

# teiphy: A Python Package for Converting TEI XML Collations to NEXUS and Other Formats

Joey McCollum<sup>1</sup> and Robert Turnbull<sup>2</sup>

<sup>1</sup> Institute for Religion and Critical Inquiry, Australian Catholic University, Australia <sup>2</sup> Melbourne Data Analytics Platform, University of Melbourne, Australia

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

## Software

- [Review](#)
- [Repository](#)
- [Archive](#)

Editor: [Open Journals](#)

Reviewers:

- [@openjournals](#)

Submitted: 01 January 1970

Published: unpublished

## License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

## Summary

Textual scholars have been using phylogenetics to analyze manuscript traditions since the early 1990s (Robinson & O'Hara, 1992). Many standard phylogenetic software packages accept as input the NEXUS file format (Maddison et al., 1997). The teiphy program takes a collation of texts encoded in TEI XML format and can convert it to any of the following formats amenable to phylogenetic analysis: NEXUS (with support for ambiguous states and clock model calibration data blocks for MrBayes or BEAST2), Hennig86, PHYLIP (relaxed for use with RAxML), FASTA (relaxed for use with RAxML), STEMMA (a unique format designed for Stephen C. Carlson's stemmatic software tailored for textual data). For machine learning-based analyses, teiphy can also convert a TEI XML collation to a collation matrix in NumPy, Pandas DataFrame, CSV, TSV, or Excel format.

## Statement of Need

For over a decade, the Text Encoding Initiative has endeavored to provide an international standard for digitally encoding textual information for the humanities (Ide & Sperberg-McQueen, 1995). Their guidelines describe a standard format for encoding material details, textual transcriptions, and critical apparatuses (TEI Consortium, 2022). Due to its rich and well-documented set of elements for expressing a wide range of features in these settings, the Text Encoding Initiative's Extensible Markup Language format (hereafter abbreviated TEI XML) has become the *de facto* format for textual data in the digital humanities (Fischer, 2020). Its expressive power has proven increasingly valuable since its release, as scholars have learned—sometimes the hard way—that digital transcriptions and collations should

1. preserve as much detail as they can from their material sources, including paratextual features;
2. reproduce the text of their sources as closely as possible, with editorial regularizations to things like orthography, accentuation, and scribal shorthand encoded alongside rather than in place of the source text; and
3. describe uncertainties about a source's contents as accurately as possible, allowing for degrees of uncertainty and multiple choices for disambiguations if necessary.

These principles have much bearing on the editing of critical texts, a task fundamental to both digital humanities and classical philology. Within the digital humanities, phylogenetic algorithms developed in the context of evolutionary biology have been popular approaches to this task. Taking the most arduous part of reconstructing a textual tradition and delegating it to a computer proved to be a promising technique, and its successful demonstration with a portion of *The Canterbury Tales* was a milestone in the development of the field (Barbrook et al., 1998). Soon after this, the same methods were applied more comprehensively to the tradition of *Lanseloet van Denemerken* in a work that would formalize many practical rules

for computer-assisted textual criticism (Salemans, 2000). Over the decades preceding and following these developments, biologists have continued to develop and improve phylogenetic methods (Felsenstein, 2004), and textual critics have adapted these improvements and even added their own innovations to make the process more suitable for their purposes (Carlson, 2015; Edmondson, 2019; Hyytiäinen, 2021; Spencer et al., 2002, 2004; Turnbull, 2020).

Phylogenetic algorithms have a natural place in textual criticism given the deep analogy between textual traditions and evolutionary trees of life: a sequence alignment, which consists of taxa, sites or characters, and the states of taxa at those characters, corresponds almost identically to a collation, which consists of witnesses to the text, locations of textual variation (which we will call “variation units” from here on), and the variant readings attested by witnesses at those points.

Most phylogenetic software, however, expects inputs not in TEI XML format, but in NEXUS format (Maddison et al., 1997). This format was conceived with versatility in mind, and this design choice has been vindicated in its applicability with textual data, but NEXUS is neither equipped nor meant to express the same kinds of details that TEI XML is. Conversely, for those interested primarily in working with the collation as an alignment, TEI XML is overkill. Thus, a chasm continues to separate data born in the digital humanities from phylogenetic tools born in the biological sciences, and the only way to bridge it is by conversion.

The challenge is made more daunting by the variety of tools available for phylogenetic and other analyses, some of which expect inputs other than NEXUS files or NEXUS files augmented in different ways. For instance, the cladistic software PAUP\*, which has historically been the tool of choice for text-critical applications that evaluate candidate trees using the criterion of maximum parsimony, reads inputs in NEXUS format, but the TNT software, its main competitor among maximum parsimony-based programs, expects inputs in Hennig86 format (Farris, 1988; Goloboff & Catalano, 2016). While Hennig86 format does not allow for as much flexibility in the input as NEXUS does (e.g., it does not support ambiguous character states that can be disambiguated as some states but not others), TNT’s extensive support for morphological state models makes it potentially more suitable for textual data, and textual critics may prefer it to PAUP\*. In the same regime, Stephen C. Carlson’s *STEMMA* software, initially developed for the cladistic analysis of biblical texts known to be affected by contamination, makes substantial adaptations to the basic maximum-parsimony phylogenetic approach to account for this problem and other constraints common in a text-critical setting (Carlson, 2015); however, the input collation data must be provided in a unique *STEMMA*-specific format. Likewise, among programs that use the maximum likelihood criterion instead of maximum parsimony, IQ-TREE accepts inputs in NEXUS format (Minh et al., 2020), but RAxML interfaces primarily with inputs in PHYLIP and FASTA format (Stamatakis, 2014). Finally, for phylogenetic programs that attempt to estimate the posterior distribution of candidate trees in a Bayesian fashion, MrBayes and BEAST2 both accept inputs in NEXUS format (or can convert NEXUS inputs to their standard input format), but they expect taxon dates (for the calibration of evolutionary clock models) to be specified in the NEXUS file in different code blocks (Bouckaert et al., 2019; Ronquist et al., 2012).

Furthermore, end users of textual collations may be interested in non-phylogenetic analyses. In this case, the desired input format is often not a NEXUS-style sequence alignment, but a collation matrix with a row for each variant reading and a column for each witness. For Python machine-learning libraries like Scikit-learn (Pedregosa et al., 2011), NIMFA (Zitnik & Zupan, 2012), and TensorFlow (Abadi et al., 2015), the standard input format is a NumPy array (Harris et al., 2020), although Pandas DataFrames, which support row and column labels (McKinney, 2010; The pandas development team, 2020), may also be supported. (The latter format also extends the conversion pipeline to many other formats, including CSV, TSV, and Excel files; Pandas DataFrames can even write their contents to database tables.) To give an example, the text of the New Testament has served as a testbed for multiple analyses of this type, which have generally applied clustering and biclustering algorithms to collation matrices (Baldwin, 2010; Finney, 2018; McCollum, 2019; Thorpe, 2002; Willker, 2008). Given the

95 prevalence of efforts like these, the need for a means of converting TEI XML collations to  
96 NumPy collation matrices or labeled Pandas DataFrames is clear.

## 97 Design

98 While the conversion process is a straightforward one for most collation data, various sources  
99 of ambiguity can make a one-to-one mapping of witnesses to readings impossible. One such  
100 source of ambiguity is lacunae, or gaps in the text due to erasure, faded ink, or damage to the  
101 page. Another is retroversions, or readings in the original language of the text reconstructed  
102 through the back-translation of subsequent versions of the text in other languages. Mechanisms  
103 for modeling ambiguous states resulting from situations like these exist in both TEI XML and  
104 NEXUS, and in both parsimony- and likelihood-based phylogenetic methods, ambiguities about  
105 the states at the leaves and even at the root of the tree can be encoded and leveraged in the  
106 inference process. For these reasons, it is imperative to ensure that these types of judgments, as  
107 well as other rich features from TEI XML, can be respected (and, where necessary, preserved)  
108 in the conversion process.

109 Collations should preserve as much detail as possible, including information on how certain  
110 types of data can be normalized and collapsed for analysis. Since one might want to conduct  
111 the same analysis at different levels of granularity, the underlying collation data should be  
112 available for use in any case, and only the output of the conversion should reflect changes  
113 in the desired level of detail. Likewise, as noted in the previous section, uncertainty about  
114 witnesses' attestations should be encoded in the collation and preserved in the conversion of  
115 the collation.

116 For text-critical purposes, differences in granularity typically concern which types of variant  
117 readings we consider important for analysis. At the lowest level, readings with uncertain  
118 or reconstructed portions are almost always considered identical with their reconstructions  
119 (provided these reconstructions can be made unambiguously) for the purpose of analysis.  
120 Defective forms that are obvious misspellings of a more substantive reading are often treated  
121 the same way. Even orthographic subvariants that reflect equally "correct" regional spelling  
122 practices may be considered too common and of too trivial a nature to be of value for  
123 analysis. Other readings that do not fall under these rubrics but are nevertheless considered  
124 manifestly secondary (due to late and/or isolated attestation, for instance), may also be considered  
125 uninformative "noise" that is better left filtered out.

## 126 Use Case

127 Due to the availability of extensive collation data for the Greek New Testament, and because  
128 this project was originally developed for use with such data, we tested this library on a sample  
129 collation of the book of Ephesians in thirty-eight textual witnesses (including the first-hand texts  
130 of manuscripts, corrections made to manuscripts by later hands, translations to other languages,  
131 and quotations from church fathers). The manuscript transcriptions used for this collation  
132 were those produced by the University of Birmingham's Institute for Textual Scholarship and  
133 Electronic Editing (ITSEE) for the International Greek New Testament Project (IGNTP); they  
134 are freely accessible at <https://itseeweb.cal.bham.ac.uk/epistulae/XML/igntp.xml>. To achieve  
135 a balance between variety and conciseness, we restricted the collation to a set of forty-two  
136 variation units in Ephesians corresponding to variation units in the United Bible Societies Greek  
137 New Testament (Aland et al., 2014), which highlights variation units that affect substantive  
138 matters of translation.

139 In our example collation, witnesses are described in the `listWit` element under the `teiHeader`.  
140 Because most New Testament witnesses are identified by numerical Gregory-Aland identifiers,  
141 these witnesses are identified with `@n` attributes; the recommended practice is to identify such  
142 elements by `@xml:id` attributes, but this software is designed to work with either identifying

143 attribute (preferring `@xml:id` if both are provided), and we have left things as they are to  
144 demonstrate this feature.

145 The witness elements in the example collation also contain `origDate` elements that provide  
146 dates or date ranges for the corresponding witnesses. Where a witness can be dated to a  
147 specific year, the `@when` attribute is sufficient to specify this; if it can be dated within a range  
148 of years, the `@from` and `@to` attributes or the `@notBefore` and `@notAfter` attributes should  
149 be used; the software will work with any of these options. While such dating elements are  
150 not required, our software includes them in the conversion process whenever possible. This  
151 way, phylogenetic methods that employ clock models and other chronological constraints can  
152 benefit from this information when it is provided.

153 Each variation unit is encoded as an `app` element with a unique `@xml:id` attribute. Within a  
154 variation unit, a `lem` element without a `@wit` attribute presents the main text, and it is followed  
155 by `rdg` elements that describe variant readings (with the first `rdg` duplicating the `lem` reading  
156 and detailing its witnesses) and their attestations among the witnesses. (Situations where  
157 the `lem` reading is not duplicated by the first `rdg` element, but has its own `@wit` attribute,  
158 are also supported.) For conciseness, we use the `@n` attribute for each reading as a local  
159 identifier; the recommended practice for readings that will be referenced elsewhere is to use  
160 the `@xml:id` attribute, and this software will use this as the identifier if it is specified, but we  
161 have only specified `@xml:id` attributes for `rdg` elements referenced in other variation units to  
162 demonstrate the flexibility of the software. For witnesses with missing or ambiguous readings  
163 at a given variation unit, we use the `witDetail` element. For ambiguous readings, we specify  
164 their possible disambiguations with the `@target` attribute and express our degrees of certainty  
165 about these disambiguations using certainty elements under the `witDetail` element.

166 The [TEI XML file](#) for this example is available in the example directory of the GitHub repository.  
167 Full instructions for converting this file using `teiphy` and analyzing it with several different  
168 phylogenetic packages are provided in the documentation, but here, we will walk through  
169 the command-line arguments involved in converting our example TEI XML collation (1) to a  
170 NEXUS file suitable for use with IQ-TREE, and (2) to input for the STEMMA program.

171 For the first case, let us suppose that we would like to treat reconstructions of unclear or  
172 missing text, defective spellings, and orthographic variations in spelling as trivial variants for  
173 the purposes of our phylogenetic analysis. We can specify this to `teiphy` with the `-t` flag for  
174 each trivial type of reading as follows:

175 `-t reconstructed -t defective -t orthographic`

176 In addition, suppose we would like to treat placeholders for overlapping variants from larger  
177 units and lacunae as missing data. We can specify this to `teiphy` with the `-m` flag for each  
178 type of reading to be read as missing data as follows:

179 `-m overlap -m lac`

180 If, in variation units where manuscripts are corrected or have alternate readings provided by  
181 other hands, our collation adds `*` and `T` suffixes to manuscript sigla to mark the work of the  
182 original hand, we can tell `teiphy` to ignore these suffixes using the `-s` flag with each trivial  
183 suffix:

184 `-s "*" -s T`

185 (Note that because the `*` character is reserved on the command-line, we must place it between  
186 quotation marks directly after the `-s` flag.)

187 When corrections are made to a manuscript, they are typically sporadic, and as a result, the  
188 text of corrector witnesses like 06C1 and 06C2 will tend to be too fragmentary to be useful for  
189 analysis. But if we wish to assume that each corrector tacitly adopted all of the readings from  
190 the previous hand that he or she did not change, then `teiphy` can “fill out” each corrector’s  
191 text using the text of the first hand (in the case of the first corrector) or the filled-out text of

192 the previous corrector (for all subsequent correctors). Thus, 06C1 would replicate the text of  
 193 06\* (i.e., the first hand responsible for the text of 06) where it does not introduce its own  
 194 readings, and 06C2 would then replicate the text of 06C1 where it does not introduce its own  
 195 readings. If we want to apply this transformation during the conversion process, then we can  
 196 specify this with the `--fill-correctors` flag.

197 Because IQ-TREE expects its NEXUS input to contain sequence alignments with single-  
 198 character symbols representing character states, we must tell `teiphy` to write the collation  
 199 data in this format with the `--states-present` flag.

200 Finally, we must specify the required arguments to `teiphy`, which are the input TEI XML  
 201 file (`example/ubs_ephesians.xml`) and the name of the output NEXUS file (`ubs_ephesians-  
 202 iqtrees.nexus`). Note that we do not have to specify the desired output format explicitly;  
 203 `teiphy` will determine from the output filename that it should write a NEXUS file. Combining  
 204 the previous options and arguments, the complete command is

```
205 teiphy -t reconstructed -t defective -t orthographic -m overlap -m lac  

  206 -s "*" -s T --fill-correctors --states-present  

  207 example/ubs_ephesians.xml ubs_ephesians-iqtrees.nexus
```

208 If we pass the resulting NEXUS file to IQ-TREE and specify appropriate settings for our textual  
 209 data (in this case, the Lewis Mk substitution model with ascertainment bias correction), we  
 210 will get an output tree like the one shown in Figure 1.

DRAFT



**Figure 1:** A phylogenetic tree inferred by IQ-TREE for the UBS Ephesians example data with support values on the branches based on 1000 bootstrap replicates. Reconstructed, defective, and orthographic sub-variants were treated as identical to their parent readings, and changes made to the text by later correctors (represented as distinct witnesses with sigla like 06C1 and 06C2) were filled in with the readings of the first hand or the previous corrector where the corrector was not active.

211 If we want to generate input files for the STEMMA program using the same options, only a few  
212 adjustments are required. First, the `--states-present` argument is no longer needed, as this  
213 is only applicable for NEXUS outputs. Second, since multiple files are written for STEMMA  
214 input (namely, a collation file with no file extension and a `.chron` file containing information  
215 about witness dates), we only specify the base of the filename for our output, and we must  
216 therefore specify the desired output format to `teiphy` explicitly with the argument

217 `--format stemma`

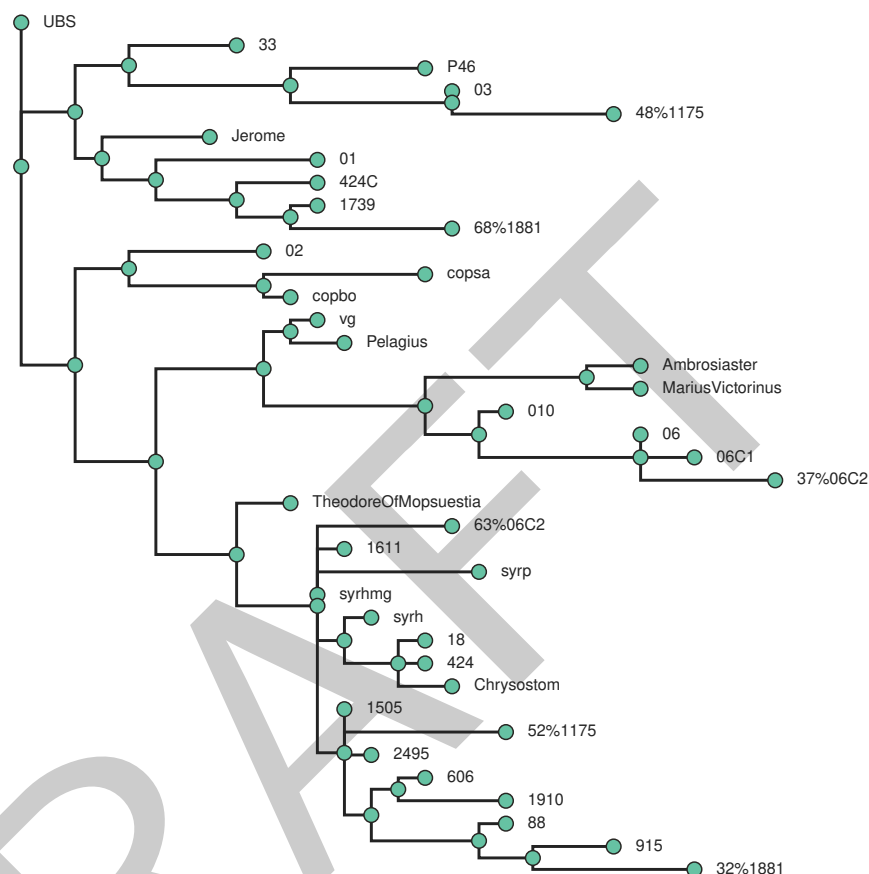
218 Combining the options and arguments we used before with these changes, the complete  
219 command is

```
220 teiphy -t reconstructed -t defective -t orthographic -m overlap -m lac
221 -s "*" -s T --fill-correctors --format stemma
222 example/ubs_ephesians.xml stemma_example
```

223 If we process the output files with the PREP utility that accompanies STEMMA and then pass



the resulting files to STEMMA, we will get an output tree like the one shown in Figure 2.



**Figure 2:** A phylogenetic tree inferred by STEMMA for the UBS Ephesians example data using 100 iterations of simulated annealing. Mixed witnesses are split (with proportions of their readings indicated by the percentages before their sigla) and located at different parts of the tree. Note that some witnesses (e.g., 012, 35) from the collation are excluded from this tree by STEMMA because they have the same reading sequence as another witness after their reconstructed, defective, and orthographic readings have been regularized.

For the small sample of variation units covered in the UBS apparatus for Ephesians, the phylogenetic results depicted in Figures 1 and 2 are impressive. The trees produced by IQ-TREE and STEMMA agree on several traditionally established groupings of manuscripts, including Family 1739 (1739, 1881, and the corrections to 424); the “Western” tradition (as preserved in the Greek-Latin diglots 06, 010, and 012, the Latin Vulgate, and the early Latin church fathers Ambrosiaster, Marius Victorinus, and Pelagius); and the later Byzantine tradition (with representative manuscripts 18 and 35 and church fathers Chrysostom and Theodore of Mopsuestia). The Harklean Syriac translation (syrh) and the witnesses to its Greek *Vorlage* (1505, 1611, 2495) are correctly placed within the Byzantine tradition, although the two programs disagree on how to describe their relationships within that tradition. While IQ-TREE does not account for mixture complicating the tradition, STEMMA identifies three witnesses suspected to exhibit Byzantine contamination: 1175, 1881, and the second corrector of 06. Both programs also identify the Codex Alexandrinus (02) as closely related to both the

238 Sahidic and Bohairic Coptic translations of Ephesians (copsa, copbo), although they disagree  
 239 on where this clade is located in the larger tradition. Despite their discrepancies regarding  
 240 certain subtrees, the extent of their agreements speaks to the level of genealogically significant  
 241 detail preserved in the TEI XML apparatus and the NEXUS and STEMMA inputs generated  
 242 from it.

## 243 Availability

244 The software can be installed through the Python Package Index (PyPI), and the source code  
 245 is available under the MIT license from the [GitHub repository](#). The automated testing suite  
 246 has 100% coverage, and functional tests where our example TEI XML file is converted and  
 247 run through RAXML, IQ-TREE, MrBayes, and STEMMA are part of teiphy's continuous  
 248 integration (CI) pipeline.

## 249 Acknowledgements and Funding

250 The authors wish to thank Stephen C. Carlson for his feedback during the development of  
 251 teiphy and the JOSS reviewers for their thorough and insightful comments on earlier drafts of  
 252 this work. This work was supported by an Australian Government Research Training Program  
 253 (RTP) Scholarship.

## 254 References

- 255 Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis,  
 256 A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia,  
 257 Y., Jozefowicz, R., Kaiser, L., Kudlur, M., ... Zheng, X. (2015). *TensorFlow: Large-Scale*  
 258 *Machine Learning on Heterogeneous Systems*. <https://www.tensorflow.org/>
- 259 Aland, B., Aland, K., Karavidopoulos, J., Martini, C. M., & Metzger, B. M. (Eds.). (2014).  
 260 *The Greek New Testament* (5th ed.). Deutsche Bibelgesellschaft.
- 261 Baldwin, C. S. (2010). Factor Analysis: A New Method for Classifying New Testament Greek  
 262 Manuscripts. *Andrews University Seminary Studies*, 48(1), 29–53.
- 263 Barbrook, A. C., Howe, C. J., Blake, N., & Robinson, P. (1998). The Phylogeny of The  
 264 Canterbury Tales. *Nature*, 394, 839. <https://doi.org/10.1038/29667>
- 265 Bouckaert, R., Vaughan, T. G., Barido-Sottani, J., Duchêne, S., Fourment, M., Gavryushkina,  
 266 A., Heled, J., Jones, G., Kühnert, D., De Maio, N., Matschiner, M., Mendes, F. K.,  
 267 Müller, N. F., Ogilvie, H. A., du Plessis, L., Poppinga, A., Rambaut, A., Rasmussen, D.,  
 268 Siveroni, I., ... Drummond, A. J. (2019). BEAST 2.5: An advanced software platform  
 269 for Bayesian evolutionary analysis. *PLOS Computational Biology*, 15(4), 1–28. <https://doi.org/10.1371/journal.pcbi.1006650>
- 271 Carlson, S. C. (2015). *The Text of Galatians and Its History*. Mohr Siebeck. <https://doi.org/10.1628/978-3-16-153324-2>
- 273 Edmondson, A. C. (2019). *An Analysis of the Coherence-Based Genealogical Method Using*  
 274 *Phylogenetics*. University of Birmingham. <http://etheses.bham.ac.uk/id/eprint/9150>
- 275 Farris, J. S. (1988). *Hennig86, ver. 1.5. Program and Documentation*. James S. Farris.
- 276 Felsenstein, J. (2004). *Inferring Phylogenies*. Sinauer Associates.
- 277 Finney, T. J. (2018). How to Discover Textual Groups. *Digital Studies/Le Champ Numérique*,  
 278 8. <https://doi.org/10.16995/dscn.291>



- 279 Fischer, F. (2020). Representing the Critical Text. In P. Roelli (Ed.), *Handbook of stemmatol-*  
280 *ogy: History, methodology, digital approaches* (pp. 405–427). De Gruyter.
- 281 Goloboff, P. A., & Catalano, S. A. (2016). TNT, Version 1.5, Including a Full Implementation  
282 of Phylogenetic Morphometrics. *Cladistics*, 32(3), 221–238. [https://doi.org/10.1111/cla.](https://doi.org/10.1111/cla.12160)  
283 [12160](https://doi.org/10.1111/cla.12160)
- 284 Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau,  
285 D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van  
286 Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., ...  
287 Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, 585, 357–362. <https://doi.org/10.1038/s41586-020-2649-2>  
288 <https://doi.org/10.1038/s41586-020-2649-2>
- 289 Hyytiäinen, P. (2021). The Changing Text of Acts: A Phylogenetic Approach. *TC: A Journal*  
290 *of Biblical Textual Criticism*, 26, 1–28.
- 291 Ide, N. M., & Sperberg-McQueen, C. M. (1995). The TEI: History, Goals and Future. *Com-*  
292 *puters and the Humanities*, 29(1), 5–15. [https://doi.org/10.1007/978-94-011-0325-1\\_2](https://doi.org/10.1007/978-94-011-0325-1_2)
- 293 Maddison, D. R., Swofford, D. L., & Maddison, W. P. (1997). NEXUS: An Extensible  
294 File Format for Systematic Information. *Systematic Biology*, 46(4), 590–621. <https://doi.org/10.1093/sysbio/46.4.590>  
295 <https://doi.org/10.1093/sysbio/46.4.590>
- 296 McCollum, J. (2019). Biclustering Readings and Manuscripts via Non-negative Matrix Factor-  
297 ization, with Application to the Text of Jude. *Andrews University Seminary Studies*, 57(1),  
298 61–89.
- 299 McKinney, W. (2010). Data Structures for Statistical Computing in Python. In S. van der Walt  
300 & J. Millman (Eds.), *Proceedings of the 9th Python in Science Conference* (pp. 56–61).  
301 <https://doi.org/10.25080/Majors-92bf1922-00a>
- 302 Minh, B. Q., Schmidt, H. A., Chernomor, O., Schrempf, D., Woodhams, M. D., von Haeseler,  
303 A., & Lanfear, R. (2020). IQ-TREE 2: New Models and Efficient Methods for Phylogenetic  
304 Inference in the Genomic Era. *Molecular Biology and Evolution*, 37(5), 1530–1534.  
305 <https://doi.org/10.1093/molbev/msaa015>
- 306 Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel,  
307 M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau,  
308 D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine Learning in  
309 Python. *Journal of Machine Learning Research*, 12(85), 2825–2830.
- 310 Robinson, P., & O'Hara, R. J. (1992). Report on the Textual Criticism Challenge 1991. *Bryn*  
311 *Mawr Classical Review*, 3(4), 331–337.
- 312 Ronquist, F., Teslenko, M., van der Mark, P., Ayres, D. L., Darling, A., Höhna, S., Larget, B.,  
313 Liu, L., Suchard, M. A., & Huelsenbeck, J. P. (2012). MRBAYES 3.2: Efficient Bayesian  
314 phylogenetic inference and model selection across a large model space. *Systematic Biology*,  
315 61(3), 539–542. <https://doi.org/10.1093/sysbio/sys029>
- 316 Salemans, B. J. P. (2000). *Building Stemmas with the Computer in a Cladistic, Neo-*  
317 *Lachmannian, Way: The Case of Fourteen Text Versions of Lanseloet van Denemerken*.  
318 Katholieke Universiteit Nijmegen. <https://hdl.handle.net/2066/147058>
- 319 Spencer, M., Wachtel, K., & Howe, C. J. (2002). The Greek Vorlage of the Syra Harclensis: A  
320 Comparative Study on Method in Exploring Textual Genealogy. *TC: A Journal of Biblical*  
321 *Textual Criticism*, 7. <http://jbtc.org/v07/SWH2002/index.html>
- 322 Spencer, M., Wachtel, K., & Howe, C. J. (2004). Representing Multiple Pathways of Textual  
323 Flow in the Greek Manuscripts of the Letter of James Using Reduced Median Networks.  
324 *Computers and the Humanities*, 38, 1–14. [https://doi.org/10.1023/B:CHUM.0000009290.](https://doi.org/10.1023/B:CHUM.0000009290.14571.59)  
325 [14571.59](https://doi.org/10.1023/B:CHUM.0000009290.14571.59)

- 326 Stamatakis, A. (2014). RAxML Version 8: A Tool for Phylogenetic Analysis and Post-  
327 analysis of Large Phylogenies. *Bioinformatics*, 30(9), 1312–1313. [https://doi.org/10.1093/](https://doi.org/10.1093/bioinformatics/btu033)  
328 [bioinformatics/btu033](https://doi.org/10.1093/bioinformatics/btu033)
- 329 TEI Consortium. (2022). *TEI P5: Guidelines for Electronic Text Encoding and Interchange:*  
330 *Critical Apparatus [v.4.4.0]*. [https://www.tei-c.org/release/doc/tei-p5-doc/en/html/TC.](https://www.tei-c.org/release/doc/tei-p5-doc/en/html/TC.html)  
331 [html](https://www.tei-c.org/release/doc/tei-p5-doc/en/html/TC.html).
- 332 The pandas development team. (2020). *Pandas-dev/pandas: pandas*. Zenodo. [https:](https://doi.org/10.5281/zenodo.3509134)  
333 [//doi.org/10.5281/zenodo.3509134](https://doi.org/10.5281/zenodo.3509134)
- 334 Thorpe, J. C. (2002). Multivariate Statistical Analysis for Manuscript Classification. *TC: A*  
335 *Journal of Biblical Textual Criticism*, 7. <http://jbtc.org/v07/Thorpe2002.html>
- 336 Turnbull, R. (2020). *The Textual History of Codex Sinaiticus Arabicus and Its Family*. Ridley  
337 College.
- 338 Willker, W. (2008). *Principal Component Analysis of Manuscripts of the Gospel of John*.  
339 <http://www.willker.de/wie/TCG/PCA/index.html>
- 340 Zitnik, M., & Zupan, B. (2012). NIMFA: A Python Library for Nonnegative Matrix Factorization.  
341 *Journal of Machine Learning Research*, 13, 849–853.

DRAFT