**SLIDE: EEBO**

* **Provenance**: based on the Early English Books Online collection hosted on ProQuest
* **collection**: copied and pasted author name data from the author panel of the advance search section into an excel spreadsheet
* **cleaning**
  + used OpenRefine to consolidate author repeats and get total counts
  + hand-edited character encoding errors
* **standardize for processing**: use excel’s “text to column” feature to split the records into their data points

**SLIDE: ESTC**

* **provenance**: based on the English Short Title catalog collection in HathiTrust (not comprehensive)
* **collection**: downloaded metadata of collection records in JSON, extracted author data with Python
* **cleaning**
  + used OpenRefine to consolidate author repeats and get total counts
  + hand-edited character encoding errors
* **standardize for processing**: use excel’s “text to column” feature to split the records into their data points

**SLIDE: Open Syllabus**

* **provenance**: Open Syllabus explorer, constrained to English literature
* **collection**: XPATH queries to extract author name and count information, copied and pasted into an excel spreadsheet
* **cleaning**
  + none really needed besides occasional hand-editing
* **standardize for processing**: use excel’s “text to column” feature to split the records into their data points

**SLIDE: OTA**

* **provenance**: Oxford Text Archive catalog hosted by the University of Oxford
* **collection**: download HTML source of catalogue, run XPATH queries to extract author name and count information, copied and pasted into an excel spreadsheet
* **cleaning**
  + used OpenRefine to consolidate author repeats and get total counts
  + hand-edited character encoding errors
* **standardize for processing**: use excel’s “text to column” feature to split the records into their data points

**SLIDE: Project Gutenberg**

* **provenance**: Project Gutenberg catalog hosted by ibiblio at UNC Chapel Hill
* **collection**: download TAR archive containing RDF files of entire collection, parse through XML and extract author attribution data
* **cleaning**
  + used OpenRefine to consolidate author repeats and get total counts
  + hand-edited character encoding errors
* **standardize for processing**: use excel’s “text to column” feature to split the records into their data points

**SLIDE: Workflow overview**

Now I’ll cover the data processing workflow I followed to bring together my intermediate datasets and create my final dataset file.

**SLIDE: Workflow (red-orange)**

For each dataset, I created a Pandas DataFrame. I then concatenated all 5 DataFrames together into one large dataframe.

**SLIDE: Workflow (orange-yellow)**

My main transformation for the final dataset was consolidating author names across the five. To do this, I split the concatenated dataframe into two sub-dataframes.

First, I used the .drop\_duplicates method to create a dataframe of author names that only appeared in one dataset, or were not duplicated.

Next, I used the .duplicated method to create a dataframe of author names that appeared in more than one dataset, or were duplicated.

**SLIDE: Workflow (yellow-green)**

For the next step, I focused on the dataframe of duplicates. I used the .groupby method to consolidate the records based on the first three identification columns (first name, last name, and title). As those records were consolidated, the fraction\_total values were added together. These consolidated records were then put into a new “consolidated” dataframe.

**SLIDE: Workflow (yellow/green-blue)**

Next, I took the consolidated dataframe and recombined it with the dataframe of no duplicates to create the final concatenated version of the dataset.

**SLIDE: Workflow (blue-purple)**

To get my final values and author representation hierarchy, I divided the entire fraction\_total column of the final concatenated dataframe by five, the number of datasets, to create my final dataset.

**SLIDE: Preliminary results**

fff