

Text Analysis and Visualization with Python

Arieda Muço

Central European University

May 22, 2020

- My research focuses in two areas: Political and Development Economics. In my research I deal with tons of data and (lots of) text data. That's why this course.
- Introduce yourself. What are your expectations? Why are you here? What kind of text/data you are currently using or plan to use?

Plan for this course

- Introduction to Python foundations
- Data collection and processing, word counts
- Supervised text methods, classification
- Unsupervised text methods, topic models and clustering
- Data Visualization

Recommended Material

- [Codecademy](#) is the place to start
- [Automate the Boring Stuff with Python](#) and [The Real Python](#) are great sources
- [Introduction to Information Retrieval](#) by Christopher D. Manning, Prabhakar Raghavan and Hinrich Schutze
- [Speech and Language Processing](#) by Dan Jurafsky and James H. Martin
- [Introduction to Machine Learning with Python: A Guide for Data Scientists](#) by Sarah Guido, and Andreas Muller

Before 2000's social scientists avoided studying texts/speech. Why?

- Time Consuming
- Not generalizable (each new data set...new coding scheme)
- Difficult to store/search
- Idiosyncratic to coders/researcher
- Statistical methods/algorithms, computationally intensive
- Hard to find

Massive collections of texts are increasingly used as a data source in social science:

- Congressional speeches, press releases, newsletters,...
- Facebook posts, tweets, emails, cell phone records, ...
- Newspapers, magazines, news broadcasts, ...
- Foreign news sources, treaties, sermons, ...

Why?

Massive increase in availability of unstructured text

- Cheap storage: 1956: \$10,000 megabyte. 2019 :<<<<< \$0.0001 per megabyte
- Explosion in methods and programs to analyze texts
 - ▶ Generalizable: one method can be used across many methods and to unify collections of texts
 - ▶ Systematic: parameters/statistics demonstrate how models make coding decisions
 - ▶ Cheap: easily applied to many new collections of texts, computing power is inexpensive
 - ▶ Replicable: using the same text and method we reach the same conclusions
- Social life (politics, economic exchanges, social interactions) occurs in texts
- Laws, Treaties, News, Campaigns, Petitions, Press Releases

What to do with Text Data?

Growth of a field called Computational Social Science

- Lots of interest across fields
- Computer Science, Computational Linguistics, Education, Sociology, Library and Information Science, Political Science, Communications, Physics, and Economics
- More and more text analysis and machine learning tools are getting incorporated into social scientific research

What is Automated Content Analysis?

- Blanket name for many things
 - ▶ Exploration of text or other media
 - ▶ Using large text corpora as data
 - ▶ Data mining of large variable datasets
- Automated: Computer assigned labels
- Connected to many different literatures
 - ▶ Machine learning
 - ▶ Natural Language Processing
 - ▶ Business Analytics
 - ▶ Visualization of Text
 - ▶ Data Mining
 - ▶ Statistics/Econometrics

What Can Text Methods Do?

Interpreting the meaning of a sentence or phrase. Analyzing a straw of hay

- Haystack metaphor: Improve Reading
 - ▶ Humans: amazing (political theory, analysis of English poetry)
 - ▶ Computers: struggle
- Comparing, Organizing, and Classifying Texts. Organizing hay stack
 - ▶ Humans: terrible. Tiny active memories
 - ▶ Computers: amazing (we'll discuss in this course)

What automated text methods don't do:

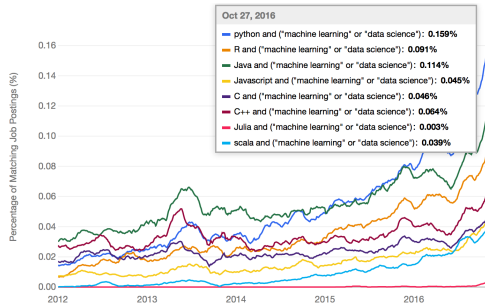
- Replace the need to read
- Develop a single tool + evaluation for all tasks

Why Python?

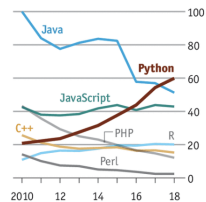
Daily chart

Python is becoming the world's most popular coding language

But its rivals are unlikely to disappear



US, Google searches for coding languages
100 = highest annual traffic for any language

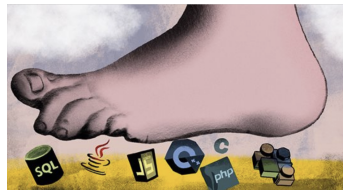


Source: TIOBE, Google Trends

The Economist



Python has soared in popularity since its invention nearly 30 years ago. Today it is the language of choice of AI researchers



ECONOMIST.COM

How Python became a dominant computing language

A bit about Python

- Programming language intended for general purpose high-level language
- Web development, scientific and numeric education, desktop graphical user interface, software development
- Free and open source
- You can do everything that you can do in a programming language
- Big community (Google, Youtube, Nasa...)
- High readability (more than R or C)
- Python was first released in early 1980
 - ▶ Python 2 in 2000 and Python 3 in 2008

Black Holes and Python



Annoying things in Python

- Python 3 is not backward compatible with Python 2
 - ▶ In this course we will use Python 3. Python 2 is not supported anymore
 - ▶ If you are starting a new project, do so in Python 3
- Pandas Library (more on this next time)
 - ▶ But very useful
- + some minor things we'll cover throughout the course
 - ▶ example: `split()` vs `join()`
 - ★ `sentence = "We will rock you!"`
 - ★ `words = sentence.split(" ")` but `sentence = " ".join(words)` (?)

Purpose of the course

- Text Analysis, Machine Learning, Data Visualization, and programming in Python are (mildly put) very broad topics, and we will not be able to cover many(!) things
- Build strong foundations such that in the future you get confidence in starting to dig deeper into these topics

Ada Lovelace a Pythonista

Ada was the first to recognize the full potential of a “computing machine” and one of the first computer programmers.

- Basic concepts:
 - ▶ Variables, subroutines, functions, methods, algorithm
 - ▶ Programs as more than number crunching



Ada's basic concepts in Python

- A variable
 - ▶ `radius = 7`
- A constant
 - ▶ `PI = 3.14159`
- An algorithm
 - ▶ `circumference = 2 * PI * radius`

Ada's basic concepts in Python

```
# A function
def get_circumference(radius):
    circumference = 2 * 3.14159 * radius
    return(circumference)

# Calling the function
get_circumference(4)
```

Time to code!!!