The Journal of Open Source Software

helayo: Reconstructing Sanskrit texts from manuscript witnesses

Charles Li[1,2]

1 Centre nationale de la recherche scientifique **2** École des hautes études en sciences sociales

DOI: 10.21105/joss.04022

Software

- Review
- Repository
- Archive

Editor: Gabriela Alessio Robles

Reviewers:

- @kinow
- @xiaohk

Submitted: 01 December 2021
Published: 22 March 2022

Summary

For most ancient and medieval texts, the original text itself is no longer extant in a material form. Instead, we have manuscripts that are copies of copies of copies made over the course of hundreds or thousands of years, which accumulate errors and other changes each time they are transcribed by hand. To reconstruct the original text from these imperfect copies, scholars create a stemma — analogous to an evolutionary tree — to determine the relationships between manuscripts and trace those textual changes over time.

Statement of need

Due to the similarities in the methods used in the fields of textual reconstruction and evolutionary biology, textual scholars have begun to employ software created for biologists to analyze texts. Specifically, textual scholars are now using sequence alignment algorithms and phylogenetic tree-building packages to help reconstruct ancient texts (Maas, 2013; Phillips-Rodriguez, 2007; Salemans, 2000). However, as bioinformatics becomes increasingly sophisticated, its models and algorithms have become more specific and less applicable to non-biological sequences.

`helayo` has been designed from the ground up to perform multiple sequence alignment specifically for Sanskrit texts. Since Sanskrit has been written in over a dozen different scripts, each with their own orthographic peculiarities depending on their time and place, `helayo` performs a crucial pre-processing step in which the texts are normalized so that they can be compared meaningfully (Li, 2017). `helayo` can also tokenize texts either as individual characters or as *akṣaras*, since the Brahmic scripts used to write Sanskrit are abugidas, in which consonant and vowel pairs are written as a single unit.

In addition, a web-based matrix editor can be used to edit an alignment. It can also automatically reconstruct a text based on an alignment and a phylogenetic tree using the Fitch algorithm (Fitch, 1971). A full tutorial, with example files, is available at https://chchch.github.io/sanskrit-alignment/docs.

Implementation

`helayo` is written in Haskell and implements the Center Star multiple sequence alignment algorithm (Gusfield, 1997, pp. 347–350) with an affine gap penalty model (Li, 2021). It can be run in three different tokenization modes (character, akṣara, or whitespace-delimited word) and outputs a TEI XML file which can then be edited using the `matrix editor`.

The `matrix editor` is written in Javascript and can be used either online or offline. It loads both TEI XML alignments produced by `helayo` as well as phylogenetic trees in NeXML format, which can be used together to reconstruct a text.

The Journal of Open Source Software

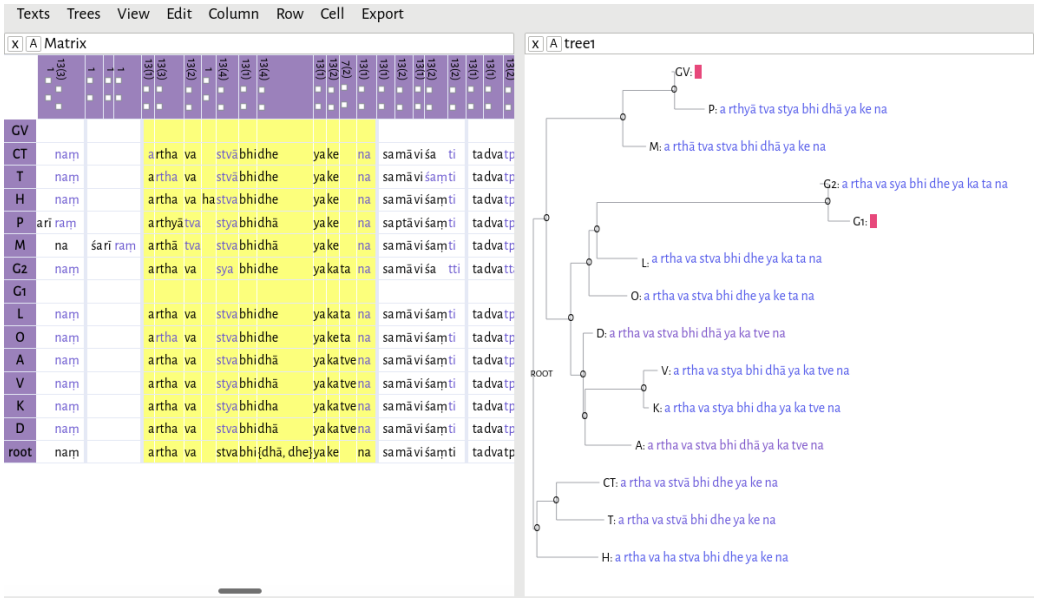

Figure 1: The matrix editor.

Discussion

Previous work on textual phylogeny has focused on texts in Western languages (Barbrook et al., 1998), and, consequently, the tools developed from those projects embody assumptions about how Western languages are spoken, written, and transmitted (e.g. Dekker & Middell, 2019), in the same way that tools produced for bioinformatics embody assumptions about biological sequences. While any set of sequences — composed of text in various languages, of DNA, of proteins — can be analyzed using the same fundamental algorithms, the results are meaningless unless we use domain-specific knowledge to refine those algorithms and to interpret those results.

`helayo`, in turn, has been conceived specifically to align Sanskrit texts, and a great deal of accumulated expertise in Sanskrit philology informs its design. But each step of the process — normalization, tokenization, and sequence alignment — is architecturally distinct, and can be modified to work with other languages. Previously, textual scholars created alignments (or "collations") by hand, comparing different manuscripts to a reference text and using their own judgement to note down the most important differences (Katre & Gode, 1941; West, 1973). The process was slow, but, more importantly, it was not reproducible. By using formal algorithms to align texts, and, furthermore, by refining those algorithms to work with texts from a specific language and tradition, we can begin to create reproducible, testable models of how textual transmission actually works.

References

Barbrook, A. C., Howe, C. J., Blake, N., & Robinson, P. (1998). The phylogeny of the Canterbury Tales. *Nature*, *394*, 839. https://doi.org/10.1038/29667

Dekker, R. H., & Middell, G. (2019). CollateX — software for collating textual sources. In *GitHub repository*. GitHub. https://github.com/interedition/collatex

Fitch, W. M. (1971). Defining the course of evolution: Minimum change for a specific tree topology. *Systematic Zoology*, *20*, 406–316.

Gusfield, D. (1997). *Algorithms on strings, trees, and sequences*. Cambridge University Press.

Katre, S. M., & Gode, P. K. (1941). *Introduction to Indian textual criticism*. Karnatak Publishing House.

Li, C. (2017). Critical diplomatic editing: Applying text-critical principles as algorithms. In P. Boot, A. Cappellotto, W. Dillen, F. Fischer, A. Kelly, A. Mertgens, A.-M. Sichani, E. Spadini, & D. van Hulle (Eds.), *Advances in digital scholarly editing. Papers presented at the DiXiT conferences in The Hague, Cologne, and Antwerp* (pp. 305–310). Sidestone Press.

Li, C. (2021). Align-affine: Sequence alignment with an affine gap penalty model. In *GitHub repository*. GitHub. https://github.com/chchch/align-affine

Maas, P. A. (2013). On what to do with a stemma – towards a critical edition of the Carakasaṃhitā Vimānasthāna 8.29. In D. Wujastyk, A. Cerulli, & K. Preisendanz (Eds.), *Medical texts and manuscripts in Indian cultural history*. Manohar.

Phillips-Rodriguez, W. J. (2007). *Electronic techniques of textual analysis and edition for ancient texts: An exploration of the phylogeny of the Dyūtaparvan* [PhD thesis]. University of Cambridge.

Salemans, B. J. P. (2000). *Building stemmas with the computer in a cladistic, Neo-Lachmannian, way: The case of fourteen text versions of Lanseloet van Denemerken* [PhD thesis]. Katholieke Universiteit Nijmegen.

West, M. L. (1973). *Textual criticism and editorial technique applicable to Greek and Latin texts*. B. G. Teubner.