INSTRUCCIONES Y EJEMPLOS DE USO DE OPENREFINE EN LA BNE
(VERSIÓN 2019)

OpenRefine es una herramienta, de código abierto, utilizada para la limpieza y transformación de datos. Se puede descargar e instalar en cualquier equipo, si bien hay que considerar que para trabajar con ficheros pesados, que contengan gran cantidad de datos, es posible que el equipo requiera tener una memoria de entre 8 y 16 GB de RAM.

Enlace a la descarga: http://openrefine.org/download.html

Instalación: descomprimir en C:/OpenRefine y ejecutar el .exe

1. Datos a utilizar

Los conjuntos de datos que vamos a utilizar serán, en principio, los catálogos bibliográfico y de autoridades. Estos conjuntos, originariamente en formato MARC y MARC-XML, han pasado por un proceso de mapeo para ser publicados en formatos reutilizables. De este modo, cada campo y subcampo MARC aparecen ahora bajo una serie de literales que hemos definido.

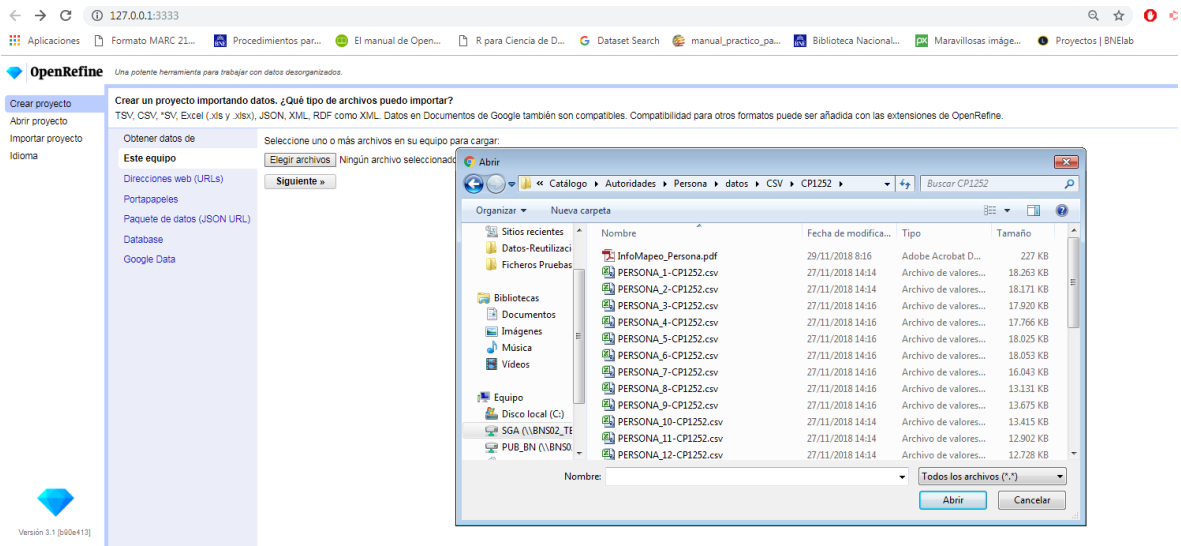
Los conjuntos de datos, la matriz de mapeo, así como las explicaciones asociadas, están ya publicadas en la web de BNElab:

- Catálogo bibliográfico
- Catálogo de autoridades

2. Crear un proyecto

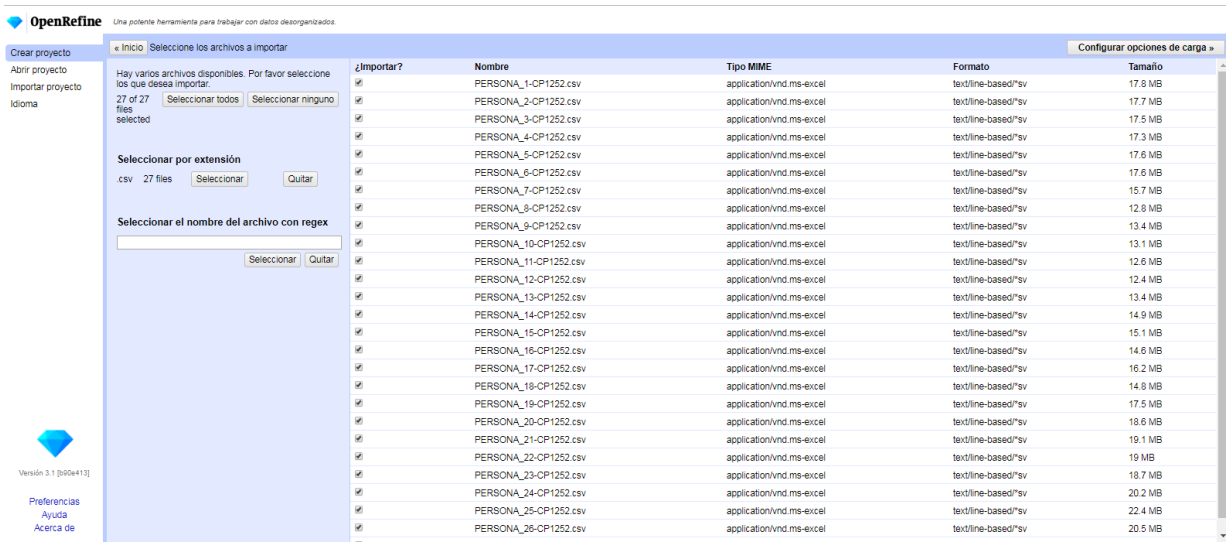
Para empezar, debemos crear un proyecto, cargando los datos con los que queremos trabajar.

1. En nuestro ejemplo, utilizaríamos los ficheros en formato CSV que hemos generado como datos abiertos; concretamente vamos a cargar todo el catálogo de autoridades de persona.



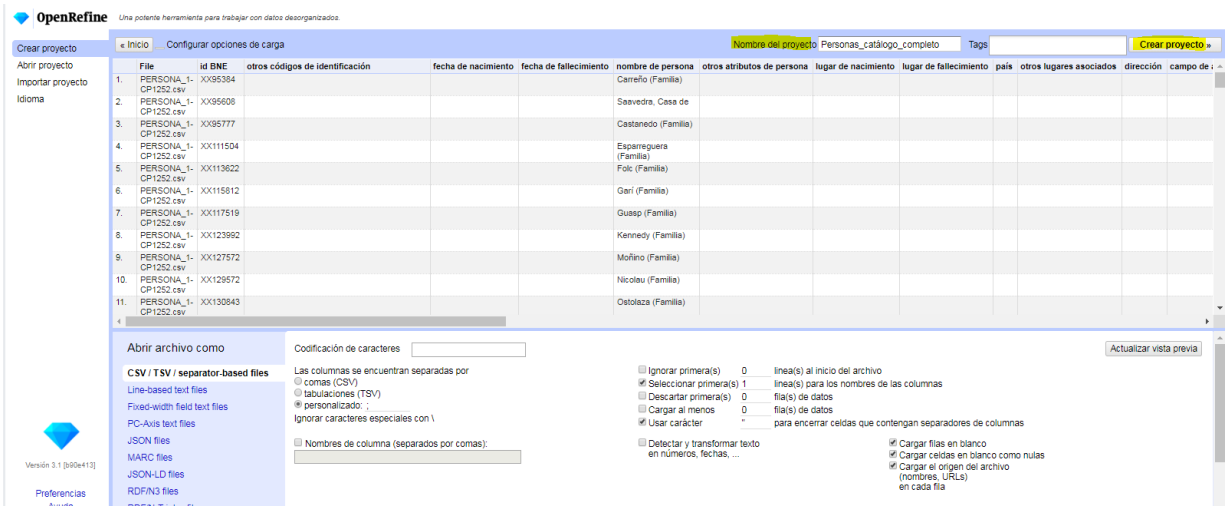
2. Para que los datos sean manejables, hemos divido el fichero de origen, en ficheros de 50.000 registros. Podemos seleccionar todos los ficheros de forma simultánea, de un mismo tipo. Por ejemplo, cargamos los 27 ficheros de autoridades persona.

3. OpenRefine nos muestra los ficheros que ha cargado, su formato, nombre, tamaño, etc. El siguiente paso será configurar las opciones de carga.

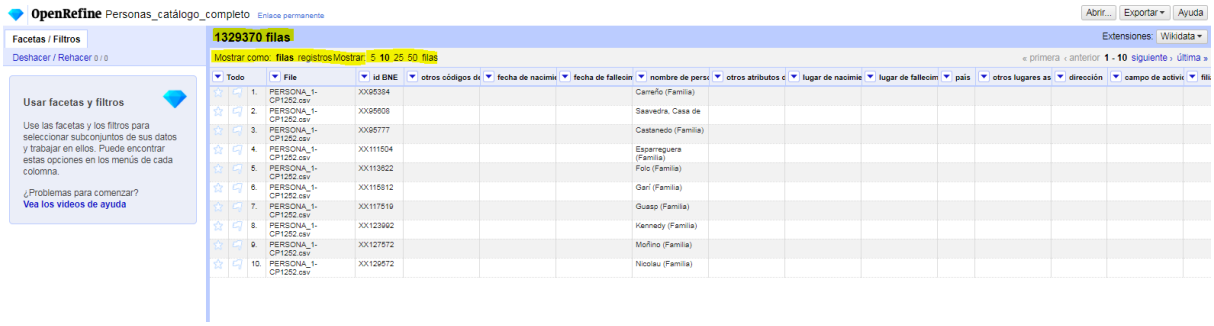


4. Ahora podemos darle un nombre al proyecto y veremos las columnas que creará, que se corresponden con los datos de nuestro mapeo[1]. Pulsaríamos en “Crear proyecto”.

[1] *La información sobre el mapeo realizado para pasar los catálogos del MARC-XML a los formatos reutilizables, está disponible al descargar cualquier conjunto en datos.gob.es, bajo el nombre “InfoMapeo”.*



5. En este punto, ya tenemos creado nuestro proyecto en OpenRefine para poder trabajar con él.

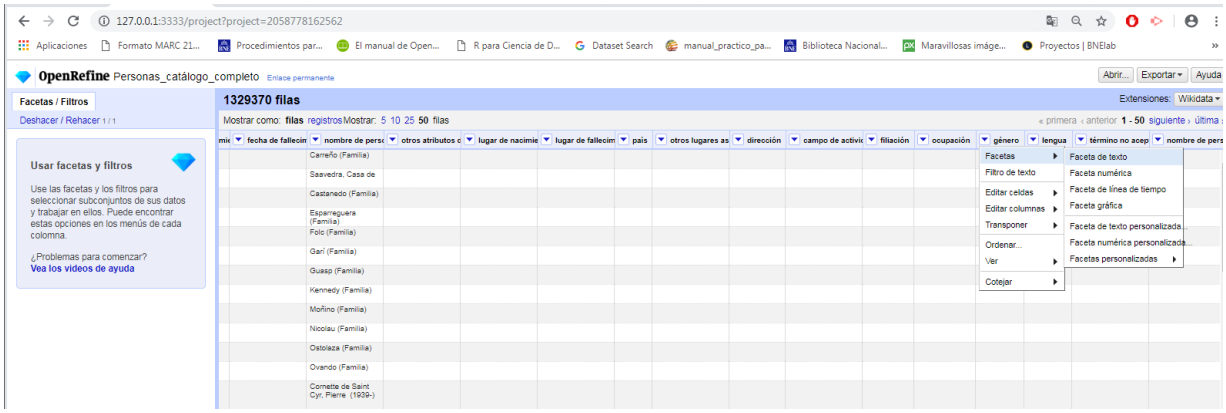


6. Por defecto, nos muestra los resultados de las primeras diez, pero esto se puede configurar fácilmente para adaptar la vista.

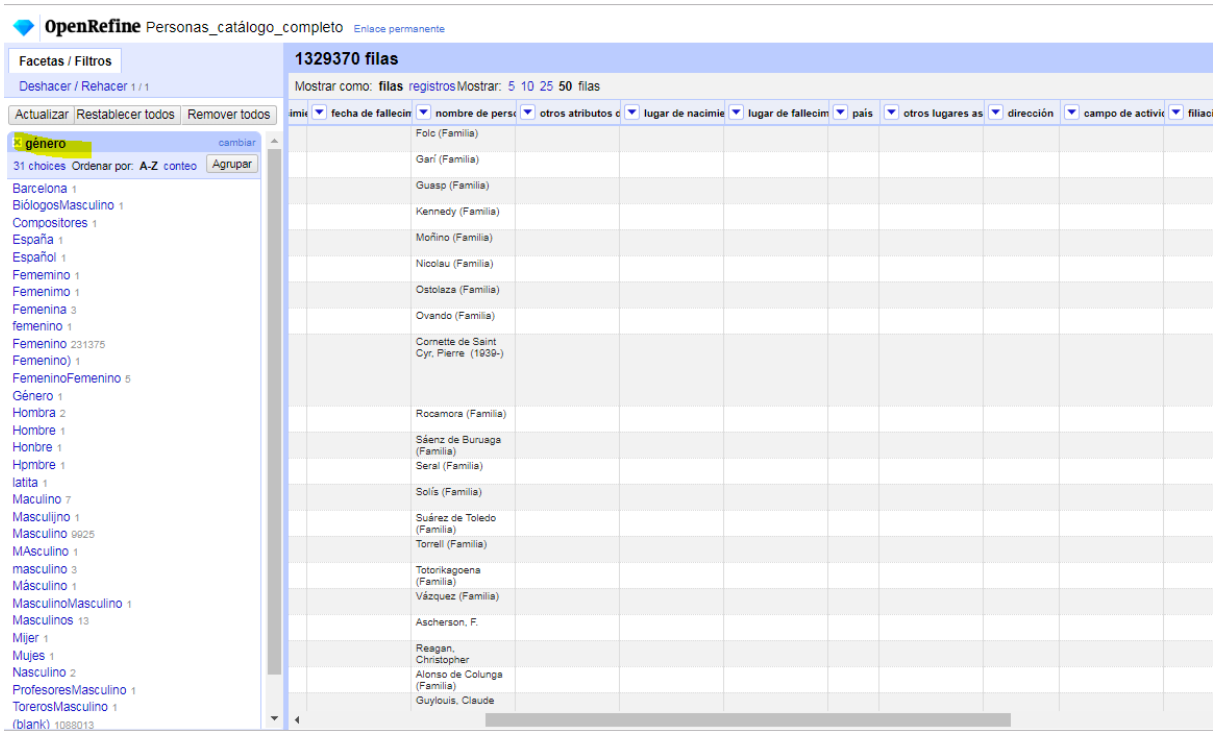
3. Uso de las facetas o filtros

Las facetas son filtros que se aplican al campo que seleccionemos. Funcionan de forma similar a un simple filtro en una hoja de cálculo, con la ventaja de que en OpenRefine podemos trabajar con los 27 ficheros unificados. Es lo que recomendamos utilizar para ver de una forma sencilla la diversidad del catálogo, especialmente en campos normalizados.

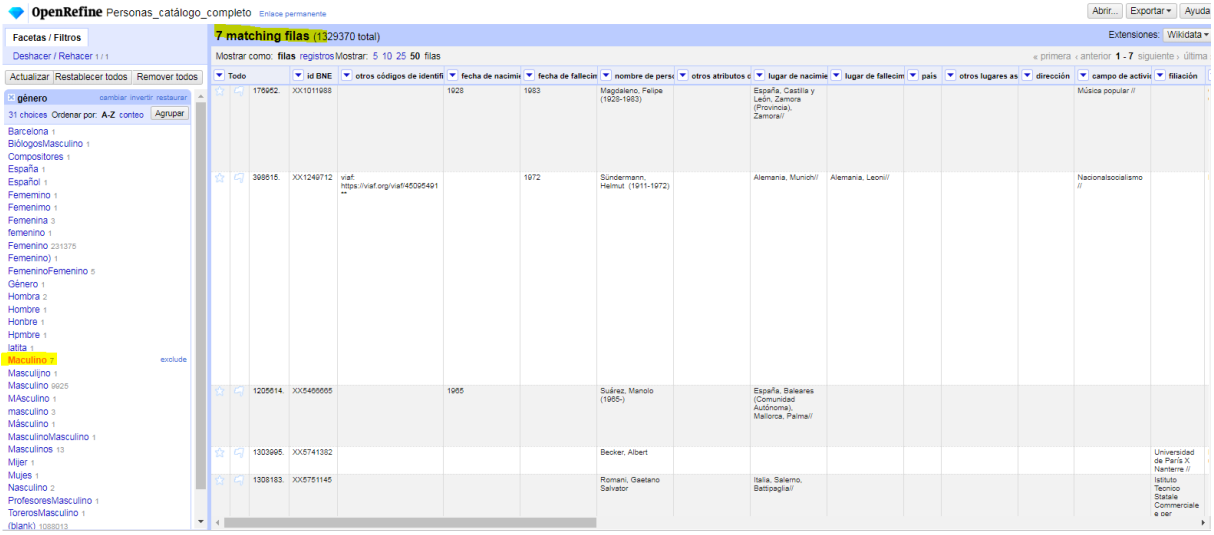
Por ejemplo, aplicando la faceta de texto al campo Género (375 $a), veremos que nuestro catálogo ofrece 31 variantes de posibles respuestas, cuando lo lógico sería que hubiera tres, a saber, Masculino/Feminino/Vacío (campo no rellenado).



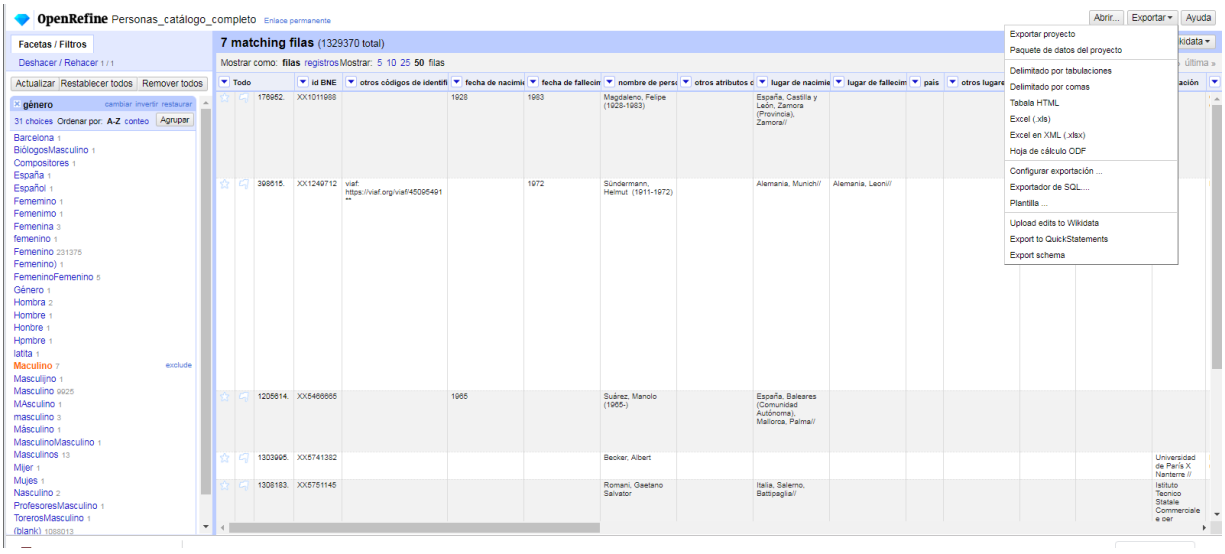
En la columna de la izquierda, se nos mostrará un cuadro con el listado de todas las opciones que el sistema encuentra para ese campo en concreto.



Haciendo clic sobre cada una de las opciones, la vista nos mostrará los registros que se corresponden con ese valor. Por ejemplo, “Maculino” se repite en 7 registros.



La vista seleccionada según las facetas establecidas es descargable en cualquier momento, en diferentes formatos desde el botón exportar.



El uso de las facetas es recomendable para detectar y subsanar errores de forma manual, esto es, registro a registro en el SIGB. No obstante, para realizar cambios masivos (desde Automatización), lo recomendable es seguir limpiando los datos como veremos a continuación, y posteriormente exportarlos.

4. Agrupar valores

Esta opción sirve para normalizar opciones que hayan sido registradas erróneamente, por fallos ortográficos o por simples errores humanos. OpenRefine se vale de una serie de algoritmos y técnicas de agrupación para encontrar coincidencias entre palabras y frases que tengan cierto nivel de similitud.

4.1. Métodos de agrupamiento

Existen dos grandes métodos de agrupamiento, a saber: métodos de colisión clave y métodos del vecino más cercano.

4.1.1. Key Collision Methods o Métodos de colisión clave

Este método se basa en la idea de crear una representación (una 'llave') que contiene solo la parte más valiosa o significativa de la cadena de texto para posteriormente asociar diferentes cadenas, basado en el hecho de que su 'llave' es la misma (de ahí el nombre de 'colisión clave'). Es uno de los métodos más rápidos, su análisis es lineal y permite agrupar, en segundos, millones de registros.

Los métodos de colisión clave disponibles en Open Refine se encuentran repartidos de la siguiente forma:

4.1.1.1. Fingerprint o Huella Digital:

Este método se caracteriza por ser rápido y simple, es el que tiene menos probabilidades de producir falsos positivos y funciona relativamente bien en una variedad de contextos. A continuación se lista el proceso que realiza el algoritmo:

- Quita espacio inicial y final de la cadena.
- Cambia todos los caracteres por su equivalente en minúsculas.
- Elimina todos los signos de puntuación y de control.
- Divide la cadena en fichas separadas por espacios en blanco.
- Ordena las fichas y elimina duplicados.
- Une las letras de nuevo.
- Normaliza caracteres occidentales extendidos a su representación ASCII; por ejemplo: ‘averigüé’ → ‘averigue’.

Este algoritmo funciona muy bien a la hora de identificar casos en los que hay comas, puntos, tildes o mayúsculas que diferencien, en teoría, los nombres, pero que realmente representan lo mismo. Ejemplo de agrución según este método:

“ Biblioteca Pedro-Martínez ” → “Biblioteca Pedro-Martínez”
“Biblioteca Pedro-Martínez” → “biblioteca pedro-martínez”
“biblioteca pedro-martínez” → “biblioteca pedro martinez”
“biblioteca pedro martinez” → “biblioteca Martinez pedro”

4.1.1.2. N-Gram Fingerprint o Huella Digital N-Gram:

Este método es similar al de la huella digital anteriormente descrito, con la diferencia de que en lugar de utilizar espacios en blanco separados por fichas, usa n-gramas, donde n (o el tamaño en caracteres de la ficha) puede ser especificado por el usuario. Este algoritmo sigue el siguiente proceso:

- Cambia todos los caracteres a su equivalente e n minúsculas
- Elimina todos los caracteres de puntuación, espacios en blanco y de control. Por ejemplo:

  “León” → “león”
  “león” → “leon”

- Obtiene todos los n-gramas
- Ordena los n-gramas y elimina duplicados

  (2-grama) “leon” > “le” “eo” “on”
  (1-grama) “leon” > “l””e””o””n”
  (2-grama) “le” “eo” “on” > “eoleon”
  (1-grama) “l””e””o””n” > “elno”

- Une los n-gramas ordenados de nuevo
- Normaliza caracteres occidentales extendidos a su representación ASCII

Ejemplo: la huella digital de 2-gramas de ‘Paris’ es “arispari” y la huella digital de 1-grama es “aiprs”.

En la práctica, según el software, el uso de valores grandes de n en los n-gramas no produce ninguna ventaja sobre el método de la huella digital anterior, pero utilizando 2-gramas y 1-grama se pueden encontrar grupos que el método anterior no es capaz de encontrar incluso con cadenas que tienen pequeñas diferencias.

4.1.2. Métodos vecino más cercano

Los métodos de colisión clave, descritos anteriormente, son muy rápidos pero tienden a ser muy estrictos o muy laxos, “sin poder afinar qué tanta diferencia entre las cadenas estamos dispuestos a tolerar”. Los métodos de vecino más cercano (también conocidos como kNN), proporcionan un parámetro (el radio o k) que representa un umbral de distancia, el cual sirve de referencia para agrupar un par de cadenas de texto si su distancia es cercana.

Los métodos de vecinos más cercanos disponibles en Open Refine se encuentran repartidos de la siguiente forma:

4.1.2.1. Distancia Levenshtein

En términos generales, la Distancia Levenshtein mide el número mínimo de operaciones de edición que se requieren para cambiar una cadena en otra. En la siguiente tabla se explica brevemente en qué consiste la Distancia Levenshtein:

Palabras	Explicación
‘París’ → ‘parís’	Tienen una distancia de edición de 1 debido a que el cambio de P a p es la única operación requerida.
‘Nueva York’ → ‘newyork’	Tiene una distancia de edición de 3: 2 sustituciones y 1 eliminación.

De acuerdo con OpenRefine, esta distancia “es útil para identificar errores tipográficos, errores de ortografía o cualquier cosa que los métodos anteriores no capturan, aunque las grandes distancias dan muchos falsos positivos (especialmente para las cadenas cortas) y no son tan útiles”.

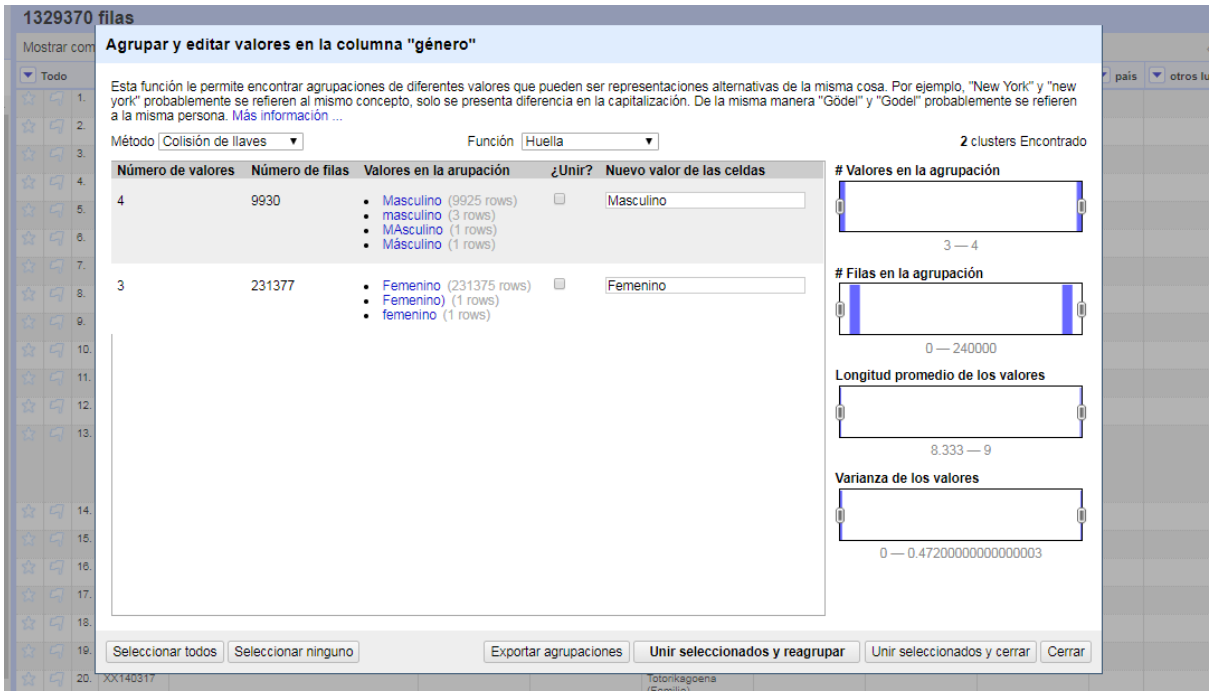
4.1.2.2. PPM o Predicción por Coincidencia Parcial

El algoritmo de Predicción por Coincidencia Parcial estima similitudes entre cadenas de texto empleando una operación que mide los recursos empleados por un computador para definir una cadena de texto. La operatividad del algoritmo surge a partir de la forma en que funcionan los compresores de texto; por ejemplo, si dos cadenas de texto A y B son idénticas, al momento de comprimirlas, es decir, al realizar la operación (A+B), se genera muy poca diferencia. Por otro lado, si A y B son muy diferentes, al momento de comprimirlas se deben producir diferencias dramáticas en longitud. La documentación del software recomienda emplear esta técnica como último recurso.

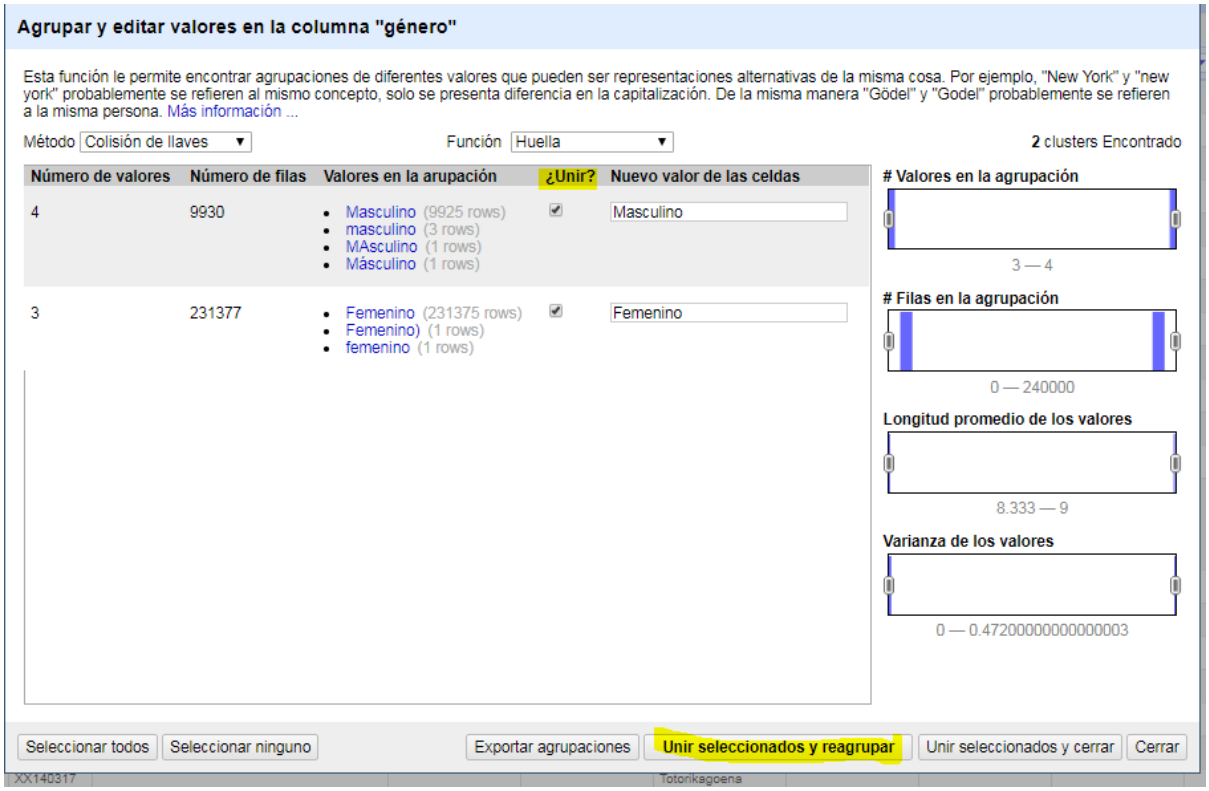
4.2. Ejemplos de agrupamiento

Una vez vistos los métodos, veremos cómo los hemos aplicado a nuestro ejemplo.

4.2.1. Usando colisión de llaves



En este caso todas las opciones propuestas se pueden agrupar, por tanto las seleccionaríamos y "Unir seleccionados y reagrupar"

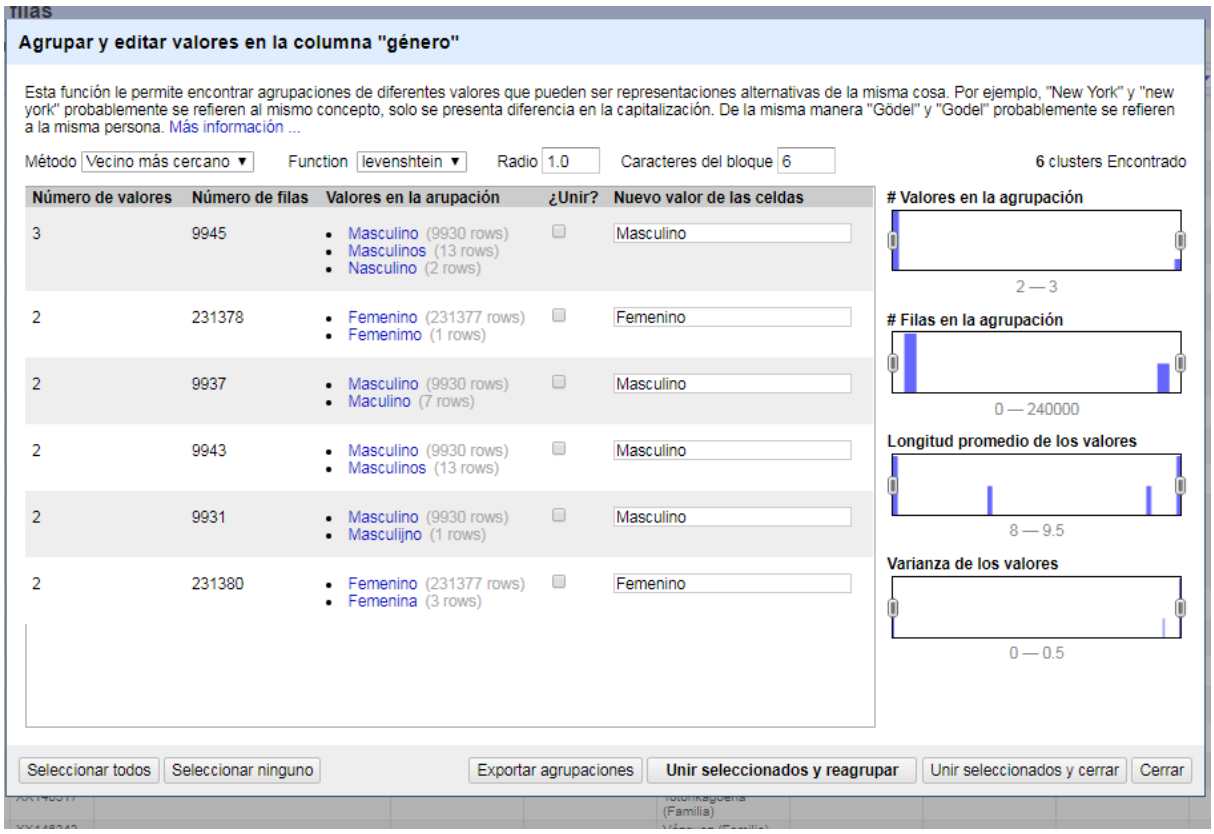


4.2.2. Usando el método del vecino más cercano / Levenshtein:

En este caso se utilizan 2 variables:

- Radio: Número de movimientos o “cambios” necesarios para que las cadenas de texto sean iguales.
- Caracteres del bloque: Número de caracteres consecutivos que compara.

En nuestro ejemplo, con radio 1 (1 cambio) y 6 caracteres consecutivos, obtenemos las siguientes agrupaciones.

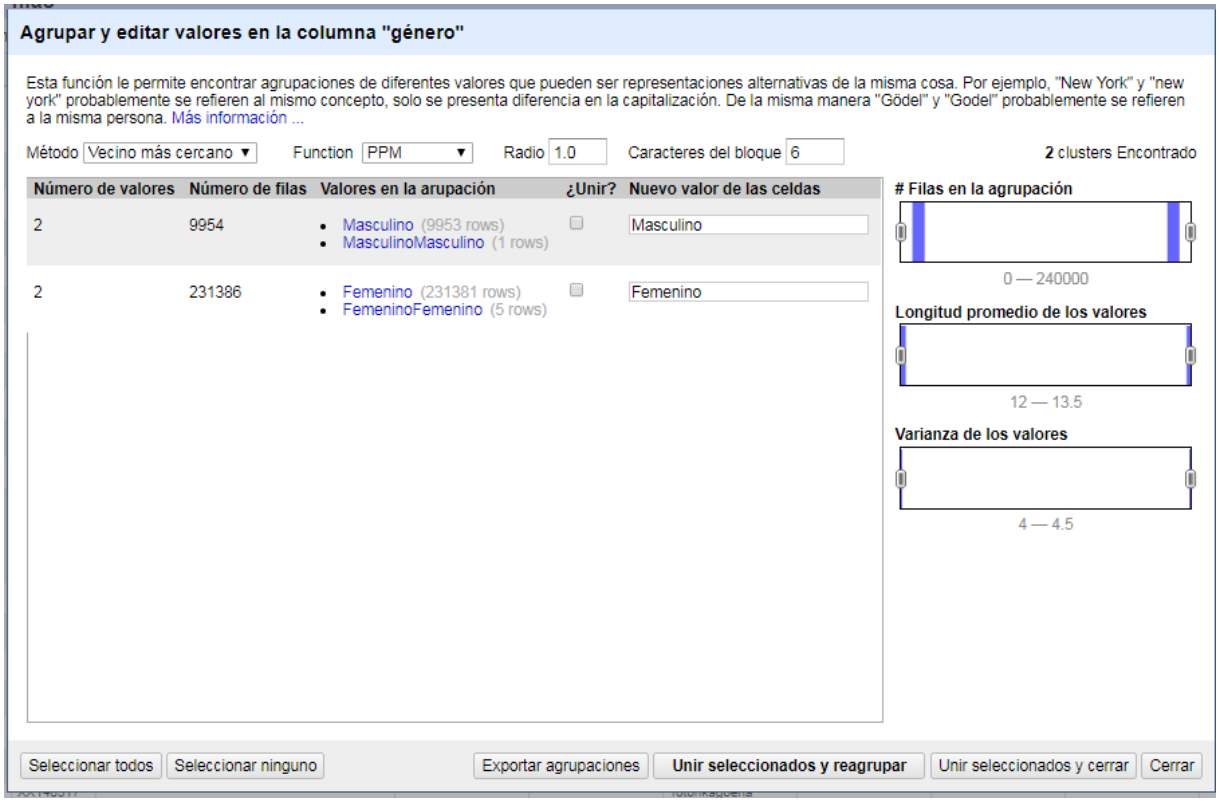


Como todas las propuestas de agrupación son válidas, las aceptamos.

4.2.3. Utilizando el método del vecino más cercano / PPM:

En este caso las variables significan:

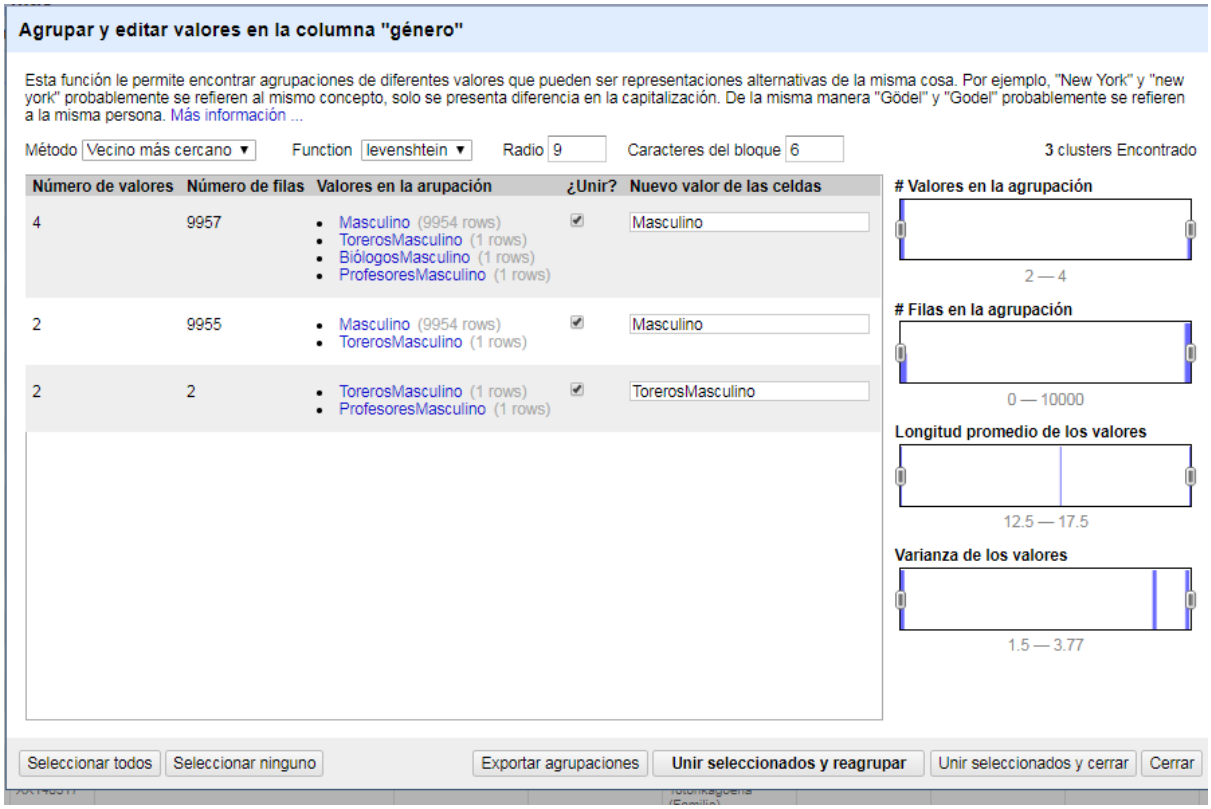
- Radio: Número de cambios aceptados para considerar 2 cadenas de caracteres iguales.
- Caracteres de bloque: Número de caracteres que utiliza para partir la cadena original.



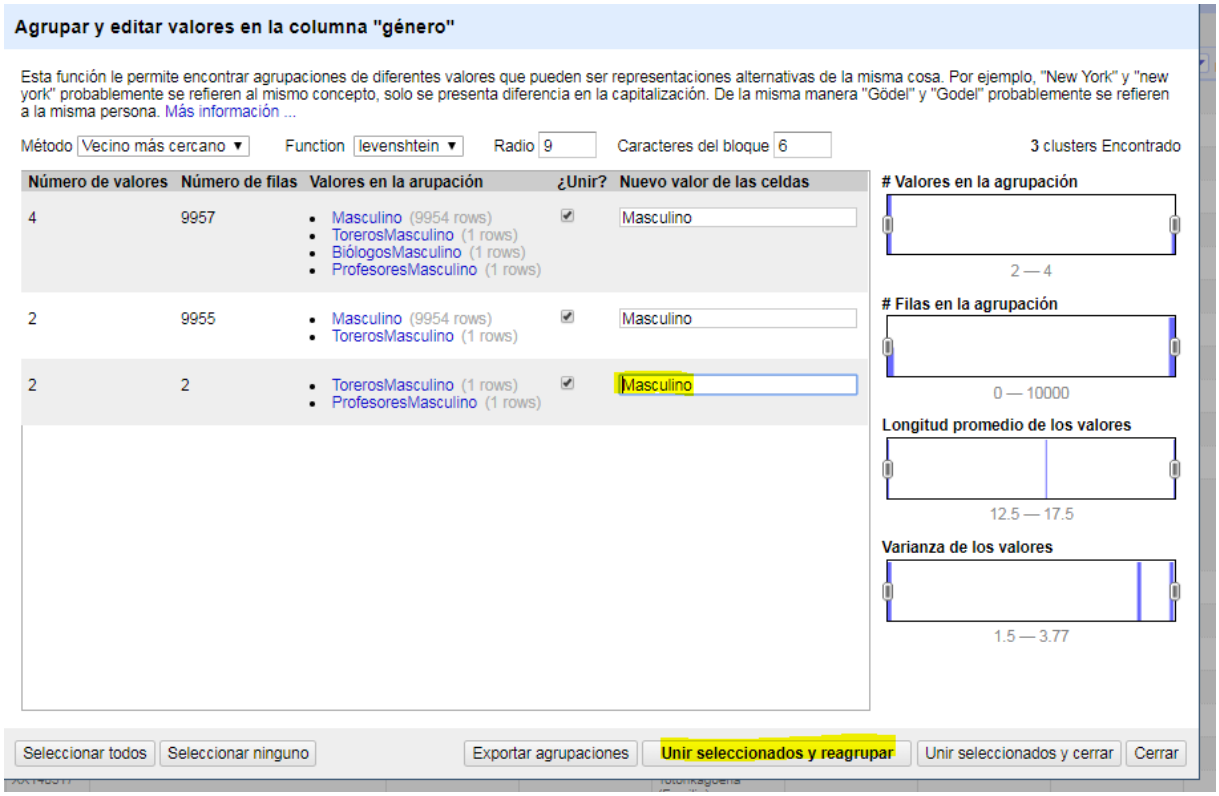
En este caso, vuelven a ser válidas las opciones propuestas.

Dada la complejidad de estos algoritmos, conviene realizar diferentes combinaciones de valores de los parámetros para intentar “cazar” el mayor número de opciones erróneas.

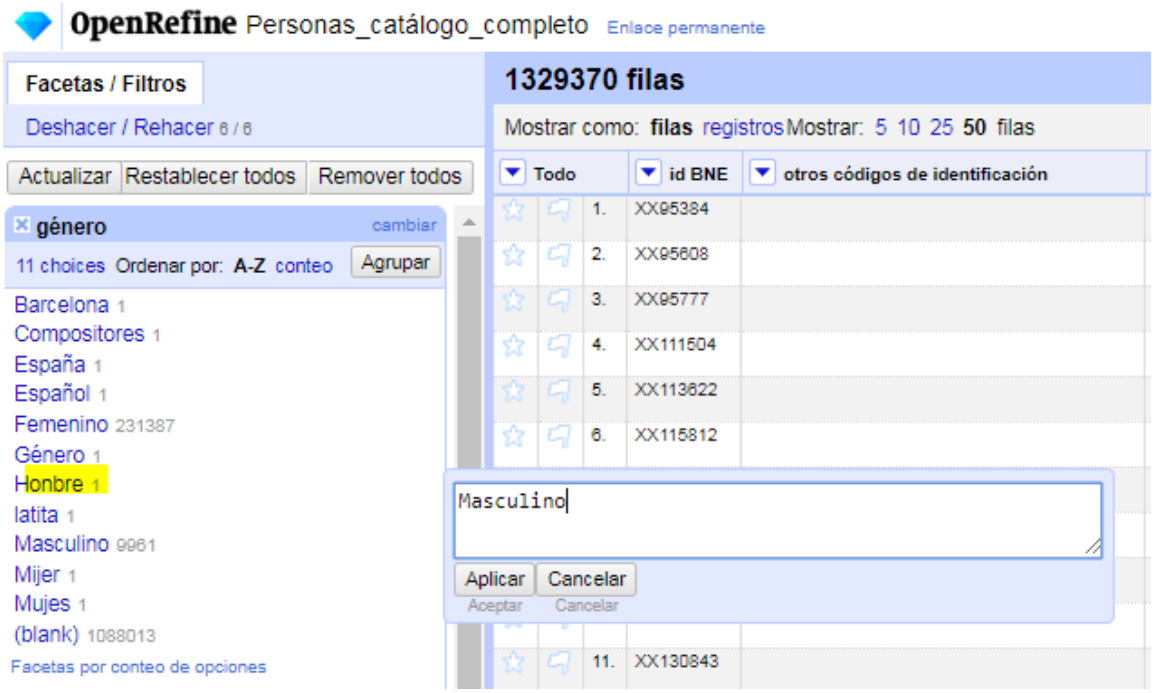
Por ejemplo, al ampliar el radio a 9, obtenemos:



En este caso, para tercera opción nos propone algo que no es lo ideal, por tanto podemos corregirlo en el momento.

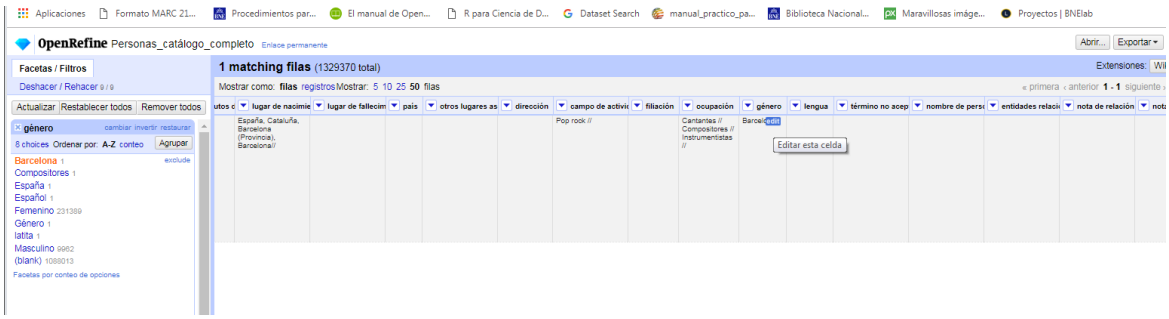


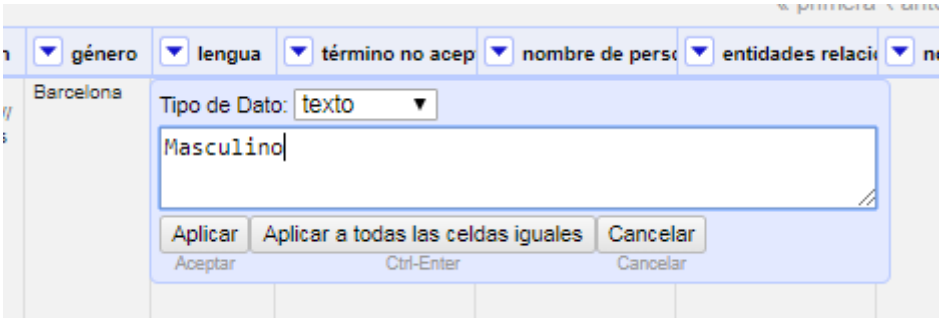
Una vez pasados los algoritmos, aún quedarán errores que se podrán corregir manualmente.



Después de haber limpiado los casos obvios, tendríamos que detenernos en mirar algunos casos particulares. Nos queda la lista de opciones reducida a 8.

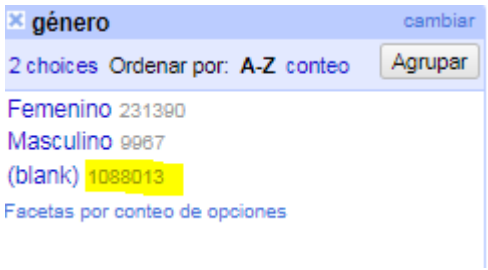
Seleccionando cada opción, se nos mostrará el registro asociado en el panel de la derecha y podremos editarlo.





Y así lo haríamos con todos los casos en duda.

Al final, conseguimos limpiar de una forma sencilla todas las opciones hasta reducirlo a lo correcto.

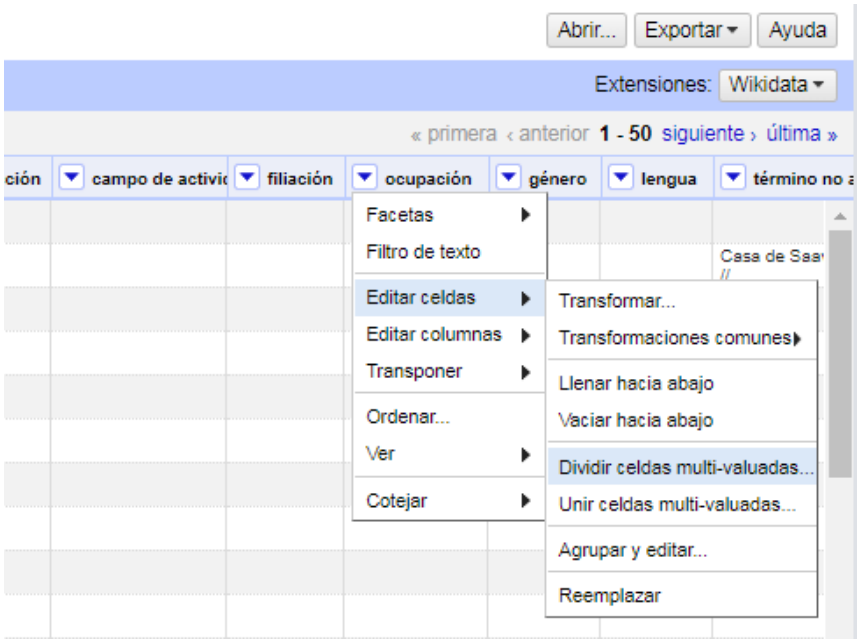


Como hecho remarcable, hay 1.088.013 registros que no tienen el campo “Género” completado. Ahora podríamos exportar el conjunto total, o en dos tandas (masculino/femenino) para poder modificar los valores en nuestro SIGB, Symphony.

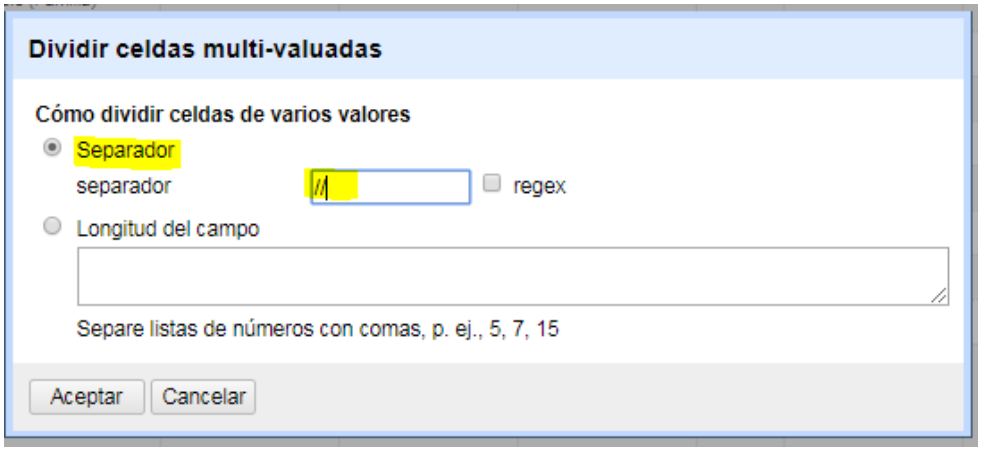
5. Las celdas multivaluadas

En muchos casos, el resultado de nuestro mapeo nos presenta en una misma celda valores agrupados. Esto ocurre cuando el campo MARC es repetible. Tomemos por ejemplo el campo Ocupación (374 $a). Veremos que cuando los valores se repiten para una misma autoridad, aparecen separados por “//”.

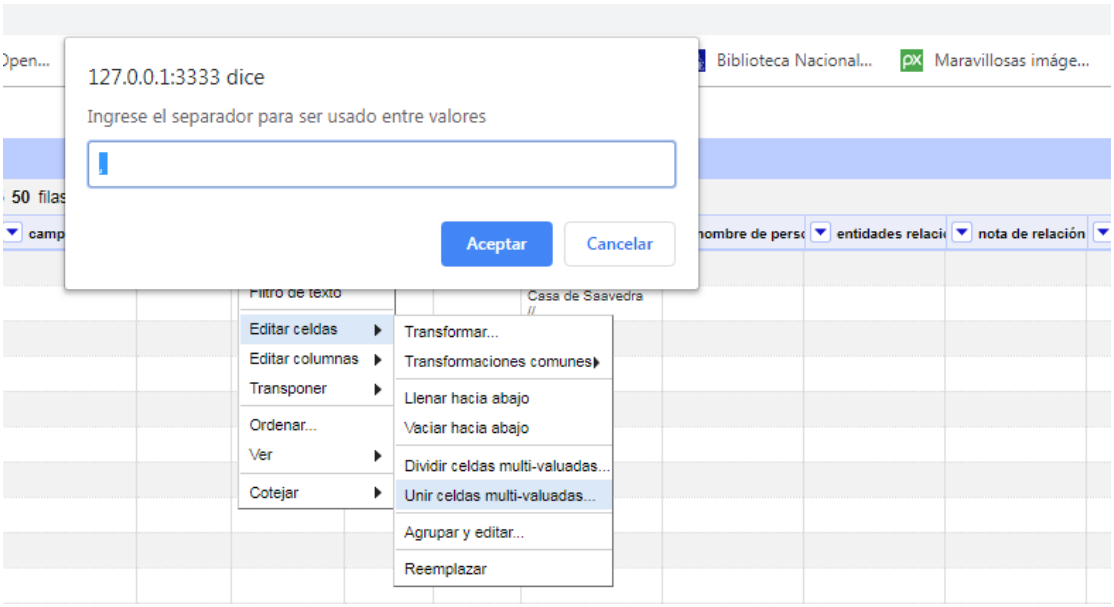
En este caso, lo primero que hay que hacer es separar las celdas multivalor en valores individuales. Para ello, desde la columna que queremos dividir: Editar celdas/Dividir celdas multi-valudadas



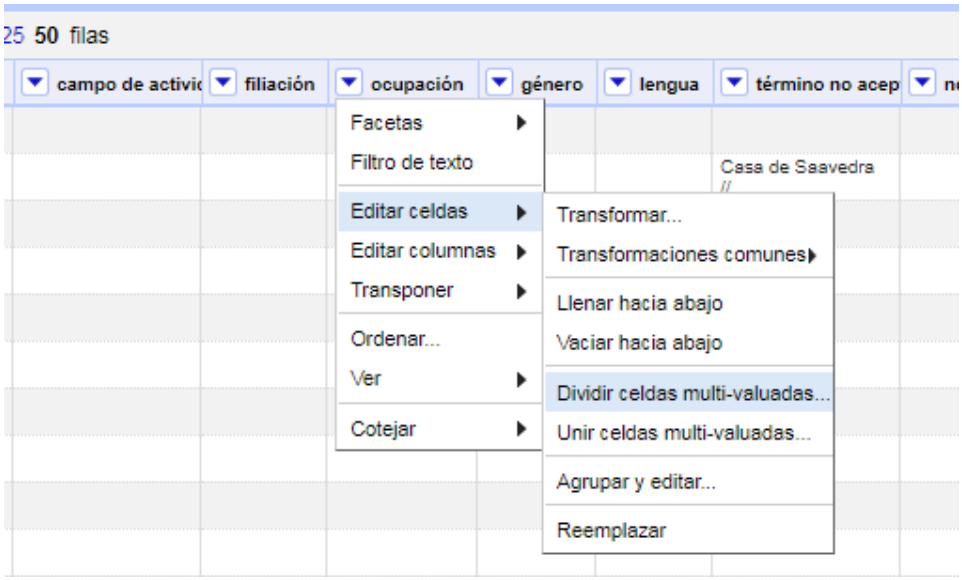
En el cuadro de diálogo debemos establecer qué caracteres queremos que use el sistema para "partir" las celdas. En nuestro caso, será "//".



Por la estructura de nuestro mapeo, es necesario realizar esta acción 2 veces, por tanto los tenemos que unir y volver a separar. Unimos con "//"

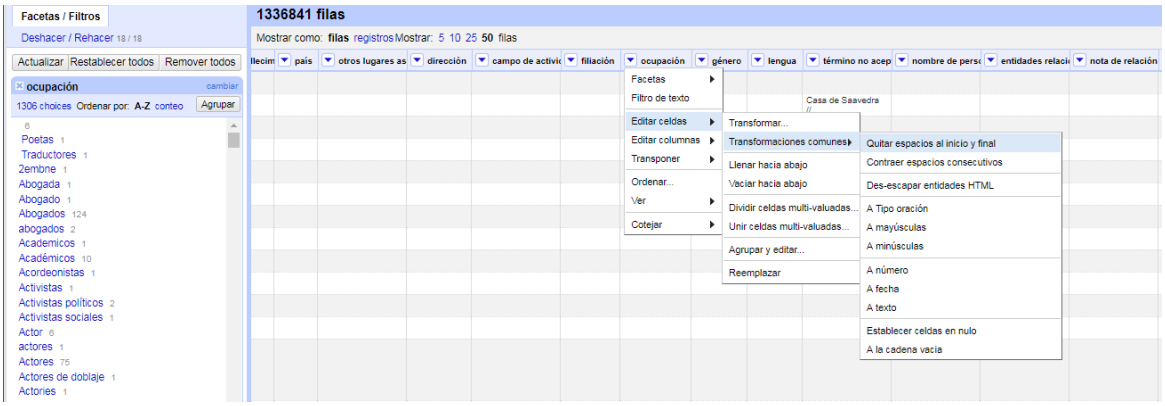


Separamos por "//"



El resultado serían 1.306 opciones:

En este caso, al haber muchas opciones, recomendamos empezar limpiando los espacios en blanco al principio y al final: Editar celdas/Transformaciones comunes/Quitar espacios al inicio y final.



Solo con esta operación, hemos reducido el número de opciones a 987.

Ahora sería el momento de repetir los diferentes métodos de agrupamiento que hemos visto anteriormente.

Al terminar de limpiar, como en este caso había celdas multivalor, debemos recordar que estamos trabajando con filas y hay que reagruparlo a nivel de registro. Eso lo haríamos utilizando de nuevo la opción unir celdas multivalor.

6. Enriquecimiento con fuentes externas

Imaginemos que, una vez detectados y corregidos los errores del catálogo, nos interesa enriquecerlo, al introducir datos que ya existen en otros catálogos y bases de datos de prestigio. OpenRefine nos ayuda también con esta tarea, ya que ofrece un servicio de “reconciliación”, que nos permite cotejar nuestros datos contra otras fuentes.

OpenRefine, por defecto, ofrece esta posibilidad contra una serie de servicios de reconciliación preestablecidos, disponibles en esta url: http://refine.codefork.com/ . Los más importantes serían Wikidata, LC y VIAF (por países).

Los pasos a seguir serían:

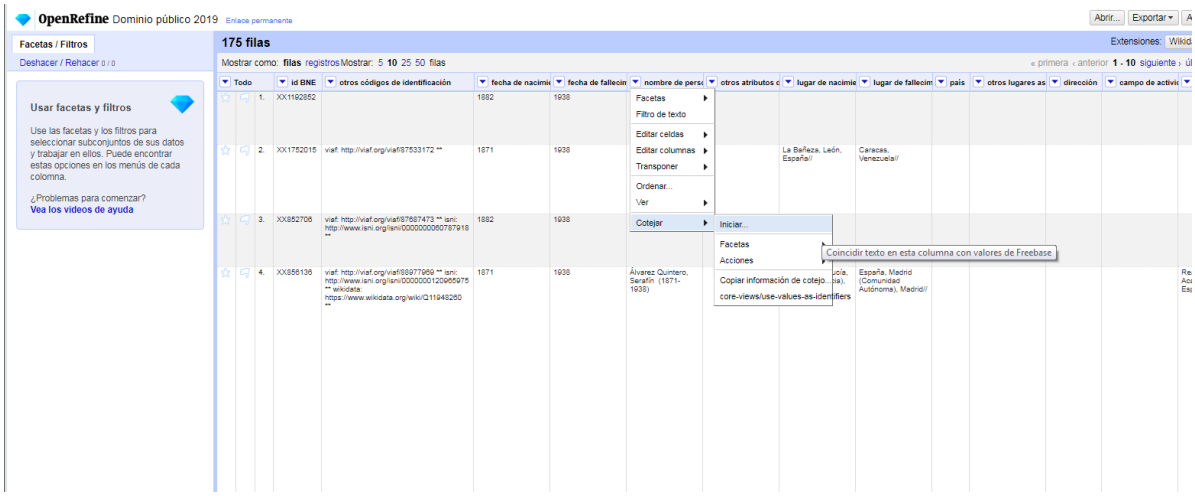
- Reconciliar los datos con un servicio preestablecido, o
- Reconciliar los datos con una fuente de datos propia
- Crear columnas basadas en datos vinculados

6.1. Ejemplo 1: Autores en dominio público

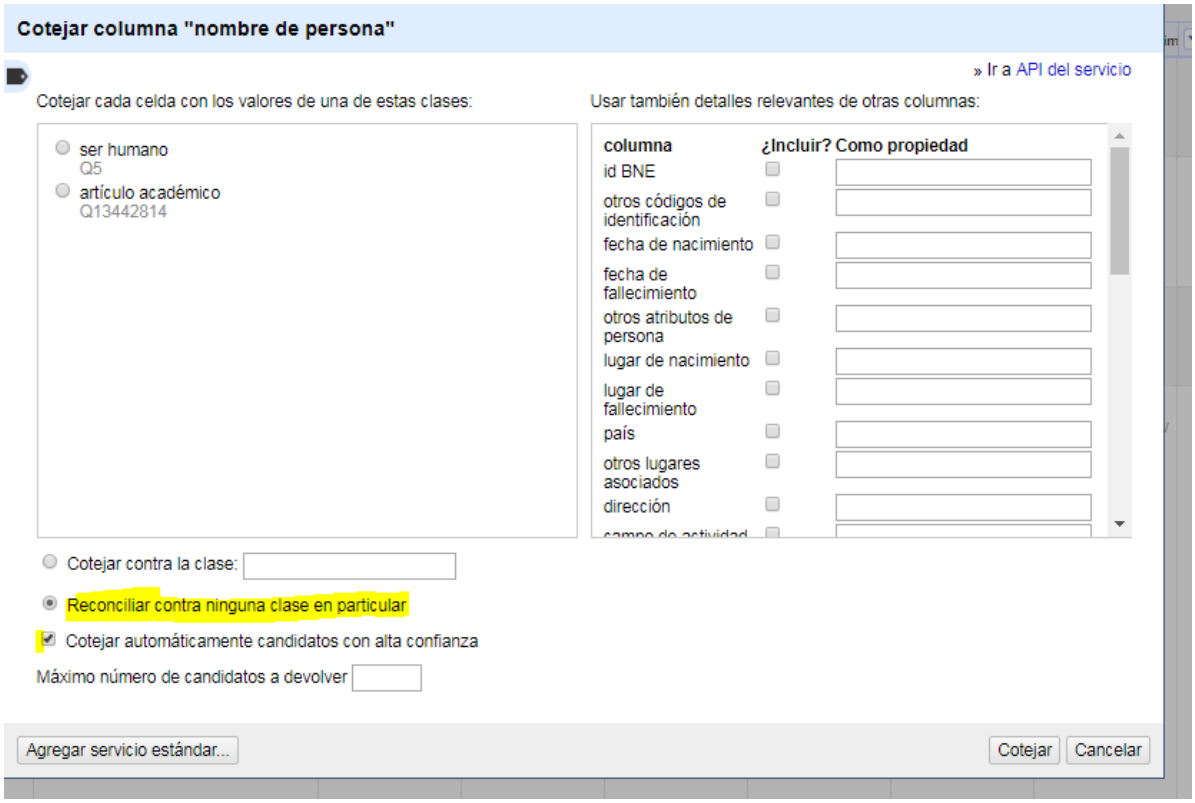
6.1.1. Reconciliar datos con Wikidata

Utilizamos para el ejemplo el conjunto de autores que entraron en dominio público en 2019, fallecidos en 1938 (https://bnelab.bne.es/dato/autores-espanoles-en-dominio-publico/).

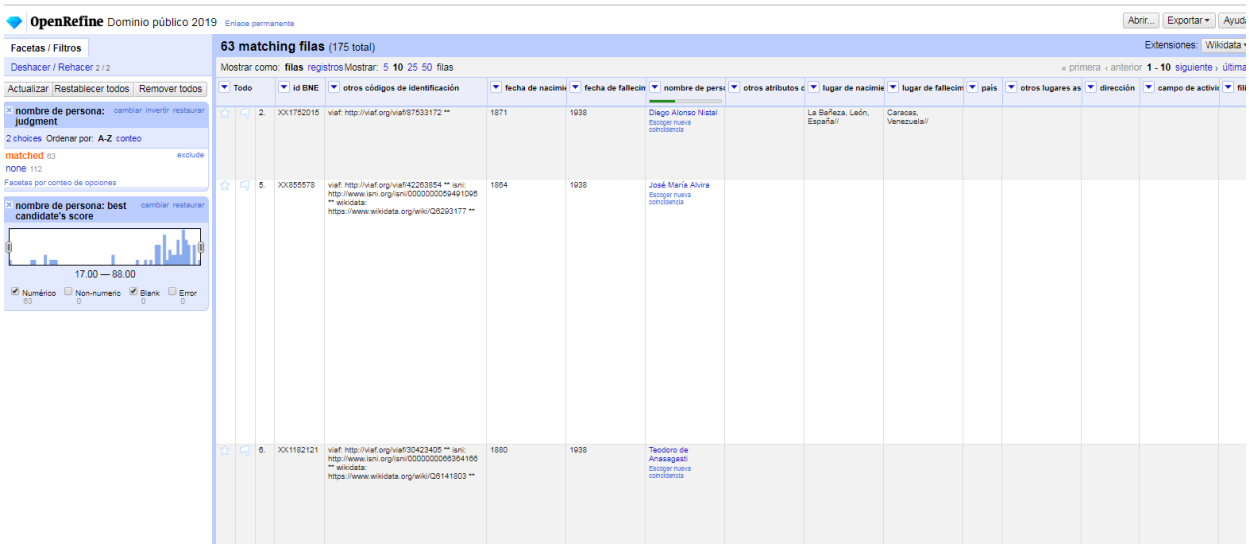
En OpenRefine, llamamos “reconciliar” o “cotejar” a la tarea de buscar nuestros datos en una fuente externa. Debemos elegir qué campo queremos utilizar para ir contra Wikidata. En este caso, buscaremos por “Nombre de persona” (MARC: 100 $a $b $c ($d)($q)). En la columna correspondiente: Cotejar/Iniciar.



En el cuadro de diálogo, elegimos los parámetros. En este ejemplo vamos a lanzar la búsqueda sin establecer una clase determinada.



Hemos conseguido vincular 63 registros, de los 175 que teníamos en origen. De esos 63 autores podemos traer datos que existen en Wikidata para completar nuestro conjunto, y posteriormente descargarlo para enriquecer nuestro catálogo.



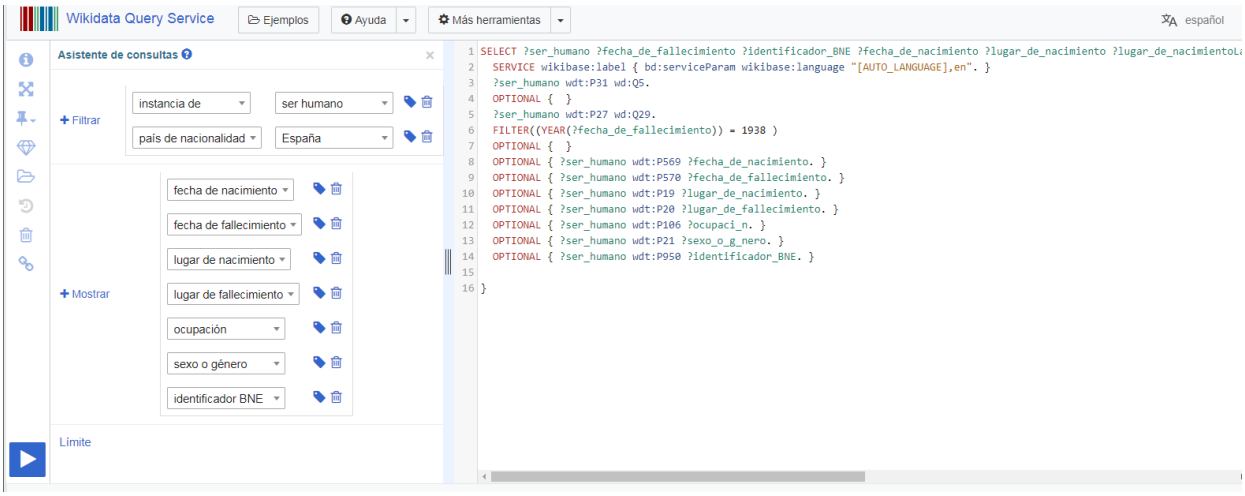
6.1.2. Reconciliar datos con fuentes propias: reconciliar con un subconjunto de Wikidata obtenido mediante consulta SPARQL

Además de reconciliar tus datos con las fuentes explicadas anteriormente, existe la posibilidad de crear tus propias fuentes.

Estas fuentes deben ser ficheros en formato CSV. Vamos a utilizar una herramienta libre que se llama Reconcile-CSV, que nos permite convertir nuestro CSV en una fuente de reconciliación:

Por ejemplo, para el conjunto de autores españoles fallecidos en el 1938, en lugar de buscar en todo wikidata, podríamos buscar únicamente en el subconjunto de personas españolas fallecidas en el año 1938.

Hemos realizado esta consulta con SPARQL

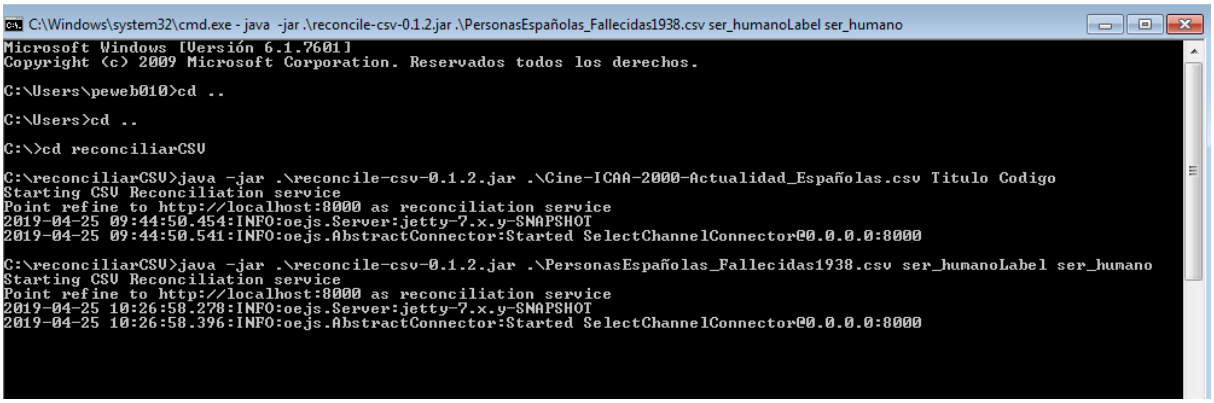


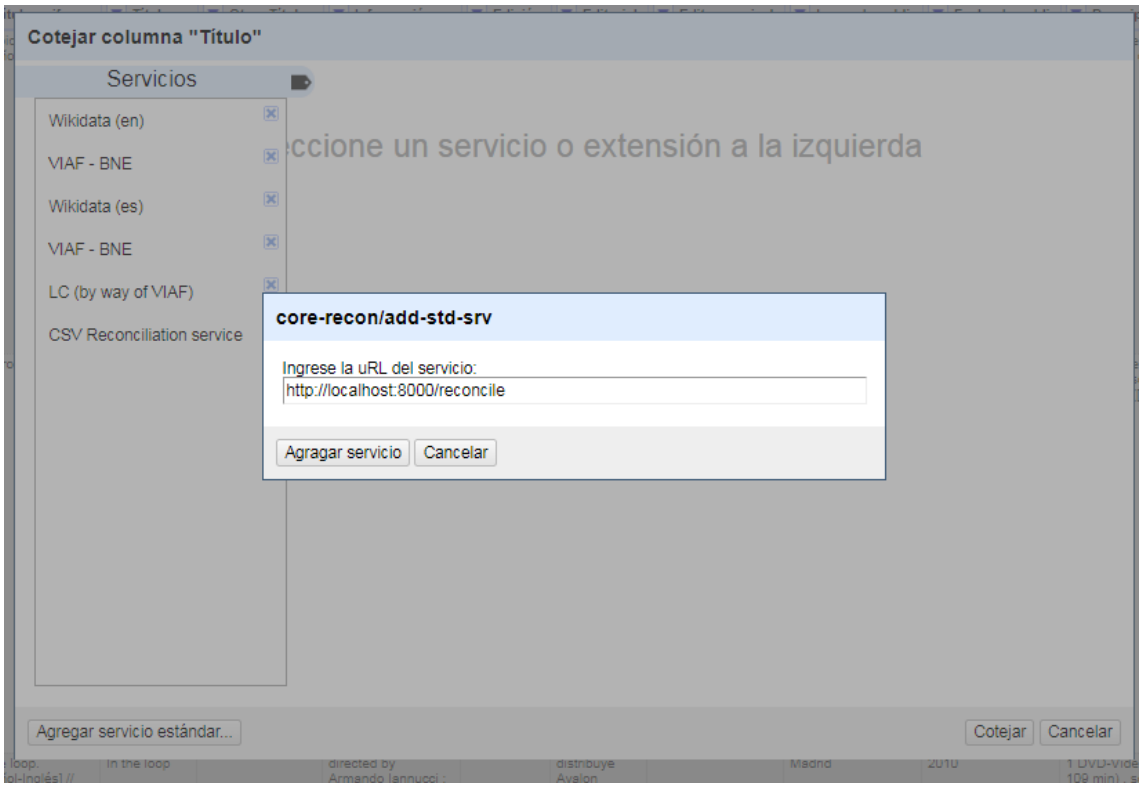

```
SELECT ?ser_humano ?fecha_de_fallecimiento ?identificador_BNE ?fecha_de_nacimiento ?lugar_de_nacimiento
?lugar_de_nacimientoLabel ?ocupaci_n ?ocupaci_nLabel ?sexo_o_g_nero ?sexo_o_g_neroLabel
?lugar_de_fallecimiento ?lugar_de_fallecimientoLabel
WHERE {
  SERVICE wikibase:label { bd:serviceParam wikibase:language "[AUTO_LANGUAGE],en". }
  ?ser_humano wdt:P31 wd:Q5.
  OPTIONAL {  }
  ?ser_humano wdt:P27 wd:Q29.
  FILTER((YEAR(?fecha_de_fallecimiento)) = 1938 )
  OPTIONAL {  }
  OPTIONAL { ?ser_humano wdt:P569 ?fecha_de_nacimiento. }
  OPTIONAL { ?ser_humano wdt:P570 ?fecha_de_fallecimiento. }
  OPTIONAL { ?ser_humano wdt:P19 ?lugar_de_nacimiento. }
  OPTIONAL { ?ser_humano wdt:P20 ?lugar_de_fallecimiento. }
  OPTIONAL { ?ser_humano wdt:P106 ?ocupaci_n. }
  OPTIONAL { ?ser_humano wdt:P21 ?sexo_o_g_nero. }
  OPTIONAL { ?ser_humano wdt:P950 ?identificador_BNE. }

}
```

Obtenemos un conjunto de 442 registros. Podemos descargarlo en CSV y trabajarlo como fuente externa para reconciliar desde OpenRefine con nuestro conjunto origen.

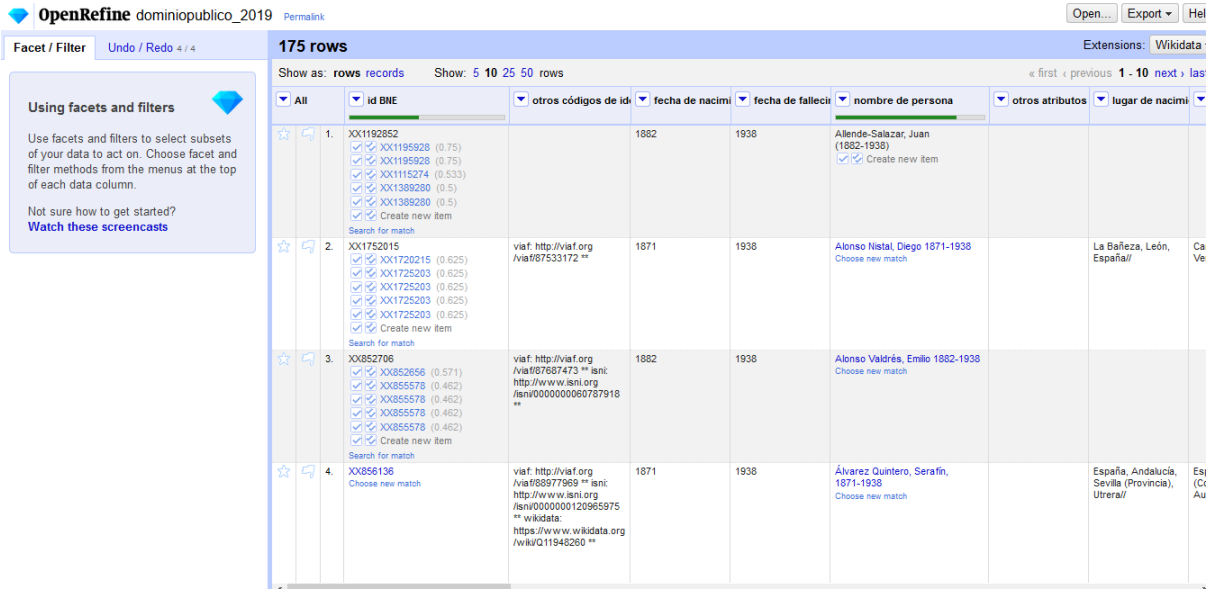
Ahora reconciliamos nuestro conjunto de 175 autores en dominio público con el conjunto de 442 de Wikidata.





Resultado:

- Al intentar vincular por nombre de persona, al 100% no existe ninguna coincidencia, pero por encima del 50% obtenemos 142. Esto se debe a que nuestra forma de escribir el nombre, va en orden inverso y además incluye las fechas de nacimiento y muerte, por lo que es imposible la coincidencia total.
- Al intentar vincular por ID BNE, obtenemos 78 coincidencias al 100%. Esto puede deberse a que muchos autores de Wikidata no tienen el metadato ID BNE completo.



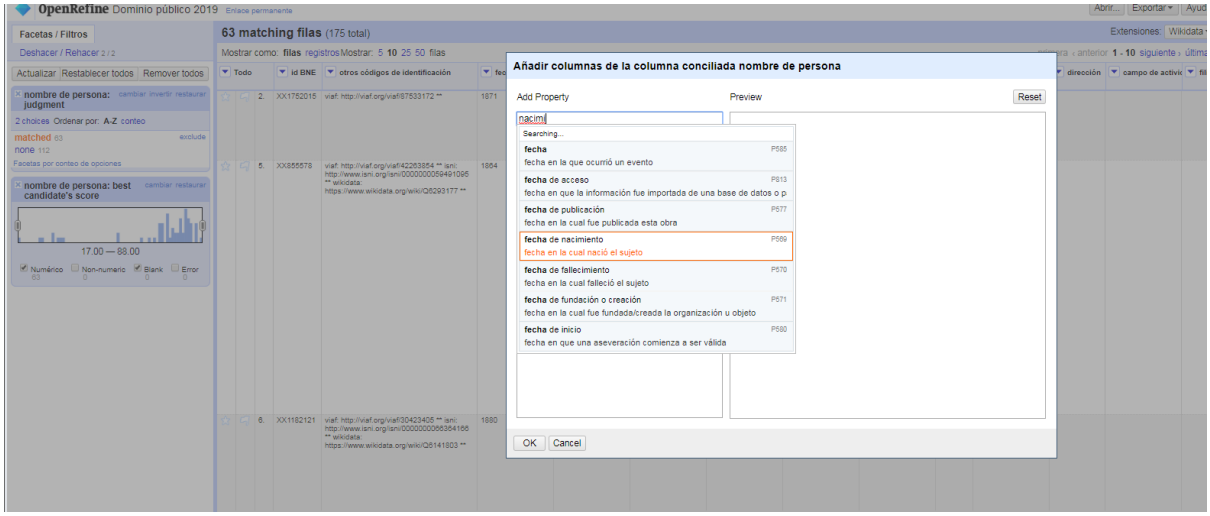
Vistos estos ejemplos, podemos decir, que si se realiza una limpieza previa de la fuente que queremos usar como repositorio de reconciliación, podemos mejorar el porcentaje de coincidencias automáticas.

6.1.3. Crear columnas basadas en datos vinculados

Una vez que hemos logrado “casar” nuestros datos con otras fuentes, podremos ahora importar los campos que nos interese incorporar a nuestro catálogo.

Del conjunto de registros, pensamos que hay una serie de datos de interés para enriquecer las autoridades:

- Fecha de nacimiento
- Fecha de fallecimiento
- Lugar de nacimiento
- Lugar de muerte
- Ocupación
- Género
- ID BNE



Ejemplo de un autor concreto del que hemos recuperado información de Wikidata:

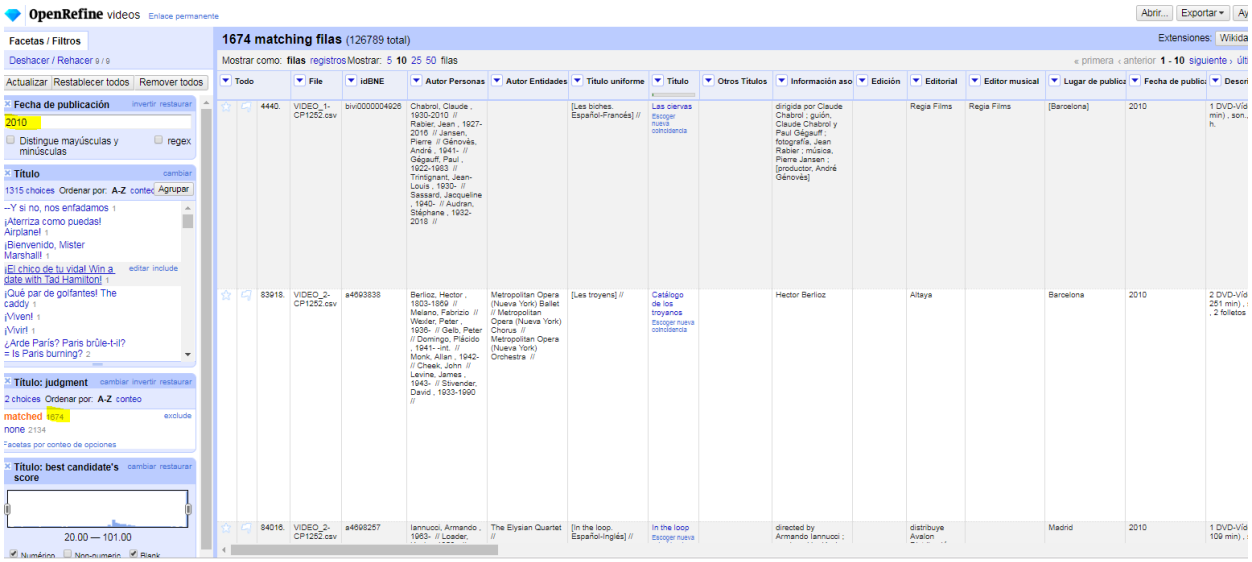
	Fuente: BNE	**Fuente: Wikidata**
Nombre de persona	José María Alvira	
id BNE	XX855578	XX855578
Fecha de nacimiento	1864	1864-01-01T00:00:00Z
Fecha de fallecimiento	1938	1938-01-01T00:00:00Z
Lugar de nacimiento		Zaragoza
Lugar de fallecimiento		Madrid
Ocupación		compositor // pianista // director de orquesta // profesor
Género/sexo		Masculine

6.2. Ejemplo 2: Videograbaciones publicadas en 2010

6.2.1. Reconciliar datos con Wikidata

Otro ejemplo podría ser reconciliar registros bibliográficos de videograbaciones, a través del campo "Título" (MARC: 245 $a, $b, $n, $p). Para tener un conjunto manejable, hemos filtrado videograbaciones con fecha de publicación 2010 y que tengan algún intérprete (MARC: 511 $a). Este conjunto queda reducido a 3.808 registros.

Después hemos llamado a Wikidata. En este caso, en lugar de buscar contra todo Wikidata, hemos preferido elegir directamente la clase Película (Q11424). Tras el proceso, se han vinculado 1.674 películas, de las que podemos traer datos de interés para el registro.



6.2.2. Reconciliar datos con fuentes propias: Reconciliar con catálogo de películas calificadas del ICAA

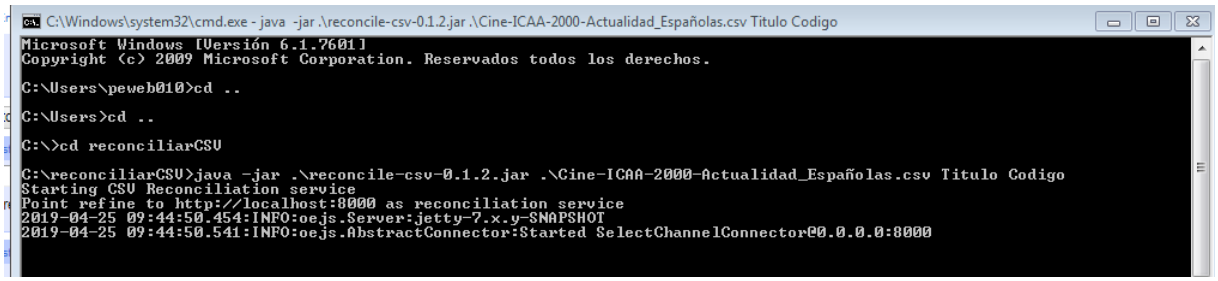
El ICAA dispone de una base de datos de películas calificadas desde los años 40 a la actualidad. Este repositorio está muy completo con información muy detallada en campos que nosotros no tenemos, por ejemplo: nacionalidad o año de producción.

Hemos podido acceder a un servicio web, que nos ofrece la posibilidad de descargar algunos de esos datos de las películas calificadas en un año concreto. Con esto podemos generar un CSV que utilizaremos como repositorio de reconciliación.

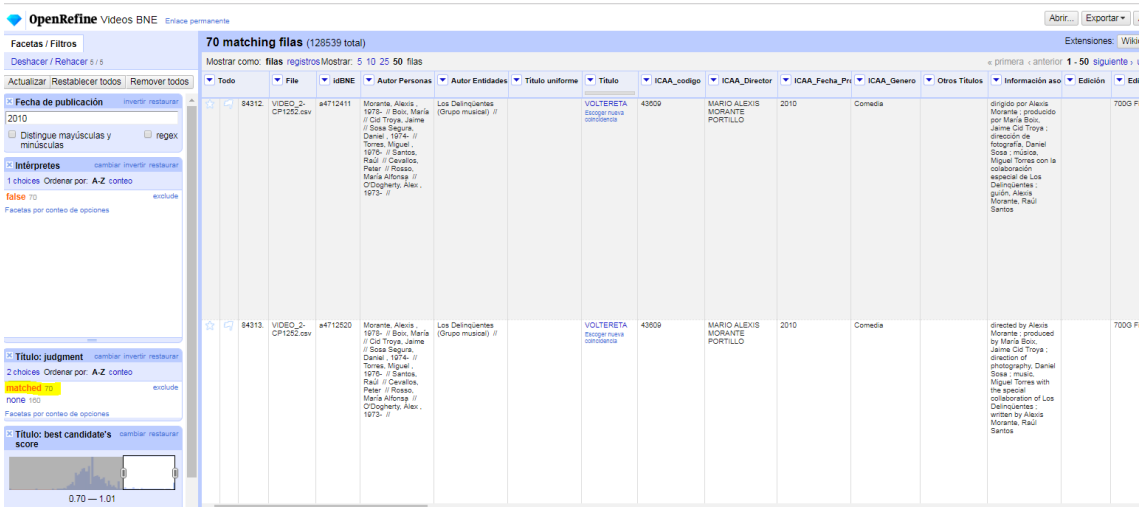
Nuestro conjunto origen son registros bibliográficos de videograbaciones (BNE) con fecha de publicación 2010 e intérpretes no nulos (3.308 registros)

Como repositorio de reconciliación vamos a tomar el fichero csv de películas del ICAA desde el año 2000 a la actualidad (7.790 películas).

Ejecutamos el reconcile-csv para nuestro fichero fuente, con las películas del ICAA.



Intentamos reconciliar desde OpenRefine y como resultado de intentar vincular por título, obtenemos 70 coincidencias al 70%. Esto significa que son 70 ediciones de película que podemos confirmar que son de nacionalidad Española.



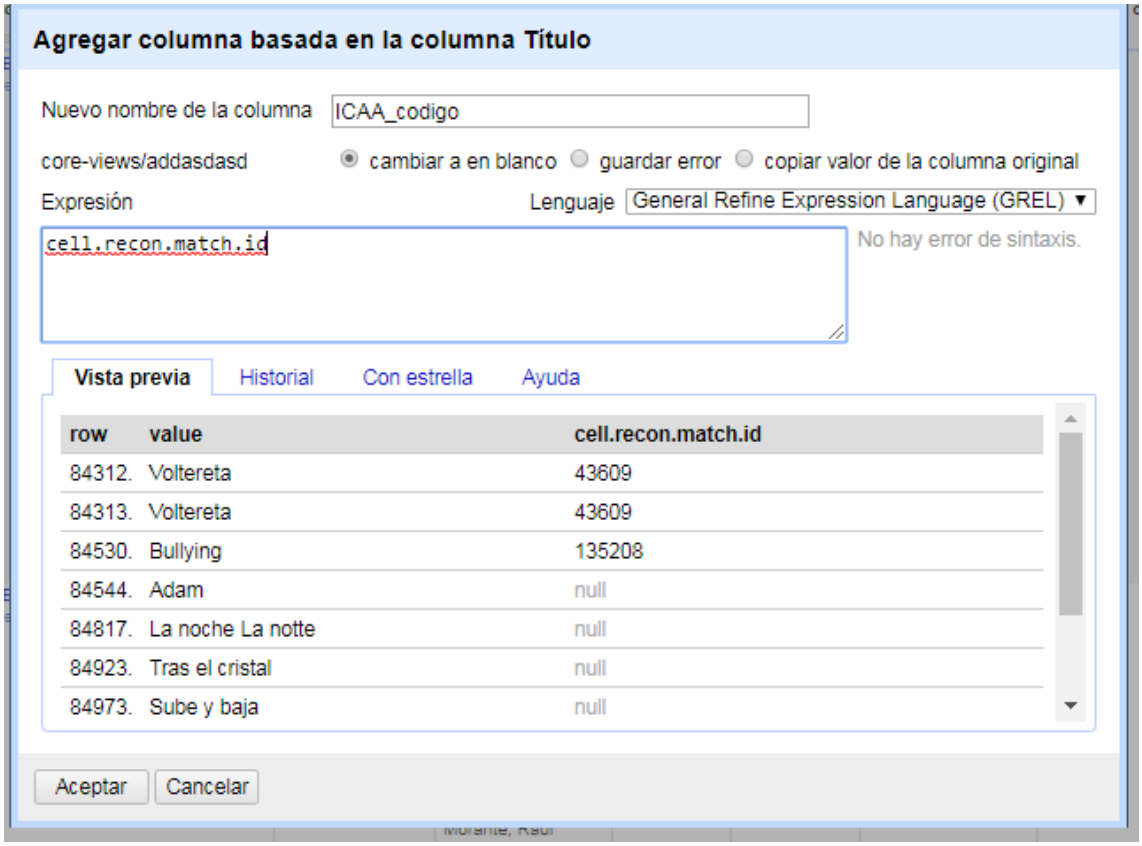
Sobre estas películas coincidentes, podemos traer sus datos asociados para completar nuestro catálogo.

6.2.3. Crear columnas basadas en datos vinculados

Si por ejemplo, quisiéramos enriquecer nuestro catálogo con los campos año de producción, género y director del fichero de datos del ICAA, necesitaríamos tener un campo en común, que debe ser el identificador único del ICAA, para así posteriormente, poder recuperar su información asociada.

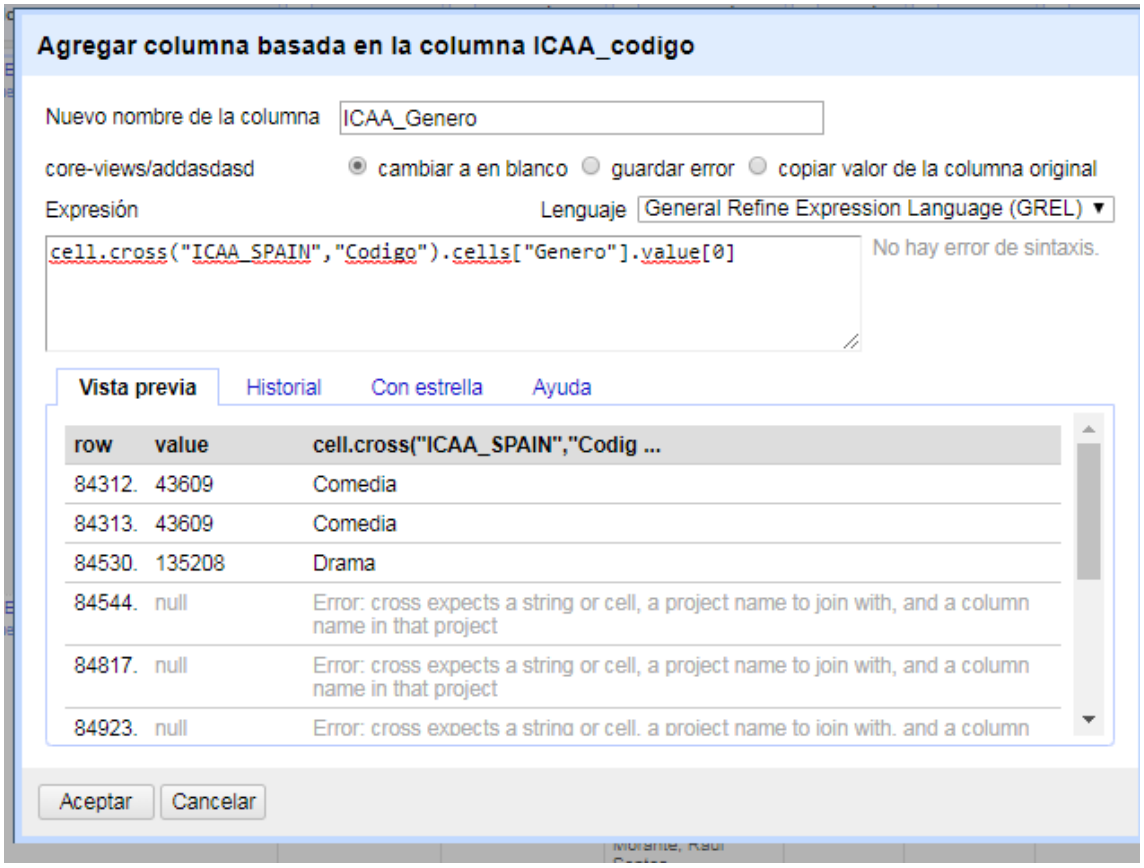
Desde OpenRefine, esto se haría trabajando con 2 proyectos simultáneamente (proyecto A: datos originales y proyecto B: el archivo usado como repositorio de reconciliación)

Para hacer esto recuperamos en el proyecto A el ID único del fichero CSV que utilizamos como fuente de reconciliación (en este caso, el código ICAA).

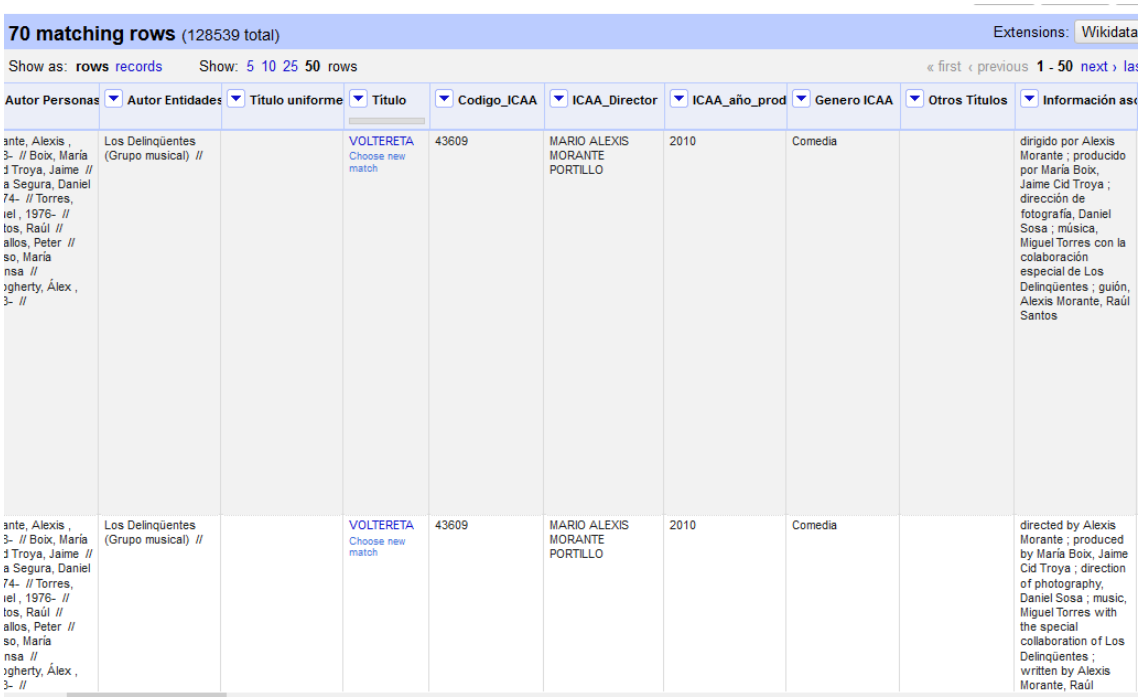


Gracias a este código relacionamos los 2 proyectos y podemos ir trayendo columnas seleccionadas del proyecto B al proyecto A, que es el que queremos enriquecer.

Ejemplo: cell.cross("ICAA_SPAIN","Codigo").cells["Genero"].value[0]



El resultado lo podemos ver en OpenRefine:



También podemos exportarlo en diferentes formatos como por ejemplo CSV. Aquí veríamos un ejemplo de campos enriquecidos: Directo, año de producción o género.

	B	C	D	E	F	G	H	I	J	K	L
1	idBNE	Autor Personas	Autor Entidades	Título uniforme	Título	Codigo_ICAA	ICAA_Director	ICAA año produccion	Genero ICAA	Otros Títulos	Información asociada al título
2	a4712411	Morante, Alexis ,	Los Delinqüentes (Grupo musical) //		VOLTERETA	43609	MARIO ALEXIS MOR	2010	Comedia		dirigido por Alexis Morant
3	a4712520	Morante, Alexis ,	Los Delinqüentes (Grupo musical) //		VOLTERETA	43609	MARIO ALEXIS MOR	2010	Comedia		directed by Alexis Morante
4	a4720394	San Mateo, José	María // García Ro	[Bullying. Español-Catal	BULLYING	135208	JOSETXO SAN MATE	2009	Drama		dirigida por Josecho San M
5	a4756847	Cameron, James ,	Academy Awards (	[Avatar (Película cinema	AVATAR	107804	LLUIS QUILEZ SALA	2005	Drama		guión y dirigida por James
6	a4770447	Huerga, Manuel , 1957- // Escribano, Francesc , 1958- // Ll			SALVADOR PU	111505	MANUEL HUERGA G	2006	Drama		dirigida por Manuel Huerg
7	a4770486	Balagueró, Jaume	Premis Gaudí 2ª , 2010 Barcelona // Festi		REC 2	155908	JAUME BALAGUERO	2009	Terror	Rec dos // (Rec) 2	dirigida por Jaume Balagu
8	a4780581	Ibáñez, Gabe , 19	Festival Internacional de Cinema de Catal		HIERRO	97208	GABRIEL IBAÑEZ RO	2009	Thriller		dirigida por Gabe Ibáñez ;
9	a4790102	Cameron, James ,	Academy Awards (	[Avatar (Película cinema	AVATAR	107804	LLUIS QUILEZ SALA	2005	Drama		guión y dirigida por James
10	a4792600	Reeve, Geoffrey , 1932-2010 // Tam		[Souvenir (Geoffrey Reev	SOUVENIR	153111	GERARDO CARRERAS	2013	Documental		directed by Geoffrey Reeve
11	a4793398	Agresti, Alejandro , 1961- // Mairal, Pedro , 1970- // Bossi,			UNA NOCHE C	117699	ALEJANDRO AGRESTI	1999	Drama		un film de Alejandro Agrest
12	a4797908	Monzón, Daniel ,	Premios Goya 24ª , 2010 Madrid // Meda		CELDA 211	127008	DANIEL MONZON JE	2009	Thriller		dirigida por Daniel Monzó
13	a4800176	Serra, Albert , 19	Mostra Internazionale d'Arte Cinematogra		HONOR DE CA	9906	ALBERT SERRA JUAN	2006	Drama	Albert Serra //	una película escrita, produ
14	a4803338	Pons, Ventura , 1945- // Cases, Carles , 1958- // Minguell, J			A LA DERIVA	85209	BONAVENTURA PON	2009	Drama		dirección y producción, Ve
15	a4813211	Recha, Marc , 1970- // Lamari, Nadine // Vidal, Jérôme // L			PETIT INDI	77307	MARC RECHA BATAL	2009	Drama		dirección, Marc Recha ; gui
16	a4826400	Aliaga, Adán // P	Semana Internacional de Cine de Valladol		ESTIGMAS	72807	AdÃin Aliaga Pasto	2009	Drama		guión y dirección, Adán Ali
17	a4840325	Moccia, Federico , 1963- // Rusic, R		[Scusa ma ti chiamo am	PERDONA SI T	105413	JOAQUIN LLAMAS	2014	Comedia		dirigida por Federico Mocc
18	a4843161	Las Heras, Pepe d	Mojinos Escozíos (Grupo musical) //		MUCHA SANG	97199	PEPE DE LAS HERAS	2002	Comedia de terror	Paul Naschy //	dirigida por Pepe de las He
19	a4843227	Yuzna, Brian , 1951- // Fernández, Julio , 1947- // Vázquez-F			ROTTWEILER	126003	BRIAN YUZNA	2005	Thriller	Paul Naschy //	dirigida por Brian Yuzna ;
20	a4847010	Torras, Carles , 1	Premis Gaudí 2ª ,	[Trash (Carles Torras). E	TRASH	136007	CARLES TORRAS PERE	2009	Drama		dirigida por Carles Torras
21	a4847385	Costa Perdomo, Renate // Andreu, Marta // Benito, Susana			CUCHILLO DE	75608	RENATE COSTA PERI	2010	Documental	108 Cuchillo de palo	written and directed by Ren
22	a4847982	Folk, Abel , 1959-	Dept. (Grupo musical) //		XTREMS	187309	ABEL FOLK GILSANZ	2009	Thriller		direcció, Abel Folk, Joan Ri

Como hemos podido ver, la herramienta es muy útil para mantener limpios los catálogos así como para enriquecerlos con datos de otras fuentes externas.