *geographical* observable.

Implemented II

We only have a lexicon observable :



Implemented III

For our probabilities, we set thresholds :

- $1 - \alpha$ representing our random evolution probability ;
- $1 - \beta$ representing our collision generation probability.

Our problem is then to find *optimal* parameters
$\alpha, \beta, \mathrm{Evolve}, \mathrm{Collision}$ and distribution $\mathcal{P}$.

Performance Checking I

We use the $\mathcal{E}$vosem project [FKD$^+$25] as a lexical bank :

- Our base languages are two proto-families, Germanic and Indo-European which derived in modern French, German, English, Dutch, Spanish, Italian and Danish.
- Accuracy is computed by the ℓ^2 distance between subsets of the shared ancestry matrices.
- Limitation of randomness is done by repetition of the experiments, though there are $\mathcal{O}\left(3^{7d}\right)$ submatrices.

Performance Checking II

However, our algorithm is quite slow. Assuming :

- Evolve is done in constant time (false for phonetics, for example).
- $\mathrm{Collision}$ is done in constant time.
- Computing $d(x, y)$ is done in constant time.
- Loss computation is in constant time.

we get a complexity in $\mathcal{O}\left(3^{2d}k\right)$ for d epochs and k base languages, to multiply by the number of repetitions and parameters.

Results I

We take for base languages for the euclidean space :

$$\left(\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \right)$$

and for the sphere $\S^1$ we take the GPS coordinates of Paris and Berlin.

Results II

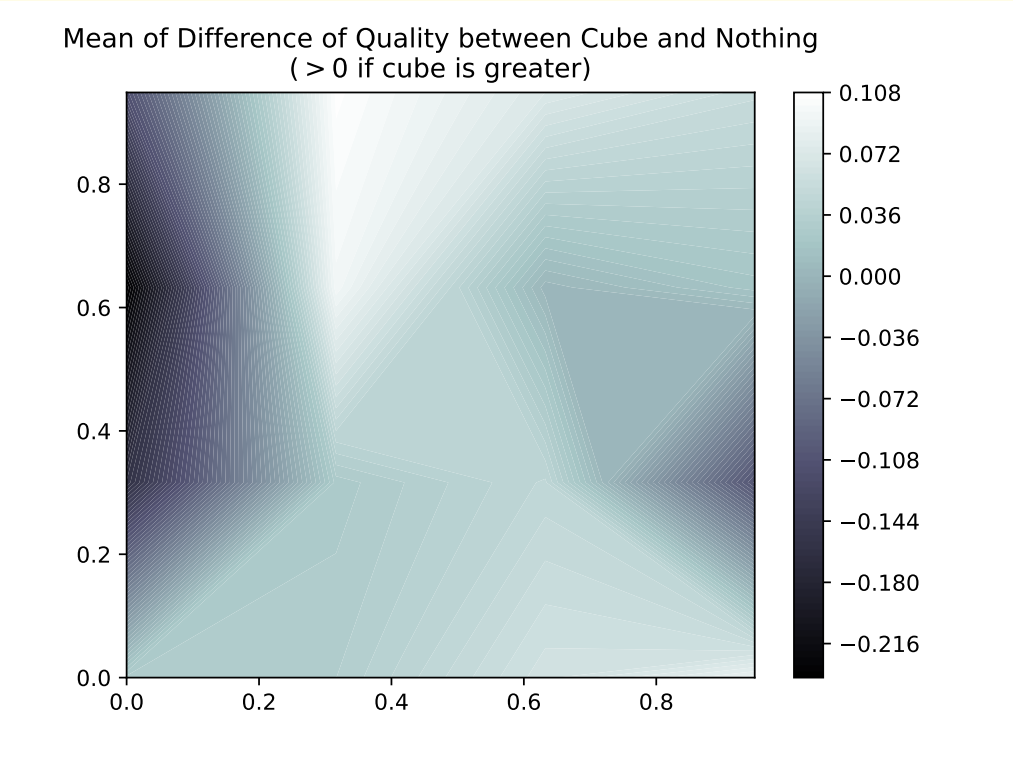


Figure – Plot of computed differences in accuracy between the euclidean space geography and no geography. Variance is around $\sqrt{2.10^{-2}}$.

Results III

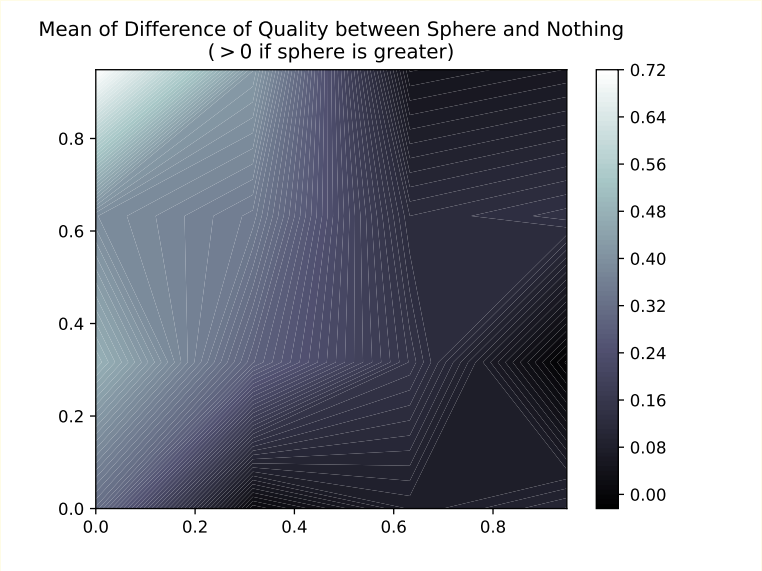


Figure – Plot of computed differences in accuracy between the $\S^1$ sphere and no geography.

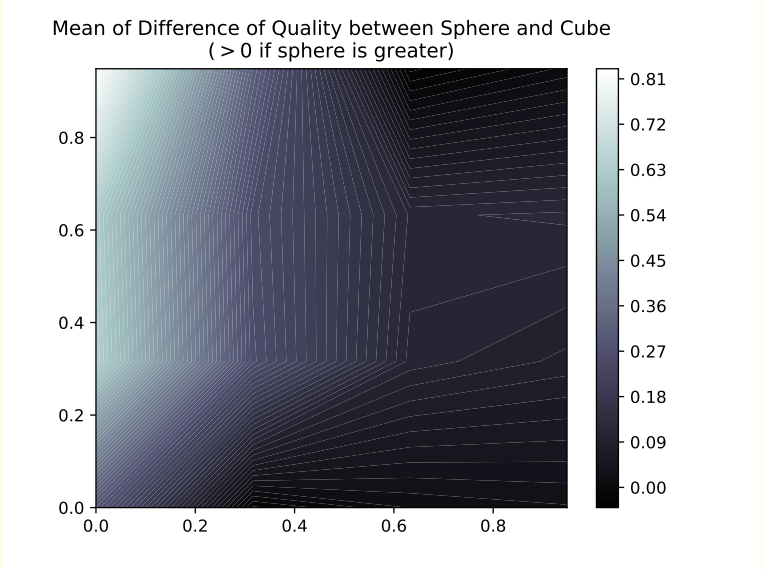


Figure – Plot of computed differences in accuracy between the euclidean space geography and the $\S^1$ sphere.

References

Alexandre François, Siva Kalyan, Mathieu Dehouck, Martial Pastor, and David Kletz.

Evosem : A database of dialexification across language families.

Online database., 2025.

Index diachronica.

`https://chridd.nfshost.com/diachronica/`.