

Manipulating Text with Python and Spacy Introduction

Virtual Environments and how to install them

- Conda : install miniconda3 on your own if you want to
 - <https://www.youtube.com/watch?v=bbIG5d3bOmk> for mac
 - https://www.youtube.com/watch?v=Avx_FYdFBcc for linux
- Use Venv for now (should come preinstalled with python3) : appears as a directory within your project so it's quite easy to manage
 - To create a venv => `python3 -m venv env_name`
 - To activate it => `source env_name/bin/activate`
 - To deactivate it => `deactivate`
 - To delete it => `rm -r env_name`

Coding exercise

- In your project directory, create a virtual environment called env_gutenberg
- With the oython package manager [pip](#), install jupyterlab and launch jupyter
- You can download the coding exercise notebook from the course github and follow the steps

Intro to Spacy

- Install the spacy package and carry out a couple of NLP analyses of the first paragraph

What is Spacy ?

- spaCy is a free, open-source library for NLP in Python. It's written in [Cython](#) and is designed to build information extraction or natural language understanding systems. It's built for production use and provides a concise and user-friendly API.

How to Download Models

- spaCy has different types of models. The default model for the English language is (usually) en_core_web_sm.
- Download models and data for the English language:
 - python -m spacy download en_core_web_sm
- Import spacy and load the model :
 - import spacy
 - nlp = spacy.load('en_core_web_sm')
- the nlp object here is a « pipeline » : it will allow you to wrap text and analyze it.

Tokenization

- **Tokenization** allows you to identify the basic units in your text.
- These basic units are called **tokens**.
- These units can then be used for further analysis, like part of speech tagging.

Lemmatization

- **Lemmatization** is the process of reducing inflected forms of a word while still ensuring that the reduced form belongs to the language. This reduced form or root word is called a **lemma**.
- For example, *organizes*, *organized* and *organizing* are all forms of *organize*. Here, *organize* is the lemma.
- The inflection of a word allows you to express different grammatical categories like tense (*organized* vs *organizes*), number (*trains* vs *train*), and so on.
- Lemmatization is necessary because it helps you reduce the inflected forms of a word so that they can be analyzed as a single item.

POS (Part of Speech) Tagging

- **Part of speech** or **POS** is a grammatical role that explains how a particular word is used in a sentence. There are eight parts of speech:
 - Noun
 - Pronoun
 - Adjective
 - Verb
 - Adverb
 - Preposition
 - Conjunction
 - Interjection
- **Part of speech tagging** is the process of assigning a **POS tag** to each token depending on its usage in the sentence. POS tags are useful for assigning a syntactic category like **noun** or **verb** to each word.

POS Tagging Ambiguity

The_DT first_JJ time_NN he_PRP was_VBD shot_VBN in_IN the_DT
hand_NN as_IN he_PRP chased_VBD the_DT robbers_NNS outside_RB ...

first	time	shot	in	hand	as	chased	outside
JJ	NN	NN	IN	NN	IN	JJ	IN
RB	VB	VBD	RB	VB	RB	VBD	JJ
		VBN	RP			VBN	NN
							RB

Look at the word « shot » for example....