

# LING/C SC/PSYC 438/538

Lecture 3  
Sandiway Fong

# Today's Lecture

- Homework 3 Review
- Quick Homework 4: due tomorrow night
- Some remarks on **Ungrammaticality**
- HLT Practical Tools: CMUDict on Perl and Python
- Ungraded Homework Exercise

# Homework 3 Review

Do parsers distinguish the two distinct senses of preposition *by* highlighted?

Section 1.1. Knowledge in Speech and Language Processing 3

---

- (1.2) How much Chinese silk was exported to Western Europe by the end of the 18th century?

To answer this question, we need to know something about **lexical semantics**, the meaning of all the words (*export* or *silk*) as well as **compositional semantics** (what exactly constitutes *Western Europe* as opposed to Eastern or Southern Europe, what does *end* mean when combined with *the 18th century*). We also need to know something about the relationship of the words to the syntactic structure. For example, we need to know that *by the end of the 18th century* is a temporal end-point and not a description of the agent, as the *by*-phrase is in the following sentence:

- (1.3) How much Chinese silk was exported to Western Europe by southern merchants?

# Homework 3 Review

1

Stanford Parser (not available online)

2

## Your query

*How much Chinese silk was exported to Western Europe by the end of the 18th century?*

## Tagging

How/WRB much/JJ Chinese/JJ silk/NN was/VBD exported/VBN to/IN West  
18th/JJ century/NN ?.

## Parse

```
(ROOT
  (SBAR
    (WHADJP (WRB How) (JJ much))
    (S
      (NP (JJ Chinese) (NN silk))
      (VP (VBD was)
        (VP (VBN exported)
          (PP (IN to)
            (NP (NNP Western) (NNP Europe)))
          (PP (IN by)
            (NP
              (NP (DT the) (NN end))
              (PP (IN of)
                (NP (DT the) (JJ 18th) (NN century))))))))
      (. ?)))
```

## Your query

*How much Chinese silk was exported to Western Europe by southern merchants?*

## Tagging

How/WRB much/JJ Chinese/JJ silk/NN was/VBD exported/VBN to

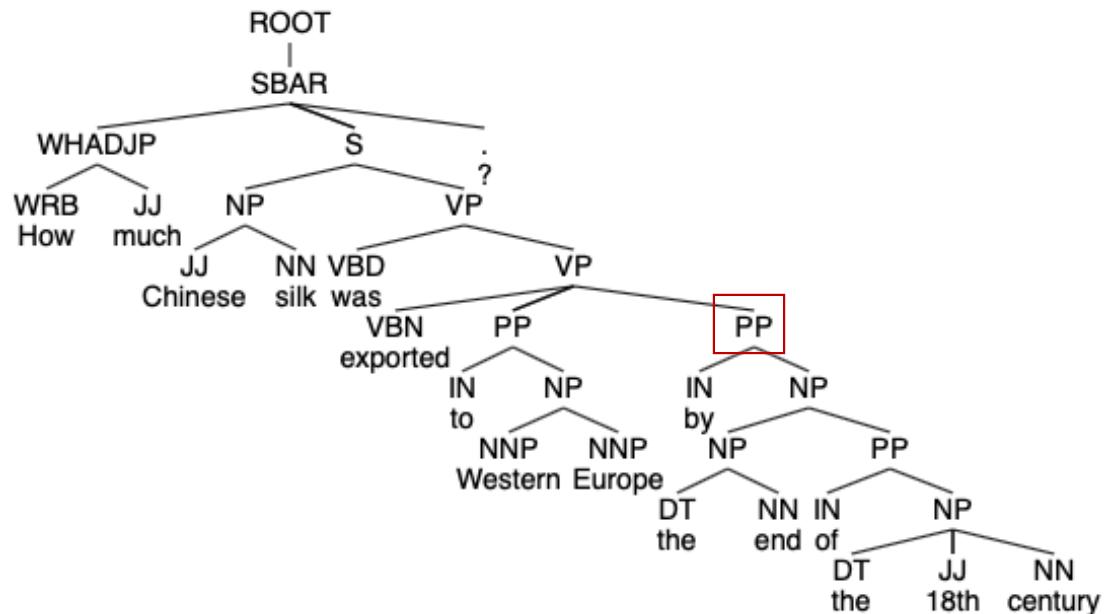
## Parse

```
(ROOT
  (SBAR
    (WHADJP (WRB How) (JJ much))
    (S
      (NP (JJ Chinese) (NN silk))
      (VP (VBD was)
        (VP (VBN exported)
          (PP (IN to)
            (NP (NNP Western) (NNP Europe)))
          (PP (IN by)
            (NP (JJ southern) (NNS merchants))))))
      (. ?)))
```

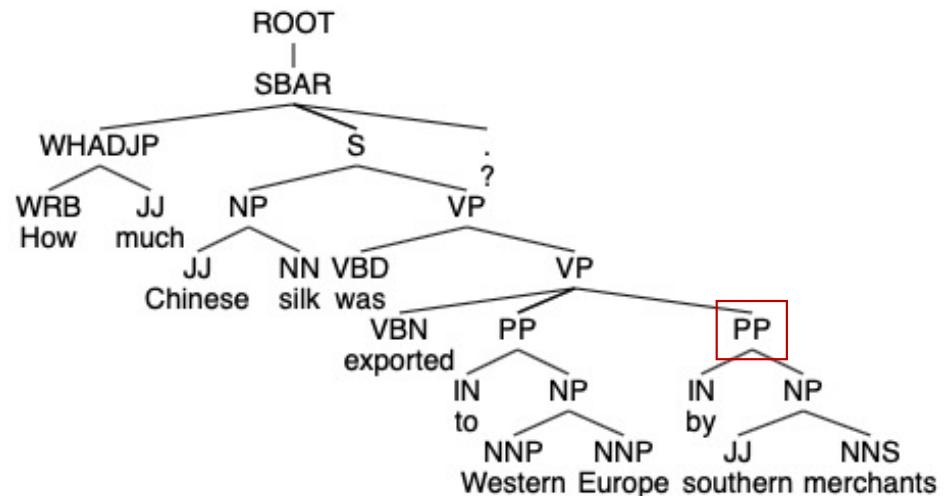
# Homework 3 Review

1

Stanford Parser

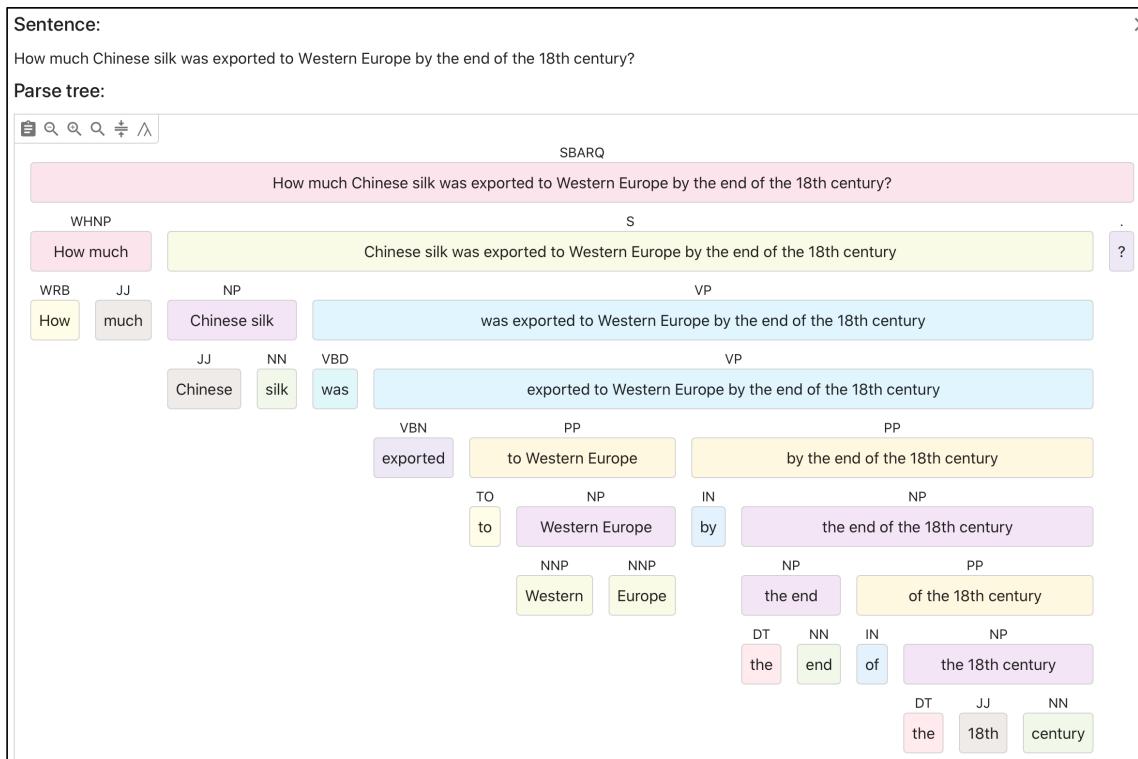


2



# Homework 3 Review

## Berkeley Neural Parser



# Homework 3 Review

1

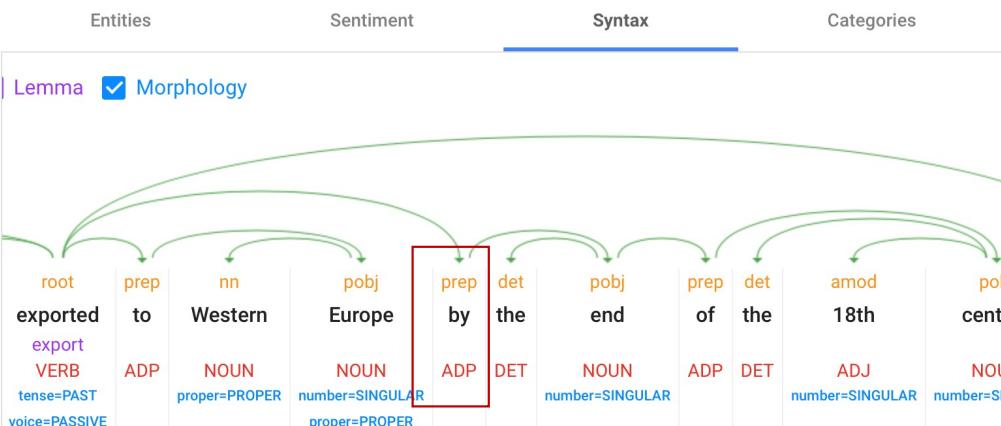
Google Deep Learning parser

2

How much Chinese silk was exported to Western Europe by the end of the 18th century?

RESET

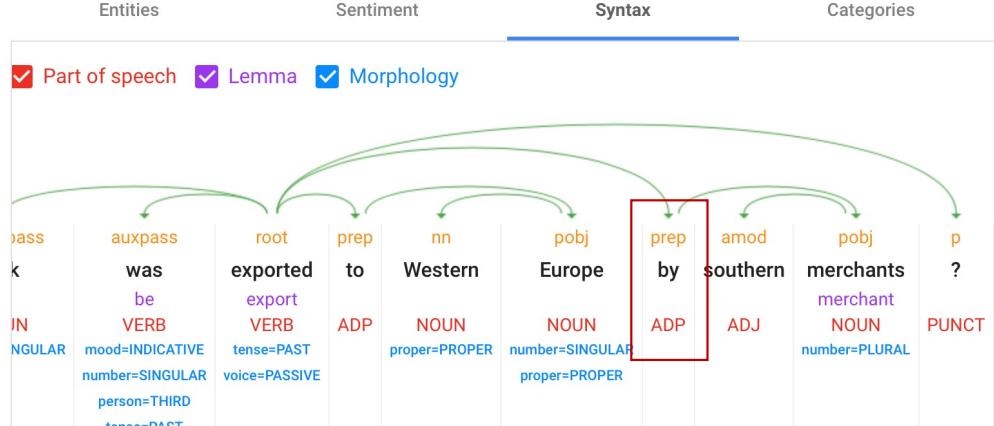
[See supported languages](#)



How much Chinese silk was exported to Western Europe by southern merchants?

RESET

[See supported languages](#)



# Homework 3 Review

---

## Section 1.1. Knowledge in Speech and Language Processing 3

- (1.2) How much Chinese silk was exported to Western Europe by the end of the 18th century?

To answer this question, we need to know something about **lexical semantics**, the meaning of all the words (*export* or *silk*) as well as **compositional semantics** (what exactly constitutes *Western Europe* as opposed to Eastern or Southern Europe, what does *end* mean when combined with *the 18th century*). We also need to know something about the relationship of the words to the syntactic structure. For example, we need to know that *by the end of the 18th century* is a **temporal end-point** and not a description of the **agent**, as the **by-phrase** is in the following sentence:

- (1.3) How much Chinese silk was exported to Western Europe by southern merchants?

# But ...

**Named Entity Recognition:**

1 How much Chinese silk was exported to Western Europe by the end of the 18th century ?

**Basic Dependencies:**

The diagram shows the following dependencies:

- "How" (WRB) is an adverb modifying "much".
- "much" (JJ) is an adjective modifying "Chinese".
- "Chinese" (NN) is a noun.
- "silk" (NN) is a noun.
- "was" (VBD) is a verb in past tense.
- "exported" (VBN) is a verb in past participle form.
- "to" (IN) is a preposition indicating the direction of export.
- "Western" (NNP) is a noun phrase, part of a compound noun.
- "Europe" (IN) is a noun.
- "by" (IN) is a preposition.
- "the" (DT) is a determiner.
- "end" (NN) is a noun.
- "of" (IN) is a preposition.
- "18th" (NN) is a noun.
- "century" (NN) is a noun.
- "?" (PUNCT) is a punctuation mark.

**Named Entity Recognition:**

1 How much Chinese silk was exported to Western Europe by southern merchants ?

**Basic Dependencies:**

The diagram shows the following dependencies:

- "How" (WRB) is an adverb modifying "much".
- "much" (JJ) is an adjective modifying "Chinese".
- "Chinese" (NN) is a noun.
- "silk" (NN) is a noun.
- "was" (VBD) is a verb in past tense.
- "exported" (VBN) is a verb in past participle form.
- "to" (IN) is a preposition indicating the direction of export.
- "Western" (NNP) is a noun phrase, part of a compound noun.
- "Europe" (IN) is a noun.
- "by" (IN) is a preposition.
- "southern" (JJ) is an adjective modifying "merchants".
- "merchants" (NNS) is a noun plural.
- "?" (PUNCT) is a punctuation mark.

- a student's well-argued answer!

This parser appears to distinguish the two senses of the preposition *by* because it was able to determine the correct object for the preposition for both (end vs. merchants) and it was able to identify the phrase *the end of the 18th century* as a temporal entity, rather than as an agent of exporting.

# Homework 4

## 11 main senses (Merriam Webster)

- select word sense(s) from the set  $\{by_1, by_2, by_3, \dots, by_{11}\}$

### by preposition



\ 'bī, before consonants also bē\

#### Definition of by (Entry 1 of 5)

1 : in proximity to : [NEAR](#)  
// standing *by* the window

2 a : into the vicinity of and beyond : [PAST](#)  
// went right *by* him

b : through or through the medium (see [MEDIUM](#) entry 1 sense 2) of : [VIA](#)  
// enter *by* the door

c : in the direction of : [TOWARD](#)  
// sailed north *by* east

3 a : during the course of  
// studied *by* night

b : not later than  
// be there *by* 2 p.m.

4 a : through the agency (see [AGENCY](#) sense 3) or instrumentality of  
// a poem written *by* Keats

// death *by* firing squad

// taken *by* force

// happened *by* luck

b : born or [begot](#) of

// had two sons *by* his first wife

// one child *by* her second husband

c : sired or borne by

// having foals *by* champion race horses

5 : with the witness or sanction (see [SANCTION](#) entry 1 sense 4c) of  
// swear *by* all that is holy

6 a : in conformity with  
// acted *by* the rules

b : ACCORDING TO

// called her *by* name

// sold *by* the pound

7 —used as a function word in multiplication, in division, and in measurements  
// divide a *by* b  
// multiply 10 *by* 4  
// a room 15 feet *by* 20 feet

8 a : with respect to  
// a lawyer *by* profession

b : on behalf of  
// did right *by* his children

9 : in or to the amount or extent of  
// win *by* a nose

b : chiefly Scotland : in comparison with : [BESIDE](#)

10 —used as a function word to indicate successive units or increments  
// little *by* little  
// walk two *by* two

11 : in the opinion of : from the point of view of  
// OK *by* me

#### by the by or by the bye

: [INCIDENTALLY](#) sense 2

# Homework 4: Question 1

## Instructions:

- email answer to me  
([sandiway@arizona.edu](mailto:sandiway@arizona.edu))
- subject line: 438/538  
Homework 4 + **YOUR NAME**
- Due date: tomorrow night

## Same two examples:

1. How much Chinese silk was exported to Western Europe  
**by** the end of the 18th century?
  2. How much Chinese silk was exported to Western Europe  
**by** southern merchants?
- Which Merriam Webster senses for *by* in example (1) and (2)?

# Homework 4: Question 2

<https://open.spotify.com> › track ::

[By the sea - song by Thor's Hammer - Spotify – Web Player](#)

Popular Albums by Thor's Hammer ; My life · 2:21 ; Better days · 2:39 ; I don't care · 2:43 ; The big beat country dance · 1:58 ; If you knew · 2:23.



- Which Merriam Webster senses for the two occurrences of *by* in the highlighted line?

# Language Modeling

- The team *is* vs. The team *are*
  - *The team is* delighted to announce a partnership with SK Telecom T1 , the team's top laner.
  - *The team are* currently working on a new line-up of players to play in the Premier League.
- The government *is* vs. The government *are*
  - *The government is* looking forward to working with the B.C. Chamber of Commerce and other community organizations to ensure that there is a fair, healthy and affordable supply of cannabis.
  - *The government are* yet to decide on the best course of action for ensuring the safety and security of our citizens, and we will continue to monitor and evaluate these developments.

# New-ish UA Slogan

[Launched as a 30 second commercial aired during the 2019 UA football game loss to Hawaii.]

How evocative/memorable do you find the slogan Wonder Makes Us?

- 160over90 has developed a new brand campaign concept to represent the institution to the state and the world ... – demonstrating how **wonder makes us** all the University of Arizona.

Wonder Makes Us

## Conference 2019



University of Arizona ONE Conference 2019

The screenshot shows a digital agenda for the University of Arizona ONE Conference 2019. The agenda is organized by time (10:30am to 4:15pm) and includes several sessions:

- 10:30am: Living Our Values
- 11:00am: Vendor Tables
- 12:00pm: 2019 Youth Outlook: Preparing for a Centennial Future
- 12:30pm: Lunch
- 1:45pm: Wonder Makes Us (This session is highlighted with a red border)
- 2:30pm: (Session title partially visible)
- 3:15pm: (Session title partially visible)
- 4:15pm: (Session title partially visible)

A modal window is open over the "Wonder Makes Us" session, showing details about the presentation:

Add to My Sched | Link

Through collaboration with the University of Arizona, 160Over90 has developed a new brand campaign concept to represent the institution to the state and the world. This presentation will provide an overview of the process to arrive at this idea and showcase the scale and reach of the idea - demonstrating how wonder makes us all the University of Arizona.

About:  
160over90 is a creative agency and strategic consultancy. We help

# "Wonder makes us"

 ALLEN INSTITUTE  
for ARTIFICIAL INTELLIGENCE

This demonstration uses the public 345M 117M parameter OpenAI GPT-2 lan

Enter some initial text and the model will generate the most likely next words. choose it and continue or just keep typing. Click the left arrow at the bottom to

Sentence: Options:

Wonder makes us feel like we're in a movie 21.2% ,  
16.0%

Prediction	Score
Wonder makes us swoon. ↫	58.5%

AI2 Allen Institute for AI

AllenNLP

Sentence: Wonder makes us feel like we're in a different world.

Predictions:

16.5%	←
11.0%	It
10.7%	We
5.2%	I
4.2%	And

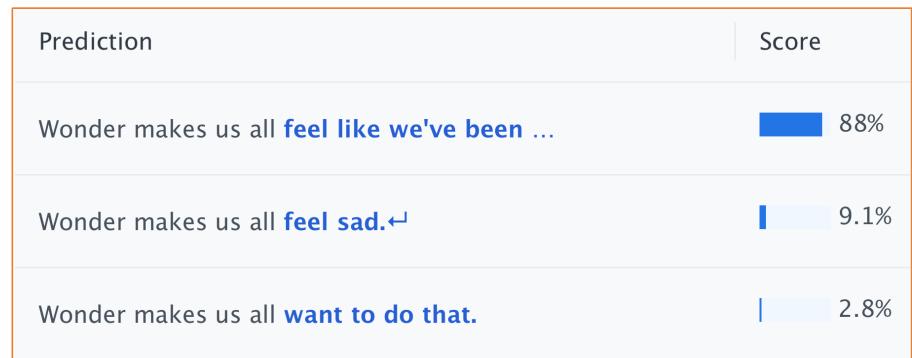
Undo

# *Wonder makes us **all***

- Wonder makes us



- Wonder makes us **all**



# *Wonder makes us all the University*

- Wonder makes us all the

Prediction	Score
Wonder makes us all the <b>more aware of the world ...</b>	97.1%
Wonder makes us all the <b>proudest of our own ...</b>	1.2%
Wonder makes us all the <b>safer, don't you ...</b>	1.1%
Wonder makes us all the <b>best. We love you ...</b>	0.4%

- Wonder makes us all the **University**

Prediction	Score
Wonder makes us all the University <b>of Florida football fans we ...</b>	60.1%
Wonder makes us all the University <b>of Michigan fans we could ...</b>	37.2%
Wonder makes us all the University <b>of Pittsburgh's best, ...</b>	2.5%
Wonder makes us all the University <b>of California, Riverside at ...</b>	0.2%
Wonder makes us all the University <b>beginning to feel that we ...</b>	0%

*Wonder  
makes us all  
the  
University of  
Arizona*

Prediction	Score
Wonder makes us all the University of <b>Alabama-Birmingham</b> .	51.6%
Wonder makes us all the University of <b>Hawai'i</b> .←	44.2%
Wonder makes us all the University of <b>Southern California, but the ...</b>	4.1%
Wonder makes us all the University of <b>Houston that we need, ...</b>	0.2%
Wonder makes us all the University of <b>Connecticut-caliber in talent ...</b>	0%

# Examples

- Ungrammaticality
  - **Which book did you file without reading?**
  - **\*Which book did you file the report without reading?**
  - \* = ungrammatical
  - what rule is being violated here?

Chomsky (1957):

- **Colorless green ideas sleep furiously.**
- **Furiously sleep ideas green colorless.**
- ungrammatical vs. incomprehensible

# Examples

- Ungrammaticality
  - \*Which book did you file the report without reading?
  - \*Which book did you file the report ...

Sentence:

Which book did you file  
the report on

Predictions:

63.3% ?

4.0% ,

2.6% ?"

2.0% and

1.7% the

← Undo

# Examples

- Chomsky (*various recent lectures*):
  - **Can** eagles that fly swim? (English has a rule of auxiliary inversion)
  - two verbs here: but question is about *swimming* not about *flying*
  - cf. Eagles that fly can swim. / Eagles that can fly swim.
  - \***Are eagles that flying swim?**

Sentence:

Can eagles that fly in the sky are not  
protected by the law?

Options:

55.7% ↪

3.9% Yes

2.6% No

# HLT: Practical Tools

The screenshot shows a BBC News page. At the top, there's a navigation bar with links for Home, Video, World, US & Canada, UK, Business, Tech, Science, Magazine, Entertainment & Arts, Health, and More. Below the navigation is a search bar and a sign-in link. The main headline reads "Business councils disband over Trump remarks". The article includes a photo of a protest, a video thumbnail, and a caption: "What Trump said versus what I saw - by the BBC's Joel Gunter". A sidebar on the right lists "Top Stories" such as "Van hits crowds in Barcelona tourist area" and "Raped Indian girl, 10, gives birth". There are also social sharing icons and a "Share" button.

- 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')`

You didn't write that cool page. You're just trying to get some data out of it. Beautiful Soup is here to help. Since 2004, it's been saving programmers hours or days of work on quick turnaround screen scraping projects.

## Beautiful Soup

"A tremendous boon." — Python@11 Podcast

[ Download | Documentation | Hall of Fame | Source | Discussion group | Zine ]

If Beautiful Soup has saved you a lot of time and money, one way to pay me back is to read *Tell Safety*, a short zine I wrote about what I learned about software development from working on Beautiful Soup. Thanks!

If you have questions, send them to the discussion group. If you find a bug, file it.

Beautiful Soup is a Python library designed for quick turnaround projects like screen-scraping. Three features make it powerful:



# HLT: Practical Tools

## program: a text file (cmudict.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";
}
```

## at the Terminal:

- **perl cmudict.perl triangle**  
*triangle: 3*
- **perl cmudict.perl trump**  
*trump: 1*

# Be careful about multiple installations!

use **cpan** or **cpanm** Lingua::EN::CMUDict (to install modules)

1. /usr/bin/perl
2. /opt/bin/perl
3. /usr/local/bin/perl

```
[ling538-20$ perl cmudict.perl triangle ]  
Can't locate Lingua/EN/CMUDict.pm in @INC (you may need to install the Lingua::E  
N::CMUDict module) (@INC contains: /opt/local/lib/perl5/site_perl/5.26/darwin-th  
read-multi-2level /opt/local/lib/perl5/site_perl/5.26 /opt/local/lib/perl5/vendo  
r_perl/5.26/darwin-thread-multi-2level /opt/local/lib/perl5/vendor_perl/5.26 /op  
t/local/lib/perl5/5.26/darwin-thread-multi-2level /opt/local/lib/perl5/5.26) at  
cmudict.perl line 1.  
BEGIN failed--compilation aborted at cmudict.perl line 1.  
[ling538-20$ which perl ]  
/opt/local/bin/perl  
[ling538-20$ /usr/bin/perl cmudict.perl triangle ]  
triangle: 3  
[ling538-20$ ]
```

# cpan: installing Lingua::EN::CMUDict

```
~$ /usr/bin/cpan      (*may need sudo command prefix)
cpan shell -- CPAN exploration and modules installation (v2.22)
cpan[1]> install Lingua::EN::CMUDict
Fetching with LWP:
http://www.cpan.org/authors/01mailrc.txt.gz
Reading '/Users/sandiway/.cpan/sources/authors/01mailrc.txt.gz'
Fetching with LWP:
http://www.cpan.org/modules/02packages.details.txt.gz
Reading '/Users/sandiway/.cpan/sources/modules/02packages.details.txt.gz'
Database was generated on Tue, 31 Aug 2021 00:17:03 GMT
Fetching with LWP:
http://www.cpan.org/modules/03modlist.data.gz
Reading '/Users/sandiway/.cpan/sources/modules/03modlist.data.gz'
DONE
Writing /Users/sandiway/.cpan/Metadata
Running install for module 'Lingua::EN::CMUDict'
Fetching with LWP:
http://www.cpan.org/authors/id/L/LM/LMETCALF/Lingua-EN-CMUDict-0.06.tar.gz
Fetching with LWP:
HASH(0x7fe38910d6d0)authors/id/L/LM/LMETCALF/CHECKSUMS
Fetching with LWP:
HASH(0x7fe38910d6d0)authors/id/L/LM/LMETCALF/CHECKSUMS.gz
Fetching with LWP:
http://www.cpan.org/authors/id/L/LM/LMETCALF/CHECKSUMS
Checksum for /Users/sandiway/.cpan/sources/authors/id/L/LM/LMETCALF/Lingua-EN-CMUDict-0.06.tar.gz ok
Scanning cache /Users/sandiway/.cpan/build for sizes
Configuring L/LM/LMETCALF/Lingua-EN-CMUDict-0.06.tar.gz with Makefile.PL
Checking if your kit is complete...
Looks good
Generating a Unix-style Makefile
Writing Makefile for Lingua::EN::CMUDict
```

```
Writing MYMETA.yml and MYMETA.json
LMETCALF/Lingua-EN-CMUDict-0.06.tar.gz
/usr/bin/perl Makefile.PL -- OK
Running make for L/LM/LMETCALF/Lingua-EN-CMUDict-0.06.tar.gz
cp lib/Lingua/EN/CMUDict.pm blib/lib/Lingua/EN/CMUDict.pm
Manifying 1 pod document
LMETCALF/Lingua-EN-CMUDict-0.06.tar.gz
/usr/bin/make -- OK
Running make test for LMETCALF/Lingua-EN-CMUDict-0.06.tar.gz
PERL_DL_NONLAZY=1 "/usr/bin/perl" "-MExtUtils::Command::MM" "-MTest::Harness" "-e" "undef
*Test::Harness::Switches; test_harness(0, 'inc', 'blib/lib', 'blib/arch')" t/*.t
t/Lingua-EN-CMUDict.t .. ok
All tests successful.
Files=1, Tests=1, 0 wallclock secs ( 0.01 usr 0.00 sys + 0.07 cusr 0.03 csys = 0.11 CPU)
Result: PASS
LMETCALF/Lingua-EN-CMUDict-0.06.tar.gz
/usr/bin/make test -- OK
Running make install for LMETCALF/Lingua-EN-CMUDict-0.06.tar.gz
Manifying 1 pod document
Installing /Users/sandiway/perl5/lib/perl5/auto/share/dist/Lingua-EN-CMUDict/cmusyldict.db
Installing /Users/sandiway/perl5/lib/perl5/Lingua/EN/CMUDict.pm
Installing /Users/sandiway/perl5/man/man3/Lingua::EN::CMUDict.3pm
Appending installation info to /Users/sandiway/perl5/lib/perl5/darwin-thread-multi-2level/perllocal.pod
LMETCALF/Lingua-EN-CMUDict-0.06.tar.gz
/usr/bin/make install -- OK
```

# HLT: Practical Tools

## Perl

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

### SYNOPSIS [†](#)

```
use Lingua::EN::CMUDict;
my $obj = new Lingua::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)

### `nltk.corpus.reader.cmudict module`

The Carnegie Mellon Pronouncing Dictionary [cmudict.0.6]

<http://ftp.cs.cmu.edu/project/speech/dict/> Copyright 1998 Carnegie Mellon University

File Format: Each line consists of an upcased word, a counter (for alternative pronunciations), and a transcription. Vowels are marked for stress (1=primary, 2=secondary, 0=no stress). E.g.: NATURAL 1 N AE1 CH ERO AHO 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 hut HH AH T AO ought AO T AW cow K AW AY hide HH  
AY D be B IY CH cheese CH IY Z D dee D IY DH thee DH IY EH Ed EH D ER hurt HH ER T  
EY ate EY T F fee F IY G green G R IY N HH he HH IY IH it IH T IY eat IY T JH gee JH IY K  
key K IY L lee L IY M me M IY N knee N IY NG ping P IH NG OW oat OW T OY toy T OY P  
pee P IY R read R IY D S sea S IY SH sha SH IY T tea T IY TH theta TH EY T AH UH hood HH  
UH D UW two T UW V vee V IY W we W IY Y yield Y IY L D Z zee Z IY ZH seizure S IY ZH ER  
`class nltk.corpus.reader.cmudict.CMUDictCorpusReader(root, fileids, encoding='utf8', tagset=None)` [\[source\]](#)

Bases: [nltk.corpus.reader.api.CorpusReader](#)

`dict()` [\[source\]](#)

**Returns:**the cmudict lexicon as a dictionary, whose keys are lowercase words and whose values are lists of pronunciations.

`entries()` [\[source\]](#)

**Returns:**the cmudict lexicon as a list of entries containing (word, transcriptions) tuples.

`raw()` [\[source\]](#)

**Returns:**the cmudict lexicon as a raw string.

`words()` [\[source\]](#)

**Returns:**a list of all words defined in the cmudict lexicon.

`nltk.corpus.reader.cmudict.read_cmudict_block(stream)` [\[source\]](#)

# HLT: Practical Tools

```
$ 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 “**P**ractical **E**xtraction and **R**eporting **L**anguage”
- 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 with the one-liner*)
  - shell scripting: c-shell and others

# Perl

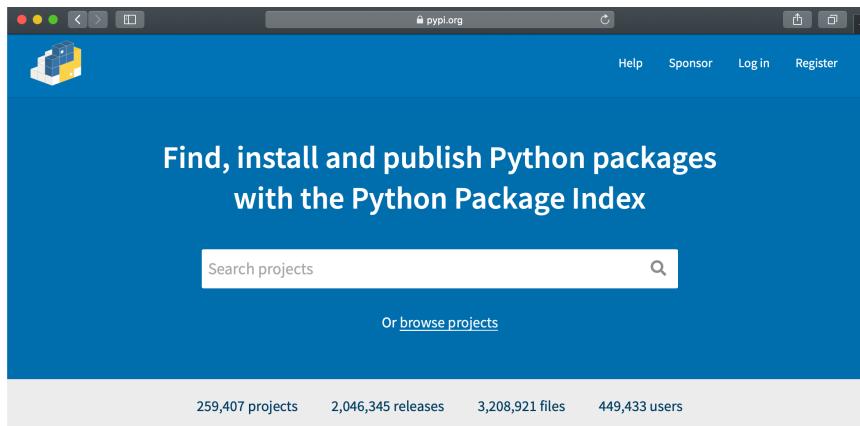
- Interpreted language  
*no compilation phase down to machine code (cf. C, C++), or a virtual machine (cf. Java) so it's slower*
  - lends itself to rapid prototyping, one-off programs, and there's a ...
- Huge collection of Perl modules (.pm):
  - freely available on CPAN (Comprehensive Perl Archive Network)

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

The Comprehensive Perl Archive Network (CPAN) currently has [174,350 Perl modules](#) in 38,734 distributions, written by 13,580 authors, [mirrored](#) on 262 servers.

# Python: pip3

- Another interpreted language, way more packages now ...



The Python Package Index (PyPI) is a repository of software for the Python programming language.  
PyPI helps you find and install software developed and shared by the Python community. [Learn about installing packages](#).  
Package authors use PyPI to distribute their software. [Learn how to package your Python code for PyPI](#).

# Optional Ungraded Exercise

- Figure out how to install cmudict on either Perl or Python and try it out.