Medieval Words, Modern Methods

B.Eng.611 / B.DH.11 / SK.DH.04

Revision of October 21, 2025

Term:	Winter 2025–2026	**Instructor:**	Dr P. S. Langeslag
Time:	Tuesdays 10–12	**Office:**	SEP 2.306
Room:	VG 3.103	**Office hours:**	(send me an email)
Credits:	see module description	**E-mail:**	planges@uni-goettingen.de
Prerequisites:	none		

This syllabus comprises an Overview (p. 1), a Schedule (p. 2), and an annotated Bibliography (p. 8).

Overview

Course Description

This seminar is a first introduction to text encoding and corpus evaluation. Carrying out a transcription project of their choice, students will learn how to encode special characters (XML entities), enrich their text with metadata (TEI), style documents for reading (CSS), retrieve XML text nodes and attribute values programmatically (Python), and run basic preprocessing and analytical tasks on the data so retrieved. Although these techniques may usefully be applied to texts of any era, our focus will be on representing, processing, and evaluating text contained in medieval manuscripts. Students will come away with their own XML edition and natural language processing (NLP) pipeline, paving the way for further work on their chosen text in future coursework or thesis research.

Assessment

English Department students should register for the new module B.Eng.611 Old Texts, New Technologies, which supersedes B.Eng.602 for the purposes of this course. Please note that unlike the latter, B.Eng.611 is a seminar-only module with **no associated lecture course!** If you are interested in medieval studies but haven't yet completed B.Eng.602, you may want to consider taking this course and module *alongside* B.Eng.602: choose an additional seminar for B.Eng.602 and attend the lecture course in that context. Only B.Eng.602 gives access to B.Eng.603 and (indirectly) B.Eng.631.

Each student will carry out their individual project transcribing, encoding, processing, and evaluating a text from manuscript images, uploading their work on a regular basis so we can discuss issues encountered and strategies for improvement. The work also entails the writing of a brief scholarly introduction to the text. Students of B.Eng.611 will additionally give a presentation of ca. 10 minutes on aspects of their chosen text, and submit their XML+CSS+Python portfolio for assessment at the end of February; students of B.DH.11 and SK.DH.04 give a 20-minute presentation, and develop their presentation into a ten-page term paper rather than resubmitting the project work itself.

Diversity

This course is run with the understanding that students bring a variety of backgrounds into the classroom in such domains as prior knowledge, socioeconomics, appearance, culture, religion, ability, health, gender, age, family situation, and sexual identity. With different backgrounds come different needs and sensitivities. If you feel your needs or those of a fellow student require special attention or are being compromised, please feel free to make this known to me by whatever channel seems most appropriate. (For more serious concerns, the University has an independent office.) I will treat all requests seriously and with confidentiality, and will seek to make accommodations within my abilities and reason. At the same time, you too owe it to your fellow students to treat them with respect regardless of their background and identity. Do not stand in the way of anyone's well-being.

Resources

Important resources for this course may be found in its dedicated GitHib repository. Apart from XML and CSS templates, it serves up the following important guides:

- The present syllabus;

- A project specification;
- Project suggestions;
- A presentation specification;
- A list of resources;
- A software guide;
- An XML guide;
- A Python guide.

Schedule

Please do the following work *ahead* of the corresponding session, and bring any issues encountered to class for discussion.

Week 1 (28 Oct): Choosing Your Project

Read:

- Project Specification
- Project Suggestions
- Resources

Do:

1. Set up (Git and) VS Code for (at least) XML following p. 2 of the software guide.

Week 2 (4 Nov): Palaeography

Read:

- From **Brown, *A Guide to Western Historical Scripts*: pp. 48–71** (24 pp.)
  Takeaway: This part of the book describes and illustrates Insular scripts above all, but it also spends a few pages on Caroline minuscule in both its Continental and Insular incarnations.
  Reading notes:
  1. A typical script display book, this volume prints colour plates with their descriptions on facing pages. Accordingly, you may find it helpful to set your PDF reader to display two pages at a time (selecting "odd pages left").
  2. The book's dating system works as follows: "*s.* v" is the 5th century (*s.* for *saeculum*); "*s.* x^{in}" is the first quarter of the 10th century (*in* for *ineunte*); "*s.* viii^{med}" is the period 725–775 (*med* for *medio*); "*s.* vii^{ex}" is the last quarter of the 7th century (*ex* for *exeunte*); "*s.* ix^{1}" is the first half of the 9th.
  3. On p. 49 where it says Insular script used more abbreviations, please note that this concerns early *Latin* manuscripts and does not extend to Old English specimens, most of which are later and are comparatively sparing with abbreviations (excepting the nasal stroke typically for *m* or *n* and the Tironian note "⁊" for *and/ond*).

Do:

1. Select an appropriate 300-word text or excerpt for your project, using the project suggestions for inspiration and guidance. Ideally, this should be a medieval verse text, but prose and even early Modern English are admissible, provided they survive in manuscript form and offer sufficient metadata to make for an appropriate encoding challenge. Bring your suggestion to class for approval.

! *Missed this session? Watch* ***Palaeography*** *(45m).*

Week 3 (11 Nov): XML and TEI

Read:

- **pp. 3–4 of the software guide** ("Navigating VS Code")

- **Module 1 of the TEI by Example tutorials** ("Common Structure, Elements, and Attributes").
  Takeaway: The most accessible and a very thorough introduction to TEI.

Do:

1. Finalize your project selection; come prepared to motivate any changes to your topic in class.
2. Follow the instructions on pp. 1–4 of the **XML guide** to acquaint yourself with TEI transcription, as your initial project transcription is due next week! Ideally you'll start your formal transcription ahead of today's session.

Further reading (optional):

- From the ***TEI Guidelines*: ch. 3** ("Elements Available in All TEI Documents")
  Takeaway: The technical specification for typical encoding needs in the TEI standard of XML.
- **Haugen et al., "Menota Handbook"** (v3.0)
  Takeaway: Description of the Menota TEI standard for the transcription of Old Norse manuscript content.

! *Missed this session? Watch **TEI** (47m).*

Week 4 (18 Nov): The Art of the Introduction, and the TEI Header

Read:

- **pp. 4–6 of the XML guide**
- **Module 2 of the TEI by Example tutorials** ("The TEI Header").
  Reading note:
  1. Not all information in this module will be equally relevant to your work. Try to determine which parts will be of use, and study those. Alternatively, read the relevant chapter of the TEI Guidelines directly.

Do:

1. Duplicate `template.xml` into `yourname_transcription.xml` (using your first name and the initial of your surname, e.g. `paul_l_transcription.xml`) and complete at least the most basic details of the TEI header: title, editor, and `<msIdentifier>`.
2. Transcribe and encode your chosen text as a TEI document, accounting for at least special characters (using entities where necessary), abbreviations (using `<ex>`), and basic textual units (`<lg>` (where applicable) and `<l>` for verse, `<p>` for prose). Upload your work to the `Projects` folder. *Your transcription will likely have errors still; you will have the rest of the term (but especially the first few weeks) to correct these, reuploading your transcription with number suffixes to indicate draft maturity, e.g. `paul_l_transcription2.xml`.*

Further reading (optional):

- From the ***TEI Guidelines*: ch. 2** ("The TEI Header"); **ch. 4** ("Default Text Structure")

Week 5 (25 Nov): Styling and Processing XML

Read:

- **Module 4 of the TEI by Example tutorials** ("Primary Sources").

Do:

1. Enrich your transcription by adding any scribal interventions (additions, deletions, replacements, notes) and palaeographical details (large initials, colour) you might want to encode; also consider whether there might be added value in converting your simple `<ex>` expansions into more complex `<choice>` nodes with `<abbr>` and `<expan>`. If you haven't already, this is also a good time to mark up your transcription with `<w>` (word) nodes. Here as with all your project work, note down any issues you encounter and anything you are unsure how to encode, so we can discuss it in class. Upload the revised document to the `Projects` folder.

2. If your excerpt has little by way of interventions and palaeographical detail, use this week as an opportunity to perfect your transcription, and start looking into enriching the TEI header and/or writing a CSS stylesheet.

Further reading (optional):

- **The Learn CSS course at web.dev**
  Takeaway: An excellent, free-as-in-beer CSS course and podcast series.
- From the ***TEI Guidelines***: **§ 12.3** ("Scope of Transcriptions") **down to and including § 12.3.3.2**

Week 6 (2 Dec): Part-of-Speech Tagging

Read:

- **pp. 7–8 of the XML guide** ("Styling and Transformation")
- At least a few pages of the **W3Schools CSS Tutorial**; and/or watch **"Learn CSS in 20 Minutes"** (Web Dev Simplified @ YouTube)
  Takeaway: Introduces the CSS specification. If already familiar, spend time with more advanced features in the web.dev Learn CSS course instead.
  Viewing note:
  1. The Web Dev Simplified video misleadingly uses "smart" quotes in its code slides; only ever use "straight" quotes in code (but "smart" quotes in your coursework prose)!

Do:

1. Duplicate `tei.css` into `yourname_tei.css`, change the stylesheet declaration in the XML accordingly, and customize the stylesheet to your needs. Here too, you will have the rest of the term to finalize your stylesheet, but you'll want to do a meaningful part of the work this week.

Week 7 (9 Dec): Lemmatization

Read:

- **Taylor et al., YCOE website**
  Takeaway: Describes the POS-tagged Old English prose corpus YCOE.
  Reading notes:
  1. Do not read the entirety of the website! Just take stock of the overall system of annotation, e.g. by looking over the POS labels and reading some of the motivation until you understand enough of the general approach to enable you to start using an annotation scheme like this yourself.
- **Schmid, "Deep Learning-Based Morphological Taggers and Lemmatizers for Annotating Historical Texts"** (4 pp.)
  Takeaway: A technical paper describing the methodology behind RNNTagger.
  Reading notes:
  1. This may be a challenging read, but you should have been primed for it in last week's class; either way just make as much sense of it as you can.

Do:

1. If the language of your text sample is represented in RNNTagger, see if you can install the software and determine the correct POS labels, and perhaps the lemmata, mechanically. The easiest way to produce an input text at this point, short of writing an XSLT stylesheet to transform your XML transcription into plaintext, is to copy the text from your browser output and paste it into a plaintext file.
   NB: although the RNNTagger website does not list Old English among the languages for which it has training data, it does in fact have Old English training data for POS, though not for lemmatization.
2. (Alternatively or additionally, you could see whether you can get CLTK 2.0 to identify parts of speech (and more!) for your language. However, this relies on OpenAI API access, which is severely limited for free accounts.)

3. Are you able to eyeball the accuracy of the tagger for your language sample? Come prepared to report on your findings in class.
4. Whether or not you are able to employ a tagger, you are expected to enrich your transcription with a first type of advanced markup, such as POS or lemmata, as attribute data attached to your `<w>` nodes, even if you have to do it manually. Upload the revised XML file to the `Projects` folder.

Week 8 (16 Dec): Poetics

Read:

- **Abrams, "Meter"** (7 pp.), plus whichever of the following is the best match for your project:
  - For Old English: **Marsden, "Beginning Poetry"** (6 pp.) alongside **Mitchell and Robinson, "Metre"** (7 pp.)
  - For Middle English: **Ad Putter, "Verse Forms"** (17 pp.)
    Reading notes:
    1. pp. 128–129: Putter's claim that printing two verses to a line better represents Orm's intentions may seem puzzling in view of the poet's own use of punctuation to separate the two halves of the line, but presumably he means to emphasize that they form a whole. You can find the verse boundaries by Putter's observation that the line contains "four iambic feet in the a-verse and three in the b-verse" (129).
  - For Old Norse: **Poole, "Metre and Metrics"** (19 pp.)
  - For Latin: **pp. 1–15 of Califf, *A Guide to Latin Meter and Verse Composition*** (further exercises at hexameter.co).
  - If your topic is early Modern English, read at least **§§ 0–2 of Kiparsky, "The Rhythmic Structure of English Verse"** (11 pp.; the full article runs 58 pp.)

Do:

1. If your text is a poem, learn its scansion. When in doubt, look it up, e.g. in CLASP for Old English.
2. Use the winter break to learn Python! See the homework for the next three sessions.

Week 9 (6 Jan): Python Basics

Read:

- **Ch. 1 of Bird et al., *Natural Language Processing with Python*** (the equivalent of 33 pp.)
  Takeaway: An accessible and practical introduction to both Python and NLP.
  Reading notes:
  1. Do not track down the hardcopy or PDF book, as it only exists in an outdated first edition; only the HTML edition is sufficiently up to date to reflect current versions of Python and its libraries.
  2. Some chapters (but not all) in the online edition omit the chapter reference in headings; thus "§ 1.1.3" in these notes appears as section "1.3 Searching Text" in the HTML.
  3. The exercises at the end of each chapter are optional; we won't discuss them in class. I do, however, urge you to try out all the book's code examples ("listings") given over the course of each chapter's main content as you do your weekly readings (see "**Do:**" below).
  4. The easiest way to download the textbook materials is simply to enter `nltk.download('book')` instead of `nltk.download()` as described in the book; you can then skip straight to importing.
  5. Please note that `nltk.FreqDist` really just reproduces `collections.Counter`, so the two may be used interchangeably. Just make sure to call it under the name by which you've imported it.

Do:

1. Enrich your XML transcription with a second type of advanced markup, such as lemmata or metrical data, as attribute values associated with your `w` nodes. Upload the revised XML file to the `Projects` folder.
2. Follow along with the code examples in Bird et al. ch. 1.
   Exercise notes:

(a) The textbook assumes you are working in the standalone, text-based interpreter, but we will work in Jupyter notebooks instead, as described on pp. 4–5 of the software guide. Interpret the work accordingly. No need to upload your code.
(b) With Python, it is always possible for functions to change their behaviour as libraries are updated. Discovering how current syntax differs from that in textbooks is part of the puzzle, and is usually helped by a well-phrased web search.

Week 10 (13 Jan): Raw Text Processing

Read:

- **Bird et al. §§ 3.1–3.2, 3.4–3.5** (the equivalent of 25 pp. in all, from ch. 3: "Processing Raw Text")
  Takeaway: Explains how to import text data and use stock functions and regular expressions to manipulate strings.
  Reading note:
  1. § 3.3 is a detailed treatment of text encoding solutions in Python. As long as we ensure we only work on UTF-8 systems and with UTF-8 files, we don't need to worry about this, but do refer back to this section if you run into issues with non-ASCII characters.

Do:

1. Follow along with the code examples in Bird et al. §§ 3.1–3.2, 3.4–3.5.
   Exercise notes:
   - § 3.1 "Electronic Books": The connection to Project Gutenberg appears rather sluggish of late; you may find the download takes 45 seconds to complete. If you encounter an `IncompleteRead` error, your connection was interrupted before the download was completed. Simply rerun your code until it works.
   - § 3.1 "Electronic Books": Since UTF-8 is the default encoding for read operations in Python 3, instead of `raw = response.read().decode('utf8')`, we may as well write `raw = str(response.read())`.
   - § 3.1 "Electronic Books": Though Project Gutenberg is thankfully accessible from Germany again, the files' front and back matter has changed somewhat since the current revision of the book was made available. Thus to locate the start of the back matter you will want to run `raw.rfind("*** END")` rather than `raw.rfind("End of Project Gutenberg's Crime")`.
   - Under § 3.1 "Reading Local Files," the U flag on Python's stock function `open()`, for universal newline mode, has been deprecated and superseded by an option `newline=None`, which is set by default, as is `'r'`; so just use `f = open('document.txt')`. Note that you do have to repeat the `open()` command after running the `.read()` method, as the garbage collector closes it at this point. As you get more Python under your belt, you'll learn about ways of retaining information that won't require you to reopen files (specifically, the `with` statement).
   - Also under § 3.1 "Reading Local Files," if you are using the remote JupyterLab instance (as opposed to a local IDE) to follow along with the examples, the easiest way to create a file is to select the Text File icon in the launcher category "Other." You can use the "right-click" context menu to rename `untitled.txt` into something more memorable after saving.
2. Copy out your project text from the XML's browser output, paste it into a new file in your coding editor, delete any elements not part of the text proper, and save it as a plaintext file `yourname_pastefile.txt`. Then repeat some of the exercises from Bird et al. §§ 3.1 (Reading Local Files and onwards) and 3.2 with this file as your data source, so as to gain some familiarity with file access and string manipulation. Upload the plaintext file, as well as a notebook `yourname_practice.ipynb` demonstrating your ability at least to open your local file and read it into one or more strings, ideally using the `readlines()` method (see the Real Python guide for more detail).

Week 11 (20 Jan): Parsing XML: The Document Object Model (DOM)

Watch:

- either **Socratica, "XML & ElementTree"** (10m), or else **Max Rohowsky, "Parse XML Files with Python"** (10m);
- as well as **Francesco Cento, Parsing XML with Namespaces with Python** (17m);

- or if this is all familiar territory to you, study the **`lxml.etree` documentation** instead.

Study notes:

1. All these videos describe `xml.etree.ElementTree`. In practice, you are probably better off using `lxml.etree` instead, as it has better XPath support as well as such additional functionality as the `getparent()` method. The `lxml` documentation assumes you are already familiar with `ElementTree`, however, and `lxml` namespaces behave slightly differently, see the documentation.
2. The indentation issue at 8m21s and (implicitly) 8m52s of the Socratica video can be avoided by defining a tail attribute for the last preceding sibling.
3. If you get impatient with the namespace video, the solutions start at 13m20s.

Week 12 (27 Jan): Preprocessing Your Trancription

Read:

- **Bird et al. §§ 3.6–3.10** (15 pp., from ch. 3: "Processing Raw Text")
  Takeaway: Teaches the first stages in the tackling of any text corpus with the help of regular expressions.

Do:

1. Use `lxml.etree` or else `xml.etree.ElementTree` to extract a normalized text (abbreviations resolved, scribal interventions accepted or ignored, editorial emendations accepted) directly from your XML document.
2. Case fold and tokenize your transcription, and consider whether any further normalization is desirable.
3. Write code to output your transcription to `yourname_transformation.txt`, one verse line or prose paragraph to a line.
4. Redefine each token in your data container as a Python dictionary with an entry `text` as well as one entry for each type of metadata you have encoded, e.g. `lemma` and `pos`, so you can query e.g. the part of speech of your first token by accessing `tokens[0]['pos']`, the lemma of your second token by accessing `tokens[1]['lemma']`, etc.
5. If you have encoded metrical metadata, think of a way of encoding this at the verse, line, and/or stanza level as well, e.g. by organizing your tokens into shorter lists corresponding to those units and carrying the relevant metadata.
6. Upload your code as `yourname_evaluation.ipynb`, along with the plaintext file you generated in step 3.

Week 13 (3 Feb): Text Evaluation

(Re)read:

- Bird et al., ch. 1 up to and including § 1.3.4 ("Counting Other Things")
- **Cattanach, `zipfs-law`** (`README.md` and `zipf.py`)

Do:

1. Starting from your evaluation notebook, revisit the lexical diversity formula from Bird et al. § 1.1.4 ("Counting Vocabulary") and apply it to a normalized list of tokens retrieved from your project transcription.
2. If we haven't already got there in class, come up with a way of redefining the `lexical_diversity` function to make it less sensitive to the length (token count) of the document.
3. Revisit Bird et al. § 1.3 "Computing with Language: Simple Statistics" and generate a full frequency ranking of the letter characters in your text, discarding spaces and punctuation.
4. If your text is poetry, take a cue from Bird et al. § 1.3.4 ("Counting Other Things") and determine the number of words per verse line (or alternatively, for Old English: per halfline) of your text. If you can find a way, also count the average number of syllables per line (or halfline). If your text is prose, count the average number of characters per word instead.
5. Upload your updated notebook to the `Projects` folder.

Further reading (optional):

- **Trott, "Intro to Text Processing, NLP, and Corpus Linguistics"** (esp. part 1; also available in notebook format here)

- **Desagulier, "Statistics for Text Analysis"** (10 pp.)

Week 14 (10 Feb): Visualization

Read: At least one of the following:

- **Trott, "Introduction to Data Visualization in Python"** (also available in notebook format here)
- From **Real Python, "Python Plotting With Matplotlib"**
  Reading note:
  1. This is a long article about a complicated library with a convoluted history. You can also just copy out existing graph code from elsewhere, but in that case you'll spend years not knowing why you use the syntax that you do.

Do:

1. Starting from your evaluation notebook, produce at least four graphs based on your transcription (along with a broader corpus of similar texts), of at least two kinds (e.g. bar graph, scatter plot, line plot). Suggestions:
   - A bar graph visualizing an absolute count of the 20 most frequent terms in your document;
   - A bar graph visualizing an absolute count of the Sievers verse types used, if your text is an Old English poem and you have encoded this information;
   - A logarithmic line plot of the Zipf distribution of terms in your document;
   - A bar graph comparing the average number of words (better: syllables) per verse line across a range of poems comparable to the one you transcribed (to this end you need a corpus that is already divided into verse lines, not sentences, so e.g. for Old English you'll need the ASPR or CLASP corpus, not DOEC or YCOEP);
   - A 2d scatter plot of the lexical profiles of peer texts within the genre, obtained through dimension reduction from their TF-IDF vectors.
2. Upload your updated notebook to the `Projects` folder.

Bibliography

§ 1: Textbooks and Reference Works

Abrams, M. H., and Geoffrey Galt Harpham. *A Glossary of Literary Terms*. 11th ed. Stamford, CT: Cengage Learning, 2015.

A valuable reference work; start with the entry entitled "Meter."

Barnard, John, David McKitterich, and I. R. Willison, eds. *The Cambridge History of the Book in Britain*. 7 vols. Cambridge: Cambridge University Press, 1999–2019.

A detailed multivolume history, the first two volumes of which cover the Middle Ages.

Beal, Peter. *A Dictionary of English Manuscript Terminology 1450–2000*. Oxford: Oxford University Press, 2008.

An alphabetically organized explanation of concepts surrounding codicology and palaeography.

Bischoff, Bernhard. *Latin Palaeography: Antiquity and the Middle Ages*. Translated by Dáibhí ó Cróinin and David Ganz. Cambridge: Cambridge University Press, 1993.

The standard textbook on Western palaeography. You may, of course, prefer to read it in the original German.

Bishop, T. A. M. *English Caroline Minuscule*. Oxford: Clarendon, 1971.

A book of plates illustrating the script that dominated Anglo-Latin manuscripts after the 10th-century Benedictine Reform.

Brown, Michelle P. *A Guide to Western Historical Scripts from Antiquity to 1600*. London: British Library, 1990.

A practical guide, containing many plates.

Califf, David J. *A Guide to Latin Meter and Verse Composition*. London: Anthem, 2002.

Cappelli, Adriano. *Dizionario di abbreviature latine ed italiane*. 6th ed. Milan: Ulrico Hoepli.

The go-to handbook of Latin abbreviations. Different editions are available in various formats online.

Coulson, Frank T., and Robert G. Babcock, eds. *The Oxford Handbook of Latin Palaeography*. Oxford: Oxford University Press, 2020.

The definitive guide to Latin script and book history, in 1072 pp. and with greyscale plates (some colour in the ebook edition).

Eggert, Paul. "Apparatus, Text, Interface: How to Read a Printed Critical Edition." In *The Cambridge Companion to Textual Scholarship*, edited by Neil Fraistat and Julia Flanders, 97–118. Cambridge: Cambridge University Press, 2013. https://doi.org/10.1017/CCO9781139044073.

A discussion of editorial and layout approaches, though without graphical illustration and focused rather exclusively on the reception of texts first published in Modern print.

Fraistat, Neil, and Julia Flanders, eds. *The Cambridge Companion to Textual Scholarship*. Cambridge: Cambridge University Press, 2013. https://doi.org/10.1017/CCO9781139044073.

A rich selection of approaches, each offering a different introductory angle on the field of textual scholarship.

Hanna, Ralph. *Editing Medieval Texts: An Introduction*. Using exemplary materials derived from Richard Rolle, *Super canticum* 4. Liverpool: Liverpool University Press, 2015.

An accessible introduction, walking the reader through the editing of Rolle's commentary on the Song of Songs.

Haugen, Odd Einar, ed. "The Menota Handbook: Guidelines for the Electronic Encoding of Medieval Nordic Primary Sources," December 12, 2019. https://www.menota.org/handbook.xml.

A detailed manual to the encoding of medieval text.

Haugen, Odd Einar, Caroline Macé, and Philipp Roelli, eds. "Parvum lexicon stemmatologicum," November 13, 2015. https://www.zora.uzh.ch/id/eprint/121539/.

A detailed glossary of text-critical terminology.

Huskey, Samuel J., and Hugh Cayless. "Guidelines for Encoding Critical Editions for the Library of Digital Latin Texts," March 4, 2020. https://digitallatin.github.io/guidelines.

A manual for TEI-encoding works for LDLT.

Langeslag, Paul, Heike Sahm, Roland Scheel, and Michael Schwarzbach-Dobson, eds. *Die Anfänge germanischsprachiger Literaturen: Ein interdisziplinäres Studienbuch*. Freiburg im Breisgau: wbg Academic, 2025. https://www.herder.de/wissen/shop/p8/89097-die-anfaenge-germanischsprachiger-literaturen-open-access-pdf/.

An introduction to early Germanic literature, including overviews of the text types represented in each language.

Leneghan, Francis. "Making Sense of Ker's Dates: The Origins of *Beowulf* and the Palaeographers." *Proceedings of the Manchester Centre for Anglo-Saxon Studies Postgraduate Conference* 1 (2005): 2–13.

Reverse-engineers the dating system used in Ker's Catalogue.

Marsden, Richard. "Beginning Poetry." In *The Cambridge Old English Reader*, 2nd ed., 29–34. Cambridge: Cambridge University Press, 2015.

A concise and accessible introduction to the reading of Old English poetry.

Maurer, Karl. "Commonest Abbreviations, Signs, Etc. Used in the Apparatus to a Classical Text." Accessed October 15, 2025. http://udallasclassics.org/wp-content/uploads/maurer_files/APPARATUSABBREVIATIONS.pdf.

A helpful glossary of Latin terms and abbreviations used in apparatuses.

Mitchell, Bruce, and Fred C. Robinson. "Metre: With Examples from the Poems in this *Guide*." In *A Guide to Old English*, 8th ed., 156–162. Malden, MA: Wiley–Blackwell, 2012.

A brief but well-informed introduction.

Piazzoni, Ambrogio, and Anna Berloco. “Latin Paleography: From Antiquity to the Renaissance.” https://spotlight.vatlib.it/latin-paleography.

A resource teaching antique and medieval scripts.

Pluta, Olaf, ed. “Abbreviationes™ Online.” https://abbreviationes.net.

A database of abbreviations, paid through a one-time fee.

Poole, Russell. “Metre and Metrics.” In *A Companion to Old Norse–Icelandic Literature and Culture*, edited by Rory McTurk, 265–84. Oxford: Blackwell, 2005.

A solid but accessible introduction to Old Norse poetics.

Putter, Ad. “The Metres and Stanza Forms of Popular Romance.” In *A Companion to Medieval Popular Romance*, edited by Raluca Radulescu and Cory James Rushton, 111–131. Cambridge: Cambridge University Press, 2009.

———. “Verse Forms.” In *Medieval Poetry: 1100–1400*. Vol. 2 of *The Oxford History of Poetry in English*, edited by Helen Cooper and Robert R. Edwards, 128–144. Oxford: Oxford University Press, 2023.

TEI Consortium. “TEI P5: Guidelines for Electronic Text Encoding and Interchange,” September 4, 2025. https://www.tei-c.org/guidelines/P5/.

The official handbook of the TEI XML framework; essential reading for anyone working in a TEI paradigm.

Terras, Melisa, Edward Vanhoutte, and Ron Van den Branden. “TEI by Example.” Accessed October 15, 2025. https://teybyexample.org.

A thorough introduction to TEI encoding.

Trott, Sean. “Intro to Text Processing, NLP, and Corpus Linguistics.” Accessed October 20, 2025. https://seantrott.github.io/nlp_basics/.

“W3Schools.” https://www.w3schools.com.

A resource for learning web-based scripting, including XML, XSLT, and CSS.

“web.dev.” Accessed October 15, 2025. https://web.dev.

Resources by the Chrome developer relations team, including tutorials on HTML and CSS.

Wills, Tarrin, et al., eds. “MUFI: The Medieval Unicode Font Initiative.” https://mufi.info.

A standard for mapping medieval glyphs onto the unicode private use area.

§ 2: Manuscript Catalogues and Facsimile Databases

“British Library: Digitised Manuscripts.” Accessed October 5, 2025. https://www.bl.uk/research/digitised-manuscripts/.

Portal to the digitized manuscripts of the British Library.

Busch, Nathanael, Jürgen Wolf, et al., eds. “Handschriftencensus: Eine Bestandsaufnahme der handschriftlichen Überlieferung deutschsprachiger Texte des Mittelalters,” 2006–. https://handschriftencensus.de.

An online catalogue of manuscripts containing German.

“Digital Bodleian.” Accessed October 15, 2025. https://digital.bodleian.ox.ac.uk.

Portal to the digitized manuscripts of Oxford’s Bodleian Libraries.

Flüeler, Cristoph, et al., eds. “e-codices.” Accessed October 15, 2025. https://e-codices.ch.

Portal to Switzerland’s digitized manuscripts.

Fox, Peter, John Hatcher, John Haeger, et al., eds. “Parker Library on the Web: Manuscripts in the Parker Library at Corpus Christi College, Cambridge.” Accessed October 15, 2025. https://parker.stanford.edu.

Digital home of a large library, including many Old English manuscripts.

Gneuss, Helmut, and Michael Lapidge. *Anglo-Saxon Manuscripts: A Bibliographical Handlist of Manuscripts and Manuscript Fragments Written or Owned in England Up to 1100*. Toronto: University of Toronto Press, 2014.

A recent catalogue covering both Old English and Latin content, but breaking off before the end of Old English transmission.

"handrit.is." Accessed October 15, 2025. https://handrit.is.

Index of online and offline Scandinavian manuscripts.

Ker, N. R. *Catalogue of Manuscripts Containing Anglo-Saxon*. Oxford: Clarendon, 1957.

The go-to catalogue for manuscripts containing Old English.

Rold, Orietta Da, Takako Kato, Mary Swan, and Elaine Treharne, eds. "The Production and Use of English Manuscripts 1060 to 1220," 2010–13. https://em1060.stanford.edu/.

An online index of manuscripts from the transitional period.

§ 3: Text Corpora and Editions

Clunies Ross, Margaret, Kari Ellen Gade, Guðrún Nordal, Edith Marold, Diana Whaley, and Tarrin Wills, eds. *Skaldic Poetry of the Scandinavian Middle Ages*. Accessed October 15, 2025. https://skaldic.org.

A current series of print editions of Old Norse poetry, as well as its digital home in the form of a collection of databases.

Faulkner, Mark, ed. *TOXIIC: Trinity Old English from the XIIth Century*. Accessed April 15, 2025. https://mark-faulkner.com/toxiic/.

A plaintext corpus, available only by email request.

Healey, Antonette diPaolo, ed. *Dictionary of Old English Web Corpus*. Edited by John Price Wilkin and Xin Xiang. Toronto, 2024. https://doe.artsci.utoronto.ca.

A 3-million-word corpus covering approximately every text (but not every manuscript witness) of Old English. A 2000 release containing HTML and SGML corpora, but not XML, is available from the Oxford Text Archive.

McSparran, Frances, et al., eds. *Corpus of Middle English Prose and Verse*. Ann Arbor, MI. Accessed October 15, 2025. https://quod.lib.umich.edu/c/cme/.

An XML text corpus of a wide range of print editions; XML at `https://public.websites.umich.edu/~pfs/mec/med/`.

"Medieval Nordic Text Archive." https://menota.org.

A database of XML transcriptions of Old Norse texts.

Nevalainen, Terttu, et al., eds. *Corpus of Early English Correspondence*, 1998. https://www.helsinki.fi/en/researchgroups/variation-contacts-and-change-in-english/research/corpus-of-early-english-correspondence.

O'Donnell, Daniel Paul, ed. "Cædmon's Hymn: A Multimedia Study, Edition and Archive," 2005. https://caedmon.seenet.org.

An edition with many witnesses in TEI P4 SGML.

Orchard, Andy, ed. *A Consolidated Library of Anglo-Saxon Poetry*. Accessed October 15, 2025. https://clasp.ell.ox.ac.uk.

An XML corpus of Old English and early Anglo-Latin poetry.

"Oxford Text Archive." https://ota.bodleian.ox.ac.uk/.

A portal providing institutional access to a range of digital corpora. Apparently moving to https://llds.ling-phil.ox.ac.uk.

Pintzuk, Susan, Ann Taylor, Anthony Warner, Leendert Plug, and Frank Beths, eds. *The York–Helsinki Parsed Corpus of Old English Poetry*, 2001. https://www-users.york.ac.uk/~lang18/pcorpus.html.

A subset of DOEC, parsed for syntax and parts of speech.

Rissanen, Matti, et al., eds. *Helsinki Corpus of English Texts*, 2011. https://varieng.helsinki.fi/CoRD/corpora/HelsinkiCorpus/.

A diachronic text corpus with minimal markup except for per-text information provided in the TEI header.

Roelli, Philipp, ed. *Corpus corporum: Repositorium operum latinorum apud universitatem Turicensem*. Accessed October 15, 2025. http://www.mlat.uzh.ch.

Rudolf, Winfried, Thomas N. Hall, Paul Langeslag, Grant L. Simpson, Charles D. Wright, Susan Irvine, Julia Josfeld, et al., eds. "ECHOE Online: Electronic Corpus of Anonymous Homilies in Old English," 2024–. Accessed October 15, 2025. https://echoe.uni-goettingen.de.

Rudolf, Winfried, Paul Langeslag, Sabine Ines Rauch, Esther M. Lemmerz, Julia Josfeld, Irina Rau, Melanie Vollbrecht, et al., eds. "ECHOE Repository," 2024–. Accessed October 15, 2025. https://github.com/ECHOEProject/echoe.

"SEENET: Society for Early English & Norse Electronic Texts." https://www.seenet.org/.

A modest archive of mostly Middle English online editions.

Taylor, Ann, Anthony Warner, Susan Pintzuk, and Frank Beths, eds. *The Toronto-Helsinki-Parsed Corpus of Old English Prose*, 2003. https://www-users.york.ac.uk/~lang22/YCOE/YcoeHome.htm.

A subset of DOEC, parsed for syntax and parts of speech.

§ 4: Programming and Software

Bird, Steven, Ewan Klein, and Edward Loper. *Natural Language Processing with Python*. Sebastopol, CA: O'Reilly, 2009. https://www.nltk.org/book/.

The official book of the NLTK Python library. The HTML version is freely available CC-BY-NC-ND, and unlike the print and Kindle editions it has been updated to work with Python 3 and NLTK 3, so the code is more likely to work. However, even the online edition relies on some deprecated functions.

Bird, Steven, and Liling Tan. *NLTK: Natural Language Toolkit*. https://www.nltk.org/.

A Python NLP library that remains in widespread use, though it is not as advanced as spaCy.

Johnson, Kyle P., Patrick J. Burns, John Stewart, Todd Cook, Clément Besnier, and William J. B. Mattingly. "The Classical Language Toolkit: An NLP Framework for Pre-Modern Languages." In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, 20–29. Association for Computational Linguistics, August 2021. https://doi.org/10.18653/v1/2021.acl-demo.3. https://aclanthology.org/2021.acl-demo.3.

The writeup of the conference presentation announcing the current version of CLTK, a Python library for the processing of dead languages.

Jurafsky, Dan, and James H. Martin. *Speech and Language Processing*. 3rd ed. draft. October 24, 2025. Accessed October 15, 2025. http://web.stanford.edu/~jurafsky/slp3/.

Online drafts of an outstanding textbook that does not teach programming but only the concepts and models used for NLP.

Lane, Hobson, and Maria Dyshel. *Natural Language Processing in Action*. 2nd ed. Shelter Island, NY: Manning, 2025.

Dense and detailed, this textbook assumes a good deal of prior knowledge and isn't the best at explaining concepts. However, if you're willing to challenge yourself, it does cover a lot of ground and it drives home the knowledge using practical Python exercises, as well as explaining key concepts not covered in other textbooks.

Lynn, Shane. "An Introduction to Word Embeddings for Text Analysis." https://www.shanelynn.ie/get-busy-with-word-embeddings-introduction/.

A short and accessible introduction.

———. "Word Embeddings in Python With Spacy and Gensim." https://www.shanelynn.ie/word-embeddings-in-python-with-spacy-and-gensim/.

A hands-on demonstration.

Python Software Foundation. *Python*. V. 3.10.0, October 4, 2020. https://www.python.org/.

An accessible interpreted computer language often used for NLP.

"Real Python." https://realpython.com.

An excellent collection of tutorials and video courses. You can read (most?) text-based tutorials free of charge, though you have to make a free account after a while, and at least the video tutorials rely on a pricey membership model.

spaCy. V. 3.8. Explosion. https://spacy.io/.

A powerful NLP library for Python which, however, has no support for premodern languages.

§ 5: Scholarship

Andrews, Tara, and Caroline Macé, eds. *Analysis of Ancient and Medieval Texts and Manuscripts: Digital Approaches*. Turnhout: Brepols, 2014.

A volume of essays offering current perspectives.

Apollon, Daniel, and Claire Bélisle. "The Digital Fate of the Critical Apparatus." In *Digital Critical Editions*, edited by Daniel Apollon, Claire Bélisle, and Philippe Régnier, 81–113. Topics in the Digital Humanities. Urbana, IL: University of Illinois Press, 2014.

A rather conceptual look at the functions and features of the apparatus in the digital era.

Apollon, Daniel, Claire Bélisle, and Philippe Régnier, eds. *Digital Critical Editions*. Topics in the Digital Humanities. Urbana, IL: University of Illinois Press, 2014.

A useful look at the digital turn in textual scholarship with a focus on digital editions.

Davis, Matthew Evan, Tamsyn Mahoney-Steel, and Ece Turnator, eds. *Meeting the Medieval in a Digital World*. Leeds: ARC Humanities Press, 2018.

A showcase of recent digital approaches to medieval literature; the contribution by Smith and Butler involves NLP.

Desagulier, Guillaume. "Statistics for Text Analysis." Unpublished preprint, 2025. https://shs.hal.science/halshs-05091183v1.

Echard, Siân, and Stephen Partridge, eds. *The Book Unbound: Editing and Reading Medieval Manuscripts and Texts*. Toronto: University of Toronto Press, 2004.

Essays on the editing mostly of specific medieval texts; of little use as a student introduction.

Horák, Aleš, and Adam Rambousek. "Lexicography and Natural Language Processing." In *The Routledge Handbook of Lexicography*, edited by Pedro A. Fuertes-Olivera, 179–196. London: Routledge, 2017.

A brief overview of the value of each of these fields for the other.

Kiparsky, Paul. "The Rhythmic Structure of English Verse." *Linguistic Inquiry* 8, no. 2 (1977): 189–247.

Kleinhenz, Christopher, ed. *Medieval Manuscripts and Textual Criticism*. North Carolina Studies in the Romance Languages and Literatures: Symposia 4. Chapel Hill, NC: University of North Carolina Department of Romance Languages, 1976.

A volume of essays, some on general principles and others on text-critical questions encountered when working with specific Romance texts. Predates the digital turn.

Lapidge, Michael. "Textual Criticism and the Literature of Anglo-Saxon England." *Bulletin of the John Rylands University Library of Manchester* 73, no. 1 (1991): 17–45.

Offers a historical account of editorial approaches in the field and argues that even speculative emendation is to be preferred over letting faulty readings stand.

———. *The Anglo-Saxon Library*. Oxford: Oxford University Press, 2005.

A study of pre-Conquest manuscript provenance.

Machan, Tim William. *Textual Criticism and Middle English Texts*. Charlottesville, VA: University Press of Virginia, 1994.

A theoretical reflection sprinkled with examples from the editing of Middle English.

Neidorf, Leonard, Madison S. Krieger, Michelle Yakubek, Pramit Chaudhiri, and Joseph P. Dexter. “Large-Scale Quantitative Profiling of the Old English Verse Tradition.” *Nature Human Behavior* 3 (6 2019): 560–567.

Plecháč, Petr, Andrew Cooper, Benjamin Nagy, and Artjoms Šeļa. “*Beowulf* Single-Authorship Claim Is Unsupported.” *Nature Human Behavior* 5 (2021): 1481–1483.

A critical response to Neidorf et al.

Smith, William H., and Charles L. Butler. “Statistical Analysis and the Boundaries of the Genre of Old English Prayer.” In Davis, Mahoney-Steel, and Turnator, *Meeting the Medieval in a Digital World*, 11–26.

Applies relative term frequency to compare Old English prayers.

Stokes, Peter A. “Recovering Anglo-Saxon Erasures: Some Questions, Tools, and Techniques.” In *Palimpsests and the Literary Imagination of Medieval England: Collected Essays*, edited by Leo M. Carruthers, Raeleen Chai-Elsholz, and Tatjana Silec, 35–60. The New Middle Ages. New York: Palgrave Macmillan, 2011.

Practical pointers on how to bring out lost readings by way of prosumer imaging tools.

Stolk, Sander, and Thijs Porck, eds. “Exploring Early Medieval English Eloquence: A Digital Humanities Approach with ‘A Thesaurus of Old English’ and Evoke.” *Amsterdamer Beiträge zur älteren Germanistik* 81.

A special issue on the Evoke application and its Old English data sets.

Torabi, Katayoun. “If (Not “Quantize, Click, and Conclude”): {Digital Methods in Medieval Studies}.” In Davis, Mahoney-Steel, and Turnator, *Meeting the Medieval in a Digital World*, 27–44.

A survey arguing that available tools do not replace close reading.