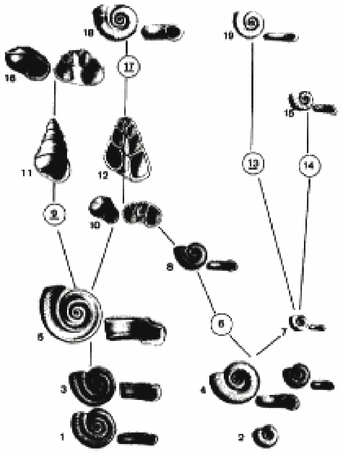
THE GENEALOGICAL WORLD OF PHYLOGENETIC NETWORKS



Automatic sound law induction (Open problems in computational diversity linguistics 3)

List, Johann-Mattis

April 2019

Cite as: List, Johann-Mattis (2019): Automatic sound law induction (Open problems in computational diversity linguistics 3). The Genealogical World of Phylogenetic Networks 6.1.

http://phylonetworks.blogspot.com/2019/04/automatic-sound-law-induction-open.html

Automatic sound law induction (Open problems in computational diversity linguistics 3)

The third problem in my list of ten open problems in computational diversity linguistics is a problem that has (to my knowledge) not even been considered as a true problem in computational historical linguistics, so far. Until now, it has been discussed by colleagues only indirectly. This problem, which I call the *automatic induction of sound laws*, can be described as follows:

> Starting from a list of words in a proto-language and their reflexes in a descendant language, try to find the rules by which the ancestral language is converted into the descendant language.

Note that by "rules", in this context, I mean the classical notation that phonologists and historical linguists use in order to convert a source sound in a target sound in a specific environment (see Hall 2000: 73-75). If we consider the following ancestral and descendant words from a fictive language, we can easily find the laws by which the input should be converted into an output — namely, an *a* should be changed to an *e*, an *e* should be changed to an *i*, and a *k* changes to *s* if followed by an *i* but not if followed by an *a*.

Input	***Output***
papa	pepe
mama	meme
kaka	keke
keke	sisi

Short excursus on linguistic notation of sound laws

Based on the general idea of sound change (or sound *laws* in classical historical linguistics) as some kind of a function by which a *source sound* is taken as input and turned into a *target sound* as output, linguists use a specific notation system for sound laws. In the simplest form of the classical sound law notation, this process is described in the form `s > t`, where *s* is the source sound and *t* is the target sound. Since sound change often relies the on specific conditions of the surrounding context — i.e. it makes a difference if some sound occurs in the beginning or the end of a word — context is added as a condition separated by a `/`, with an underscore _ referring to the sound in its original phonetic environment. Thus, the phenomenon of voiced stops becoming unvoiced at the end of words in German (e.g. *d* becoming *t*), can be written as `d > t / _$`, where *$* denotes the end of a word.

One can see how close this notation comes to regular expressions and according to many scholars, the rules by which languages change with respect to their sound systems do not exceed the complexity of regular grammars. Nevertheless, sound change notation does differ in the scope and the rules for annotation. One notable difference is the possibility to explain how full *classes of sounds* change in a specific environment. The German rule of devoicing, for example, generally affects *all* voiced stops in the end of a word. As a result, one could also annotat it as `G > K / _$`, where *G* would denote the sounds `[b, d,` g] and *K* their counterparts [p, t, k]. Although we could easily write a single rule for each of the three phenomena here, the rule by which the sounds are grouped into two classes of voiced sounds and their unvoiced counterparts is linguistically more interesting, since it reminds us that the change by which word-final consonants loose the feature of voice is a *systemic change*, and not a phenomenon applying to some random selection of sounds in a given language.

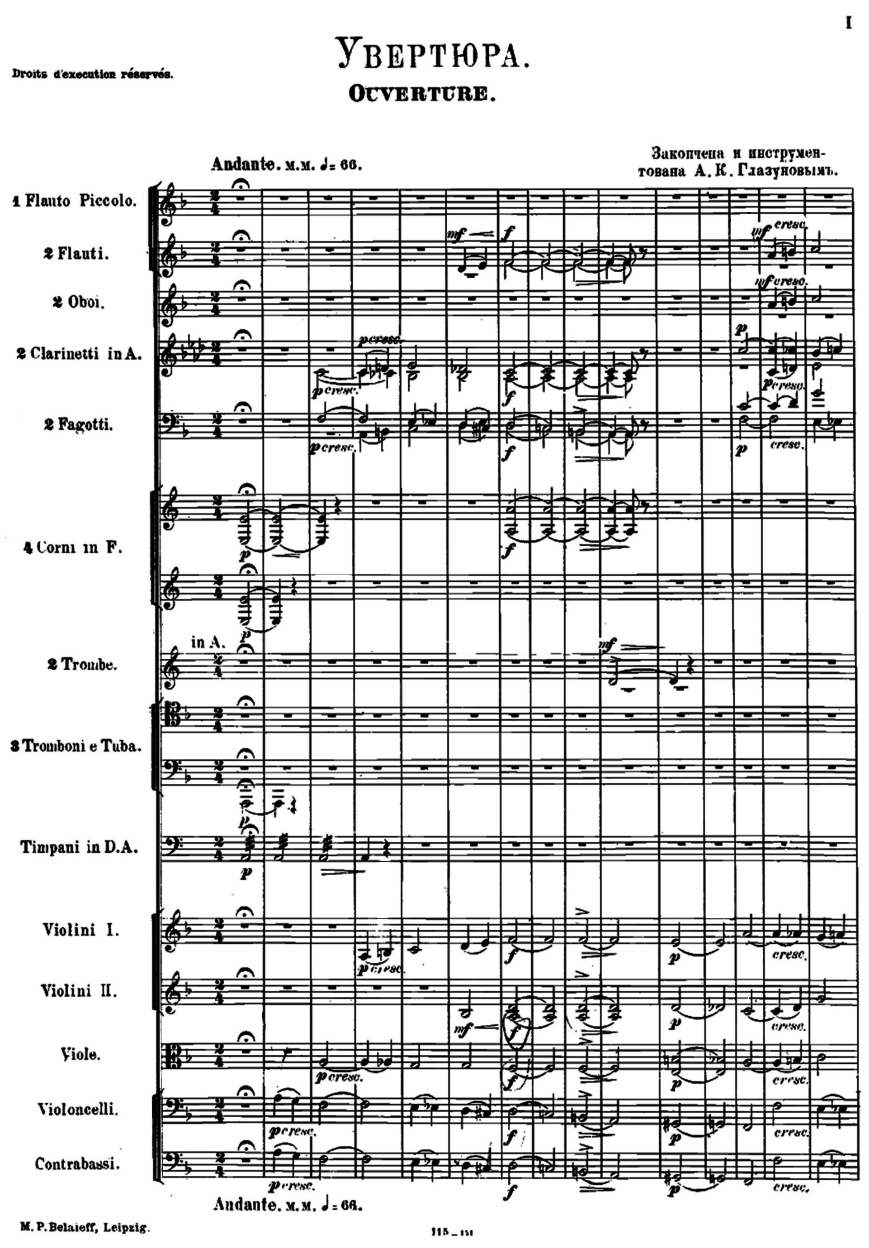
The problem of this *systemic annotation*, however, is that the grouping of sounds into classes that change in a similar form is often language-specific. As a result, scholars have to propose new groupings whenever they deal with another language. Since neither the notation of sound values nor the symbols used to group sounds into classes are standardized, it is extremely difficult to compare different proposals made in the literature. As a result, any attempt to solve the problem of automatic sound law induction in historical linguistics would at the same time have to make strict proposals for a standardization of sound law notations used in our field. Standardization can thus be seen as one of the first major obstacles of solving this problem, with the problem of accounting for systemic aspects of sound change as the second one.

Beyond regular expressions

Even if we put the problem of inconsistent annotation and systemic changes to one side, the analogy with regular expressions cannot properly handle all aspects of sound change. When looking at the change from Middle Chinese to Mandarin Chinese, for example, we find a complex pattern, by which originally voiced sounds, like [b, d, g, dz] (among others), were either devoiced, becoming [p, t, k, ts], or devoiced *and* aspirated, becoming [pʰ, tʰ, kʰ, tsʰ]. While it is not uncommon that one sound can change into two variants, depending on the context in which it occurs, the Mandarin sound change in this case is interesting because the context is not a neighboring sound, but is instead the Middle Chinese tone for the syllable in question — syllables with a flat tone (called *píng* tone in classical terminology) are nowadays voiceless and aspirated, and syllables with one of the three remaining Middle Chinese tones (called *shǎng*, *qù*, and *rù*) are nowadays plain voiceless (see List 2019: 157 for examples).

Since tone is a feature that applies to whole syllables, and not to single sound segments, we are dealing with so-called *supra-segmental* features here. As the meaning of the term *supra-segmental* indicates, the features in question cannot be represented as a sequence of sound, but need to be thought of as an additional *layer*, similar to other supra-segmental features in language, including *stress*, or *juncture* (indicating word or morpheme boundaries).

In contrast to sequences as we meet them in mathematics and informatics, linguistic sound sequences do not consist solely of letters drawn from an alphabet that is lined up in some unique order. They are instead often composed of multiple layers, which are in part hierarchically ordered. Words, morphemes, and phrases in linguistics are thus multi-layered constructs, which cannot be represented by one sequence alone, but could be more fruitfully thought of as the same as a *partitura* in music — the score of a piece of orchestra music, in which every voice of the orchestra is given its own sequence of sounds, and all different sequences are aligned with each other to form a whole.



The multi-layered character of sound sequences can be seen as similar to a partitura in musical notation.

This multi-layered character of sound sequences in spoken languages comprises a third complication for the task of automatic sound law induction. Finding the individual laws that trigger the change of one stage of a language to a later stage, cannot (always) be trivially reduced to the task of finding the finite state transducer that translates a set of input strings to a corresponding set of output strings. Since our input word forms in the proto-language are not simple strings, but rather an alignment of the different layers of a word form, a method to induce sound laws needs to be able to handle the multi-layered character of linguistic sequences.

Background for computational approaches to sound law induction

To my knowledge, the question of how to induce sound laws from data on proto- and descendant languages has barely been addressed. What comes closest to the problem are attempts to *model* sound change from known ancestral languages, such as Latin, to daughter languages, such as Spanish. This is reflected, for example, in the PHONO program (Hartmann 2003), where one can insert data for a proto-language along with a set of sound change rules (provided in a similar form to that mentioned above), which need to be given in a specific order, and are then checked to see whether they correctly predict the descendant forms.

For teaching purposes, I adapted a JavaScript version of a similar system, called the *Sound Change Applier²* (http://www.zompist.com/sca2.html) by Mark Rosenfelder from 2012, in which students could try to turn Old High German into modern German, by assigning simple rules as they are traditionally used to describe sound change processes in the linguistic literature. This adaptation (which can be found at http://dighl.github.io/sound_change/SoundChanger.html) compares the attested output with the output generated by a given set of rules, and provides some assessment of the general accuracy of the proposed set of rules. For example, when feeding the system the simple rule `an > en /_#`, which turns all final instances of *-an* into *-en*, 54 out of 517 Old High German words will yield the expected output in modern Standard German.

The problem with these endeavors is, of course, the handling of exceptions, along with the comparison of different proposals. Since we can think of an infinite number of rules by which we could successfully turn a certain amount of Old High German strings into Standard German strings, we would need to ask ourselves how we could evaluate different proposals. That some kind of parsimony should play a role here is obvious. However, it is by no means clear (at least to me) how to evaluate the complexity of two systems, since the complexity would not only be reflected in the number of rules, but also in the initial grouping of sounds to classes, which is commonly used to account for systemic aspects of sound change. A system accounting for the problem of sound law induction would try to automate the task of finding the set of rules. The fact that it is difficult even to compare two or more proposals based on human assessment further illustrates why I think that the problem is not trivial.

Another class of approaches is that of word prediction experiments, such as the one by Ciobanu and Dinu (2018) (but see also Bodt and List 2019), in which training data consisting of the source and the target language are used to create a model, which is then successively applied to new data, in order to test how well this model predicts target words from the source words. Since the model itself is not reported in these experiments, but only used in the form of a black box to predict new words, the task cannot be considered to be the same as the task for sound law induction — which I propose as one of my ten challenges for computational historical linguistics — given that we are interested in a method that explicitly returns the model, in order to allow linguists to inspect it.

Problems with the current solutions to sound law induction

Given that no real solutions exist to the problem up to now, it seems somewhat useless to point to the problems of current solutions. What I want to mention in this context, however, are the problems of the solutions presented for word prediction experiments, be they fed by manual data on sound changes (Hartmann 2003), or based on inference procedures (Ciobanu and Dinu 2018, Dekker 2018). Manual solutions like PHONO suffer from the fact that they are tedious to apply, given that linguists have to present all sound changes in their data in an ordered fashion, with the program converting them step by step, always turning the whole input sequence into an intermediate output sequence — the word prediction approaches thus suffer from limitations in feature design.

The method by Ciobanu and Dinu (2018), for example, is based on orthographic data alone, using the Needleman-Wunsch algorithm for sequence alignment (Needleman and Wunsch 1970); and the approach by Dekker (2018) only allows for the use for the limited alphabet of 40 symbols proposed by the ASJP project (Holman et al. 2008). In addition to the limited representation of linguistic sound sequences, be it by resorting to abstract orthography or to abstract reduced phonetic alphabets, none of the methods can handle those kinds of contexts which result from the multi-layered character of speech. Since we know well that these aspects are vital for certain phenomena of sound change, the methods exclude from the beginning an aspect that traditional historical linguists, who might be interested in an automatic solution to the sound law induction problem, would put at the top of their wish-list of what the algorithm should be able to handle.

Why is automatic sound law induction difficult?

The handling of supra-segmental contexts, mentioned above, is in my opinion also the reason why sound law induction is so difficult, not only for machines, but also for humans. I have so far mentioned three major problems as to why I think sound law induction is difficult. First, we face problems in defining the task properly in historical linguistics, due to a significant lack in standardization. This makes it difficult to decide on the exact output of a method for sound law induction. Second, we have problems in handling the systemic aspect of sound change properly. This does not apply only to automatic approaches, but also to the evaluation of different proposals for the same data proposed by humans. Third, the multi-layered character of speech requires an enhanced modeling of linguistic sequences, which cannot be modeled as mono-dimensional strings alone, but should rather be seen as alignments of different strings representing different layers (tonal layer, stress layer, sound layer, etc.).

How humans detect sound laws

There are only a few examples in the literature where scholars have tried to provide detailed lists of sound changes from proto- to descendant language (Baxter 1992, Newman 1999). Most examples of individual sound laws proposed in the literature are rarely even tested exhaustively on the data. As a result, it is difficult to assess what humans usually do in order to detect sound laws. What is clear is that historical linguists who have been working a lot on linguistic reconstruction tend to acquire a very good intuition that helps them to quickly check sound laws applied to word forms in their head, and to convert the output forms. This ability is developed in a learning-by-doing fashion, with no specific techniques ever being discussed in the classroom, which reflects the general tendency in historical linguistics to trust that students will learn how to become a good linguist from examples, sooner or later (Schwink 1994: 29). For this reason, it is difficult to take inspiration from current practice in historical linguistics, in order to develop computer-assisted approaches to solve this task.

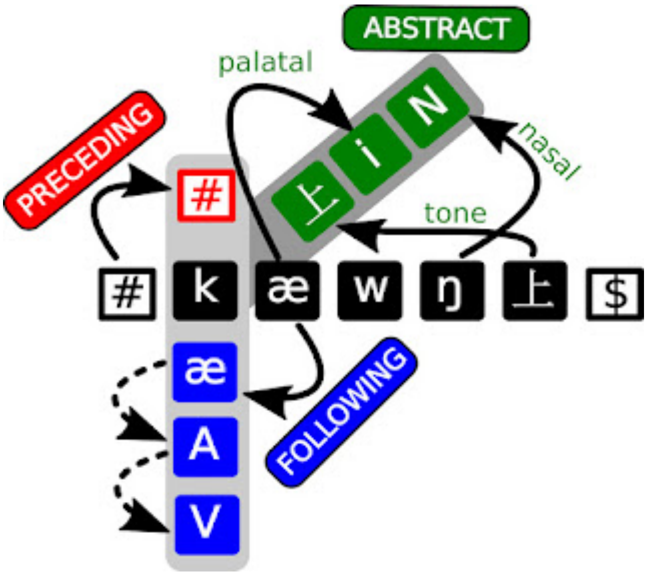
Potential solutions to the problem

What can we do in order to address the problem of sound law induction in automatic frameworks in the future?

As a first step, we would have to standardize the notation system that we use to represent sound changes. This would need to come along with a standardized phonetic transcription system. Scholars often think that phonetic transcription *is* standardized in linguistics, specifically due to the use of the International Phonetic Alphabet. As our investigations into the actual application of the IPA have shown, however, the IPA cannot be seen as a standard, but rather as a set of recommendations that are often only loosely followed by linguists. First attempts to standardize phonetic transcription systems for the purpose of cross-linguistic applications have, however, been made, and will hopefully gain more acceptance in the future (Anderson et al. forthcoming, https://clts.clld.org).

As a second step, we should invest more time in investigating the systemic aspects of language change cross-linguistically. What I consider important in this context is the notion of *distinctive features* by which linguists try to group sounds into classes. Since feature systems proposed by linguists differ greatly, with some debate as to whether features are innate and the same for all languages, or instead language-specific (see Mielke 2008 for an overview on the problem), a first step would again consist of making the data comparable, rather than trying to decide in favour of one of the numerous proposals in the literature.

As a third step, we need to work on ways to account for the multi-layered aspect of sound sequences. Here, a first proposal, labelled "multi-tiered sequence representation", has already been made by myself (List and Chacon 2015), based on an idea that I had already used for the phonetic alignment algorithm proposed in my dissertation (List 2014), which itself goes back to the handling of hydrophilic sequences in ClustalW (Thompson et al. 1994). The idea is to define a sound sequence as a sequence of vectors, with each vector (called *tier*) representing one distinct aspect of the original word. As this representation allows for an extremely flexible modeling of context — which would just consist of an arbitrary number of vector dimensions that could account for aspects such as tone, stress, preceding or following sounds — this representation would allow us to treat words as sequences of sounds while at the same time accounting for their multi-layered structure. Although there remain many unsolved aspects on how to exploit this specific model for phonetic sequences to induce sound laws from ancestor-descendant data, I consider this to be a first step in the direction of a solution to the problem.



Multi-tiered sequence representation for a fictive word in Middle Chinese.

Outlook

Although it is not necessarily recognized by the field as a real problem of historical linguistics, I consider the problem of automatic sound law induction as a very important problem for our field. If we could infer sound laws from a set of proposed proto-forms and a set of descendant forms, then we could use them to test the quality of the proto-forms themselves, by inspecting the sound laws proposed by a given system. We could also compare sound laws across different language families to see whether we find cross-linguistic tendencies.

Having inferred enough cross-linguistic data on sound laws represented in unified models for sound law notation, we could also use the rules to search for cognate words that have so far been ignored. There is a lot to do, however, until we reach this point. Starting to think about automatic, and also manual, induction of sound laws as a specific task in computational historical linguistics can be seen as a first step in this direction.

References

Anderson, Cormac and Tresoldi, Tiago and Chacon, Thiago Costa and Fehn, Anne-Maria and Walworth, Mary and Forkel, Robert and List, Johann-Mattis (forthcoming) A Cross-Linguistic Database of Phonetic Transcription Systems. *Yearbook of the Poznań Linguistic Meeting*, pp 1-27.

Baxter, William H. (1992) *A handbook of Old Chinese Phonology.* Berlin: de Gruyter.

Bodt, Timotheus A. and List, Johann-Mattis (2019) Testing the predictive strength of the comparative method: An ongoing experiment on unattested words in Western Kho-Bwa langauges. 1-22. [Preprint, under review, not peer-reviewed]

Ciobanu, Alina Maria and Dinu, Liviu P. (2018) Simulating language evolution: A tool for historical linguistics. In: *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pp 68-72.

Dekker, Peter (2018) *Reconstructing Language Ancestry by Performing Word Prediction with Neural Networks.* University of Amsterdam: Amsterdam.

Hall, T. Alan (2000) *Phonologie: Eine Einführung.* Berlin and New York: de Gruyter.

Hartmann, Lee (2003) Phono. Software for modeling regular historical sound change. In: *Actas VIII Simposio Internacional de Comunicación Social*. Southern Illinois University, pp 606-609.

Holman, Eric W. and Wichmann, Søren and Brown, Cecil H. and Velupillai, Viveka and Müller, André and Bakker, Dik (2008) Explorations in automated lexicostatistics. *Folia Linguistica* 20.3: 116-121.

List, Johann-Mattis (2014) *Sequence Comparison in Historical Linguistics.* Düsseldorf: Düsseldorf University Press.

List, Johann-Mattis and Chacon, Thiago (2015) Towards a cross-linguistic database for historical phonology? A proposal for a machine-readable modeling of phonetic context. Paper, presented at the workshop *Historical Phonology and Phonological Theory* [organized as part of the 48th annual meeting of the SLE] (2015/09/04, Leiden, Societas Linguistica Europaea).

List, Johann-Mattis (2019) Automatic inference of sound correspondence patterns across multiple languages. *Computational Linguistics* 1.45: 137-161.

Mielke, Jeff (2008) *The Emergence of Distinctive Features.* Oxford: Oxford University Press.

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

Newman, John and Raman, Anand V. (1999) *Chinese Historical Phonology: Compendium of Beijing and Cantonese Pronunciations of Characters and their Derivations from Middle Chinese.* München: LINCOM Europa.

Schwink, Frederick (1994) *Linguistic Typology, Universality and the Realism of Reconstruction.* Washington: Institute for the Study of Man.

Thompson, J. D. and Higgins, D. G. and Gibson, T. J. (1994) CLUSTAL W: Improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Research* 22: 4673–4680.