NLTK Tutorial

CSC 485/2501
September 17, 2015

Krish Perumal
krish@cs.toronto.edu / t4peruma@cdf.toronto.edu

Based on slides by Katie Fraser and Sean Robertson

CDF

- Computing Disciplines Facility
  - www.cdf.toronto.edu
- Collection of computer labs and computing environments provided by the University
- Admin office: Bahen
- Most labs in Bahen, one in Gerstein
  - See CDF website for complete list
  - Should be able to access with T-card

CDF Account

- Must be enrolled in CS course
- Account name lookup:
  - http://www.cdf.toronto.edu/resources/cdf_username_lookup.html
  - Requires UTORid
- Password will initially be student number, but you must change it on first log-in
- For more information: User's Guide
  - http://www.cdf.toronto.edu/resources/general_student_guide_to_cdf.html

Accessing CDF outside the lab

- Use ssh (on MacOS, Linux):

```
ssh -Y <CDF_login>@cdf.toronto.edu
```

or

- NX Remote Access (on Windows, MacOS, Linux)
  - Can download and install NX client from CDF webpage -- http://www.cdf.utoronto.ca/using_cdf/remote_access_server.html
  - Step-by-step instructions provided -- https://www.cdf.toronto.edu/nx/nx.php

or

- Use sshfs to mount file system locally on your machine

```
sshfs <CDF_login>@cdf.toronto.edu:<remote_filepath>
<local_mount_path>
```

Submitting Assignments

- From the command line:

```
submit –c <course> –a <assignment_name>
–f <filename_1> … <filename_n>
```

- Can also submit from CDF Student Secure Website -- https://www.cdf.toronto.edu/students/

Python

- High-level, general-purpose language
- Readable code, clear syntax
- Dynamic typing
- Automatic garbage collection and memory management
- Large standard library

Python Editors and IDEs

- Installed on CDF:
  - emacs (powerful, but steep learning curve)
  - IDLE (X forwarding, comes with Python)
- Others:
  - eclipse with Python plug-in (slow, but good)
  - Notepad++ (basic editor with highlighting)

Natural Language Toolkit (NLTK)

- Python package that implements many standard NLP data structures, algorithms
- First developed in 2001 as part of a CL course at University of Pennsylvania
- Many contributors since then
  - led by Steven Bird, Edward Loper, Ewan Klein
- Open-source
- http://www.nltk.org
  - Documentation also at this address

Goals of NLTK

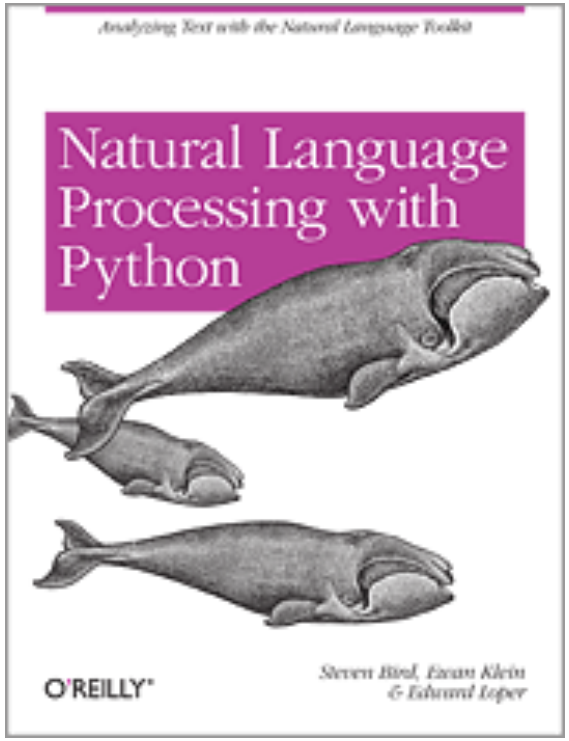
- GOALS:
  - Simplicity
  - Consistency
  - Extensibility
  - Modularity
- NON-GOALS:
  - Encyclopedic coverage
  - Optimization/clever tricks

(Some) Modules in NLTK

Language Processing Task	NLTK module	Some functionalities
Accessing corpora	Nltk.corpus	Standardized interfaces to corpora and lexicons
String processing	Nltk.tokenize	Sentence and word tokenizers
	Nltk.stem	Stemmers
Part-of-speech tagging	nltk.tag	Various part-of-speech taggers
Classification	Nltk.classify	Decision tree, maximum entropy
	Nltk.cluster	K-means
Chunking	Nltk.chunk	Regular expressions, named entity tagging

NLTK Book

- Very useful resource
- Can buy a physical copy (~$45 amazon.ca)
- Also available for free online: http://nltk.org/book/



Python/NLTK Versions

- We will use:
  - Python 2.7
  - NLTK 2.0.4

  (default on CDF)

Accessing Python and NLTK

- Option 1: Log in to your CDF account

  % python

  >>> import nltk

- Option 2: Install on your own machine (but make sure your code for assignments runs on CDF!)
  - Python 2.7 (https://www.python.org/)
  - PyPi (https://pip.pypa.io/en/latest/installing.html)
  - NLTK 2.0.4 (http://www.nltk.org/download)

    pip install nltk

Getting Started: Corpora

- **Task:** Accessing corpora
- **NLTK module:** nltk.corpus
- **Functionality:** standardized interfaces to corpora and lexicons
- Example:

```
>>> from nltk.corpus import gutenberg
>>> gutenberg.fileids()
>>> hamlet = gutenberg.words('shakespeare-hamlet.txt')
>>> hamlet[1:100]
```

- Also: Brown, Reuters, chats, reviews, etc.

Getting Started: String Processing

- **Task:** string processing
- **Modules:** nltk.tokenize, nltk.stem
- **Functionality:** word tokenizers, sentence tokenizers, stemmers
- Example:

```
>>> text = nltk.word_tokenize("The quick brown fox jumps over the lazy dog")

>>> text = nltk.sent_tokenize("The quick brown fox jumps over the lazy dog. What a lazy dog!")

>>> from nltk.stem.wordnet import WordNetLemmatizer

>>> WordNetLemmatizer().lemmatize('dogs','n')

>>> WordNetLemmatizer().lemmatize('jumps','v')
```

Getting Started: Part-of-Speech Tagging

- **Task:** Part-of-speech tagging
- **Module:** nltk.tag
- **Functionality:** Brill, HMM, TnT taggers
- Example:

```
 >>> text = nltk.word_tokenize("It was the best of times, it was the worst of times.")

>>> nltk.pos_tag(text)
```

(Penn Treebank tag set: http://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html)

List of Tutorials

- General Python
  - http://docs.python.org/tutorial
- NLTK-specific
  - http://www.nltk.org/book