OpenRefine Instructor Workshop Notes

**Introduction:**

OpenRefine is described as "a power tool for working with messy data" [David Huynh]

Great for simple files, tab, or comma delimited that has internal inconsistencies, formats, terminology

Things it’s great for:

overview of data/analysis of data per column

\* Resolve inconsistencies (date format)

\* Split data into separate cells (multiple names, address parts)

\* Matching data, like your local data to L of C data

\* Enhance data with data from other sources

Scenarios:

\* how many times does a value appear in a column?

\* How is data distributed across the dataset?

\* Changing all data in a column into one format

\* splitting data into separate columns

\* adding data

**Creating Project**

Doaj file

UTF-8

First line is column header

DO NOT check parse text cells into numbers

Records vs. rows

Record # displayed

Work with 1 column at a time, show dropdown

One row per record in current mode

**Splitting cells**

Author column

Edit cells->Split multi-valued cells

Change to record mode, note changed numbering

Joining cells

Author column

`Edit cells->Join multi-valued cells`

Note changes

Good separators – why not a comma. Choose one that is NOT in your data values

Challenge:

1. What separator character is used in the Subjects cells?

2. How would you split these subject words into individual cells?

Challenge:

Join them back together

GREEN STICKY

**Faceting**

Used for overview of the data and creating consistency

Text is the simplest, appears in left-hand column

Publisher column

`Facet -> Text Facet`.

Include includes the values in the facet

Invert shows the values not selected by the filter

Challenge:

1. Which licences are used for articles in this file? (ccby)

2. How many articles in the file don't have a licence assigned? (6)

**Filtering**

It looks for a particular piece of text in a column.

Choose filter>>text filter

You can use regular expressions in a filter

IMPORTANT: anything you do while in a filter only applies to that filter.

**MORE FACETS**

Go through all the facet types

Show how to edit on a fact using the License column

Challenge:

Find all publications without a DOI (use facet by blank). Click on True

You can amend groups of data via a facet. Show edit feature.

Show how to edit on a fact using the License column

Challenge: Normalize the language values. (change English to EN)

**Clustering**

Clustering gathers together inconsistent values and lets you merge them into a value that you choose. Incredibly helpful with names, geographical places, etc.

Show the clustering function on the publisher column

Challenge: use clustering to clean up author data. Remember, it will easier if you first split the multi-value cells into separate rows.

Try the key collision and fingerprint algorithms first and do some cleaning up. Once you’ve got things tidy, explore the other options and see what they do.

**Working with columns**

Upper left functions

Sort function – can turn off or on

**Intro to transformations**

Facets and filteres are great, but sometimes you need to go further:

* Splitting data that is in a single column into multiple columns (e.g. splitting an address into multiple parts)
* Standardising the format of data in a column without changing the values (e.g. removing punctuation or standardising a date format)
* Extracting a particular type of data from a longer text string (e.g. finding ISBNs in a bibliographic citation)

Create a text facet on the Publisher column

Note that in the values there are two that look identical - why does this value appear twice?

On the publisher column use the dropdown menu to select Edit cells->Common transforms->Trim leading and trailing whitespace

Look at the publisher facet now - has it changed? (if it hasn’t changed try clicking the Refresh option to make sure it updates)

**Writing transformations**

Go to Publications,

Edit cells->Transform

Can use

value.function(options)

function(value, options)

We will use the first one.

To put titles into title case

Facet by publisher

Akshantala Enterprises" and "Society of Pharmaceutical Technocrats"

Click dropdown on title column

Edit cells->Transform

value.toTitlecase()

Check preview

Demonstrate undo, redo and extract.

**Grel and data types**

To be effective in using GREL for transformations, you need to understand the different data types that can be present.

The most common is the “string”, which is a piece of text.

But there are others, for example, date.

Remove all facets and filters

Go to date column

Edit cells->Common transforms->To date

Edit column->Add column based on this column

'New column name' type "Formatted Date"

'Expression' box type the GREL expression

value.toString("dd MMMM yyyy")

**Booleans**

You can create a Boolean text fact.

Facet->Custom Text Facet

Type value.contains(“Complex”)

Discard facet.

This can be used for more powerful grel expressions. For example, if I wanted to replace the word “Complex” with “Simple” I could use the expression

if(value.contains("Complex"),"Simple",value)

Before we do this next demo, make sure that you’re author names are still separated in their own cells.

Authors

Facet->Custom text facet

value.contains(",")

Next lesson, we’re going to learn about arrays so that we can reverse these names.

**Arrays**

See lesson for array explanation

Note the quotes around the strings. This means that now OR can deal with each of these as a separate string, not just one. Therefore you can perform grell functions like sort on them.

Select “true” in the Boolean facet box.

Author column

Edit cells->Transform

value.match(/(.\*),(.\*)/)

See how that creates an array

value.match(/(.\*),(.\*)/).reverse().join(" ")

Click OK

Explain Export function

**Looking up data from a url**

OpenRefine can retrieve data from URLs. Ask who has done authority or reconciliation work? OpenRefine is the tool for you.

Typically this is a two step process - firstly a step to retrieve data from a remote service, and secondly to extract the relevant information from the data you have retrieved.

To retrieve data from an external source, from the drop down menu at a column heading use the option 'Edit column->Add column by fetching URLs'.

Data retrieved stored in new cell

We are going to retrieve ISSNs from the Crossref API

Normally we would run this for the entire set, but since we don’t want to crash anything, we’ll each pick a row.

Clicking the star icon for the relevant row in the first column

Facet by Star

Choose the single row

ISSN column

Edit column->Add column by fetching URLs

Name column Journal details

<http://api.crossref.org/journals/+value>

We should now have a long string of stuff in the new column. This is the return of the request from the API

Under new column

Edit column->Add column based on this column'

Journal Title

value.parseJson().message.title

The reason we added a new column is to preserve the JSON output in case you want to extract more from it later.

OR has a parser for parsing more of this data, and the link to learn more about that is in the lessons.

**Reconciliation**

Reconciliation services allow you to lookup terms from your data in OpenRefine against external services, and use values from the external services in your data.

USE SLIDES

Publisher

Reconcile->Start Reconciling

Click 'Add Standard Service...'

<http://refine.codefork.com/reconcile/viaf>

Click on VIAF Reconciliation Service

Choose “Corporate Name”

Unclick auto match

Start reconciling

You should get

>\* Publisher: Judgement

>\* Publisher: best candidate's score

These are two of many

Close Publisher: best candidate's score

Create text facet on publisher column

International Union of Crystallography

If confident in all:

Reconcile->Actions->Match each cell to its best candidate

Awesome right

Also extract ID

Edit column->Add column based on this column...'

VIAF ID

cell.recon.match.id

Extensions:

SLIDES