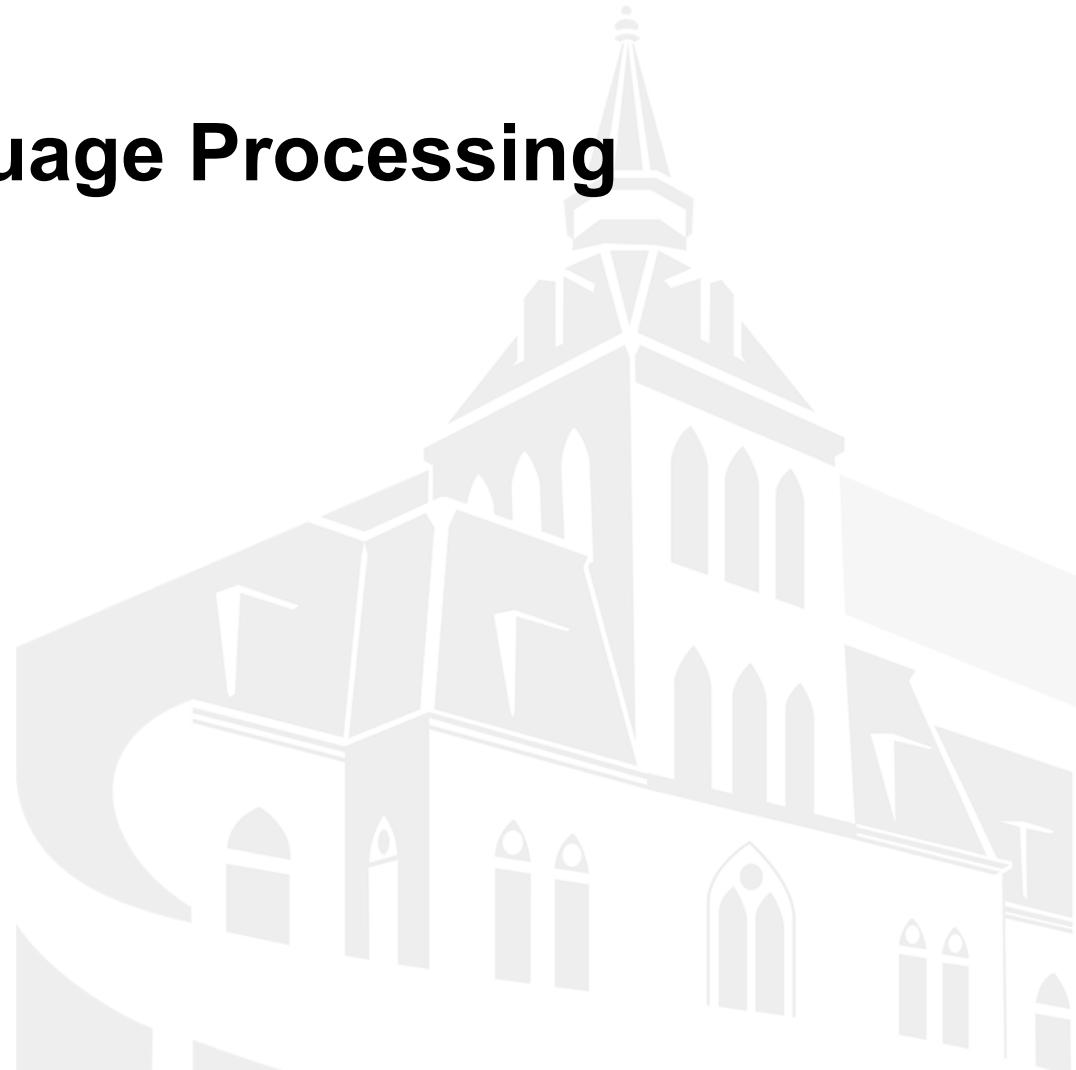




CS 584 Natural Language Processing

Introduction

Department of Computer Science
Yue Ning
yue.ning@stevens.edu





Course Logistics

- ❖ Instructor: Yue Ning
- ❖ Time: Wednesday 6:30 pm - 9:00 pm
- ❖ Meeting classroom: McLean 218B
- ❖ Webpage: <https://yue-ning.github.io/cs584-f19.html>
- ❖ Office hours: Monday 2-4pm
- ❖ Slides uploaded before each lecture



Prerequisites

- ❖ Python Programming
- ❖ Linear algebra
- ❖ Probability and optimization

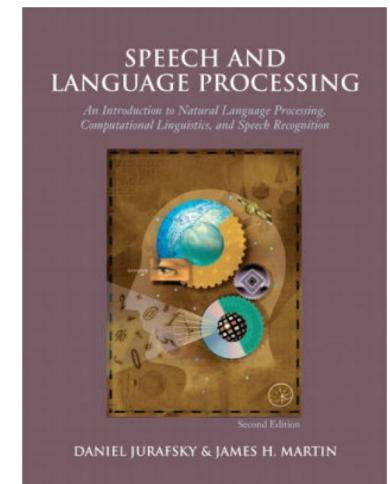
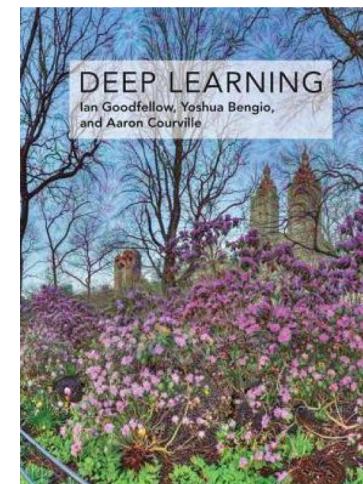
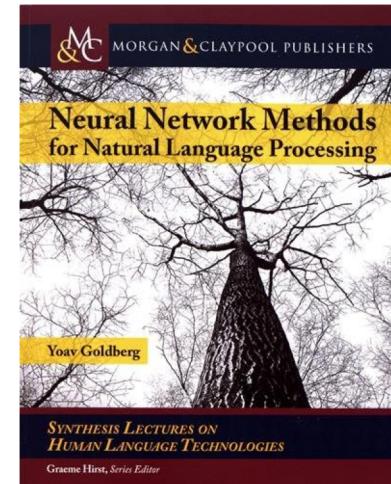


What do we learn from this course?

1. Understanding of basic techniques of processing human languages and the difficulties in understanding languages.
2. Understanding of effective modern methods for NLP
 - a. recurrent neural networks, attention, etc
3. Understanding of and ability to build systems for some major tasks in NLP
 - a. word meaning, dependency parsing, document classification etc.

Recommended Textbooks

- Yoav Goldberg. **Neural Network Methods for Natural Language Processing (NNLP).**
- Ian Goodfellow and Yoshua Bengio and Aaron Courville, 2016. **Deep Learning (DL)**, MIT Press. We will cover topics including basic neural networks, back propagation, and CNN.
- Dan Jurafsky and James H. Martin. 2018. **Speech and Language Processing (3rd ed. draft) (SLP).**





Course work and grading policy

- Assignments (50%): 5 in total; 10% each.
 - All released in the first 2 weeks;
 - Deadlines: Sep. 19, Oct. 1, Oct. 23, Nov. 6, Nov. 23
- Final project (25%):
 - Project proposal 5%: Oct. 27
 - Presentation 10%: Dec. 4 and Dec. 11
 - Final report: 10%: Dec. 11
- Midterm exam (15%): Oct. 16
 - Scheduled in the middle of the semester
- Participation (5%):
 - Lecture attendance (random two lectures); Canvas discussion
- Quizzes (5%):
 - Pop quizzes; only top 2 quizzes will be counted.



Late Submission Policy

- 10% penalty for late submission within **24 hours**.
- 40% penalty for late submissions within **24-48 hours**.
- After 48 hours, you get **NO** points on the assignment.



NO cheating/plagiarism
Stevens Honor System



About this course

Lectures won't always have all the details

- It's up to you to search online/ do some reading to find out more
- This is an active research field. Sometimes there is no clear-cut answer
- Instructors/TAs are happy to discuss with you, but you need to think for yourself

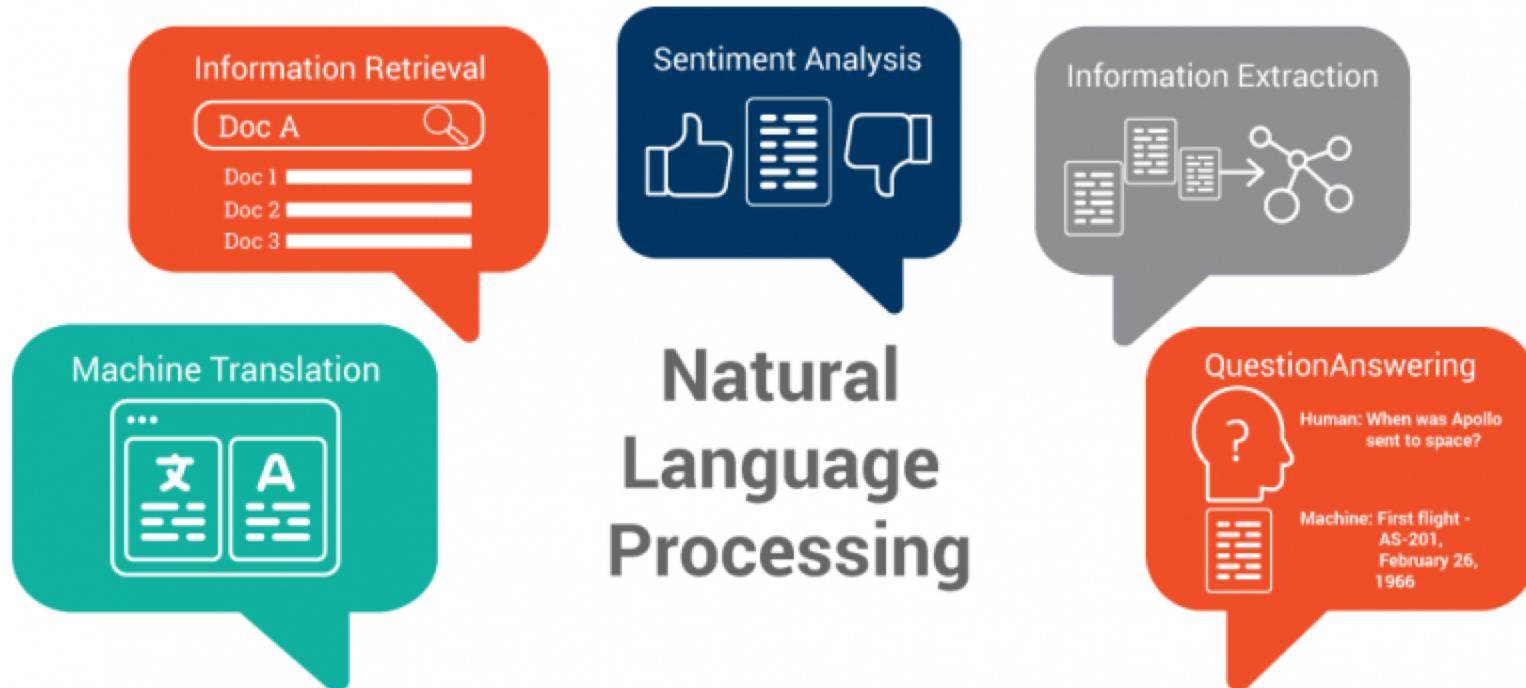


This lecture

- ❑ Course Information
- ❑ Course Overview
 - ❑ What is NLP? Why it is important?
 - ❑ What will you learn from this course?
- ❑ What are the challenges?
- ❑ Key NLP components

What is NLP?

Wikipedia: Natural language processing (NLP) is a subfield of **computer science**, **information engineering**, and **artificial intelligence** concerned with the interactions between computers and human (natural) languages, in particular how to program computers to process and analyze large amounts of natural language data.



Go beyond the keyword matching



- Identify the structure and meaning of words, sentences, texts and conversations
- Deep understanding of broad language
- NLP is all around us

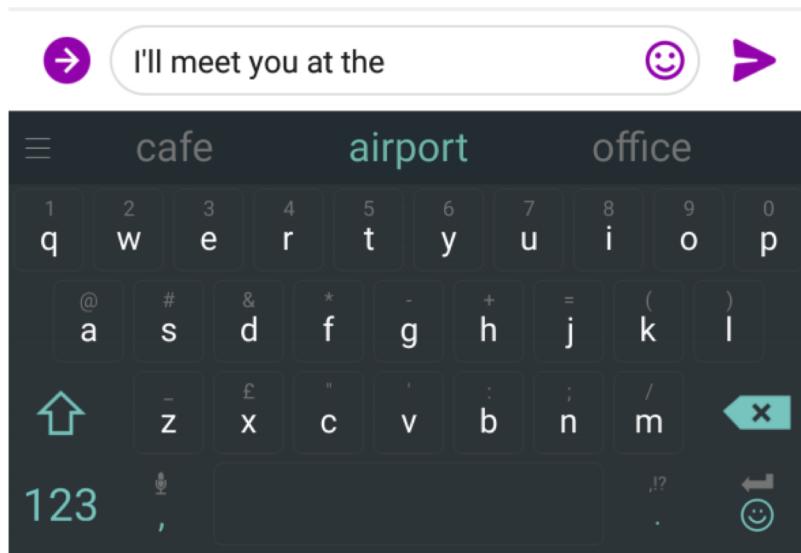
Spell Check

ection and surveillience lag, accurate
are often delayed mak
action has been fo
.s~\cite{kermack:2
on with statistica
; 3) time series :
lowell2008seasonal

surveillance maki
surveillance's
salience s as
succulence ed pri
silence s Aut
surliness soci
surveying's



Predictive Text



Google

how to maintain a

- how to maintain a **beard**
- how to maintain a **pool**
- how to maintain a **healthy diet**
- how to maintain a **hot tub**
- how to maintain a **healthy relationship**
- how to maintain a **car**
- how to maintain a **long distance relationship**
- how to maintain a **saltwater pool**
- how to maintain a **septic tank**
- how to maintain a **healthy lifestyle**

Report inappropriate predictions



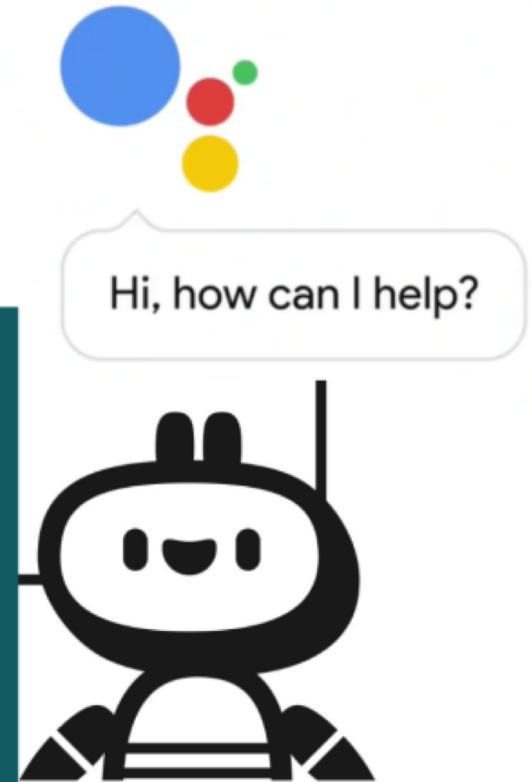
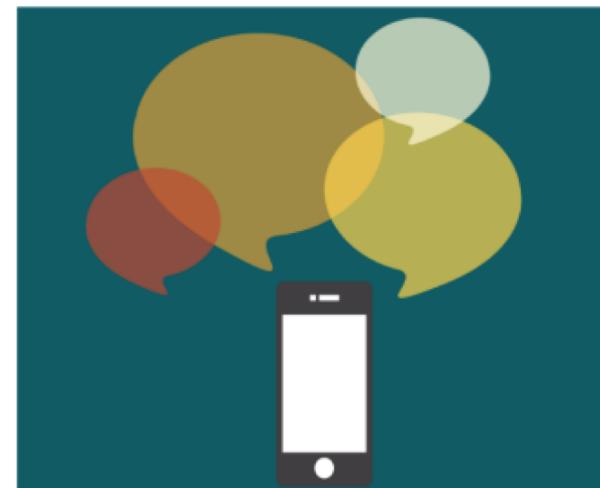
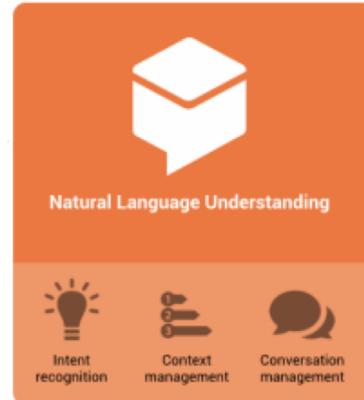
Machine Translation

Google Translate

The screenshot shows the Google Translate interface. At the top, there are tabs for "Text" and "Documents", with "Text" selected. Below that, language pairs are listed: ENGLISH - DETECTED, ENGLISH, SPANISH, FRENCH, CHINESE (SIMPLIFIED), ARABIC, and SPANISH. The main area contains two text boxes. The left box, under English, contains the following text: "Natural language processing (NLP) is a subfield of computer science, information engineering, and artificial intelligence concerned with the interactions between computers and human (natural) languages, in particular how to program computers to process and analyze large amounts of natural language data." The right box, under Chinese (Simplified), contains the translated text: "自然语言处理 (NLP) 是计算机科学, 信息工程和人工智能的子领域, 涉及计算机和人类 (自然) 语言之间的交互, 特别是如何对计算机进行编程以处理和分析大量自然语言数据。". There are also audio icons and edit/copy/share buttons at the bottom.

This screenshot shows another view of the Google Translate interface. It has language pairs: English (detected), Arabic, and English. The left box (Arabic) contains the text: "معالجة اللغة الطبيعية (NLP) هو مجال فرعى من علوم الكمبيوتر، هندسة المعلومات، والذكاء الاصطناعي المعنى بالتفاعلات بين أجهزة الكمبيوتر واللغات البشرية (الطبيعية)، ولا سيما كيفية برمجة أجهزة الكمبيوتر لمعالجة وتحليل كميات كبيرة من بيانات اللغة الطبيعية.". The right box (English) contains the translated text: "Natural language processing (NLP) is a subfield of computer science, information engineering, and artificial intelligence concerned with the interactions between computers and human (natural) languages, in particular how to program computers to process and analyze large amounts of natural language data.". There are also audio icons and edit/copy/share buttons at the bottom.

Dialog Systems



Sentiment/Opinion Analysis



POSITIVE



JOY



SURPRISE



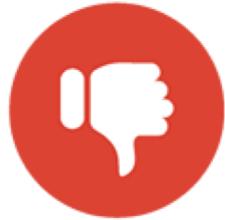
NEUTRAL



ANGER



DISGUST



NEGATIVE



FEAR

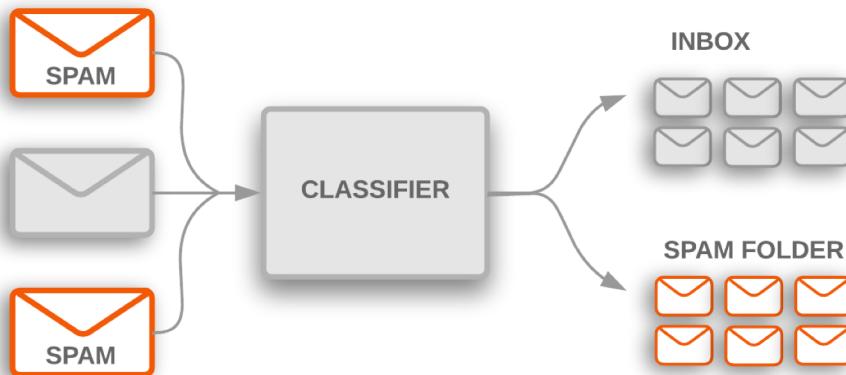


SADNESS

SENTIMENT

EMOTION

Text Classification



Primary Social Promotions

<input type="checkbox"/>	<input type="star"/>	cbc-inform	Call for Paper: Brain Informatics 2019 (BI'19) - ***** CALL FOR...	May 19
<input type="checkbox"/>	<input type="star"/>	cbc-inform	IEEE/WIC/ACM WI 2019 - [Deadline Extended] Call for Papers - *****...	May 19
<input type="checkbox"/>	<input type="star"/>	uvmxwu	IEEE ICDM 2019: Final Call for Papers (by June 5) - ICDM 2019: The 19th IEEE International Conference on ...	May 14
<input checked="" type="checkbox"/>	<input type="star"/>	panagiotis	IEEE ICDM 2019 Workshop Proposals by 5/21 - Dear Yue, (Sorry 2020 in our last message shoul...	   



Wednesday, June 19, 2019

ENGLISH ESPAÑOL 中文

World U.S. Politics N.Y. Business Opinion Tech Science Health Sports Arts Books Style Food Travel Magazine T Magazine

Survey Analysis

Project 1 - IBM SPSS Text Analytics for Surveys

File Edit View Categories Tools Help

Q3_Thinking_of_your_last_LEISURE/VACATION_car_rental_please_provide_any_comments_or_suggestions_regarding

Category	#Descri...	#Respo...
All Records	-	200
Uncategorized	-	200
No concepts extracted	-	1

Unused Extractions All Extractions

Extract Concept Pattern

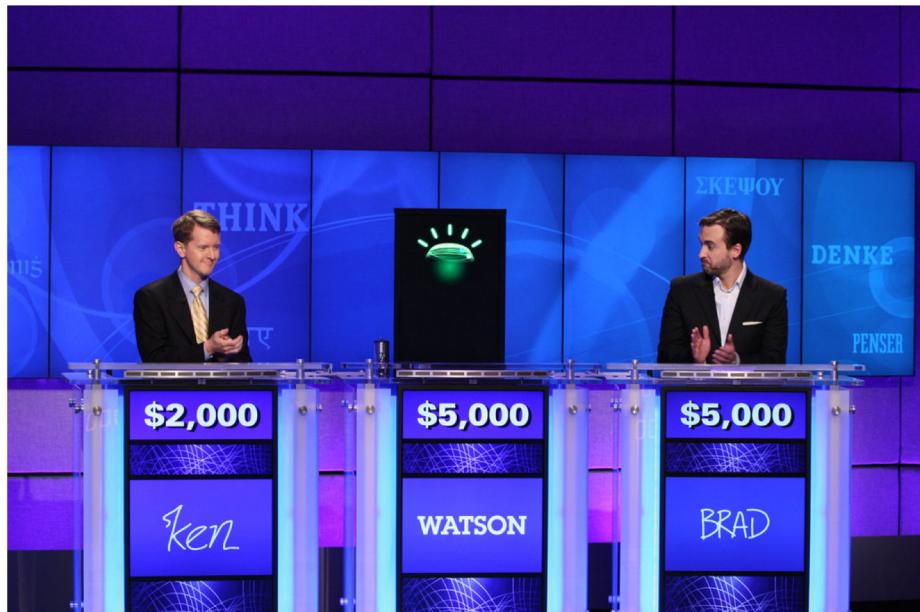
- car + . (33)
- [customer service + <Positive> (21)]
 - good + . (18)
 - no problem + . (14)
 - customer service + . (11)
- [service + <Positive> (11)]
 - friendly + . (10)
 - car + <Positive> (10)
 - company + . (9)
- [experience + <Positive> (8)]
 - rental car + . (7)

0 (0) Categories

	ID	Response	Categories
1	3	I would have to say that Enterprise car rental has the best customer service . They are very friendly and helpful.	
2	15	Customer service was great , but I wish they had searched the car when I returned it. I left my Palm Pilot in the backseat for 2 days before I noticed it was missing! Fortunately the car was still in the lot when I called and they were able to ship it to me.	
3	41	Customer service was fine . We didn't care for the off site location. Would not use that car rental company at that airport again.	
4	45	customer service was OK	
5	49	Customer service was just as I recall . No problems.	
6	61	Customer service was outstanding . Got a free upgrade without even asking. Everyone was very courteous and friendly .	
7	72	Anywhere you go you will find good customer service and bad customer service . Since I have rented so many cars , there have been too many experiences to recall.	
8	73	The car rental company that I went with had very good customer service . They were out of a certain car I reserved and gave me a upgrade and apologized.	
9	76	No suggestions . Customer Service was good .	
10	77	customer service was great until we returned the car . I experienced an issue with damage to the car , and felt that things were left unanswered and still waiting for a response from them.	



Question Answering

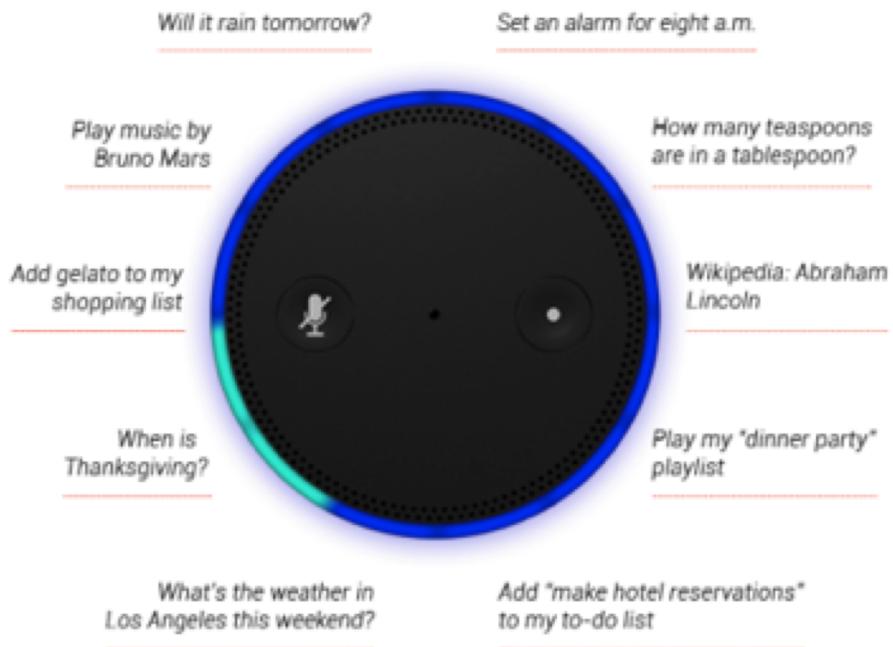


[Watson computer wins at Jeopardy](#)





Natural Language Instruction



Digital personal assistant

More on natural language instruction



Cortana.



Siri



amazon echo



Google now



Facebook M

Semantic parsing - understand tasks

Entity linking - "my husband's mom" = "Kelly" in the phone book



Information Extraction

- ❖ Unstructured text to database entries

New York Times Co. named Russell T. Lewis, 45, president and general manager of its flagship New York Times newspaper, responsible for all business-side activities. He was executive vice president and deputy general manager. He succeeds Lance R. Primis, who in September was named president and chief operating officer of the parent.

Person	Company	Post	State
Russell T. Lewis	New York Times newspaper	president and general manager	start
Russell T. Lewis	New York Times newspaper	executive vice president	end
Lance R. Primis	New York Times Co.	president and CEO	start



Language Comprehension

Beyonce Giselle Knowles-Carter (born September 4, 1981) is an American singer, songwriter, record producer and actress. Born and raised in **Houston, Texas**, she performed in various **singing and dancing** competitions as a child, and rose to fame in the late 1990s as lead singer of R&B girl-group Destiny's Child. Managed by her father, Mathew Knowles, the group became one of the world's best-selling girl groups of all time. Their hiatus saw the release of Beyoncé's debut album, *Dangerously in Love* (**2003**), which established her as a solo artist worldwide, earned five Grammy Awards and featured the Billboard Hot 100 number-one singles "Crazy in Love" and "Baby Boy"

- ❖ Q: In what city and state did Beyoncé grow up?
❖ A: **Houston, Texas**

- ❖ Q: What areas did Beyoncé compete in when she was growing up?
❖ A: **Singing and dancing**

- ❖ Q: When did Beyoncé release *Dangerously in love*?
❖ A: **2003**

A (wikipedia) passage from the SQuAD 2.0 dataset with 3 sample questions and the labeled answer spans. [\[Rajpurkar et al. 2018\]](#)



Learning Map

- Key components for understanding text
- NLP systems/applications
 - Current techniques and limitations
- Build realistic NLP tools



What's not covered by this course

- ❖ Speech recognition - no signal processing
- ❖ Natural language generation
- ❖ Text mining/Information retrieval
- ❖ Programming details



This lecture

- ❑ Course Information
- ❑ Course Overview
 - ❑ What is NLP? Why it is important?
 - ❑ What will you learn from this course?
- ❑ **What are the challenges?**
- ❑ Key NLP components

Challenges - Ambiguity

- ❖ Word sense ambiguity: semantic ambiguity, when a word has more than one meanings.

- *John killed the wolf*
- *Bill killed the project*
- *Mary killed Jane* (at tennis or murdered her?)



credit: A. Zwicky



Challenges - Ambiguity

- ❖ Word sense/meaning ambiguity



Challenges - Ambiguity

❖ PP attachment ambiguity

San Jose cops kill man with knife

Text Paper Close Translate Listen

San Jose cops kill man with knife

Ex-college football player, 23, shot 9 times allegedly charged police at fiancee's home

By Hamed Aleaziz and Vivian Ho

A man fatally shot by San Jose police officers while allegedly charging at them with a knife was a 23-year-old former football player at De Anza College in Cupertino who was distraught and depressed, his family said

Thursday.

Police officials said two officers opened fire Wednesday afternoon on Phillip Watkins outside his fiancee's home because they feared for their lives. The officers had been drawn to the home, officials said, by a 911 call reporting an armed home invasion

that, it turned out, had been made by Watkins himself.

But the mother of Watkins' fiancee, who also lives in the home on the 1300 block of Sherman Street, said she witnessed the shooting and described it as excessive. Faye Buchanan said the confrontation happened shortly after she called a suicide intervention hotline in hopes of getting Watkins medical help.

Watkins' 911 call came in at 5:01 p.m., said Sgt. Heather Randol, a San Jose police spokeswoman. "The caller stated there was a male breaking into his home armed with a knife," Randol said. "The caller also stated he was locked in an upstairs bedroom with his children and requested help from police."

She said Watkins was on the sidewalk in front of the home when two officers got there. He was holding a knife with a 4-inch blade and ran toward the officers in a threatening manner, Randol said.

"Both officers ordered the suspect to stop and drop the knife," Randol said. "The suspect continued to charge the officers with the knife in his hand. Both officers, fearing for their safety and defense of their life, fired at the suspect."

On the police radio, one officer said, "We have a male with a knife. He's walking toward us."

"Shots fired! Shots fired!" an officer said moments later.

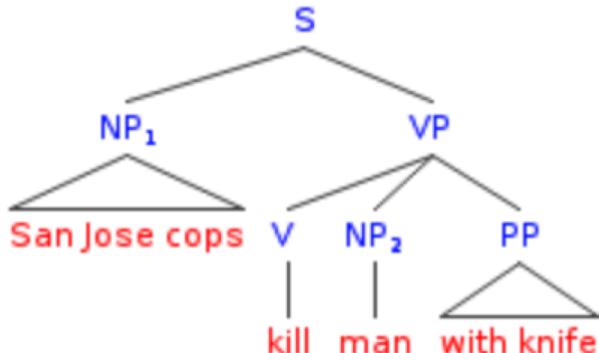
A short time later, an officer reported, "Male is down. Knife's still in hand."

Buchanan said she had been prompted to call the

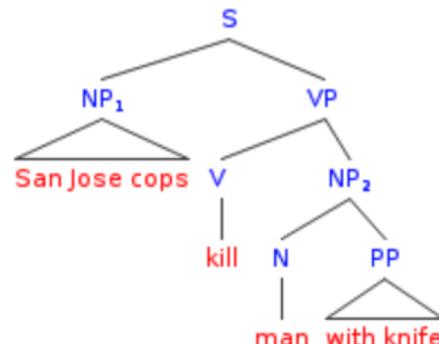
Shoot continues on D8

Back Continue

Is the prepositional phrase attached high, modifying the verb "kill"?



Or is it attached low, to the object NP "man"?





Challenges - Ambiguity

Ambiguous headlines

- Include your children when baking cookies
- Local High School Dropouts Cut in Half
- Hospitals are Sued by 7 Foot Doctors
- Iraqi Head Seeks Arms
- Safety Experts Say School Bus Passengers Should Be Belted
- Teacher Strikes Idle Kids

Challenges - Ambiguity

Pronoun reference ambiguity



Dr. Macklin often brings his dog Champion to visit with the patients. **He** just loves to give big, wet, sloppy kisses!



Challenges - Language is dynamic

Language evolves and changes, e.g. cyber lingo

LOL	Laugh out loud
G2G	Got to go
BFN	Bye for now
B4N	Bye for now
Idk	I don't know
FWIW	For what it's worth
LUWAMH	Love you with all my heart



Challenges - Scale

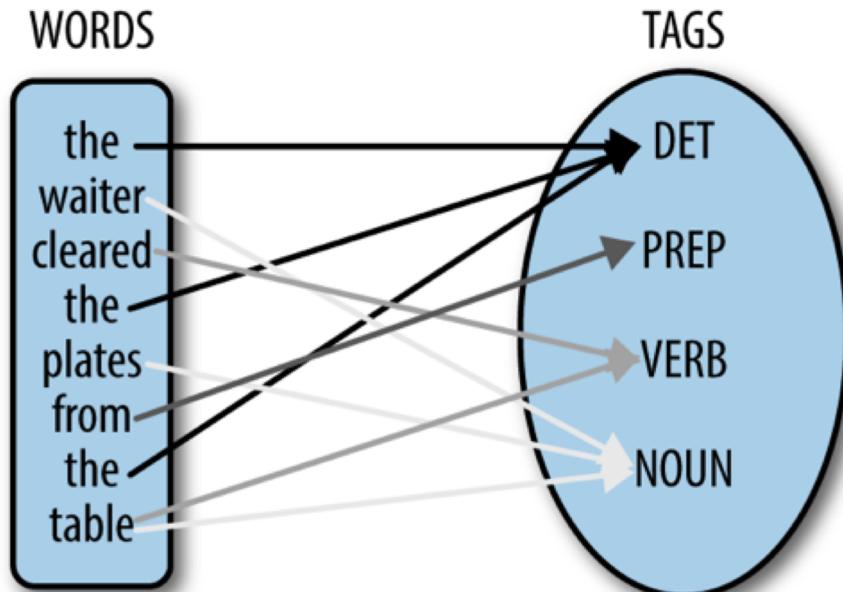
Bible (King James version)	~700K
Penn Tree bank	~1M from wall street journal
Newswire collection	~500M+
Wikipedia	~3 billion words
Web	several billions of words



This lecture

- ❑ Course Information
- ❑ Course Overview
 - ❑ What is NLP? Why it is important?
 - ❑ What will you learn from this course?
- ❑ What are the challenges?
- ❑ **Key NLP components**

Part of speech tagging



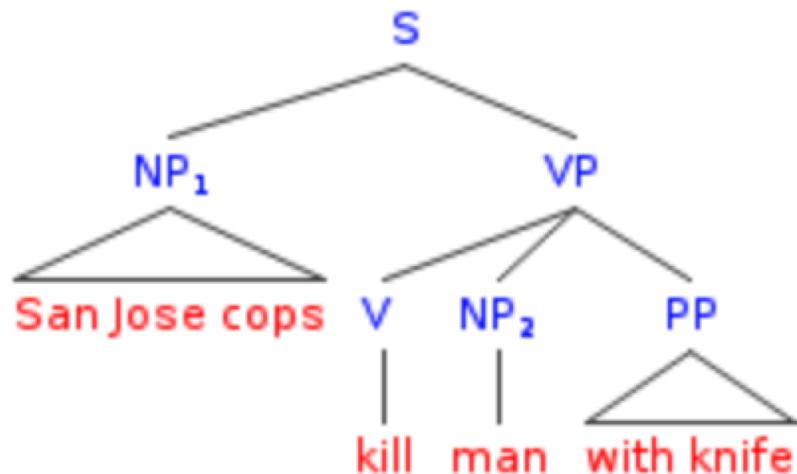
Common closed-classes:

- prepositions: on, under, over, near, by, at, from, to, with
- particles: up, down, on, off, in, out, at, by
- determiners: a, an, the
- conjunctions: and, but, or, as, if, when
- pronouns: she, who, I, others
- auxiliary verbs: can, may, should, are
- numerals: one, two, three, first, second, third

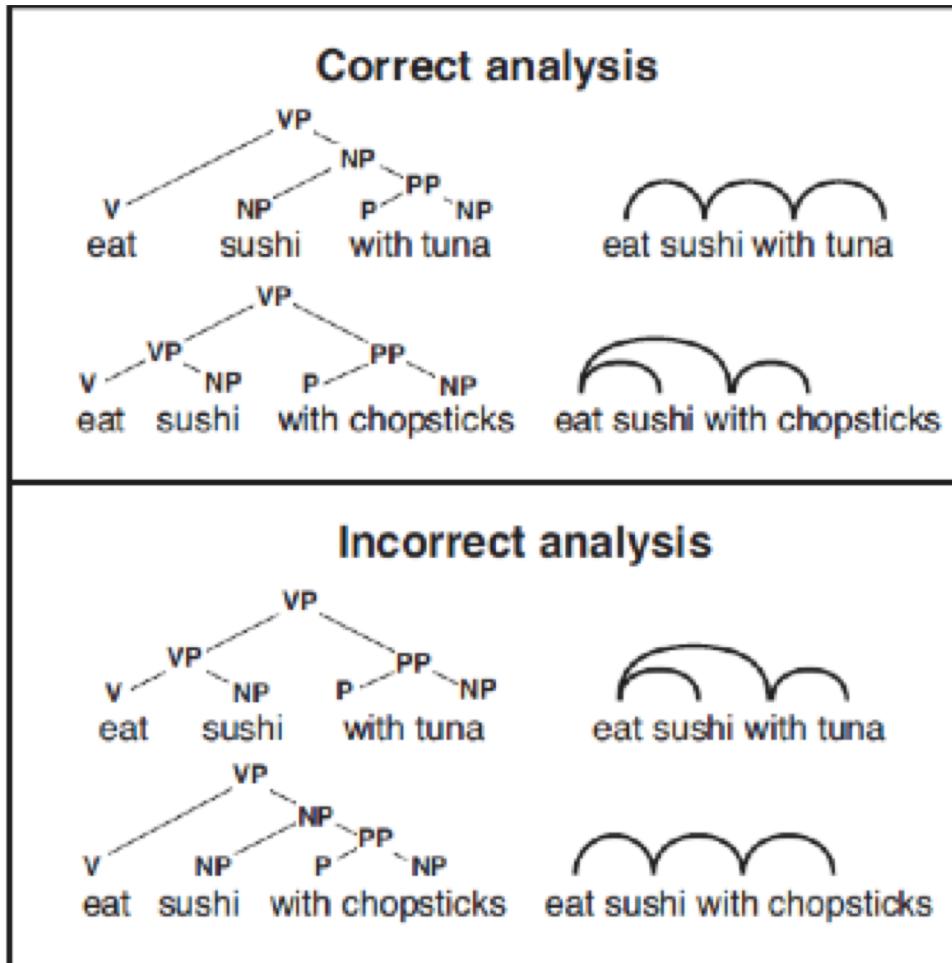
Common open-classes:

- adjectives
- adverbs
- nouns
- verbs (except auxiliary verbs)
- interjections

Syntactic (Constituency) tagging

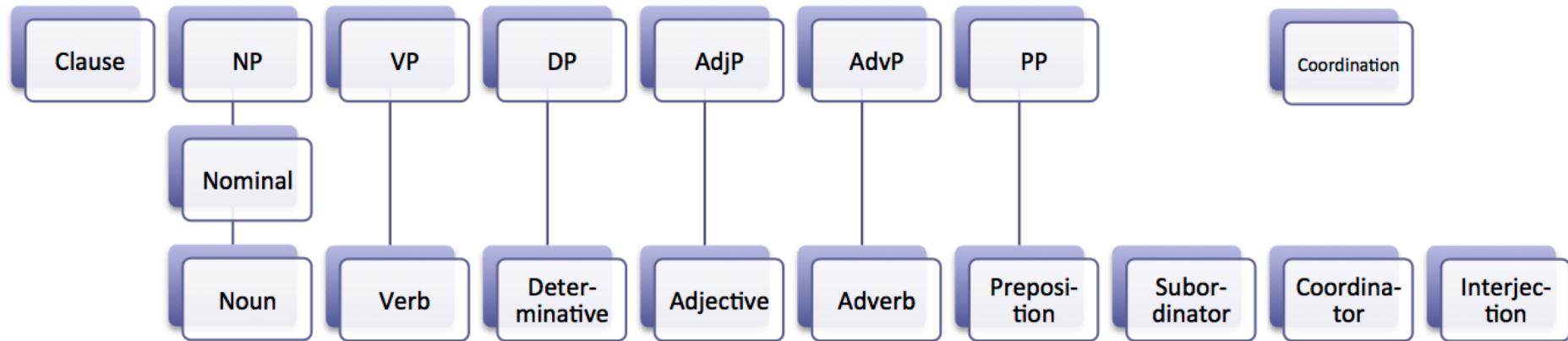


Syntactic structure -> meaning

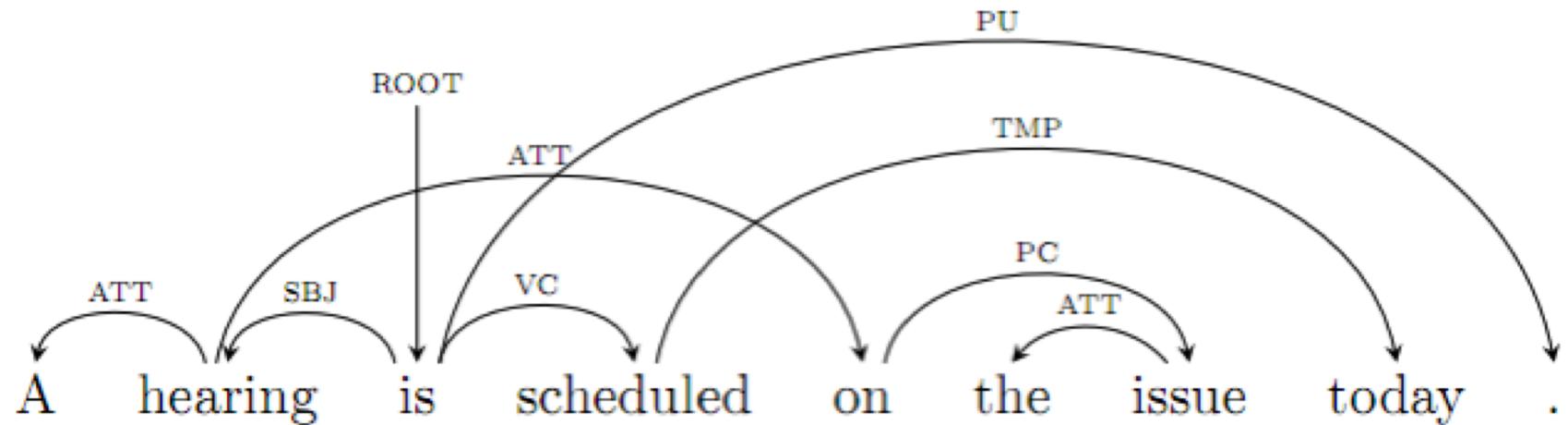


Syntactic categories

Lexical and phrasal categories together are called syntactic categories:

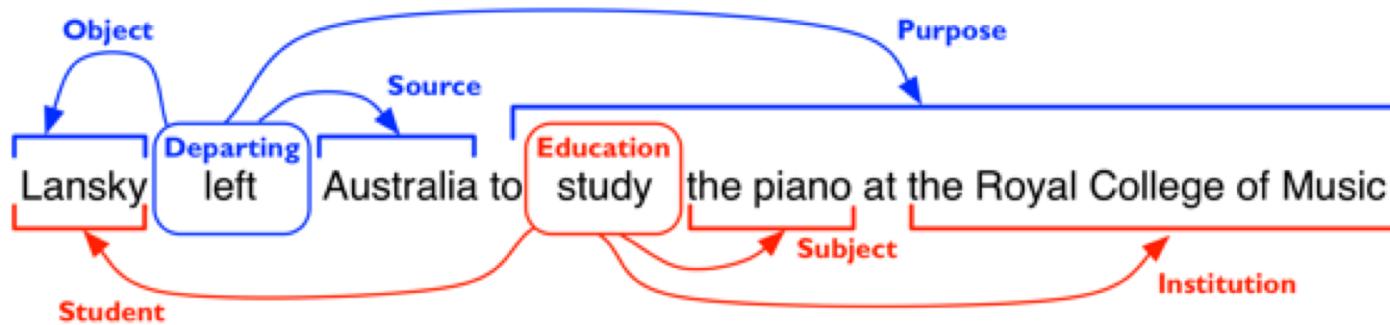


Dependency Parsing



Semantic analysis

- ❖ word sense disambiguation
- ❖ semantic role labeling





Co-reference resolution

Beyonce Giselle Knowles-Carter (born September 4, 1981) is an American singer, songwriter, record producer and actress. Born and raised in Houston, Texas, she performed in various singing and dancing competitions as a child, and rose to fame in the late 1990s as lead singer of R&B girl-group Destiny's Child. Managed by her father, Mathew Knowles, the group became one of the world's best-selling girl groups of all time. Their hiatus saw the release of Beyoncé's debut album, Dangerously in Love (2003), which established her as a solo artist worldwide, earned five Grammy Awards and featured the Billboard Hot 100 number-one singles "Crazy in Love" and "Baby Boy".

Q: Is Beyoncé's father Mathew Knowles?



Quiz 0



Linear Algebra Review

- $s = \mathbf{a}^T \mathbf{b}$ where \mathbf{a} and \mathbf{b} are both column vectors with dimension of d .
 - what is $\frac{\partial s}{\partial \mathbf{a}}$ and what is the dimension of $\frac{\partial s}{\partial \mathbf{a}}$?
- $\mathbf{y} = \mathbf{A}\mathbf{x}$ where \mathbf{y} is a $m \times 1$ column vector, \mathbf{A} is a $m \times n$ matrix, \mathbf{x} is a $n \times 1$ column vector, \mathbf{A} does not depend on \mathbf{x} .
 - what is $\frac{\partial \mathbf{y}}{\partial \mathbf{x}}$?
- $s = \mathbf{y}^T \mathbf{A} \mathbf{x}$ where \mathbf{y} is a $m \times 1$ column vector, \mathbf{A} is a $m \times n$ matrix, \mathbf{x} is a $n \times 1$ column vector, \mathbf{A} and \mathbf{y} do not depend on \mathbf{x} .
 - what is $\frac{\partial s}{\partial \mathbf{x}}$?

Reference: https://en.wikipedia.org/wiki/Matrix_calculus



Python Review

Why Python?

- + Python is a widely used, general purpose programming language.
- + Easy to start working with.
- + Scientific computation functionality similar to Matlab and Octave.
- + Used by major deep learning frameworks such as PyTorch and TensorFlow.



Language basics

Python is a **strongly-typed** and **dynamically-typed** language.

- Strongly-typed: Interpreter always “respects” the types of each variable.[1]
- Dynamically-typed: “A variable is simply a value bound to a name.” [1]
- Execution: Python is first interpreted into bytecode (.pyc) and then compiled by a VM implementation into machine instructions. (Most commonly using C.)
 - Python is “slower”, but it can run highly optimized C/C++ subroutines which make scientific computing (e.g. matrix multiplication) really fast.

<https://wiki.python.org/moin/Why%20is%20Python%20a%20dynamic%20language%20and%20also%20a%20strongly%20typed%20language>

Language basics

- Strongly-typed: `1+'1'` -> Error
- Dynamically-typed: `foo = [1,2,3]` and later, `foo = "hello"`
- Execution: `np.dot(x, w) + b` -> fast!



Python Review

Language Basics

Anyone wants to guess what this function does?

```
def QuickSort(arr):  
    if len(arr) <= 1:  
        return arr  
  
    pivot = arr[len(arr) // 2]  
  
    left = [x for x in arr if x < pivot]  
    middle = [x for x in arr if x == pivot]  
    right = [x for x in arr if x > pivot]  
  
    return QuickSort(left) + middle + QuickSort(right)
```

Example code from Andrej Karpathy's tutorial: <http://cs231n.github.io/python-numpy-tutorial/>



Python Review - Common operations

```
x = 10                                # Declaring two integer  
variables  
  
y = 3                                  # Comments starts  
with the hash symbol  
  
  
x + y >> 13                            # addition  
  
x - y >> 7                            # subtraction  
  
x ** y >> 1000                          # exponentiation  
  
x/y >> 3                               # dividing two  
integers  
  
x / float(y) >> 3.333..                 # type casting for float  
division  
  
  
str(x) +"+" + str(y) >>"10+3" # casting and string concatenation
```



Build-in values

```
True, False           # usual boolean values  
None                # represents the null  
  
x = None             # variables can be None  
array=[1,2,None]     # lists can contain None  
  
def func():  
    return None        #functions can return None  
  
if [1,2]![3,4]:       #can check for equality  
    print 'error!'
```



List

```
names = ['zach', 'emily']
names[0] == 'zach'
names.append('alex')
len(names) == 3
print names >> ['zach', 'emily', 'alex']
names.extend(['aba', 'kevin'])
print names >> ['zach', 'emily', 'alex', 'aba', 'kevin']
names = [] # creates an empty list
names = list() # also creates an empty list
stuff=[1,['hello','bye'], 1.3, None] # can mix types
```



List slicing

List elements can be accessed in convenient ways

basic format: mylist[start_index:end_index]

```
numbers = [0,1,2,3,4,5,6]
```

```
numbers[0:3] == numbers[:3] >> [0,1,2]
```

```
numbers[5:] == numbers[5:7] >>[5,6]
```

```
numbers[-1] == 6 #negative index wraps around
```

```
number[-3:] == [4,5,6]
```

```
numbers[3:-2] == [3,4] # can mix and match
```



Tuple

Tuple are immutable arrays

```
names = ('zach', 'emily')  
names[0] == 'zach'  
len(names) == 2  
print names >> ('zach', 'emily')  
names[0] = 'richard'  
>>TypeError: 'tuple' object does not support item assignment  
  
empty = tuple() # empty tuple  
single = (10,) # single-element tuple, comma matters!
```



Dictionary

Dictionaries are hash maps

```
phonebook = dict()                                     #empty dictionary

phonebook = { 'zach' : '12-68' }                      # dictionary with one item

phonebook['emily'] = '35-65'                           #add another item

print('zach' in phonebook)                            >> True

print('john' in phonebook)                            >>False

del phonebook['zach']                                #delete an item

print (phonebook)                                    >> { 'emily' : '36-65' }

for name, number in phonebook.items():
    print name, number                               >> emily 35-66
```



Classes

```
class Animal(object):

    def __init__(self, species, age): # Constructor `a = Animal('bird', 10)`

        self.species = species          # Refer to instance with
`self`                                         # All instance variables are

        self.age = age                  public

def isPerson(self):

    return self.species == "Homo Sapiens"

def ageOneYear(self):

    self.age += 1

class Dog(Animal):                                # Inherits Animal's
methods

    def ageOneYear(self):                         # Override for dog years

        self.age += 7
```



Numpy - ndarray operations

```
import numpy as np  
  
z=np.array([[6,7],[8,9]])
```

```
print(np.max(z, axis=1))          >> [7,9]  
print(np.max(z, axis=1, keepdims=True))      >> [[7]
```

```
[9]]
```

```
np.dot(x, w) or x.dot(w)
```

Note: shapes (N,) !=(1,N)

```
print (np.array([1,2,3]).T)          >> [1  2  3]  
np.sum(np.array([1,2,3]), axis = 1) >> Error!
```



Indexing

```
x = np.random.random((3, 4)) # Random (3,4) matrix  
x[:] # Selects everything in x  
x[np.array([0, 2]), :] # Selects the 0th and 2nd rows  
x[1, 1:3] # Selects 1st row as 1-D vector  
# and 1st through 2nd elements  
x[x > 0.5] # Boolean indexing
```



Broadcasting

```
x = np.random.random((3, 4)) # Random (3, 4) matrix
y = np.random.random((3, 1)) # Random (3, 1) matrix
z = np.random.random((1, 4)) # Random (3,) vector
x + y                      # Adds y to each column of x
x * z                      # Multiplies z element-wise with
each row of x
print((y + y.T).shape)      # Can give unexpected results!
```



Efficiency

For loops will dramatically slow down your code (~10-100x)

```
for i in range(x.shape[0]):  
    for j in range(x.shape[1]):  
        x[i,j] **= 2  
  
x **=2  
  
for i in range(100, 1000):  
    for j in range(x.shape[1]):  
        x[i, j] += 5  
  
x[np.arange(100,1000), :] += 5
```



List comprehension

- Similar to map() from functional programming languages.
- Can improve readability & make the code succinct.
- Format: [func(x) for x in some_list]
- Following are equivalent:

```
squares = []

for i in range(10):

    squares.append(i**2)

squares = [i**2 for i in range(10)]
```

- Can be conditional:

```
odds = [i**2 for i in range(10) if i%2 == 1]
```

Convenient syntax

- Multiple assignment / unpacking iterables

```
x, y, z = ['Tensorflow', 'PyTorch', 'Chainer']
```

```
age, name, pets = 20, 'Joy', ['cat']
```

- Returning multiple items from a function

```
def some_func():
```

```
    return 10, 1
```

```
ten, one = some_func()
```

- Joining list of strings with a delimiter

```
" ".join([1, 2, 3]) == '1, 2, 3'
```

- String literals with both single and double quotes

```
message = 'I like "single" quotes.'
```

```
reply = "I prefer 'double' quotes."
```



Debugging tips

- Python has an interactive shell where you can execute arbitrary code
 - Great replacement for TI-84 (no integer overflow!)
 - Confused by syntax? Just try it in the shell!

```
$ python
```

```
Python 2.7.10 (default, Jul 15 2017, 17:16:57)
```

```
>>> 2 ** 5 / 2
```

```
16
```

```
>>> 2 ** (5 / 2)
```

```
4
```

- Can import any module (even custom ones in the current directory)
- Try small test cases in the shell



Debugging tips

Unsure of what you can do with an object? **Use type() and dir()!!**

```
>>> class Duck(object):
...     def quack(self): pass
...
>>> bird = Duck()
>>> type(bird)
<class '__main__.Duck'>
>>> dir(bird)
['__class__', '__delattr__', '__dict__', '__doc__', '__format__',
 '__getattribute__', '__hash__', '__init__', '__module__',
 '__new__', '__reduce__', '__reduce_ex__', '__repr__',
 '__setattr__', '__sizeof__',
 '__str__', '__subclasshook__', '__weakref__', 'quack'] >>>
```



Numpy debugging

- Print shapes to see if they match what you expect: print `x.shape`
- Print shapes!! Make sure broadcasting is done properly.
- Print types and values.
- Checking if two float arrays are approximately equal (element-wise)
 - `np.allclose(x, y) # Can also specify tolerance`
- Checking if an array is close to zero (e.g. gradient)
 - `np.allclose(x, 0) # Broadcasting`
- Selecting all elements less than 0 from an array
 - `x[x < 0] # Returns 1-dim array`

Environment management

- Problem:
 - Python 3 is not backward-compatible with Python 2
 - Countless Python packages and their dependencies
 - Different projects require different packages
 - Even worse, different versions of the same package!
- Solution:
 - Keep multiple Python environments that are isolated from each other
 - Each environment...
 - can use different Python versions
 - keeps its own set of packages
 - can be easily replicated (e.g. on a VM, friend's laptop, etc.)



Anaconda

- Anaconda is a popular Python environment/package manager
 - Install from <https://www.anaconda.com/download/>
 - Supports Windows, Linux, macOS
 - Basic workflow

```
$ source activate <environment_name>
```

```
<... do stuff ...>
```

```
$ deactivate
```

- Other environments won't be affected by anything you do
- Allows you to run a different version of Python for each environment



Virtualenv

- Virtualenv is another popular Python environment manager
 - Only specifies different packages per environment
 - Doesn't help run different Python version
 - Installation from
 - <https://virtualenv.pypa.io/en/stable/installation/>
 - Basic workflow

```
$ mkdir <environment_directory>
$ virtualenv <environment_directory>
$ source <env_dir>/bin/activate
$ pip install <package>
```



Other Python Resources

1. Official Python 2 documentation: <https://docs.python.org/2/>
2. Official Python 2 tutorial: <https://docs.python.org/2.7/tutorial/index.html>
3. Numpy Quickstart: <https://docs.scipy.org/doc/numpy-dev/user/quickstart.html>
4. Python Tutorial from CS231N at Stanford:
<http://cs231n.github.io/python-numpy-tutorial/>
5. Stanford Python course (CS41): <http://stanfordpython.com/>



Acknowledgements

Slides adapted from Dr. Christopher Manning's Natural Language Processing with Deep Learning at Stanford.



STEVENS
INSTITUTE *of* TECHNOLOGY
THE INNOVATION UNIVERSITY®

stevens.edu

Thank You