Unsupervised Cognate Identification with Variational Autoencoders

Marlon Betz

August 21, 2016

Contents

1 Introduction **3**

2 Sound Change as a Walk in Latent Space **4**
2.1 Motivation 4
2.2 $P(z|X)$ as Prior for Evolutionary Stable Word Forms 5
2.3 Sound Change as Posterior $P(w_{recent}|w_{ancient}, z)$ 6

3 Related Research **8**

4 Architecture **8**
4.1 Phoneme Vectorization 8
4.1.1 Hand-crafted Vectorization Models 8
4.1.2 Data-driven Embeddings 8
4.2 Word Embeddings 9
4.2.1 Autoencoders 9
4.2.2 Variational Autoencoders 9
4.3 Clustering 11
4.3.1 Affinity Propagation 11

5 Evaluation **11**
5.1 Data 11
5.2 Results 11

6 Resume **11**

7 Acknowledgements **11**

1 Introduction

Historical Linguistics investigates language from a diachronic perspective, i.e. it seeks to uncover the history of languages and the structure of the hidden forces that drive language change. Computational Historical Linguistics accordingly deals with computational methods to explore the history of languages and topics closely related to it, such as phylogenetic inferences of language families [Bouckaert et al.2012], migration of language speakers [Gray et al.2009], inferring lexical flows between languages [Dellert2015] or modeling sound change [Bouchard-Côté et al.2013].

On the other hand, deep neural networks have been proven to uncover latent features of data and use them for a variety of tasks. Deep Autoencoders perform well on information retrieval tasks.

However, Computation Historical Linguistics has hardly been touched yet by the current Deep Learning boom (a notable exception is [Rama2016]). The aim of this thesis is hence

1. to combine methods from both Computational Historical Linguistics and the emerging field of Deep Learning

2. to propose a model of modeling sound change as a walk in latent space, which is suitable for neural networks

3. to use variational autoencoders as a means to uncover the the latent structure that describes the connection between the phonological shape and the meaning of a given word, as well as the geographical location of the speakers of the language the word belongs to, and to investigate it

4. to show how this uncovered structure can be used to identify cognates in an unsupervised way

I will first start by giving an overview of the problem of cognate identification and related fields of research. I will give a background on why cognate identification is important to discuss the historical connections between languages and further provide an overview on several established methods to detect cognates.

Then I will proceed to introduce the concept of sound change as a walk in latent space, which serves as a background for the actual inference model. Here I will talk about the main motivations for this approach as well as its major drawbacks.

I will then discuss the actual architecture of the inference model. This first covers a general overview on the components included in the model. I will then in detail look over all components in particular. That will first cover a discussion of different methods of phoneme vectorizations. This is followed by a general overview on autoencoders as non-linear dimensionality reduction architectures first and then a description of variational autoencoders in particular, which build the backbone of the model described here. I then come to the discussion of possible ways to cluster the words, i.e. to assign the actual inferred cognacy labels.

Then I will document how well those methods can be used to infer cognacy between words. I will compare the inferred labels with expert judgements first and then see how the inferred labels can infer language phylogenies, using established bayesian models of cognate evolution.

Finally, I will give a resume on the model described here.

2 Sound Change as a Walk in Latent Space

2.1 Motivation

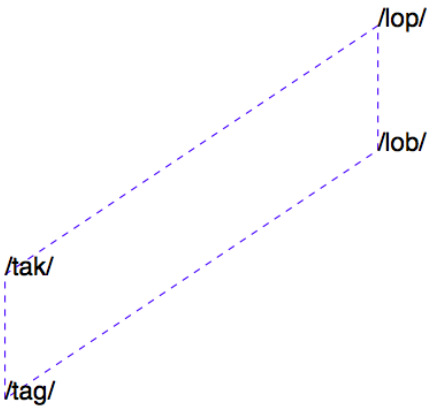

Figure 1: Visualization of the concept of sound change as a walk in latent space. Here, sound changes are vectors from one word to another, where both word forms are given as points in latent space. The vector from /lob/ to /lop/ would be the same as from /tag/ to /tak/, as both vectors would describe the loss of voice in the word-final phoneme. This linear dependence means that if we fit a regression model from /tag/ to /tak/, we could generalize well to predict /lop/ from /lob/. The different lengths of the vectors are than proportional to probabilities of such as sound change to appear. Here, final devoicing should be more probable than a change from /lop/ to /tak/.

In linguistic literature, sound change is usually described as a change of distinctive phonological features over phonological symbols. For instance, a sound change such as final devoicing as in

$$MHG/hund/ \rightarrow NHG/hunt/ \tag{1}$$

would be captured by an string edit rule like

$$/d/ \rightarrow /t/ \_\# \tag{2}$$

that says that /d/ becomes /t/ before the end of a word. Such sound changes are shown not to appear randomly, but to co-occur according to some latent features. E.g. a sound change like above should leave the language without final /d/, while final /b/ and /g/ would be allowed. Such situations are evolutionary highly unstable, so we would expect the other voiced plosives to change to /p/ and /k/ accordingly. This is due to latent phonological features such as [VOICE], [LABIAL] or [ATR], which are usually described to appear either in a binary or privative feature space. Accordingly, our rule from above could be generalized to

$$[+\text{VOICE}] \rightarrow [\text{-VOICE}]/\_\# \tag{3}$$

which describes the loss of voice at the end of a word.

Computational models of sound change usually treat words as string of symbols and sound change as substitution of substrings.[Bouchard-Côté et al.2007, Bouchard-Côté et al.2013] use a generative model that learns to substitute substrings in some latent order and achieve good performances on the reconstruction of ancient forms of words as well as on cognate identification. However, such a method has two major drawbacks: On the one hand, it needs the topology of the languages phylogeny to be known beforehand. This problem does not exist for certain language families with established tree topologies, but can be problematic if language relationships are not totally clear. On the other hand, such purely symbolic models need lot of data to generalize well. [Bouchard-Côté et al.2007] for example underline that their model cannot model chain shifts as such.

Coming from rules that use distinctive phonological features as in Eq. 3, we can encode every word as a sequence of phonemes that are points in some phonological feature space $\mathcal{F}$, where each dimension corresponds to phonological features such as [VELAR] or [APPROXIMANT].

If we have established such a latent feature space $\mathcal{Z}$ over $\mathbb{R}^n$, a sound change sc that derives a recent form of word w_{recent} from an ancient form $w_{ancient}$ should then correspond to a vector v_{sc} in such a way that

$$v_{w_{ancient}} + v_{sc} = v_{w_{recent}} \tag{4}$$

From this follows that

$$v_{sc} = v_{w_{recent}} - v_{w_{ancient}} \tag{5}$$

That is, we can formulate sound changes such as 3 without neither the actual sound change nor the conditioning specifying where the sound change should apply, but only the respective words involved. If we further assume that that sound change affects another word w', we have

$$v_{w_{recent}} - v_{w_{ancient}} = v_{w'_{recent}} - v_{w'_{ancient}} \tag{6}$$

which is equivalent to

$$v_{w_{recent}} = v_{w_{ancient}} + (v_{w'_{recent}} - v_{w'_{ancient}}) \tag{7}$$

If we want to evaluate that latent feature space, we investigate in how far that compositional structure is preserved in our latent space. In fact, such analogy tasks can be used as an evaluation method to test whether the learned embedding space encodes the structure expected to be inherently contained in the data [Mikolov et al.2013b].

2.2 $P(z|X)$ as Prior for Evolutionary Stable Word Forms

The question is then how to construct such a latent space $\mathcal{Z}$. If we know that all words share a common structure, such as syllables and their respective internal structure, we can assume that it should be possible to model that structure by a latent variable z embedded in our latent space. As z walks through $\mathcal{Z}$, we assume that the resulting word x changes in a *meaningful* way - for instance it should change certain phonological features of some phonemes in x given that such changes are plausible given the context of the other phonemes in the words. It should also be possible to add whole new affixes, but it should not inherit implausible phoneme sequences or change phoneme

features in some random way. Given that we have a decoder function $f_{decode} : \mathcal{Z} \rightarrow \mathcal{X}$, we hence except that f_{decode} is non-linear, as a linear decoder function should not allow for those desired properties. This non-linearity than leads to the situation that similar points in $\mathcal{Z}$ should not necessarily coincide with similar points in our data space $\mathcal{X}$. For instance, a fictional word such as /aban/ should be closer to /ovã/ than to /a‡an/ in $\mathcal{Z}$ but not necessarily in $\mathcal{X}$, as it should be more probable that intervocalic voiced plosives lenite to fricatives and syllable-final nasals drop but leave some compensatory nasalization - here even at the same time - but there should be hardly any evidence of a direct sound change from /b/ to a palatal click.

That is, we are interested in the posterior distribution of z - $P(z|X)$, i.e. the probability of z given our data X. We expect the posterior to be highly multimodal, as we expect e.g. that certain phoneme sequences that appear more often attract more probability density than less frequent ones. That is, we want to cluster words under the assumption that certain words are *inherently evolutionary stable sequences of phonemes* and hence appear more often, independently of their respective history. It is well known, for example, that syllables without codas are much more frequent than syllables without onsets.

To model the posterior $P(z|X)$, we can use Bayes' theorem:

$$P(z|X) = \frac{P(z)P(X|z)}{\int P(z)P(X|z)dz} \tag{8}$$

Here, $P(z)$ is the prior probability of z and $P(X|z)$ the likelihood of our data given our latent variable. The term in the denominator is the marginal probability $P(X)$ of our Data under the given model. If we want to model z with the help of X, our objective is to maximize that marginal probability of our word list $P(X)$:

$$P(X) = \int P(z)P(X|z)dz \tag{9}$$

We further assume that $P(X|z)$ can be modeled by some output distribution and its parameters θ. This output distribution can be any distribution with continuous parameters, as they should be differentiable to allow for the backpropagation of error gradients through the model.

As we expect that the words in our word list follow the central limit theorem and in the long run should be captured by a normal distribution, our latent variable z is sampled from a isotropic multivariate normal prior:

$$z \sim \mathcal{N}(0, I) \tag{10}$$

where I is the identity matrix. To arrive at such linear dependencies as in Eq. 4, we then just have to expect them in our data X in some way - e.g in form of predefined phoneme features as in [Kondrak2000, List2012, Rama2016], pre-trained phoneme embeddings or as simple phoneme unigrams, in which case we would have the system learn latent phoneme features on its own. As X in our model would be conditioned on z, we expect the same linear dependencies over words in z as we find them in X.

2.3 Sound Change as Posterior $P(w_{recent}|w_{ancient}, z)$

As we usually do not know the ancient version of a word but only recent ones, we expect the cognates to have spread through the latent space from the original ancient word form to some directions. Since we assume that bigger jumps from one point in $\mathcal{Z}$ are less likely than small ones,

we can expect that recent versions of an ancient word form should still be in the vicinity of the ancient word. Under the simplifying assumption that words in our word list did not jump from one semantic concept to another, this allows us embed all words of a given semantic concept into $\mathcal{Z}$ and to cluster them. This is then what this thesis focusses on - we embed words in $\mathcal{Z}$ and cluster the embeddings, as we expect cognates to appear close to each other.

However, there are some problems with this approach. The actual distribution of recent word forms in $\mathcal{Z}$ given the ancient forms would be given by the posterior

$$P(w_{recent}|w_{ancient}, z) = \frac{P(w_{recent}|z)P(w_{ancient}|w_{recent}, z)}{\int P(w_{recent}|z)P(w_{ancient}|w_{recent}, z)dw_{ancient}} \tag{11}$$

Unfortunately, we only know $P(w_{recent}|z)$, while we do not know $P(w_{ancient}|w_{recent}, z)$ directly, i.e. we cannot say anything specific about the likelihood of an ancient word given the recent words, that is the distribution of probabilities over points in $\mathcal{Z}$ where an ancient word form could be, given the distribution of its own child words. However, if we only consider those two terms and ignore the normalizing constant in the denominator, we have

$$P(w_{recent}|w_{ancient}, z) \propto P(w_{recent}|z)P(w_{ancient}|w_{recent}, z) \tag{12}$$

This means that if our prior $P(w_{recent}|z)$, which is known to us, has low probability density for the area around w_{recent}, we know that $P(w_{recent}|w_{ancient}, z)$ should also be in such a low probability density area. As we except that the likelihood $P(w_{ancient}|w_{recent}, z)$ itself prefers points in $\mathcal{Z}$ that are close to w_{recent} over points that are far away from it, we can say that w_{recent} should be close to $w_{ancient}$. However, if our prior $P(w_{recent}|z)$ is in an area with high probability density, the likelihood does not have such a regulating effect, as the high prior probability would allow for many points in Z to a possible point for $w_{ancient}$.

If we image that we have a word such as /o/ , we can hardly make any claims about how some ancient form would look like - it could come from /a$\overset{\cup}{\underset{\frown}{}}$t/ (Latin to Italian), /ok/ (Old Norse to Bokmål), /ob/ (Proto-Slavic to Slovak) or /ak$^{\text{w}}$a/ (Latin to French). This is because /o/ as such receives high prior probability $P(w_{recent}|z)$ and hence gives high posterior probabilities over many possible ancient forms. However, if we have a word such as Polish /swuxat$\widehat{\text{c}}$/ "listen", the remarkable structure of that word allows us to make much more detailed assumptions about the underlying ancient form, as we except more complex structures to be less frequent in our data and hence attract less probability density.

2.4 Cognate Identification as Hypothesis Comparison

If we want to compare the two hypotheses directly and compute the Bayes Factor, we have

$$\frac{P(w_{ancient} = w'_{ancient}|w_{recent}, w'_{recent})}{P(w_{ancient} \neq w'_{ancient}|w_{recent}, w'_{recent})} \tag{13}$$

$$= \frac{P(w_{ancient} = w'_{ancient})P(w_{recent}, w'_{recent}|w_{ancient} = w'_{ancient})}{P(w_{ancient} \neq w'_{ancient})P(w_{recent}, w'_{recent}|w_{ancient} \neq w'_{ancient})} \tag{14}$$

That is, we would have to define priors for both hypotheses. We cannot simply assume that both hypotheses are equally likely and share a flat prior, as we expect $P(w_{recent}, w'_{recent}|w_{ancient} = w'_{ancient})$ to be different to $P(w_{recent}, w'_{recent}|w_{ancient} \neq w'_{ancient})$.

Another critical point is that if we have to ancient words $w_{ancient}$ and $w'_{ancient}$, the distributions that model their respective child words can show large overlaps. That is, only because two word look the same, they do not have to be related. This is the major challenge of any cognate identification system. However, as we assume that

3 Related Research

4 Architecture

Hence, the model should have three major components:

1. The phonemes should be embedded in a feature space, where similar phonemes should cluster in similar subspaces of the feature space.
2. The words as sequences of such phoneme embeddings should themselves be embedded in another feature space, where words with similar shape should cluster among each other.
3. The word embeddings are then clustered in such a way that words that appear together in a cluster are assigned a common label, which is then predicted cognate class.

4.1 Phoneme Vectorization

4.1.1 Hand-crafted Vectorization Models

4.1.2 Data-driven Embeddings

If we assume that phonemes do not change unconditionally nor randomly, but instead only change distinctive features given the context, it should be able to embed phonemes in a latent space where local subspaces contain clusters of phonemes that appear in similar environments.

There are several families of algorithms that perform such an embedding. Earlier models are based on factorized co-occurrence matrices, such as Latent Semantic Analysis [Landauer et al.2013]. This approach is inherently intuitive, as the factorized context of a given phoneme would then be taken as point in latent space, so similar contexts than inherently lead to proximity in latent space. However, over the past few years, more recent neural embedding models such as word2vec [Mikolov et al.2013a, Mikolov et al.2013b, Goldberg and Levy2014] have been shown to outperform those count-based models, although GloVe [Pennington et al.2014], as more recent count-based model, seems to achieve similar performance.

That is, we train a model that either predicts the phoneme given its context or vice versa.

To evaluate if such data driven phoneme embeddings are able to capture the natural distinction between phoneme classes, an test set over 108 analogy tests was used to conduct a grid search over several word2vec architectures and their parameters. Each such analogy test was of the form

$$v(phoneme_1) + v(phoneme_2) - v(phoneme_3) \approx v(phoneme_4) \tag{15}$$

where $v(phoneme_1)$ is the vector corresponding to $phoneme_1$ and $v(phoneme_2) - v(phoneme_3)$ can be seen as the latent phonological information available in $phoneme_2$ but not in $phoneme_3$, while $v(phoneme_1)+v(phoneme_2)-v(phoneme_3)$ is then this latent phonological information added to $phoneme_1$, which should be close to the vector corresponding to $phoneme_4$. For example, in

$$v(/\mathbf{i}/) + v(/\mathbf{u}*/) - v(/\mathbf{u}/) \approx v(/\mathbf{i}*/) \tag{16}$$

$v(/\mathbf{u}*/) - v(/\mathbf{u}/)$ describes a latent feature that should correspond to [+NASALIZED], which is then added to the phoneme /**i**/ to yield a nasalized /**i***/.

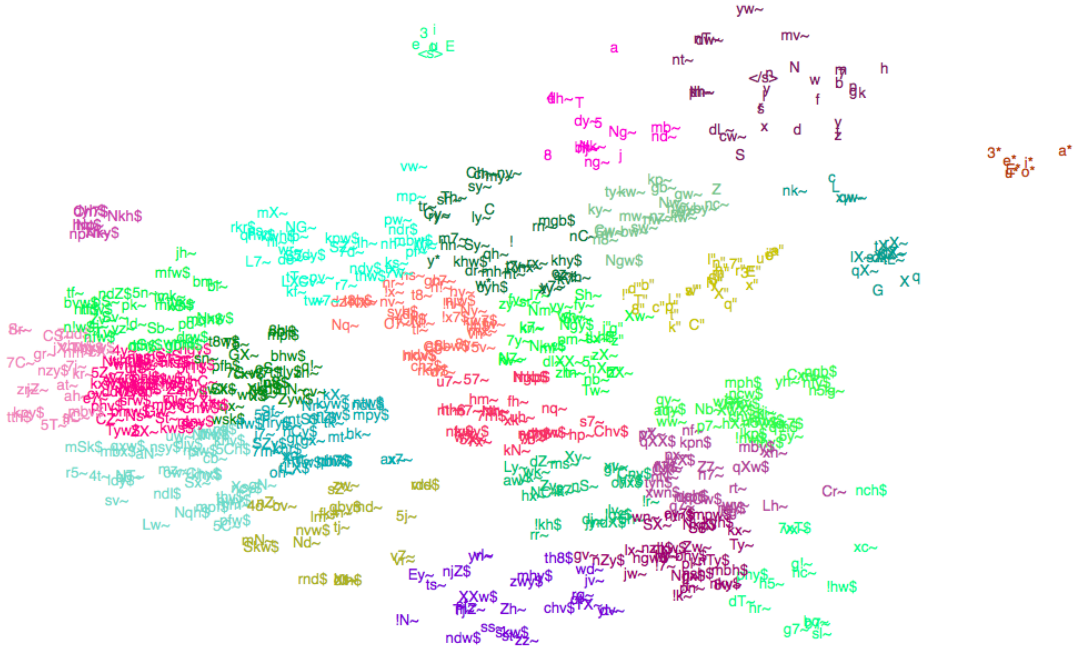

Figure 2: t-SNE visualization of the embeddings of all 720 phonemes contained in ASJP, trained with word2vec. The colors represent clusters inferred by Affinity Propagation. The model can clearly separate vowels and various forms of special articulation types. Pulmonic consonants can also be separated clearly from other coarticulation variants, as can uvular or pre-nasalized sounds. Less frequent phonemes, however, cannot be clearly differentiated.

4.2 Word Embeddings

4.2.1 Autoencoders

4.2.2 Variational Autoencoders

Following this, 4 different models are investigated here:

1. A model that tries to maximize $p(W = w|z; W)$, i.e. that learns the manifold creating the words as such

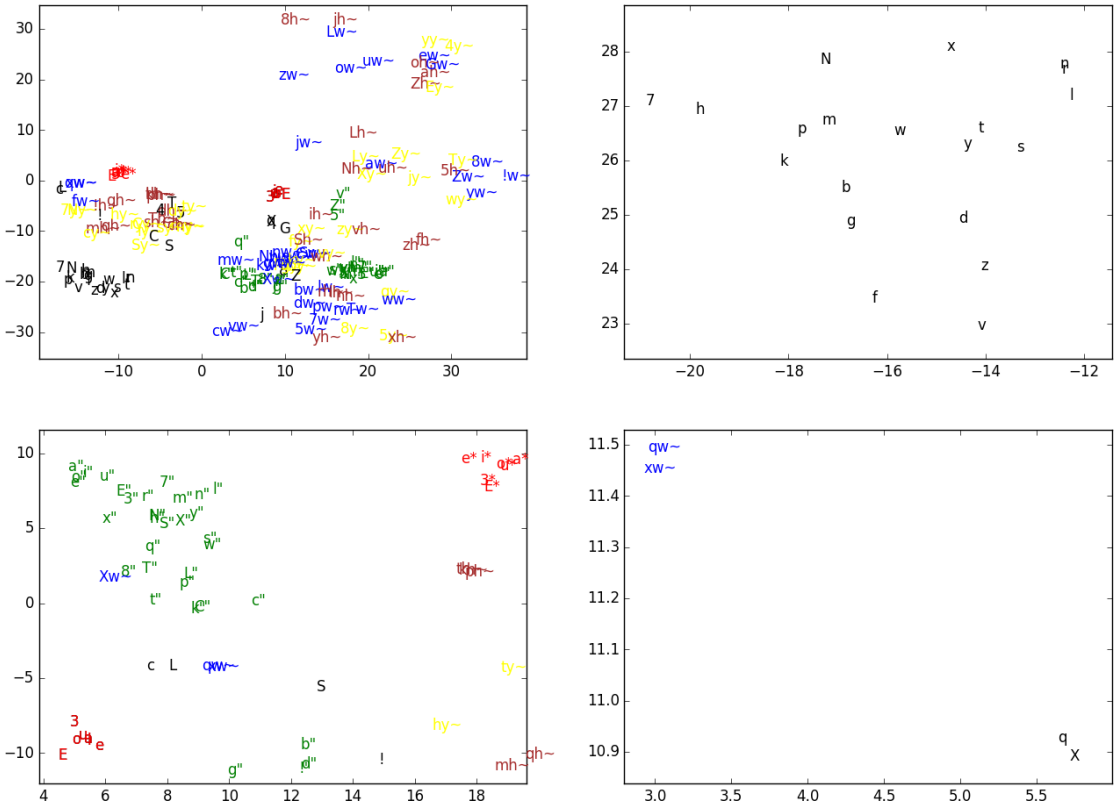

Figure 3: Other t-SNE visualizations of the embeddings created by word2vec. (top left) The model learns to clearly separate natural classes such as vowels, plain pulmonic or glottalized consonants, while other articulations seem to spread over the feature space. The colors indicate membership of a natural phonological class. (top right) A more detailed view on plain pulmonic consonants. Note the linear dependencies between voiced and unvoiced plosives and their respective nasal variant. (bottom left) Another detailed view. Note how the labialized uvular sounds cluster among glottalized consonants. (top right) The model seems to capture different manners of articulations across articulation type boundaries, as the linear dependency shows here.

2. A model that tries to maximize $p(W = w, C = c|z; W, C)$, i.e. that learns the manifold creating the words and the respective concepts

3. A model that tries to maximize $p(W = w, C = c, G = g|z; W, C, G)$, i.e. that learns the manifold creating the words, the respective concepts and the geographical location

4. A model that tries to maximize $p(W = w|z, C = c; W)$, i.e that learns the manifold creating words given the respective concept.

4.3 Clustering

4.3.1 Affinity Propagation

5 Evaluation

5.1 Data

5.2 Results

6 Resume

7 Acknowledgements

For training the phoneme embeddings, I used the word2vec implementations provided by the gensim package [Řehůřek and Sojka2010]. The Autoencoder was implemented with Keras [Chollet2015] and Tensorflow [Abadi et al.2015]. The clustering algorithms used here were provided by scikit-learn [Pedregosa et al.2011]. All code connected to this thesis can be found on my github [1]

References

[Abadi et al.2015] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

[Bouchard-Côté et al.2013] Bouchard-Côté, A., Hall, D., Griffiths, T. L., and Klein, D. (2013). Automated reconstruction of ancient languages using probabilistic models of sound change. *Proceedings of the National Academy of Sciences*, 110(11):4224–4229.

[Bouchard-Côté et al.2007] Bouchard-Côté, A., Liang, P., Griffiths, T. L., and Klein, D. (2007). A probabilistic approach to diachronic phonology. In *EMNLP-CoNLL*, pages 887–896. Citeseer.

[1] https://github.com/marlonbetz/BA

[Bouckaert et al.2012] Bouckaert, R., Lemey, P., Dunn, M., Greenhill, S. J., Alekseyenko, A. V., Drummond, A. J., Gray, R. D., Suchard, M. A., and Atkinson, Q. D. (2012). Mapping the origins and expansion of the indo-european language family. *Science*, 337(6097):957–960.

[Chollet2015] Chollet, F. (2015). Keras. https://github.com/fchollet/keras.

[Dellert2015] Dellert, J. (2015). Uralic and its neighbors as a test case for a lexical flow model of language contact.

[Goldberg and Levy2014] Goldberg, Y. and Levy, O. (2014). word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*.

[Gray et al.2009] Gray, R. D., Drummond, A. J., and Greenhill, S. J. (2009). Language phylogenies reveal expansion pulses and pauses in pacific settlement. *science*, 323(5913):479–483.

[Kondrak2000] Kondrak, G. (2000). A new algorithm for the alignment of phonetic sequences. In *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference*, pages 288–295. Association for Computational Linguistics.

[Landauer et al.2013] Landauer, T. K., McNamara, D. S., Dennis, S., and Kintsch, W. (2013). *Handbook of latent semantic analysis*. Psychology Press.

[List2012] List, J.-M. (2012). Sca: phonetic alignment based on sound classes. In *New Directions in Logic, Language and Computation*, pages 32–51. Springer.

[Mikolov et al.2013a] Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

[Mikolov et al.2013b] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

[Pedregosa et al.2011] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

[Pennington et al.2014] Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–43.

[Rama2016] Rama, T. (2016). Siamese convolutional networks based on phonetic features for cognate identification. *arXiv preprint arXiv:1605.05172*.

[Řehůřek and Sojka2010] Řehůřek, R. and Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA. http://is.muni.cz/publication/884893/en.