The ABC of computational Text Analysis

10: NLP with Python

Alex Flückiger
7 May 2020

Recap last Lecture

- introduce Python
  - *editor*
  - *syntax*

Outline

- NLP, getting serious! 
- interactive coding

    interrupt, ask, complement
- mini-project


Primer on NLP

What is a Word?

- word ~ segments between whitespace
- yet, there are ...

  contractions: `U.S.`, `don't`
  collocations: `New York`

Token

- token ~ computational unit

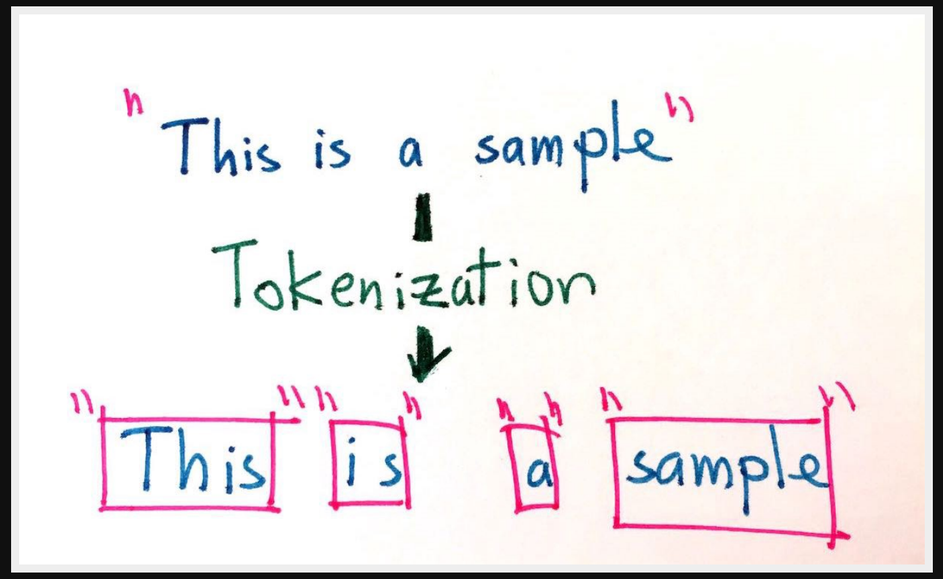
  representation of words

- lemma ~ base form of a word

  texts → text
  goes → go

- stop words ~ functional words

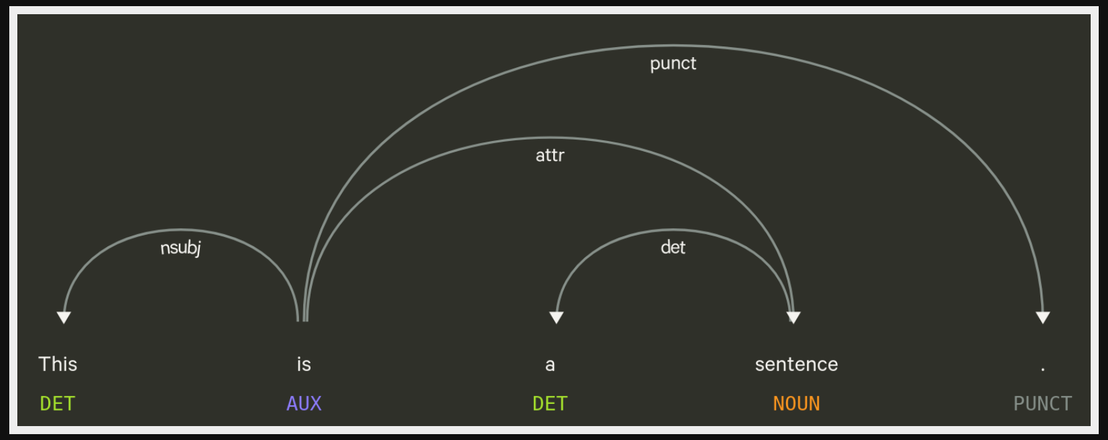
  lack meaning
  the, a, on, and



tokenization (Medium)

Tokenize this: `Let's tokenize! Isn't this easy?` 🤓

NLP Processing Steps



1. tokenizing

    segmenting text into words, punctuations etc.

2. tagging part-of-speech (POS)

    assigning word types (e.g. verb, noun)

3. parsing

    describing syntactic relations

How does the computer “know”?

- using language specific models
- supervised machine learning
- trained on human-annotated data

Representation of Corpus

document term matrix

- Doc 1: `NLP is great. I love NLP.`
- Doc 2: `I understand NLP.`
- Doc 3: `NLP, NLP, NLP.`

	`NLP`	`I`	`is`	***term***
Doc 1	2	1	1	…
Doc 2	1	1	0	…
Doc 3	3	0	0	…
Doc ID	…	…	…	*term frequency*

Mini-Project

- multiple documents
- write a script
- compare ...

  historically
  across actors

- relative frequency 👍, absolute frequency 👎

Optional Seminar Paper

- writing a seminar paper (4 ECTS)
- get in touch to discuss your idea

Outlook NLP

- explore corpus differences by political party
- term frequency over time
- data

    party programmes
    1 August speeches by Swiss Federal Councillors

Do NLP

Check out the Python code

Resources

tutorials on spaCy

- official spaCy 101
- official online course spaCy
- Hitchhiker’s Guide to NLP in spaCy