Generating an Excel file from a PDF using OpenRefine

Introduction

This is a tutorial for generating a table based on a standardized .pdf file. It will provide instructions on how to download and effectively use OpenRefine and some of its operations. Most of the tools used in this tutorial are free and open-source. One of tools is not open-source, but it is free.

Tools Needed

OpenRefine - OpenRefine, previously named Google Refine, is an extremely powerful tool for working with messy data, cleaning it up, and transforming it into different formats. It can be extended with a variety of web services and external data. While it is executed in a browser, OpenRefine actually runs locally on the host machine. As a result, none of the data manipulated is uploaded online, ensuring security for the user's private data.

Any Text Editor (**Sublime Text** or **Notepad++*** recommended) - The pdf's text has to be pre-formatted before being uploaded to OpenRefine, so I would recommend using Sublime for ease. Any text editor would be acceptable, but this tutorial does a quick shortcut only available on more sophisticated text editors. The shortcuts used work on both Sublime and Notepad++. Sublime is technically not free, but has a free indefinite trial period. *Notepad++ is only available for Windows.

Any PDF Reader – The tutorial needs a pdf reader to open the given pdf file. It should allow the user to select the text on the document. Beyond that, there are no additional requirements. A PDF reader typically comes installed on the operating system, so downloading another one is not usually required.

Downloading

Links are not explicitly named in this tutorial because these programs still have regular updates. Each of these programs can be found simply by searching: "[program name] download [your operating system]".

OpenRefine is rather straightforward to download. Just download the latest version of OpenRefine using the appropriate kit. More specific instructions are located right next to the download links.

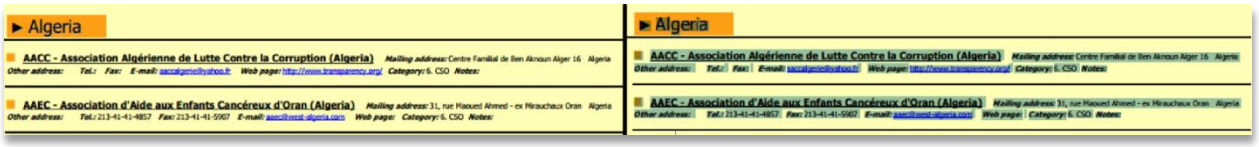
Sublime is a very popular text editor. It provides syntax highlighting and a high level of customizability. We will be using none of those features in this tutorial. Just hit the link for your operating system. Notepad++ even more powerful though less stylish,

with the same perks of syntax highlighting and customizability. Once again, we won't be using that. Just download the link called Notepad++ Installer on their website.

If the pdf reader is not already installed, then Foxit Reader is recommended. It is lightweight, fast, and secure. Additionally is available for most common operating systems. However, nine out of ten times, the pre-installed pdf reader would suffice.

Getting Started

The first step in the process is to get the raw text from the pdf file. Open the pdf and click anywhere on the text. Then click Ctrl-a (Whenever Ctrl is referenced, just assume that it refers to Command on OS X) on the keyboard to select all the text.



1 Highlighting all the text in a pdf

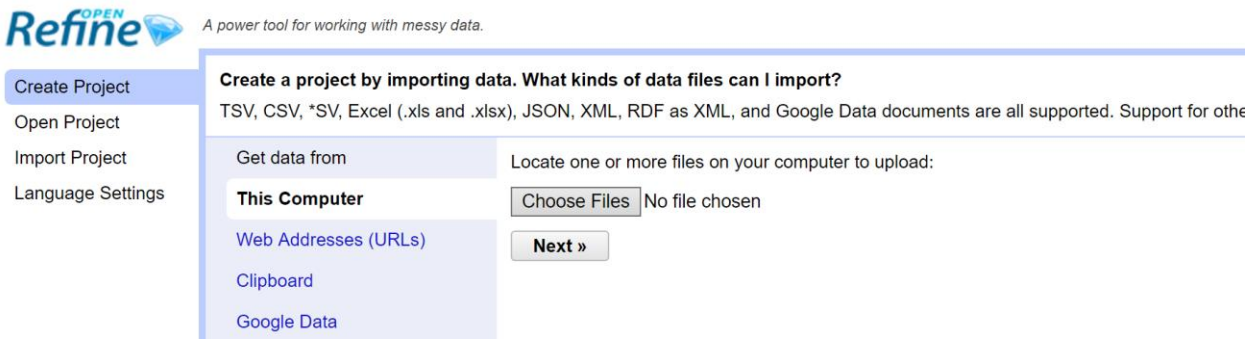
Then copy and paste (Ctrl-c and Ctrl-v) that text into a text editor. It should be really long and look really messy. This is because it is the raw text from the pdf. There should also be a section at the beginning without values in them. This is the introduction to the paper, which can be deleted. There are also headers and footers that need to be found and deleted. Do not remove the square symbols that precede the data, as these will be useful later. Use the Ctrl-f function to find each of the headers and footers as well. All non-data text should be deleted. Sublime has a useful find-all function for text.

2 The raw text copied from the pdf

The final manipulation to the text is to put it all on a single line. The shortcut for this Ctrl-A Ctrl-J. This is a very tasking operation, so Notepad++ is recommended. Save it as a text file.

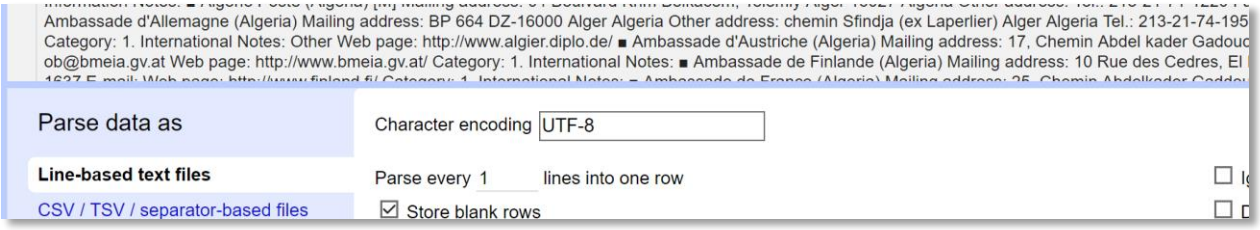
Importing files with OpenRefine

Open OpenRefine.exe, which should automatically open your browser. Click on the Create Project tab and upload the text file.



3 Create the project

After OpenRefine loads the text file, it generates a preview. Notice that the text in the preview is sometimes a bit messed up. This is because the character encoding is not specified. The unusual characters are OpenRefine trying to interpret special characters without knowing the encoding. Click on UTF-8. This should fix foreign characters.

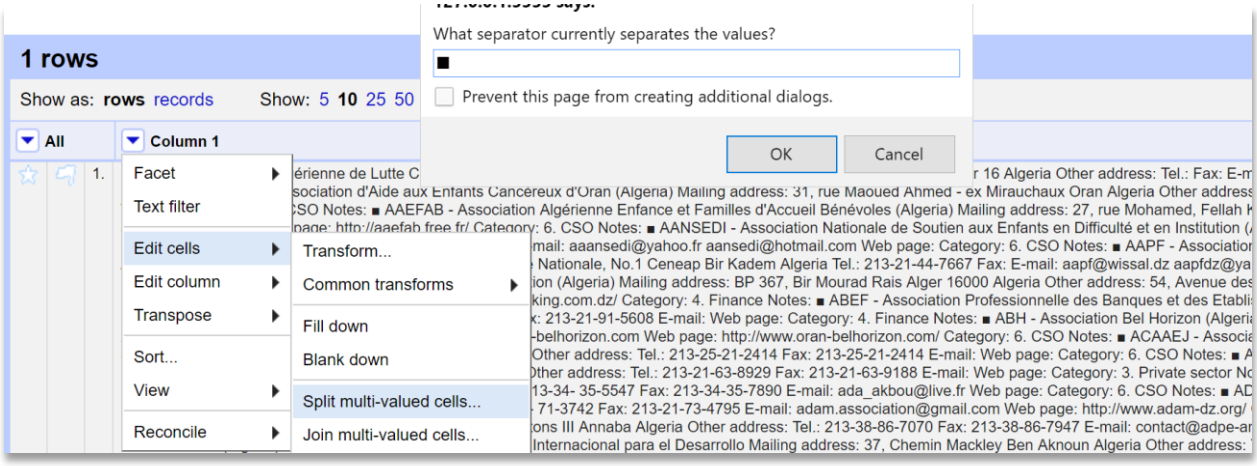


4 Changing character encoding

After that, name the project in the upper left corner and create the project.

Manipulating Data with OpenRefine

OpenRefine will be very slow after the table is created. This is because it is loading a table with a single cell is several thousand words in it. The first step to to break the data up into rows. In Column 1, click on the dropdown menu (the dropdown menu refers to the little down arrow). Go to Edit cells -> Split multi-valued cells. When the dialog box appears, copy and paste the separator character (in this case a black box) and press OK.

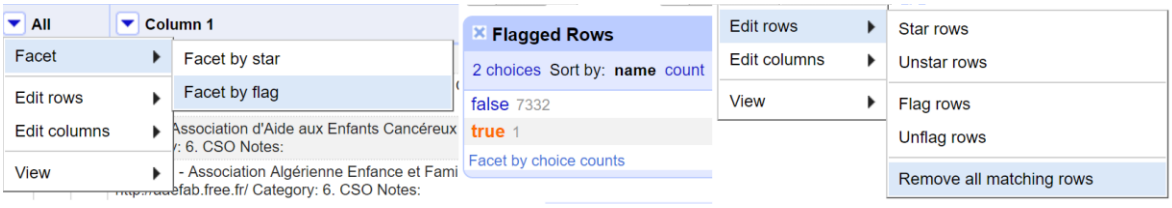


5 Separating the single cell into several rows

After this operation, OpenRefine should run much faster. The rows should be also be separated now.

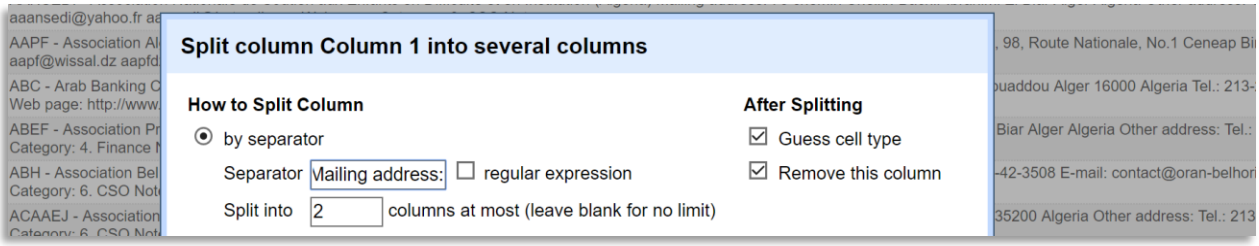
6 Data is separated into rows

Notice that there is an empty row at the top of the table. This can be removed by clicking on the flag icon on the row. Click the All dropdown menu and go to Facet -> Facet by flag. A new box should show up on the left side. Click on true. Now only the empty row should be visible. Go back to the All dropdown and go to Edit rows -> Remove all matching rows. Exit out of the Flagged Rows box and the table should return without the empty row at the top.

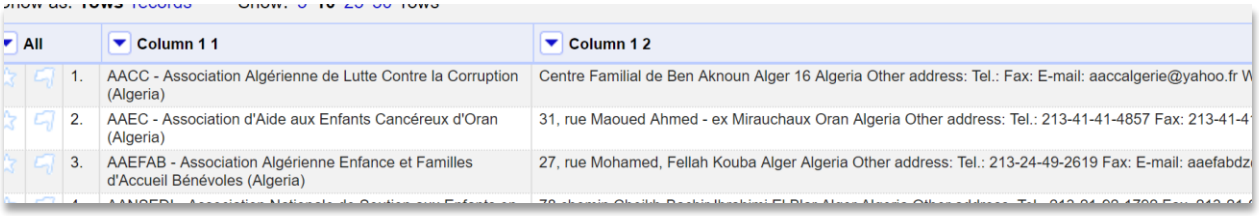


7 Removing top empty row

The next step is to separate the values into different columns. Click on the dropdown next to Column 1, go to Edit column -> Split into several columns… Specify the separator as the title of the next column. This will change over each column. Also be sure to specify that it should split into 2 columns at most.

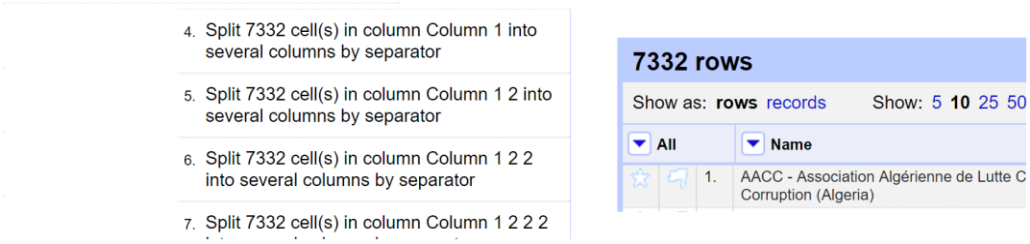


8 Split column into several columns



9 Now the table has two columns

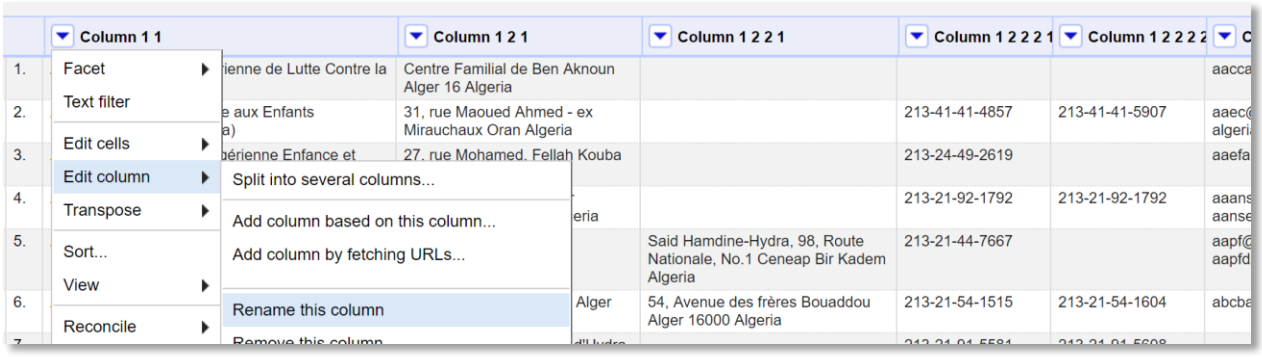
Repeat this process for every column in the table. After that, each value should be separated. Be sure the number of cells manipulated is equal to the number of rows in the table. This means that each row is changed when the columns are split. If the number of cells changed do not match the number of rows, then check the FAQ section below.



10 Be sure each operation matches the number of rows

Formatting and Exporting from OpenRefine

Now that all the data is properly split into columns, it needs to be properly labeled. Use the dropdown menu on each column, go to Edit column -> Rename this column and change the name. Repeat for every column.



11 Renaming each column

Now that the data is all properly formatted, export it using the top right button labeled Export… OpenRefine can export as an Excel file, tsv, csv, and other data types. There is no limit to how many time a dataset can be exported. Once exported, the data set is formatted and complete. If there are any plans to perform the same synthesis on any future datasets, then go to the Undo/Redo tab and click Extract… Copy the text to another text document and save it for future use. This is highly recommended. Even if there are no plans to replicate this in the future, a small amount of data in a text file could potentially save a lot of work in the future.

FAQs

Why are all the characters (tildes, accents) messed up?

This means that the file was imported with the wrong text settings. Please refer to the Importing files with OpenRefine section. The solution to this is creating a new project and repeating all the operations again. Fortunately, there is a shortcut to this in the Repeating Extraction on Same Formatted Files section below.

I have a file that is the exact same format as the last pdf I generated, is there any shortcut to skip all the OpenRefine steps?

Why yes there is! This question deserves its own section (called Repeating extraction on Same Formatted Files).

I've completed separating values for most lines, but some of the lines are still lumped together. Why is that?

Unfortunately, this means that some of the changes applied to some rows did not apply to other rows, which messed up all future operations as well. An example of this would be if each row had an initialism and a title such as "ABBAB – Always Be Berating and Belittling". A common operation would be to split these into two different columns, however if there were rows without the initialism then it would not apply the changes to that row. If there were a row with just "Phrasing", then each future change made to the other rows would not apply to this row. Given that, it

is important to make the operations as general as possible. The goal is to make each of the changes apply to every single row. Fixing this would entail creating a new project with the same source file and repeating the same extraction with very minor modification. See the splitting rows problem below for instructions on how to fix it.

There is a row I don't want, how to I get rid of it?

The quickest way to get rid of a row is to flag it, by clicking the flag icon next to the star on the far left side of the row, and to click on the dropdown menu next to All. Go to Facet -> Facet by flag. On the far left, there should be a new panel called Flagged Rows. Click on True. Now there should be only the flagged rows. Go to the All dropdown menu again. Go to Edit rows -> Remove all matching rows. Now exit out of the facet panel on the far left and the table should return without the rows.

How do I reorder or remove columns from my table?

In the top left dropdown next to All, click Edit columns -> Re-order / remove columns… Now rearrange the columns in the left side or move it to the right side to remove it.

I split a column, but more than one column showed up. Why is that?

More than one column appearing is due to unspecified column split count. When splitting the column, be sure to specify 2 columns. Otherwise, OpenRefine defaults to 0, which is as many columns as it deems necessary.

Oh no, I made a mistake and I want to undo it. How do I do that?

To undo in OpenRefine, click on the Undo/Redo tab on the far left side. Then click on the action before the mistake. This should revert the table to its previous form. The redo the action, click on the desired redone action. One thing to keep in mind is that any actions done after an undo function will overwrite functions previously undone.

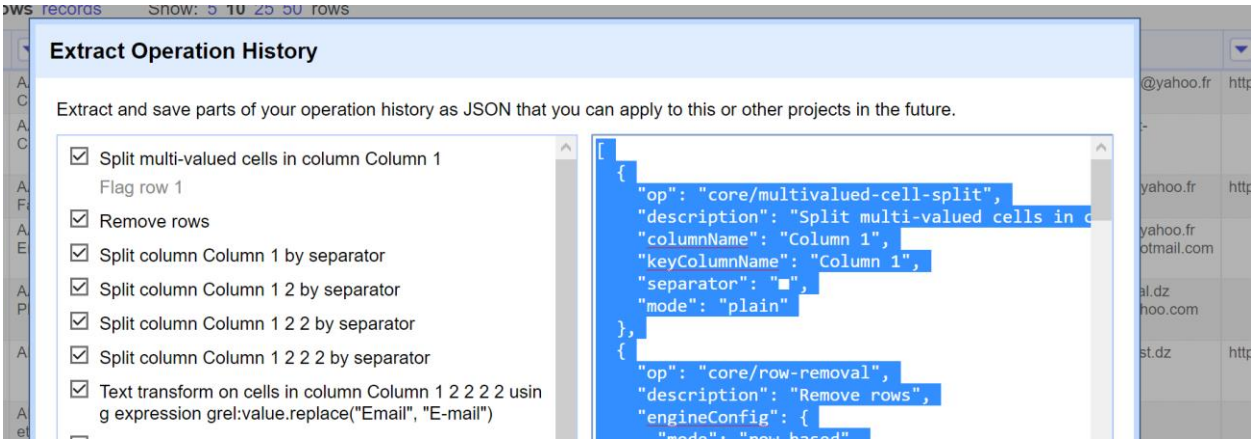
I am splitting rows, but some of the rows aren't splitting. I don't know which ones or why. How do I fix this?

The first step is to check the previous actions. If some of the actions are not occurring on all rows, then the problem may be with the action before. Revert to two actions before the number of cells changed got smaller. From there facet (view the rows of a feature that the user specifies) the latest row to see if it contains the splitting title. For example, if the next row was to represent "E-mail", then one should facet the latest column to see if the term "E-mail" is in all rows. Faceting is done by clicking on the dropdown menu of the latest column, then selecting Facet -> Custom text facet… From there, a menu should appear with a field named Expression. In expression there should be the word 'value'. Replace value with 'value.contains("Whatever the next split title is")'. Click OK and a new panel should appear on the left. If the value is in every single row, then there should be only one category. If so, then perform the

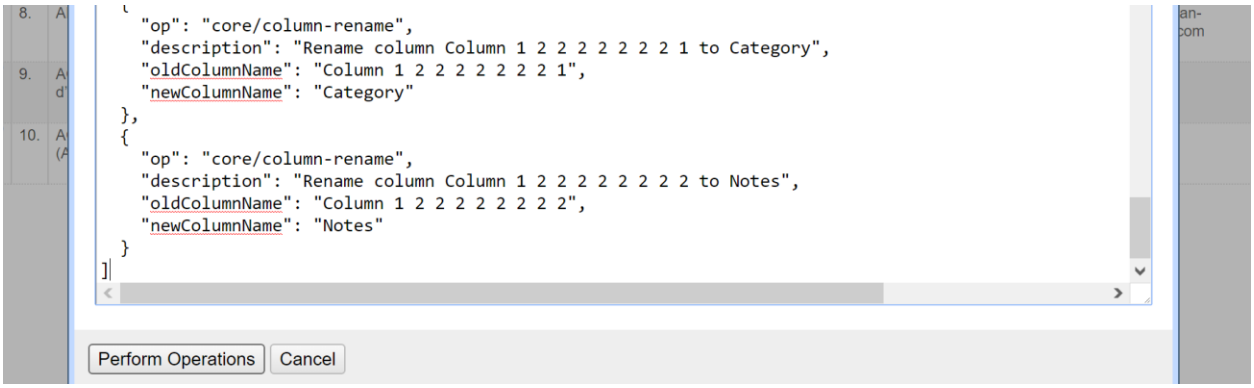
column split and perform the previous steps again with the newest column. If the value is not in every single row, then there should be two distinct categories, either True or False/none. Select the smaller one to view the values that are unusual. Often times this is due to a variable being mislabeled (e.g. “Email” instead of “E-mail”). If there is such an error, then this can be fixed by replacing all of the mislabeled values with the correct standardized one. Click on the dropdown arrow in the latest column, select Edit cells -> Transform… Another menu should appear, similar to the previous menu. In the expression field, replace ‘value’ with ‘value.replace(“The incorrect value”, “Your standardized value”)’. Be sure to put quotation marks around the changed values. Now, when the split is performed, it should apply across all rows.

Repeating Extraction on Same Formatted Files

This process is one of the reasons that OpenRefine is so powerful. It cuts redundant actions down in the blink of an eye. It is slightly more technical, but bear with it because it is rather useful. It does require some preparation. When the original document is completed, be sure to save the text from ‘Extract…’ in the Undo/Redo tab. This is essential because one the file is closed, the manipulation data is gone. Unfortunately, if the text is not saved, then the user has to do the process manually again. If the text from ‘Extract…’ is saved, then when the text file is uploaded into the single cell, the user just needs to go to Undo/Redo, click the ‘Apply…’ button and then copy the text back into the field. If the file is the exact same format, then it should transfer all the changes over.



12 Extracting the operation history



13 Copying operation history to new file

Understanding Extracted Operation History

If one wants to change the operation history, then s/he must have at least very basic knowledge about it. There are several blocks (within {}) inside the operation history. These have details of what operations occurred. The details for each of the actions vary depending on what operation happens. If one wants to remove an action from the operation history, then s/he should delete the entire block in which the operation occurs. Each block has a description field for the user's readability. The machine does not do anything with those sections. Each detail of the operation is recorded like a dictionary. In a dictionary, there are words and definitions. It is similar to the code, which has words, which in this case are variables called keys, and it has definitions, which are the values that those keys correspond to. Keys should typically not be altered, but values could be. For example, if the user recognizes an error in which s/he forgot to specify the number of maximum columns, then s/he could change the "maxColumns" value from 0 to 2. However, when manipulations like that are made, it could have unintended side effects. If the user changes a value that is later referenced (e.g. removing a column that is later referenced by another column) then it could change the build operations. The keys are usually descriptive enough to give a general understanding of which values they are supposed to represent.

Contact Us

For any questions, comments, or concerns, email Dan.Jelf@hotmail.com.