Adminstrivia

• Reminder:
  – Homework 1: JM Chapter 1
  – Homework 2: Install Perl and Python (if needed)
Today’s Topics

• App of the Day
  – Homework 3
• Start with Perl
App of the Day

• Text summarization service
  – available on Macs (turned off by default?)
  – Open Text Summarizer (ots) (on Linux)
App of the Day

Open Text Summarizer: Web interface: (maybe others)

– https://www.splitbrain.org/services/ots
Open Text Summarizer (ots)

(code: *read about how it works*)
- https://github.com/neopunisher/Open-Text-Summarizer/

• Web interface
  - http://www.splitbrain.org/services/ots
App of the Day

• System Preferences (10.12.6)
App of the Day

• Let’s try it on the Elon Musk article (from last time)

App of the Day

1. Highlight text
2. Control-click for menu...
App of the Day

We Should Be as Scared of Artificial Intelligence as Elon Musk Is
Homework 3

• due next Monday

1. Run the OTS on the Elon Musk article. What does it report for 5% summarization?

2. Did it do a good job summarizing the article? Explain.

3. Write a single sentence to summarize the article. Explain whether it’s plausible that a program could come with your sentence (or something similar).
Homework 3

• Clickbait:
  – “Clickbait headlines typically aim to exploit the "curiosity gap", providing just enough information to make readers curious, but not enough to satisfy their curiosity without clicking through to the linked content.” (Wikipedia)

A U.S. Army soldier sent her parents a surprise for the ages, and it's pure magic.

upworthy.com
Homework 3

• 4. Extra credit:
  – find a clickbait article
    • (give url, copy text of article if short)
  – text summarize it
  – did the summary bridge the **curiosity gap**?
Homework General Instructions

• Submit your answers by midnight of due date
• Email submission (one file only!) to sandiway@email.arizona.edu
• Format of file should be PDF (no .doc/.docx etc.)
• Subject of email:
  – 538 or 438 Homework # Your NAME
  – Your name at the top of the file (I can’t always tell by the email id.)
  – Also whether 538 or 438.
Programming

• Simplest level:
  – how do I automate repetitive tasks (scripting) on my laptop?

• Search corpora:
  – how do I look for patterns (regex) in a text corpus?
    – grep -Eo ':\s+Yea[^h]' 2150Mon.txt
    – perl -nle 'print $1 if /(.):\s+Yep/' 2149Mon.txt
I recorded something on my iPhone: can I get a transcript of that?

– play a sound file through Enhanced Dictation on OSX (note: not Siri)

Unfortunately still impractical...
Practical Tools

- How do I write a program to grab (scrape) the article and process it?
- programming Language: Python
  - nltk (Natural Language Toolkit)
    - tokenization, POS tag, chunk, parse, etc...
  - urllib
    request.urlopen(url).read().decode('utf8')
  - Beautiful Soup:
    soup = BeautifulSoup(html, 'html')
HLT: Practical Tools

**Program:** cmudict.perl

```perl
use Lingua::EN::CMUDict;
my $obj = new Lingua::EN::CMUDict;
my $n = $obj->number_of_syllables($ARGV[0]);
if ($n) {
    print "$ARGV[0]: $n\n"
} else {
    print "Not in cmudict\n"
}
```

- **cpanm** Lingua::EN::CMUDict  
  *(download from CPAN)*
- **perl** cmudict.perl triangle
  
  triangle: 3
- **perl** cmudict.perl trump
  
  trump: 1
HLT: Practical Tools

Perl

Linguas:EN::CMUDict - Perl extension for utilizing the CMU dictionary file

SYNOPSIS

```perl
use Linguas::EN::CMUDict;
my $obj = new Linguas::EN::CMUDict;
print $obj->number_of_syllables("test");
```

DESCRIPTION

This version of the CMU Pronouncing dictionary was generated from the original dictionary and designed to syllabify it. The paper On the Syllabification of Phonemes by Susan Bartlett, Grzegorz Kondrak and Colin Cherry (NAACL-HLT 2009) covers the methods used to generate the dictionary.

EXPORT

None by default.

METHODS

`new(cmudict=>file)`

Creates a new object, populating it with the cmusyldict db file. If the cmudict argument is passed with a filename as the argument, that file is used. If you do not use that argument, the default cmusyldict db file installed with the module is used.

`rhymes(word)`

In the case of an array being returned, returns all rhymes to the given word. In a scalar context, returns a single rhyme.

`number_of_syllables(word)`

Returns the number of syllables in the word. Many pluralities do not add syllable counts and are therefore not in the original database. This code tries to be intelligent by looking for those and returning the number of syllables. Also, if a sentence is passed in, returns the number of syllables in the sentence. Doesn't currently deal with punctuation very well.

`get_word(word)`

Returns the pronunciation for the word with syllable boundaries.

Python (NLTK)

nlkt.corpus.reader.cmudict module

The Carnegie Mellon Pronouncing Dictionary [cmudict.0.6]


File Format: Each line consists of an uppercase word, a counter (for alternative pronunciations), and a transcription. Vowels are marked for stress (1=primary, 2=secondary, 0=none stress). E.g.: NATURAL I N AE1 CH ER0 AH0 L

The dictionary contains 127069 entries. Of these, 119400 words are assigned a unique pronunciation, 6830 words have two pronunciations, and 839 words have three or more pronunciations. Many of these are fast-speech variants.

Phonemes: There are 39 phonemes, as shown below:

Phoneme Example Translation Phoneme Example Translation
-------- -------------- -------------- --------------
AA odd AA D AE at AE T AH huh HH AH T AO ought AO T AW cow K AW AY hide HH
AY D B be B Y CH cheese CH YZ D dey D Y DH thee DH YH EH Ed EH D ER hurt HH ER T
DY ate EY T F see F Y Y G green G R IR Y N HH he HH YH IH K IH T YH eat IY T JH gee JH YK
key K YL lee L YM me M YN N kne Y NG ping P IH NG D WO oot OW T OY toy T OY P
pee P YR read R YD D sea S YL SH she SY I Y T tea T IY TH theta TH EY T AH hoo HY
HH UH D UW two T UW V vee V IY W we W Y Y yield Y IY L D Z zee Z IY ZH seize S IY ZH ER

```python
class nlkt.corpus.reader.cmudict.CMUDictReader:
```
sandiway$ python3
Python 3.5.2 (v3.5.2:4def2a2901a5, Jun 26 2016, 10:47:25)
[GCC 4.2.1 (Apple Inc. build 5666) (dot 3)] on darwin
Type "help", "copyright", "credits" or "license" for more
information.
>>> from nltk.corpus import cmudict
>>> lookup = cmudict.dict()
>>> lookup['trump']
[['T', 'R', 'AH1', 'M', 'P']]
>>> lookup['triangle']
[['T', 'R', 'AY1', 'AE2', 'NG', 'G', 'AH0', 'L']]
>>>
Perl

- Learn Perl
  - Books...
  - Online resources
    - http://learn.perl.org/
    - we begin with http://perldoc.perl.org/perlintro.html

Basic syntax overview

A Perl script or program consists of one or more statements. These statements are simply written in the script in a straightforward fashion. There is no need to have a `main()` function or anything of that kind.

Perl statements end in a semi-colon:

```
1. print "Hello, world";
```
Perl History

• invented by UCB linguist Larry Wall in the mid-1980s
• Perl stands for “Practical Extraction and Reporting Language”
• It has history:
  – pulls together features from many pre-existing Unix-based tools:
  – efficient text processing: awk, sed
  – search: grep (regex search, see earlier slide)
  – shell scripting: c-shell and others
Perl

- Interpreted language
  
  _no compilation phase down to machine code_ (cf. C, C++), _or a virtual machine_ (cf. Java) so it’s slower
  – lends itself to rapid prototyping, one-off programs, and

- Huge collection of Perl modules (.pm):
  – freely available on CPAN (Comprehensive Perl Archive Network)

  The Comprehensive Perl Archive Network (CPAN) currently has 190,889 Perl modules
Perl

- **Notes from the tutorial:**
  - whitespace not always necessary, e.g.
    - `print"Hello class!\n";`
  - is fine, but good idea to consistently use spacing (not just for readability)
  - variable names must not begin with a number (use a letter), so
    - `$538students` is out
    - `$students538` is ok
  - error messages are frequently completely uninformative (and sometimes misleading), e.g.
    - Bareword found where operator expected at example.prl line 3, near "$538students"
      (Missing operator before students?)
    - *error not associated with the variable starting with a number*