Using the HathiTrust Research Center's Tools for Text Analysis

All workshop-related files are at: http://bit.ly/1T3lu9B

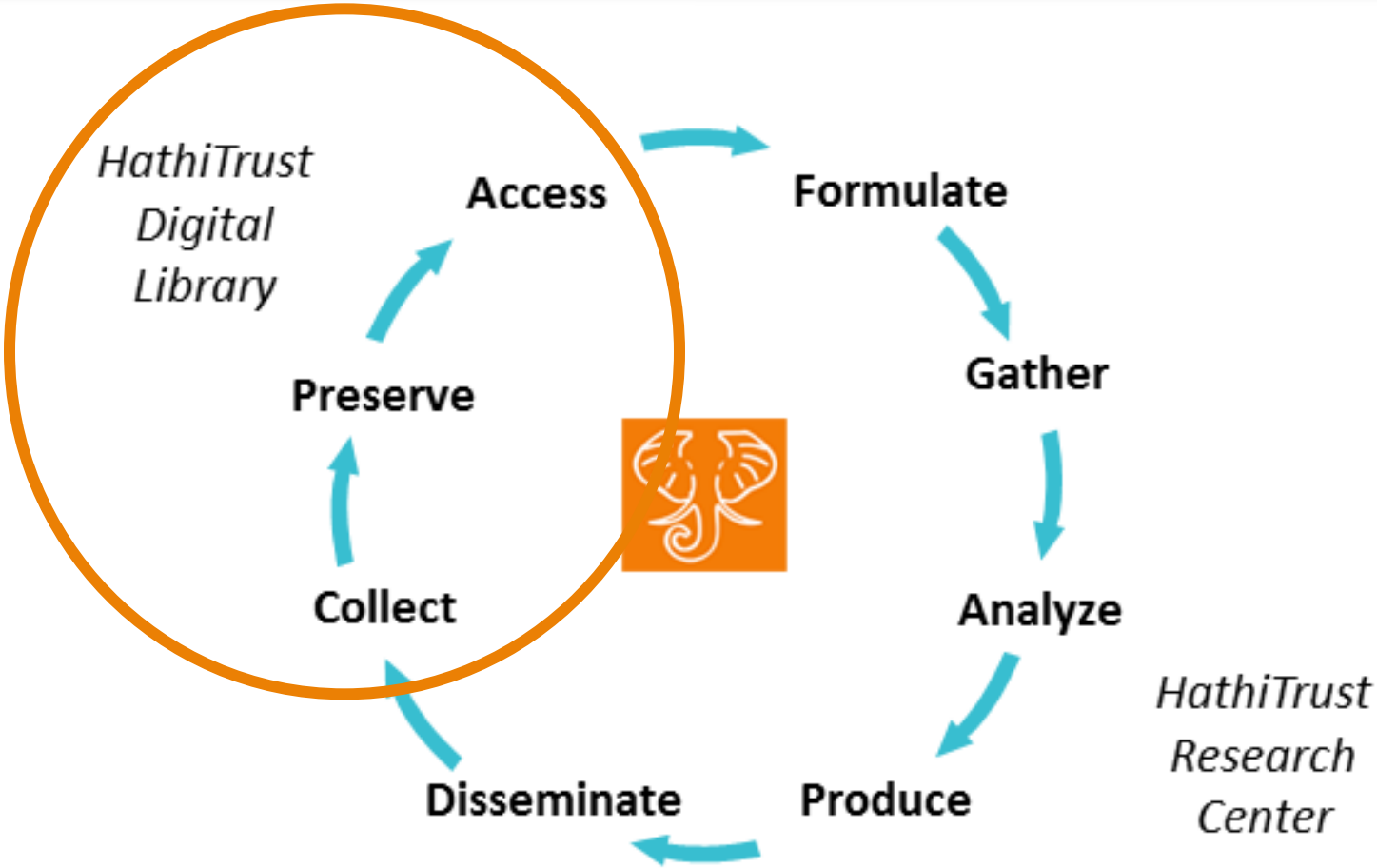
Handouts and slides will also be posted there within a few days.

http://bit.ly/1T3lu9B

Sayan Bhattacharyya
Postdoctoral Research Associate (CLIR Fellow)
Graduate School of Library and Information Science
University of Illinois at Urbana-Champaign

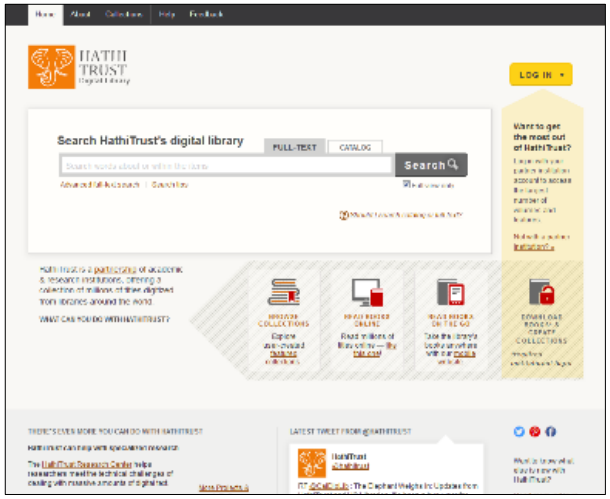
HathiTrust ecosystem

HathiTrust ecosystem

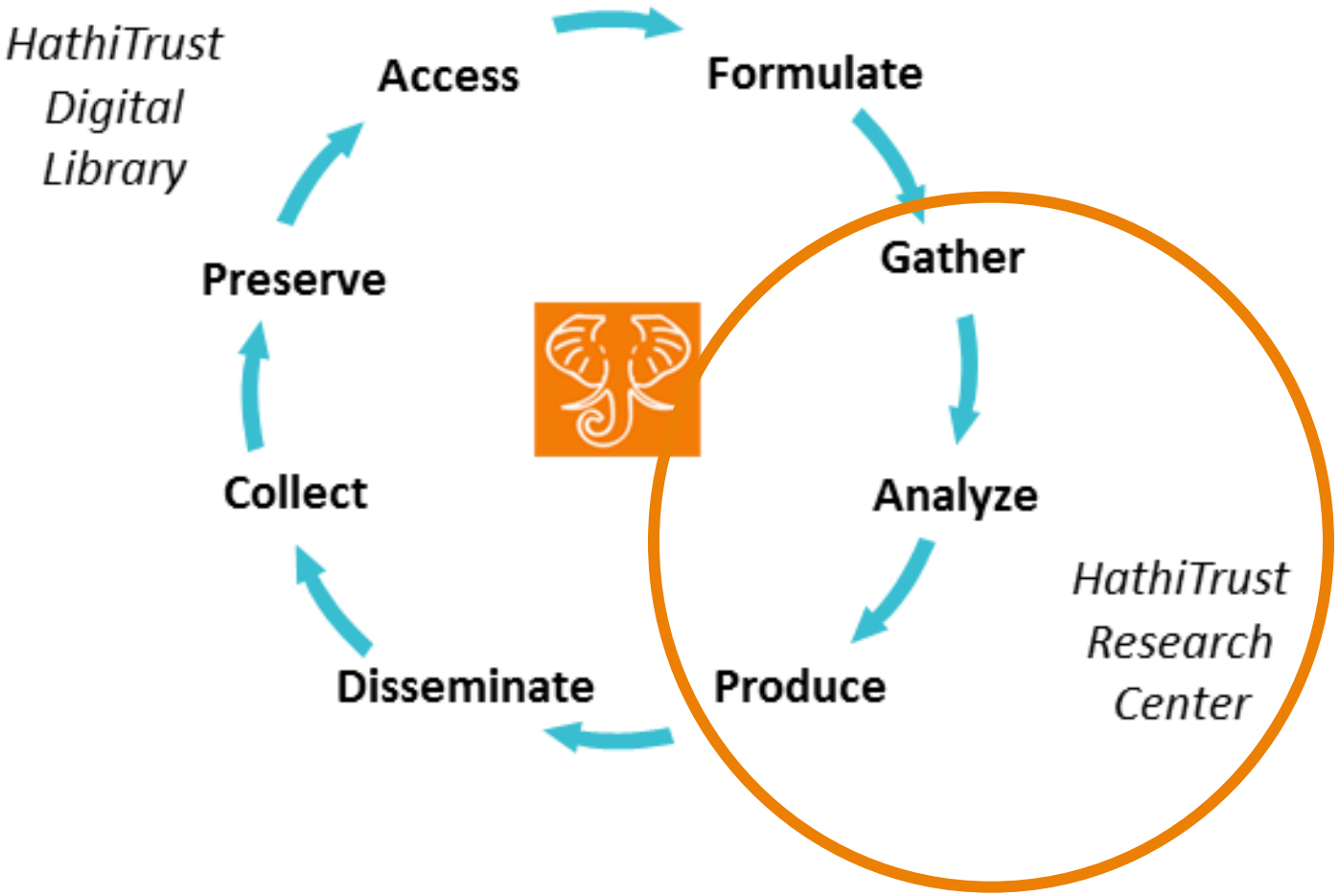


About the HathiTrust Digital Library

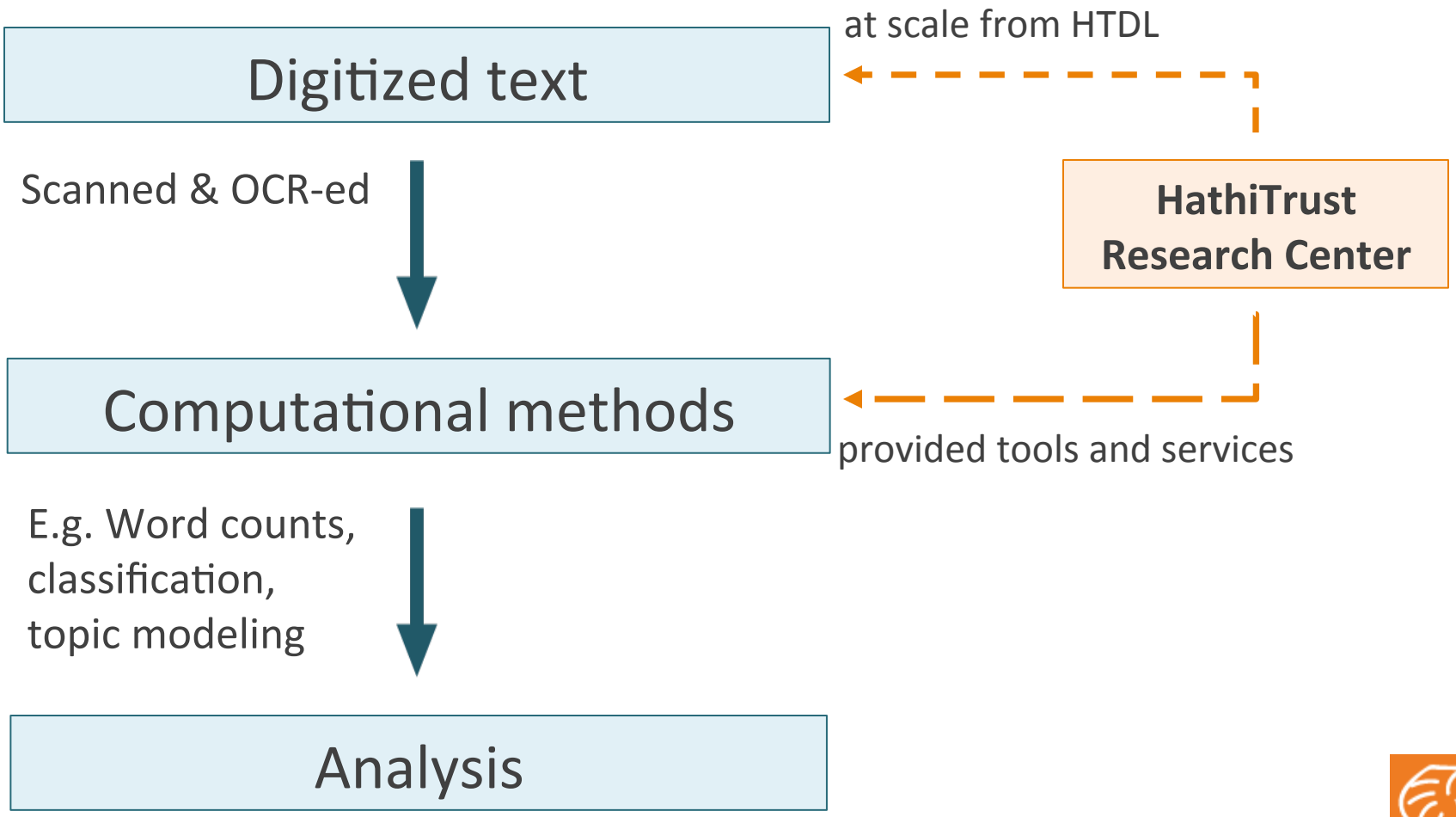
- Repository
  - 13+ million volumes | 3+ billion pages
  - 50% of volumes are in English
  - Material from the 15th C. on | 20th C. concentration
  - 70% in copyright or undetermined | 30% open
- Interface
  - Search and read books in the public domain



HathiTrust ecosystem



HTRC for Text Analysis



About the HTRC

Part content provider (digitized text)

- Early English Books Online
- JSTOR Data for Research
- *Etc…*

Part tool provider (computational methods)

- Voyant
- MALLET
- *Etc..*

We provide access to textual data from the HTDL and a suite of services to analyze it.

And more!

Non-consumptive research paradigm

Technology-aided research that allows for distant, computational analysis without permitting close, human reading.

- Complies with copyright law
  - Ideas vs Expressions
- Foundation underlying structure of HTRC work
- Other terms: non-expressive use

HT Points of Access

- HTRC Portal and Workset Builder
- HTRC Data Capsule
- HathiTrust Digital Library
  - HT data request
  - HathiFiles
  - Bibliographic and Data APIs
- HTRC datasets
  - Extracted Features dataset
  - Genre in English Language Literature (Underwood)
- HT+BW

Portal & Workset Builder

A set of tools for assembling collections of digitized text and performing text analysis on them

Using the HTRC Portal

- Create account with institutional ID
  - From non-profit institution of higher-ed
- Gateway to:
  - Workset Builder
  - Access HTRC Data Capsule
  - Extracted Features and genre datasets
- Access to:
  - Existing worksets (collections)
  - Off-the-shelf algorithms

Worksets

A workset is a user-created collection of text from the HathiTrust Digital Library.

- Textual datasets
- Built through searching OCR data and/or metadata
  - In full text and titles
  - By author, time period, subject classification, etc.
- Currently Workset Builder includes non-copyrighted material NOT digitized by Google

Why worksets?

- Worksets allow a researcher to curate subcorpora of text relevant to her
- Building research collections is familiar activity for humanists
- Worksets can be shared and cited

HTRC Algorithms

- Functions for text analysis
- Extract, refine, analyze, and visualize textual data
- Built-in to Portal
  - Some customization/optimization available, but mostly static tools
  - For users who don't require, or don't know how to write, personalized algorithms

HTRC Algorithms

Extracted Features Rsync Script Generator

MARC Downloader

Meandre **Classification** Naïve Bayes

Meandre **Dunning Log-Likelihood** to Tag Cloud

Meandre **OpenNLP Date Entities** to Simile

Meandre **OpenNLP Entities List**

Meandre **OpenNLP Report** per Volume

Meandre **Tag Cloud**

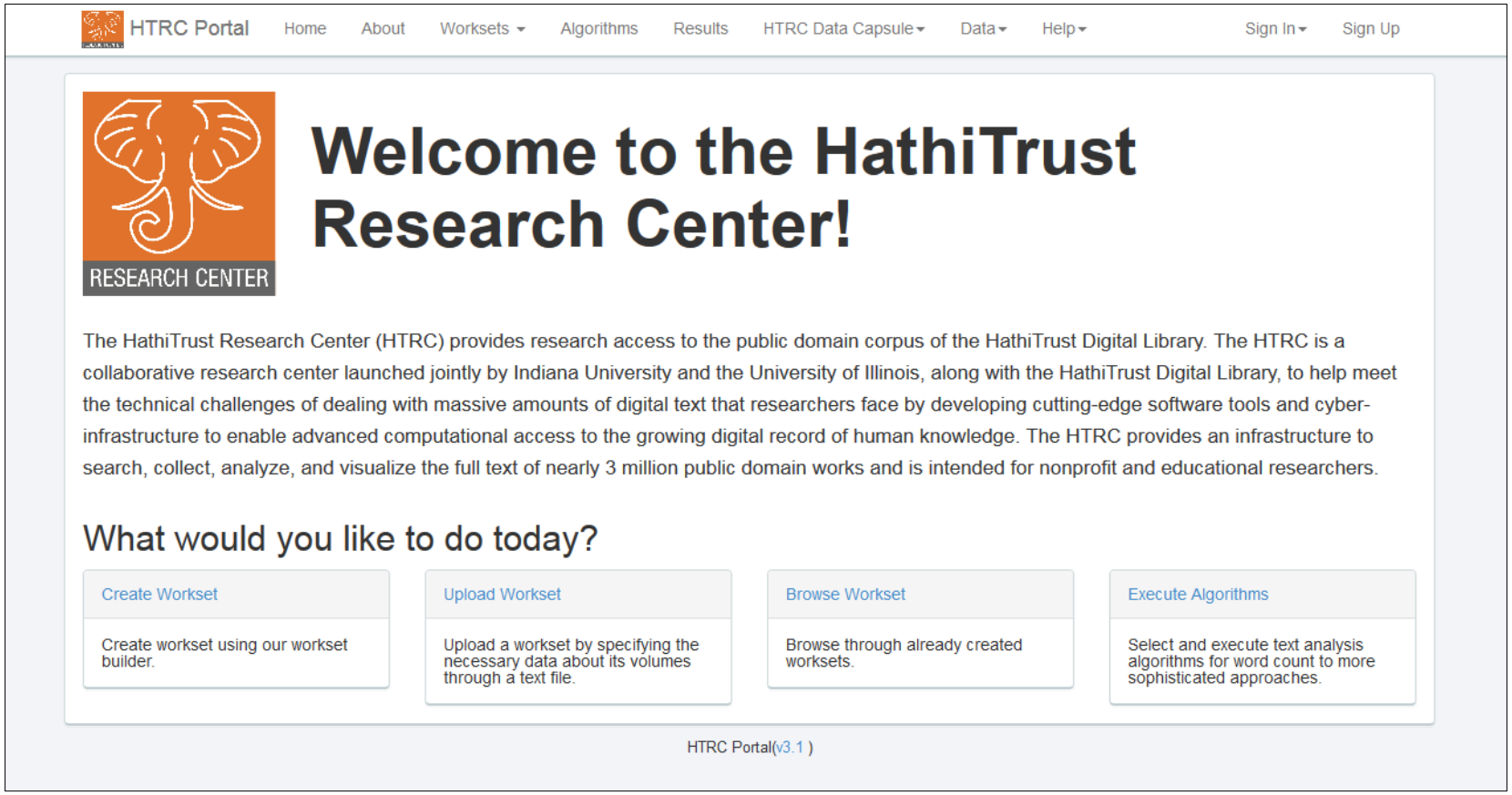
Meandre **Tag Cloud with Cleaning**

Meandre **Topic Modeling**

Simple Deployable **Word Count**

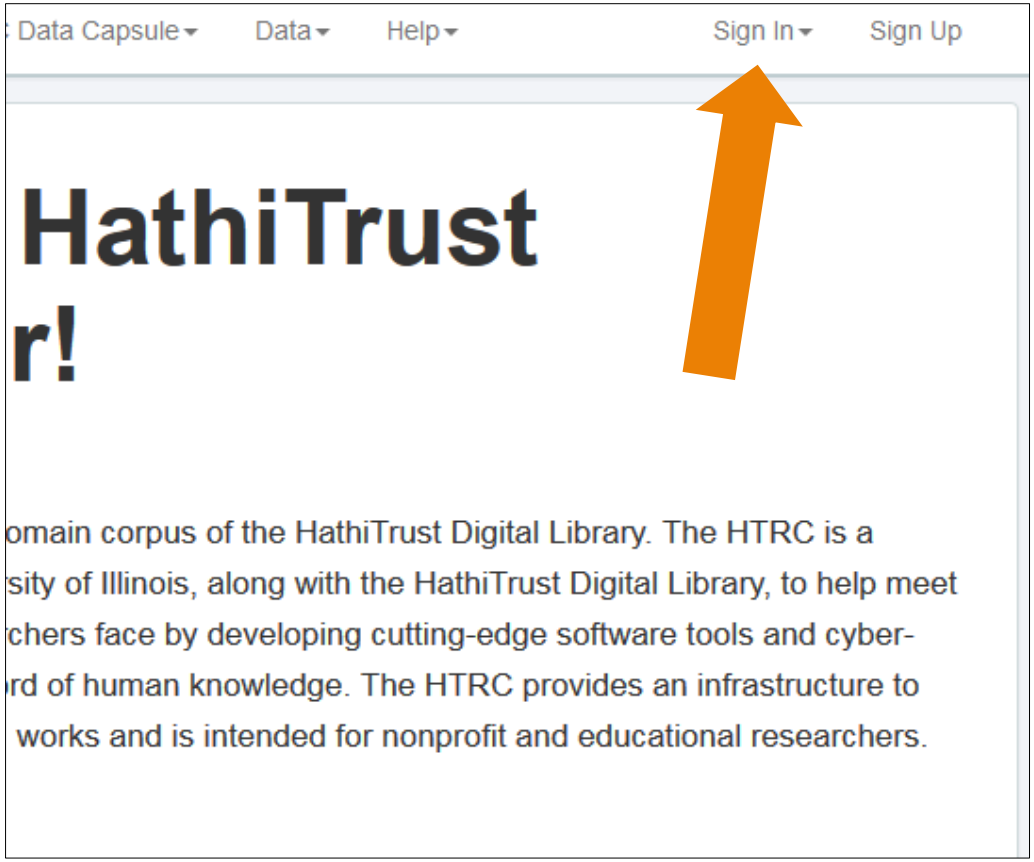
Basic Portal Workflow

HTRC Portal



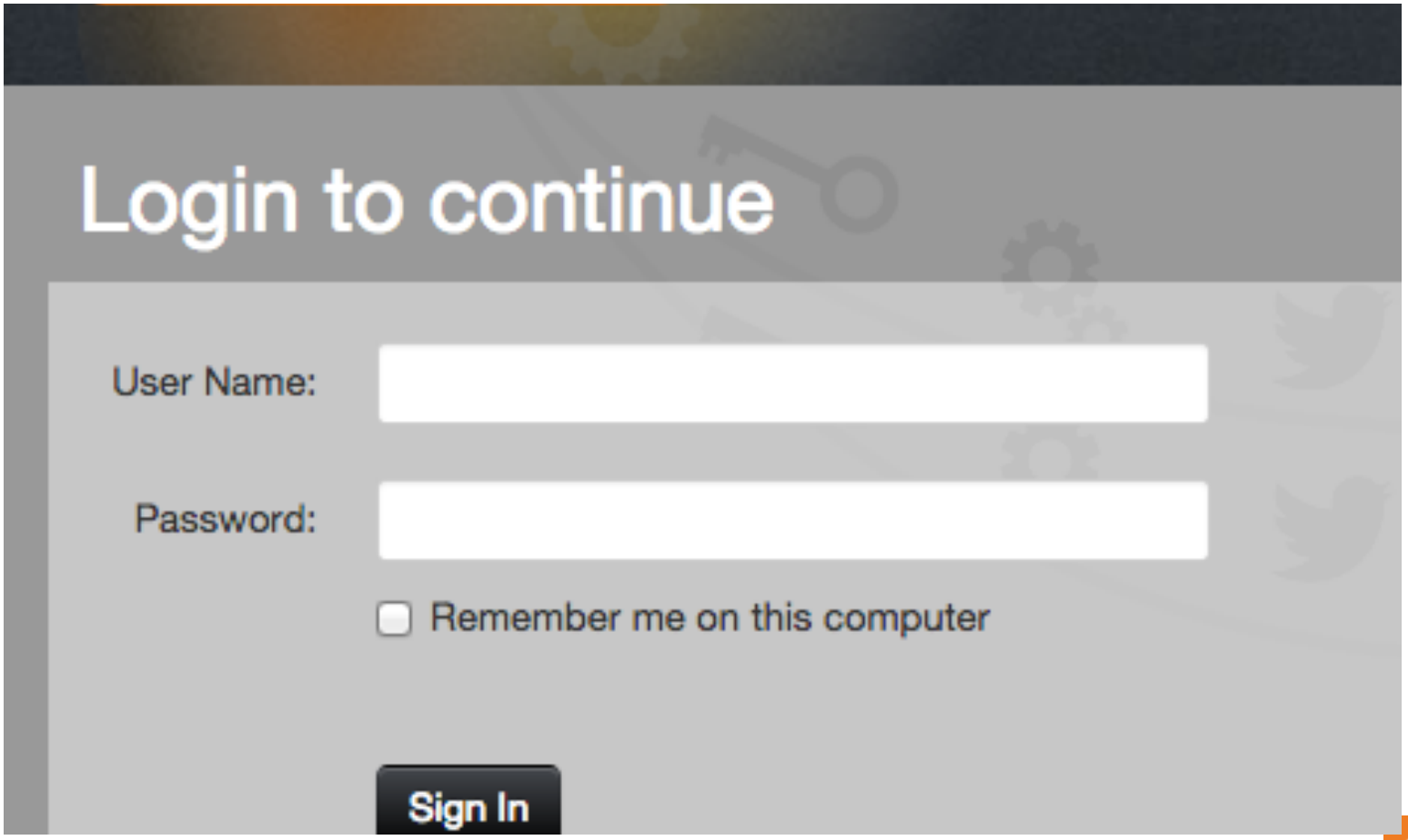
https://sharc.hathitrust.org

Sign in / Sign up

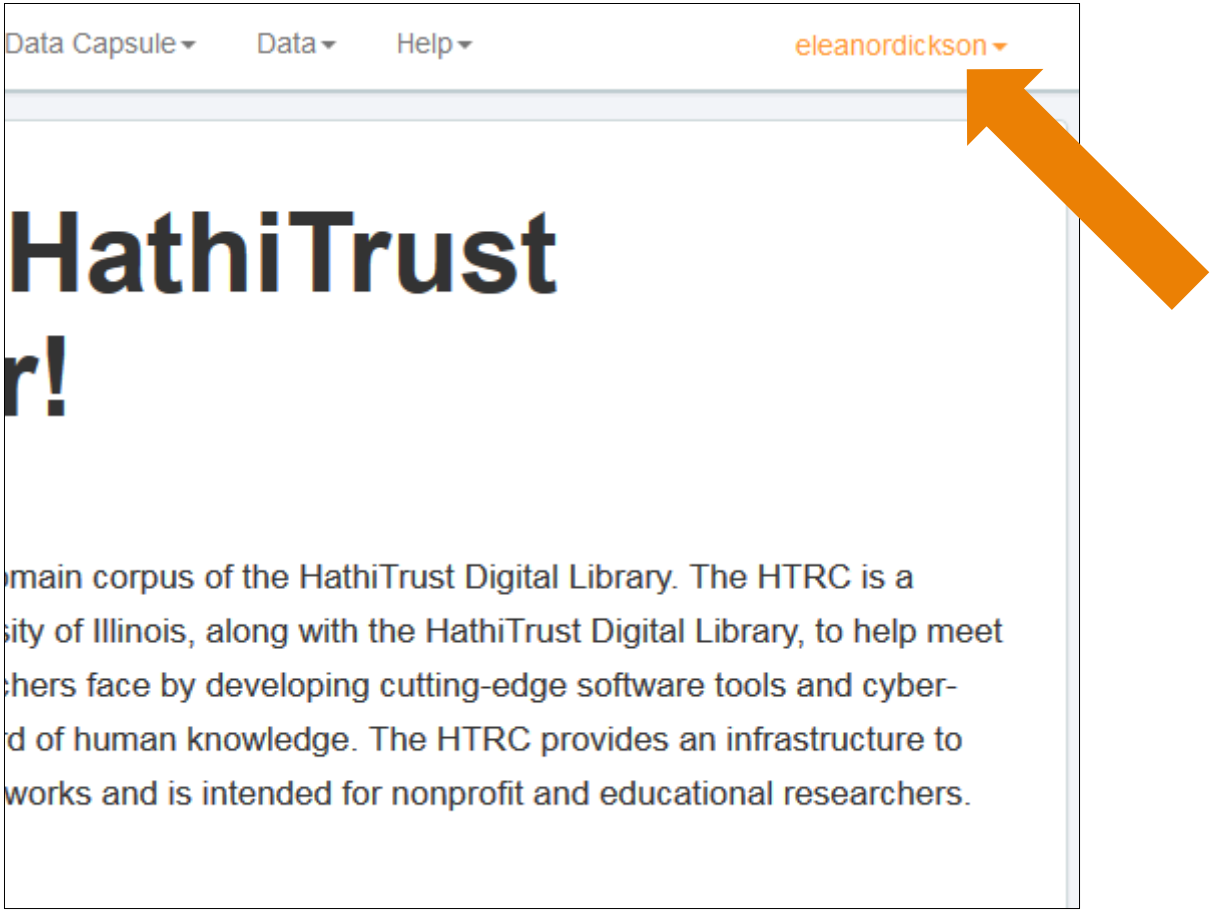


https://sharc.hathitrust.org

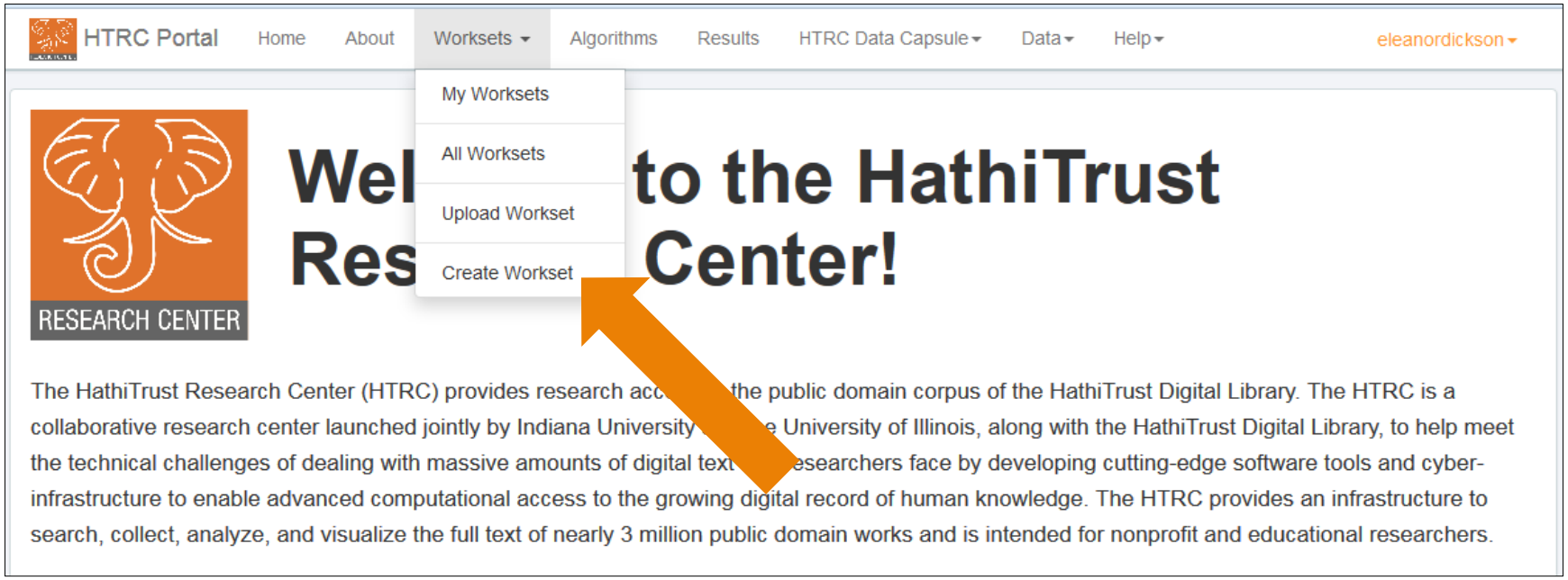
Sign in


Login to continue
User Name:
Password:
Remember me on this computer
Sign In

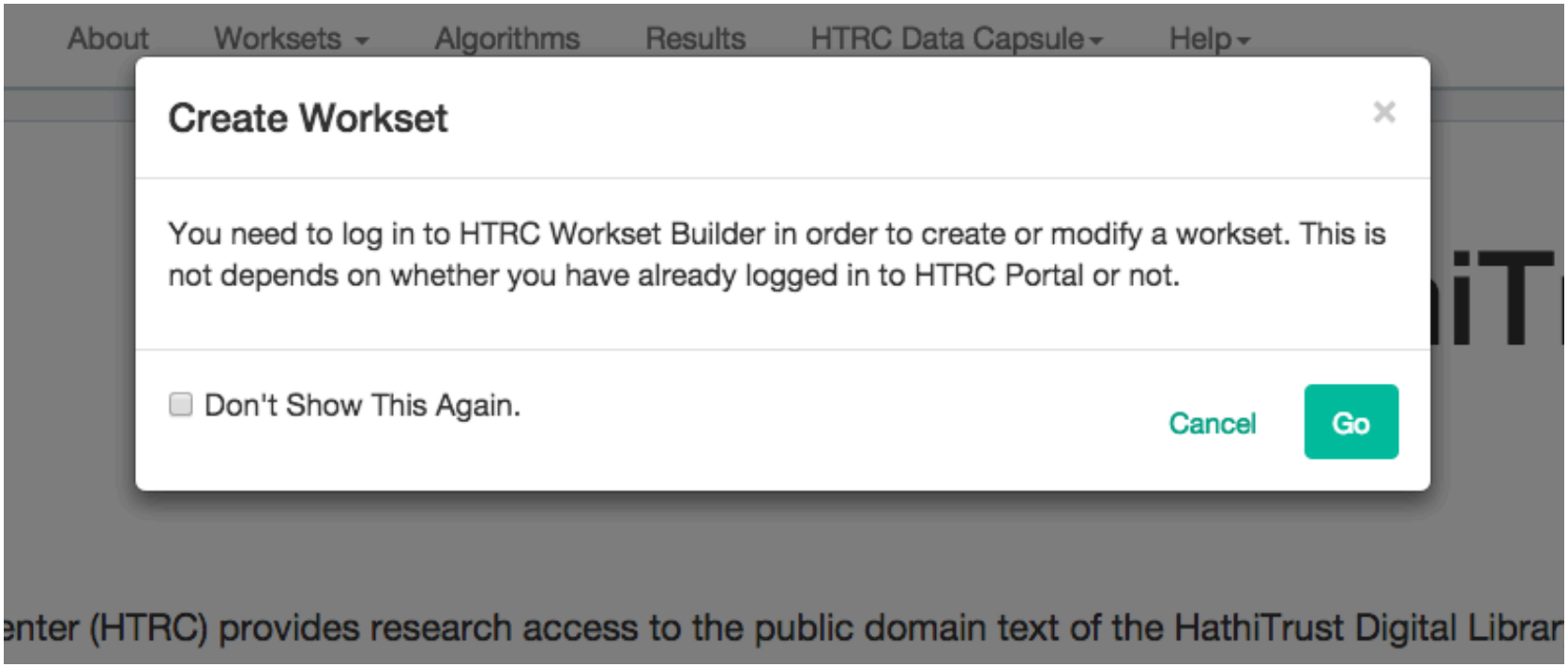
Sign in


Data Capsule
Data
Help
eleanordickson
HathiTrust
r!
main corpus of the HathiTrust Digital Library. The HTRC is a
sity of Illinois, along with the HathiTrust Digital Library, to help meet
chers face by developing cutting-edge software tools and cyber-
d of human knowledge. The HTRC provides an infrastructure to
works and is intended for nonprofit and educational researchers.

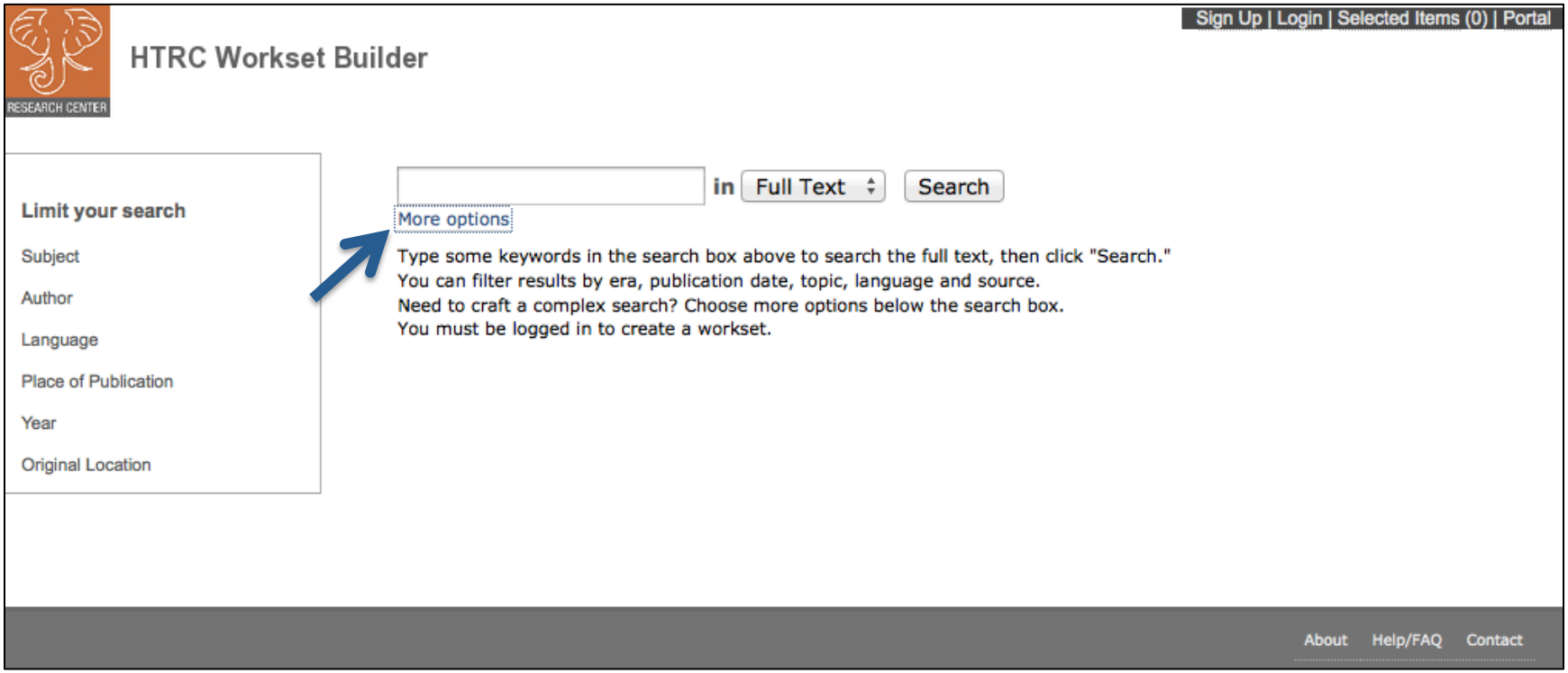
Choose to create workset



Log in to access Workset Builder

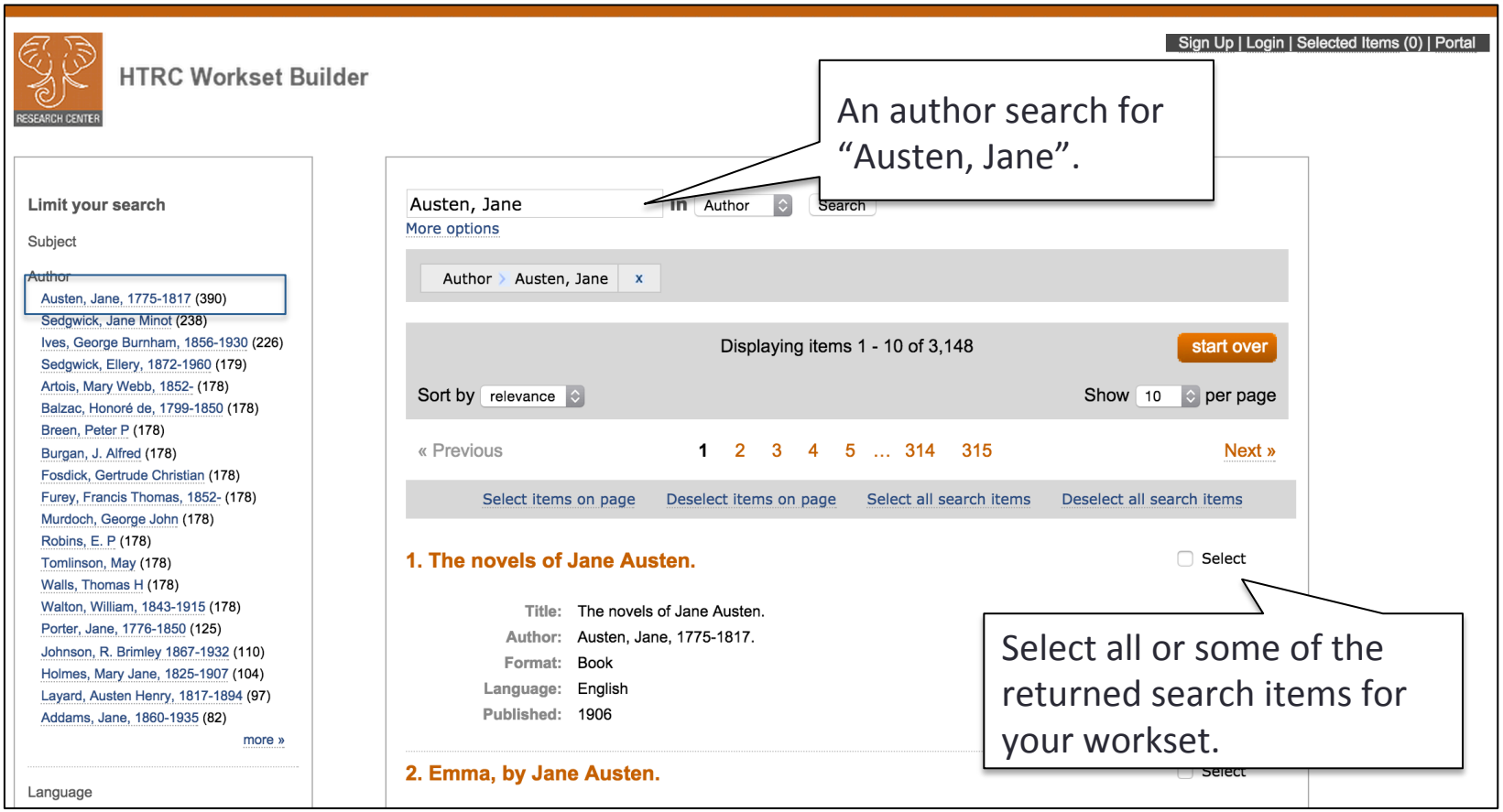


Build a workset



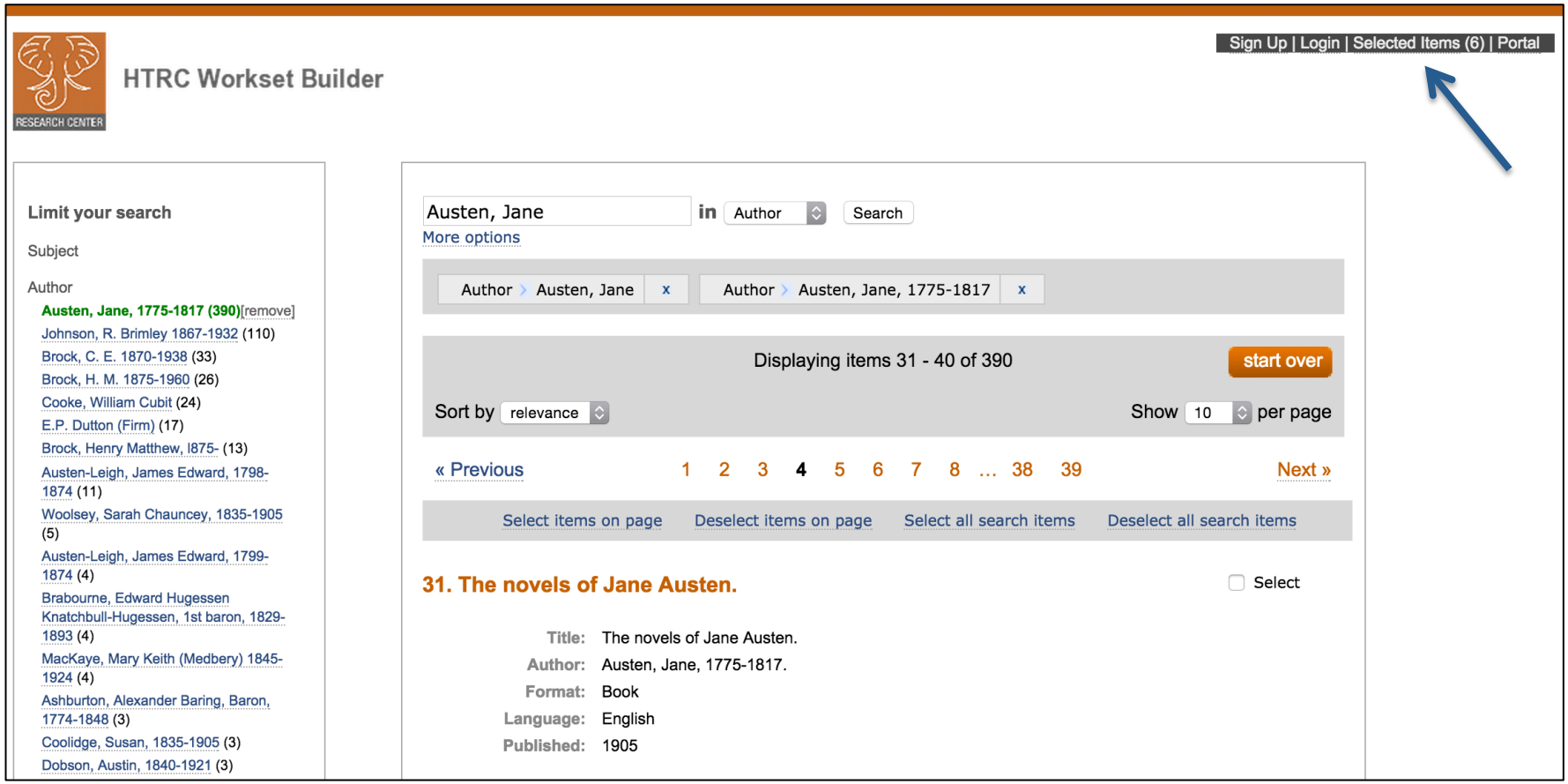
NB: Click "More options" for advanced search options

Workset Builder: Search Results



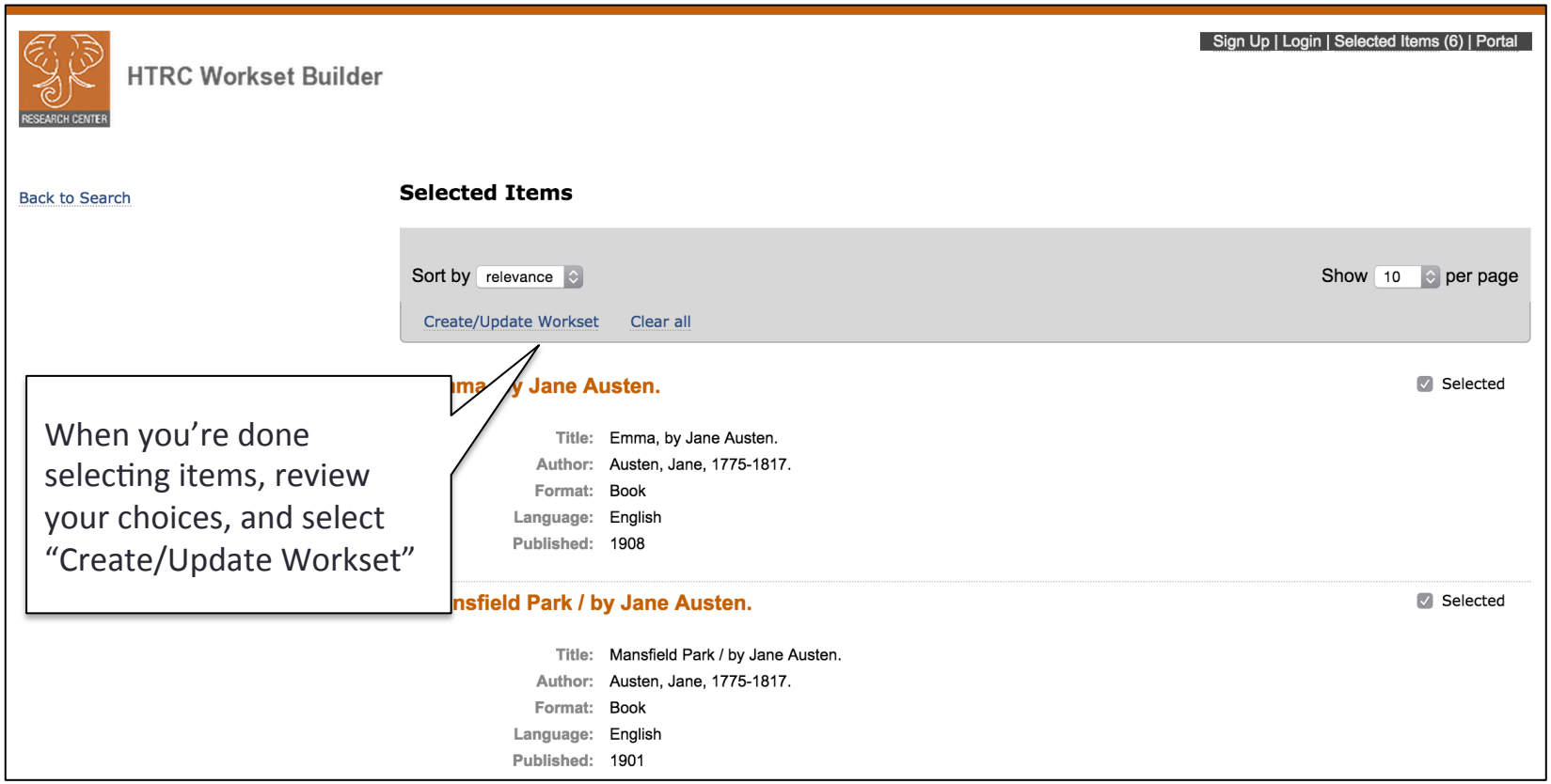
Filtering results by "Austen, Jane, 1775-1817" reduces results from 3,148 to a more manageable 390.

Workset Builder: Search Results



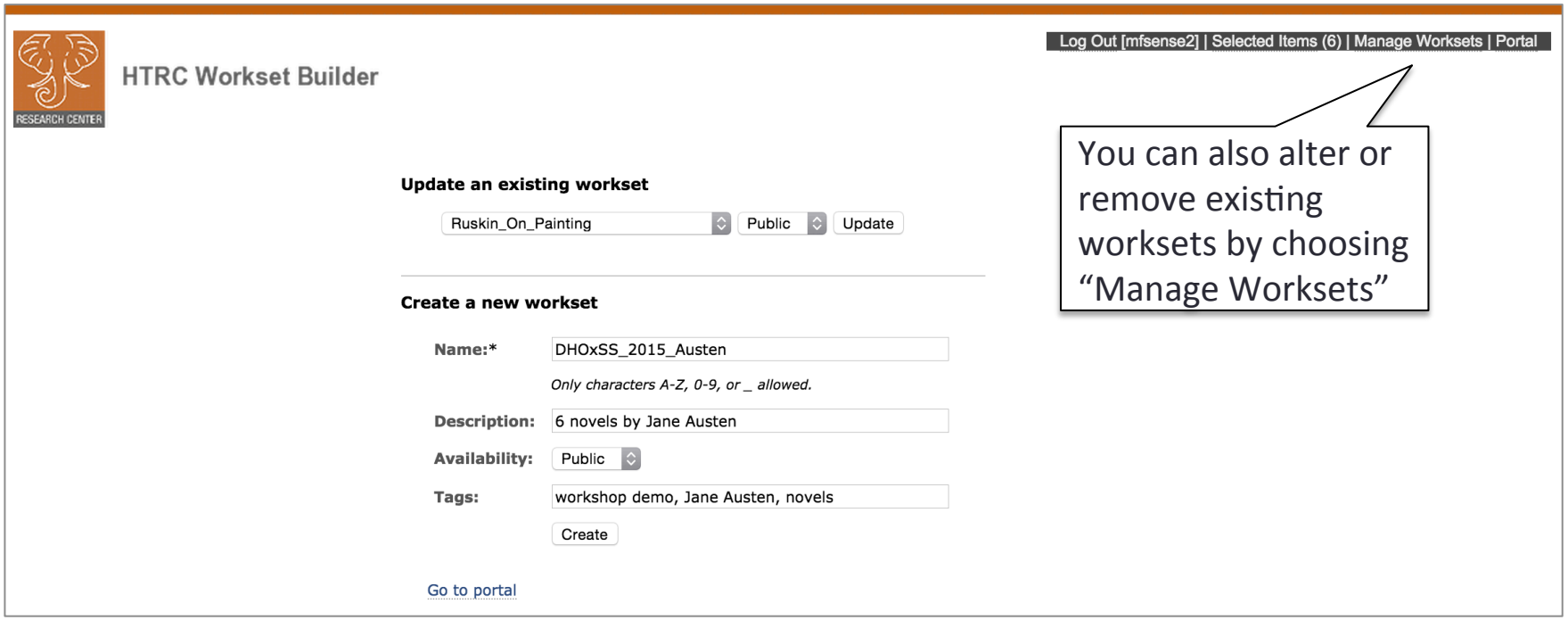
Once texts are selected, click “Selected Items”, located in upper right-hand menu

Workset Builder: Create Workset



When you're done selecting items, review your choices, and select "Create/Update Workset"

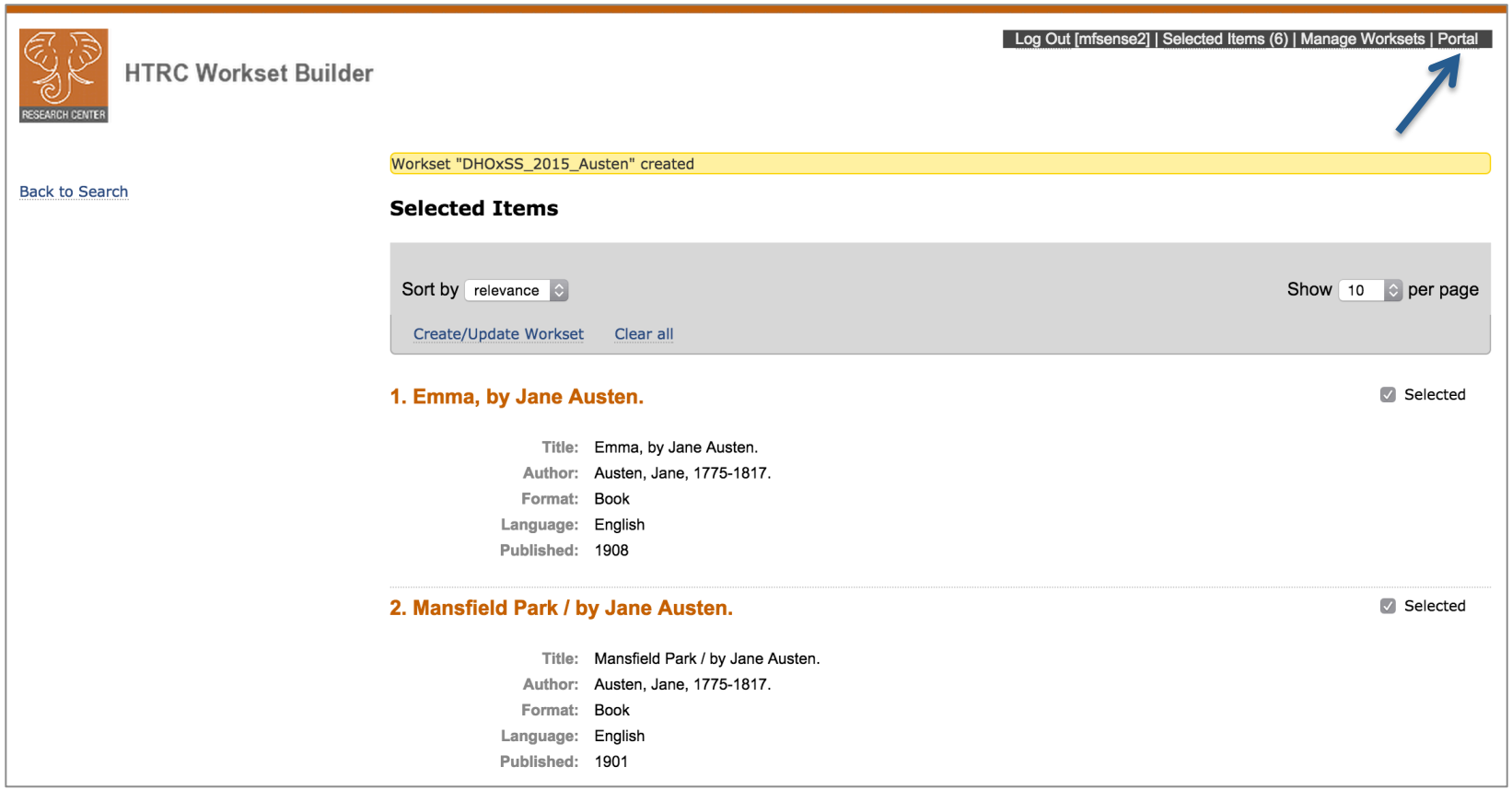
Workset Builder: Workset Metadata



You can also alter or remove existing worksets by choosing "Manage Worksets"

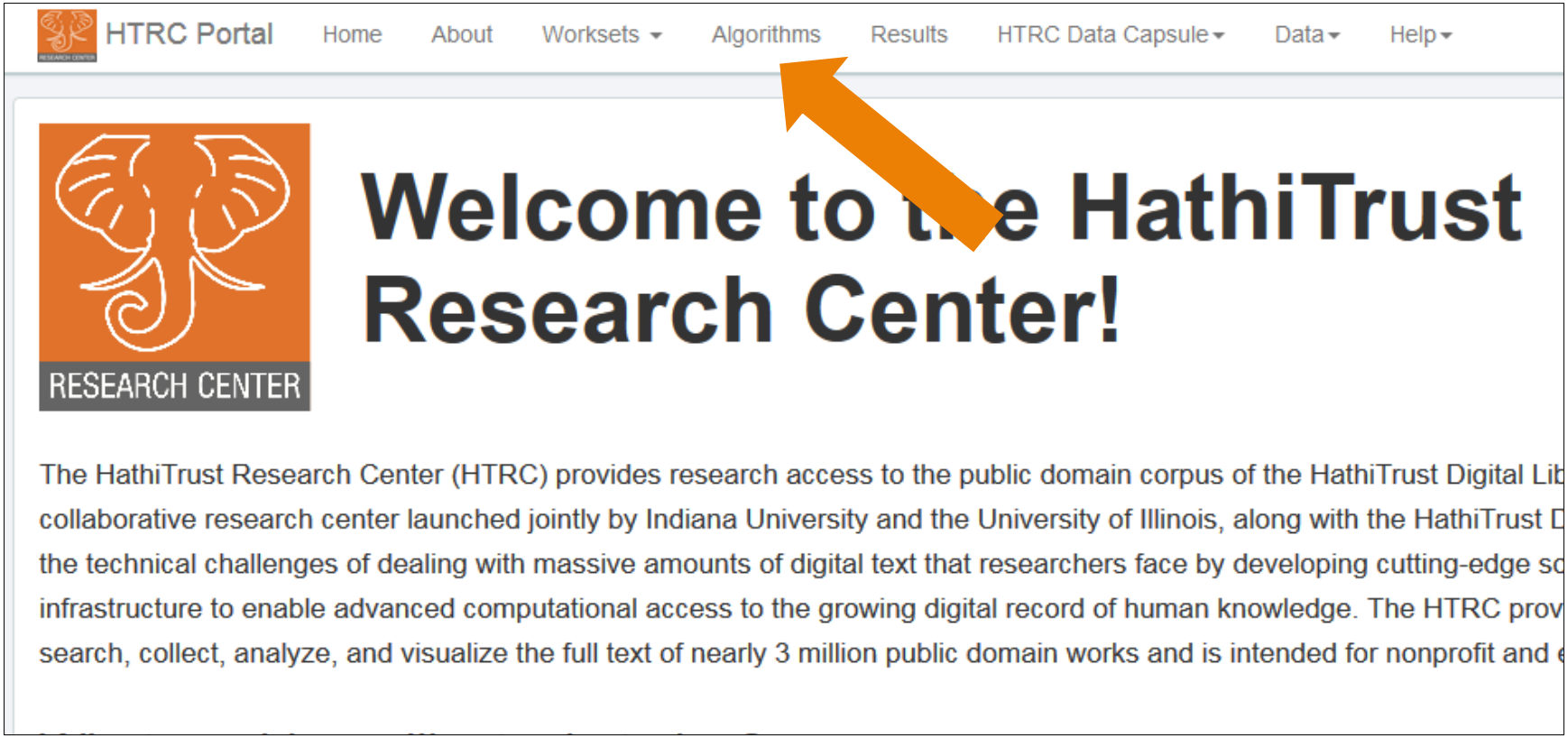
NB: you have the option to add selected items to an existing workset or to create a new workset.

Return to Portal



Questions?

Using the Portal Algorithms



Analysis in the HTRC Portal

Algorithms

#	Name	Description	Author	Version	
1	EF_Rsync_Script_Generator	This algorithm generates a script to download the extracted features (EF) data for the specified workset. The script uses Show more	Colleen Fallaw	2.2	Execute
2	Marc_Downloader	This algorithm downloads marc records for each volume in the input workset.	Zong Peng	1.6	Execute
3	Meandre_Classification_NaiveBayes	This analysis will perform NaiveBayes Classification. Given a workset (that was uploaded as a csv file) containing the workset Show more	Loretta Auvil	1.2	Execute
4	Meandre_Dunning_LogLikelihood_to_Tagcloud	This analysis calculates Dunning LogLikelihood based on the input of two worksets, an analysis workset and a reference Show more	Loretta Auvil	1.2	Execute
5	Meandre_OpenNLP_Date_Entities_To_Simile	Information extraction is used to extract date entities that can be displayed on a timeline. This allows a researcher to review Show more	Loretta Auvil	1.1	Execute
6	Meandre_OpenNLP_Entities_List	Information extraction is used to extract named entities and provide this information in a table. We are using the OpenNLP Show more	Loretta Auvil	1.2	Execute
7	Meandre_Spellcheck_Report_Per_Volume	This spellcheck analysis will load data and create several reports with results of spellchecking. This version of spellcheck Show more	Loretta Auvil	1.2	Execute

Prepare to run an algorithm

Description: Information extraction is used to extract named entities and provide this information in a table. We are using the OpenNLP system to extract the entities from the text in an automated fashion. Loads each page of each volume from HTRC. Removes the first and last line of each page. Joins hypenated words that occur at the end of the line. Extracts entity types specified from the text. Displays each entity with the volume_id, page_id, sentence_id and character position within the sentence. NOTE: The volume limit is 100.

Version: 1.2

Author: Loretta Auvil

Please enter a job name: (required)

testJobName

Please select a workset for analysis (required)

- (•) Select a workset from my worksets
- ( ) Select a workset from all worksets

Select a workset from my worksets

The workset containing the volume ids to be used for analysis.

Please provide a comma separated list of entity types to be extracted. Acceptable values are: date, location, money, organization, percentage, person, time. (default: person) (optional)

person

The entity type to be extracted.

Submit

Prepare to run an algorithm

Description: Information extraction is used to extract named entities and provide this information in a table. We are using the OpenNLP system to extract the entities from the text in an automated fashion. Loads each page of each volume from HTRC. Removes the first and last line of each page. Joins hypenated words that occur at the end of the line. Extracts entity types specified from the text. Displays each entity with the volume_id, page_id, sentence_id and character position within the sentence. NOTE: The volume limit is 100.

Version: 1.2

Author: Loretta Auvil

Please enter a job name: (required)

testJobName

Please select a workset for analysis (required)

- (•) Select a workset from my worksets
- ( ) Select a workset from all worksets

Select a workset from my worksets

The workset containing the volume ids to be used for analysis.

Please provide a comma separated list of entity types to be extracted. Acceptable values are: date, location, money, organization, percentage, person, time. (default: person) (optional)

person

The entity type to be extracted.

Submit

Prepare to run an algorithm

Description: Information extraction is used to extract named entities and provide this information in a table. We are using the OpenNLP system to extract the entities from the text in an automated fashion. Loads each page of each volume from HTRC. Removes the first and last line of each page. Joins hypenated words that occur at the end of the line. Extracts entity types specified from the text. Displays each entity with the volume_id, page_id, sentence_id and character position within the sentence. NOTE: The volume limit is 100.

Version: 1.2

Author: Loretta Auvil

Please enter a job name: (required)

testJobName

Please select a workset for analysis (required)

- (•) Select a workset from my worksets
- ( ) Select a workset from all worksets

Select a workset from my worksets

The workset containing the volume ids to be used for analysis.

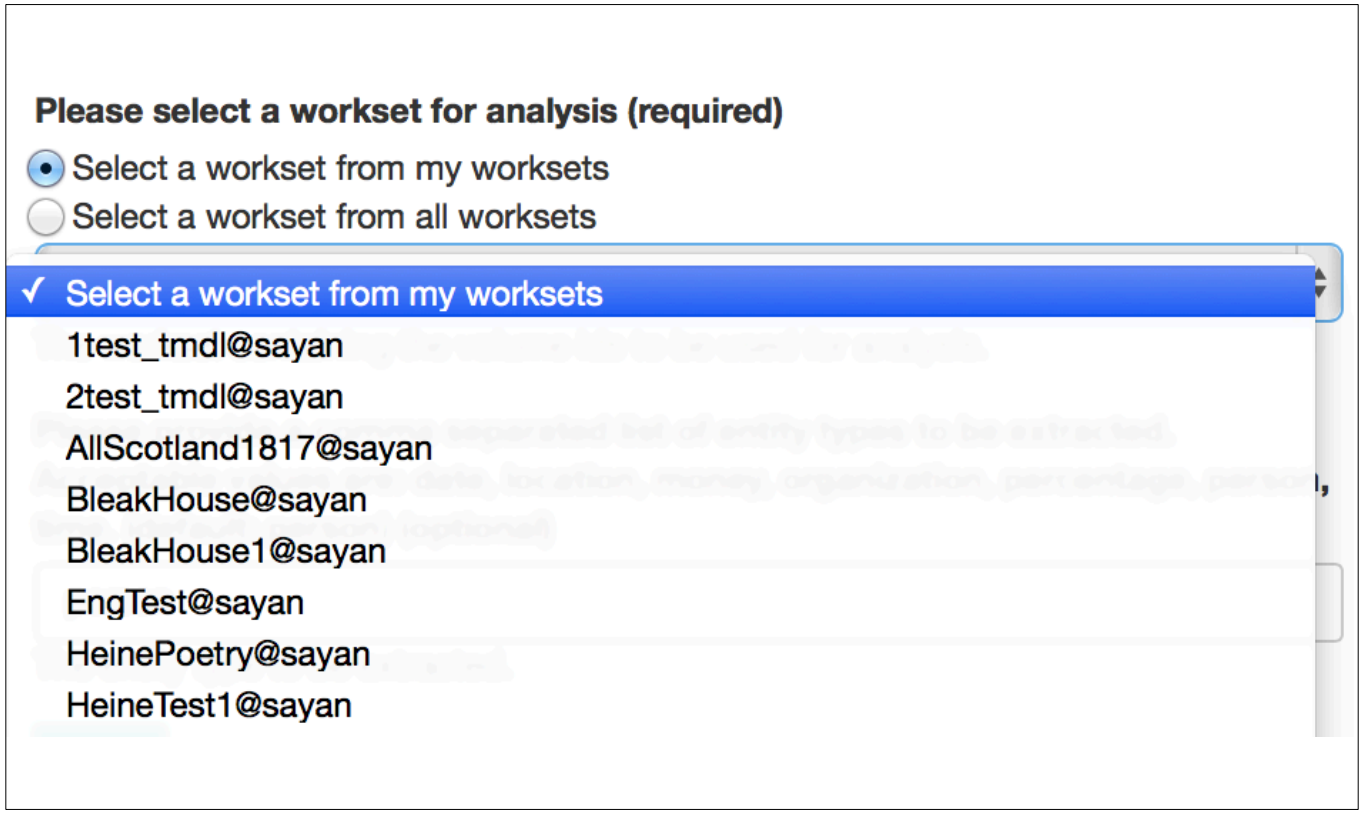
Please provide a comma separated list of entity types to be extracted. Acceptable values are: date, location, money, organization, percentage, person, time. (default: person) (optional)

person

The entity type to be extracted.

Submit

Choose workset(s) for analysis



Run the analysis…

HTRC Portal | Home | About | Worksets | Algorithms | Results | HTRC Data Capsule | Data | Help | eleanordickson

Active Jobs(1) Cancel

Job Name	Algorithm	Last Updated	Status	Cancel?
Meandre_Topic_Modeling-2015-06-25-13:6:24	Meandre_Topic_Modeling	2015-06-25 14:06:55	Staging	☐

Completed Jobs(7) Delete

Job Name	Algorithm	Finished Time	Status	Delete?
Meandre_Tagcloud_with_Cleaning-2015-06-25-13:2:44	Meandre_Tagcloud_with_Cleaning	2015-06-25 14:03:57	Finished	☐
Meandre_Tagcloud-2015-06-25-12:58:43	Meandre_Tagcloud	2015-06-25 14:01:51	Finished	☐
EF_Rsync_Script_Generator-2015-06-18-16:13:46	EF_Rsync_Script_Generator	2015-06-18 17:15:33	Finished	☐

View Results

HTRC Portal | Home | About | Worksets | Algorithms | Results | HTRC Data Capsule | Data | Help

eleanordickson

Name	Job ID	Algorithm	Last Updated	Status
Meandre_Topic_Modeling-2015-06-25-16:46:43	84825179-6fa9-4df5-a269-ce53f97ddf61	Meandre_Topic_Modeling	2015-06-25 17:48:47	Finished

Input Parameters

Name	Value
n_top_tokens	200
num_topics	10
input_collection	play1@eleanordickson

Output

topic_tagclouds.html | stderr.txt | stdout.txt | topic_top_words.xml

HT Points of Access

- HTRC Portal and Workset Builder
- HTRC Data Capsule
- HathiTrust Digital Library
  - HT data request
  - HathiFiles
  - Bib and Data APIs
- HTRC datasets
  - Extracted Features dataset
  - Genre in English Language Literature (Underwood)
- HT+BW

HTRC Data Capsule

An environment for performing non-consumptive text analysis

Acknowledgements

- This research is funded through a grant from the Alfred P. Sloan Foundation
- This research is also based in part on work supported by the National Science Foundation and by Samsung.
- Special thanks to Samitha Liyanage, Milinda Pathirage, and Zong Peng at Indiana University, and Earlence Fernandes and Ajit Aluri at the University of Michigan for discussions contributing to this work.

HTRC v2.0

- There is a mismatch between what HTRC v2.0 provides and the needs of some users.
  - HTRC v2.0 provides predefined algorithms to users and runs them on their behalf.
    - This is to prevent leakage of copyrighted data.
  - However, users typically want:
    - to run their *own* algorithms
    - to examine the results interactively
- The HTRC Data Capsule is being developed to:
  - strike a balance between:
    - preventing data leakage, and
    - keeping HTRC as flexible as possible to users.

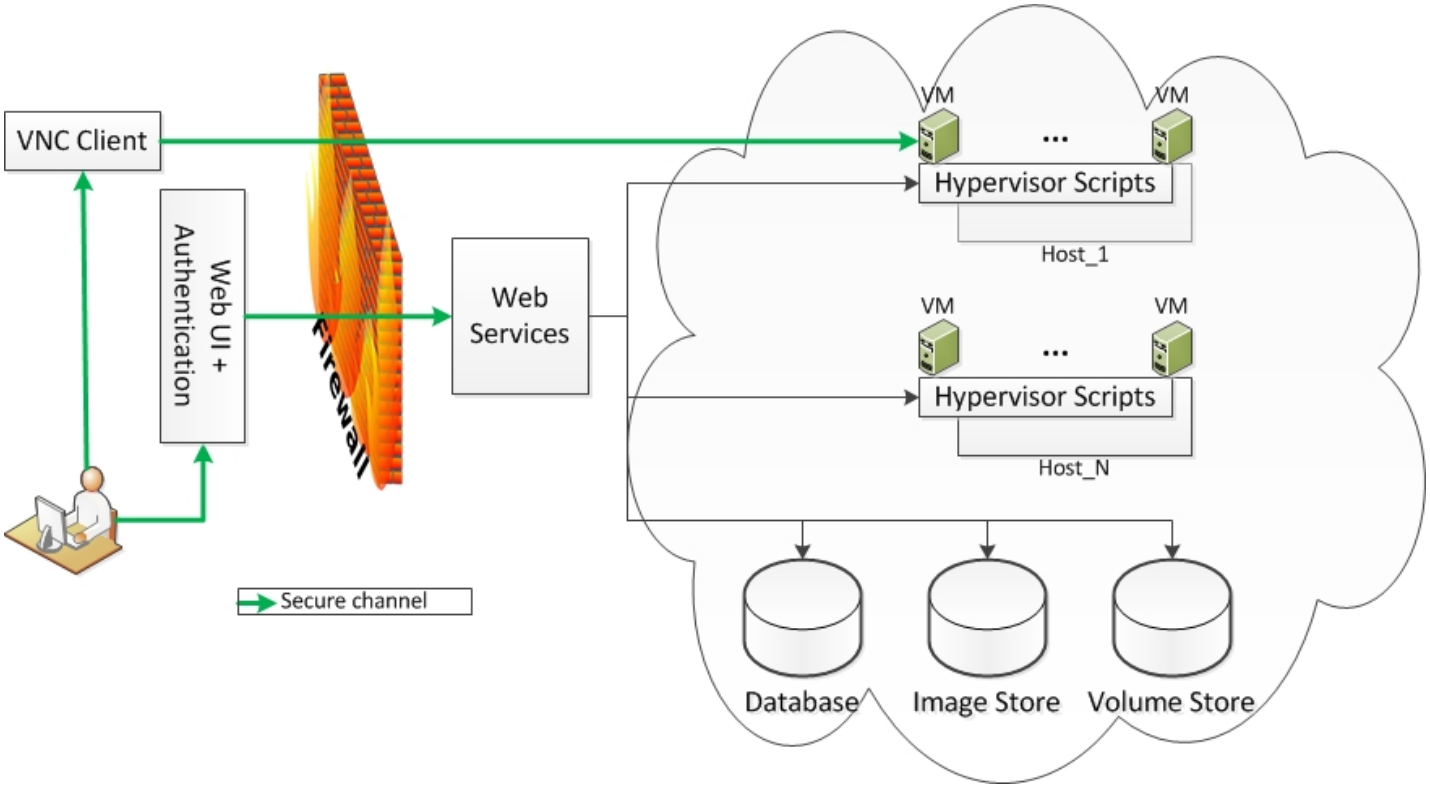
Research Questions

- **Non-consumptive use**: can framework provide safe handling of large amounts of protected data?
- **Openness**: can framework support user-contributed analysis without resorting to code walkthroughs prior to acceptance?
- **Large-scale** and **low cost**: can protections be extended to utilization of large-scale national (public) computational resources?

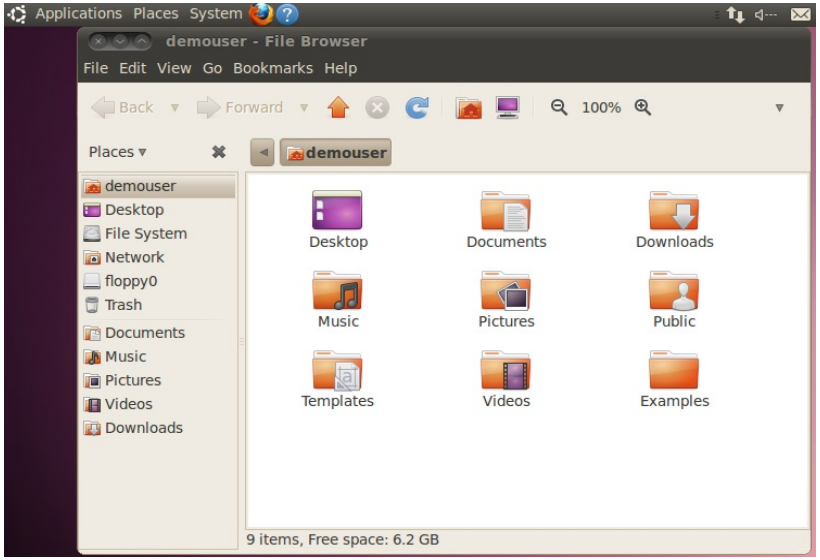
HTRC Data Capsule

- Provisions virtual machines (VM) for researchers to run their algorithms over copyrighted data.
- Trusts researchers to not deliberately leak copyrighted data.
- Prevents malware acting on researcher's behalf from leaking data.

HTRC Data Capsule Workflow

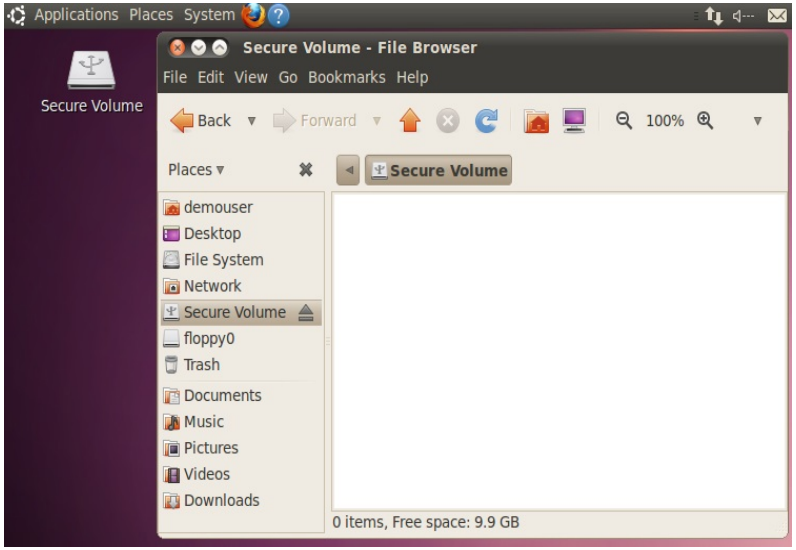


Secure Mode vs. Maintenance Mode

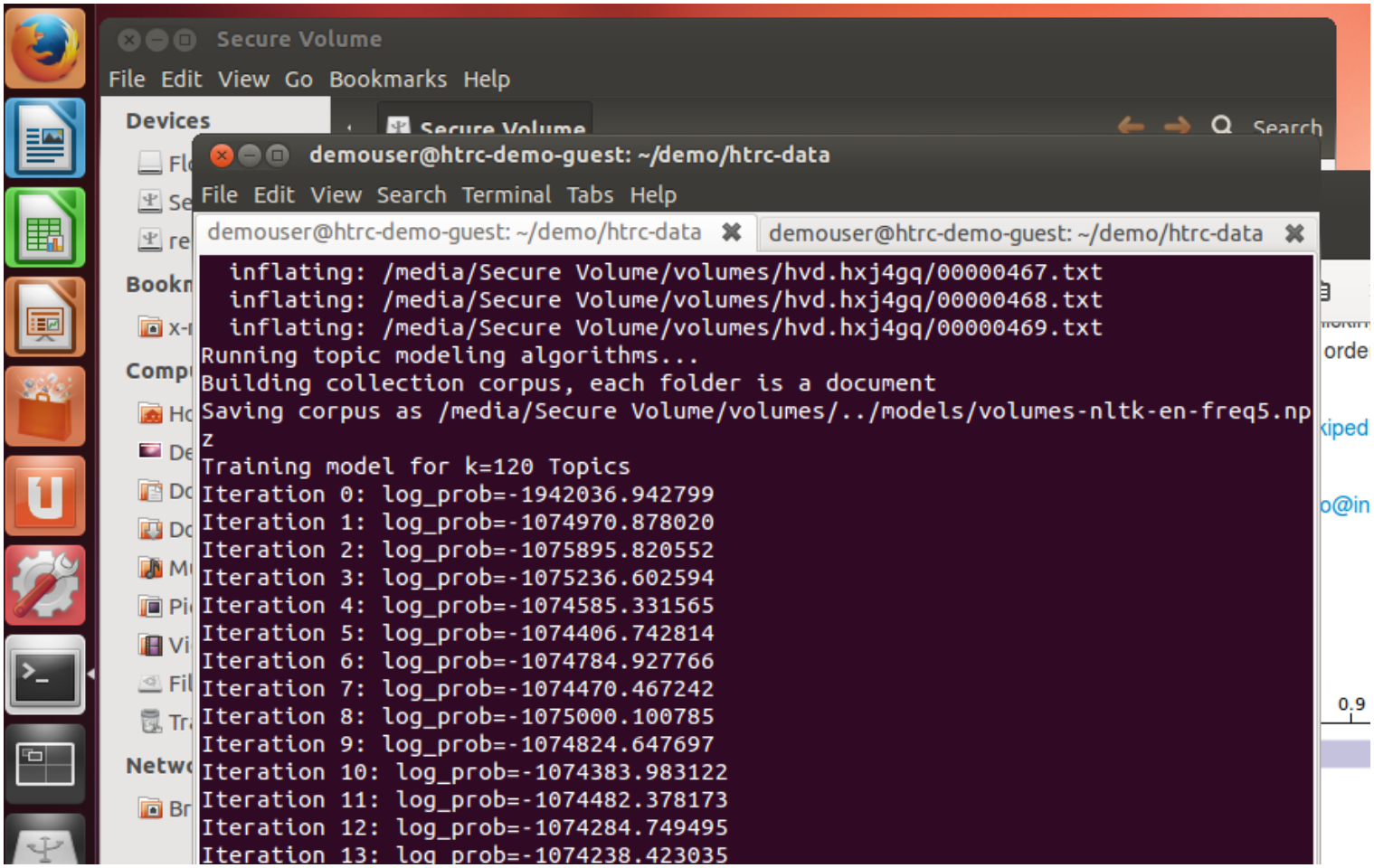


Maintenance Mode

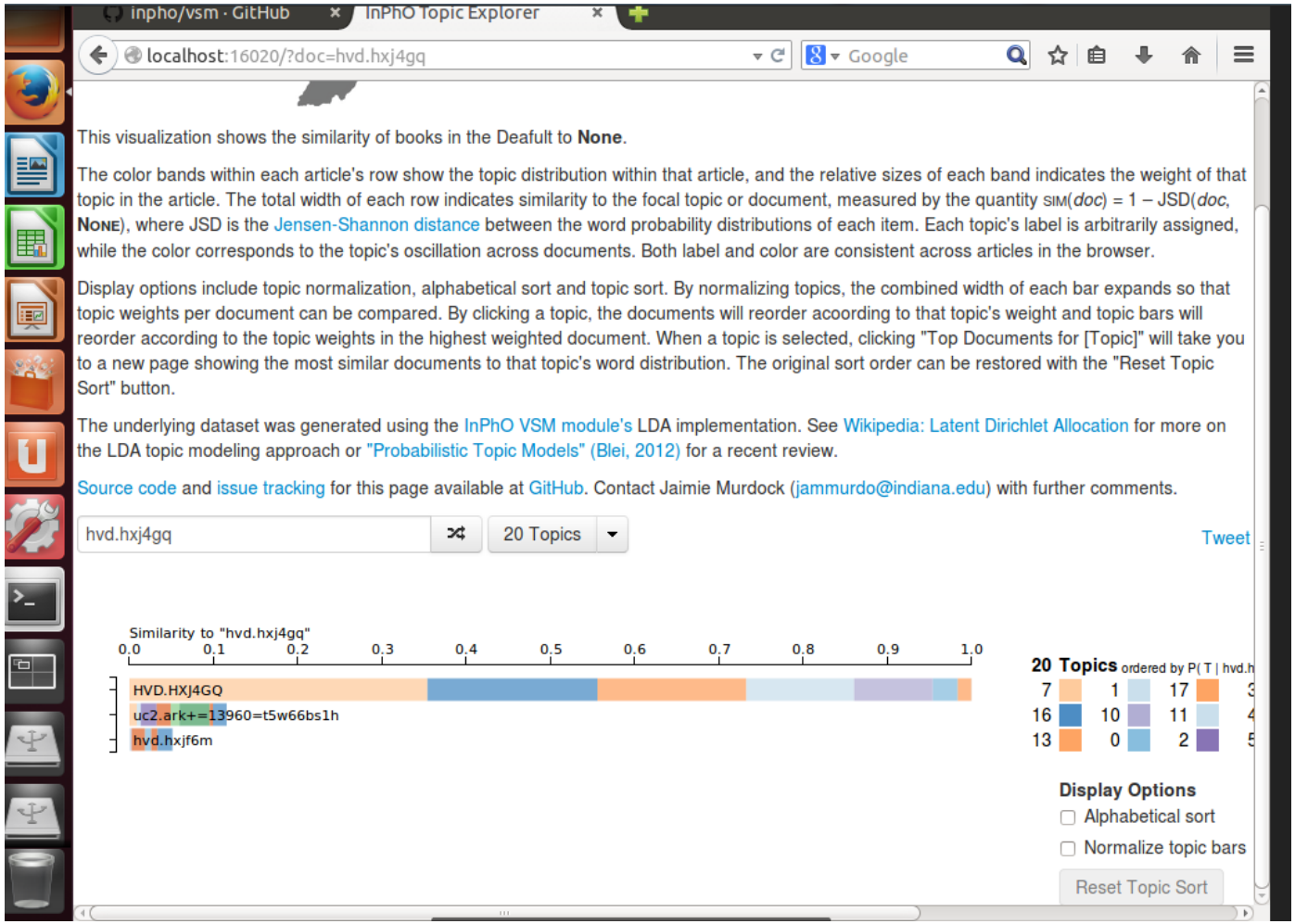
Secure Mode



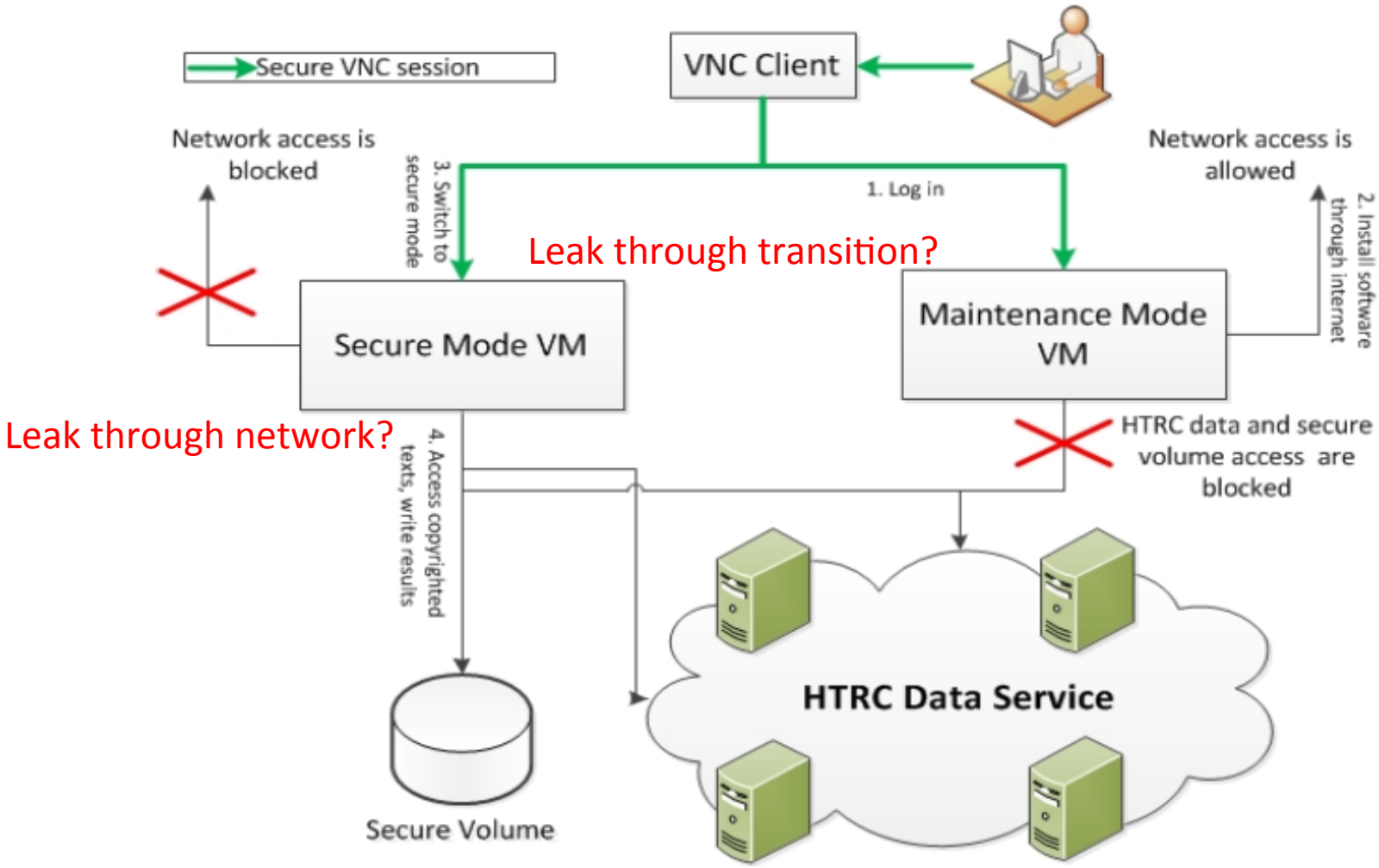
Run Experiments (Secure Mode)



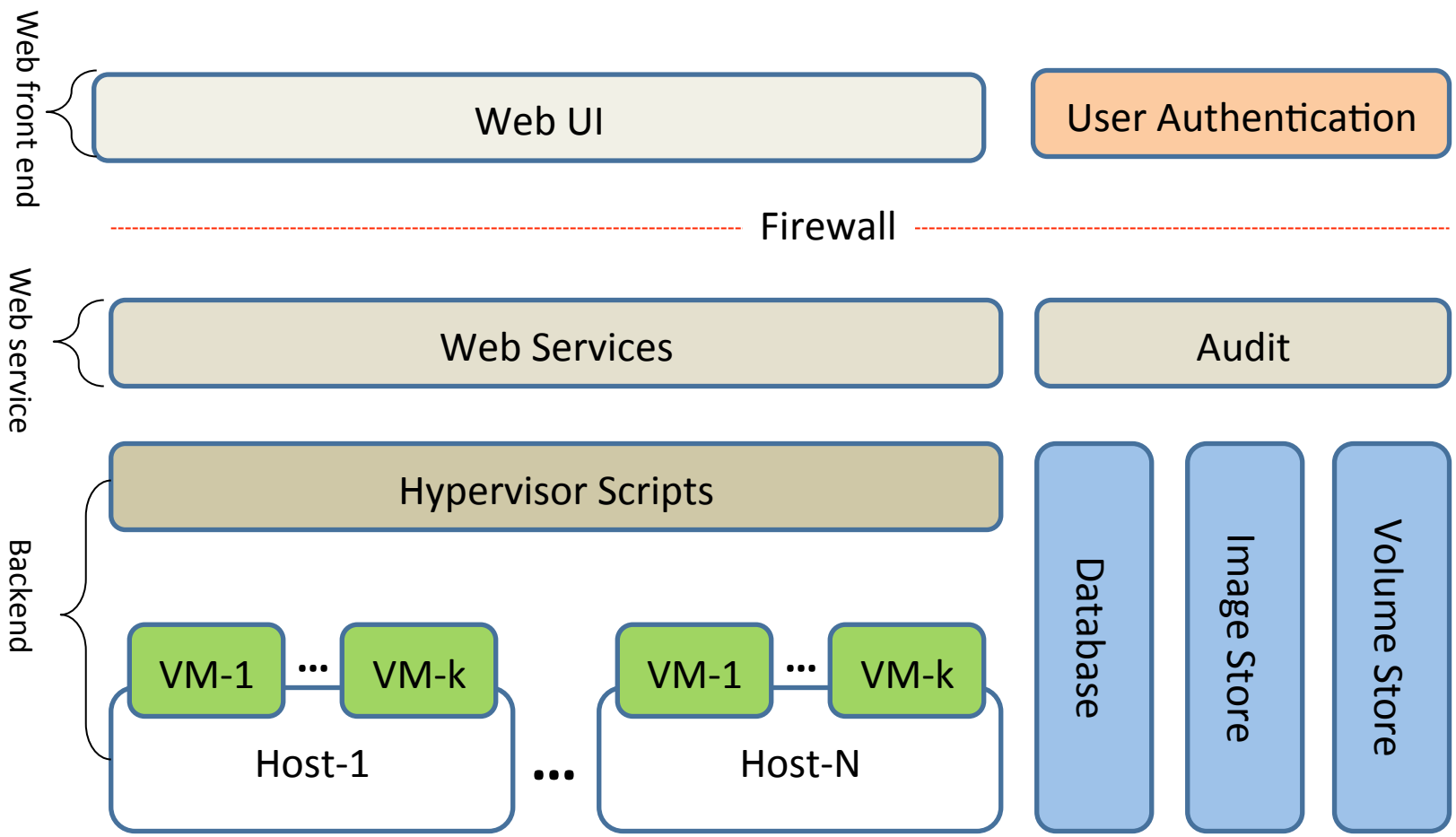
Run Experiments (Secure Mode)



HTRC Data Capsule Access



HTRC Data Capsule Architecture



Threat Model

- The user is trustworthy.
- The virtual machine manager and the host on which it runs are trusted.
- The VM is NOT trusted. We assume the possibility of malware being installed as well as other remotely initiated attacks on the VM, which are undetectable to the user.

Threat Model (Cont.)

- The VNC session and final result download are two channels for potential data leak. **BUT:**
  - For VNC session, we could encrypt the session to prevent eavesdropping.
  - For final result download, we could monitor traffic on the release channel as a means to automatically detect leakage.
- Covert channels between VMs on the same host could also potentially leak data.
  - In the future, we could run VMs on separated hosts to provide strong isolation.

HT Points of Access

- HTRC Portal and Workset Builder
- HTRC Data Capsule
- HathiTrust Digital Library
  - HT data request
  - HathiFiles
  - Bib and Data APIs
- HTRC datasets
  - Extracted Features dataset
  - Genre in English Language Literature (Underwood)
- HT+BW

Getting Data from the HTDL

- HathiTrust Digital Library
  - HathiFiles
    - Bibliographic metadata that can be useful for identifying volumes of interest
  - Bibliographic and Data APIs
    - Bib API – access bibliographic metadata
    - Data API – access content (images, OCR, and metadata)
  - HT data request
    - Request public domain content for research purposes

Extracted Features Dataset

A function for downloading selected data and metadata elements from a workset

Extracted Features (EF) dataset defined

- Downloadable page- and volume-level features extracted from a workset
  - Currently only pre-1923, public domain volumes available
  - Downloaded as JSON files
- Provides non-consumptive access **at scale** to volumes otherwise restricted by copyright

Extracted Features (EF) dataset: uses

- EF dataset includes part-of-speech-tagged words and their frequencies
- Any algorithm that works with "bags of words" will work with the EF dataset
  - Ex. topic modeling, Dunning's log-likelihood estimation for "compare-and-contrast",… etc.

Extracted Features (EF) Dataset: location

- Available at: https://sharc.hathitrust.org/features

Partial view of an EF data file json (“basic”)


```
"features":{
…
…
"header":{
        "tokenCount":7,
        "lineCount":3,
         …
        "tokenPosCount":{
          "I.":{"NN":1},
          "THE":{"DT":1},
          "INTRODUCTION":{"NN":1},
          ....
          "AND":{"CC":1}}},
"body":{
        "tokenCount":205,
        "lineCount":35,
        "emptyLineCount":9,
        "sentenceCount":6,
        "tokenPosCount":{
          "striking":{"JJ":1},
           "London":{"NNP":1},
          "four":{"CD":1},
          ".":{".":7},
          "dramatic":{"JJ":2},
          "stands":{"VBZ":1},
          ...
          "growth":{"NN":1}
        }
      },
      "footer":{
```

Extracted Features (EF) data has utility even for out-of-copyright material

- Page-level extracted features (EF) provide "a *sense of the content*" of each workset at the level of individual works.
  - EF dataset allows the flexibility of developing/running one's own queries and algorithms against the text data, *without* depending on provided algorithms

A use case for "Basic" EF data

- Charles Dickens' novels *Little Dorrit* and *Bleak House* are a bit *similar…*
  - …in that they both speak about the injustices of the 19th century British judicial system.
  - However, *Little Dorrit* and *Bleak House* are also *different*.
    - *Little Dorrit:*
      - emphasis is on civil law rather than prisons.
    - *Bleak House:*
      - *prisons* and *incarceration* play an important role
- We can corroborate this using EF data for the two novels:
  - Count (using the '`grep -o`' command at the shell command-line) the number of pages in which the word 'prison' occurs at least once, for:
    - the workset *LittleDorrit1; and*
    - for the workset *BleakHouse1*
  - Compare these two counts

Occurrences of “prison” in *Bleak House* and *Little Dorrit*

- Count (separately for *Little Dorrit* and *Bleak House*)
  - the number of pages in which the word ‘prison’ occurs at least once:

    - Downloaded EF dataset corresponding to the *LittleDorrit1* workset: *`hvd.32044025670571.basic.json`*
      - `grep –o prison hvd.32044025670571.basic.json | wc –l`
        - » Returns: **173**

    - Downloaded EF dataset corresponding to the *BleakHouse1* workset: *`miun.aca8482,0001,001.basic.json`*
      - `grep –o prison miun.aca8482,0001,001.basic.json | wc –l`
        - » Returns: **29**

- Compare these two counts:

173 >> 29

(‘Prison’ is way more prevalent in *Bleak House* than in *Little Dorrit !)*

(Q.E.D.)

Occurrences of "prison" in *Bleak House* and *Little Dorrit* (caveats)

- We simply counted the occurrences of the word "prison" within each *json*. This does give us a rough estimate, but to make a precise, accurate computation, you should actually parse the json to count the *frequency* of occurrences of "prison".

- Another simplification we made was to simply count occurrences of the exact word "prison".
  - For a large sized sample, this ought not matter so much.
  - However, as an exercise, you may want to redo the activity by considering any expression that contains the string prison — e.g. "imprisonment", "prisoner", "prisons", "prisoners", *etc.*

Showcase: Community contribution to enhance Extracted Features Dataset:
Eric Morgan Lease's Workset Browser

- Eric Lease Morgan, Digital Initiatives Librarian at the University of Notre Dame, has recently contributed a 'Workset Browser' toolkit
  - Makes use of EF data for a workset to provide analysis of content and metadata information for that workset.
  - Provides workset-level derived metadata and word usage statistics
  - Provides visualizations (as well as text lists) of "'big' names" and "'great' ideas" occurring in the workset
  - Walkthrough of a 32-item 'Thoreau' workset as example of what is possible.

Page-level maps of genre from English-language monographs in the HathiTrust Digital Library, 1700-1922 (Ted Underwood's work)

Sign In Sign Up

Word Frequencies in English-Language Literature, 1700-1922

Genre-specific wordcounts for 178,381 volumes from the HathiTrust Digital Library [v.0.1]

Note that this is a **beta** data release. Please send feedback to tunder@illinois.edu.

Many of the questions scholars want to ask about large collections of text can be posed using simplified representations – for instance, a list of the words in each volume, together with their frequencies. This dataset represents a first attempt to provide that information for English-language fiction, drama, and poetry published between 1700 and 1922, and contained in the HathiTrust Digital Library.

The project combines two sources of information. The word counts themselves come from the HathiTrust Research Center (HTRC), which has tabulated them at the page level in 4.8 million public-domain volumes. Information about genre comes from a parallel project led by Ted Underwood, and supported by the National Endowment for the Humanities and the American Council of Learned Societies. This project applied machine learning to recognize genre at the page level in 854,476 English-language volumes. Mapping genre at the page level is important because genres are almost always mixed within volumes. Volumes of poetry can have long nonfiction introductions; volumes of fiction can be followed by many pages of publishers' advertisements. Fortunately, text categories of this broad kind (fiction/nonfiction/poetry/drama/paratext) can be identified fairly accurately by statistical models

Contents

volumes of fiction	101,948
volumes of poetry	58,724
volumes of drama	17,709

Resources

- Spelling normalization and OCR correction
- Methods used for genre prediction

Page-level maps of genre from English-language monographs in the HathiTrust Digital Library, 1700-1922 (Ted Underwood's work)

- Goal:
  - To show how:
    - literary scholars can use machine learning
      - » for selecting genre-specific collections
        - from digital libraries
- What genres have been extracted?
  - "Imaginative literature"
    - Prose fiction, drama, poetry
      - » Assumed to be non-overlapping, discrete categories

Choice of genres

- Prose fiction, drama, poetry, non-fiction prose, paratext
  - Assumed to be non-overlapping, discrete categories
- Why *these* genres?
  - Broad categories like these easier to map
    - As we descend from broad categories to subgenres (e.g. “detective fiction”, “verse tragedy”, etc.),
      - » taxonomies become more like “folksonomies”
        - less sharp categories

HathiTrust and Genre Dataset

- Genre Mapping project (Ted Underwood) and Feature Extraction project (HathiTrust) both *mapped volumes at page level*
  - Created opportunity for collating results to produce genre-specific feature datasets
    - The provided genre datasets aggregate pages *at the volume level*
      - » E.g.:
        - The provided fiction dataset:
          - contains only volumes that were identified as containing fiction
          - aggregates the wordcounts for only those pages that were identified as fiction

Assumptions behind how genre was mapped

- Genre as "empirical social problem"
  - Not as constituted by "formal characteristics"
- Different strategies adopted for different aspects of the problem
- Overall strategy is supervised machine learning, mapping each page to a genre
  - Training set created by group of five human readers
    - Performed closed readings of training set and assigned categories
    - 1 graduate student and 4 undergraduate students
    - Agreed with each other only 76% of the time
      - But agreed 94.5% of time about *broad categories*

Assumptions behind how genre was mapped (contd.)

- *Predictive* rather than *explanatory* model of genre
  - Explanatory models:
    - Try to identify key factors behind a phenomenon
    - Identify deep structure
    - Capture causal logic
  - Predictive models:
    - Do not try to capture causal logic
    - Establish mapping between predictor variables and response variables
    - Do not require identification of key variables
    - Do not require structured data
    - Employ a "skeptical epistemology"
      - Appropriate for the humanities
        - as the humanities grapple with concepts that are not well-understood

Overview of how genre was mapped (contd.)

- Different combinations of methods and features were tried
  - Best method was found to be: regularized logistic regression
    - "regularization" in a supervised learning algorithm is a technique to prevent overfitting
  - Best features were selected to be 1062 features maximizing predictive accuracy (by optimizing a weighted average of precision and recall)
    - Consisted of:
      - 1034 words
      - 36 "structural features" (other page-level or volume-level information)
- With this best methods and best feature set, the regularized logistical regression model was trained for each genre:
  - using a one-versus-all training (each genre contrasted to all other genres collectively)
  - page-sequence information about genre was used to train a Hidden Markov Model that smoothed page-level predictions

How the training data was selected

- To ensure representativeness of sampling of training data:
  - a previous round of volume-level genre classification using Naïve Bayes classification served as guide to dividing corpus into volumes for subsequent random sampling
- Training data labeled by hand to match specific goal of project:
  - Separating the data into a few broad categories that literary critics distinguish between in practice:
    - Prose fiction, Drama, Poetry
  - Everything else categorized as prose non-fiction
- Historical specificity was traded off for increased training data volume:
  - Rather than train different models for each century separately,
    - as much training data as possible was used:
      - One training set covering the years 1700-1899 (to make predictions about volumes in this range)
      - Another training set covering the years 1725-1922 (to make predictions about volumes in the range 1900-1922)

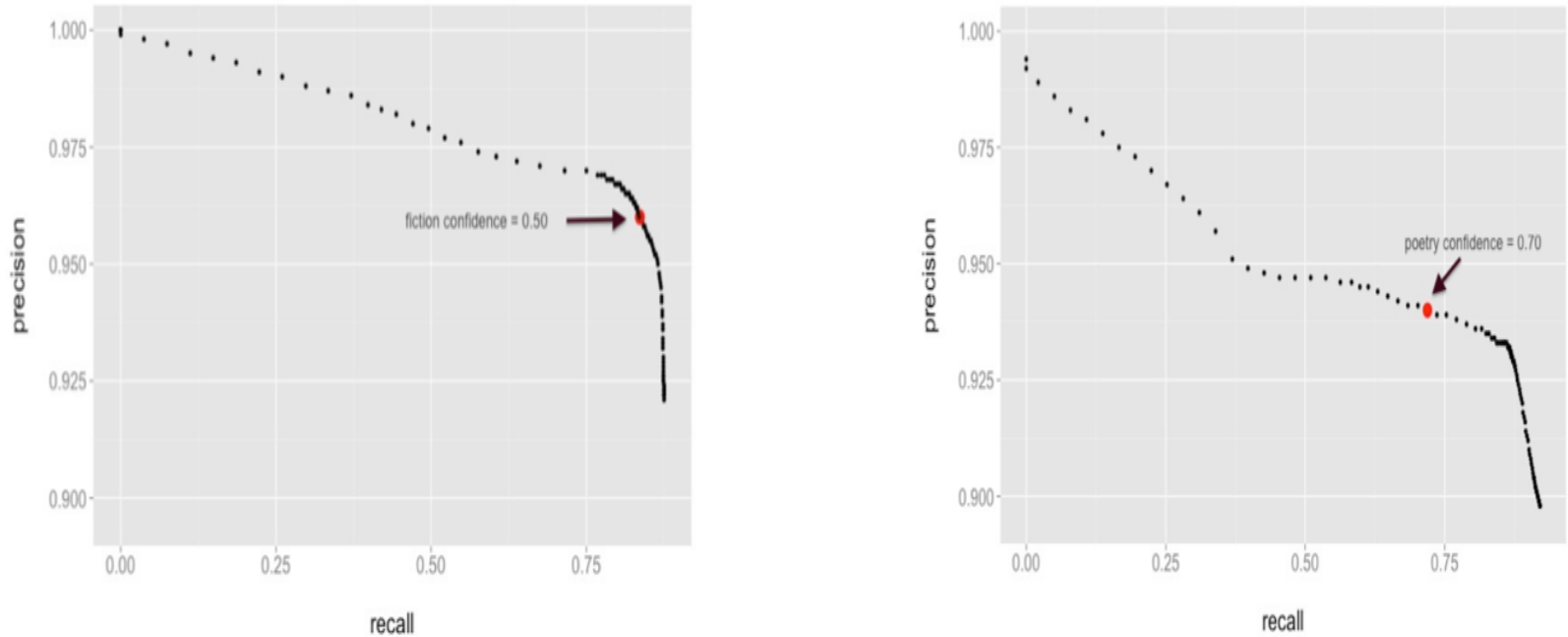
Empowering users to tune tradeoff between precision and recall

- Users were allowed to trade off precision against recall as needed:
  - This means:
    - Allowing users to make the genre definitions
      - more inclusive (higher recall but lower precision); or
      - less inclusive (higher precision but lower recall)
  - This capability was provided at volume level, not at page level
    - as most users are likely to treat texts as volumes.
      - Volume-level confidence metrics were computed and recorded:
        - » on the basis of:
          - page-level confidence metrics; and
          - additional volume-level evidence

Empowering users to tune tradeoff between precision and recall (contd.)

- Users can define their own corpus of fiction, poetry or drama,
  - either by:
    - setting whatever confidence threshold they desire; and
    - pulling out volumes that meet that threshold
  - or by:
    - choosing from some special pre-packaged "filtered" collections for these three genres
      - (which have been provided by identifying, for each genre, confidence thresholds that increase precision for relatively little loss of recall)

Empowering users to tune tradeoff between precision and recall (contd.)



Precision-recall graph for fiction datasets (left) and poetry datasets (right), limited to various thresholds of confidence.

These curves are "incorporated" into the JSON metadata for all volumes where any pages of poetry, drama or fiction are predicted. So, a researcher concerned (for example) that the provided dataset excludes too much poetry, can recreate his/her own filtered dataset with a confidence threshold that creates his/her desired precision/recall trade-off.

Feature engineering

- Best features were selected to be 1062 features maximizing predictive accuracy
  - Selected by grouping pages into categories corpus was planned to be classified into
    - Top 500 words selected from each category
    - Words from each category grouped into a master list limited to the top N most frequent words
  - Consisted of:
    - 1034 words
    - 26 “structural features” (other page-level or volume-level information):
      - per-page features:
        - *wordsPerLine*, *totalWords* (highly predictive for fiction), *capRatio* (highly predictive for poetry), *etc.*
      - volume-level features:
        - metaFiction (information in metadata indicating fiction), *etc.*
    - 10 categories:
      - e.g. *personalname*, *placename*, *propernoun*, *romannumeral*, *arabic1digit*, etc.

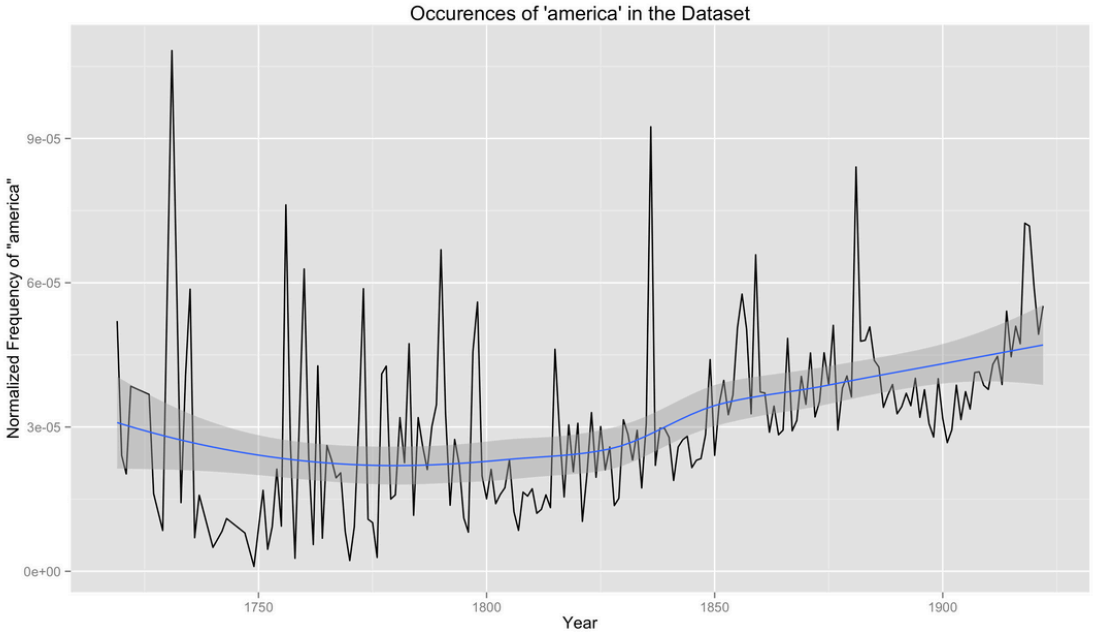
Performance of model

- On broad categories:
  - Prose fiction, drama, poetry, non-fiction prose, paratext:
    - Achieved 93.5% accuracy overall
      - Compared to 94.5% for the 5 human readers
  - Weak but slight correlation between pages where model failed, and pages where human readers disagreed
- Performance always has a tradeoff between precision and recall
  - Precision:
    - Better precision means minimization of the proportion of false positives to true positives
  - Recall:
    - Better recall means minimization of the proportion of false negatives to true negatives

Plotting lexical trends within a specific genre dataset (e.g. fiction)

- Chris Forster (Syracuse University) recently made a blog post showing simple explorations of the genre dataset using R.
- One such script plots normalized frequencies for any word in a given genre
  - E.g. A plot of normalized frequencies for the word 'america' in the fiction dataset:

or so:



Plotting lexical trends within a specific genre dataset: What is happening under the hood?

The file fiction_yearly_summary.csv in the fiction dataset contains:

- per-year frequencies of top 10,000 most frequently occurring tokens in fiction dataset.

R script for plotting trends in genre dataset for words occurring among top 10,000 (adapted from Chris Forster):

```
require (ggplot2)
yearly.summary <- read.csv('~/Downloads/fiction_yearly_summary.csv')
lady <- subset(yearly.summary, yearly.summary$word==’america')
yearly.total <-
        subset(yearly.summary,yearly.summary$word=='#DICTIONARYWORD')
yearly.total <- yearly.total[c('year','termfreq')]
colnames(yearly.total) <- c('year','total')
america <- subset(yearly.summary, yearly.summary$word==’america')
lady.normalized <- merge(america, yearly.total, by=c('year'))
ggplot(america.normalized, aes(x=america.normalized$year, y=(america.normalized
$termfreq/america.normalized$total)))+ geom_line() + geom_smooth() +  xlab('Year') +
ylab('Normalized Frequency of ”america"') +  ggtitle("Occurences of ’america' in the
Dataset")
```

Plotting lexical trends within a specific genre dataset: What is happening under the hood? (contd.)

- What if you are interested in a word that does *not* occur top 10,000 most frequently occurring tokens?
  - You will need to traverse files for individual volumes (loop over all the files)
    - For each file:
      - open the file
      - look for occurrence of the word
      - If it occurs, add its count to the total

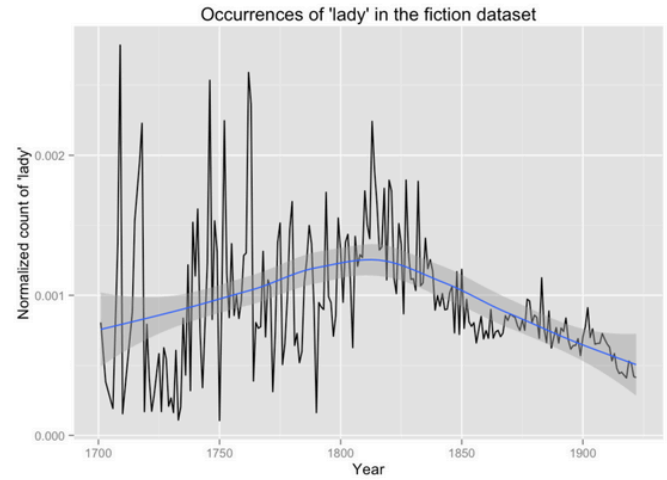
Plotting trends within the fiction dataset (contd.) “lady” versus “woman”

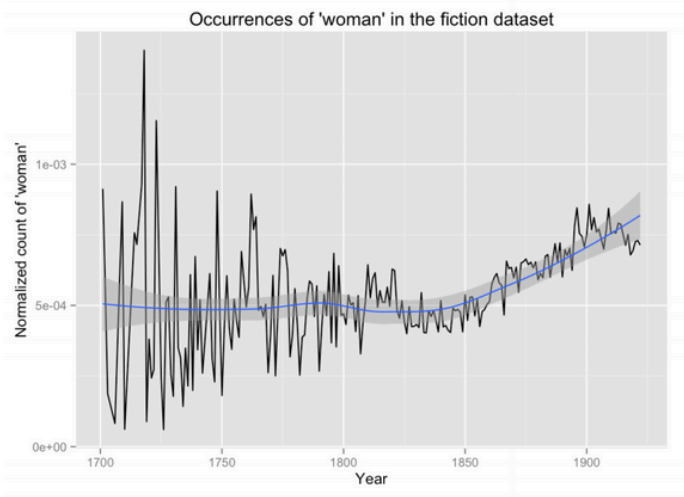
- The plot of normalized occurrences of "lady" has an inverted 'v'-shaped curve:
  - slowly and steadily rising until about 1810, and
  - slowly and steadily falling subsequently.

- Plot of normalized occurrences of 'woman' remain more or less constant until about 1840,
  - subsequently keeps rising slowly and steadily.

- Hypothesis seems borne out:
  - As we reach modern times,
    - occurrences of "lady" start declining in fiction
    - occurrences of "woman" increase.

Plotting trends within the fiction dataset (contd.) “lady” (top plot) versus “woman” (bottom plot)





The HathiTrust+Bookworm project

Team Members:

- Current:

  J. Stephen Downie, University of Illinois, Urbana-Champaign
  Erez Lieberman Aiden, Baylor College of Medicine
  Benjamin Schmidt, Northeastern University
  Robert McDonald, Indiana University
  Loretta Auvil, University of Illinois, Urbana-Champaign
  Peter Organisciak, University of Illinois, Urbana-Champaign
  Muhammad Shamim, Baylor College of Medicine
  Sayan Bhattacharyya, University of Illinois, Urbana-Champaign
  Leena Unnikrishnan, Indiana University

- Past:

  Colleen Fallaw, University of Illinois, Urbana-Champaign
  Matt Nicklay, Baylor College of Medicine

Funded by an NEH Implementation Grant (2014-2016)

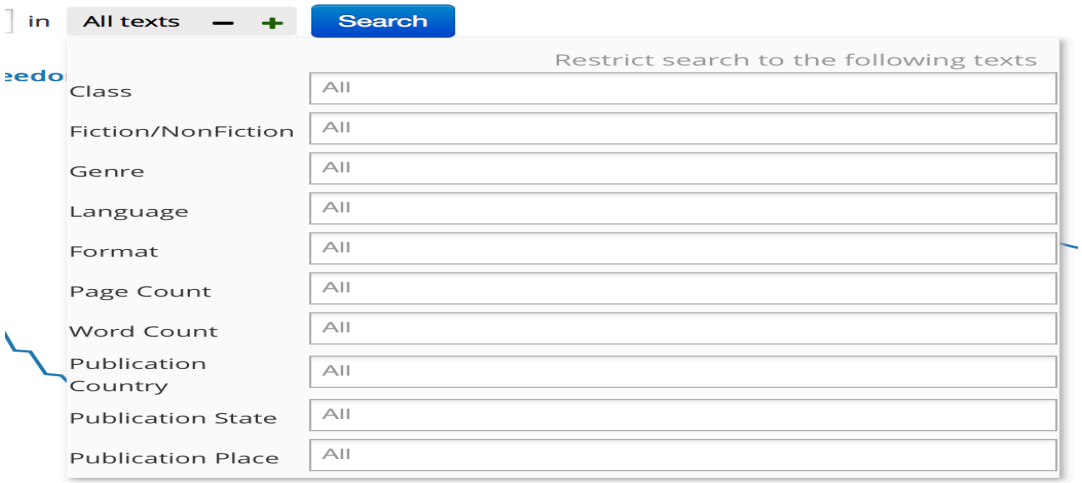
Current prototype of HT+BW

- Approximately 14 million books
- Approximately 4.8 million of the 14 million books are in the public domain
  - Current prototype of HT+Bookworm set up to work with these 4.8 million books

Hooking up HTDL with Bookworm: Advantages?

First advantage: Good metadata!

- HTDL has good and detailed **metadata**
  - (relatively speaking!)
  - historically, metadata was meticulously created by librarians from contributing libraries
  - ➢ allows for highly faceted queries:

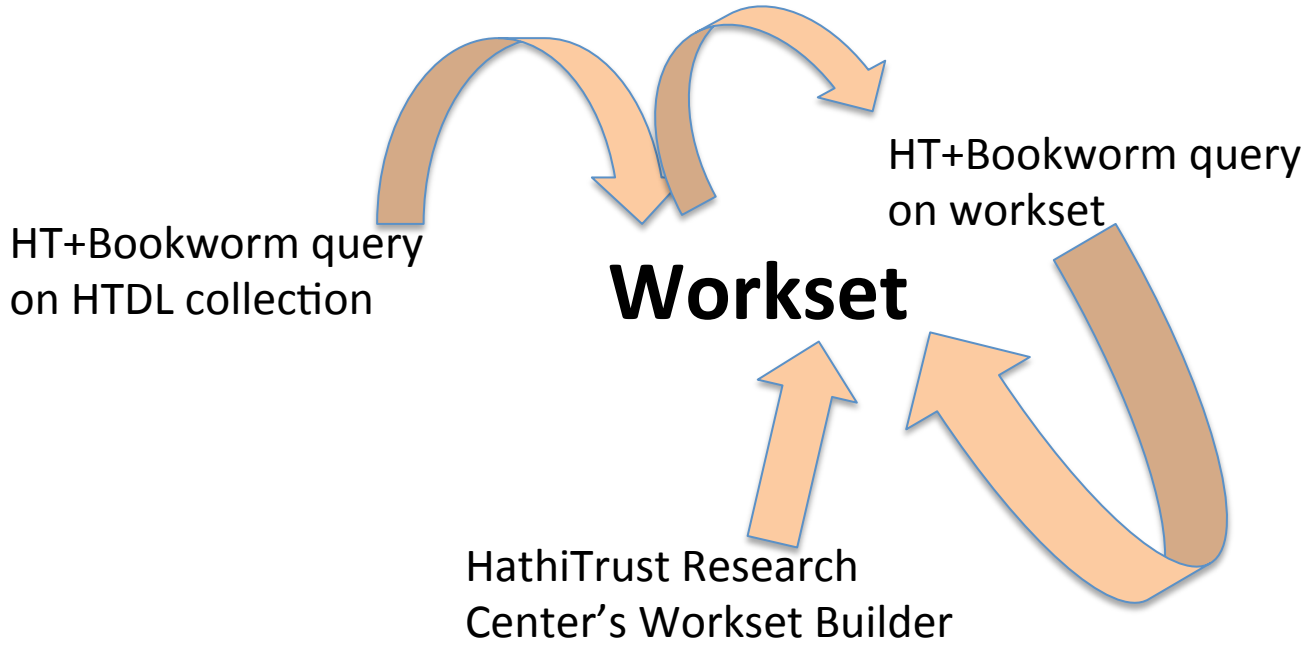


Hooking up HTDL with Bookworm: Advantages?
Second advantage: HTDL's *workset* functionality

- HTDL has a rich notion of **workset**
  - A workset is a user-determined sub-collection from the HTDL's collection
    - so called because it is a *set* of books meant to be *worked* on by algorithms for text analysis
  - HT+Bookworm leverages the workset functionality
    - HT+Bookworm will eventually:
      - *query* a specific workset (rather than the entire collection)
      - *generate* a specific workset from the result of a faceted query

Hooking up HTDL with Bookworm: Advantages?

Second advantage: HTDL's *workset* functionality (contd.)



Workset creation and refinement workflow

How *scientific inquiry* meets *humanistic inquiry* in culturomics as performed by HT+Bookworm

- Scientific inquiry concerns:
  - *Generalization* across entities
  - Discovery of patterns *across* entities
  - Epistemological determination

- Humanistic inquiry concerns:
  - Close engagement with *specific* entities
  - Attending to singular instances among entities
  - Epistemological skepticism

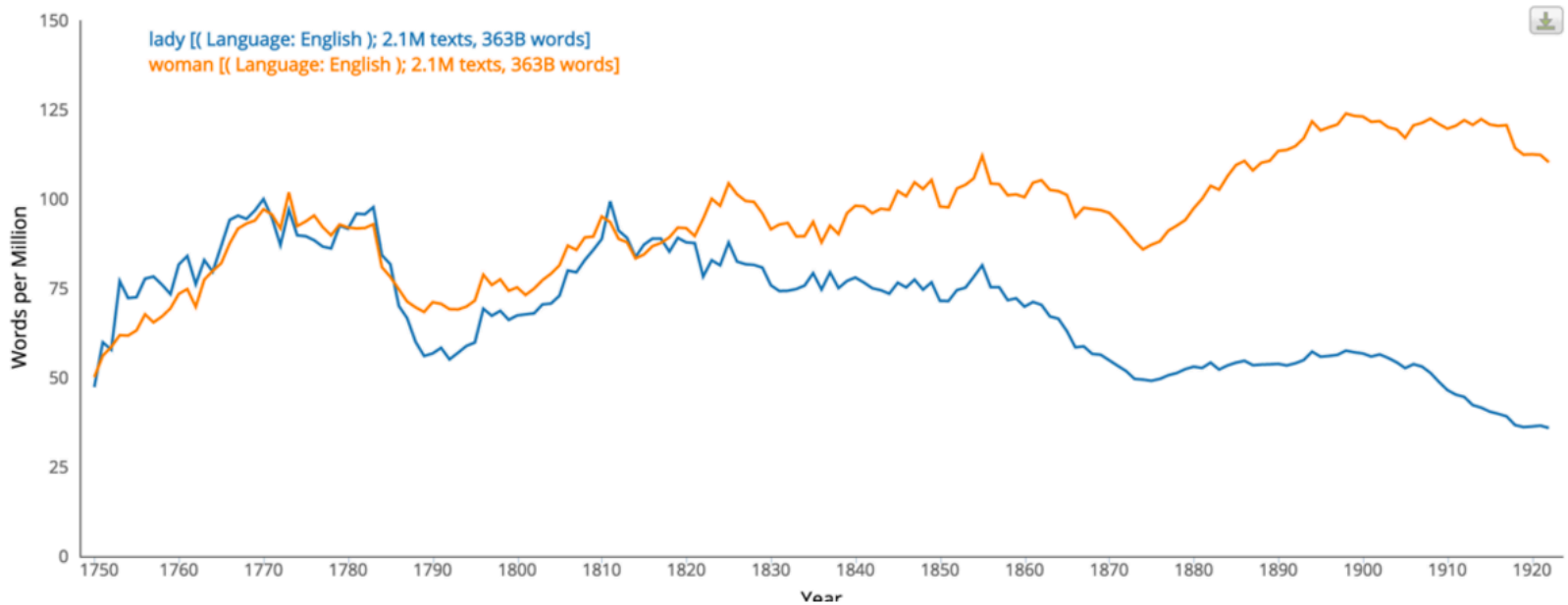
How *scientific inquiry* meets *humanistic inquiry* as performed by HT+Bookworm (contd.):

Three use cases

- How:
  - change in social context *over time* correlates with change in preponderance of one word-concept over another *over time*
  - occurrences of related word-concepts in multiple languages/places compare with each other *over time*
  - metaphorical associations of a word vary *over time*

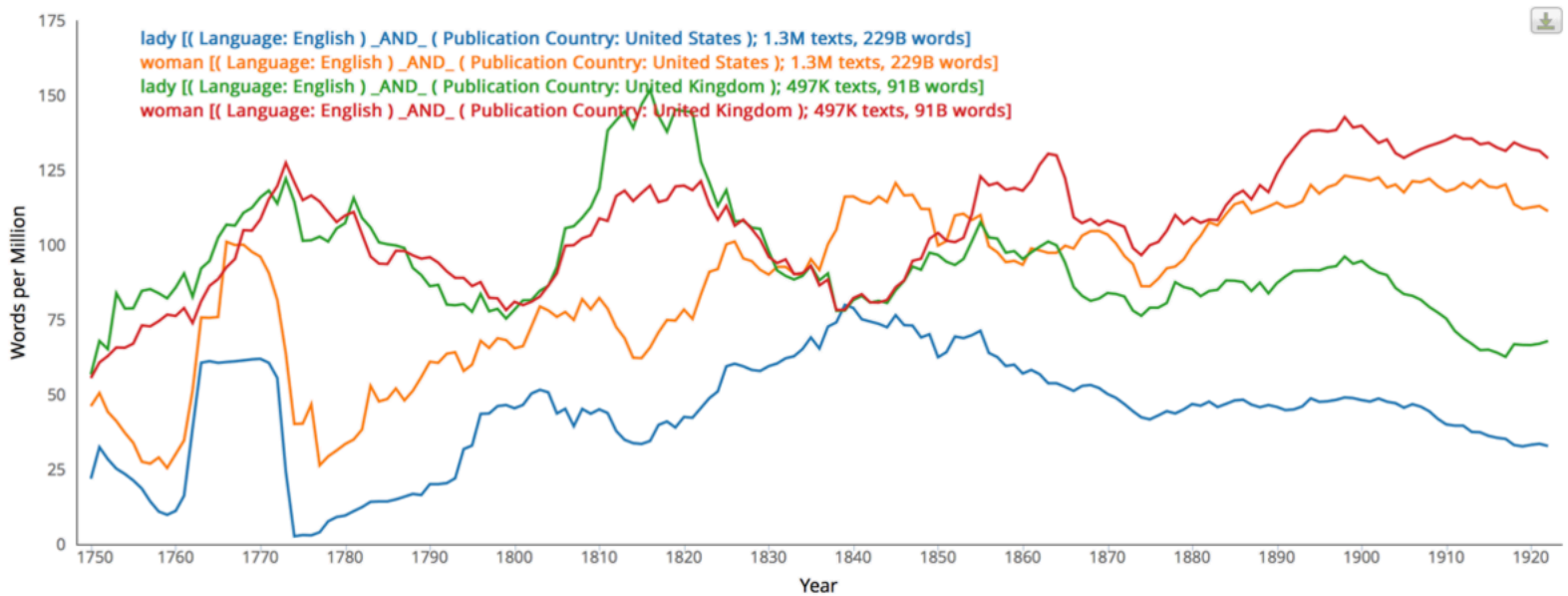
How change in social context across time correlates with change in preponderance of one word-concept over another

Search for *lady* in (Language: English)
Search for ***woman*** **in (Language: English)**



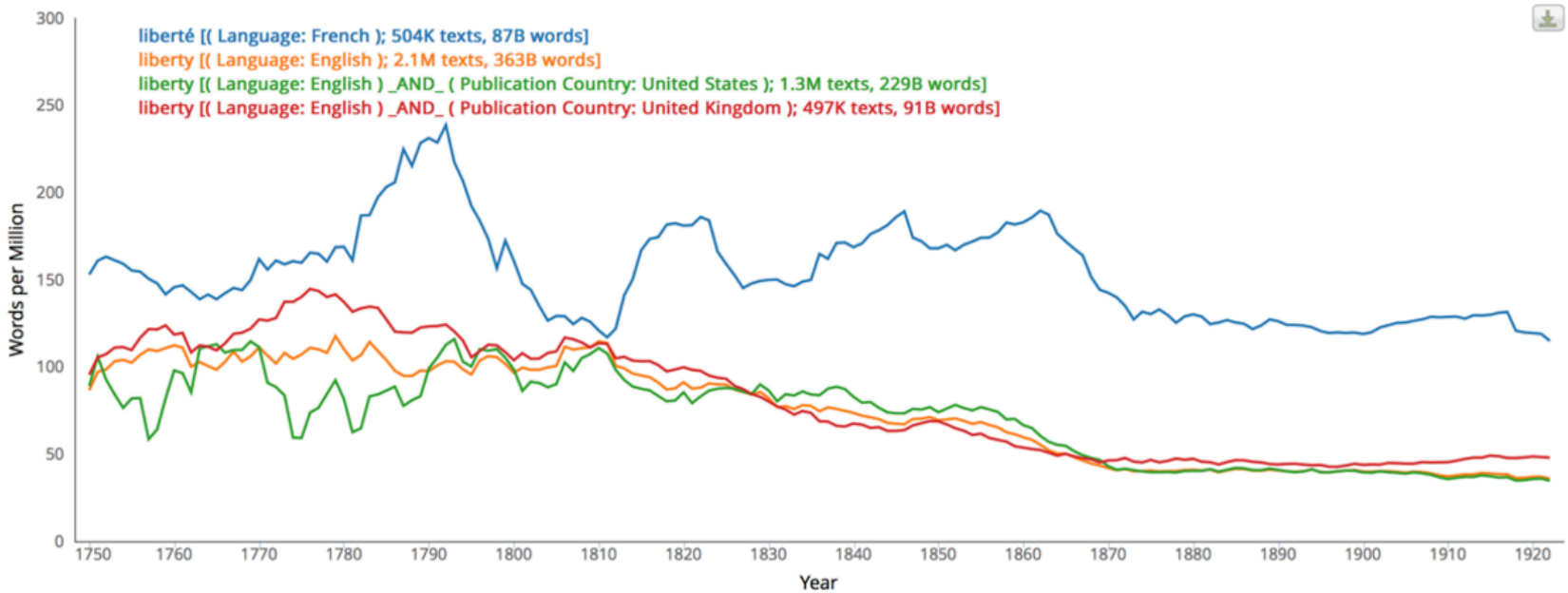
How change in social context across time correlates with change in preponderance of one word-concept over another (contd.)

Search for *lady* in (Language: English) AND (Publication Country: United States)
Search for *woman* in (Language: English) AND (Publication Country: United States)
Search for *lady* in (Language: English) AND (Publication Country: United Kingdom)
Search for *woman* in (Language: English) AND (Publication Country: United Kingdom)



How occurrences of related word-concepts in multiple languages/ places compare with each other

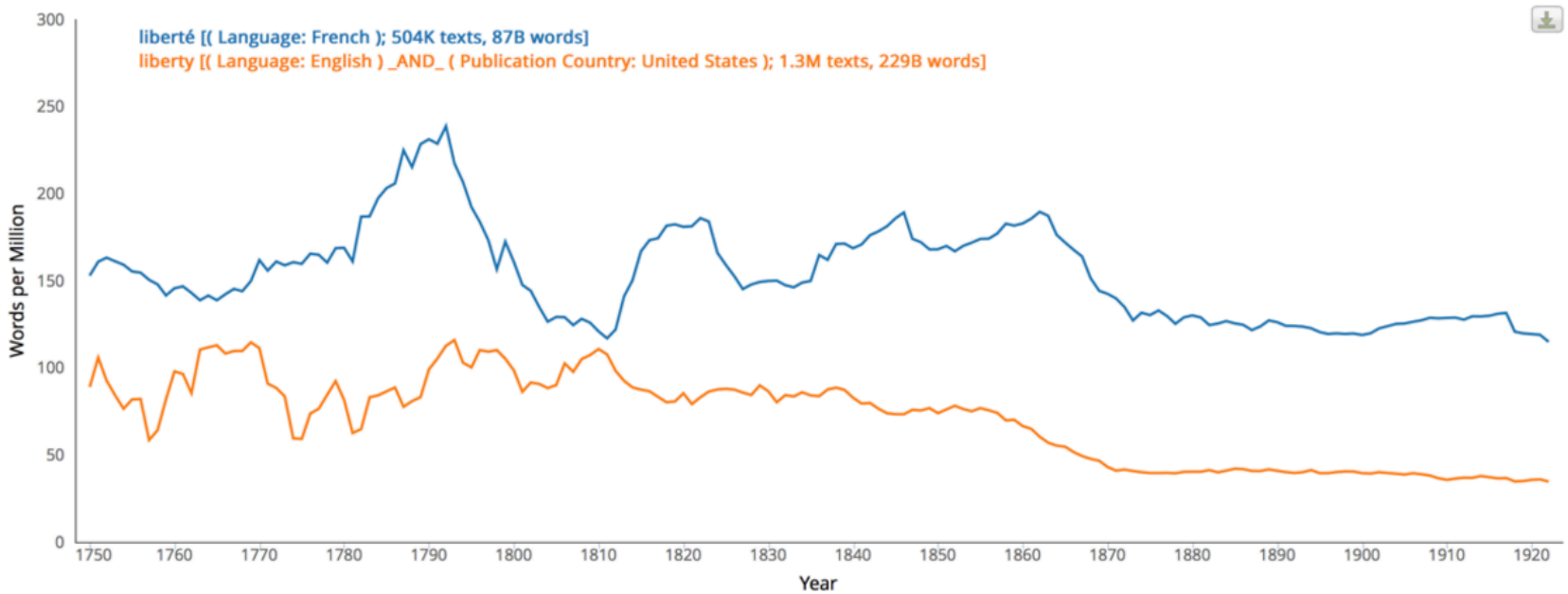
Search for *liberté* in (Language: French)
Search for *liberty* in (Language: English)
Search for *freedom* in (Language: English **)**



How occurrences of related word-concepts in multiple languages/ places compare with each other (contd.)

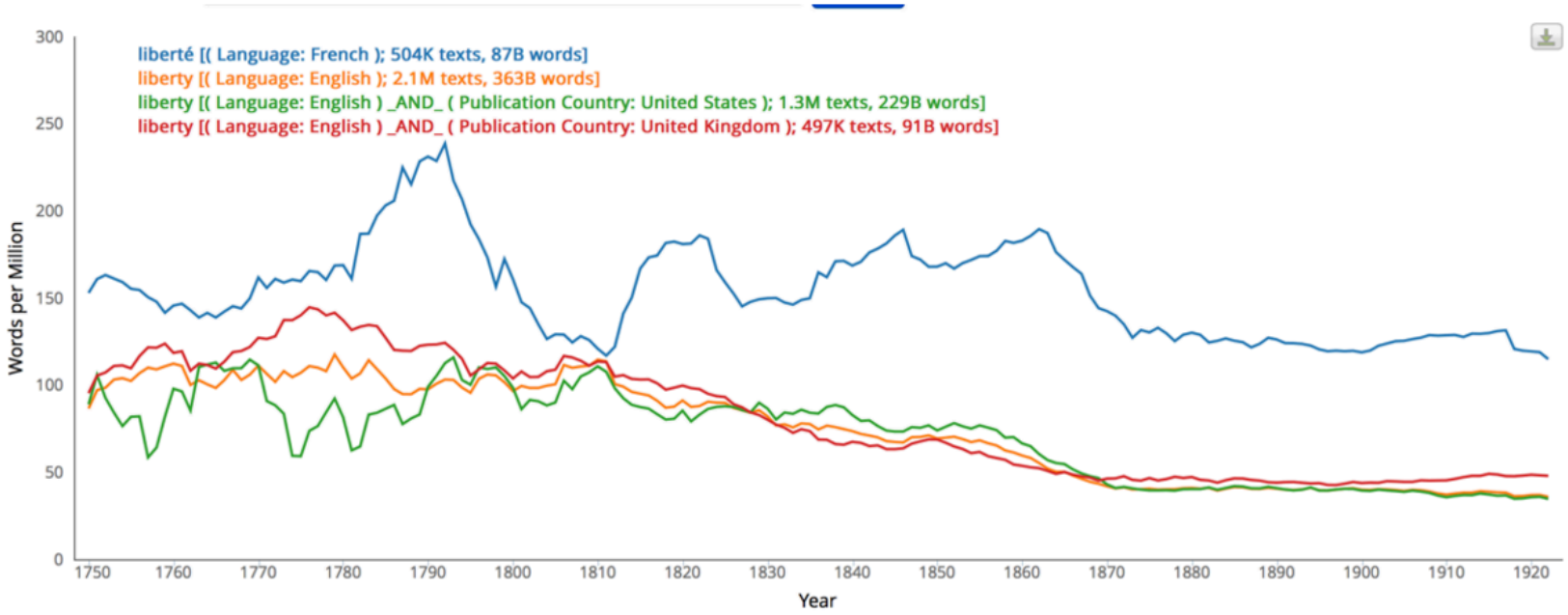
Search for *liberté* in (Language: French)

Search for *liberty* in (Language: English) AND (Publication Country: United States)



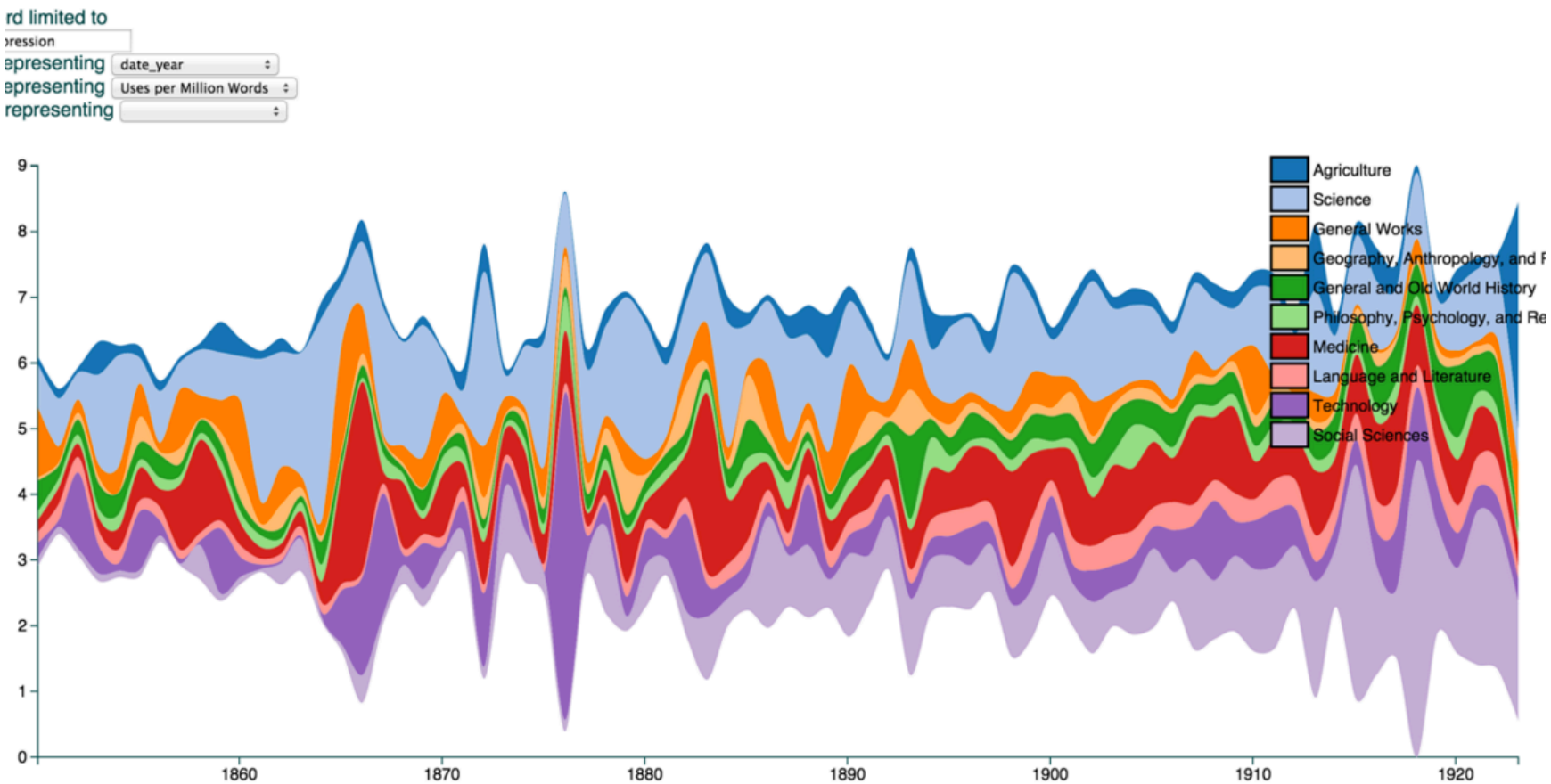
How occurrences of related word-concepts in multiple languages/ places compare with each other (contd.)

Search for liberté in (Language: French)
Search for liberty in (Language: English)
Search for liberty in (Language: English) AND (Publication Country: United States)
Search for liberty in (Language: English) AND (Publication Country: United Kingdom)



How metaphorical associations of a word vary across time

Streamgraph plot of *depression* with Library of Congress categories, between 1850 and 1923 : http://bit.ly/1QatsPs



Advanced Collaborative Support (ACS)

- CFP issued approximately once per year
- Award for dedicated HTRC developer support and/or time
- First round of ACS was in spring 2015
- Projects required access at scale, using half a million to a million volumes each

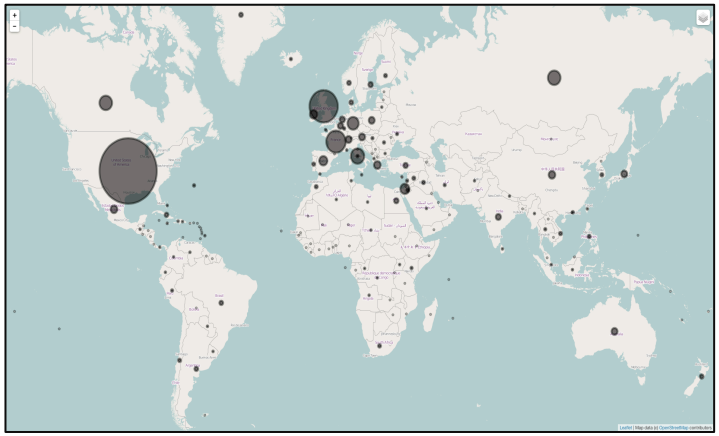
2015 Project Examples

- *Literary Geography at Scale*
  - Matthew Wilkens, University of Notre Dame
- *Off the Books*
  - Michelle Alexopoulos, University of Toronto
- *Taxonomizing the Texts: Towards Cultural-scale models of full text*
  - Colin Allen & Jaimie Murdock, Indiana University

2015 Project Examples:
Literary Geography at Scale

Matthew Wilkens, University of Notre Dame

- Geolocation of world literature
- Will extract and geocode place names from eleven million volumes in HathiTrust
- Pilot analysis completed on 10,000 randomly selected volumes
- Next step: Processing the entire corpus



2015 Project Examples: *Off the Books*

Michelle Alexopoulos, University of Toronto

- Creation of new measures of technological innovation and diffusion
- Analyzes diffusion of technology-related terms
- Will lead to mapping of waves of innovation over time and space
- Will aid in understanding how technological change produces economic change

2015 Project Examples: *Taxonomizing the Texts: towards Cultural-Scale Models of Full Text*

Colin Allen & Jaimie Murdock, Indiana University

- Tests and visualizes topic models leveraging the HTRC Data Capsule
- Shows the relationship between a topic model based on a random sample of volumes and the entire category from which it is drawn
- Proof-of-concept models Library of Congress subject headings and visualizes them using their Topic Explorer tool

Other Projects

- Word similarity tool – David Mimno

This table shows the most similar words to a given query word, for each year, in 250,000 pre-1923 books provided by the Hathi Trust Research Center. This collection is a small, random subset of the millions of digitized books held by the Hathi Trust, so these results should be used with caution. The goal of this work is, in fact, to raise questions, not answer them.

Find a word: lincoln

We represent each word with the set of words that occur on the same pages as that word. We consider two words to be similar if these sets are similar, indicating that they occur in similar contexts. More similar words are shown in darker text. Years with no dark words probably have few instances of the query word.

1800	lincoln	william	cathedral	richmond	esq	thomas	andrew	abbey	bishop	alexander	kent	charles	chapel	canterbury	windsor
1801	lincoln	henry	granted	london	edward	founded	richard	earl	william	incorporated	borough	market	knights	friars	caftle
1802	lincoln	profession	born	bred	london	doctor	school	acted	favourite	provincial	dean	business	reputation	oxford	exeter
1803	lincoln	york	county	norwich	year	named	removed	north	miles	called	england	years	church	chief	viz
1804	lincoln	bath	esq	william	county	oxford	rev.	cambridge	thomas	sept.	containing	vol	dr.	vicar	robert
1805	lincoln	york	london	king	henry	year	richard	william	oxford	england	coins	paid	granted	reign	kingdom
1806	lincoln	winchester	canterbury	rochester	salisbury	oxford	durham	peterborough	worcester	chester	exeter	dean	gloucester	wells	york
1807	lincoln	george	henry	york	richard	william	served	major	thomas	new	mr.	king	earl	parliament	defence
1808	lincoln	william	richard	james	was	named	henry	edward	bishop	norfolk	king	oxford	were	vol	other
1809	lincoln	william	kent	july	henry	john	essex	edmund	dec.	richard	may	james	died	year	thomas
1810	lincoln	college	hall	oxford	dean	chapel	founder	warner	magdalen	exeter	benefactors	winchester	trinity	durham	cambridge
1811	lincoln	william	john	clerk	jones	thomas	samuel	examined	county	year	eastern	named	college	alexander	records
1812	lincoln	middlesex	northampton	ann	talbot	kent	pari	norfolk	francis	windsor	bedford	thomas	berkshire	george	salisbury
1813	lincoln	essex	durham	william	james	london	francis	rogers	norwich	nicholas	york	norfolk	john	anthony	lord
1814	lincoln	oxford	henry	exeter	edward	robert	earl	mr.	richard	london	nicholas	john	elizabeth	church	dr.

Acknowledgements: HTRC Team

HTRC @ Illinois

(GSLIS and the University Library):

- Stephen Downie
- Beth Sandore Namachichivaya
- Tim Cole
- Loretta Auvil
- Sayan Bhattacharyya
- Boris Capitanu
- Eleanor Dickson
- Ryan Dubnicek
- Harriett Green
- Peter Organisciak
- Megan Senseney

Indiana University

- Beth Plale
- Robert McDonald
- Angela Courtney
- Nicholae Cline
- Miao Chen
- Samitha Liyanage
- Leena Unnikrishnan
- Milinda Pathirage
- Leanne Mobley

Get Involved!

HTRC Announcements:

htrc-announce-l @ list.indiana.edu

HTRC User Group:

htrc-usergroup-l @ list.indiana.edu

Questions?

Eleanor Dickson

Visiting HTRC Digital Humanities Specialist

dicksone@Illinois.edu

Sayan Bhattacharyya

Postdoctoral Research Associate, GSLIS

sayan@illinois.edu