Bayesian Phylogenetic Cognate Prediction

Gerhard Jäger
University of Tübingen
Seminar für Sprachwissenschaft
gerhard.jaeger@uni-tuebingen.de

Abstract

In Jäger (2019) a computational framework was defined to start from parallel word lists of related languages and infer the corresponding vocabulary of the shared proto-language. The SIGTYP 2022 Shared Task is closely related. The main difference is that what is to be reconstructed is not the proto-form but an unknown word from an extant language. The system described here is a re-implementation of the tools used in the mentioned paper, adapted to the current task.

1 Introduction

In Jäger (2019) I presented a pilot study of a computational historical linguistics workflow. Starting from parallel word lists (taken from Wichmann et al. 2016) of 29 Romance languages and dialects, covering 40 core concepts, it produced reconstructions of the Proto-Romance words for the same concepts.

The intermediate steps of this workflow are

1. for each concept, cluster the corresponding sound strings into *cognate classes*,
2. infer a posterior distribution of phylogenies of the covered doculects using Bayesian inference,
3. apply Bayesian inference to identify the *maximum a posteriori* cognate class at the root of the tree for each concept (*ancestral state reconstruction*, ASR),
4. apply multiple sequence alignment (MSA) to the words of each cognate class,
5. apply ASR to each alignment column of the MSAs of the cognate classes identified in step 3; gaps are treated as regular characters, and
6. concatenate the reconstructions and removing gaps.

The result turned out to be an imperfect but reasonable approximations of the attested Latin wordlist.

The SIGTYP 2022 Shared Task on the Prediction of Cognate Reflexes (https://github.com/sigtyp/ST2022, List et al. 2022) is very similar in nature. The system described here is an adaptation of Jäger's (2019) workflow to this task.

2 Data

The authors of the Shared Task made parallel word lists for 20 language families available. For details of the provinence of the data and the pre-processing steps performed, see List et al. (2022). Each dataset comprises between four and 19 related languages, and between 500 and ca. 10,000 words. Words are classified according to cognate classes, which are based either on expert judgments or are inferred via automatic cognate detection. No information about the meanings of the words are available for training or inference. All words are transcribed in IPA and tokenized.

The data are arranged in a table with cognate classes as rows and languages as columns. In Table 1, a small part from the dataset `kesslersignificance` (based on Kessler 2001) is shown for illustration.

Each dataset was split into a training set and a test set. The proportion of test data was varied between 10%, 20%, 30%, 40% and 50%, leading to a total of 50 datasets, each consisting of a training and a test set. For the test data, one word per row was masked, using each attested word for masking in turn. The task is to predict the masked words from the other cognates in the same row.

Table 2 contains an example row from such a test set. The task is to infer the French word which is cognate to Albanian *piski*, English *fɪʃ*, German *fiʃ* and Latin *piski*. In a separate file which is only to be used for evaluation, the correct solution — *pʃ* in this case — is given.

COGID	Albanian	English	French	German	Latin
920		h ɑ r t	k œ r	h e r ts ə n	k o r d
1083		h ɔ r n	k ɔ r n	h o r n	k o r n u:
1150	ʃ k u r t ə r	ʃ ɔ r t	k u r t	k u r ts	

Table 1: Example training data

COGID	Albanian	English	French	German	Latin
353-3	p e ʃ k	f ɪ ʃ	?	f i ʃ	p i s k i

Table 2: Example test data

For each of the 50 datasets, a system can be trained using the complete training set. For prediction, the trained system only "sees" one row of the test data and has to predict the masked word.

3 Methods

This task differs from the one described in (Jäger, 2019) mainly by the fact that not some ancestral word form has to be inferred but a word from an extant language. For the particular inference methods used, this difference is actually inessential, since it is based on a *time-reversible* model of language change.

The first step of the workflow by Jäger (2019), identifying cognate classes, has already been performed here. This led to the following workflow:

1. Train a pair-hidden Markov model (pHMM; see Durbin et al. 1989) for pairwise string alignment.[1]

2. Infer a preliminary phylogenetic tree via UPGMA (Sokal and Michener, 1958).

3. Perform MSA per cognate class using the T-Coffee algorithm (Notredame et al., 2000).

4. Join all MSA matrices and use this as character matrix for Bayesian phylogenetic inference.[2]

5. Infer the posterior distribution of the mutation rate of symbols within the columns of the MSAs.

6. Apply MSA to the non-masked entries in the test row using the model trained in steps 1 and 2.

7. Find the *maximum a-posteriori* state for each MSA column for the masked entry, using the posterior distributions inferred in steps 4 and 5 as priors. Concatenate the states inferred in the previous step and remove gap symbols.

Each of these steps will be briefly explained in the following subsections.

3.1 Training a Pair-Hidden Markov Model

A *pair-Hidden Markov Model* (pHMM) is a Hidden Markov model with two parallel output tapes. In each state, the model may emit a symbol on the first, the second or on both tapes. The architecture used here is taken from Durbin et al. (1989) and schematically displayed in Figure 1.

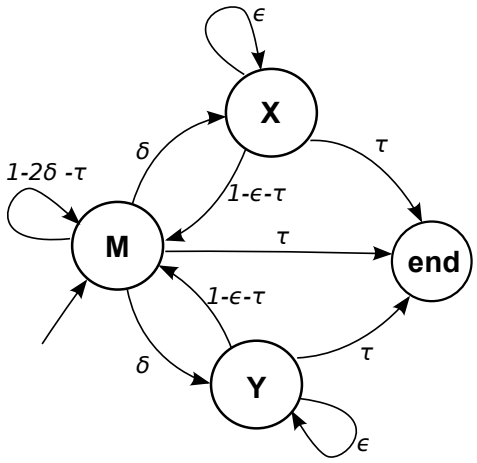

Figure 1: Pair Hidden Markov Model

The state **M** is the *match* state, where the model simultaneously emits one symbol on each tape. In

[1] In Jäger (2019), pairwise string alignment was performed using the Needleman-Wunsch algorithm (Needleman and Wunsch, 1970) with parameters trained on the entire ASJP database (Wichmann et al., 2016). Since the rules of the Shared Task precludes the use or external data for parameter training, I opted for a method here were parameters can be estimated from scratch using only the licit training data.

[2] In Jäger (2019) phylogenetic inference was performed using cognate data, but since the Shared Task does not make information about the meaning of the words available, this was not possible here.

state **X** only a symbol to the first tape is emitted, and likewise for state **Y** and the second tape. When the model reaches the **end** state (where no symbol is emitted), each tape contains a symbol sequence. The joint probability of this sequence and the simultaneous sequence of hidden states is determined by the product of the transition and emission probabilities used.

Crucially, the sequence of hidden states of one pass of the model determines a pairwise alignment of the strings produced. **M** identifies a match column, **X** a column with a gap in the second string, and **Y** a gap in the first string. If the parameters of the model are known, the maximum likelihood alignment between two strings can be found using the *Viterbi algorithm*.[3]

It is assumed that the alphabet from which the words are constructed are known in advance. The parameters of the model are the transition probabilities δ, ϵ and τ, and the emission probabilities for each state. For state **M**, this is a probability distribution over pairs of symbols from the alphabet. I assume the emission probabilites for states **X** and **Y** to be identical; both are a probability distribution over the alphabet.

Given a training set of pairs of strings, parameters of the model can be estimated using the *Baum-Welch algorithm*, an incarnation of the EM algorithm. If values for all parameters of the model are given, the frequency of all transitions and all emissions for a given set of string pairs are estimated (expectation step). The conditional relative frequencies for each transition and emission are then used as new parameter values (maximization step). This procedure is repeated many times, starting from an arbitrary initial state.

In the system described here, the pHMM was initialized with transition probabilities $\delta = \tau = 0.25, \epsilon = 0.375$. The initial emission probabilities at the gap states **X** and **Y** are uniform distributions. The emission probabilities in the match state **M** are

$$p(a, b) \propto 1 \qquad \text{if } a \neq b$$
$$p(a, a) \propto |\text{alphabet}| + 1$$

These choices are motivated by the idea that Viterbi alignment in the initial state should approximate Levenshtein alignment.

[3]This inference step amounts to a notational variant of the Needleman-Wunsch algorithm, cf. Needleman and Wunsch (1970).

For training and MSA, all training strings (and later test strings) were converted into the ASJP alphabet (Brown et al., 2008), which comprises just 41 sound classes, to keep the number of parameters to be estimated manageable.[4] The conversion was performed using the software package *LingPy* (List and Forkel, 2021). Training word pairs, i.e., all pairs of cognate words from the training set, were arranged in random order and split into mini-batches of size 20. An EM step was performed for each mini-batch. This procedure was repeated for two epochs over all mini-batches.

3.2 UPGMA Tree

As preparation for multiple sequence alignment, a guide tree over the languages is required. For this purpose, the pairwise normalized Levenshtein distance (i.e., the edit distance divided by the length of the longer string) was computed between any pair of cognate words. The distance between two languages was then computed as the average word distance between any two cognate words from these languages.

The resulting pairwise language distances were used as input for the UPGMA algorithm to infer a language tree. E.g., for the dataset *kesslersignificance* with 10% test data, the resulting tree has the topology ((Latin, (French, Albanian)), (English, German)).

This topology is evidently not perfect (Albanian having the wrong location), but the next step, while requiring a guide tree, is not very sensitive to the specific tree topology.

3.3 Multiple Sequence Alignment

The alignment method described in Subsection 3.1 above is only capable of performing pairwise sequence alignment. Modifying it to multiple strings would require to increase the number of states, and concommittantly computation time, exponentially in the number of sequences. The T-Coffee method of multiple sequence alignment (Notredame et al., 2000) represents a compromise combining good results with computational efficiency.

To compute an MSA for a group of words, first all pairs of words are aligned pairwise. For this step, I used Viterbi alignment with the pHMM parameters described in Subsection 3.1. During the next step of T-Coffee, all threefold alignments are

[4]Here and elsewhere, symbols indicating morpheme boundaries were ignored.

computed simply by combining two pairwise alignments from the previous step. The alignment scores between any pair of symbol *tokens* are obtained by counting all threefold alignments where these symbols occur in the first and last column, weighted by the Hamming similarity between the entire first and last row.

Using these scores, *progressive alignment* (Feng and Doolittle, 1987) is performed using a guide tree.

To continue the example mentioned above, the MSA covering the first row of Table 1 comes out as in Table 3.

Albanian	- - - - - -
English	h o r t - -
French	k E r - - -
German	h e r C I n
Latin	k o r d - -

Table 3: Example MSA

3.4 Bayesian Phylogenetic Inference

The MSAs for the training data thus obtained were used to perform more sophisticated, Bayesian phylogenetic inference. For this purpose each symbol in the MSA is replaced by the corresponding Dolgopolsky class (Dolgopolsky, 1986). This conversion was performed using LingPy (List and Forkel, 2021) as well.

For each alignment column, the symbols in this column are conveived of as states of a continuous time Markov process. The specific type of Markov process used is due to Jukes and Cantor (1969).

Let a phylogeny — i.e., a tree with branch lengths — over the languages in question be given. It is assumed that the types of symbols within an alignment colum are the states of a continuous time Markov process. A complete model is one where each node is assigned exactly one state. For the leaf nodes, these are the entries of the MSA column. Let u and l be the states at the top and at the bottom of a branch of the phylogenetic tree, and let t be the length of the branch.

The likelihood of this branch is

$$P(l|u) = \begin{cases} \frac{1}{n} + \frac{n-1}{n} e^{-rt} & \text{if } u = l \\ \frac{1}{n} - \frac{1}{n} e^{-rt} & \text{else,} \end{cases}$$

where n is the number of distinct symbols occurring in the MSA column. The rate r is a model paramter and is always positive.

The total likelihood of an assignment of states to the nodes of the tree is the product of all branch likelihood, times the likelihood of the state at the root. For this I assumed a uniform distribution.

The marginal likelhood of the states at the leaves, given a phylogeny $\mathcal{T}$ and rate r is the sum of the likelihoods of all assignments of states to non-leaf nodes. The likelihood of a complete character matrix, given a phylogeny and an assignment of a rate value for each character (i.e., MSA column), is the product of the likelihoods of the individual characters. When a character state for a language is unknown — either because it is a gap in the MSA, or the language does not have a reflex for the corresponding cognate class — the marginal likelihood is computed as the sum of the likelihoods for all possible character states.

Given suitable priors for the phylogeny and the rates, the posterior distribution over trees can be estimated via Bayesian inference for the collection of MSAs as data.

This step was carried out using the software *MrBayes* (Ronquist and Huelsenbeck, 2003). Rates were allowed to vary between characters, but are drawn from a discretized Gamma distribution with equal mean and variance. The mean of this hyperprior distribution is drawn from a standard exponential distribution. A uniform prior distribution over tree topologies was assumed, paired with a standard exponential prior distribution over the tree age and a uniform prior distribution over the branch lengths.

The posterior tree distribution for the running example is visualized in Figure 2 (produced with the software *densitree*, Bouckaert and Heled 2014). It can be seen that there is considerable uncertainty regarding the position of French and Albanian in the tree, as well as regarding the height of the tree.

3.5 Inferring Mutation Rates

While I used Dolgopolsky sound classes for phylogenetic inference, cognate inference has to operate on IPA characters. For this purpose, I used the posterior tree distribution from the previous step as prior distribution. Data are MSAs of IPA strings. For the running example, this looks as in Table 4. (Note that the MSA is computed on the basis of ASJP strings, and ASJP symbols are replaced by the corresponding IPA symbols afterwards.)

As a further deviation from the previous step, gaps (indicated by "-") are treated as normal char-

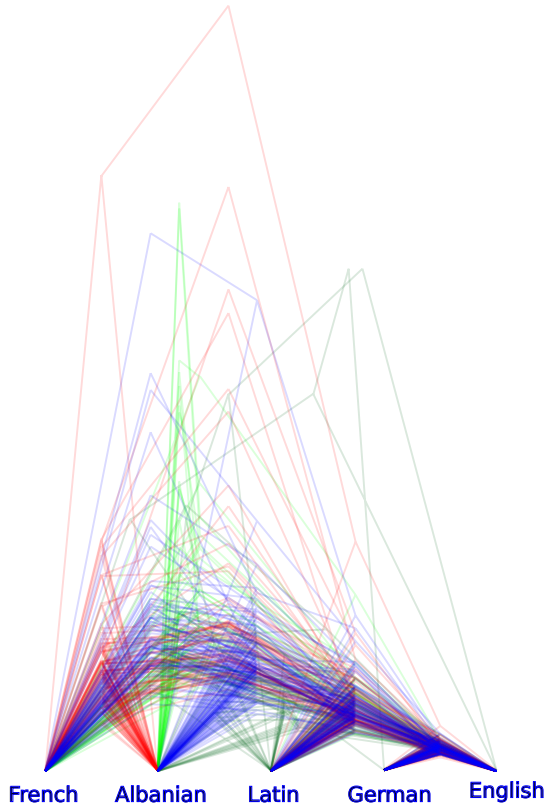

Figure 2: Posterior tree distribution

Albanian	.	.	.	.	.	.
English	h	ɑ	r	t	–	–
French	k	œ	r	–	–	–
German	h	e	r	ts	ə	n
Latin	k	o	r	d	–	–

Table 4: Example MSA with IPA characters

acter states, while missing data (indicated by ".") are marginalized out.

For this step I assumed a constant rate over all characters. Inference was performed using the *Julia* package *MCPhylo.jl* (Wahle 2021; https://juliapackages.com/p/mcphylo), leading to a sample from the posterior distribution over rates.

3.6 Multiple Sequence Alignment of Test Data

For cognate prediction, the attested entries of the cognate class in question are aligned using the procedure and the model described in Subsection 3.3. If the test data contain symbols not occurring in the training data, their emission probabilities are set to the minimal emission probability of any symbol from the training data, and emission probabilities are re-normalized in the trained pHMM.

For the running example, the MSA is shown in Table 5. The entries for French (shown in boldface) are unknown and have to be inferred in the final step.

Albanian	p	e	ʃ	k	–
English	f	ɪ	ʃ	–	–
French	**p**	**i**	**ʃ**	**k**	**–**
German	f	i	ʃ	–	–
Latin	p	i	s	k	i

Table 5: MSA for cognate prediction

3.7 Cognate Prediction

Missing-value imputation is done column-wise. Using the posterior distribution over trees and rates described in Subsections 3.4 and 3.5, for each slot the posterior probability distribution over the symbols occurring elsewhere in the column was computed. This was practically implemented by separately computing the posterior probabilities for all candidate symbols separately and normalizing them.

As prediction, the symbol with the highest posterior probability was chosen. The final cognate prediction is the result of removing all gap symbols — *piʃk* in the example.

4 Discussion

Let me close with a brief reflection on what kind of information this system extracts from the training set to perform cognate prediction. There are mainly two patterns the system pays attention to. The first is the regularity of sound correspondences which are encapsulated in the emission probabilites of the trained pHMM, especially its **M** state. The system does not pay attention to the specific languages the words to be aligned come from, so it is unaware of language-specific sound correspondences. Therefore the prediction step does not make use of specific sound laws in any way.

Second, the system employs phylogenetic information. This amounts to a weighing of the importance of the cognates from other languages when deciding on the choice of the missing value imputation.

Also, since the missing value imputation is performed column-wise for the alignment matrix, no syntagmatic information is being used. It is not checked which candidate predictions are phonotactically or morphologically most similar to the training words from the same languages.

In future research, it is worth considering to extend the system towards the usage of language-specific sound correspondences and syntagmatic information.

Supplementary Material

The source code and instructions how to run the system are publicly available at https://github.com/gerhardJaeger/gerhardSigtyp2022 (also archived on Zenodo under the doi 10.5281/zenodo.6559085). Most of the workflow was implemented in the *Julia* language (https://julialang.org/), a relatively new language combining the convenient syntax and interactive functionality of languages such as *Python* with execution speed of optimized code close to *C* or *Java*. Essential *Julia* packages used are Johannes Wahle's *MCPhylo.jl* (which is based on Brian J. Smith' *Mamba.jl* package; https://mambajl.readthedocs.io/en/latest/) for phylogenetic Bayesian inference and my own package *SequenceAlignment.jl* (https://github.com/gerhardJaeger/SequenceAlignment.jl, v0.9.1) for sequence alignment.

For conversions between different sound class systems, the *Python* package *LingPy* (List and Forkel, 2021) was used. Besides *MCPhylo.jl*, I used *MrBayes* (Ronquist and Huelsenbeck, 2003) for phylogenetic inference. Postprocessing of the output of *MrBayes* was done with the *R* package *ape* (Paradis et al., 2004).

Acknowledgements

I am grateful to Johannes Wahle for technical support during the implementation.

This research was supported by the DFG Centre for Advanced Studies in the Humanities Words, Bones, Genes, Tools (DFG-KFG 2237) and by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement 834050).

References

Remco R. Bouckaert and Joseph Heled. 2014. Densitree 2: Seeing trees through the forest. *BioRxiv*. doi.org/10.1101/012401.

Cecil H. Brown, Eric W. Holman, Søren Wichmann, and Viveka Velupillai. 2008. Automated classification of the world's languages: A description of the method and preliminary results. *STUF — Language Typology and Universals*, 4:285–308.

Aaron B. Dolgopolsky. 1986. A probabilistic hypothesis concerning the oldest relationships among the language families of Northern Eurasia. In V. V. Shevoroshkin, editor, *Typology, Relationship and Time: A collection of papers on language change and relationship by Soviet linguists*, pages 27–50. Karoma Publisher, Ann Arbor.

Richard Durbin, Sean R. Eddy, Anders Krogh, and Graeme Mitchison. 1989. *Biological Sequence Analysis*. Cambridge University Press, Cambridge, UK.

Da-Fei Feng and Russell F. Doolittle. 1987. Progressive sequence alignment as a prerequisitetto correct phylogenetic trees. *Journal of Molecular Evolution*, 25(4):351–360.

Gerhard Jäger. 2019. Computational historical linguistics. *Theoretical Linguistics*, 45(3-4):151–182.

Thomas H. Jukes and Charles R. Cantor. 1969. Evolution of protein molecules. In H. N. Munro, editor, *Mammalian protein metabolisum*, pages 21–132. Academic Press, New York and London.

Brett Kessler. 2001. *The significance of word lists*. CSLI Publications, Stanford.

Johann-Mattis List and Robert Forkel. 2021. Lingpy. A Python library for historical linguistics. version 2.6.9. URL: https://lingpy.org, DOI: https://zenodo.org/badge/latestdoi/5137/lingpy/lingpy. With contributions by Greenhill, Simon, Tresoldi, Tiago, Christoph Rzymski, Gereon Kaiping, Steven Moran, Peter Bouda, Johannes Dellert, Taraka Rama, Frank Nagel. Leizpig: Max Planck Institute for Evolutionary Anthropology.

Johann-Mattis List, Ekaterina Vylomova, Robert Forkel, Nathan W. Hill, and Ryan Cotterell. 2022. The SIGTYP 2022 shared task on the prediction of cognate reflexes. In *The Fourth Workshop on Computational Typology and Multilingual NLP*, Online. Association for Computational Linguistics.

Saul B. Needleman and Christian D. Wunsch. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48:443–453.

Cédric Notredame, Desmond G Higgins, and Jaap Heringa. 2000. T-Coffee: A novel method for fast and accurate multiple sequence alignment. *Journal of Molecular Biology*, 302(1):205–217.

Emmanuel Paradis, Julien Claude, and Korbinian Strimmer. 2004. APE: analyses of phylogenetics and evolution in R language. *Bioinformatics*, 20(2):289–290.

Frederik Ronquist and John P. Huelsenbeck. 2003. MrBayes 3: Bayesian phylogenetic inference under mixed models. *Bioinformatics*, 19(12):1572–1574.

Robert R. Sokal and Charles D. Michener. 1958. A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin*, 38:1409–1438.

Johannes Wahle. 2021. No-U-Turn sampling for phylogenetic trees. *bioRxiv*. doi.org/10.1101/2021.03.16.435623.

Søren Wichmann, Eric W. Holman, and Cecil H. Brown. 2016. The ASJP database (version 17). http://asjp.clld.org/.