

# CMSC5733 Social Computing

## Tutorial I: Python and Web Crawling

Yuanyuan, Man

The Chinese University of Hong Kong

sophiaqhsw@gmail.com

# Tutorial Overview

- Python basics and useful packages
- Web Crawling

# Why Python?

- Simple, easy to read syntax
- Object oriented
- Huge community with great support
- Portable and cross-platform
- Powerful standard libs and extensive packages
- Stable and mature
- FREE!

# Python Programming Language

- Download Python 2.7.5 at
  - <http://www.python.org/download/>
- Set up tutorials
  - <http://www.youtube.com/watch?v=4Mf0h3HphEA>
  - or
  - <https://developers.google.com/edu/python/set-up>

# Python Programming Language

- Video tutorials for python
  - <http://www.youtube.com/watch?v=4Mf0h3HphEA>
  - <http://www.youtube.com/watch?v=tKTZoB2Vjuk>
- Document tutorials for python
  - <http://www.learnpython.org/>
  - <https://developers.google.com/edu/python/>  
(suggested!)

# Installing Packages

- Tools for easily download, build, install and upgrade Python packages
  - easy\_install
    - Installation instruction:  
<https://pypi.python.org/pypi/setuptools/1.1.4#installation-instructions>
  - pip
    - In terminal input: `easy_install pip`

# Python Packages

- mysql-python package for MySQL

- Quick install

- ✓ [Download: http://sourceforge.net/projects/mysql-python/](http://sourceforge.net/projects/mysql-python/)

- ✓ `easy_install mysql-python` or `pip install mysql-python`

- MySQL Python tutorial:

- ✓ <http://zetcode.com/db/mysqlpython/>

- Example

```
# remember to install MySQLdb package before import it
import MySQLdb as mdb
# connect with mysql
con = mdb.connect('localhost','root','', 'limitssystem')
# get connection
cur = con.cursor()
sql = "select f_id,f_name,f_action from function"
# execute sql
cur.execute(sql)
# get the result
result = cur.fetchall()
for r in result:
    f_id = r[0]
    f_name = r[1]
    f_action = r[2]
    print f_id,unicode(f_name,"utf-8"),f_action
```



# Python Packages

- urllib2 package
  - Reading a web page
  - Example:

```
import urllib2
# Get a file-like object for the Python Web site's home page.
url = "http://www.python.org"
try:
    f = urllib2.urlopen(url)
    # Read from the object, storing the page's contents in 's'.
    s = f.read()
    print unicode(s,"utf-8")
    f.close()
except IOError:
    print 'problem reading url:',url
```

# Python Packages

- BeautifulSoup package
  - Extracting HTML data
  - Quick Install
    - `easy_install BeautifulSoup` or `pip install BeautifulSoup`
  - Example

```
from BeautifulSoup import BeautifulSoup
html = '<html><body><p class="title">Title</p></body></html>'
soup = BeautifulSoup(html)
print soup.p           # <p class="title">Title</p>
print soup.p["class"]  # title
print soup.p.string     # Title
```

# Python Packages

- Scrapely package

- What is Scrapely:

Scrapely is a library for extracting structured data from HTML pages. Given some example web pages and the data to be extracted, scrapely constructs a parser for all similar pages.

- <https://github.com/scrapy/scrapely>

- Example

```
from scrapely import Scraper
s = Scraper() #instantiating the Scraper class
#proceed to train the scraper by adding some page and the data you expect to
#scrape from there
url1 = 'http://pypi.python.org/pypi/w3lib/1.1'
data = {'name': 'w3lib 1.1', 'author': 'Scrapy project', 'description': 'Library of web-
related functions'}
#train the data
s.train(url1, data)
url2 = 'http://pypi.python.org/pypi/Django/1.3'
print s.scrape(url2)
```

# Python Packages

- Scrapy Package
  - What is Scrapy:
    - Scrapy is a fast high-level screen scraping and web crawling framework, used to crawl websites and extract structured data from their pages.
  - <http://scrapy.org/>
  - Quick Install:
    - `easy_install -U Scrapy` or `pip install Scrapy`

# Web Crawling

- Demo for crawling book.douban.com
  - Crawling all the book information
  - Book link:  
<http://book.douban.com/subject/24753751/>
  - Douban API:  
<https://api.douban.com/v2/book/24753751>
  - Steps: Breadth first crawling all the webpages, get all the book link and book id. According to every book id, get douban API information.
  - Demo

# Web Crawling

- Python package for twitter API
  - Twitter libraries for Python:  
<https://dev.twitter.com/docs/twitter-libraries>
  - Python-twitter for demo:  
<https://dev.twitter.com/docs/twitter-libraries>
  - Applying for oAuth authentication:  
<https://dev.twitter.com/apps/new>
  - Demo

# Web Crawling

- Python package for linkedin API
  - Website: <https://github.com/ozgur/python-linkedin>
  - Installation:
    - \$ pip install python-linkedin
    - \$ pip install requests
    - \$ pip install requests\_oauthlib
  - Applying for Authentication
    - <https://developers.linkedin.com/documents/authentication>



# Web Crawling

- An easy way to extract my connection profile from linkedin

The screenshot shows the LinkedIn 'All Connections' page. The top navigation bar includes the LinkedIn logo, a search bar with the text 'Search for people, jobs, companies, and more...', and an 'Advanced' search button. Below the navigation bar, the 'Filter Connections' section is visible, with a search input and a 'Manage' link. The 'All Connections (6)' section is expanded, showing a list of connections. The connections are listed in a table with columns for a selection checkbox, a profile picture, the name, the title, and a count of mutual connections. The connections listed are:

| Select                   | Profile Picture | Name                   | Title                                                                                    | Mutual Connections |
|--------------------------|-----------------|------------------------|------------------------------------------------------------------------------------------|--------------------|
| <input type="checkbox"/> |                 | <b>Gao, Jiayang</b>    | Development Engineer - Taobao Marketplace                                                | 0                  |
| <input type="checkbox"/> |                 | <b>King, Irwin</b>     | Associate Dean (Education) - Faculty of Engineering, The Chinese University of Hong Kong | 0                  |
| <input type="checkbox"/> |                 | <b>Li, Baichuan</b>    | PhD Student - The Chinese University of Hong Kong                                        | 0                  |
| <input type="checkbox"/> |                 | <b>Shan, Bob</b>       | CEO - Fireflies software                                                                 | 0                  |
| <input type="checkbox"/> |                 | <b>WANG, SHUAI</b>     | Postgraduate at The Chinese University of Hong Kong                                      | 0                  |
| <input type="checkbox"/> |                 | <b>xu, bubblegrace</b> | reporter - media                                                                         | 0                  |

On the right side of the connections list, there is a text box that says: 'Quickly view and organize your connections? Select a category or individual to see contact info, send a message and more.'

At the bottom of the page, there is a status bar that says '238 outstanding sent invitation' and a button labeled 'Export connections' which is highlighted with a red rectangle.

# Web Crawling

- Tools for crawling Sina Weibo
  - Using Sina Weibo API:  
<http://open.weibo.com/wiki/%E5%BE%AE%E5%8D%9AAPI>
  - Using tools from cnpameng.com:  
<http://www.cnpameng.com/>
  - Download data source from datatang.com:  
<http://www.datatang.com>

# Web Crawling

- Scrapinghub
  - Providing web crawling and data processing solutions
  - <http://scrapinghub.com/>
  - Demo

# References

- <http://www.python.org>
- <https://developers.google.com/edu/python/>
- <https://github.com/scrapy/scrapely>
- <http://scrapy.org/>
- <https://dev.twitter.com/docs/twitter-libraries>