Welcome!

When you arrive:
- Sign in
- Pick up full handout
- Connect to Wi-Fi

Materials for Today
- Workshop materials and resources:
  [http://go.illinois.edu/ddrf-curriculum](http://go.illinois.edu/ddrf-curriculum)
- HTRC and HTDL accounts:
  [https://analytics.hathitrust.org](https://analytics.hathitrust.org)
  [https://www.hathitrust.org](https://www.hathitrust.org)
- PythonAnywhere account:
  [https://pythonanywhere.com](https://pythonanywhere.com)

☞ See Handout P1-2, Getting Set Up
Digging Deeper, Reaching Further

Text analysis with the HathiTrust Research Center
GOALS:

▪ Arm librarians with instructional content and tool skills in digital scholarship and digital humanities;
▪ Empower librarians to become research partners on digital projects;
▪ Enable librarians to build foundations for digital scholarship centers and services
Thanks to all our team members!

- **PI**
  - Harriet Green, [University of Illinois at Urbana-Champaign Library](https://library.illinois.edu)

- **Co-PIs**
  - Angela Courtney, [Indiana University-Bloomington Libraries](http://library.indiana.edu)
  - J. Stephen Downie, [School of Information Sciences, University of Illinois](http://info.lib.univ.edu)
  - Terese Heidenwolf, [Lafayette College Library](http://library.lafayette.edu)
  - Geoffrey Morse, [Northwestern University Libraries](http://library.nwu.edu)
  - Amanda Henley, [University of North Carolina at Chapel Hill Library](http://library.unc.edu)

**Project Members (Past and Present)**

- **University of Illinois at Urbana-Champaign**
  - Eleanor Dickson Koehl
  - Ruohua Han
  - Rachel Blomer
  - *Gabriella Stuardi
  - *Beth Sandore Namachchivaya
  - *Ryan Dubnicek
  - *Sayan Bhattacharyya

- **Lafayette College**
  - Lijuan Xu
  - Ben Jahre
  - *Neil McElroy

- **University of North Carolina at Chapel Hill**
  - Sarah Morris
  - *Stewart Varner
  - *Jacob Hill

- **Indiana University**
  - Leanne Nay
  - Robert McDonald
  - Nicholae Cline
  - *Ewa Zegler-Poleska

- **Northwestern University**
  - *Geoffrey Swindells
  - * = former members
Getting started

- Workshop materials and resources:
  
  [http://go.illinois.edu/ddrf-curriculum](http://go.illinois.edu/ddrf-curriculum)

- HTRC Analytics and the HTDL:
  
  [https://analytics.hathitrust.org](https://analytics.hathitrust.org) | [https://www.hathitrust.org](https://www.hathitrust.org)

- PythonAnywhere:
  
  [https://pythonanywhere.com](https://pythonanywhere.com)

- Friendly reminder: Please *do not use Internet Explorer* for our activities today!
Digging Deeper, Reaching Further

Module 1: Getting Started
In this module we’ll…

- Introduce text analysis and broad text analysis workflows
  → Make sense of digital scholarly research practices

- Introduce HathiTrust and the HathiTrust Research Center
  → Understand the context for one text analysis tool provider

- Introduce our hands-on example and case study
  → Recognize research questions text analysis can answer
What is text analysis?

- Using computers to reveal information in and about text
  
  (Hearst, 2003)
  
  - Algorithms discern patterns
  - Text may be “unstructured”
  - More than just search

- What is it used for?
  
  - Seeking out patterns in scientific literature
  - Identifying spam e-mail
How does it work?

- Break textual data into smaller pieces
- Abstract (reduce) text so that a computer can crunch it
- Counting!
  - Words, phrases, parts of speech, etc.
- Computational statistics
  - Develop hypotheses based on counts of textual features
How does it impact research?

- Shift in perspective, leads to shift in research questions
  - Scale-up to “distant reading” (Moretti, 2013)
- One step in the research process
  - Can be combined with close reading
- Opens up:
  - Questions not provable by human reading alone
  - Larger corpora for analysis
  - Studies that cover longer time spans
Discussion

- What examples have you seen of text analysis?

- In what contexts do you see yourself using text analysis? What about the researchers you support?
Text analysis research questions

- May involve:
  - Change over time
  - Pattern recognition
  - Comparative analysis
Hands-on activity

In pairs or small groups, review the summarized research projects available at http://go.illinois.edu/ddrf-research-examples. Then discuss the following questions:

- How do the projects involve change over time, pattern recognition, or comparative analysis?
- What kind of text data do they use (time period, source, etc.)?
- What are their findings?
Example: Rowling and “Galbraith”: an authorial analysis

Question:

Did JK Rowling write The Cuckoo’s Calling under the pen name Robert Galbraith?

Would be impossible to prove through human reading alone!

comparative | patterns

Read more: Rowling and “Galbraith”: an authorial analysis (Juola, 2013)
Example: *Rowling and “Galbraith”: an authorial analysis*

**Approach:**

- Reading led to hunch about authorship
- Computational comparison of diction between this book and others written by Rowling
- Statistical ‘proof’ of authorial fingerprint

*Read more:* Rowling and “Galbraith”: an authorial analysis (Juola, 2013)
Example: *Significant Themes in 19th Century Literature*

**Question:**

*What themes are common in 19th century literature?*

Answering this question requires a very large corpus and an impossible amount of human reading!

patterns | comparative

Read more: Significant Themes in 19th Century Literature (Jockers and Mimno, 2012)
Example: *Significant Themes in 19th Century Literature*

**Approach:**

- Run large quantities of text through a statistical algorithm
- Words that co-occur are likely to be about the same thing
- Co-occurring words are represented as topics

**Read more:** Significant Themes in 19th Century Literature (Jockers and Mimno, 2012)
Example: Significant Themes in 19th Century Literature

From paper - Figure 3: Word cloud of topic labeled “Female Fashion.”
Example: *The Emergence of Literary Diction*

Question:

*What textual characteristics constitute “literary language”?*

This question covers a very large time span!

*change over time | patterns*

Read more: The Emergence of Literary Diction (Underwood and Sellers, 2012)
Example: *The Emergence of Literary Diction*

**Approach:**

- Train a computational model to identify literary genres
- Compare which words are most frequently used over time in non-fiction prose versus “literary” genres
- Demonstrated tendency for poetry, drama, and fiction to use older English words

**Read more:** The Emergence of Literary Diction (Underwood and Sellers, 2012)
Example: *The Emergence of Literary Diction*

**Y axis:** Yearly ratio of words that entered English before 1150 / words that entered from 1150-1699

**From paper:** graph of diction patterns between genres, using frequency counts

**X axis:** Year
HTRC for text analysis

Digitized text

Scanned & OCR-ed

Computational methods

E.g. Word counts, classification, topic modeling

Analysis

at scale from the digital library

providied tools and services

HathiTrust Research Center
HathiTrust

- Founded in 2008
- Grew out of large-scale digitization initiative at academic research libraries
  - With roots in Google Books project
- Over 120 partner institutions continue to contribute
HathiTrust Digital Library

- Contains over 16 million volumes
  - ~ 50% English
  - From the 15th to 21st century, 20th century concentration
  - ~ 63% in copyright or of undetermined status
- Search and read books in the public domain
HathiTrust Research Center

- Facilitates text analysis of HTDL content
- Research & Development
- Located at Indiana University and the University of Illinois
Non-consumptive research

Research in which computational analysis is performed on text, but not research in which a researcher reads or displays substantial portions of the text to understand the expressive content presented within it.

- Complies with copyright law
- Foundation of HTRC work
- Other terms: non-expressive use
Discussion

Are you (or your colleagues) currently offering research support for text analysis?

- How so?
- Why or why not?
- What kinds of questions and/or projects does your library handle?
Workshop outline

- Follow the research process:
  - Gathering textual data: 2 modules
  - Working with textual data: 1 module
  - Analyzing textual data: 2 modules
  - Visualizing textual data: 1 module

- Hands-on activities around a central research question & case study example at each step
  - Using both HTRC and non-HTRC tools
Workshop outline

- Build skills to engage with text analysis research
- Covers programming concepts
  - But won’t teach you to code!
- Introduces computational methods
  - But won’t delve into all nuances
Sample Reference Question

Question:

I’m a student in history who would like to incorporate digital methods into my research. I study American politics, and in particular I’d like to examine how concepts such as liberty change over time.

Approach:

- We’ll practice approaches for answer this question throughout the workshop
Case Study

Inside the Creativity Boom | Researcher: Samuel Franklin

Question:

How do the use and meaning of creative and creativity change over the 20th century?

Approach:

▪ We’ll discuss how this researcher approached his question throughout the workshop

Learn more: https://wiki.htrc.illinois.edu/x/CADiAQ
A word of caution...

Workshop outline suggests research workflow like:

1. Find text
2. Prepare text
3. Apply algorithm
4. Visualize results
A word of caution...

Actual research workflow like:

- Search for text
- Clean text
- Get access to text
- Exploratory visualization
- Prepare text
- Apply algorithm
- Visualize results
Discussion

- What are some of the characteristics of a good candidate research question/project for using text analysis methods?
Questions?
References

Digging Deeper,  
Reaching Further  

Module 2.1: Gathering Textual Data  
Finding and Curating Text Data
In this module we’ll…

- Explore the concept of a text data and where to find it
  
  \(\rightarrow\) Provide data reference for researchers

- Build a HathiTrust workset
  
  \(\rightarrow\) Gain experience in building a textual dataset

- Learn how Sam built a Creativity Corpus of HathiTrust volumes
  
  \(\rightarrow\) Understand real-world data collection strategies
Where we’ll end up

Create a collection of volumes from the HathiTrust Digital Library and prepare it for analysis in HTRC Analytics as a workset.
“Text analysis projects share in common 3 challenges. **First**, data of interest must be found. **Second**, data must be gettable. **Third**, if it’s not already formed according to wildest dreams, ways must be known of getting data into a state that they are readily usable with desired methods and tools.”

Kludging: Web to TXT (Padilla, 2015)

[http://www.thomaspadilla.org/2015/08/03/kludge/](http://www.thomaspadilla.org/2015/08/03/kludge/)
Finding text

- Not always easy
  - copyright restrictions
  - licensing restrictions
  - format limitations
  - hard-to-navigate systems

** issues more pronounced at scale**
Vendor databases

- Be aware of licensing restrictions

Strategies

- Addendums to libraries’ contracts
- Vendor-provided services
- Asking for special permission case-by-case

Example: JSTOR Data for Research
Library/archives digital collections

- Wealth of material, but:
  - Often siloed
  - Access not formulated for research at scale

- Things to look for:
  - Plain text
  - Bulk download

- Example: UNC’s DocSouth Data
Social media

- Popular with social science researchers

- To access:
  - Some provide systems to access text
  - Or there are 3rd-party tools on the market

- Example: Twitter API (Application Programming Interface)
## Hands-on activity

Building a corpus for political history, what are the strengths and weaknesses of each of these broad sources for textual data?

<table>
<thead>
<tr>
<th></th>
<th>Strengths</th>
<th>Weaknesses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vendor database</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Library/archives</td>
<td></td>
<td></td>
</tr>
<tr>
<td>digital collections</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Social media</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Evaluating sources of text data

<table>
<thead>
<tr>
<th>Does the researcher already have a data source in mind?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Are there copyright and licensing concerns?</td>
</tr>
<tr>
<td>How technically experienced is the researcher?</td>
</tr>
<tr>
<td>Is the text they want to use already digitized?</td>
</tr>
<tr>
<td>What format does the researcher expect the data in?</td>
</tr>
<tr>
<td>How much flexibility is needed for working with the data?</td>
</tr>
<tr>
<td>Does the researcher have funding?</td>
</tr>
</tbody>
</table>

What is the period, place, person of interest?
Building corpora

- Identify texts through full text search
  - Use a key term or phrase
- Identify texts through metadata
  - Search by certain author(s)
  - Search within a date range
  - Search for a specific genre
- Or some combination of the two!
Building corpora

- Process usually involves deduplication
- What to keep/discard is project dependent
- Examples of deduplication:
  - OCR quality
  - Earliest edition
  - Editions without forewords or afterwords
HTRC Worksets

- User-created collections of text from the HathiTrust Digital Library
  - think of them as textual datasets
- Can be shared and cited
- Suited for non-consumptive access
HTRC Worksets

Workset viewed on the web

Workset manifest
Building worksets

- Stored in HTRC
  - Require account with university email address

- Ways to build:
  - Import from HT Collection Builder
  - Compile volume IDs elsewhere
Sample Reference Question

I’m a student in history who would like to incorporate digital methods into my research. I study American politics, and in particular I’d like to examine how concepts such as liberty change over time.

Approach:

- Create a textual dataset of volumes related to political speech in America with the HT Collection Builder, and upload it to HTRC Analytics as a workset for analysis
Hands-on activity

In this activity, you will log in to HTDL and create a collection containing volumes of the public papers of the presidents of the United States, and import it into HTRC Analytics as a workset.

Follow the instructions on the handout to build your workset.

Websites:

- HTDL: https://www.hathitrust.org
- HTRC Analytics: https://analytics.hathitrust.org
Go to HTDL interface

https://www.hathitrust.org
Log in
Log in
Search for volumes

• Click on “Advance full-text search”
Search for volumes

Advanced Full-text Search:

Search information within or about an item

Search Tips

this exact phrase \(\downarrow\) United States

AND \(\downarrow\) this exact phrase \(\downarrow\) public papers

+ Add a pair of search fields

Limit to:

Full view only \(\n\) Year of publication: During or after \(\downarrow\)

Language

Limit to Original Format

All

All
Filter results and select volumes

Filter results on the left sidebar

An advanced search for volumes that contain all the words/phrases below in the title field: “public papers” and “United States”

Select all or some of the returned search items for your collection.
Add volumes to collection

Once texts are selected, click “Select Collection” → choose “[CREATE NEW COLLECTION]” → click “Add Selected”
Add collection metadata

Collection Name: PoliticalSpeech
Description: A collection of volumes of public speeches by the presidents of the United States
Private/Public: Public
View your collection
View your collection
Download the metadata of your collection
Go to HTRC Analytics

HathiTrust Research Center Analytics
Supports large-scale computational analysis of the works in the HathiTrust Digital Library to facilitate non-profit and educational research.

Featured Services

Extracted Features
Text Analysis Algorithms
Data Capsules

https://analytics.hathitrust.org
Welcome! Returning users signing into the new HTRC Analytics interface for the first time must reset their password using the "Forgot Password" link below.
HathiTrust Research Center Analytics

Supports large-scale computational analysis of the works in the HathiTrust Digital Library to facilitate non-profit and educational research.

Featured Services
Go to Worksets page
Choose to create a workset
Add metadata and upload TSV file

Create A Workset

Upload a list of HathiTrust volume IDs to create an HTRC workset.

To generate a list of volume IDs, create a HathiTrust collection, download the metadata, and upload the metadata file here. Currently, only public domain volumes can be analyzed with HTRC Algorithms, so ensure your collection contains only "Full View" volumes from the digital library if you want to use HTRC Algorithms.

File must be in CSV (comma-separated-value) or TXT format. The only required field is HathiTrust ID (for example, hvd.hn5f64) which should be the first column of your data. A collection metadata file downloaded from HathiTrust can be uploaded without any editing.

Name

poli_science_DDRF

Only characters A-Z 0-9 ()- or _ allowed.

Description

Political science collection for DDRF workshop

File

Choose File mb-6.txt

You can only upload worksets in CSV (comma-separated-values) or TXT format.

Private Workset

Create Workset
View created workset
Workset review

- How did it go?
- What kind of search criteria did you use?
- Did you find any challenges?
Case study: *Inside the Creativity Boom*

Building a creativity corpus

- Searched across full text of HTDL for creativ*
- Made initial list of over million volumes
- Deduplicated
  - Kept different editions of same work; discard multiple copies of same edition
- Ended up with refined list (workset) of volumes
What expertise do librarians already have to help with building a corpus for textual analysis?
Questions?
References

Module 2.2: Gathering Textual Data

 Bulk retrieval
In this module we’ll…

- Introduce options for automated data retrieval
  - Facilitate researcher needs for data access

- Practice web scraping with `wget` on the command line
  - Gain confidence automating data collection

- See how Sam gathered data for his Creativity Corpus
  - Learn how one scholar used automated data collection
Where we’ll end up

Run command from command line interface...

...to scrape the text from a webpage.
Bulk retrieval

- Most researchers will need more than 1 or 10 texts
  - Hundreds, thousands, or millions of texts

- Getting lots of data could take lots of time!
  - Point-and-click is inefficient
  - Automate when possible
Automating retrieval

Transferring files

- FTP or SFTP: (Secure) File Transfer Protocol
  - moves files from one place to another on the internet

- rsync
  - Efficient: sends only the differences
  - Run from command line
  - Used by HathiTrust, can be used to download Extracted Features data
Automating retrieval

Web scraping (grabbing text on the web)

- Avoids tedious copying-and-pasting
- Some ways to scrape text from the web:
  - Run commands such as wget or curl in the command line
  - Write and run a script (a file of programming statements)
  - Use software such as webscraper.io or Kimono

*We’ll do an activity with this one later!*
Web scraping for the wise

- Web scraping puts a large workload on targeted server
  - This can upset the data holder
- Some data providers are more than willing to share
  - Ask for access
  - Check for an API
- Otherwise, time your requests to add a delay between server hits
  - It’s polite
  - Also signifies you are not a malicious attacker
Automating retrieval

APIs (Application Programming Interfaces)

- Digital pathways to or from content
  - Sometimes need a “key” for access
- Can be used to gain programmatic access
  - Usually need to write code to retrieve content
  - Sometimes have graphical user interface (GUI)
- Examples: A number of digital content providers have APIs
  - Twitter API: display tweets on a non-Twitter website
  - Chronicling America API: https://chroniclingamerica.loc.gov/about/api/
HathiTrust Bibliographic API

- Access at: https://www.hathitrust.org/bib_api
  - Provides programmatic access to bibliographic metadata for volumes in the HathiTrust
  - Metadata retrieved using a specially formatted URL and the volume ID number
Hands-on activity

- Use the HathiTrust Bibliographic API to retrieve the metadata for a volume of your choice
  - Search HTDL for a volume (hathitrust.org)
  - Click on the “Full view” or “Limited (search-only)” option of the volume
  - Find the volume ID in the URL (all characters between “id=” and “,;”, including all numbers and letters)

- Retrieve the volume’s metadata with an API call
Anatomy of an API call

- Types of volume ID: htid, OCLC, record number, lccn, isbn, issn
- A structured URL

```
https://catalog.hathitrust.org/api/volumes
                /brief OR /full
                /<ID type>
                /<ID.json>
```

```
https://catalog.hathitrust.org/api/volumes/full/htid/mdp.39015005337046.json
```
Intro to the command line
Background

- Command line interfaces
  - On Mac: Terminal
  - On Windows: Command Prompt
  - They do not speak the same language!

- Bash shell = Unix shell
  - Default on Mac
  - To use on Windows, install GitBash or Cygwin
    - *Bash on Ubuntu on Windows* comes installed on Windows 10

**We’re going to use the built-in Bash shell environment in PythonAnywhere**
What’s PythonAnywhere?

- PythonAnywhere is a browser-based programming environment
  - Write code
  - Save files
- Comes with a built-in Bash console
  - Run commands
- Does not interact with your local file system
Tips for working in a shell

- Directory = folder
- Case, spaces, and punctuation matter
- Tab to autocomplete a line
- Hit up/down arrow to see last commands entered
- Use “q” to quit viewing a file
Basic Bash commands

- Introductory video:

http://go.illinois.edu/ddrf-bash-video
Hands-on activity

Open a Bash console in PythonAnywhere, and practice the basic commands to:

- See your current directory
  `pwd`
- Are there any files in your directory? List them:
  `ls`
In your Bash console, practice the following:

- Unzip activity files (activity_files.zip) in your user directory:
  
  ```
  unzip activity_files.zip
  ```

- Files were unzipped to a new directory called activity_files.

- Move the files from activity_files to your user directory. Substituting your Python Anywhere user name for `[your_username]`:
  
  ```
  mv activity_files/* /home/[your_username]/
  ```
Hands-on activity

Use the `ls` command to check if files were successfully unzipped and moved:

- List files and directories under your current directory: `ls`

- List should include:

  ```
  1930 README.txt activity_files.zip remove_tag.py
  top_adjectives.py 1970 activity_files
  remove_stopwords.py stopwords.txt word_count.py
  ```
Hands-on activity

More practice:

- Make a directory called ‘test’
  
  mkdir test

- Change into the directory you just made, then back out of it
  
  cd test
  cd ..

Note: Please make sure you are back in your main directory after finishing the activity!
Command line tool: `wget`

- `wget` is a powerful way to transfer files from a server
  - i.e. web scraping
  - Follows links on the page and grabs that content, too

- You can specify **options**
  - Options tailor the command to do what you want
  - They are often written as single “flagged” letters, such as `–d` or `–r`

- Options for `wget` include:
  - `–l`: how far in the hierarchy you want to go
  - `--limit-rate=___`: sets the file transfer speed
Sample Reference Question

I’m a student in history who would like to incorporate digital methods into my research. I study American politics, and in particular I’d like to examine how concepts such as liberty change over time.

Approach:

Download text from an open resource, WikiSource, to start building a corpus.
Hands-on activity

In this activity, you will run a command to scrape text from a webpage version of George Washington’s Fourth State of the Union Address.

Website Source: https://en.wikisource.org/wiki/George_Washington%27s_Fourth_State_of_the_Union_Address
Web scraping

What You Need:

- PythonAnywhere: [www.pythonanywhere.com](http://www.pythonanywhere.com)
- Website URL: [https://en.wikisource.org/wiki/George_Washington%27s_Fourth_State_of_the_Union_Address](https://en.wikisource.org/wiki/George_Washington%27s_Fourth_State_of_the_Union_Address)
- Command:

```
```
Fellow-Citizens of the Senate and of the House of Representatives:

It is some abatement of the satisfaction with which I meet you on the present occasion that, in felicitating you on a continuance of the national prosperity generally, I am not able to add to it information that the Indian hostilities which have for some time past distressed our Northwestern frontier have terminated.

You will, I am persuaded, learn with no less concern than I communicate it that reiterated endeavors toward effecting a pacification have hitherto issued only in new and outrageous proofs of persevering hostility on the part of the tribes with whom we are in contest. An earnest desire to procure tranquility to the frontier, to stop the further effusion of blood, to arrest the progress of expense, to forward the prevalent wish of the nation for peace has led to strenuous efforts through various channels to accomplish these desirable purposes; in making which efforts I consulted less my own anticipations of the event, or the scruples which some considerations were calculated to inspire, than the wish to find the object attainable, or if not attainable, to ascertain unequivocally that such is the case.

A detail of the measures which have been pursued and of their consequences, which will be laid before you, while it will confirm to you the want of success thus far, will, I trust, evince that means as proper and as efficacious as could have been devised have been employed. The issue of some of them, indeed, is still depending, but a favorable one, though not to be despaired of, is not promised by anything that has yet happened.

In the course of the attempts which have been made some valuable citizens have fallen victims to their zeal for the public service. A sanction commonly respected even among savages has been found in this instance insufficient to protect from massacre the emissaries of peace. It will, I presume, be duly considered whether the occasion does not call for an exercise of liberality toward the families of the deceased.

It must add to your concern to be informed that, besides the continuation of hostile appearances among the tribes north of the Ohio, some threatening symptoms have of late been revived among some of those south of it.

https://en.wikisource.org/wiki/George_Washington%27s_Fourth_State_of_the_Union_Address
Open shell and enter command

```
19:00 ~ $ wget -l 1 --limit-rate=20k https://en.wikisource.org/wiki/George_Washington%27s_Fourth_State_of_the_Union_Address --output-document=washington_4.txt
```

Enter:
```
```

Tip: Copy and paste the URL!
Run command successfully

Resolving proxy.server (proxy.server)... 10.0.0.82
Connecting to proxy.server (proxy.server)|10.0.0.82|:3128... connected.
Proxy request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: ‘washington_4.txt’

[                    ] 42,088  20.0KB/s  in 2.1s

2017-07-06 19:00:48 (20.0 KB/s) – ‘washington_4.txt’ saved [42088]
19:00 ~ $
View new file
View new file
What happened?

- You executed a command to scrape the text from the URL
- It grabbed everything on the page
- What’s next?
  - Prepare the data for analysis

*What might need to be removed from the files before analysis?*
On your own

- Edit the command to scrape George Washington’s second State of the Union speech and save to a new text file.
- Can you view your scraped files using the *less* command?
Other options to explore

- **Beautiful Soup**
  - [https://www.crummy.com/software/BeautifulSoup/](https://www.crummy.com/software/BeautifulSoup/)
  - Python-based web scraping
  - Lots of options, very powerful

- **Write web scraping into a script**
  - A script is a list of directions for your computer to follow
  - Direct computer to grab many files to retrieve content in bulk
  - Remember to time your requests!
# Bulk HathiTrust data access

## HT and HTRC datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Kind of data</th>
<th>Description</th>
<th>Volumes available</th>
</tr>
</thead>
<tbody>
<tr>
<td>HT Custom data request</td>
<td>Full text</td>
<td>Download page images and plain text OCR</td>
<td>Public domain</td>
</tr>
<tr>
<td>HTRC Extracted Features</td>
<td>Abstracted text and metadata</td>
<td>JSON files for each of 15.7 million volumes in HathiTrust</td>
<td>All</td>
</tr>
<tr>
<td>HTRC Data API</td>
<td>Full text</td>
<td>Plain text OCR</td>
<td>Public domain for now, in HTRC Data Capsule only</td>
</tr>
</tbody>
</table>
Case Study: *Inside the Creativity Boom*

After creating final list of volumes:

- Used `rsync` to retrieve HTRC Extracted Features for the volumes
- Remember `rsync` is a command line utility, like `wget`, that transfers files between computers
Case Study: *Inside the Creativity Boom*

- What exactly is the HTRC Extracted Features dataset?

  JSON files for each volume in the HathiTrust that researchers can download

```json
{
  "id": "uc1.b3419888",
  "metadata": {
    "schemaVersion": "1.2",
    "dateCreated": "2015-02-12T13:30",
    "title": "Zoonomia = or The laws of organic life / by Erasmus Darwin.",
    "pubDate": "1809",
    "language": "eng",
    "htBibUrl": "http://catalog.hathitrust.org/api/volumes/full/htid/uc1.b3419888.json",
    "handleUrl": "http://hdl.handle.net/2027/uc1.b3419888",
    "oclc": "3679915",
    "imprint": "Thomas and Andrews, 1809."
  },
  "features": {
    "schemaVersion": "2.0",
    "dateCreated": "2015-02-20T23:58",
    "pageCount": "616",
    "pages": [1, 2, 3, ...]
  }
}
```

**Metadata**: bibliographic; inferred

**Data**: words and word counts
Read and reflect

Santa Barbara Statement on Collections as Data (Collections as Data National Forum, 2017)
https://collectionsasdata.github.io/statement/

Provides a set of high level principles to guide collections as data work
“With a few exceptions, cultural heritage institutions have rarely built digital collections or designed access with the aim to support computational use. Thinking about collections as data signals an intention to change that.”

“Any digital material can potentially be made available as data that are amenable to computational use. Use and reuse is encouraged by openly licensed data in non-proprietary formats made accessible via a range of access mechanisms that are designed to meet specific community needs.”

“Ethical concerns are integral to collections as data.”

**Principle 2 for collections as data:** “Collections as data development aims to encourage computational use of digitized and born digital collections.”
Read and reflect  

§ Does your library provide access to digital collections as data? 

§ How so? Why not? How could it?
Questions?
Digging Deeper, Reaching Further

Module 3: Working with Textual Data
In this module we’ll…

- Think about what happens when text is data
  - Understand best practice in the field
- Consider common steps to cleaning and preparing text data
  - Make recommendations to researchers
- Practice data cleaning on the speeches we scraped
  - Gain experience with Python for working with data
- Learn how Sam prepared his Creativity Corpus for analysis
  - See how one scholar data prepared data
Where we’ll end up

Go from HTML-formatted text...
...to cleaner text ready for analysis.
Humanities data

- Data is material generated or collected while conducting research
- Examples of humanities data:
  - Citations
  - Code/Algorithms
  - Databases
  - Geospatial coordinates

Can you think of others?

Text as data

- Data quality
  - Clean vs. dirty OCR
  - HathiTrust OCR is dirty (uncorrected)

- Analyzed by corpus/corpora
  - Text corpus: a digital collection OR an individual’s research text dataset
  - Text corpora: “bodies” of text

- Text decomposition/recomposition (Rockwell, 2003)
  - Cleaning data involves discarding data
  - Prepared text may be illegible to the human reader
Preparing data

A researcher may:

- Correct OCR errors
- Remove title, header information
- Remove html or xml tags
- Split or combine files
- Remove certain words, punctuation marks
- Lowercase text
- Tokenize the words
Key concepts

Chunking text

Splitting text into smaller pieces before analysis. May be divided by paragraph, chapter, or a chosen number of words (e.g. 1000 word chunks).

Speeches of Abraham Lincoln

- Gettysburg Address
- First Inaugural
- Second Inaugural
- House Divided Speech
Key concepts

Grouping text

Combining text into larger pieces before analysis.

- Gettysburg Address
- First Inaugural
- Second Inaugural
- House Divided Speech
- Speeches of Abraham Lincoln
- Farewell Address
- First Inaugural
- Second Inaugural
- 1796 State of the Union
- Speeches of George Washington
Key concepts

Tokenization

Breaking text into pieces called tokens. Often certain characters, such as punctuation marks, are discarded in the process.

[four], [score], [and], [seven], [years], [ago], [our], [fathers], [brought], [forth], [on], [this], [continent], [a], [new], [nation], [conceived], [in], [liberty], [and], [dedicated], [to], [the], [proposition], [that], [all], [men], [are], [created], [equal]
Preparing data

- Preparation affects results
  - Amount of text and size of chunks
  - Which stop words removed; which characters are included
  - Whether to lowercase and normalize words

- Preparation for analysis takes time, effort
  - This is where scripting becomes useful!
Hands-on Activity

• In groups of 2 or 3, assign each person several of the text preparation actions seen in the table to the right (Denny and Spirling, 2017).
• Read the descriptions. Then take turns explaining each to your group.

<table>
<thead>
<tr>
<th>Term</th>
</tr>
</thead>
<tbody>
<tr>
<td>Punctuation</td>
</tr>
<tr>
<td>Numbers</td>
</tr>
<tr>
<td>Lowercasing</td>
</tr>
<tr>
<td>Stemming</td>
</tr>
<tr>
<td>Stopword Removal</td>
</tr>
<tr>
<td>n-gram Inclusion</td>
</tr>
<tr>
<td>Infrequently Used Terms</td>
</tr>
</tbody>
</table>
Scripting with Python

- Python is a scripting language
- Good for working with data
  - Interpreted language $\rightarrow$ follows step-by-step directions
- Relatively straightforward syntax
  - Avoids excess punctuation
Using Python: Interactive programming

- Using a **python interpreter**
- Run each step at the prompt
- If you enter “Python” on the command line, you will start the interpreter
- *We aren’t using it today!*
Using Python: Write & run scripts

- Scripts are directions for your computer to follow
- Save the script as a file ending in .py
- On the command line, run the script

→ This is how we’ll do it!
Anatomy of running a Python script

```
python <script filename> <arguments>
```

1. Tells the computer to be ready for Python
2. Directs which file to run
3. The file may ask for additional information to be given at runtime, called arguments

**Example:** `python myscript.py newfile.txt`
Sample Reference Question

I’m a student in history who would like to incorporate digital methods into my research. I study American politics, and in particular I’d like to examine how concepts such as liberty change over time.

Approach:

Use a Python script to prepare the text data we scraped from Wikisource.
In this activity, you will run a Python script that will remove the HTML tags from George Washington’s *Fourth State of the Union Address* you scraped earlier from Wikisource.
Strip HTML tags

- What You Need:
  - PythonAnywhere bash console
  - washington_4.txt
  - remove_tag.py script
Go to Files to view script
```python
import sys
import re

file_name = sys.argv[1]

with open(file_name, 'r') as myfile:
    contents = myfile.read().replace('
', '')
    pattern = r"<p>(.*)</p>"
    new_text = str(re.findall(pattern, contents))
    new_text = new_text.replace("<i>", " "
    new_text = new_text.replace("</i>", " "
    new_text = new_text.replace("<a>", " "
    new_text = new_text.replace("</a>", " "
    new_text = new_text.replace("<b>", " "
    new_text = new_text.replace("</b>", " "
    cleaned = open('tagless_file.txt', 'w')
    cleaned.write(new_text)
```
Open shell and run script

Enter command:
python remove_tag.py washington_4.txt
Find new file

- .bash_history
- .bashrc
- .gitconfig
- .my.cnf
- .mysql_history
- .profile
- .python_history
- .pythonstartup.py
- .sqlite_history
- .vimrc
- README.txt
- get_verbs.py
- mdp.49015002221837.json.bz2
- remove_stopwords.py
- remove_tag.py
- see_nouns_new.py
- stopwords.txt
- tagless_file.txt
- washington_4.txt
Review results

1 ['Fellow-Citizens of the Senate and of the House of Representatives: ']
2 'It is some abatement of the satisfaction with which I meet you on the present occasion that, in felicitating you on a
3 continuance of the national prosperity generally, I am not able to add to it information that the Indian hostilities which
4 have for some time past distressed our Northwestern frontier have terminated.' 'You will, I am persuaded, learn with no less
5 concern than I communicate it that reiterated endeavors toward effecting a pacification have hitherto issued only in new and
6 outrageous proofs of persevering hostility on the part of the tribes with whom we are in contest. An earnest desire to procure
7 tranquility to the frontier, to stop the further effusion of blood, to arrest the progress of expense, to forward the prevalent
8 wish of the nation for peace has led to strenuous efforts through various channels to accomplish these desirable purposes; in
9 making which efforts I consulted less my own anticipations of the event, or the scruples which some considerations were
10 calculated to inspire, than the wish to find the object attainable, or if not attainable, to ascertain unequivocally that
11 such is the case.' 'A detail of the measures which have been purveyed and of their consequences, which will be laid before
12 you, while it will confirm to you the want of success thus far, will, I trust, evince that means as proper and as efficacious
13 as could have been devised have been employed. The issue of some of them, indeed, is still depending, but a favorable one,
14 though not to be despair of, is not promised by anything that has yet happened.' 'In the course of the attempts which have
15 been made some valuable citizens have fallen victims to their zeal for the public service. A sanction commonly respected even
16 among savages has been found in this instance insufficient to protect from massacre the emissaries of peace. It will, I presume,
17 be duly considered whether the occasion does not call for an exercise of liberality toward the families of the deceased.'
18 'It must add to your concern to be informed that, besides the continuation of hostile appearances among the tribes north of
19 the Ohio, some threatening symptoms have of late been revived among some of those south of it.' 'A part of the Cherokees,
20 known by the name of Chickamaugas, inhabiting five villages on the Tennessee River, have long been in the practice of
21 committing depredations on the neighboring settlements.' 'It was hoped that the treaty of Holston, made with the Cherokee
22 Nation in July, 1791, would have prevented a repetition of such depredations; but the event has not answered this hope. The
23 Chickamaugas, aided by some banditti of another tribe in their vicinity, have recently perpetrated wanton and unprovoked
24 hostilities upon the citizens of the United States in that quarter. The information which has been received on this subject
25 will be laid before you. Hitherto defensive precautions only have been strictly enjoined and observed.' 'It is not understood
26 that any breach of treaty or aggression whatsoever on the part of the United States or their citizens is even alleged as a
27 pretext for the spirit of hostility in this quarter.' 'I have reason to believe that every practicable exertion has been made
28 (pursuant to the provision by law for that purpose) to be prepared for the alternative of a prosecution of the war in the event
29 of a failure of pacific overtures. A large proportion of the troops authorized to be raised have been recruited, though the
30 number is still incomplete, and pains have been taken to discipline and put them in condition for the particular kind of
What happened

- Looks for `< >` to denote html tags
  - Using regular expressions and Python’s `replace()`
- Writes the characters that are NOT html tags and their contents to a file
- Note: this script is simple, but not the most robust way to remove tags
On your own

Can you run a script to remove stop words?

- The script is called remove_stopwords.py

- Hint: The remove_stopwords.py script requires three arguments:
  - The input filename of your tagless file
  - The filename of the list of stop words
  - The output filename you make up

- Edit the stopwords.txt file to customize your list

See Handout P13
Case Study: *Inside the Creativity Boom*

After downloading the Extracted Features data for the relevant volumes, used scripting to:

- Narrow corpus to individual pages that contained creativ*
  - Discarded all other pages

- Discard certain tokens such as pronouns and conjunctions
  - To keep only the most “meaningful” terms
Read and Reflect…

- Passage from “Against Cleaning” by Katie Rawson and Trevor Muñoz

- They suggest a strategy for dealing with humanities data:
  - Shared authority control across data sets
  - Indexes for nuance
  - Tidy, not clean data
“When humanities scholars recoil at data-driven research, they are often responding to the reductiveness inherent in this form of scholarship. This reductiveness can feel intellectually impoverishing to scholars who have spent their careers working through particular kinds of historical and cultural complexity… From within this worldview, data cleaning is then maligned because it is understood as a step that inscribes a normative order by wiping away what is different. The term “cleaning” implies that a data set is ‘messy.’ “Messy” suggests an underlying order. It supposes things already have a rightful place, but they’re not in it—like socks on the bedroom floor rather than in the wardrobe or the laundry hamper.”

- “Against Cleaning” (Rawson and Muñoz, 2016)
Discussion

- What does this excerpt suggest about the nuances of data cleaning?
- What does “clean” imply?
- How might you talk to researchers on your campus who would be uncomfortable with the idea of clean v. messy data?
Questions?
References


Module 4.1: Analyzing Textual Data

Using Off-the-Shelf Tools
In this module we’ll…

- Weigh the benefits and drawbacks of pre-built tools for text analysis
  - Evaluate researcher questions and requests, and match tool to request

- Learn how a web-based topic modeling algorithm works
  - Gain experience with off-the-shelf solutions text mining

- Run the HTRC Topic Modeling algorithm and analyze the results
  - Build confidence with the outcomes of data-intensive research

- See how Sam explored HTRC Algorithms for his research
  - Understand how a researcher evaluated an off-the-shelf tool
Where we’ll end up

Topic models created using HTRC algorithm.
Pre-built tools

- **Benefits**
  - Easy to use, good for teaching

- **Drawbacks**
  - Less control, limited capabilities

- **Examples:**
  - Voyant, Lexos
  - HTRC algorithms: e.g. Topic Modeling algorithm
Do-it-yourself tools

- Alternative to pre-built, off-the-shelf tools
- Involve programming

Benefits:
- Run on your own, allow for more parameterization and control

Drawback:
- Require technical knowledge

*We’ll return to this later…*
Choosing a pre-built tool

Depending on the goal of the researcher:

- **Quick analysis and visualizations:**
  - Voyant
  - Lexos

- **Concordances:**
  - AntConc
  - Voyant

- **Machine learning**
  - WEKA Workbench aids machine learning
HTRC algorithms

- Plug-and-play text analysis
- Built into the HTRC interface
  - Mostly “as-is”
  - Limited parameterization
  - Analyze HTRC worksets
- Good when you want to use HT text specifically
Choosing an HTRC algorithm

- Some are task oriented:
  - Run spellcheck report
  - Create word count
  - Visualize most frequently used words

- Others are more analytic
  - Generate topic models
  - Compare most relevant words in two worksets
Key terms in text analysis

Bag-of-words

Concept where grammar and word order of the original text are disregarded and frequency is maintained.

created the four in new are ago Liberty fathers that forth continent a nation seven and conceived equal score dedicated on to years this all our men brought and proposition
Key terms in context

**Topic Modeling**

- **Chunk** text into documents
- **Documents** = bags of words
- **Stop words** are removed
- Each word in each document is compared
- Words that tend to occur together in documents are likely to be about the same thing
- **Topics** are predictions of words co-occurrence
Tips for topic modeling

- Treat topic modeling as step in analysis
- Input affects output
  - Number of texts analyzed, number of topics generated
  - Be familiar with your input data
  - Know that stop words can shape results
- Examine results to see if they make sense
- Understand the tool
HTRC topic modeling description

Meandre Topic Modeling

Description
Identify "topics" in a workset based on words that have a high probability of occurring close together in the text. Topics are models trained on co-occurring text using Latent Dirichlet Allocation (LDA), where each topic is treated as a generative model and volumes are assigned a probability of how likely each topic is to have generated that text. The most likely words for a topic are displayed as a word cloud.

How it works:
- loads each page of each volume from HTRC;
- removes the first and last line of each page;
- joins hyphenated words that occur at the end of the line;
- removes all tokens that do not consist of alphanumeric characters;
- filters stop words;
- replaces "not " with "not_" to deal with negations;
- creates a topic model using Mallet;
- displays the top 200 tokens in a tag cloud

Note: The upper limit on the number of volumes is 1000.
Result of job: xml file with topics, and visualizations of them in the form of tag clouds.

Version
1.2

Author(s)
Loretta Auville

https://analytics.hathitrust.org/algorithms/Meandre_Topic_Modeling
Hands-on activity

- With a partner, each read the algorithm descriptions.
- Take turns explaining what they do.
- **Bonus:** Do you have experience with a research question well-suited to each algorithm? Describe it to your group.

<table>
<thead>
<tr>
<th>Tool</th>
<th>What does it do?</th>
<th>Example research question?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tag cloud</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OpenNLP Entities</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Topic modeling</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

https://wiki.htrc.illinois.edu/x/HoJnAQ
Sample Reference Question

I’m a student in history who would like to incorporate digital methods into my research. I study American politics, and in particular I’d like to examine how concepts such as liberty change over time.

Approach: Run topic modeling algorithm to get a feel for the topics present in your workset.
Hands-on activity

In this activity you will run the topic modeling algorithm in the HTRC Analytics to explore the most prevalent topics in our president public papers workset.

What You Need:

Website: https://analytics.hathitrust.org

Workset: poli_science_DDRF
About the political science workset

- Government-published series: *Public papers of the presidents of the United States*
  - “Public Messages, Speeches, and Statements of the President”
- 16 volumes from U.S. presidents during the 1970s:
  - Jimmy Carter
  - Gerald Ford
  - Richard Nixon
- We’ll use the same workset so that we can all examine the same results!
Using the HTRC Algorithms

See Handout P16

https://analytics.hathitrust.org
Analysis in the HTRC

Meandre Tagcloud with Cleaning (v1.2)
Performs cleaning of the text before it allows you to create a tag cloud visualization of the most frequently occurring words in a workset. In a tag cloud, the size of the word is displayed in proportion to the number of times it occurred.

Execute

Meandre Topic Modeling (v1.2)
Identify "topics" in a workset based on words that have a high probability of occurring close together in the text. Topics are models trained on co-occurring text using Latent Dirichlet Allocation (LDA), where each topic is treated as a generative model and volumes are assigned a probability of how likely each topic is to have generated that text. The most likely words for a topic are displayed as a word cloud.

Execute

Simple Deployable Word Count (v1.4)
Identify the words that occur most often in a workset and the number of times they occur.

Execute
Prepare to run an algorithm

- Removes all tokens that lie at the end of the line;
- joins hyphenated words that occur at the end of the line;
- removes all tokens that do not consist of alphanumeric characters
- filters stop words;
- replaces "not " with "not_" to deal with negations;
- creates a topic model using Mallet;
- displays the top 200 tokens in a tag cloud

Note: The upper limit on the number of volumes is 1000.

Result of job: xml file with topics, and visualizations of them in the form of tag clouds.

Version
1.2

Author(s)
Loretta Auvi

Job Name (required)

Please select a workset for analysis (required)
Select workset
INCLUDE public worksets

Please provide the number of tokens to be displayed in the tagcloud (default: 200)

Please provide the number of topics to be created (default: 10)
Prepare to run an algorithm

Please select a workset for analysis (required)

Select workset

Please provide the number of tokens to be displayed in the tagcloud (default: 200)

Please provide the number of topics to be created (default: 10)

Submit
Choose workset(s) for analysis

Check box to include public worksets first
Prepare to run an algorithm
Set the number of topics

Job Name (required)
TestJobName

Please select a workset for analysis (required)
poli_science_DD@eleanordickson

Please provide the number of tokens to be displayed in the tagcloud (default: 200)
200

Please provide the number of topics to be created (default: 10)
5

Submit
Run the analysis

<table>
<thead>
<tr>
<th>Job Name</th>
<th>Algorithm</th>
<th>Last Updated</th>
<th>Status</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test.JobName</td>
<td>Meandre_Topic_Modeling</td>
<td>2017-10-05 17:41:11</td>
<td>Staging</td>
<td></td>
</tr>
</tbody>
</table>

Completed Jobs:

<table>
<thead>
<tr>
<th>Job Name</th>
<th>Algorithm</th>
<th>Last Updated</th>
<th>Status</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Showing 1 to 10 of 1 entries
View results

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>num_topics</td>
<td>5</td>
</tr>
<tr>
<td>input_collection</td>
<td>poli_science_DDRF@eleanordickson</td>
</tr>
<tr>
<td>n_top_tokens</td>
<td>200</td>
</tr>
</tbody>
</table>

Output:
- topic_tagclouds.html
- topic_top_words.xml
- stdout.txt
- stderr.txt

Time, good, make, country, nation, today, great, years, put, give, American, months, n't, remarks, special, report, policy, members, message, law, time, world, peace, government.
Topics visualized
Topics in XML file

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<topicTopWords>
  <topic id="0">
    <word weight="19879">_</word>
    <word weight="7383">remarks</word>
    <word weight="5629">_</word>
    <word weight="5395">members</word>
    <word weight="4685">!</word>
    <word weight="4547">message</word>
    <word weight="4058">report</word>
    <word weight="3517">law</word>
    <word weight="2968">statement</word>
    <word weight="2952">special</word>
    <word weight="2836">member</word>
    <word weight="2737">meetings</word>
    <word weight="2592">met</word>
    <word weight="2583">agencies</word>
    <word weight="2423">announced</word>
    <word weight="2287">order</word>
    <word weight="2135">information</word>
    <word weight="2099">public</word>
    <word weight="2083">policy</word>
  </topic>
  <topic id="1">
    <word weight="7925">time</word>
    <word weight="7426">people</word>
    <word weight="6798">good</word>
    <word weight="6618">make</word>
  </topic>
</topicTopWords>
```
Topics listed

**topic id = 0**
_,remarks,___, members, ], message, report, law, statement, special, member, meeting, met, agencies, announced, order, information, public, policy

**topic id = 1**
time, people, good, make, don’t, ’s, made, country, question, n’t, give, months, put, government, back, american, year, point, problem

**topic id = 2**
people, great, country, years, world, today, american, nation, peace, time, life, opportunity, future, government, first, spoke, strong, day, work

**topic id = 3**
countries, nations, economic, world, international, peace, policy, security, military, continue, agreement, cooperation, progress, relations, nuclear, efforts, problems, foreign, defense

**topic id = 4**
energy, year, program, programs, oil, tax, percent, increase, local, system, budget, bill, legislation, provide, health, economy, jobs, private, work
Analyzing results

- What would you name these topics?
- Are you skeptical of any of the results?
- Did you learn anything new from the topics produced?
Case Study: Inside the Creativity Boom

- Before making his Creativity Corpus, Sam experimented with the HTRC topic modeling algorithm.
- His practice HTRC workset included public domain texts from 1950 to present.
  - Creativ* in the title
Case Study: *Inside the Creativity Boom*

Are these good topics?
Tips for topic modeling

- Treat topic modeling as step in analysis
- Be familiar with input text
- Examine results to see if they make sense
- Know that stop words can shape results
- Understand the tool
Case Study: *Inside the Creativity Boom*

- Sam ended up using HTRC Extracted Features to get the data needed to analyze contemporary material.
- The fits and starts of his project are a great real-world example!
Other pre-built tools

Depends on your goal for analysis:

- **Voyant**: easy to use, good for instruction
- **Lexos**: text cleaning, visualizations
- **AntConc**: good for concordances
- **WEKA Workbench**: aids machine learning
- **HTRC algorithms**: good for using HT text specifically
Discussion

- To what kinds of researchers on your campus would you recommend pre-built text analysis tools?

- Do you have any techniques for introducing these tools that have worked well in the past?

- If you have not taught digital scholarship tools, what techniques appeal most to you at this point?
Questions?
Module 4.2: Performing Text Analysis

Basic Approaches with Python
In this module we’ll…

- Review text analysis strategies for advanced researchers
  - Make skill-appropriate recommendations
- Explore text analysis methods in-depth
  - Understand the kinds of research available in the field
- Use an HTRC dataset to conduct exploratory data analysis
  - Practice programming skills for data-driven research
- See how Sam analyzed his Creativity Corpus
  - Learn how a researcher used text analysis methods
Where we’ll end up

Use a Python library to create a list and a graph of top adjectives in a volume.
Key approaches to text analysis

Broad Area: Natural Language Processing

Using computers to understand the meaning, relationships, and semantics within human-language text

• Specific Methods:
  • **Named entity extraction**: what names of people, places, and organizations are in the text?
  • **Sentiment analysis**: what emotions are present in the text?
  • **Stylometry**: what can we learn from measuring features of style?
Key approaches to text analysis

Broad Area: Machine Learning

Training computers to recognize patterns.

• Specific Methods
  • Topic modeling – What thematic topics are present in the text?
  • Naïve Bayes classification – Which of the categories that I have named does the text belong to?
### Activity: Identify the method

<table>
<thead>
<tr>
<th>Broad area</th>
<th>Specific method</th>
</tr>
</thead>
<tbody>
<tr>
<td>'Rowling and &quot;Galbraith&quot;: an authorial analysis</td>
<td></td>
</tr>
<tr>
<td>Significant Themes in 19th Century Literature</td>
<td></td>
</tr>
<tr>
<td>The Emergence of Literary Diction</td>
<td></td>
</tr>
</tbody>
</table>

**Note:**

**Broad areas/specific methods are those defined in the previous two slides**

**Link to project summaries:** [http://go.illinois.edu/ddrf-research-examples](http://go.illinois.edu/ddrf-research-examples)
Text analysis workflow

RAW TEXT
▼
TEXT PREPARATION
▼
TRANSLATION INTO FEATURES
▼
ALGORITHMIC USE

(you are here)
Features in the HTRC

- HTRC Extracted Features dataset
- Downloadable
- Structured data consisting of features
- 5 billion pages, in 13.6 million volumes

https://analytics.hathitrust.org/datasets#ef
HTRC Extracted Features (EF)

- The features are
  - Selected data and metadata
  - Extracted from raw text
- Position the researcher to begin analysis
  - Some of the preprocessing is already done
- Form of non-consumptive access
Per-volume features

- Pulled from bibliographic metadata
- Title
- Author
- Language
- Identifiers

```json
1 {  
  "id":"uc1.b3419888",  
  "metadata":{  
    "schemaVersion":"1.2",  
    "dateCreated":"2015-02-12T13:30",  
    "title":"Zoonomia = or The laws of organic life / by Erasmus Darwin.",  
    "pubDate":"1809",  
    "language":"eng",  
    "htBibUrl":"http://catalog.hathitrust.org/api/volumes/full/htid/uc1.b3419888.json",  
    "handleUrl":"http://hdl.handle.net/2027/uc1.b3419888",  
    "oclc":"3679915",  
    "imprint":"Thomas and Andrews, 1809."  
  },  
  "features":{  
    "schemaVersion":"2.0",  
    "dateCreated":"2015-02-20T23:58",  
    "pageCount":616,  
    "pages":[]  
  }  
}
```
Per-page features

- Page sequence
- Computationally-inferred metadata
  - Word, line, and sentence counts
  - Empty line count
  - Language

```json
{  
  "seq": "00000035",
  "tokenCount": 507,
  "lineCount": 44,
  "emptyLineCount": 0,
  "sentenceCount": 14,
  "languages": [  
    {  
      "en": "1.00"
    }
  ]
}
```
development. We hope to find an answer within the next few days, the next week, so that the Congress and the President can work together, not at odds. What I am saying to you is that despite political differences—and there are some—if we are going to continue to be a great country—and I am optimistic that we will—you have to find a way to disagree without being disagreeable. You have to find a way to solve a problem with no easy losing face and everybody doing a job for the country. And the experiences you are having right now at this present time—that is a training ground for the time when all of you have an opportunity at the local, the State, or the Federal level to come down and be an active participant.

A long time ago, back when the ball was round, I played a little football for the University of Michigan—[laughter]—and that is the truth, it was round, and none of these older fellows can remember it here.

But anyhow, you know in those days we had some other problems. But by working together, the American people finally found a way to solve some of them. And some of us and others in our vintage found an inspiration to come here and to be part of the Congress—House, Senate—and to be a part of the execu
tive branch of the Government. And that is what we need from all of you—that drive, that stimulation to be a part of your Government.

And I am absolutely convinced that, as I look around here, you have got all the talent, all the enthusiasm. We are not going to solve all the problems—my generation—that we are building slowly into a better America.

But you, because of your better education, better opportunity, and all the other things that bless us in this country, can take what we build and make it the kind of America that we dream about and hope for. And that is the message I would like to leave with you from the Rose Garden and the White House.

Thank you very, very much.

James M. Watkins (national commander, American Legion): Thank you very much, Mr. President, for those very inspiring words to those young people who are here with us this morning.

Mr. President, you will recall a day this past December, at the Alexandria railroad depot, when you launched the Bicentennial American Freedom Train as its historic 21-month journey throughout the United States. About the Freedom Train is the American Legion's Freedom Bell, a bell twice the size of the second Liberty Bell. But unlike the Liberty Bell, our bell has no crack in it and is perfectly capable. Mr. President, of ringing loud and clear to remind Americans now and in the future of their precious liberties. To that end, American Legionnaires and their Auxiliary throughout the

Public papers of the presidents of the United States (Gerald R. Ford, book 2)
LIST OF ITEMS ................................................ vii
CABINET .................................................... lxvii
PUBLIC PAPERS OF GERALD R. FORD, JULY 21—DECEMBER 31, 1975 ........................................ 1005
Appendix A—Additional White House Releases .......................................................... 2021
Appendix B—Presidential Documents Published in the Federal Register ................................. 2049
Appendix C—Presidential Reports to the 94th Congress, 1st Session ................................. 2057
Appendix D—Rules Governing This Publication ......................................................... 2061
INDEX .......................................................... A–1

Public papers of the presidents of the United States (Gerald R. Ford, book 2)
Page section features

Header, body, footer

- Line, empty line, and sentence count
- Counts of beginning- and end-line characters
- Token counts
  - Homonyms counted separately
  - Part-of-speech codes are from the Penn Tree Bank

```json
"body": {
  "tokenCount": 504,
  "lineCount": 43,
  "emptyLineCount": 0,
  "sentenceCount": 12,
  "tokenPosCount": {
    "synthetic": {"NNP": 1},
    "Laws": {"NNP": 1},
    "beautiful": {"JJ": 1},
    "philosopher": {"NN": 1},
    "uponthe": {"IN": 1},
    "for": {"IN": 1}
  }
}
```
Using HTRC Extracted Features

- Identify parts of a book
  - From descriptive metadata
- Perform any method that works with bags-of-words
  - Topic modeling
  - Dunning’s log-likelihood
- Classify volumes
  - Compare with bibliographic metadata
Do-it-yourself text analysis

- Some researchers won’t use off-the-shelf tools
  - Want more control over processes
  - Set their own parameters

- Mix-and-match approaches to create “toolkit” of strategies
The toolkit

- Researcher-dependent
- Requires understanding of statistics
- Often draws on expert collaborators
- Consists of command line tools and programming languages
Command-line tools

- Tools you download and run from the command line
  - MALLET (Java-based)
    - Topic modeling and classification
  - Stanford NLP (Java-based)
    - Natural language processing
Code libraries

- Part of programming is re-using code, often shared as **libraries**
- The existing code is installed via **packages**
- Packages are collections of bits of code, made up of **modules**
- Modules facilitate programming tasks
Using libraries in practice

BenM
@benmiller314

me on python: spend 2 hrs working out how to do X. discover library designed to do X better. spend 3 hrs wading into that library. repeat.

Likes
5

12:54 PM - 20 Jun 2017
Installing packages for Python

- Install using a package manager

- Example: pip
  - Package manager for Python
  - Generally comes with Python install

- Basic syntax
  - pip install <name>

- Other installers: Homebrew, Conda
Using a library’s module in a script

- Once library is installed...
- Must declare that it will be used at the top of the script
  - “Import” the module
- Use the desired module in the body of the script

```python
from htrc_features import FeatureReader
import pandas as pd
import sys

filename = sys.argv[1]
idx = pd.IndexSlice

def get_proper_nouns(vol):
    tl = vol.tokenlist(pages=False)
    tl.index = tl.index.droplevel(0)
    try:
        proper_nouns = tl.loc[idx[:, :, ('VB', 'VBD', 'VBG', 'VBN', 'VBP', 'VBZ')],]
        proper_nouns.index = proper_nouns.index.droplevel(2)
        return proper_nouns[proper_nouns['count'] < 150]
    except:
        return pd.DataFrame()
```

Imports modules, i.e. the components of the package

Use the module
Python text analysis libraries

- Machine learning
  - SciKit Learn
- Data science
  - Pandas
- Natural language processing
  - Natural Language Toolkit (NLTK)

**Example:** `nltk.words_tokenize()`
  - *Can you guess what this does?*
HTRC Feature Reader Python library

- Python library for working with HTRC Extracted Features
  - Bits of pre-written code to parse the JSON
- Install using a package manager, like pip
  - Source code lives on Github
- Also need to have Pandas installed
  - Remember: pandas = Python library for working with data
  - Luckily it’s included in PythonAnywhere!
Sample Reference Question

I’m a student in history who would like to incorporate digital methods into my research. I study American politics, and in particular I’d like to examine how concepts such as liberty change over time.

Approach: view adjectives in Extracted Features file
Hands-on activity

In this activity, you will install the Feature Reader library and run a Python script to create a list of the most-used adjectives and the number of times they occur in different sets of volumes of presidential papers (using the extracted features files of these volumes).

What You Need:

Script: top_adjectives.py

Files for analysis, in 2 directories:

- 1970
- 1930
Files and directories

- 1930 and 1970 directories and required files should already be in your files in PythonAnywhere
- Check using the “Browse files” button
- 1970 = Extracted Features files of presidential speeches from the 1970s (corresponds to our political science workset!)
- 1930= Extracted Features files of presidential speeches from the 1930s
Install Feature Reader library

```
Bash console 4179031
19:56 ~ $ pip install --user htrc-feature-reader
```

```
pip install --user htrc-feature-reader
```
Examine code

```python
from htrc_features import FeatureReader
import glob
import pandas as pd
import sys

decade = sys.argv[1]
paths = glob.glob(decade+'/*.json.bz2')
idx = pd.IndexSlice
fr = FeatureReader(paths)
vol = next(fr.volumes())

vol = vol.tokenlist(pages=False)

adjectives = tl.loc[idx[:,('JJ')],]
adj_dfs = [adjectives for vol in fr.volumes()]
all_adj = pd.concat(adj_dfs).groupby(level='token').sum().sort('count', ascending=False)[:50]

print(all_adj)
```

Prepares us to use the libraries needed for this task.

‘JJ’ is the code for adjectives from the Penn TreeBank.
Open shell and run command

```
13:45 ~ $ python top_adjectives.py 1970
```

```
python top_adjectives.py 1970
```
View results

```
13:45 ~ $ python top_adjectives.py 1970

<table>
<thead>
<tr>
<th>token</th>
<th>count</th>
</tr>
</thead>
<tbody>
<tr>
<td>token</td>
<td>21248</td>
</tr>
<tr>
<td>_</td>
<td>17824</td>
</tr>
<tr>
<td>other</td>
<td>11552</td>
</tr>
<tr>
<td>own</td>
<td>10400</td>
</tr>
<tr>
<td>new</td>
<td>9984</td>
</tr>
<tr>
<td>good</td>
<td>9440</td>
</tr>
<tr>
<td>great</td>
<td>7936</td>
</tr>
<tr>
<td>many</td>
<td>7840</td>
</tr>
<tr>
<td>major</td>
<td>6480</td>
</tr>
<tr>
<td>last</td>
<td>6288</td>
</tr>
<tr>
<td>public</td>
<td>6224</td>
</tr>
<tr>
<td>important</td>
<td>6096</td>
</tr>
<tr>
<td>first</td>
<td>5808</td>
</tr>
<tr>
<td>such</td>
<td>5760</td>
</tr>
<tr>
<td>economic</td>
<td>5504</td>
</tr>
<tr>
<td>human</td>
<td>5424</td>
</tr>
<tr>
<td>international</td>
<td>5216</td>
</tr>
<tr>
<td>national</td>
<td>4864</td>
</tr>
<tr>
<td>same</td>
<td>4688</td>
</tr>
<tr>
<td>next</td>
<td>3920</td>
</tr>
<tr>
<td>nuclear</td>
<td>3840</td>
</tr>
<tr>
<td>local</td>
<td>3744</td>
</tr>
<tr>
<td>foreign</td>
<td>3696</td>
</tr>
<tr>
<td>political</td>
<td>3648</td>
</tr>
<tr>
<td>comprehensive</td>
<td>3488</td>
</tr>
<tr>
<td>few</td>
<td>3472</td>
</tr>
<tr>
<td>sure</td>
<td>3440</td>
</tr>
<tr>
<td>possible</td>
<td>3408</td>
</tr>
</tbody>
</table>
```
On your own

- Compare the adjectives used by presidents in the 1970s with those used in the 1930s.
- Work and discuss with your neighbor.

- How do you need to change your command?
- What differences do you see? Similarities?

Challenge

Try modifying the script to search verbs or another part of speech.
(Hint: Don’t forget the Penn Tree Bank!)
Exploratory data analysis

- Approach for getting familiar with data
- Easier to recognize patterns (and problems)
  - Hard to see trends in a spreadsheet or text file!
- There are whole books about it

- Strategies:
  - Plot raw data
  - Plot simple statistics
  - Compare plots to look for patterns
Visualization libraries

- **Python**
  - Matplotlib `pyplot()` function
  - `ggplot`

- **Others:**
  - R: `ggplot2`
  - D3.js: JavaScript library for visualizations
Sample Reference Question

I'm a student in history who would like to incorporate digital methods into my research. I study American politics, and in particular I’d like to examine how concepts such as liberty change over time.

Approach:

View word count over a volume using an Extracted Features file
Hands-on activity

- In this activity, you will create a visualization using the native plotting functionality in Pandas, called pyplot, to see the word count over a volume based on its Extracted Features file.

Script: word_count.py

Files for analysis: mdp.49015002203033.json.bz2 (*Presidential Papers of the Presidents of the United States: Gerald Ford*)
Examine code

```python
from htrc_features import FeatureReader
import os
import glob
import pandas as pd
import matplotlib
import matplotlib.pyplot as plt
matplotlib.use("Agg")

path = glob.glob('1970/mcp.49015002203033.json.bz2')
fr = FeatureReader(path)
vol = fr.first()
tokens = vol.tokens_per_page()
tokens.plot()
plt.savefig("words.png")
```

- Imports our modules so we can call functions later, including matplotlib's pyplot
- Set variable "tokens"
- Plot tokens and save plot as png
Open shell and run command

```
12:28 ~ $ python word_count.py
```
The graph visualizes how many words appear on each page in the volume.

Page numbers are on the horizontal axis, and the word count is on the vertical axis.

E.g. The 600th page has about 350 words.
On your own

- Can you modify the script to look at another volume?
  - Edit, save, and run the file
- View and discuss with your neighbor

☞ See Handout P20
Another look

- Even basic counts can lead to sophisticated research
- Ted used HTRC EF
- Classified genre at the page level for public domain works

From paper - graph of diction patterns between genres, using frequency counts (Underwood and Sellers, 2012)
Case Study: Inside the Creativity Boom

After reducing Creativity Corpus to pages containing forms of creativ*: 

- Performed topic modeling on those pages
- Ended up with topics that reflect what themes are prevalent around concept of “creativity” in the 20th century
- Graphed the topics over time to see how their usage changed
Case Study: *Inside the Creativity Boom*

- Topics that decreased in usage over time
  - god, christ, jesus, creation, word
  - species, animals, natural, plants, soil
  - nature, mind, creative, world, human
  - invention, power, creative, own, ideas
Case Study: *Inside the Creativity Boom*

- Topics that increased in usage over time

  - advertising, media, marketing, sales, television
  - economic, development, capital, economy, production
  - poetry, language, poet, poets, poems
  - social, creative, study, development, behavior

![Creativity topics with increasing usage](image)
Discussion

- *In what ways can librarians support advanced text analysis research?*
- *What additional skills would you need to learn in order to do so?*
Questions?
References

Module 5: Visualizing Textual Data

An Introduction
In this module we’ll…

- Introduce common visualization strategies for text data
  - Communicate with researchers about their options
- Use a web-based visualization tool, HathiTrust+Bookworm
  - Gain experience creating and reading data visualizations
- See how Sam used HathiTrust+Bookworm for his project
  - Learn how HT+BW was utilized in research
Where we’ll end up

Create visualization of word usage trends across the HathiTrust corpus.
Data visualization

- Data visualization is the process of converting data sources into a visual representation.
- Visualizations present particular ways of interpreting data.
- Data visualization is an entire field of study; we’re barely scratching the surface.
Why visualize text data?

- Understand broader themes of a dataset
- Explore patterns in the data
- Cluster texts for overview or classification
- Compare data to other data (e.g., correlating with social networks)

Adapted from Jason Chuang’s Text Visualization course at Stanford University
Place in research process

- In the earlier exploration stage of a project:
  - Explore full range of data
  - Discover characteristics and themes in data

- In the later explanation stage of a project:
  - Communicate findings to others in a clearer and more efficient way
Common text data visualizations

Word cloud

- Relatively unsophisticated, but effective
- Size of word relates to prominence or salience

Topic models from HTRC Algorithms
Common text data visualizations

Trees or hierarchies

- Word trees

Occurrences of “I have a dream” in Martin Luther King’s historical speech.
(Wattenberg and Viégas, 2008)
Common text data visualizations

Networks

- Node-link diagrams
- Good for representing topic models
- Visualize connections between named entities

Topic model of English books, 1850-1899
(1850-1899)
(Underwood, 2012)
Common text data visualizations

Temporal- or spatial-based visualizations

- Temporal visualizations

Percent representation of female characters in English literature (Underwood and Bamman, 2016)

Common text data visualizations

Temporal or spatial visualizations

- Maps

Percent of newspaper pages containing the term “hoosier” (Palmer, Polley, & Pollock, n.d.)
http://centerfordigschol.github.io/chroniclinghoosier/map1.html
Common text data visualizations

Other “multi-dimensional” visualizations

- Bubble charts
- Heat maps

Common text data visualizations

Other “multi-dimensional” visualizations

- Heat maps

Heatmap of MARC cataloging at the Library of Congress by book year and cataloging year (Schmidt, 2017)
http://sappingattention.blogspot.com/2017/05/a-brief-visual-history-of-marc.html
Activity

Match type of use to the type of visualization:

<table>
<thead>
<tr>
<th>Visualization</th>
<th>What would it be good for?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Word cloud</td>
<td></td>
</tr>
<tr>
<td>Trees or hierarchies</td>
<td></td>
</tr>
<tr>
<td>Networks</td>
<td></td>
</tr>
<tr>
<td>Timeline</td>
<td></td>
</tr>
<tr>
<td>Map</td>
<td></td>
</tr>
<tr>
<td>Bubble chart</td>
<td></td>
</tr>
<tr>
<td>Heatmap</td>
<td></td>
</tr>
</tbody>
</table>

Uses

- Change over time
- Spatial
- Topical density
- Relationships
- Word distribution

** Bonus: what kinds of variables (i.e. data points) you would need for each visualization?
Common visualization tools

- **Word clouds**
  - Voyant
  - Wordle

- **Word use trends**
  - Google Books Ngram Viewer
  - HathiTrust+Bookworm

- **Tabular data visualization**
  - Tableau

- **Mapping**
  - ArcGIS Online with StoryMaps
  - Tableau

- **Network graphs**
  - Gephi
  - NodeXL
  - DH Press
Common visualization libraries

- **Python**
  - matplotlib, pyplot
  - ggplot library

- **R**
  - ggplot2

- **D3.js**
  - Javascript library for visualizations
Review: key terms in text analysis

**N-gram**

A contiguous chain of n items from a sequence of text where n is the number of items. Example: Bigram.

four score, score and, and seven, seven years, years ago, ago our, our fathers, fathers brought, brought forth, forth on, on this, this continent, continent a, a new, new nation, nation conceived, conceived in, in liberty, liberty and…
N-gram visualization: HathiTrust + Bookworm

Brings together:

- Text data (unigrams)
- Bibliographic metadata
- Visualization tool
- Track trends in a repository
Bookworm framework

- Visualizes categories
- The category is plotted along the x-axis
  - Often plot years along the x-axis
  - Can plot other things!
- HathiTrust+Bookworm is just one implementation of the framework

Adapted from Ben Schmidt, “Bookworm API Philosophy”
Track social change: lady vs. woman over time
Reading an HT+BW graph

- Let’s look at how verbs change over time
  - Eg. Burned vs. burnt

Do you see any trends?
Bookworm interface

Limit your search with facets

https://bookworm.htrc.illinois.edu/develop
Bookworm interface

Fine-tune your results
Bookworm interface

Links directly to texts in the HTDL
Sample Reference Question

I’m a student in history who would like to incorporate digital methods into my research. I study American politics, and in particular I’d like to examine how concepts such as liberty change over time.

Approach:

Explore word usage trends of political concepts within the HathiTrust using HT+BW
Hands-on activity

- In this activity, you will use HT+BW to explore lexical trends

Website: https://bookworm.htrc.illinois.edu/develop
Examples
Examples
Discussion

- What trends did you discover?
Case Study: *Inside the Creativity Boom*

- Sam used HT+Bookworm to visualize the use of “creative” in the HTDL over time
Case Study: *Inside the Creativity Boom*

- Sam also used an experimental HT+BW interface to create different kinds of visualizations…
Case Study: *Inside the Creativity Boom*

- “Creative” by language and year
Case Study: *Inside the Creativity Boom*

- “Creativity” by library classification and year
Discussion

- Where does visual literacy fit into data literacy overall?
- What would it mean to be visually literate, particularly with regard to text analysis?
Questions?
References


References


How’d we do?

- Link to assessment survey:
  
  go.illinois.edu/ddrf-assess
Thanks to all our team members!

- **PI**
  - Harriett Green, University of Illinois at Urbana-Champaign Library

- **Co-PIs**
  - Angela Courtney, Indiana University-Bloomington Libraries
  - J. Stephen Downie, Graduate School of Library and Information Science, University of Illinois
  - Terese Heidenwolf, Lafayette College Library
  - Geoffrey Morse, Northwestern University Libraries
  - Amanda Henley, University of North Carolina at Chapel Hill Library

- **Project Members (Past and Present)**
  - University of Illinois at Urbana-Champaign
    - Eleanor Dickson Koehl
    - Ruohua Han
    - Rachel Blomer
    - *Gabriella Stuardi
    - *Beth Sandore Namachchivaya
    - *Ryan Dubnicek
    - *Sayan Bhattacharyya
  - Lafayette College
    - Lijuan Xu
    - Ben Jahre
    - *Neil McElroy
  - University of North Carolina at Chapel Hill
    - Sarah Morris
    - *Stewart Varner
    - *Jacob Hill
  - Northwestern University
    - *Geoffrey Swindells
    - * = former members
QUESTIONS?

HTRC_workshop@library.illinois.edu

https://teach.htrc.illinois.edu

award #RE-00-15-0112-15