Software Project NLP with NNs

Proto-Word Reconstruction with RNNs
Project Sketch

Julius Steuer   Morgan Wixted

August 15, 2020

Goal of the Project

- Provide a tool to automatically reconstruct proto-words for a given sample of cognate sets
- Initial hypothesis if not much is known about the family
- Allow for integration of linguistic knowledge
  - Alignment between cognate loci in different languages
  - Exclude word parts assumed to be innovations (e. g. spanish *nos-otros* from latin *no:s*)
- Examine the influence of input representations on model performance (alignments, transcriptions)
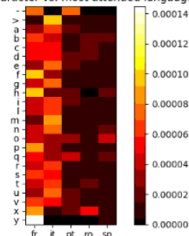
Baseline

Start from Meloni et al. (2019):

- Romance dataset (not publicly availabe)
- Character-based encoder-decoder architecture with Bahdanau attention
- Characters encoded as 100-bit vectors (localist representation)
- Output one character of the reconstructed word per timestep
- Evaluate impact of individual languages on reconstruction:
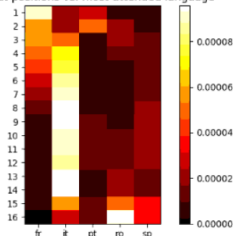
Model

Idea:

– Use distributed feature encodings to represent characters, not embeddings

– Use the stem of the Latin word as the proto-form, since nominative inflection is seldom preserved in modern Romance

– Start with ASJP (easily available) transcriptions (Brown et al. (2008)), later switch to IPA (richer annotation).

– Pairwise alignments of sequence chunks (following Ciobanu and Dinu (2018))

Latin acc. sing *corticem* 'bark' $\rightarrow$ ASJP {kortike}									
	1	2	3	4	5	6	7	8	9	10
latin	-	k	o	-	r	t	i	k	-	e
italian	-	k	o	-	r	t	e	t	C	a
spanish	-	k	o	-	r	t	e	8	-	a
french	e	k	o	-	r	-	-	-	-	-
romanian	s	k	o	a	r	-	-	c	-	3

Model, feature encoding

ASJP consonant feature encodings

segment	voiced	labial	dental	alveolar	palatal	velar	uvular	glottal	stop	fricative	affricate	nasal	click	approximant	lateral	rhotic
p	0	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0
b	1	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0
f	0	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0
v	1	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0
m	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0
8	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0
4	1	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
t	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
d	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
s	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0
z	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0
c	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0
n	1	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0
S	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0
Z	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0
C	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0
j	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0
T	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0
5	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
k	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0
g	1	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0
x	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0
N	1	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0
q	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
G	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
X	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0
7	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
h	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0
l	1	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0
L	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
w	1	1	0	0	0	1	0	0	0	0	0	0	0	1	0	0
y	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0
r	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
!	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

ASJP vowel features

segment	high	mid	low	front	central	back	rounded
i	1	0	0	1	0	0	1
e	0	1	0	1	0	0	1
E	0	0	1	1	0	0	1
3	1	1	0	0	0	1	0
a	0	0	1	0	1	0	0
u	1	0	0	0	0	1	1
o	0	1	1	0	0	1	1

Model, continued

Input:

– One column represents the input at a single timestep:

$$I_{T=t_1} = (-, -, -, e, s)$$
$$I_{T=t_2} = (k, k, k, k, k)$$
$$\cdots$$
$$I_{T=t_{10}} = (e, a, a, -, 3)$$

– Attention: Meloni et al. (2019) attend on different languages in the cognate set.
In our case it depends on how whe present the data to the model: Either (as in the paper) perform a pass through the model to reconstruct a single character (each $I_{T=t_i}$ a sequence of inputs), or

– Use the matrix/vector $I_{T=t_i}$ at a single time step, and perform only one pass

Model, limitations

But: The direct precursor of italian *corteccia*, spanish *corteza* etc. is the latin adjective *corticeus, -a, -um*

- We expect the model to reconstruct an a-stem noun instead of a consononat stem cortex, cortices
- Spurious sounds (french *e-*, romanian *s-*) should be dropped (?)
- Ambiguous sounds may not indicate a clear rule (latin *-k-* vs. rom. *-c-* vs. spanish *-θ-*)

And: We have to produce alignments ourselves (manually)

Milestones

Minimal:

- Reconstruction with ASJP encodings, small Swadesh list as data
- ~~Then switch to IPA encodings~~ Using the epitran [1] Python package to get IPA transcriptions
- Try different language family (initial guess on unseen data)
- Examinate the influence of differrent (or absent) alignments on model performance

Great to have:

- ~~Use larger dataset (ask Ciobanu for hers)~~ Use Ciobanu's dataset from 2014
- Ensure compatibility with the LingPy [2] WordList [3] class
- Use LingPy to work with pre-aligned data (Austronesian)

---

[1] https://github.com/dmort27/epitran

[2] http://lingpy.org/

[3] http://lingpy.org/reference/lingpy.basic.html#lingpy.basic.wordlist.Wordlist

References

Brown, C. H., Holman, E. W., Wichmann, S., and Velupillai, V. (2008). Automated classification of the world's languages: a description of the method and preliminary results. Language Typology and Universals, 61(4):285–308.

Ciobanu, A. M. and Dinu, L. P. (2018). Ab Initio: Automatic Latin Proto-word Reconstruction. In Proceedings of the 27th International Conference on Computational Linguistics, pages 1604–1614, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Meloni, C., Ravfogel, S., and Goldberg, Y. (2019). Ab Antiquo: Proto-language Reconstruction with RNNs. arXiv: 1908.02477.