Introduction to Computational Linguistics

Section

Olga Zamaraeva

University of Washington

April 24, 2020

Plan for today: Finding your package

- General remarks
- Issues with reproducibility
- Things to learn
- Demo

A point of view

The best way to write code is not to write it.

The Mythical Man-Month

What happened?!

- Instead of preparing a sandbox, we threw you into the real world
- It has both pros and cons, wrt learning outcomes
  - Making something in the real world run is just about the most useful thing you could learn
  - The time you are spending now is the time you are not writing your own package
    - You are probably **saving** tons of time!
- Doing the project this way is fine so long as you are not punished (in terms of grades) for the world being the way it is

General Remarks about Finding Packages

- The goal is not to make you suffer
- We will not punish you for someone else's poor packaging
- But the assignment was not supposed to be easy either
- You should demonstrate effort for today's checkpoint
- Learning outcomes
  - Very few runnable packages: What does it mean?
  - Making a package run: Sometimes possible, sometimes not
    - If it is possible but not easy, that's OK
    - Only persist with a package for longer than 30 min if you have an idea what you are doing
    - Otherwise just keep looking; you will find something

Issues with Reproducibility

- Reproducibility crisis in research
  - Basically, nothing can be replicated, ever
  - Part of it is because people publish poorly (no data, code doesn't run, no documentation etc.)
  - That's bad
- However!
  - Reproducing a scientific experiment requires expertise
  - Not all of the responsibility is on the publishing researcher
- To summarize: Do not feel like it is your fault that you can't run something but do not always assume the researcher did a bad job either
- This assignment gives you a taste of how hard it is to package something well
- You will all find something in the end

Things to learn about

- Operating systems
  - Packages may only run on some but not all OS
    - Particularly Windows may be harder to find something for
  - Always check which OS is supported
- Different programming platforms
  - Python is most popular and often easiest to set up
  - Other possibilities: Java, C++
    - Only pursue of instructions seem very clear or if you have experience with these languages
- Python versions and modules
  - Python3 vs python2
    - Use the correct version for your package
    - May need to install python2 on new machines and python3 on old machines
    - May need to learn how to set up **virtual environments**

Virtual environments; Installation repositories and tools (pip)

- Virtual environments help aviod conflics between various versions of software
  - May seem like extra work but highly recommended, will save time long term
  - OSX: https://help.dreamhost.com/hc/en-us/articles/215489338-Installing-and-using-virtualenv-with-Python-2
  - Windows: https://programwithus.com/learn-to-code/Pip-and-virtualenv-on-Windows/
- Command line installation
  - Specially packaged software is available in repositories along with installation tools
  - Pip, homebrew, etc
  - Pip is probably a must, try to get it to work
  - The rest: do not persist unless works right away

Demo

- What I did to find a package that runs!
- 20 or so packages, about 2 hours
- Will take a less experienced person more
- ...but that's between all members of your group, and only once in the quarter!