Unsupervised Cognate Identification with Variational Autoencoders

Marlon Betz

August 17, 2016

Contents

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 Motivation

3 Sound Change as a Walk in Latent Space

Sound change is usually described as a change of distinctive phonological features. Accordingly, a sound change such as final devoicing as in

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

would be captured by a rule like

$$[+\text{VOICE}] \rightarrow [\text{-VOICE}]/\_\# \tag{2}$$

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

If we use

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

$$v_{w_{ancient}} + v_{sc} = v_{w_{recent}} \tag{3}$$

From this follows that

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

That is, we can formulate sound changes such as 2 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

$$v_{w_{recent}} - v_{w_{ancient}} = v_{w'_{recent}} - v_{w'_{ancient}} \tag{5}$$

which is equivalent to

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

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].

The task is then 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. If we have a word list X, we can use Bayes' theorem to model the probability of our word list given z:

$$P(X|z) = \frac{P(z)P(X|z)}{\int P(z)P(X|z))dz} \tag{7}$$

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 $P(X)$:

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

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. We further need a family of functions $f : \mathcal{Z} \times \Theta \rightarrow \mathcal{X}$ that maps points in $\mathcal{Z}$ to points in our parameter space Θ. The probability of our data given z then becomes

$$P(X|z;\theta) = P_{out}(X|f(z,\theta)) \tag{9}$$

As we expect that the words in our word list follow the central limit theorem and ind 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.

As we usually do not know the ancient version of a word but only recent ones, we expect the cognates to have spread in the feature space from the origin to some directions.

$$v_{w_{recent}} \sim \mathcal{N}(\mu = v_{w_{ancient}}, \sigma^2 I) \tag{11}$$

4 Related Research

5 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.

5.1 Phoneme Vectorization

5.1.1 Hand-crafted Vectorization Models

5.1.2 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

$$v(phoneme_1) + v(phoneme_2) - v(phoneme_3) \approx v(phoneme_4) \tag{12}$$

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

$$v(/\mathbf{i}/) + v(/\mathbf{u*}/) - v(/\mathbf{u}/) \approx v(/\mathbf{i*}/) \tag{13}$$

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

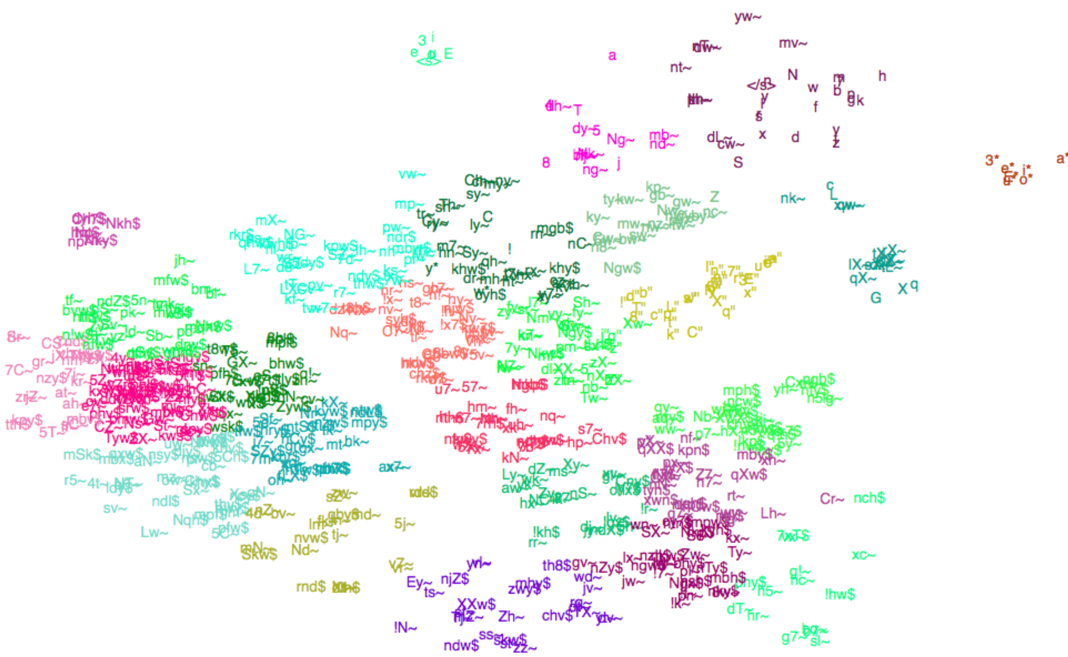

Figure 1: t-SNE visualization of the embeddings of all 720 phonemes contained in ASJP, trained with word2vec. The colors represent clusters inferred by Affinity Propagation. The model can clearly separate vowels and various forms of special articulation types. Pulmonic consonants can also be separated clearly from other coarticulation variants, as can uvular or pre-nasalized sounds. Less frequent phonemes, however, cannot be clearly differentiated.

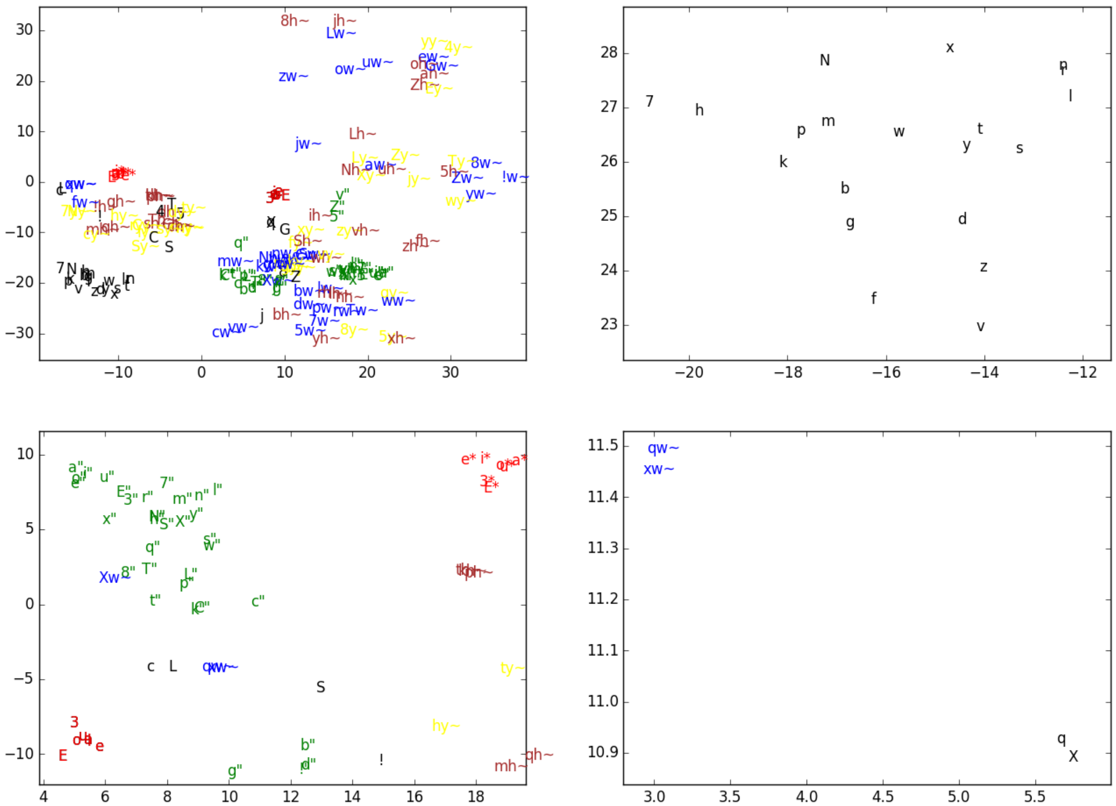

Figure 2: Other t-SNE visualizations of the embeddings created by word2vec. (top left) The model learns to clearly separate natural classes such as vowels, plain pulmonic or glottalized consonants, while other articulations seem to spread over the feature space. The colors indicate membership of a natural 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.

5.2 Word Embeddings

5.2.1 Autoencoders

5.2.2 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
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.

5.3 Clustering

5.3.1 Affinity Propagation

6 Evaluation

6.1 Data

6.2 Results

7 Resume

8 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.

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

[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.

[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.

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

[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.