Introduction to CLTK (Classical Language ToolKit)

Eleftheria Chatziargyriou & Clément Besnier

06/11/2019

Outline

CLTK: Philosophy and Organization

Plan

CLTK

- **Free and Open-Source Python** library
- Founded in 2014 by **Kyle P. Johnson**
- Provides **NLP**[1] **tools** for **historical languages**
- **Shares** a **high-quality code** for **academic research**

Co-maintainers: Patrick J. Burns and Kyle P. Johnson.

[1] NLP: Natural Language Processing

CLTK

Main goals:

1. Compile analysis-friendly corpora
2. Collect and generate linguistic data[2]
3. Act as a free and open platform for generating scientific research

[2] https://github.com/cltk/latin_models_cltk

Plan

CLTK among other NLP Tools in Python

- NLP: SpaCy[3], NLTK[4], StanfordNLP[5]
- Python is a programming language widely used by researchers

[3] https://spacy.io/
[4] https://www.nltk.org/
[5] https://stanfordnlp.github.io/stanfordnlp/

Plan

Historical Languages

- Early Antiquity: Sumerian, Akkadian, Old Egyptian, etc
- Late Antiquity: Ancient Greek, Latin, Sanskrit, Classical Chinese, Gothic, etc
- Middle Ages: Medieval Latin, Coranic Arabic, Koine, Old and Middle High German, Old Norse, etc

Historical Languages

- Handles languages written and spoken before Gutenberg
- Documents written in these languages have specific features:
  - often relatively small and fragmentary surviving texts
  - spelling not normalized
  - diachronic component of languages must be handled
  - no more living speakers
  - no more produced texts
- Expert skills needed

Plan

Principles

- Decentralization
- Disintermediation
- Extensibility
- Standardization
- Simplicity

Community Design Principles

- Transparency
- Inclusion
- Multi-disciplinary
- Mutual benefit

Free, open-source

- Free
- MIT license[6], you can share and reuse it, even for commercial code
- Inclusion
- Multi-disciplinary

[6] https://choosealicense.com/licenses/mit/

How to cite the project:

```
@Misc{johnson2014,
    author = {Kyle P. Johnson et al.},
    title = {CLTK: The Classical Language Toolkit},
    howpublished = {\url{https://github.com/cltk/cltk}},
    note = {{DOI} 10.5281/zenodo.<current_release_id>},
    year = {2014--2019},
}
```

You can also cite the precise contributors[7] if you use a specific module.
To this day, CLTK has been cited more than 50 times[8]

[7] https://github.com/cltk/cltk/blob/master/contributors.md
[8] from Google scholar

CLTK – Code and Contribution

Plan

What can CLTK do?

- Corpora importing
- Text preprocessing
  - File Parsing
  - Orthographic Normalization
  - ASCII/Unicode Conversion
  - Stopword Filtering
- Text processing
  - Syllabification
  - Syllable/Word Stressing
  - Phonetic Indexing
  - Word/line Tokenization
  - IPA Transcription
  - Lemmatization
  - Stemming
  - POS Tagging
  - Poetry Scansion
  - Named Entity Recognition

CLTK as **B**asic **LA**nguage **R**esource **K**it (Krauwer 2003, p. 33)

The **basic tools** to make **analysis** and **automatic tasks** on a language like text summarisation, question answering, machine translation, etc.

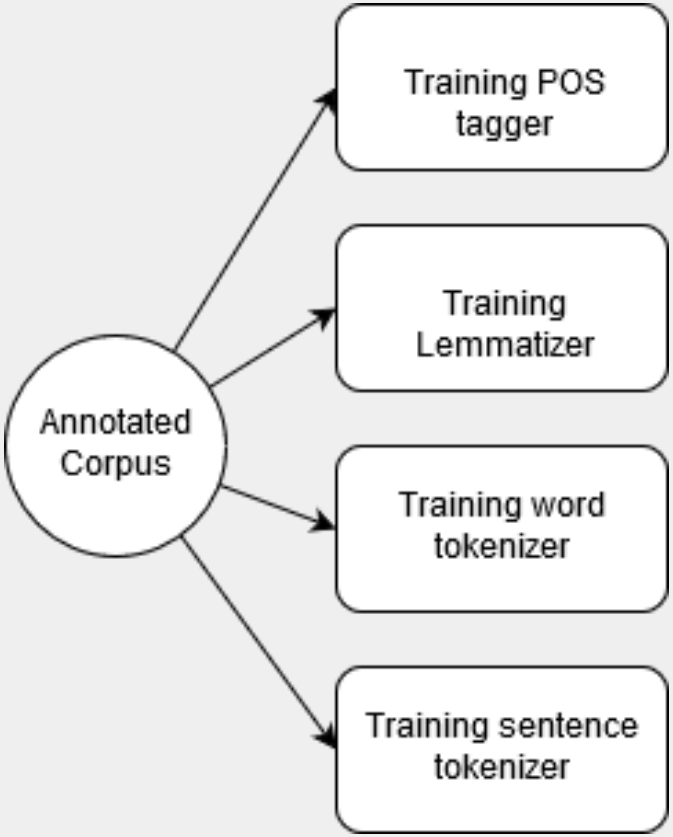
Currently Supported Languages

Corpora: Bengali, Chinese, Coptic, Egyptian, Gujarati, Hebrew, Javanese, Malayalam, Odia, Old Church Slavonic, Old Swedish, Pali, Persian, Prakrit, Telugu, Tibetan, Urdu

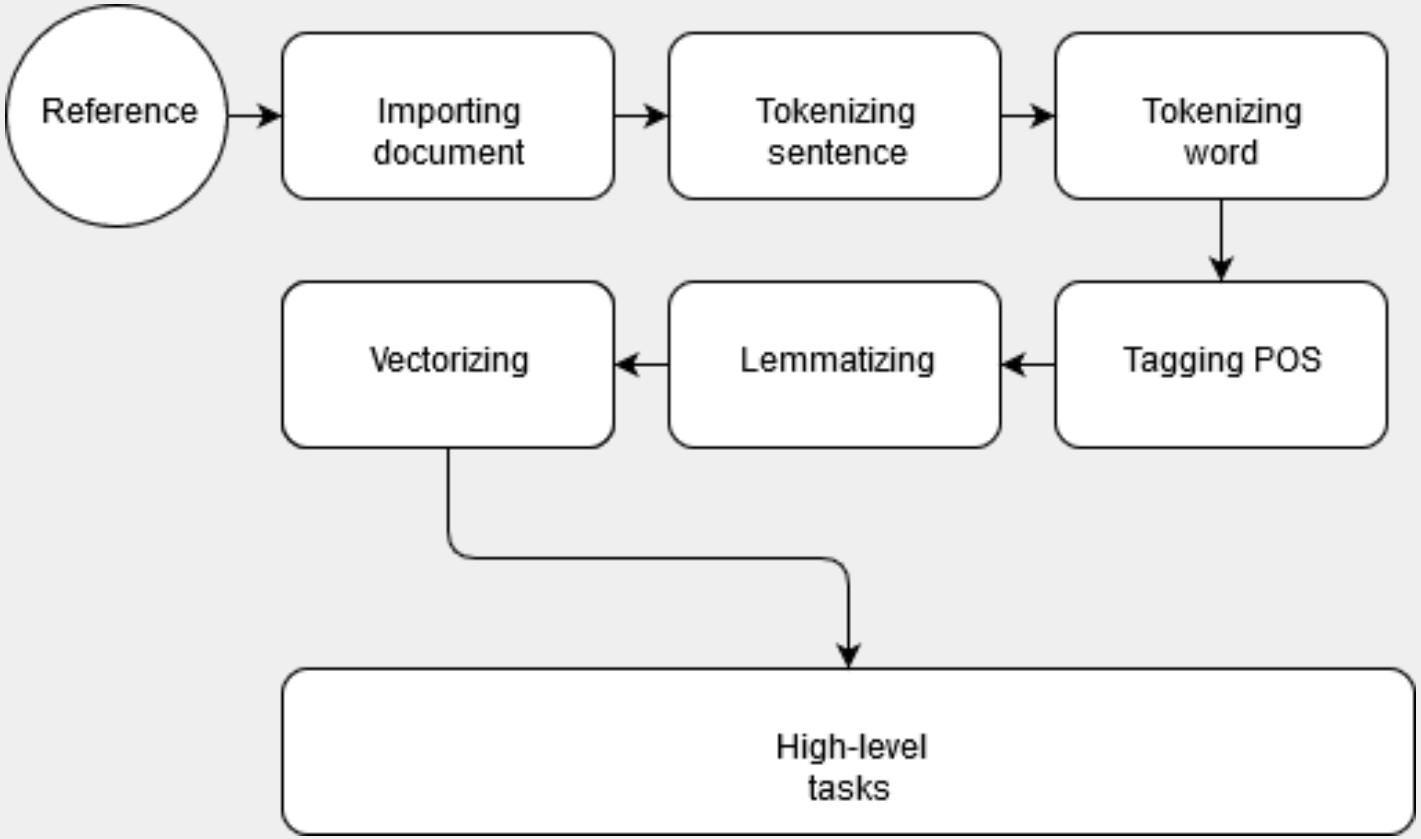
	Akkadian	Arabic	Hindi	**Greek**	**Latin**	Marathi	Middle English	**Middle High German**	Middle Low German	Old English	Old French	Old Norse	Punjabi	Sanskrit
Corpora	•	•	•	•	•	•	•	•	•	•	•	•	•	•
Stoplist	•	•	•	•	•	•	•	•		•	•	•	•	•
Sentence tokenizer				•	•						•			
Word tokenizer	•	•		•	•		•	•			•	•		•
Stemmer	•				•		•	•			•			•
Lemmatizer				•	•		•				•			
POS tagger				•	•			•	•	•		•		
Prosody tagger				•	•			•				•		
NER				•	•						•			

Plan

Building a Text Analysis Pipeline for Classical Languages (Burns 2019, p. 33)



CLTK PIPELINES



Plan

Contribution

- Collaborative effort / open to a virtually infinite talent pool
- Avoid “re-inventing the wheel”
- Closer to the needs of the community
- Constant patches
  - Bugs are quickly resolved
  - New features are constantly developed
- Transparency of development
- Generally results in safer software
- Easily customizable

Why contribute?

- Best reason: ensure scientific reproducibility of your research
- Expand your skill set
- Give back to the community
- Open Source culture
- It's Fun!

How to contribute

- You can check out the **CLTK tutorials**[9] and **docs**[10]
- Take a look at the **open issues**[11] or simply make **your own contribution**[12].
- Don't hesitate to ask for help in the **IRC channel**[13]!

[9] https://github.com/cltk/tutorials
[10] http://docs.cltk.org
[11] https://github.com/cltk/cltk/issues
[12] https://github.com/cltk/cltk/pulls
[13] https://gitter.im/cltk/cltk

Contribute with GSOC

- Open for all university students
- You can work on an open source project for the summer
- CLTK participated for 3 years (2016, 2017, 2018)

Previous GSOC Projects

- Additional support Akkadian, Germanic languages, Old and Middle French
- Greek/Latin Backoff lemmatizers.
- Support of more synonyms, translations and word embeddings for Greek and Latin
- Annotation support for the CLTK Archive

CLTK Stats

- Hosted by Github https://github.com/cltk/cltk
- 2723 commits
- 83 contributors
- 73 watchers, 560 stars, 283 forks
- 45 releases, current version 0.1.112
- Code coverage 89

Summary

- Digital tools can be used to aid academics and speed up mundane and well-defined processes
- Classical languages have their own unique set of challenges compared to modern languages
- CLTK offers an easy to use and well-documented API for Classical Natural Language Processing

Thank you for your attention![14]

[14] You can now get stickers!

Bibliography

- Krauwer, S. 2003. “The Basic Language Resource Kit (BLARK) as the First Milestone for the Language Resources Roadmap.” Proceedings of the 2003 International Workshop on Speech and Computer (SPECOM 2003) : 8-15; cf. also Passarotti in SALTMIL 2010: 29.
- Building a Text Analysis Pipeline for Classical Languages, https://www.degruyter.com/view/books/9783110599572/9783110599010/9783110599572-010.xml

Contact us

- Eleftheria Chatziargyriou, email address: ele.hatzy@gmail.com, Github: Sedictious
- Clément Besnier, email address: clemsciences@aol.com, personal website: clementbesnier.fr, Twitter: clemsciences, Github: clemsciences

- Patrick J Burns, *Constructing stoplists for historical languages*, Digital Classics Online (2018), 4–20.
- Thibault Clérice, *Capitains toolkit, digital editing and data reuse*, Medievales-Paris- **73** (2017), no. 73, 115–131.
- Oksana Dereza, *Lemmatisation for under-resourced languages with sequence-to-sequence learning: A case of early irish*, EPiC Series in Language and Linguistics **4** (2019), 113–124.
- Leopold Hess and Corien Bary, *Narrator language and character language in Thucydides: A quantitative study of narrative perspective*, Digital Scholarship in the Humanities (2019), fqz026.
- Tom Keeline and Tyler Kirby, *Auceps syllabarum: A digital analysis of latin prose rhythm*, Journal of Roman Studies **109** (2019), 161–204.
- Maria Moritz and Marco Büchler, *An automated approach to model the transformation process of the reuse of bernard de clairvaux: How do lexical resources help?.*, DH, 2017.