Beyond Excel: how to start cleaning data with OpenRefine

Within our different roles as information professionals, we are all expected to handle larger and larger amounts of data, from the resources we manage to the analytics we collect. However as this data gets bigger it can become harder to analyse. Ham explains that this is often due to errors and inconsistencies in the collection and management of data (2013, p.233), not to mention the time involved in learning how to analyse all of this information, along with the analysis itself. The following guide hopes to address some of these issues by introducing readers to OpenRefine (formerly Google Refine), an open source piece of software that can help to remove some of the errors and inconsistencies in datasets, in a timely manner, without expert knowledge being required.

The guide is inspired by a Library Carpentry course organised by Dr. James Baker in November 2015; an exploratory programme of software skills training aimed at librarians that was held in the Centre for Information Science at City University London. In addition to being introduced to a range of software skills (Playforth, 2015), participants were encouraged to disseminate what they had learnt, which in this case has progressed from discussions with colleagues to the writing of this guide.

A number of practical blog-posts can already be found about the use of OpenRefine in Libraries. Margaret Heller will be running an online course for the Library Juice Academy later in the year and has written several posts on the subject including 'A Librarian's Guide to Open Refine' (2013). Owen Stephens, who delivered the excellent OpenRefine session on the Library Carpentry course, has also published a number of blog posts including a worked example of fixing problem MARC data with OpenRefine (2015). Other case-studies describe how to use the software for tasks such as analysing usage logs (Bedoya, 2014), reconciling subject headings in EAD (Huffman, 2015), and augmenting metadata in a Library Repository (Tillman, 2016). Some institutions are going even further, providing training for staff and PhD students in "Cleaning messy data with OpenRefine" (University of Leicester, 2016).

Despite this, OpenRefine does not seem to be as widely used as it could be, either by staff or students. In addition to further promoting OpenRefine to practitioners, this guide will add another case-study to the literature, giving the example of one particular process we are developing at the University of Sussex Library: the removal of non-identical duplicates from a list of over 58,000 journal titles (something that is not possible with Excel's 'Remove Duplicates' function). Although this is a very specific task, removing duplicates is something that many will need to do with their particular dataset and illustrates the power of OpenRefine.

To begin, OpenRefine will need to be downloaded and installed. This can be done at: https://github.com/OpenRefine/OpenRefine/wiki/Installation-instructions.
Although it uses a web browser for an interface (old browsers may need updating to work), it is not actually connected to the internet. This means that once downloaded, it is possible to manipulate data held locally without being online.

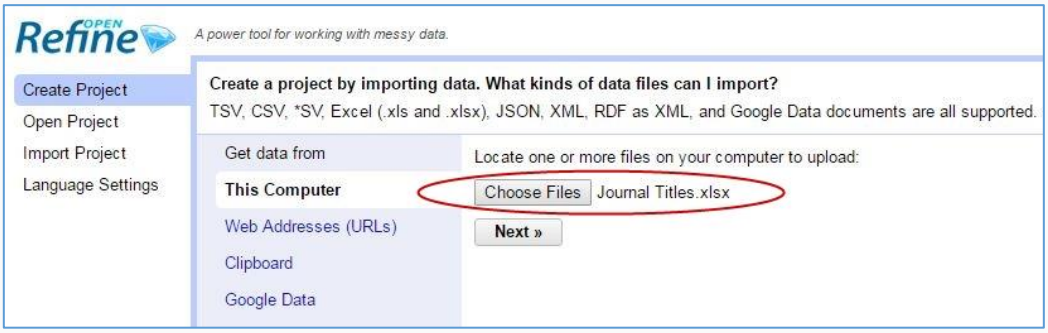
The following list of journal titles is a very small subset of the data that was left after using Excel's 'Remove Duplicates' function and shows some of the limitations of Excel (namely that special characters and punctuation stop duplicates from being recognised):

AQ : journal of contemporary analysis.
AQ journal of contemporary analysis.
Accounting, auditing and accountability
Accounting, auditing & accountability
Acta sociologica
Acta sociológica /

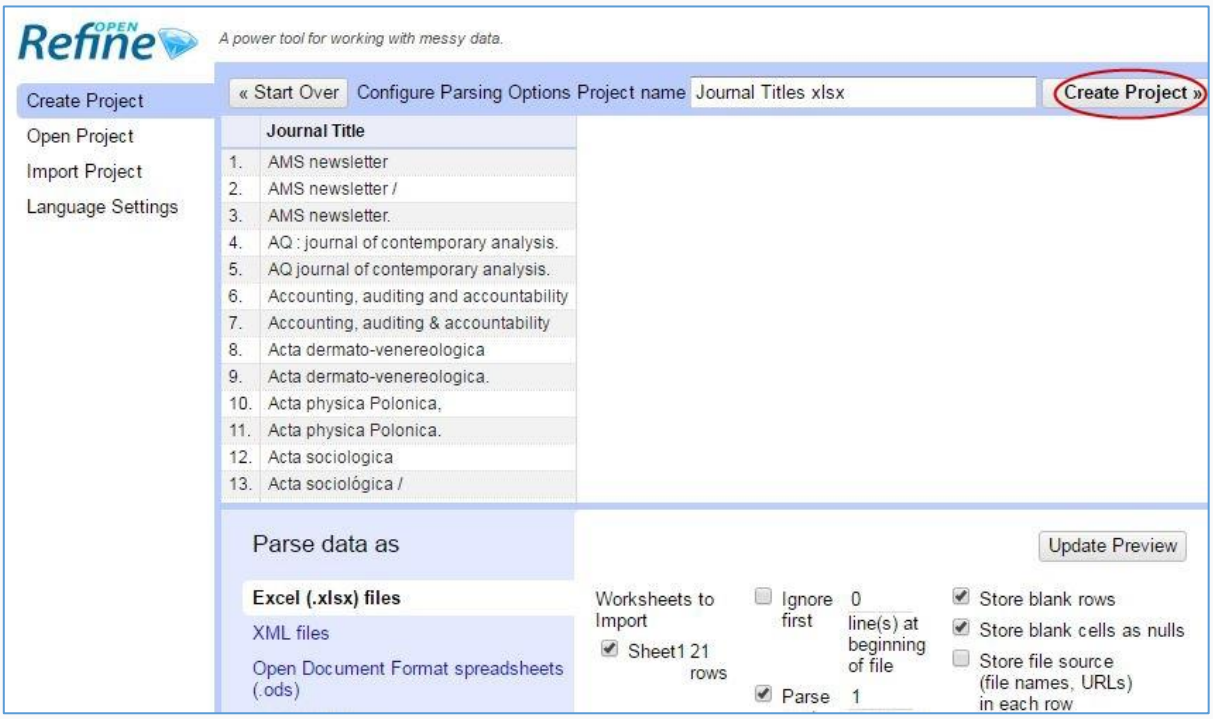
The following steps will show how OpenRefine can be used to clean this data more effectively.

Step 1: Create a project

Select **'Create Project'** and '**Choose Files'** to import the desired dataset. OpenRefine will manipulate a copy of this data not the original file:



Once you have chosen the file click on **'Next'**.
When the data has uploaded you will be shown a preview of the project:

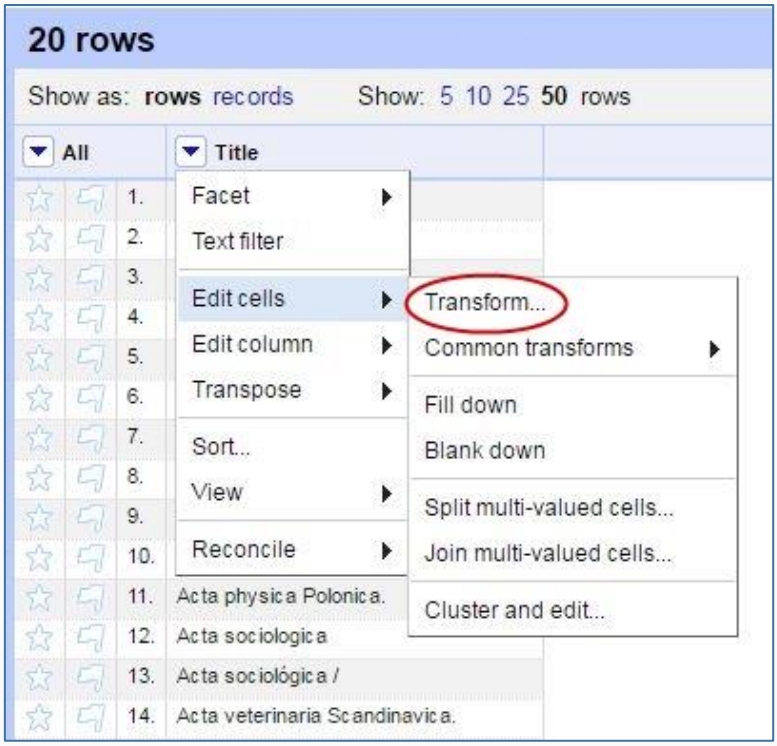


There are default options at the bottom of the page which can be changed to alter the data that is imported; do not adjust these. Edit the project name if you wish and click on **'Create Project'**, both at the top right of the screen.

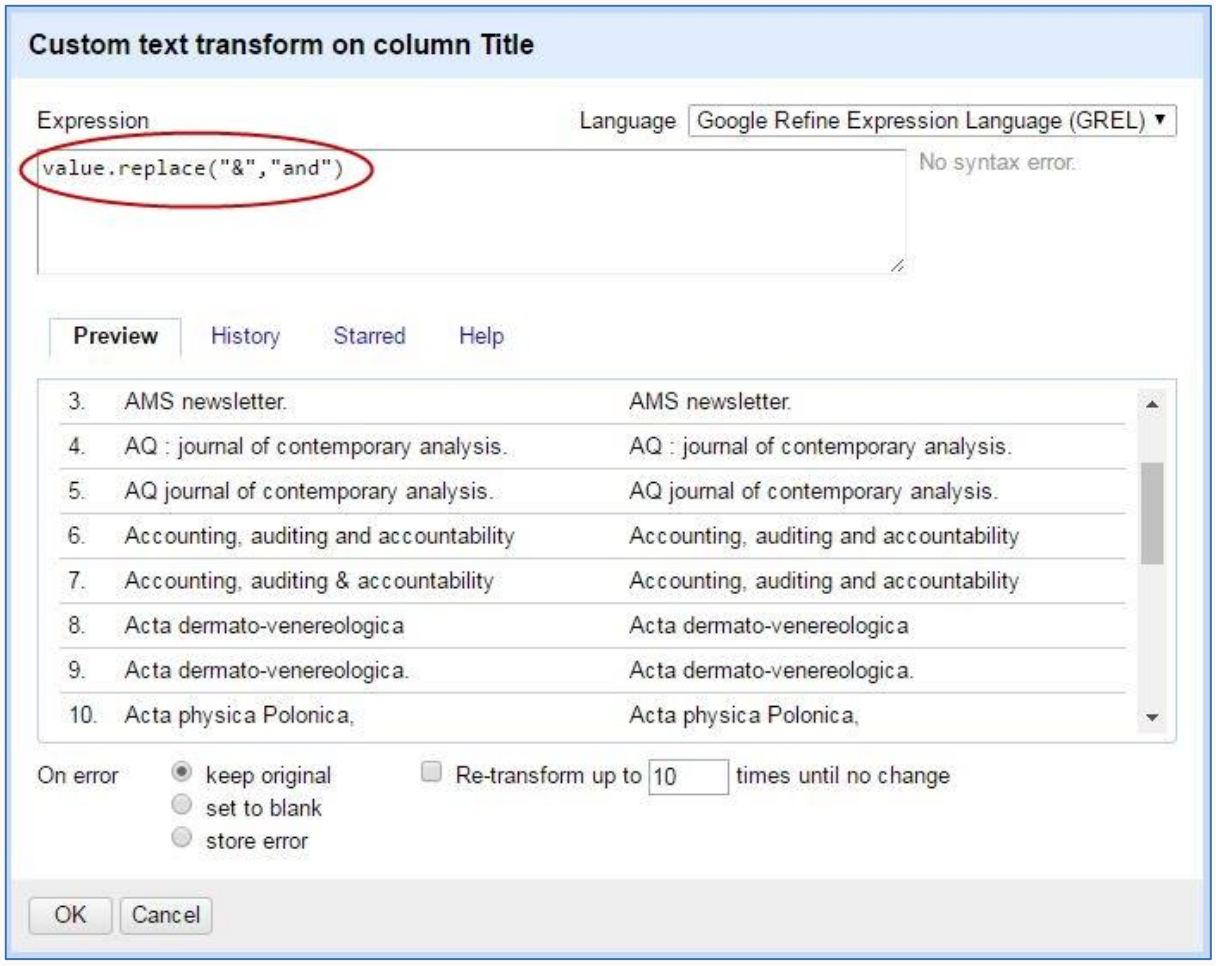
Step 2: Use GREL to replace ampersands

GREL (Google Refine Expression Language) is the programming language of OpenRefine, use of which can enable greater manipulation of data. In this example GREL will be used to replace all instances of **'&'** with **'and'.** More information about GREL can be found at:
https://github.com/OpenRefine/OpenRefine/wiki/General-Refine-Expression-Language

Begin by clicking on the downward arrow next to the title of the desired column (in this case **'Title'**).
Select **'Edit cells'** followed by **'Transform...'**:



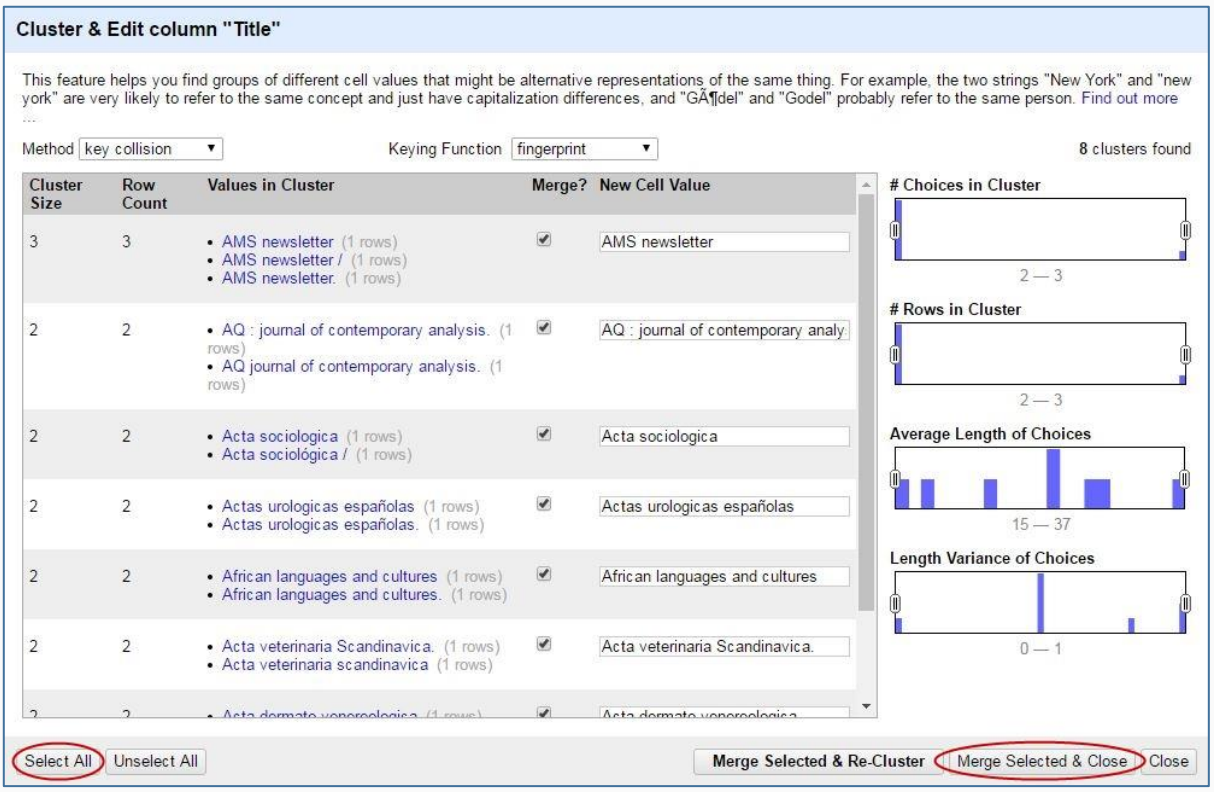
The following pop-up window will appear. In the **'Expression'** box enter the GREL command: **value.replace("&","and").** This expression is telling OpenRefine to replace what appears in quotation marks before the comma, with what appears after:



The preview screen shows how the data will appear after this transformation; with **'and'** replacing **'&'** wherever it appears. Click **'OK'** to complete this process.

Step 3: Cluster similar pieces of data

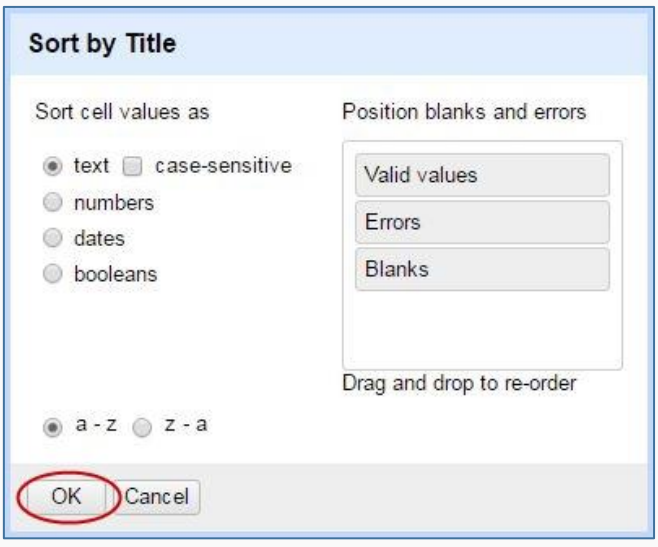
Click on the arrow next to the **'Title'** header again. This time select **'Edit cells'** and **'Cluster and edit...'.** You will now be presented with a number of clustering options. The default method is the **'key collision: fingerprint'** function which works by removing all punctuation and control characters in addition to normalising western characters (Morris, 2012). Titles are grouped together and a **'New Cell Value'** suggested, which each title in the cluster will be changed to.



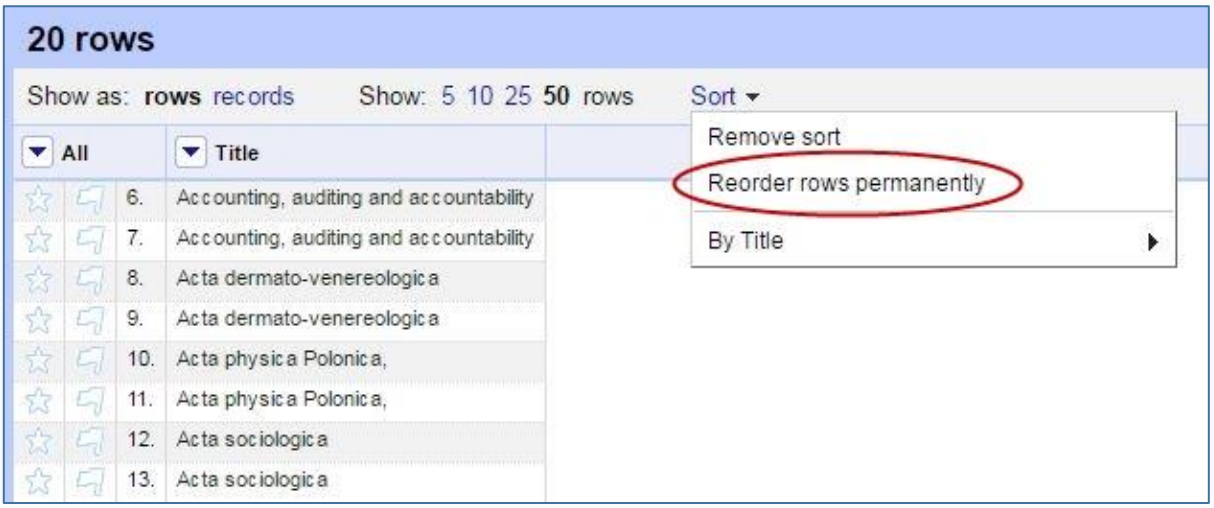
Click **'Select All'** and **'Merge Selected & Close'.**

Step 4: Sort and reorder

Click on the arrow next to the **'Title'** header and select **'Sort...'.** The following pop-up window will appear. Click **'OK'.**



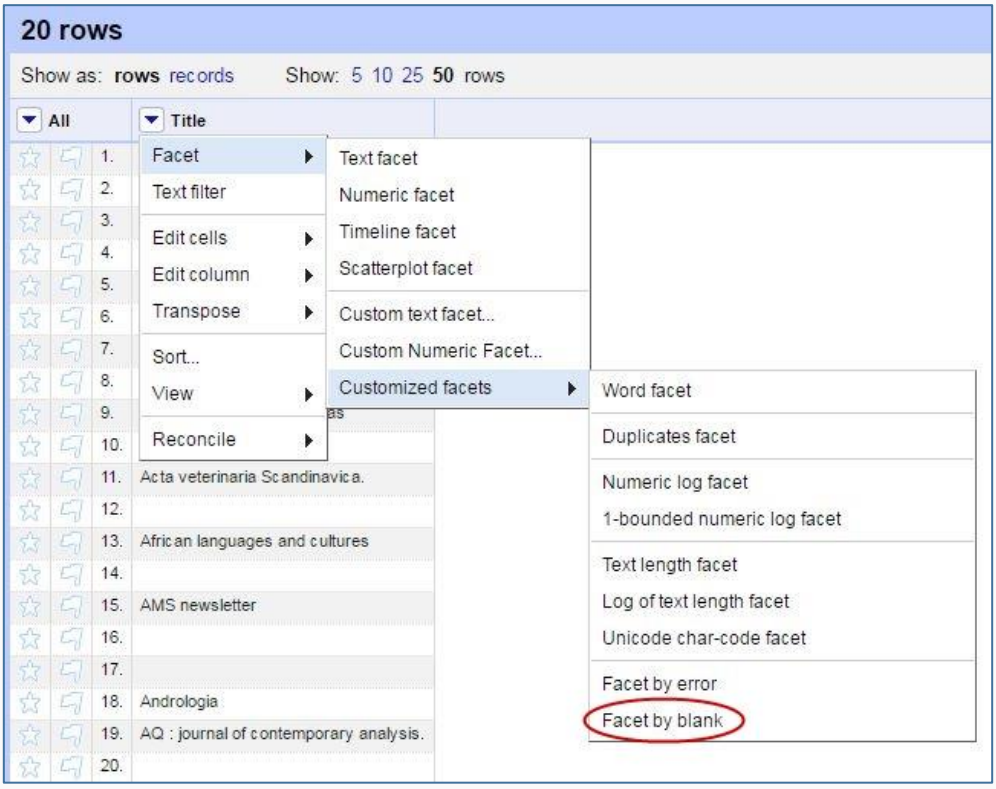
A new **'Sort'** option will become visible at the top of the page. Click on the arrow next to this and then on **'Reorder rows permanently'**:



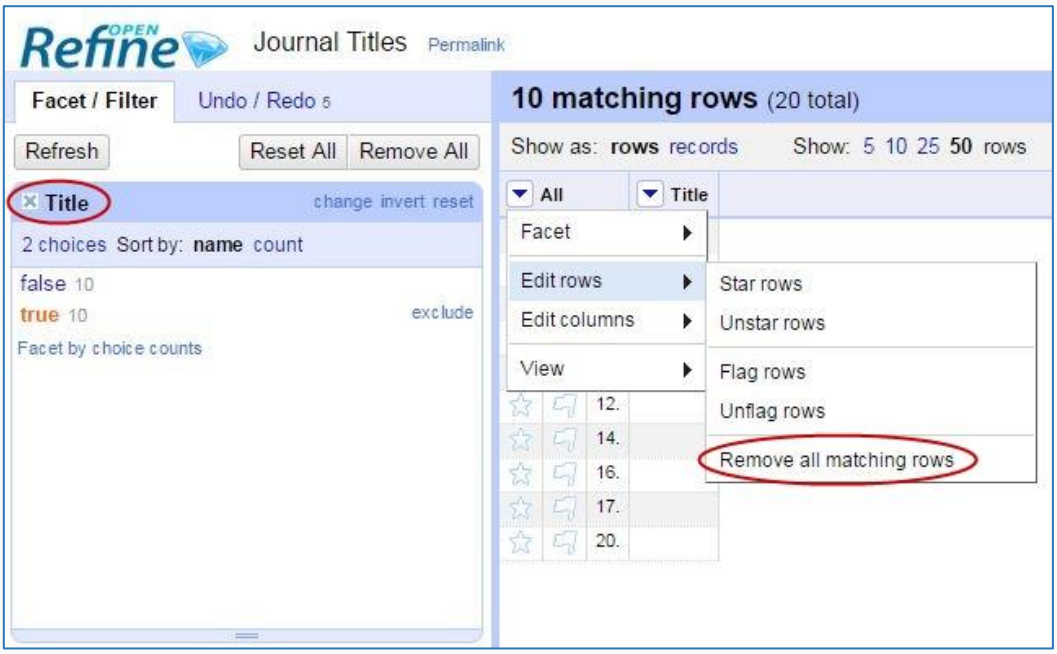
Step 5: Remove duplicates

Click on the arrow next to the **'Title'** header, select **'Edit cells'** and **'Blank down'.** If there is a cluster of four identical titles, the blank down function will keep the top entry and turn the three beneath it to blank cells. This means there should now only be one instance of each title.

These blank cells now need to be removed. Click on the arrow next to **'Title'**, select **'Facet'**, **'Customized facets'** and **'Facet by blank':**



Select **'true'** from the facet that will have appeared on the left of the page. This will display just the list of blank cells. Click on the arrow next to the **'All'** column and select **'Edit rows'**, followed by **'Remove all matching rows':**



Remove the facet by clicking **'X'.** All blank cells will have been removed and you will now see the list of non-duplicated titles.

Step 6: Remove trailing punctuation with Regex

Although not essential to the process of removing duplicates, this final step in cleaning the data will remove the trailing punctuation that appears after some titles, quickly making the list more presentable if it is being shared with colleagues. We will finish, as we started, by using GREL. However this particular command will also contain a Regular Expression, another type of programming language that is supported by GREL. As in Step 2, click on the arrow next to the **'Title'** header, select **'Edit cells'** and then **'Transform...'**. This time, when the pop-up window appears, type the following command into the **'Expression'** box: **value.replace(/\W$/,“ ”)**

The forward slashes contain the regular expression; **\W** denotes any non-word character and **$** the end of a line. The expression is therefore telling OpenRefine to replace any non-word character at the end of a line with nothing, as nothing is entered between the quotation marks. Click **OK** to complete this process and remove all trailing punctuation.

To export this new de-duplicated dataset, click on the **Export** option on the top right of the screen and select the desired file type, in this case for Excel:

	A	B	C	D	E
1	Title				
2	Accounting, auditing and accountability				
3	Acta dermato-venereologica				
4	Acta physica Polonica				
5	Acta sociologica				
6	Actas urologicas españolas				
7	Acta veterinaria Scandinavica				
8	African languages and cultures				
9	AMS newsletter				
10	Andrologia				
11	AQ : journal of contemporary analysis				

Conclusion

Although this is an extremely small sample, it is possible to clean much larger datasets in this way; there is a thriving community on GitHub and at www.openrefine.org where further instruction and support can be found.

One thing to be aware of with this particular process, and something that should always be considered, is the potential for erroneous transformations. For example, with the method used above, 'Business and Society Review' and 'Society and Business Review' will be clustered together. However there is often more than one way to achieve a particular task using OpenRefine and it is designed to complement other tools, so find a way that suits your needs best.

Because OpenRefine is working on a copy of the dataset, it is possible to experiment with different transformations knowing that the original data is safe. The second tab in the **'Facet/Filter'** column also allows any step to be undone, so the next time you are wrangling data try using OpenRefine. As Ham concludes, "the only way to know if OpenRefine is right in a particular setting is to try it" (2013, p.234).

References

Bedoya, J. (2014) 'Analyzing Usage Logs with OpenRefine', *ACRL TechConnect Blog*, 7 May. Available at: http://acrl.ala.org/techconnect/post/analyzing-usage-logs-with-openrefine (Accessed: 5 April 2016)

Ham, K. (2013) 'Electronic Resources Reviews: OpenRefine (version 2.5)', *Journal of the Medical Library Association*, 101(3), pp.233-234.

Heller, M. (2013) 'A Librarian's Guide to OpenRefine', *ACRL TechConnect Blog*, 1 May. Available at: http://acrl.ala.org/techconnect/post/a-librarians-guide-to-openrefine (Accessed: 3 April 2016)

Huffman, N. (2015) 'Adventures in metadata hygiene: using Open Refine, XSLT, and Excel to dedup and reconcile name and subject headings in EAD', *Bitstreams,* 1 May. Available at: http://blogs.library.duke.edu/bitstreams/2015/05/01/metadata-adventures/ (Accessed: 5 April 2016)

Morris, T. (2012) *Clustering In Depth.* Available at: https://github.com/OpenRefine/OpenRefine/wiki/Clustering-In-Depth (Accessed: 9 April 2016)

Playforth, C. (2015) 'Why the information profession needs Library Carpentry', *Software Sustainability Institute*, 3 December. Available at: http://software.ac.uk/blog/2015-12-03-why-information-profession-needs-library-carpentry-0?bw (Accessed: 14 December 2015)

Stephens, O. (2015) 'A worked example of fixing problem MARC data: Part 1 – The Problem', *Overdue Ideas,* 13 July. Available at: http://www.meanboyfriend.com/overdue_ideas/2015/07/worked-example-fixing-marc-data-1/ (Accessed: 3 April 2016)

Tillman, R. (2016) 'Extracting, Augmenting, and Updating Metadata in Fedora 3 and 4 Using a Local OpenRefine Reconciliation Service', *The Code4Lib Journal*, 31. Available at: http://journal.code4lib.org/articles/11179 (Accessed: 5 April 2016)

University of Leicester (2016) *PhD Students/Staff: Cleaning messy data with OpenRefine* [Advertisement]. Available at: http://www2.le.ac.uk/library/help/workshops/events/phd-students-staff-cleaning-messy-data-with-openrefine (Accessed: 5 April 2016)