Introduction to OpenRefine

Joelen Pastva
CARLI Technical Services Spring Forum
Thursday, April 21, 2016
What is OpenRefine?
OpenRefine

- Open source, community supported and developed tool for analyzing and enhancing data
- Browser-based, but does not require internet connection and data is stored locally on machine
- Utilizes web services to interact with external data
- Can run in Linux, Mac, or Windows environment – requires Java
- Designed for tabular data, especially large data sets
- Supports TSV, CSV, Excel, XML, JSON, Google documents, and RDF files
History

May 2010
Freebase Gridworks developed by Metaweb Technologies

July 2010
Google acquires Metaweb, Freebase Gridworks becomes Google Refine

2012
Google transfers Google Refine to open source community, which becomes Open Refine

October 2013
Google no longer supports Open Refine
Applications

• Data analysis
• Data cleanup
• Data reconciliation

Especially useful for non-MARC metadata projects

*Working with MARC (and outputting back to MARC) is more challenging
Using OpenRefine
Getting started:

2. Install OpenRefine – Windows: extract ZIP file to preferred directory
3. Launch application – double-click refine.exe file

Create a project:

Select file to work with, then click “Next”
Preview import:

- Check for column headers and data types in columns
- Confirm correct data parsing
- Pay special attention to this box

- Use record preview to confirm import settings

<table>
<thead>
<tr>
<th>Title</th>
<th>Creator</th>
<th>Description</th>
<th>Date.ISO</th>
<th>Date</th>
<th>Geographic Coverage</th>
<th>Community Area</th>
<th>Subject</th>
<th>Language</th>
<th>Language Code</th>
<th>Type</th>
<th>Form</th>
<th>Physical Description</th>
<th>Physical Extent</th>
<th>Digital Format</th>
<th>Host</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Bookstore, Navy Pier Campus</td>
<td>University of Illinois at Chicago Navy Pier Campus (unidentified students, employee). Photo taken for school catalog.</td>
<td>1947-10-14</td>
<td>October 14, 1947</td>
<td>Near North Side (Chicago, Ill.)</td>
<td></td>
<td>Students; Workers; Educational Facilities; Navy Pier (Chicago, Ill.)</td>
<td>English</td>
<td>eng</td>
<td>Image</td>
<td>Photographs</td>
<td>B&amp;W/negative</td>
<td>4 x 5 in.</td>
<td>jpeg</td>
<td>University of Illinois at Chicago Library</td>
</tr>
</tbody>
</table>
# Learning the ropes:

- **Toggle between row and record view**

- **Expand data view**

- **Column menu dropdown**

- **Record or row numbers with tag options**

- **Single record viewed as a row**

### Table Example

<table>
<thead>
<tr>
<th>No.</th>
<th>Title</th>
<th>Description</th>
<th>DateISO</th>
<th>Date</th>
<th>Geographic Code</th>
<th>Community Area</th>
<th>Language</th>
<th>Language Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>University of Illinois at Chicago Circle, Board of Trustees</td>
<td>Bookstore at University of Illinois at Chicago Navy Pier Campus (unidentified student, employee). Photo taken for school catalog.</td>
<td>1947-10-14</td>
<td>October 14, 1947</td>
<td>Near North Side (Chicago, Ill.)</td>
<td></td>
<td>English</td>
<td>eng</td>
</tr>
<tr>
<td>2</td>
<td>Accounting, Navy Pier Campus</td>
<td>Accounting at University of Illinois at Chicago Navy Pier Campus (unidentified student, faculty member). Photo taken for school catalog.</td>
<td>1947-10-14</td>
<td>October 14, 1947</td>
<td>Near North Side (Chicago, Ill.)</td>
<td>Students; Teachers; Educational Facilities; Navy Pier (Chicago, Ill.)</td>
<td>English</td>
<td>eng</td>
</tr>
</tbody>
</table>
Analyzing data:

**Text filter** – filter column values based on input

**Text facet** – displays number of occurrences of each unique value in column, similar to Excel’s Filter.

Can edit at the facet level.
Analyzing data:

Split column
- Create multiple columns based on data in single column
- Can split by separator value or field lengths, specify maximum number of new columns to create, and delete or retain original column
- Useful for making data more granular (ex: Name → LastName, Firstname)

Four columns created from original
Analyzing data:

Split multi-value cells
- Useful when metadata fields repeat and are grouped in one field
- Will prompt for character used to separate values (semi-colon (;) used in this data)
- Remember to join multi-value cells before export!

Record view displays split values in new “rows” within record
Analyzing data:

Don’t be afraid to play around with other facets to see different views of your data!

Undo/Redo window lets you go back step by step, or all the way to the beginning if needed. You can also reuse these steps again with a different data set by using the Extract and Apply features.
Working with data:

Cluster and edit

Groups similar terms based on algorithm

Helpful for identifying spelling variants, singular vs. plural, and other input variants for normalization
Working with data:

Cluster and edit

Explore different methods and functions to identify possible matches

Merge matches or input new cell values to edit all cells at once
Working with data:

Advanced functions for editing cell values

Advanced users can experiment with using Refine Expression Language (GREL) and regular expressions to perform more complicated functions to edit cell data.

• Edit cells—Transform...

See https://github.com/OpenRefine/OpenRefine/wiki/Recipes for common operations and pre-built code “recipes.”
Reconciling data:

- Allows OpenRefine to compare your data with structured RDF data (locally or externally) to identify matches.

- Use when trying to match against controlled vocabularies.

- Before reconciling, you must install the RDF extension in OpenRefine and establish a reconciliation service so that OpenRefine knows where to look for data to compare.
Reconciling data:

The RDF extension is available for download here: http://openrefine.org/download.html

To install:
1. Download and unpack RDF extension.
2. Launch OpenRefine, click on Browse workspace directory in bottom left (for 2.6: find AppData/Local/OpenRefine folder).
3. Locate folder “extensions,” or create one.
4. Copy unpacked folder into “extensions” folder.
5. Restart OpenRefine browser and application, new RDF button should appear in top right corner.
Reconciling data:

Setting up a reconciliation service – two common options are to use SPARQL endpoints or to upload an RDF file locally.

SPARQL endpoints can be used when a vocabulary is hosted externally as RDF in its own namespace and is too large to download.

See http://freeyourmetadata.org/reconciliation/ for helpful instructions for using SPARQL endpoints for LCSH.

Use the RDF file option if you have a locally created vocabulary that can be serialized as RDF.
Reconciling data:

Example: using a local thesaurus of genre terms, structured using SKOS/RDF (http://metadataregistry.org/uri/punk#)

View of a single concept or term in the thesaurus:

```
<!-- Concept: Afro-punk -->
<skos:Concept rdf:about="http://metadataregistry.org/uri/Punk/1002" xml:lang="en">
    <skos:prefLabel xml:lang="en">Afro-punk</skos:prefLabel>
    <skos:altLabel xml:lang="en">Afropunk</skos:altLabel>
    <skos:altLabel xml:lang="en">Afro punk</skos:altLabel>
</skos:Concept>
```

Open Refine asks for a name for the reconciliation service, a location for the RDF file, and the properties used to label concepts in the thesaurus. In this case, SKOS is selected.
Reconciling data:

Select a column of terms to compare and start reconciling

Once a reconciliation service is selected, OpenRefine will check for matches. It sees that some of the data matches the skos:Concept terms, which is the intended result.

Click “Start Reconciling” to continue, then wait…
Reconciling data:

After reconciliation, it is necessary to review the results to confirm matches and select others that may have been overlooked.

Summary of total matches found

Suggested matches sorted by top possibility w/option to search manually for a match or add a topic

Matched term with link to URI for term in thesaurus

Facet window shows number of terms matched and not matched
Reconciling data:

- Can quickly implement controlled terms over a large data set.

- Linking terms with their corresponding URLs is the first step to turning data into linked data.

- URLs for terms can be separated out into new column for future work.
  - Edit—Add column based on this column
  - Give column new name (e.g. Term URLs)
  - Expression: cell.recon.match.id
MarcEdit and OpenRefine:

The latest version of MarcEdit (6) includes a toolset to better integrate with OpenRefine for importing and exporting MARC data, which were previously complicated operations.

Go to Tools—OpenRefine
• Options exist for importing OpenRefine projects and for exporting data from MarcEdit to OpenRefine.

See Terry Reese’s blog for further information, including step-by-step instructions, screenshots, and a demo video:

http://blog.reeset.net/archives/1873
Questions?

Joelen Pastva
jpastva@uic.edu

