Library Carpentries: OpenRefine

8/13/21

Instructor: Whitney Johnson-Freeman

Helpers: Gio Gottardi & Sarah Lynn Fisher

Introduction

- What is OpenRefine?
  - Overview of your data
  - Standardize data formatting
    - Dates: 01/01/2014 → 2014-01-01
    - Names: John M. Smith → Smith, John M.
    - Addresses: 1155 Union Cir, Denton, TX 76203
  - Split data into separate rows/columns
  - Import data from local/online sources

Introduction

- Why use OpenRefine over other tools?
  - Graphic User Interface (GUI)
  - User friendly for non-programmers
  - See changes to your data as you make them
  - Some tasks are faster then using Excel
  - History for all changes, so you can undo/redo
  - Import data from other sources (other datasets or websites)

Quick Tips/Things to Consider

- Opens in a browser, but it doesn't need an internet connection
  - Recommended: Chrome, Chromium, Opera, & Edge
  - Firefox has some minor issues, but your mileage may vary. Internet Explore isn't supported at all.
- A command line window is open while using
- Autosaves every 5 minutes*
- Don't hit the back button in your browser!
- Closing: Close all browser tabs or windows, and in the command line window hit Control + C to save last changes.

Quick Tips/Things to Consider Part 2

- You won't be modifying your original/raw data
  - Original file → OpenRefine file → Export file
- Your work lives on your computer. If you want to share your work:
  - Import/Export Project Archives (modifications/transformations/history)
  - Permalink (facets/filters/view)

Importing Data

- A short list:
  - TSV (tab-separated values)
  - CSV (comma-separated values)
  - Excel
  - JSON (javascript object notation)
  - XML (extensible markup language)
  - Google Sheets

Now Let's…

Open OpenRefine!
http://127.0.0.1:3333/

Choosing a good separator…

- Examples:
  - Jones, Andrew
  - Davis, S.
  - https://librarycarpentry.org/lc-open-refine/03-working-with-data/index.html
- Recommended for our dataset: | (pipe symbol)

Facets & Filters

- Facets
  - Text facet
  - Numeric facet
  - Timeline facet
  - Scatterplot facet
  - Custom facets
- Filters
  - Text filters
    - Like a search feature
    - Supports regular expressions (regex)

Facets

- Numeric facets
  - Sorts numbers smallest to biggest.
- Timeline facets
  - Sorts dates chronologically, but data must be formatted at dates
  - https://docs.openrefine.org/manual/exploring#dates
- Scatterplot facets
  - Compares 2 or more columns containing numeric data

Facets

- Some Custom facets
  - Word facet - this breaks down text into words and counts the number of records each word appears in
  - Duplicates facet - this results in a binary facet of ‘true’ or ‘false’. Rows appear in the ‘true’ facet if the value in the selected column is an exact match for a value in the same column in another row
  - Text length facet - creates a numeric facet based on the length (number of characters) of the text in each row for the selected column. This can be useful for spotting incorrect or unusual data in a field where specific lengths are expected (e.g. if the values are expected to be years, any row with a text length more than 4 for that column is likely to be incorrect)
  - Facet by blank - a binary facet of ‘true’ or ‘false’. Rows appear in the ‘true’ facet if they have no data present in that column. This is useful when looking for rows missing key data.

Facets & Filters

- They can stack, so you can narrow your data subset
- Exporting data with facets or filters applied will only export the matching rows.

Regex In Action

- In Publisher column:
  - \bphos

Clustering

- https://github.com/OpenRefine/OpenRefine/wiki/Clustering-In-Depth
- Key Collision Methods
  - Fingerprint
  - N-Gram Fingerprint
  - Phonetic Fingerprint
- Nearest Neighbor Methods
  - Levenshtein Distance
  - PPM

Clustering

- Key Collision Methods
  - Fingerprint
    - Least likely to produce false positives
    - Good place to start
  - N-Gram Fingerprint
  - Phonetic Fingerprint
    - Useful to spot errors of misunderstanding or misspelling

Clustering

- Nearest Neighbor Methods
  - Levenshtein Distance
    - "edit distance"
    - General applicability
  - PPM
    - Lots of false positives
    - Used as a "last resort"

Transforming Data

- Common uses include:
  - Splitting data from one column to multiple
    - Splitting an address into multiple parts.
  - Standardizing the format of data
    - Removing punctuation or standardizing date formats
  - Extracting a particular type of data from a longer string
    - Finding ISBNs in bibliographic citations

Transforming Data

- GREL
  - General Refine Expression Languages
  - Similar to Excel formulas
  - https://docs.openrefine.org/manual/grel
  - Supports regex

Transforming Data

- Some default transformations are available in menu options:
  - To Uppercase
  - To Lowercase
  - To Titlecase
  - Trim leading and trailing whitespace

Transforming Data

- Basic format of GREL expressions:
  - Value.function(options)
  - Function(value, options)
- Either works, but we'll use the first in this workshop.
- For example, Title Case:
  - Value.toTitleCase()

Transforming Data

- Based on data types:
  - String
  - Number
    - Found in common transforms
  - Date
    - ISO-8601(extended format with time in UTC: YYYY-MM-DDTHH:MM:SSZ)
    - value.toDate()
    - https://docs.openrefine.org/manual/grelfunctions#todateo-b-monthfirst-s-format1-s-format2-
  - Boolean
    - True/False
    - value.contains("test") → True or false if test appears in the cell
    - if(value.contains("test"),"Test data",value) → Replaces cell value with Test data if it contains test
  - Array

Transforming Data

- Based on data types:
  - Arrays:
    - List of values
    - Found in […]
    - Used to sort, de-duplicate, and manipulate
      - Data must be in the same cell
    - Numbering starts at 0

Transforming Data

- Arrays
  - “Monday,Tuesday,Wednesday,Thursday,Friday,Saturday,Sunday”
  - value.split(“,”)
    - [“Monday”,”Tuesday”,”Wednesday”,”Thursday”,”Friday”,”Saturday”,”Sunday”]
  - value.split(“,”).sort()
    - [“Friday”,”Monday”,”Saturday”,”Sunday”,”Thursday”,”Tuesday”,”Wednesday”]
  - value.split(“,”)[0]
    - “Monday”

Extensions

- Extensions add functionality to OpenRefine.
- Some require earlier versions of OpenRefine.
- Take a look: https://openrefine.org/download.html

Real Life Examples

- Deng, S. (2018). Linked data in the library & OpenRefine. *Faculty Scholarship and Creative Works*. https://stars.library.ucf.edu/ucfscholar/774
- Hill, K. M. (2016). In Search of Useful Collection Metadata: Using OpenRefine to Create Accurate, Complete, and Clean Title-level Collection Information. *Serials Review*, *42*(3), 222–228. https://doi.org/10.1080/00987913.2016.1214529
- Sterner, E. (2019). Cleaning Collections Data Using OpenRefine. *Issues in Science and Technology Librarianship*, *92*, Article 92. https://doi.org/10.29173/istl30
- Stonebraker, I. (2015). Good Library Data Made Better With Technology! Using OpenRefine and Google Fusion Tables in Academic Business Libraries Instruction. *Academic BRASS*. https://docs.lib.purdue.edu/lib_fsdocs/118

More Help

- *Data Cleaning with OpenRefine for Ecologists*. (n.d.). Retrieved August 12, 2021, from https://datacarpentry.org/OpenRefine-ecology-lesson/
- Feustle, M. (2017, May 18). *The Life-Changing Magic of OpenRefine: The Open-Source Art of Data Decluttering and Organizing* [Presentation]. 2017 University of North Texas Open Access Sympoisum. May 19, 2017. Frisco, TX. https://digital.library.unt.edu/ark:/67531/metadc980821/
- *Library Carpentry: OpenRefine*. (n.d.). Retrieved August 12, 2021, from https://librarycarpentry.org/lc-open-refine/
- Little, J. (n.d.). *Cleaning Data with OpenRefine*. Retrieved August 12, 2021, from https://libjohn.github.io/openrefine/preamble.html
- Najmi, A., & Keralis, S. D. C. (2014, January 29). *Cleaning up Messy Data with Open Refine* [Presentation]. Tech Talks Series, 2014, Denton, Texas, United States. https://digital.library.unt.edu/ark:/67531/metadc275785/
- *OpenRefine user manual | OpenRefine*. (n.d.). Retrieved August 12, 2021, from https://docs.openrefine.org//
- Williamson, E. P. (2017). Fetching and Parsing Data from the Web with OpenRefine. *Programming Historian*. https://programminghistorian.org/en/lessons/fetch-and-parse-data-with-openrefine