FACULTAD DE INGENIERÍA

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

Calidad de datos e información

Informe final: metodología de calidad CaDQM

Grupo 12

5 de junio de 2025

Facundo Barboza	5.211.682-9	facubarboza1702@gmail.com
Mauricio Simón	4.845.776-6	mauricio.simon@fing.edu.uy

Índice

1. Introducción

Dos librerías L1 y L2 se fusionan formando la librería NL. Los encargados de ambas librerías saben que los datos tienen muchos problemas de calidad y que éstos se verán potenciados por la integración de los mismos. El objetivo de este trabajo es evaluar la calidad de los datos de la librería NL, con las nuevas reglas de negocio y requisitos de usuarios y establecer un conjunto de especificaciones que permitan mejorar la calidad de los datos actuales y futuros.

Para esto se utilizara la metodología CaDQM, la cual permite evaluar la calidad de los datos en base al contexto de los mismos. En la fase 1 se realiza un acercamiento a los datos y la definición del modelo de contexto. Durante la fase 2 se define el modelo de calidad, se actualiza el modelo de contexto para integrar este y se ejecutan las métricas de calidad. La fase 3 no es ejecutada pero se provee de un breve plan para realizar la mejora de los datos.

Durante el transcurso de las etapas se utilizaran herramientas como Kettle, PosgreSQL, DataCleaner y Python.

El presente informe integra las entregas anteriores, con las correcciones necesarias y debidamente señaladas en el pie de página.

2. DQ Planning phase

2.1. Stage 1 Elicitation

2.1.1. Entradas

La etapa 1 no cuenta con entradas.

2.1.2. Selección del Data at Hand

Si bien el Data at Hand está conformado por el dataset de NL es necesario realizar la integración de los dataset de L1 y L2 para generarlo.

L1 esta conformado por 2 archivos en formato csv: books_data.csv, que contiene datos de descripción sobre los libros, y books_ratings.csv que contiene información sobre calificaciones dadas sobre ellos. Por otro lado, L2 esta formado por 3 archivos en el mismo formato: books.csv con información sobre cada libro, ratings.csv con las calificaciones y users.csv que contiene datos sobre los usuarios que realizan las calificaciones.

De la estructura de los datasets se detectan los siguientes problemas inmediatos al momento de llevar a cabo la integración:

- Identificador de libros: En el caso de L1 los libros de books_data.csv no contienen un identificador, sin embargo existe un identificador en books_rating.csv. Para el caso de L2 se tienen los libros identificados por el campo ISBN.

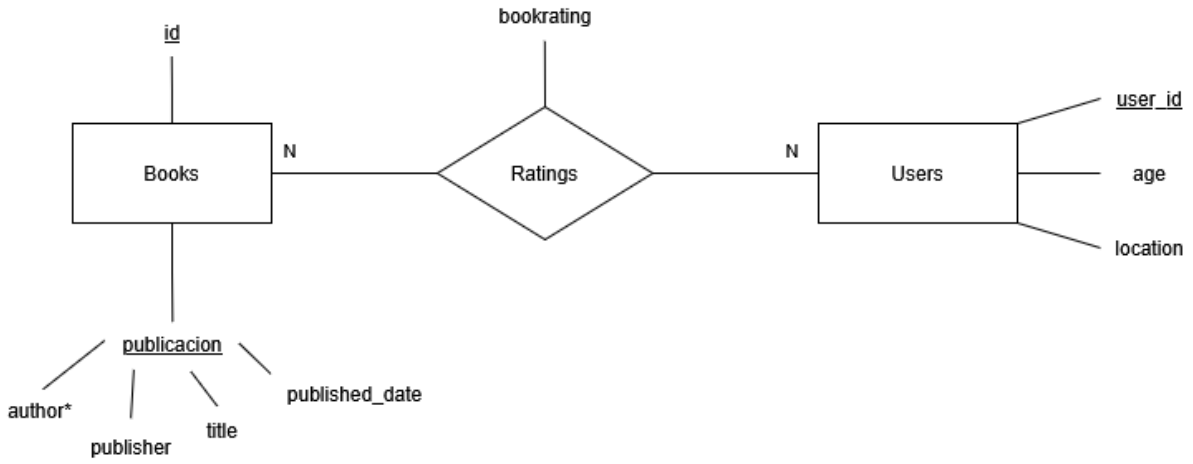


Figura 1: MER de la base de datos de NL

- Usuarios: Si bien para L1 books_rating.csv contiene para cada calificación un id de usuario y un nombre de perfil en el dataset no se encuentra más información sobre ellos.

Con el análisis previo y respetando las estructuras y datos originales se propone realizar la integración siguiendo la siguiente estrategia:

1. Se Genera una base de datos relacional utilizando PostgreSQL, esta surge de la integración de los datasets de ambas librerías. Utilizar una estructura relacional pretende facilitar la gestión de calidad en el futuro así como permitir un análisis exhaustivo de los datos resultantes de la fusión.
2. Para el caso de los identificadores, se asume que el id de books_rating.csv corresponde a los ISBN de los libros, y utilizando el título de estos, se relacionan con el atributo título del archivo books_data.csv para así obtener un ISBN candidato.
3. Para el caso de usuarios, se utilizará los datos de L2, ya que estos contienen la información de interés para el negocio.
4. Se genera una tabla de autores por ser atributo multivaluado en los datos de los libros, donde se generara una clave automática para identificarlos.

A partir de lo analizado y propuesto, se elabora un Modelo Entidad-Relación (Figura 1) que permite visualizar la estructura principal. *Notar que no se representan varios de los atributos que se originan de la unión de ambos datasets, sin embargo estos existen en el resultado de la integración.*

2.1.3. Análisis de los elementos de la organización

El dominio del problema son libros y sus clasificaciones por usuarios, en el cual se definen 3 tipos de usuarios: Administrador, Publicista, Analista de datos. Los cuales desempeñan las siguientes tareas:

- Administrador: Gestión de datos de la librería.
- Publicista: Recomendar y promocionar libros.
- Analista de datos: Analizar comportamientos, preferencias y relaciones entre los clientes.

Para cada uno de ellos se identifican los siguientes requisitos de calidad:

- Administrador
  - Necesidad de filtrado por varios campos.
- Publicista
  - NL debe ser actualizada todos los viernes.
  - Se espera que el 60 % de los libros tengan al menos un score mayor a 5.
  - El 80 % de los usuarios que califican los libros deben ser mayor a 18 años.
- Analista de datos
  - Al menos 95 % de los libros debe tener ISBN.
  - Al menos 95 % de los títulos bien escritos:
  - Al menos 95 % de los libros deben tener un nombre y apellido de autor.

2.1.4. Identificación de problemas de calidad

Los encargados de las librerías son conscientes de varios de los problemas que tienen actualmente. Al observar la integración y la estructura de los datasets de L1 y L2 se pueden anticipar una serie de problemas concretos sin necesidad de analizar los datos en detalle:

1. Libros con ISBN incorrectos.
2. Usuarios inexistentes.
3. Reseñas asociadas a usuarios incorrectos.
4. Posibilidad de datos duplicados sobre libros.
5. Escala de calificaciones incorrecta para gran parte de las reseñas.

2.1.5. Definición del modelo de contexto

- Dominio (DA): Librería con datos de libros y usuarios que realizan reseñas sobre ellos.
- Tipo de usuario
  - U1[1]: Administrador

[1] Todos los identificadores fueron colocados como corrección en la Tarea 2.

    - U2: Publicista
    - U3: Analista de datos
- Tareas
    - T1: Administración y gestión de la librería
    - T2: Recomendación de libros y promoción de la librería
    - T3: Análisis de dato
- Reglas de negocio
    - BR1: Cada libro deberá tener asociado un ISBN.
    - BR2: Cada libro deberá tener asociado, al menos, un título.
    - BR3: Cada libro deberá tener asociado al menos, un autor.
    - BR4: Cada libro deberá tener asociado al menos, un editor.
    - BR5: Se pretende tener al menos 500 libros en total.
    - BR6: El 20 % de ellos debe ser parte de la lista de los 100 mejores libros de Goodreads.
- Data filtering
    - Necesidad de filtrado por varios campos.
    - Muchas consultas a la base de datos por parte del administrador.
    - Algunos ejemplos de consultas son:
        - DF1: los libros cuya publicación sea del año actual.
        - DF2: el top 3 de los libros con mayor score, según el rating de los lectores.
        - DF3: los libros de la editorial Wiley.
- Requerimientos del sistema
    - SR: Los tiempos de respuesta del sitio web de NL no deben superar 3 segundos
- Requerimientos de calidad
    - DRQ1: La base de datos debe ser actualizada todos los viernes.
    - DRQ2: Se espera que el 60 % de los libros tengan al menos un score mayor a 5.
    - DRQ3: El 80 % de los usuarios que califican los libros deben ser mayor a 18 años.
    - DRQ4: Al menos 95 % de los libros debe tener ISBN.
    - DRQ5: Al menos 95 % de los títulos deben estar bien escritos.
    - DRQ6: Al menos 95 % de los libros deben tener un nombre y apellido de autor
- Metadatos
    - M: Descripción de los datasets de L1 y L2.
- Otros datos

- OD1: La lista de los 100 mejores libros de Goodreads será utilizada como documento de referencia.
- OD2: La base de datos isbndb.com[2] será utilizada como documento de referencia.

Comp. de CTX.	**Usuarios**			
	Todos	**U1**	**U2**	**U3**
Dominio de aplicación	DA			
Tareas		T1	T2	T3
Reglas de negocio	BR1, BR2, BR3, BR4, BR5, BR6			
Req. Sistema	SR			
Req. CD		DRQ4, DRQ7	DQR1, DQR2, DQR3	DQR4, DQR5, DRQ6
Filtrado de datos	DF1, DF2, DF3			
Metadatos	M			
Metadatos de CD	-			
Otros datos	OD1, OD2			

Cuadro 1: Componentes de contexto por usuario

2.1.6. Salidas

Las salidas de esta etapa corresponden al data at hand contenido en la Subsubsección 2.1.2, los problemas de calidad identificados en Subsubsección 2.1.4 y la definicion del modelo de contexto en Subsubsección 2.1.5.

2.2. Stage 2 Data Analysis

Esta etapa se desarrolló en paralelo con la etapa 3. Las salidas esperadas son un reporte del análisis de los datos, un reporte de los problemas de CD y el modelo de contexto actualizado.

[2]Se agrego como corrección en la Tarea 2.

Figura 2: Distribución de valores para ISBN en *Books*.

2.2.1. Entradas

Como entradas se recibe los data at hand descritos en Subsubsección 2.1.2, el reporte de problemas de CD de la Subsubsección 2.1.4 y el modelo de contexto en Subsubsección 2.1.5 así como los requerimientos que surgieron de la interacción con los usuarios (clases de consulta, e-mails o foro).

2.2.2. Data Profiling

El principal objetivo es estimar la calidad de las principales entidades y sus relaciones. Para cada una de ellas —*Books, Users y Reviews*— se analizo el volumen, se realizó una búsqueda de valores nulos, patrones, identificadores duplicados, dependencias funcionales y un primer acercamiento a los requerimientos de calidad definidos por los usuarios, tales como el rango de edad de los usuarios y la disponibilidad de los autores.

Los principales elementos del contexto que fueron analizados incluyen: las tareas *(Task at hand)*, reglas de negocio, filtrado de datos *(Data Filtering)*, requerimientos del sistema, dominio de la aplicación y un primer acercamiento a los requerimientos de calidad.

Para realizar el *data profiling* se empleó la herramienta DataCleaner con el fin de obtener un análisis preliminar de los datos y estudiar aspectos particulares de cada entidad. Para realizar análisis más complejos, especialmente entre entidades, se utilizaron consultas SQL.

Books

La tabla *Books* cuenta con el atributo id, que representa el ISBN, este es el identificador de un libro. La Figura 2 muestra que existen 476.321 tuplas en la tabla, en donde el ISBN es único y ademas no nulo.

Para el atributo published_date, que representa la fecha de publicación la distribución de valores representada en Figura 3 muestra 23.999 valores nulos. La búsqueda de patrones de la Figura 4 resalta la diversidad de los mismos. Se reconocen formatos estándares como ####-##-##. Algunas tuplas parecen tener información incorrecta o formatos no estandarizados.

El atributo publisher representa la editorial la Figura 5 muestra que existen 74.226 tuplas con publisher NULL. Este atributo, junto a title, author_id y published_date deberian ser suficientes para reconocer un libro. La Figura 6 muestra que no existe ninguna tupla que no cuente con ninguno de ellos. Por otra parte, hay 7 libros que tienen title con el valor NULL (Figura 7).

Value distribution

(book PUBLISHED DATE)

Value	COUNT(*)
<null>	23999

Figura 3: Distribución de valores para published_date en *Books*.

	Match count	Sample
####	359504	1954
####-##-##	81976	2003-11-01
<null>	23999	<null>
####-##	10496	2018-06
####*	142	2005*
###?	75	196?
##??	49	19??
####-##-?????:##:???	47	2020-07-21T18:13:34Z
####-##-?????:##.##+##:##	10	2016-11-15T00:00:00+01:00
Aaaaaaaa Aaaaaaaaaa	10	Luella Hill
AA Aaaaaaaaaa Aaa	2	DK Publishing Inc
Aaaaaa A. Aaaaaaaaaaa	2	George H. Scherr
#	1	0
Aaaaaaaa"	1	Learning"
####-##-?????:##.##-##.##	1	2021-10-01T00:00:00-04:00
A.A. Aaaaaaaaaaa	1	K.C. Constantine
AAAAAA A. AAAAAA	1	ROBERT A. WILSON
Aaaa Aaaaaaaa Aaaaa	1	John Alderson Foote
Aaaaaaaa aa Aaaaaaaaa	1	Salvador de Madariaga
Aaaaaaaaa	1	Gallimard
\"Aaaaaaa Aaaa\""	1	\"Freedom Song\""

Figura 4: Distribución de valores para published_date en *Books*.

Value distribution

(book PUBLISHER)

Value	COUNT(*)
<null>	74226

Figura 5: Distribución de valores de publisher en *Books*.

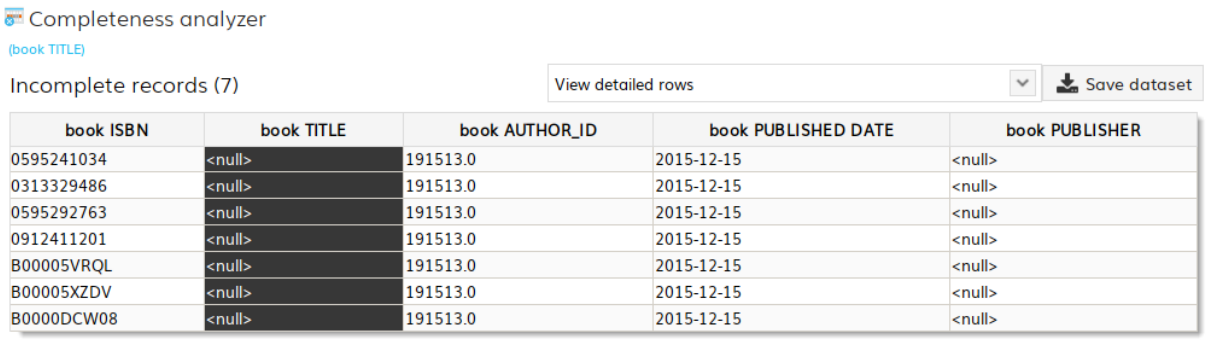
Completeness analyzer

(book PUBLISHER,book TITLE,book AUTHOR_ID,book PUBLISHED DATE)

Incomplete records (0)

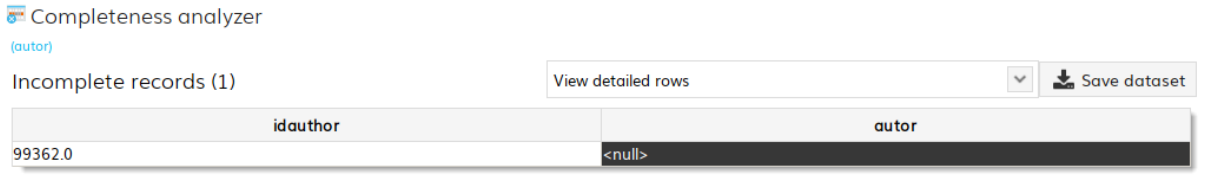
No records to display.

Figura 6: Completitud de atributos *Books*.


Completeness analyzer
(book TITLE)
Incomplete records (7)
View detailed rows
Save dataset

book ISBN	book TITLE	book AUTHOR_ID	book PUBLISHED DATE	book PUBLISHER
0595241034	<null>	191513.0	2015-12-15	<null>
0313329486	<null>	191513.0	2015-12-15	<null>
0595292763	<null>	191513.0	2015-12-15	<null>
0912411201	<null>	191513.0	2015-12-15	<null>
B00005VRQL	<null>	191513.0	2015-12-15	<null>
B00005XZDV	<null>	191513.0	2015-12-15	<null>
B0000DCW08	<null>	191513.0	2015-12-15	<null>

Figura 7: Densidad de title en *Books*.


Completeness analyzer
(autor)
Incomplete records (1)
View detailed rows
Save dataset

idauthor	autor
99362.0	<null>

Figura 8: id asignado al autor NULL.

Los autores son representados en la tabla *Author*. Sin embargo, esta fue generada posterior a la integración. El atributo identificador del autor es auto generado por lo que es único y todas las tuplas lo tienen definido. Esta decisión de diseño le asigna un identificador al autor NULL, como muestra la Figura 8 el id 99362.0, corresponde al mismo. Contemplando este resultado en la tabla *Books* existen 29.971 libros asignados a este identificador.

Por último en la tabla libros se debe cumplir la dependencia funcional 1 para todas las tuplas. Dado que estos atributos deberían ser suficientes para identificar un único libro publicado. La consulta de la Figura 9 arroja que 10.282 tuplas violan esta dependencia. La Figura 10 muestra ejemplos de estas tuplas.

$$\{title, author, publish\_date, publisher\} \rightarrow ISBN \tag{1}$$

Users

La tabla *Users* tiene 3 atributos, *user_id* que es el identificador, *location* que representa la ubicación y *age* que representa su edad. La Figura 11 muestra que existen 278.858 usuarios, donde todos tienen identificador único y no nulo. El atributo *location* no tiene valores en NULL. Sin embargo, como muestra la Figura 12, existen algunas tuplas con caracteres mal codificados y 38.012 que tienen valor único.

La Figura 13 muestra la distribución del atributo *age*. 167 tuplas tienen valor único y 110.761. Además el valor mas alto es de 244 y el menor de 0 mientras que el promedio es de 34. Se considera de manera arbitraria el rango 18 a 90 como valido la Figura 14 muestra que 155.356 usuarios pertenecen al rango, mientras que 430 lo superan, varios de estos valores son supe-

Query Query History

```
1  SELECT title, author_id, publisher, published_date, COUNT(DISTINCT id) AS isbn_variants
2  FROM libros
3  GROUP BY title, author_id, publisher, published_date
4  HAVING COUNT(DISTINCT id) > 1;
5
```

Data Output Messages Notifications

```
Successfully run. Total query runtime: 17 secs 236 msec.
10282 rows affected.
```

Figura 9: Tuplas que violan dependencia funcional de ISBN.

	title text	author_id double precision	publisher text	published_date text	id text
1	'New Raiments of Self': African American Clothing in the Antebellum South (Dress, Body, Cult…	154812	Berg Publishers	1997-06-01	1859731848
2	'New Raiments of Self': African American Clothing in the Antebellum South (Dress, Body, Cult…	154812	Berg Publishers	1997-06-01	1859731899
3	1 is one	239018	Simon and Schuster	2015-01-27	B0007EYUM8
4	1 is one	239018	Simon and Schuster	2015-01-27	B0000CJK13
5	1,000 Questions of Who You Are and Your View on Life	192381	Protea Pub	2004-04-01	1593440480
6	1,000 Questions of Who You Are and Your View on Life	192381	Protea Pub	2004-04-01	1593440472
7	1,000 Years, 1,000 People: Ranking the Men and Women Who Shaped the Millennium	100868	Kodansha Amer Incorporat…	1998	1568362730
8	1,000 Years, 1,000 People: Ranking the Men and Women Who Shaped the Millennium	100868	Kodansha Amer Incorporat…	1998	0965088189
9	10,000 Dreams Interpreted or What's in a Dream	152396	Bell	1988	B000K52CK6
10	10,000 Dreams Interpreted or What's in a Dream	152396	Bell	1988	B000MVQG…
11	10,000 dreams interpreted: A dictionary of dreams	152396	Barnes & Nobles Books	1995	1566196264
12	10,000 dreams interpreted: A dictionary of dreams	152396	Barnes & Nobles Books	1995	1566196256
13	101 Secrets a Cool Mom Knows	236669	Thomas Nelson	2005-03-05	1401601359
14	101 Secrets a Cool Mom Knows	236669	Thomas Nelson	2005-03-05	B0006959XS

Figura 10: Ejemplos de tuplas que violan DF.

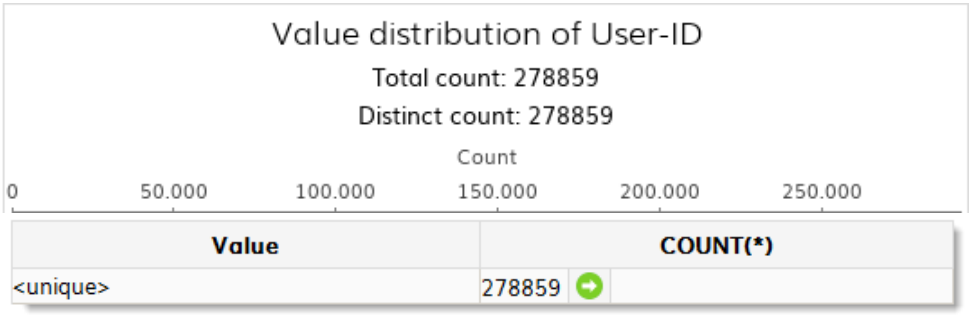
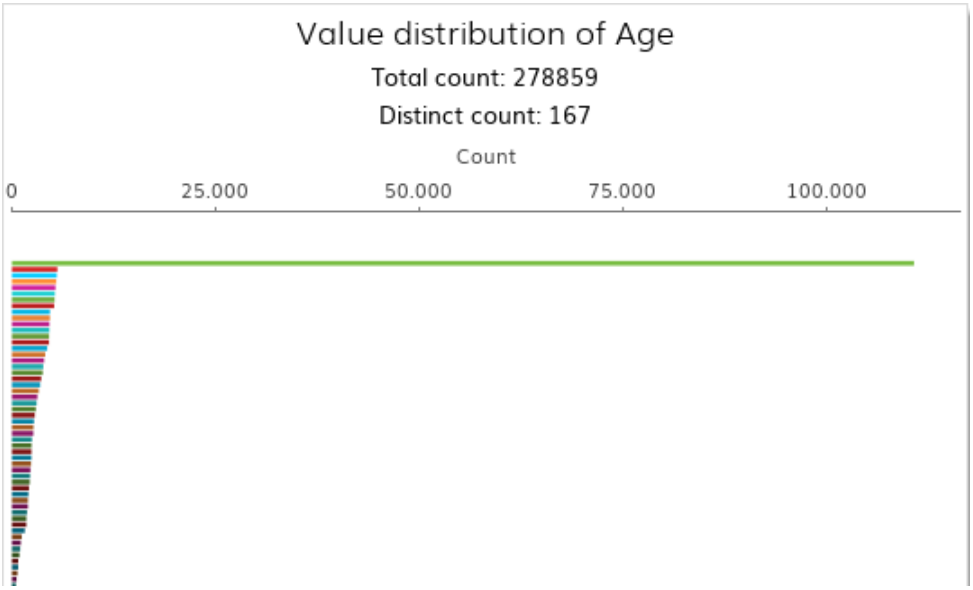


Figura 11: Distribución de valores para user_id

z�rich, z�rich, stadt, switzerland	2
�bach-palenberg, nordrhein-westfalen, germany	2
�berlingen, baden-wuerttemberg, germany	2
ød�kra, sk�ne, sweden	2
�hringen, baden-wuerttemberg, germany	2
�hringen, baden-w�rttemberg, germany	2
�kersberga, stockholm, sweden	2
�kersberga, uppland, sweden	2
�stanbul, marmara, turkey	2
�stersund, j�mtland, sweden	2
�vila, castilla y le�n, spain	2
�vila, �vila, spain	2
�vora, �vora, portugal	2
��d�, mazowsze, poland	2
����, n/a, china	2
□□□□, □□, □□□	2
□□□□□□, □□□, china	2
������, ������, china	2
<unique>	38012

Figura 12: Distribución de valores para location



Number analyzer

(Age (as number))

	Age (as number)
Row count	278859
Null count	110763
Highest value	244
Lowest value	0
Sum	5.841.577
Mean	34,751
Geometric mean	0
Standard deviation	14,428
Variance	208,17
Second moment	34.992.335,154
Sum of squares	237.995.511

Figura 13: Distribución de valores para age

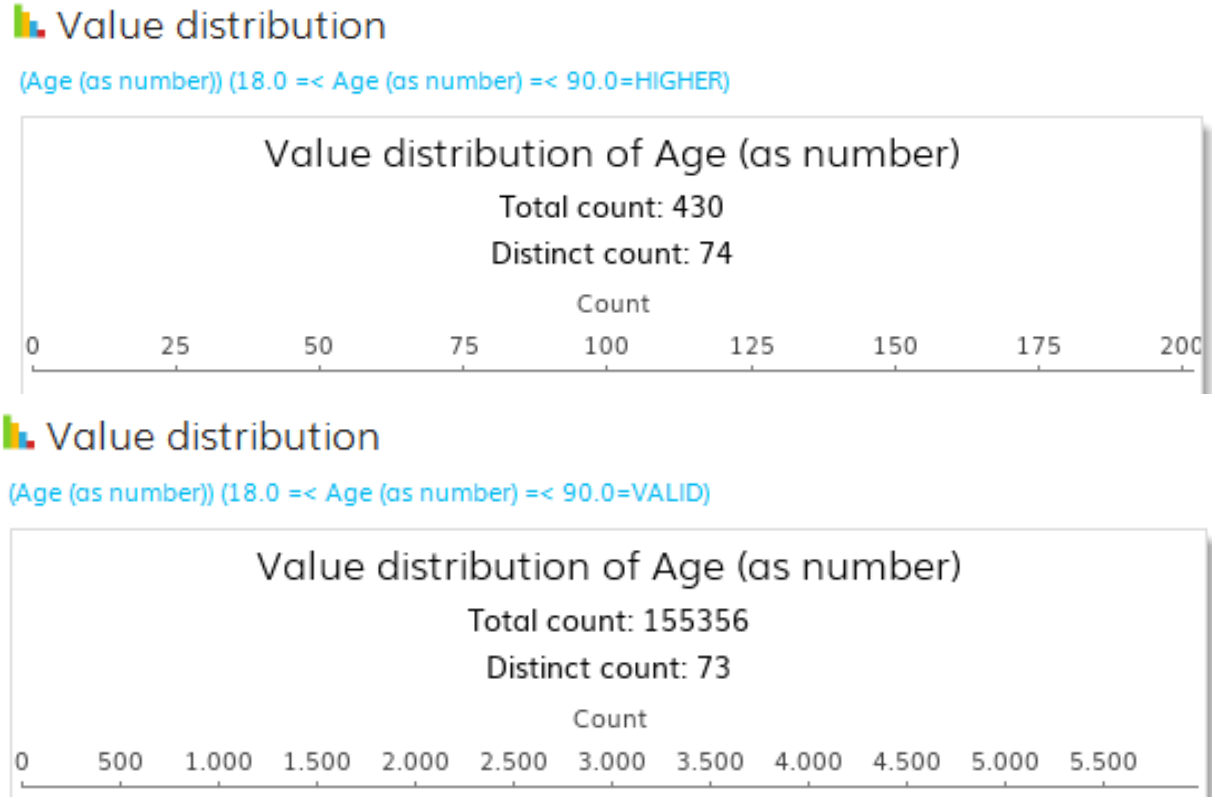


Figura 14: Rango $age \in [18, 90]$

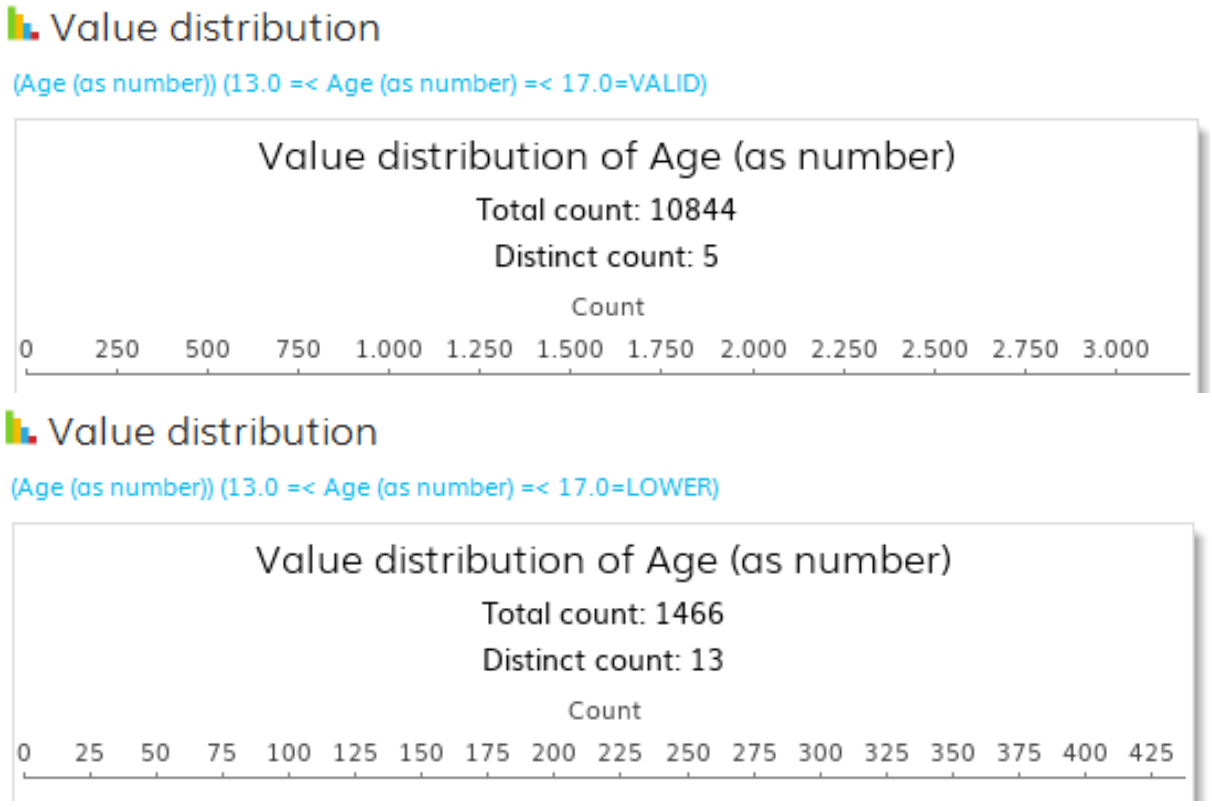
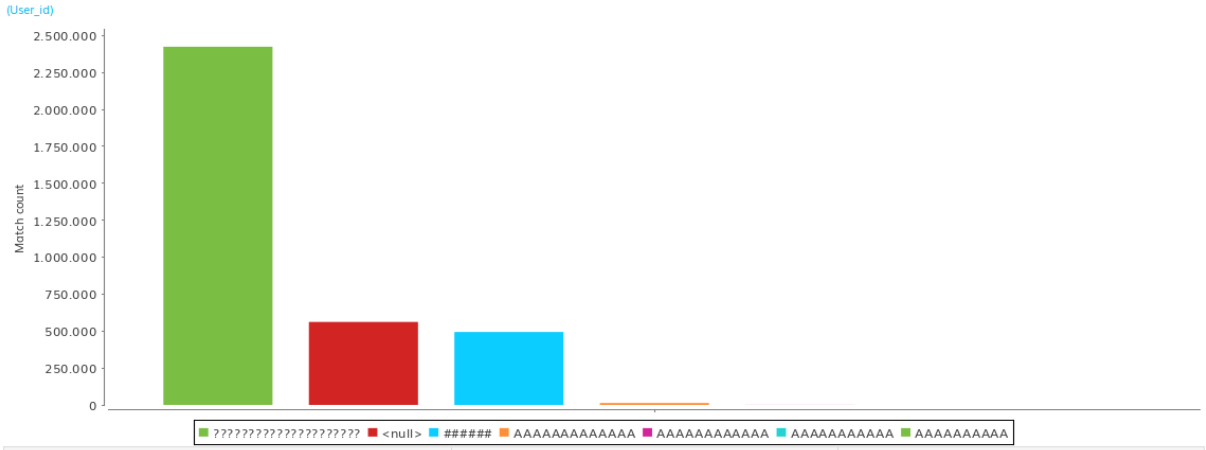


Figura 15: $age \in [13, 17]$



	Match count	Sample
?????????????????????	2424695	A2F6NONFUDB6UK
<null>	561787	<null>
######	493813	276729
AAAAAAAAAAAA	13007	ARXQPCLEIGYIR
AAAAAAAAAAA	503	AYTOTYTMWCSN
AAAAAAAAAA	5	ANPLDIOPBUG
AAAAAAAAA	3	ALJEWNHIED

Figura 16: Distribución de user_id en tabla *Ratings*.

riores a 100. Para el rango 13 a 17 la Figura 15 muestra que 10.844 usuarios pertenecen al mismo y otros 1466 se encuentran por debajo de 13. En particular 416 y 288 usuarios tienen 0 y 1 respectivamente. Para los análisis de rangos se ignoran los valores nulos.

Ratings

La tabla *Ratings* pertenece a la relación entre usuarios y libros la Figura 16 muestra que de las 3.493.813 tuplas, 561.787 tienen el atributo user_id en NULL. Además se identifican diferentes formatos para el mismo. La otra clave foránea de la tabla es el atributo ISBN, que es clave de *Books* la Figura 17 muestra diferentes formatos para el ISBN, donde varios de ellos no cumplen el estándar.

Un usuario no debería poder calificar a un libro mas de una vez Figura 18 muestra una consulta SQL que arroja que existen 76.697 calificaciones donde existe otra con el mismo usuario para el mismo libro. Además como se ve en la Figura 19, 3.000.000 de calificaciones tienen usuarios inexistentes. Este valor es esperable debido a la integración y el resultado de la tabla *Users*.

El atributo *score* representa el puntaje que le otorga un usuario a un libro. La Figura 20 muestra que los scores tienen dos formatos, uno con el separador decimal y el otro sin el. El mínimo es 0, el máximo 10 y el promedio es 4.0. La Figura 21 muestra que hay 50.059 calificaciones para las cuales no existe un libro con el ISBN asociado a ella.

	Match count	Sample
##########	1767268	0826414346
?????????????	1726101	B000OVX7JG
###########.#	196	2.02.032126.2
##########/##########	58	100940/86
#########=##########	18	006=094049
## #### ###### ###	10	0 907 062 008
##########.#.A	9	2.02.025462.X
######/####/###	9	30/6646/1
AAAAAAAAAAAA	9	DITISEENSOORT
##########.	7	0684826801.
##########/A.	7	7560013805/H.
##########/AA	7	9029556366/NU
##########>>##	7	0440203856>>5
##########(AA	6	0060911131(PB
A.#########	5	B.265121972
AAAA:########	5	ISBN:08112003
AAAAAAAAAA	5	NONFICTION
???.??????	4	M47.O57202
AAAAAAAA	4	NOTGIVEN
AAAAAAAAAAA	4	THEFLYINGACE
##########.#.	3	1740451456.1.
###*#########	3	112*0854
##/###/#####/#	3	0/330/25864/8
\##########\	3	\0432534220\
##########	2	0330299891
# ##### #### a	2	0 85550 000 x

Figura 17: Distribución de ISBN en tabla *Ratings*.

```sql
SELECT
  "User_id",
  "Id",
  COUNT(*) AS cantidad_calificaciones
FROM public.ratingsall
GROUP BY "User_id", "Id"
HAVING COUNT(*) > 1;
```

Data Output | Messages | Notifications

```
Successfully run. Total query runtime: 1 min 14 secs.
76697 rows affected.
```

	User_id text	Id text	cantidad_calificaciones bigint
1	A02660181QI9HHAVFK06O	B000NDSX6C	2
2	A03816223LL3Q1P48HRU	B000NDSX6C	2
3	A07532193EC6QULM9ZS...	B000GQG5...	2
4	A08604952BQMNOFP9OI...	B000GQG5...	2
5	A08854851CPI1JRVJJQVQ	B000GQG5...	2
6	A10022CWZZNE5U	B0009RJUVE	2

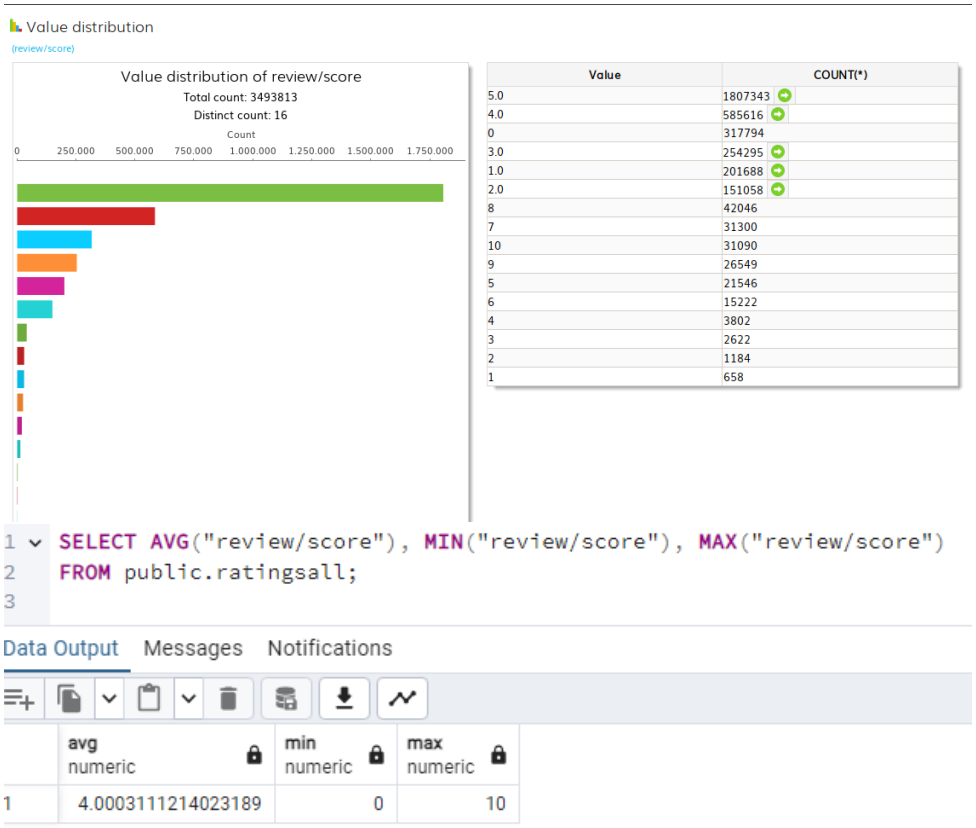
Figura 18: Distribución de ISBN en tabla *Ratings*.

```sql
SELECT r.*
FROM public.ratingsall r
LEFT JOIN public.users u
  ON r."User_id"::TEXT = u."User-ID"::TEXT
WHERE u."User-ID" IS NULL;
```

Data Output | Messages | Notifications

```
Successfully run. Total query runtime: 2 min 52 secs.
3000000 rows affected.
```

Figura 19: Califiaciones con usuarios inexistentes.



Value	COUNT(*)
5.0	1807343
4.0	585616
0	317794
3.0	254295
1.0	201688
2.0	151058
8	42046
7	31300
10	31090
9	26549
5	21546
6	15222
4	3802
3	2622
2	1184
1	658

```sql
SELECT AVG("review/score"), MIN("review/score"), MAX("review/score")
FROM public.ratingsall;
```

Data Output | Messages | Notifications

	avg numeric	min numeric	max numeric
1	4.0003111214023189	0	10

Figura 20: Distribución de *score*.

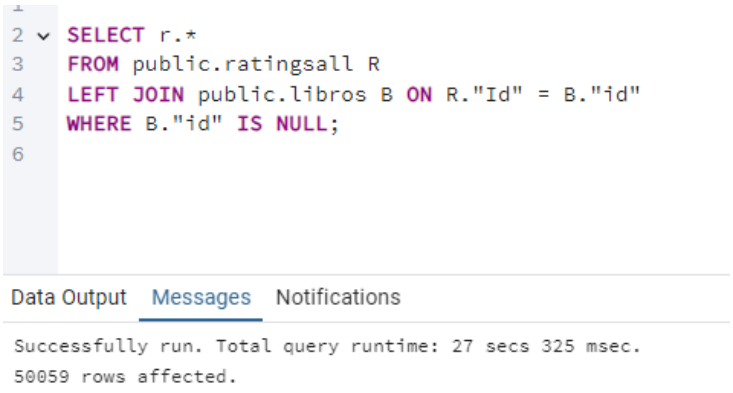


Figura 21: Calificaciones para las cuales no existen libros.

2.2.3. Identificación de problemas de calidad

De la actividad de *data profiling* realizada previamente se identifican los siguientes problemas de calidad[3]:

- DQP1: Valor del atributo ISBN con patrones o formatos distintos, por tanto podrían ser también incorrectos.
- DQP2: Cierta cantidad de libros con autores sin identificar.
- DQP3: Gran cantidad de nulos en la fecha de publicación.
- DQP4: La fecha de publicación no sigue un formato claro.
- DQP5: Título, autor, editorial y fecha de publicación no determinan un único ISBN.
- DQP6: Existen libros duplicados.
- DQP7: Un gran número de usuarios tiene el campo Age con valores nulos.
- DQP8: Un número considerable de usuarios tiene posiblemente valores incorrectos en cuanto a su edad (se detectan valores muy grandes o muy chicos).
- DQP9: Un número importante de locaciones tienen caracteres extraños, valores en blanco/vacíos.
- DQP10: Múltiples ratings para un mismo libro con un mismo usuario. Las calificaciones no son únicas por libro.
- DQP11: Gran número de Ratings para libros con el atributo ISBN en formatos no uniformes e incorrectos.
- DQP12: Gran cantidad de ratings con usuarios inexistentes en la tabla de usuarios.

[3]Los problemas identificados como DQP14, DQP15, DQP16 se agregaron como corrección en la tarea 2, así como los identificadores para todos ellos.

- DQP13: Gran número de ratings para libros inexistentes.
- DQP14: Gran cantidad de nulos en el campo editorial.
- DQP15: Una pequeña cantidad (7) libros tiene el campo título como nulo.
- DQP16: Los ratings se dividen en 2 escalas con un diferentes formatos, una del 1 al 5 y otra del 1 al 10.

2.2.4. Estimación de DQ

El data profiling revela que los datos poseen muchos errores sintácticos, como para el caso del ISBN y la fecha de publicación. Además, el no cumplimiento de la dependencia funcional1 pone en riesgo la estructura relacional y representa un problema de calidad importante, ya que estos atributos son identificadores de los libros.

Para los usuarios, el mayor trabajo se centra en los rangos de edades, donde algunos son nulos y otros parecen ser incorrectos.

Se estima que la tabla de calificaciones sera la que tiene mayor cantidad de problemas de calidad. Es la de mayor volumen y de surgir entre la relación de usuarios y libros podría acarrear problemas de ellas. Se anticipa que la mayor carga de trabajo estará sobre esta tabla.

2.2.5. Actualización del Modelo de Contexto

- Dominio (DA): Librería con datos de libros y usuarios que realizan reseñas sobre ellos.
- Tipo de usuario
  - U1: Administrador
  - U2: Publicista
  - U3: Analista de datos
- Tareas
  - T1: Administración y gestión de la librería
  - T2: Recomendación de libros y promoción de la librería
  - T3: Análisis de dato
- Reglas de negocio
  - BR1: Cada libro deberá tener asociado un ISBN.
  - BR2: Cada libro deberá tener asociado, al menos, un título.
  - BR3: Cada libro deberá tener asociado al menos, un autor.
  - BR4: Cada libro deberá tener asociado al menos, un editor.
  - BR5: Se pretende tener al menos 500 libros en total.
  - BR6: El 20 % de ellos debe ser parte de la lista de los 100 mejores libros de Goodreads.

- Data filtering
  - Necesidad de filtrado por varios campos.
  - Muchas consultas a la base de datos por parte del administrador.
  - Algunos ejemplos de consultas son:
    - DF1: los libros cuya publicación sea del año actual.
    - DF2: el top 3 de los libros con mayor score, según el rating de los lectores.
    - DF3: los libros de la editorial Wiley.
- Requerimientos del sistema
  - SR: Los tiempos de respuesta del sitio web de NL no deben superar 3 segundos
- Requerimientos de calidad
  - DRQ1: La base de datos debe ser actualizada todos los viernes.
  - DRQ2: Se espera que el 60 % de los libros tengan al menos un score mayor a 5.
  - DRQ3: El 80 % de los usuarios que califican los libros deben ser mayor a 18 años.
  - DRQ4: Al menos 95 % de los libros debe tener ISBN.
  - DRQ5: Al menos 95 % de los títulos deben estar bien escritos.
  - DRQ6: Al menos 95 % de los libros deben tener un nombre y apellido de autor.
  - DRQ7: Las fechas deben seguir un formato estandarizado.
  - DRQ8: Los libros deben contener al menos 1 imagen.
  - DRQ9: Los nombres propios deben estar estandarizados, minúsculas con mayúscula inicial y sin punto final.
- Metadatos
  - M: Descripción de los datasets de L1 y L2.
- Otros datos
  - OD1: La lista de los 100 mejores libros de Goodreads será utilizada como documento de referencia.
  - OD2: La base de datos isbndb.com será utilizada como documento de referencia.

Comp. de CTX.	Usuarios			
	Todos	**U1**	**U2**	**U3**
Dominio de aplicación	DA			
Tareas		T1	T2	T3
Reglas de negocio	BR1, BR2, BR3, BR4, BR5, BR6			
Req. Sistema	SR			
Req. CD		DRQ4, DRQ7	DQR1, DQR2, DQR3, DRQ8, DRQ9	DQR4, DQR5, DRQ6, DRQ7, DRQ9
Filtrado de datos	DF1, DF2, DF3			
Metadatos	M			
Metadatos de CD				
Otros datos	OD			

Cuadro 2: Modelo de contexto actualizado

2.2.6. Salidas

Las salidas de esta etapa están constituidas por el análisis de *Data Profiling* de la Subsubsección 2.2.2, la estimación de calidad de Subsubsección 2.2.4 y la actualización del modelo del contexto en Subsubsección 2.2.5.

2.3. Stage 3 User requirements analysis

Durante la etapa 3 se analiza en profundidad los requerimientos del usuario, las salidas esperadas son nuevos requerimientos de calidad y el modelo de contexto actualizado. Esta etapa fue ejecutada en paralelo con la etapa 2.

2.3.1. Entradas

Las entradas de esta etapa consisten en los Data at hand, el reporte de calidad y el modelo de contexto.

2.3.2. Requerimientos de usuario

Estos requerimientos de calidad fueron relevados durante la etapa 3. Estos surgen del intercambio con los usuarios (docentes) en las clases de consulta, e-mails y foros.

- Las fechas deben estar en un formato estandarizado.
- Los libros deben tener alguna imagen.
- Los nombres propios deben estar estandarizados, minúsculas con mayúscula inicial y sin punto final.

2.3.3. Actualización del Modelo de Contexto

Permanece igual que en la etapa 2 descrito en Subsubsección 2.2.5, donde ya había sido incluido los requerimientos de usuarios nuevos.

2.3.4. Salida

Las salidas de esta etapa consisten en lo descrito en la Subsubsección 2.3.2 y el modelo de contexto actualizado descrito en Subsubsección 2.2.5.

3. DQ Assessment phase

3.1. Stage 4 DQ Model Definition

Esta etapa consiste en definir el modelo de calidad de datos y se compone por las actividades de priorización de problemas de calidad, selección de dimensiones y factores de calidad, definición de métricas de calidad e implementación de métodos de calidad.

3.1.1. Entradas

Como entradas para esta parte se encuentran el reporte del análisis de requerimientos de usuarios presentado en Subsubsección 2.3.2, el reporte del análisis de datos de la Subsubsección 2.2.2, el reporte de problemas de CD Subsubsección 2.2.3 y el Modelo de Contexto de la Subsubsección 2.2.5.

3.1.2. Priorización de problemas de calidad

Los problemas de calidad se priorizan en las categorías alta, media y baja como muestra el Cuadro 3.

Prioridad alta	Aquellos que incumplan las reglas de negocio o impidan la correcta ejecución de las tareas de los usuarios.
Prioridad media	Aquellos que afecten directamente a los requerimientos de calidad o dificulten las tareas de los usuarios.
Prioridad baja	Aquellos que no afectan, o afectan en menor medida, a los requerimientos de calidad.

Cuadro 3: Priorización de los problemas

Problemas de calidad con prioridad alta

Cuadro 4: Problemas con prioridad alta

DQP	Elementos del Modelo de contexto en conflicto	Análisis
DQP1	T1, T2, T3, BR1, DRQ4	ISBN con patrones o formatos incorrectos evitan el cumplimiento de las tareas de todos los usuarios. Además, formatos incorrectos son funcionalmente equivalentes a ISBN nulos.
DQP2	BR3, BR6, DRQ6, DRQ9	Al no contener autor dificulta identificar el libro, se afectan reglas de negocio y de calidad. En particular, evita la correcta identificación de libros con la referencia externa a la tabla de autores.
DQP3	T3, DF1, DRQ7	Los valores nulos en la fecha de publicación, afectan directamente a T3, DF1 y DRQ7. Además, este problema se vincula estrechamente con DQP5.
DQP4	T3, DF1, DRQ7	Fechas no estandarizadas son, en la práctica, equivalentes a nulas. Esto afecta directamente a T3, DF1 y DRQ7. Además, este problema se vincula estrechamente con DQP5.
DQP5	BR2, BR3, BR4, BR5, DRQ5, DRQ6, DRQ7	El incumplimiento de la dependencia funcional afecta a varias reglas de negocio y problemas de calidad. Este problema tiene una estrecha relación con el resto.

DQP	Elementos del Modelo de contexto en conflicto	Análisis
DQP6	T1, T3, BR5, BR6	La duplicidad de los libros afecta el correcto desempeño de los análisis y la gestión de la librería. Además, puede comprometer el cumplimiento de BR5, ya que una cantidad significativa de duplicados podría inflar artificialmente el total de libros, dando lugar a un cumplimiento ficticio de la regla. De forma análoga, los duplicados pueden distorsionar el cálculo del porcentaje de libros que pertenecen al recurso externo, afectando el cumplimiento de BR6.
DQP14	BR4, DF3	Además de incumplir la regla de negocio, la existencia de nulos en el campo editorial dificultan las tareas de consulta.
DQP15	BR2, BR6, DRQ5	Aunque el número de casos es bajo, la ausencia de título en libros infringe BR2, dificulta la verificación contra el recurso externo (BR6), y afecta el cumplimiento de DRQ5. Por afectar reglas de negocio, su prioridad es alta.

Problemas de calidad con prioridad media

DQP	Elementos del Modelo de contexto en conflicto	Análisis
DQP7	T2, T3, DRQ3	Afecta T2 y T3, pero no las impide. Además afecta directamente a DRQ3.
DQP10	T2, T3, DF2, DRQ2.	Afecta T2 pero no la impide. DRQ2 se ve afectado ya que las calificaciones de los libros podrían ser artificiales.
DQP11	T2, T3, DF2, DRQ2	La presencia de ISBN en formatos no estandarizados dentro de las calificaciones impide vincular correctamente los ratings con los libros reales. Esto podría afectar incluso a BR1 ya que un ISBN malformado es funcionalmente equivalente a un valor nulo. Teniendo incluso un impacto estructural. Sin embargo, se lo califica como prioridad media en lugar de alta debido a que los ISBN son definidos en la entidad Libro, y por eso es tomado como incorrecto el Rating.
DQP12	T2, T3, DRQ3	El usuario inexistente es un usuario Anónimo, las tareas se dificultan pero no se impiden. Además DRQ3 se ve afectado directamente. No considerar el usuario inexistente como el Anónimo tiene un impacto en la estructura de la base.
DQP16	T2, DF2, DRQ2	La coexistencia de 2 escalas de calificación distorsiona los resultados que se obtengan de DF2 y dificulta T2 pero no impide el uso de los datos. Afecta directamente a DRQ2.

Cuadro 5: Problemas con prioridad media

Problemas de calidad con prioridad baja

DQP	Elementos del Modelo de contexto en conflicto	Análisis
DQP8	DRQ3	Estos datos afectan a DRQ3 y se pueden tratar como datos atípicos, excluyendolos de los análisis.
DQP9	DRQ9	Si se espera tomar las locaciones, con un formato determinado, por ejemplo “Montevideo, Uruguay” Este podría entrar en conflicto con DRQ9. Sin embargo, las locaciones no interfieren con las reglas de negocio y tienen poco impacto en las tareas de los usuarios.
DQP13	DF2	Se dificultan las tareas de filtrado, en particular DF2. El caso es similar a DQP11. Sin embargo, los libros no existen en la base de datos. Por tanto estos ratings se pueden ignorar.

Cuadro 6: Problemas con prioridad baja

3.1.3. Selección de dimensiones y factores de calidad

Dada la priorización de los problemas de calidad y considerando el modelo de contexto, se seleccionan las siguientes dimensiones y factores de calidad.

Exactitud

Factores asociados: Exactitud semántica, Exactitud sintáctica

Análisis: La sintaxis de los atributos es crítica para el funcionamiento del sistema de información estudiado. Se identificaron múltiples problemas de calidad relacionados con los diferentes formatos o errores en atributos clave como por ejemplo el ISBN de los libros. Además, varios requerimientos de calidad están directamente asociados a la sintaxis de los atributos. Por otro lado, la exactitud semántica resultará útil para medir, por ejemplo, la cantidad de ISBN que realmente existen.

Completitud

Factores asociados: Cobertura, Densidad

Análisis: La cobertura es clave para medir algunos requerimientos de calidad, por ejemplo, la cantidad de libros del top 100 de Goodreads presentes en el SI. En el caso de la densidad, es crítico contar con la mayor cantidad de datos posibles para los libros, puesto que estos resultan de interés y aportan un valor considerable a la organización. Este tiene asociados

múltiples problemas de calidad, así como requerimientos de calidad y reglas de negocio.

Consistencia

Factores asociados: Integridad intra-relación, Integridad inter-relación

Análisis: Es esperable que al combinar dos sistemas de información se obtengan problemas de consistencia, varios problemas de calidad identificados están directamente relacionados. Un caso destacable es la dependencia funcional de los datos de identificación del libro con el ISBN. Además, al existir múltiples orígenes de datos la estructura del sistema de información también puede contener errores, conteniendo tuplas que se asocian con elementos inexistentes.

Unicidad

Factores asociados: No-duplicación, No-contradicción

Análisis: Es importante evaluar esta dimensión y factores, ya que debido al origen de los datos es esperable que existan datos duplicados y muchos de ellos contradictorios. Algunas tareas como las estadísticas y los análisis se podrían alterar drásticamente y muchos requerimientos de negocio y de calidad se podrían ver afectados por contradicciones.

3.1.4. Definición de métricas de calidad

Métrica	**M1-ExactSintáctica-Bool**
Descripción	Mide si un dato está escrito correctamente.
Granularidad	Celda
Dominio del resultado	{0,1}

Cuadro 7: M1-ExactSintáctica-Bool

Métrica	**M2-ExactSemántica-Bool**
Descripción	Mide si un dato existe en la realidad.
Granularidad	Celda
Dominio del resultado	{0,1}

Cuadro 8: M2-ExactSemántica-Bool

Métrica	**M3-CompDensidad_CamposIdentificadores**
Descripción	Mide el grado de densidad de una tupla.
Granularidad	Tupla
Dominio del resultado	[0,1]

Cuadro 9: M3-CompDensidad_CamposIdentificadores

Métrica	**M4-CompCobertura_Top100**
Descripción	Mide la cobertura de libros del top 100 que efectivamente están en la tabla *Books*.
Granularidad	Tabla
Dominio del resultado	[0,1]

Cuadro 10: M4-CompCobertura_Top100

Métrica	**M5-ConsIntraRelacion-Libros**
Descripción	Mide la consistencia de libros en la tabla *Books*, donde tilte, author_id, publisher y published_date corresponden a un único id.
Granularidad	Tupla
Dominio del resultado	{0,1}

Cuadro 11: M5-ConsIntraRelacion-Libros

Métrica	**M6-ConsInterRelacion-ISBNRatings**
Descripción	Mide la consistencia entre los id de los libros en *Ratings* y el id de los libros en *Books*
Granularidad	Columna
Dominio del resultado	[0,1]

Cuadro 12: M6-ConsInterRelacion-ISBNRatings

Métrica	**M7-UnicNoContradiccion-Libros**
Descripción	Mide el porcentaje de libros que no tienen datos contradictorios.
Granularidad	Tabla
Dominio del resultado	[0,1]

Cuadro 13: M7-UnicNoContradiccion-Libros

Métrica	**M8-UnicNoDuplicidad-Libros**
Descripción	Mide el porcentaje de libros que no están duplicados.
Granularidad	Tabla
Dominio del resultado	[0,1]

Cuadro 14: M8-UnicNoDuplicidad-Libros

3.1.5. Implementación de métodos de calidad

Método	**MET1_ExactSintáctica-Bool_ISBN**
Métrica	Implementa M1
Descripción	Verifica el formato estándar de ISBN.
Tipos de datos de entrada	String
Tipos de datos de salida	Boolean
Proceso	Verifica que el string de entrada cumpla con el formato ISBN-10 o ISBN-13. Siguiendo el algoritmo y utilizando el digito verificador.

Cuadro 15: MET1_ExactSintáctica-Bool_ISBN

Método	**MET1AP1_ExactSintáctica-Bool_ISBN**
Tipo	Medición
Descripción	Verifica el formato del string.
Aplicado a	Atributo «id» en tablas *Books* y *Ratings*.

Cuadro 16: MET1AP1_ExactSintáctica-Bool_ISBN

Método	**ISBN_AccuracyRatio**
Tipo	Agregación
Descripción	Calcula el porcentaje de id con sintaxis correcta de ISBN.
Agregación	Columna
Aplicado a	Atributo «id» en tablas *Books* y *Ratings*.
Fórmula	$ISBN\_AccuracyRatio = \frac{ISBN\_Validos}{ISBN\_Evaluados}$

Cuadro 17: ISBN_AccuracyRatio

Método	**MET2_ExactSintáctica-Bool_Fecha**
Métrica	Implementa la métrica M1
Descripción	Corrobora que las fechas sigan un formato estándar.
Tipos de datos de entrada	String
Tipos de datos de salida	Boolean
Proceso	Verifica si la fecha sigue el formato aaaa, aaaa-MM o aaaa-MM-dd.

Cuadro 18: MET2_ExactSintáctica-Bool_Fecha

Método	**MET2AP1_ExactSintáctica-Bool_Fecha**
Tipo	Medición
Descripción	Verifica el formato del string.
Aplicado a	Atributo «published_date» en la tabla *Books*

Cuadro 19: MET2AP1_ExactSintáctica-Bool_Fecha

Método	**MET3_ExactSemantica-Bool_ISBNDB**
Métrica	Implementa M2
Descripción	Comprueba si el id está registrado en isbndb
Tipos de datos de entrada	String
Tipos de datos de salida	Boolean
Proceso	Llamada a la *API* de isbndb.com con el id.

Cuadro 20: MET3_ExactSemantica-Bool_ISBNDB

Método	**MET3AP1_ExactSemantica-Bool_ISBNDB**
Tipo	Medición
Descripción	Verifica si está registrado el id
Aplicado a	Atributo «id» en tabla *Books* y *Ratings*.

Cuadro 21: MET3AP1_ExactSemantica-Bool_ISBNDB

Método	**ISBNSemantico_AccuracyRatio**
Tipo	Agregación
Descripción	Calcula el porcentaje de id registrados en isbndb
Agregación	Columna
Aplicado a	Atributo «id» en tablas *Books* y *Ratings*.
Fórmula	$ISBNSemantico\_AccuracyRatio = \frac{ISBNSemantico\_Valido}{ISBN\_Evaluados}$

Cuadro 22: ISBNSemantico_AccuracyRatio

Método	**MET4CompDensidad_CamposIdentificadores_Libro**
Métrica	Implementa M3
Descripción	Mide el grado de densidad de un libro, cuántos de los atributos identificadores no son nulos.
Tipos de datos de entrada	Atributo?
Tipos de datos de salida	Float
Proceso	Cuenta la cantidad de atributos no nulos del libro y los divide entre el total de atributos evaluados.

Cuadro 23: MET4CompDensidad_CamposIdentificadores_Libro

Método	**MET4AP1CompDensidad_CamposIdentificadores_Libro**
Tipo	Medición
Descripción	Mide la densidad para el caso de los atributos críticos de la tabla *Books*.
Aplicado a	Atributos «id»,«title», «author_id», «publisher», «published_date»

Cuadro 24: MET4AP1CompDensidad_CamposIdentificadores_Libro

Método	**LibrosDensidad_Promedio**
Tipo	Agregación
Descripción	Calcula el promedio de densidad de la tabla
Agregación	Tabla
Aplicado a	Tabla *Books*
Fórmula	$LibrosDensidad\_Promedio = \frac{\sum CompDensidad\_CamposIdentificadores}{CantidaddeTuplas}$

Cuadro 25: LibrosDensidad_Promedio

Método	**MET5CompCobertura_Top100**
Métrica	Implementa M4
Descripción	Válida si los libros del top 100 de los mejores libros goodreads.
Tipos de datos de entrada	String
Tipos de datos de salida	Boolean
Proceso	Para cada libro de la lista de goodreads verifica si se encuentra en la tabla *Books*. Divide el total de encontrados entre 100.

Cuadro 26: MET5CompCobertura_Top100

Método	**MET5AP1_CompCobertura_Top100**
Tipo	Medición
Descripción	Verifica la pertenencia en la tabla *Books*
Aplicado a	Atributo «title», «author_id» en tabla *Books*

Cuadro 27: MET5AP1_CompCobertura_Top100

Método	**MET6ConsIntraRelacion-Libros**
Métrica	Implementa M5
Descripción	Verifica la dependencia funcional (1)
Tipos de datos de entrada	Atributo
Tipos de datos de salida	Boolean
Proceso	Para cada conjunto de atributos si existe más de 1 id asociado se retorna 0, si no se retorna 1.

Cuadro 28: MET6ConsIntraRelacion-Libros

Método	**MET6AP1ConsIntraRelacion-Libros**
Tipo	Medición
Descripción	Verifica la pertenencia en la tabla *Books*
Aplicado a	Atributos «title», «author_id», «publisher», «published_date», «id» en *Books*

Cuadro 29: MET6AP1ConsIntraRelacion-Libros

Método	**ConsIntraRelacion-Libros_RatioDeIntegridad**
Tipo	Agregación
Descripción	Calcula el porcentaje de libros que cumplen la dependencia funcional.
Agregación	Tabla
Aplicado a	Tabla *Books*
Fórmula	$RatioDeIntegridad = \frac{\sum MET6AP1ConsIntraRelacion - Libros}{CantidaddeTuplas}$

Cuadro 30: ConsIntraRelacion-Libros_RatioDeIntegridad

Método	**MET7ConsInterRelacion-ISBNRatings**
Métrica	Implementa M6
Descripción	Válida qué porcentaje de id de *Ratings* existen en la tabla *Books*.
Tipos de datos de entrada	Atributo
Tipos de datos de salida	Float
Proceso	Toma todos los valores de id de la tabla *Ratings*, sin duplicados. Cuenta cuántos de ellos están presentes en la tabla *Books*. Divide el resultado entre la cantidad de id únicos de *Ratings*.

Cuadro 31: MET7ConsInterRelacion-ISBNRatings

Método	**MET7AP1ConsInterRelacion-ISBNRatings**
Tipo	Medición
Descripción	Verifica la pertenencia de cada id único de *Ratings* en la tabla *Books*.
Aplicado a	Atributos «id» en tabla *Ratings*, referenciado en «id» en la tabla *Books*.

Cuadro 32: MET7AP1ConsInterRelacion-ISBNRatings

3.1.6. Modelo de Calidad de Datos

El modelo de calidad de datos se presenta dividido en 4 tablas, con la columna de Factor de CD para mantener la referencia. El Cuadro 33 muestra los factores, el Cuadro 34 las métricas, el Cuadro 35 los métodos y Cuadro 36 aplicados.

Cuadro 33: Modelo de CD (1)

Problemas CD	Dimensiones CD	Factores CD
DQP1, DQP4, DQP11	**ID**: Exact_D1 **Nombre**: Exactitud **Descripción**: Evalúa si un dato es correcto. **Sugerido por**: BR1, DRQ4, DRQ5, DRQ7, DRQ9	**ID**: Exact_sint_F1 **Nombre**: Exactitud sintáctica **Descripción**: Evalúa si un dato está correctamente escrito. **Representa**: BR1, DRQ4, DRQ5, DRQ7, DRQ9
No identificados	**ID**: Exact_D1 **Nombre**: Exactitud **Descripción**: Evalúa si un dato es correcto. **Sugerido por**: BR1, DRQ4, DRQ5, DRQ7, DRQ9	**ID**: Exact_sem_F2 **Nombre**: Exactitud semántica **Descripción**: Evalúa si un dato es real. **Representa**: BR1, DRQ4
DQP2, DQP3, DQP7, DQP14, DQP15	**ID**: Comp_D2 **Nombre**: Completitud **Descripción**: Indica si el SI contiene toda la información de interés. **Sugerido por**: BR1, BR2, BR3, BR4, BR5, BR6, DRQ4, DRQ6, DRQ8	**ID**: Comp_dens_F3 **Nombre**: Densidad **Descripción**: Evalúa la cantidad de información disponible y faltante en el SI. **Representa**: BR1, BR2, BR3, BR4, DRQ4, DRQ6, DRQ8
No identificados	**ID**: Comp_D2 **Nombre**: Completitud **Descripción**: Indica si el SI contiene toda la información de interés. **Sugerido por**: BR1, BR2, BR3, BR4, BR5, BR6, DRQ4, DRQ6, DRQ8	**ID**: Comp_cob_F4 **Nombre**: Cobertura **Descripción**: Mide la porción de los datos de la realidad contenidos en el SI. **Representa**: BR5, BR6
DQP8, DQP16	**ID**: Cons_D3 **Nombre**: Consistencia **Descripción**: Captura la satisfacción de reglas semánticas definidas sobre los datos. **Sugerido por**: BR2, BR3, BR4, BR5, DRQ2, DRQ3, DRQ5, DRQ6	**ID**: Cons_dom_F5 **Nombre**: Integridad de dominio **Descripción**: Satisfacción de reglas sobre el contenido de un atributo. **Representa**: DRQ2, DRQ3
DQP5	**ID**: Cons_D3 **Nombre**: Consistencia **Descripción**: Captura la satisfacción de reglas semánticas definidas sobre los datos. **Sugerido por**: BR2, BR3, BR4, BR5, DRQ2, DRQ3, DRQ5, DRQ6	**ID**: Cons_intra_F6 **Nombre**: Integridad intra-relación **Descripción**: Satisfacción de reglas entre atributos de una misma tabla. **Representa**: BR2, BR3, BR4, BR5

Problemas CD	Dimensiones CD	Factores CD
DQP12, DQP13	**ID**: Cons_D3 **Nombre**: Consistencia **Descripción**: Captura la satisfacción de reglas semánticas definidas sobre los datos. **Sugerido por**: BR2, BR3, BR4, DRQ2, DRQ3, DRQ5, DRQ6	**ID**: Cons_inter_F7 **Nombre**: Inter-relación **Descripción**: Satisfacción de reglas entre atributos de varias tablas. **Representa**: DRQ2, DRQ3
DQP6	**ID**: Unic_D4 **Nombre**: Unicidad **Descripción**: Indica el nivel de duplicación o contradicción entre los datos. **Sugerido por**: BR1, BR5, BR6, DRQ2	**ID**: Unic_ndup_F8 **Nombre**: No-Duplicación **Descripción**: Evalúa si la misma entidad aparece repetida en forma exacta. **Representa**: BR1, BR5, BR6, DRQ2
DQP10	**ID**: Unic_D4 **Nombre**: Unicidad **Descripción**: Indica el nivel de duplicación o contradicción entre los datos. **Sugerido por**: BR1, BR5, BR6, DRQ2	**ID**: Unic_ncon_F9 **Nombre**: No-contradicción **Descripción**: Evalúa si la misma entidad aparece repetida con contradicciones. **Representa**: DRQ2

Cuadro 34: Modelo de CD (2)

Factores CD	Métricas
Exactitud sintáctica	**ID**: M1 **Nombre**: ExactSintáctica-Bool **Descripción**: Mide si un dato está escrito correctamente. **Influenciado por**: {T1, DRQ4, DRQ5, DRQ7, DRQ9} **Granularidad**: Celda **Dominio del resultado**: {0,1}
Exactitud semántica	**ID**: M2 **Nombre**: ExactSemántica-Bool **Descripción**: Mide si un dato existe en la realidad. **Influenciado por**: {OD1, OD2, DRQ4, DRQ5, DRQ7, DRQ9} **Granularidad**: Celda **Dominio del resultado**: {0,1}

Factores CD	Métricas
Densidad	**ID**: M3 **Nombre**: CompDensidad_CamposIdentificadores **Descripción**: Mide el grado de densidad de una tupla. **Influenciado por**: {DRQ4, DRQ5, DRQ7} **Granularidad**: Tupla **Dominio del resultado**: [0,1]
Cobertura	**ID**: M4 **Nombre**: CompCobertura_Top100 **Descripción**: Mide el grado de cobertura respecto a los datos de la realidad. **Influenciado por**: {BR6, DRQ5, DRQ6, DRQ9, OD1} **Granularidad**: Columna **Dominio del resultado**: [0,1]
Integridad de dominio	–
Integridad intra-relación	**ID**: M5 **Nombre**: ConsIntraRelacion-Libros **Descripción**: Mide la consistencia de libros en la tabla, donde title, author_id, publisher y publish_date corresponden a un único id. **Influenciado por**: {DRQ4, DRQ5, DRQ6, DRQ7} **Granularidad**: Tupla **Dominio del resultado**: {0,1}
Inter-relación	**ID**: M6 **Nombre**: ConsInterRelacion-ISBNRatings **Descripción**: Mide la consistencia entre las calificaciones y los libros. **Influenciado por**: {DRQ4} **Granularidad**: Columna **Dominio del resultado**: [0,1]
No-Duplicidad	**ID**: M8 **Nombre**: UnicNoDuplicidad-Libros **Descripción**: Mide el porcentaje de libros que no están duplicados. **Influenciado por**: {DRQ4, DRQ5, DRQ6, DRQ7, DRQ8, BR5} **Granularidad**: Tabla **Dominio del resultado**: [0,1]

Factores CD	Métricas
No-Contradicción	**ID**: M7 **Nombre**: UnicNoContradiccion-Libros **Descripción**: Mide el porcentaje de libros que no tienen datos contradictorios. **Influenciado por**: {DRQ4, DRQ5, DRQ6, DRQ7, DRQ8, BR5} **Granularidad**: Tabla **Dominio del resultado**: [0,1]

Cuadro 35: Modelo de CD (3)

Factores CD	Métodos
Exactitud sintáctica	**ID:** MET1_ExactSintáctica-Bool_ISBN **Implementa:** Métrica M1 **Descripción:** Verifica el formato estándar de ISBN. **Tipos de dato de entrada:** String **Tipos de dato de salida:** Boolean **Algoritmo:** Verifica que el string de entrada cumpla con el formato ISBN-10 o ISBN-13. Siguiendo el algoritmo y utilizando el digito verificador. **ID:** MET2_ExactSintáctica-Bool_Fecha **Implementa:** Métrica M1 **Descripción:** Corrobora que las fechas sigan un formato estándar. **Tipos de dato de entrada:** String **Tipos de dato de salida:** Boolean **Algoritmo:** Verifica si la fecha sigue el formato aaaa, aaaa-MM o aaaa-MM-dd.
Exactitud semántica	**ID:** MET3_ExactSemantica-Bool_ISBNDB **Implementa:** Métrica M2 **Descripción:** Comprueba si el id está registrado en isbndb. **Tipos de dato de entrada:** String **Tipos de dato de salida:** Boolean **Algoritmo:** Llamada a la *API* de isbndb.com con el id.

Factores CD	Métodos
Densidad	**ID:** MET4CompDensidad_CamposIdentificadores_Libro **Implementa:** Métrica M3 **Descripción:** Mide el grado de densidad de un libro, cuántos de los atributos identificadores no son nulos. **Tipos de dato de entrada:** Atributo? **Tipos de dato de salida:** Float **Algoritmo:** Cuenta la cantidad de atributos no nulos del libro y los divide entre el total de atributos evaluados.
Cobertura	**ID:** MET5CompCobertura_Top100 **Implementa:** Métrica M4 **Descripción:** Válida si los libros del top 100 de los mejores libros goodreads. **Tipos de dato de entrada:** String **Tipos de dato de salida:** Boolean **Algoritmo:** Para cada libro de la lista de Goodreads, verifica si todos los identificadores están en la tabla de libros. Si pertenece, devuelve 1; si no, 0.
Integridad de dominio	–
Integridad intra-relación	**ID:** MET6_ConsIntraRelacion-Libros **Implementa:** Métrica M5 **Descripción:** Verifica la dependencia funcional (1) **Tipos de dato de entrada:** Atributos **Tipos de dato de salida:** Boolean **Algoritmo:** Para cada conjunto de atributos si existe más de 1 id asociado se retorna 0, si no se retorna 1.
Inter-relación	**ID:** MET7_ConsInterRelacion-ISBNRatings **Implementa:** Métrica M6 **Descripción:** Válida qué porcentaje de id de *Ratings* existen en la tabla *Books*. **Tipos de dato de entrada:** Atributo **Tipos de dato de salida:** Float **Algoritmo:** Toma todos los valores de id de la tabla *Ratings*, sin duplicados. Cuenta cuántos de ellos están presentes en la tabla *Books*. Divide el resultado entre la cantidad de id únicos de *Ratings*.
No-Duplicidad	–
No-Contradicción	–

Cuadro 36: Modelo de CD (4)

Factores CD	Métodos aplicados
Exactitud sintáctica	**ID:** *MET1AP1_ExactSintáctica-Bool_ISBN* **Tipo:** Medición **Aplicado a:** Atributo id en tablas *Books* y *Ratings*. **ID:** *ISBN_AccuracyRatio* **Tipo:** Agregación **Aplicado a:** Atributo id en tablas *Books* y *Ratings*.
Exactitud sintáctica	**ID:** *MET2AP1_ExactSintáctica-Bool_Fecha* **Tipo:** Medición **Aplicado a:** Atributo published_date en la tabla *Books*
Exactitud semántica	**ID:** *MET3AP1_ExactSemantica-Bool_ISBNDB* **Tipo:** Medición **Aplicado a:** Atributo id en tabla *Books* y *Ratings*. **ID:** *ISBNSemantico_AccuracyRatio* **Tipo:** Agregación **Aplicado a:** Atributo id en tabla *Books* y *Ratings*.
Densidad	**ID:** *MET4AP1CompDensidad_CamposIdentificadores_Libro* **Tipo:** Medición **Aplicado a:** Atributos id, title, author_id, publisher, published_date. **ID:** *LibrosDensidad_Promedio* **Tipo:** Agregación **Aplicado a:** Tabla *Books*
Cobertura	**ID:** *MET5AP1_CompCobertura_Top100* **Tipo:** Medición **Aplicado a:** Atributo title, author_id en tabla *Books*. **ID:** *Porcentaje_CompCobertura* **Tipo:** Agregación **Aplicado a:** Atributo title, author_id en tabla *Books*.
Dominio	–

Factores CD	Métodos aplicados
Intra-relación	**ID:** *MET6AP1ConsIntraRelacion-Libros* **Tipo:** Medición **Aplicado a:** Atributos id, tilte, author_id, publisher, published_date. **ID:** *ConsIntraRelacion-Libros_RatioDeIntegridad* **Tipo:** Agregación **Aplicado a:** Tabla *Books*
Inter-relación	**ID:** *MET7AP1ConsInterRelacion-ISBNRatings* **Tipo:** Medición **Aplicado a:** Atributos id en tabla *Ratings*, referenciado en id en la tabla *Books*.
No-Duplicidad	–
No-Contradicción	–

3.1.7. Salidas

Las salidas de esta actividad constan de la priorizacion de problemas de calidad descritos en la Subsubsección 3.1.2 y el modelo de CD de la Subsubsección 3.1.6

3.2. Stage 5 DQ Measurement

Del desarrollo de esta etapa se espera obtener la especificación de la BD de metadatos de CD que permita almacenar los resultados de las métricas correctamente, así como un reporte de medición de la CD con los resultados de ejecutar estas ademas de las herramientas utilizadas para hacerlo. Por ultimo, el modelo de contexto se debe actualizar para incluir los metadatos de CD.

3.2.1. Entradas

Las entradas de esta etapa consisten en el reporte de problemas de CD priorizados y del modelo de CD contextual. Ambos artefactos fueron definidos en la etapa anterior.

3.2.2. Diseño de la base de datos DQ metadata

La base de datos de metadatos de calidad (BDDQ) se diseñó para almacenar los resultados de las métricas implementadas en la etapa anterior así como aquellas que se pudieran implementar en el futuro. La Figura 22 muestra el diagrama entidad relación generado por PostgreSQL, el manejador utilizado.

Se destinan 3 tablas *ResultTable*, *ResultColumn* y *ResultCellRow* para las métricas de granularidad tabla, columna y celda/tupla respectivamente. Se utiliza la tabla *Executions* para registrar la fecha y hora de cada ejecución junto a un identificador auto generado, de esta manera se permiten múltiples ejecuciones de un mismo método aplicado en diferentes instantes de tiempo. Lo que además permite analizar la evolución y el monitoreo de la calidad de los datos a lo largo del tiempo.

Por otra parte, las tablas *Dimensions, Factors, Metrics, Methods* y *ApMethods* se utilizan para registrar la información de las métricas (granularidad, dominio, etc) y la relación con los factores, dimensiones así como métodos y métodos aplicados. Esto permite recuperar fácilmente los resultados de las ejecuciones.

Por último la tabla *Users* registra a los usuarios y sus roles para ser utilizado en la tabla *UserEvaluations*, esta última permite asignarle un umbral a cada relación entre usuario y método aplicado. De esta manera es fácil reconstruir las ejecuciones y conocer las evaluaciones según las necesidades de cada usuario.

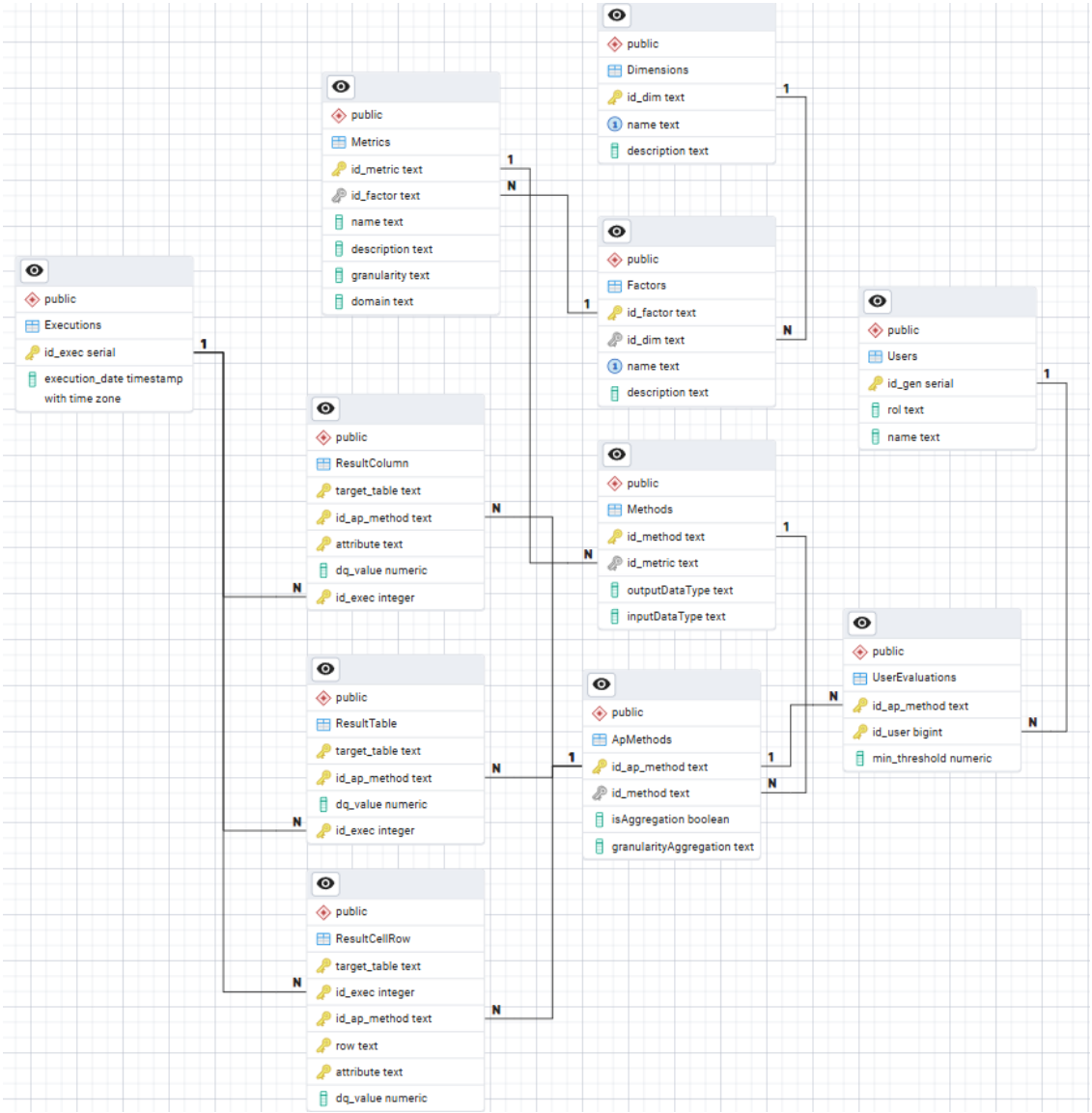


Figura 22: Diagrama ER BDDQ

3.2.3. Ejecución de métricas y almacenamiento de resultados

Para ejecutar las métricas de calidad se implementaron en Python, este lenguaje cuenta con varias bibliotecas idóneas para analizar datos. Se implementó un programa principal donde este se conecta con BDDQ y con la base de datos de NL, así mediante consultas SQL los datos se procesan y almacenan el resultado de los métodos aplicados en BDDQ. Se utilizaron las bibliotecas *Psycopg* para realizar la conexión y las consultas SQL y *Matplotlib* para graficar los resultados. Por último, se utilizo ChatGPT para extraer la lista de los 100 libros de la web de Goodreads.

De las metricas definidas en la etapa anterior se ejecutaron solo aquellas implementadas, es decir, M1, M2, M3, M4, M5 y M6.

Métrica: M1-ExactSintáctica -Bool

La métrica **M1** mide la exactitud sintáctica de un atributo, fue aplicada a los atributos id y fecha de la tabla *Books* y al atributo id de la tabla *Ratings*. La Figura 23 presenta un ejemplo de la implementación de esta métrica, así como del método aplicado correspondiente.

```python
def isbn_valid(isbn):
    suma = 0
    try:
        if len(isbn) == 10:
            for i in range(9):
                suma += int(isbn[i]) *
                    (i+1)
            suma = suma % 11
            if isbn[9] == 'X':
                return suma == 10
            else:
                return suma ==
                    int(isbn[9])
        elif len(isbn) == 13:
            for i in range(12):
                if i % 2 == 0:
                    suma += int(isbn[i])
                else:
                    suma += int(isbn[i])
                        * 3
            return ((suma +
                int(isbn[12])) % 10 == 0)
    except:
        return False
    return False
```

```python
def met1ap1_exact_sintactica_bool_isbn(
    conn_db, conn_dbdq, table,
    id_row_col, isbn_column, id_exec):
    rows = []
    with conn_db.cursor() as cur:
        cur.execute(
            f"SELECT {id_row_col},
                {isbn_column} FROM
                \"{table}\"")
        rows = cur.fetchall()
        with conn_dbdq.cursor() as cur2:
            for row in rows:
                id_tuple, isbn_tuple = row
                cur2.execute(
"INSERT INTO ResultCellRow (...) VALUES
    (%s, %s, %s, %s, %s, %s)",
    (
    table,
    id_exec,
    'MET1AP1_ExactSintactica-Bool_ISBN',
    id_tuple, isbn_column,
    int(isbn_valid(isbn_tuple))
    )
                )
```

Figura 23: Código Python para validación y almacenamiento de ISBN

La Figura 24 muestra la evaluación de *published_date* en *Books*, se toma como valido cualquier variante entre aaaa, aaaa-MM o aaaa-MM-dd. 447.350 tuplas cumplen con la condición, esto representa al 93 % de las tuplas de *Books*.

El atributo identificador de un libro es su ISBN, como esta descrito en Figura 23 se valida mediante la cantidad de dígitos y el dígito verificador que se genera mediante un algoritmo.

Existen ISBN de 10 dígitos y de 13, el sistema de información de NL podría contar con ambos. La Figura 25 muestra la evaluación de la métrica, donde 80.325 tuplas contienen un ISBN sintácticamente incorrecto. Es decir, poseen caracteres no numéricos o no cumplen con el algoritmo del dígito verificador.

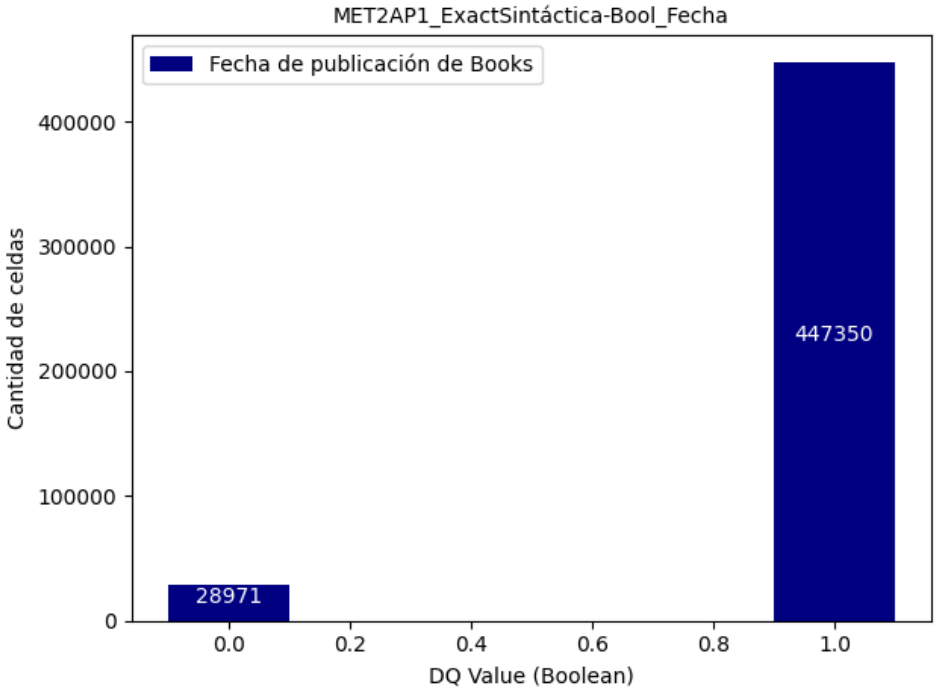


Figura 24: Métrica 1 evaluando exactitud sintáctica de Fecha en *Books*

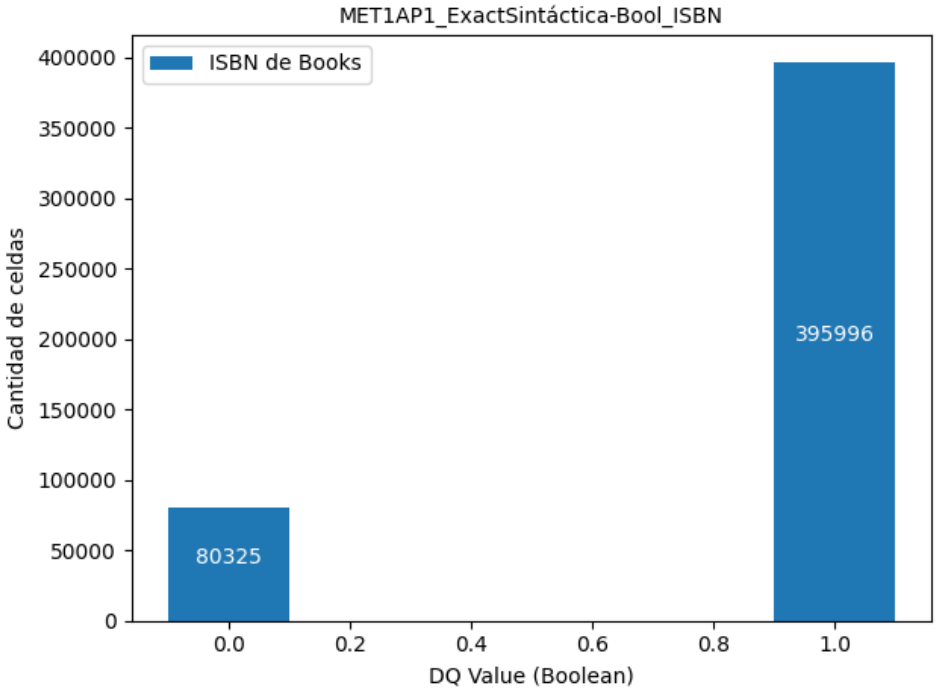


Figura 25: Métrica 1 evaluando exactitud sintáctica de ISBN en *Books*

El atributo *ISBN* es una clave foránea en la tabla *Ratings* y forma parte de su clave primaria. En esta tabla, la métrica fue aplicada a un total de 3.493.813 tuplas, identificando 1.572.336 con un ISBN incorrecto, como se muestra en la Figura 26.

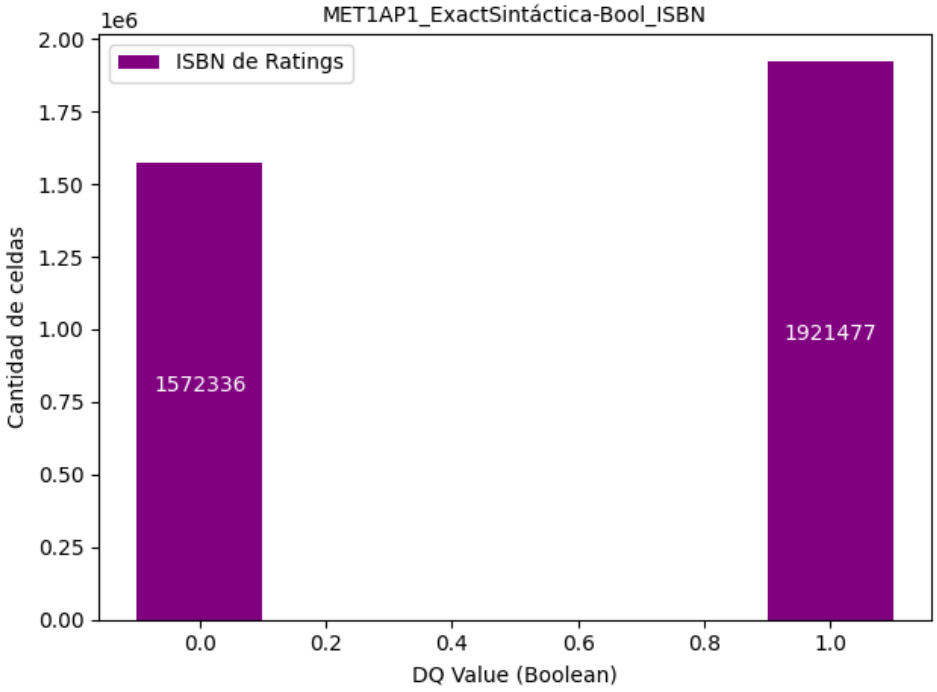


Figura 26: Métrica 1 evaluando exactitud sintáctica de ISBN en *Ratings*

La agregación aplicada a la métrica para el atributo *ISBN*, representada en la Figura 27, muestra que el 88,14 % de las tuplas en *Books* presentan un ISBN sintácticamente correcto, mientras que en *Ratings* el porcentaje (55,00 %) es considerablemente menor.

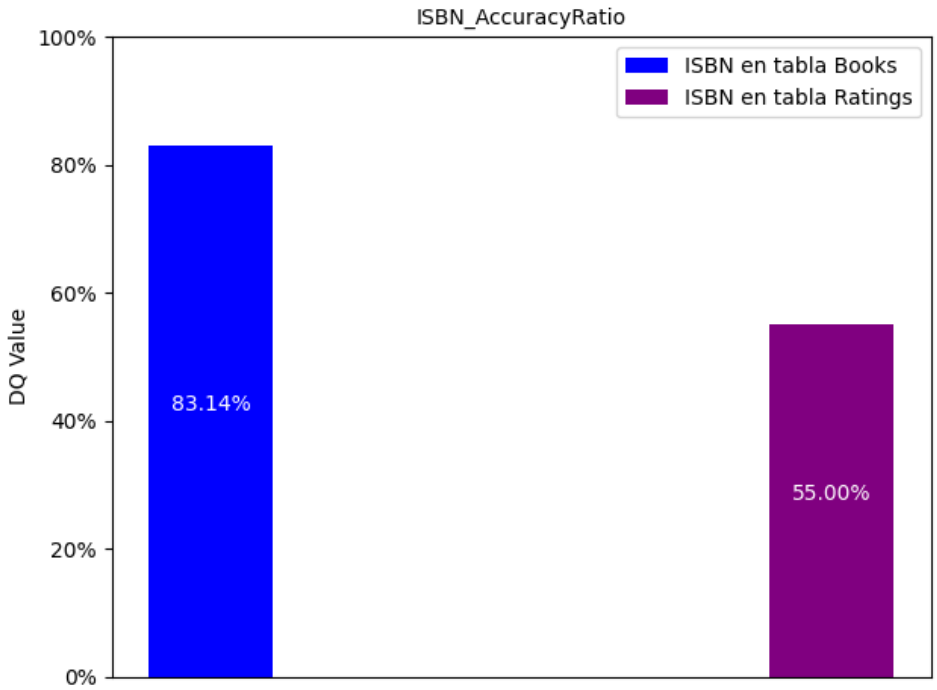


Figura 27: Agregación de M1

Estos resultados son esperables, ya que el atributo *ISBN* en *Ratings* hace referencia al de *Books*, pero no constituye una clave primaria por sí solo, dado que múltiples usuarios pueden registrar reseñas sobre un mismo libro. Además, considerando que el volumen de datos en *Ratings* es mucho mayor, este problema tiende a propagarse con mayor facilidad.

Métrica: M2-ExactSemántica-Bool

La métrica m2 mide la exactitud semántica de un atributo, tiene granularidad booleana y el método definido busca el atributo *ISBN* en la base de datos isbndb.com mediante su *API*. Este método es aplicado al atributo *id* de la tabla *Books* y a su homónimo en la tabla *Ratings*.

La base de datos isbndb.com posee un amplio catalogo de libros registrados, sin embargo se requiere de una licencia paga para utilizar su *API*. Su plan mas económico $7.50 USD al mes restringe las llamadas a 2.000 por día. La suma de tuplas de ambas tablas es de 3.970.134 lo que lo hace una operación costosa. Por lo tanto, para esta métrica se utilizo la siguiente estrategia:

1. Aplicar la métrica solo sobre *Books*.
2. Aplicar el método sobre una muestra pequeña de datos.
3. Utilizar *Google Books API*, considerando:
    - Plan gratuito.
    - 1.000 llamadas por día.
    - Catálogo de libros reducido.

Para respetar la independencia de los factores, la muestra se tomó sobre el total de las tuplas. No obstante, se podría aplicar únicamente sobre aquellas con un ISBN sintácticamente correcto, lo que habría hecho el proceso más eficiente.

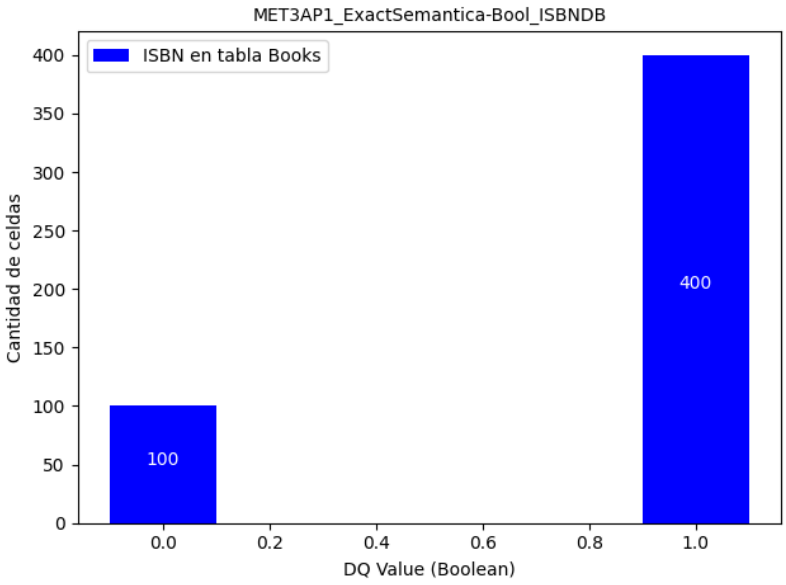


Figura 28: Métrica 2 aplicada a muestra aletoria de 500 tuplas de *Books*

De los 500 ISBN evaluados, 400 son semánticamente correctos, lo que representa el 80 % de la muestra. La Figura 29 muestra, además, que el 0,80 % de los ISBN incorrectos son sintácticamente válidos (8 libros de los 500). Esto puede deberse a dos motivos: en primer lugar, el ISBN no está asociado a ningún libro real, por lo tanto, el resultado es negativo; sin embargo, también podría tratarse de un falso negativo si la fuente de datos consultada no tiene registrado ese ISBN.

Aun así, para los fines de evaluación de la métrica estos datos son incorrectos y serán tratados como tales.

Métrica: M3-CompDensidad_CamposIdentificadores

Esta métrica de granularidad tupla, evalua la densidad de los atributos identificadores. Fue aplicada a las tuplas de la tabla *Books*.

Un libro es identificado por su id (ISBN) pero también mediante su titulo, autores, fecha de publicación y editorial. Si todos sus atributos están presentes la evaluación es 1 y 0 para el caso donde todos ellos sean nulos.

La Figura 30 muestra que no existen tuplas con todos los atributos nulos y que 399.964 poseen todos ellos. El promedio de densidad es del 95 % y solo el 16 % de las tuplas no poseen todos los atributos.

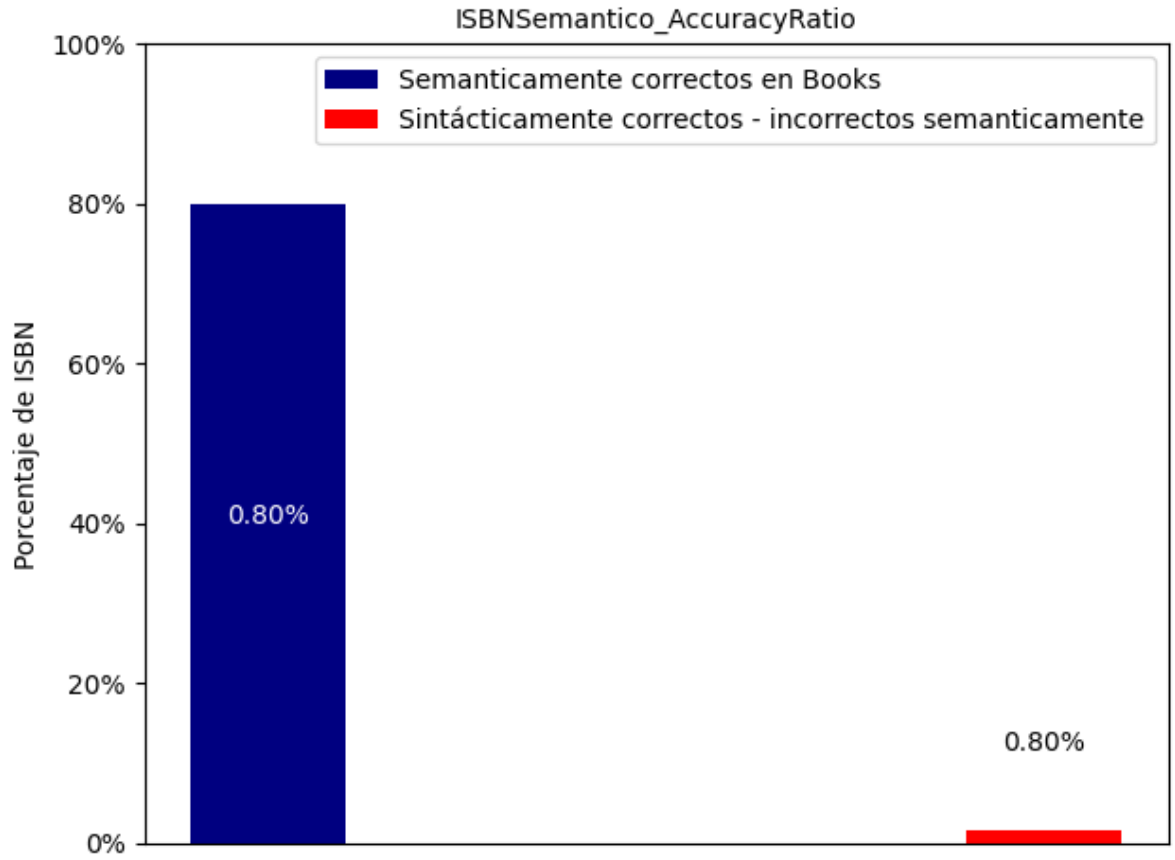


Figura 29: Agregación de Métrica 2 y porcentaje de incorrectos que cumplen el formato ISBN

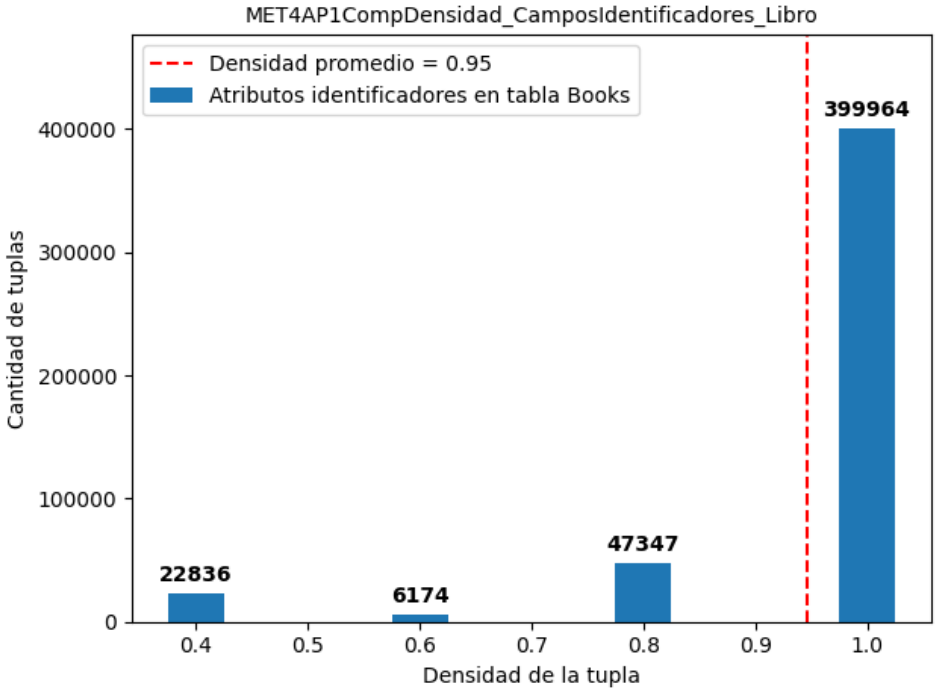


Figura 30: M3-CompDensidad_CamposIdentificadores

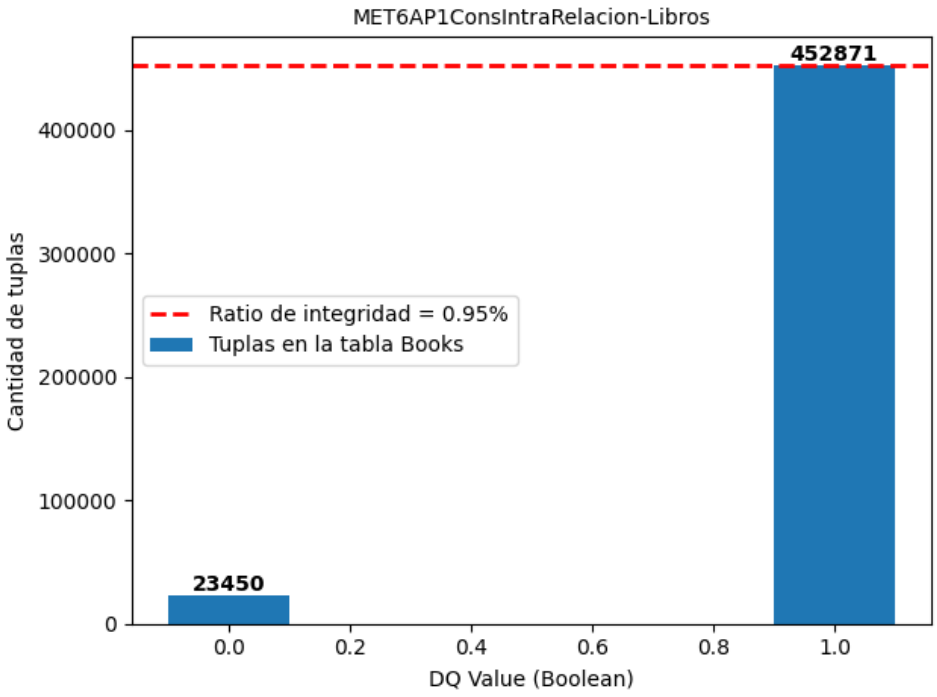


Figura 31: $\{title, author, publish\_date, publisher\} \rightarrow ISBN$

Métrica: M4-CompCobertura_Top100

Esta métrica fue evaluada en la tabla *Books*. El Cuadro 37 muestra la ejecución, donde 78 % de los libros del top 100 pertenecen a *Books*, es decir 78 libros.

La comparación se realiza utilizando el titulo y autor, errores en estos campos pueden perjudicar la evaluación.

Método aplicado	Correctos
MET5AP1_CompCobertura_Top100	0,78

Cuadro 37: Cobertura en Top 100 de GoodReads

Métrica: M5-ConsIntraRelacion-Libros

En la tabla *Books* se debe cumplir la dependencia funcional 1 definida en . La Figura 31 muestra el resultado de la ejecución. Donde hay 23.450 tuplas que la violan. Sin embargo, el 95 % la cumple.

Métrica: M6-ConsInterRelacion-ISBNRatings

Cualquier tupla de la tabla *Ratings* deberia tener un id (ISBN) existente en la tabla *Books*. La métrica evalua que esto es cierto para el 91 % de las tuplas de *Ratings*.

Método aplicado	Correctos
MET7AP1_ConsInterRelacion-ISBNRatings	0,91

Cuadro 38: Consistencia inter relación *Ratings* y *Books*

3.2.4. Actualización del modelo de contexto

Las ejecuciones de las métricas generan metadatos de calidad que son componentes en el modelo de contexto. *Ej_MX* denota la ejecución de la métrica X.

Comp. de CTX.	Usuarios			
	Todos	U1	U2	U3
Dominio de aplicación	DA			
Tareas		T1	T2	T3
Reglas de negocio	BR1, BR2, BR3, BR4, BR5, BR6			
Req. Sistema	SR			
Req. CD		DRQ4, DRQ7	DQR1, DQR2, DQR3, DRQ8, DRQ9	DQR4, DQR5, DRQ6, DRQ7, DRQ9
Filtrado de datos	DF1, DF2, DF3			
Metadatos	M			
Metadatos de CD	Ej_M1, Ej_M2	Ej_M3, Ej_M5, Ej_M6	EJ_M4	Ej_M3, Ej_M4, Ej_M6
Otros datos	OD			

Cuadro 39: Modelo de contexto con metadatos de calidad

3.2.5. Salidas

Las salidas de esta etapa consisten en la especificación de la BD de metadatos de calidad descrita en Subsubsección 3.2.2, el reporte de medición de la CD conformado por el análisis de la Subsubsección 3.2.3 y la actualización del modelo de contexto para incluir los metadatos de CD en la Subsubsección 3.2.4.

3.3. Stage 6 DQ Assessment

El objetivo de esta etapa es comparar los resultados de las métricas con los requerimientos de los usuarios y cuantificar su completitud, teniendo en consideración el modelo de contexto.

3.3.1. Entradas

Las entradas para esta etapa consisten en las salidas de la etapa anterior, estas están descritas en la Subsubsección 3.2.5.

3.3.2. Definición de enfoques de evaluación

Se definen los umbrales cualitativos para todas las métricas en la el Cuadro 40, sin embargo algunos usuarios y componentes de contexto podrían exigir una evaluacion mas estricta, para ello se define los umbrales del Cuadro 41.

Umbral	Valor cualitativo
0 % - 30 %	Mala
31 % - 60 %	Buena
61 % - 90 %	Muy buena
91 % - 100 %	Excelente

Cuadro 40: Umbrales generales

Umbral	Valor cualitativo
0 % - 60 %	Mala
61 % - 79 %	Buena
80 % - 94 %	Muy buena
95 % - 100 %	Excelente

Cuadro 41: Umbrales estrictos

El Cuadro 42 muestra que umbral se utilizara para cada tipo de usuario en cada métrica ejecutada.

Métrica	Tabla evaluada	Atributo evaluado	Usuario	Umbral	Comp. Ctx.
M1	Books	ISBN	U1	Estricto	T1, BR1, DF, DRQ4
M1	Books	ISBN	U2	General	BR1, T2
M1	Books	ISBN	U3	Estricto	T3, BR1, DRQ4
M1	Ratings	ISBN	U1	Estricto	T1, BR1, DF
M1	Ratings	ISBN	U3	Estricto	T3, BR1
M1	Books	Fecha	U1	Estricto	DF, DRQ7
M1	Books	Fecha	U3	Estricto	T3, DRQ7
M2	Books	ISBN	U1	Estricto	BR5, BR6, DRQ4
M2	Books	ISBN	U2	Estricto	T2 BR5, BR6
M2	Books	ISBN	U3	Estricto	T3, BR5, BR6, DRQ4
M3	Books	Tupla	U1	Estricta	T1, DF, BR1, BR2, BR3, BR4
M3	Books	Tupla	U3	General	T3, BR1, BR2, BR3, BR4
M4	Books	Tabla	U2	General	T2, BR6
M4	Books	Tabla	U3	General	BR6
M5	Books	Tupla	U1	Estricto	T1, BR1
M6	Books	Columna	U1	General	T1, DRQ4
M6	Books	Columna	U3	Estricto	T3, DRQ4

Cuadro 42: Asignación de umbrales según usuarios U1: Administrador, U2: Publicista y U3: Analista

3.3.3. Ejecución de los enfoques de evaluación y almacenaje de los resultados

Métrica	Tabla	Atributo	Resultado (%)	U1	U2	U3
M1	Books	ISBN	83	Muy buena	Muy buena	Muy buena
M1	Books	Fecha	93	Muy buena	-	Muy buena
M1	Ratings	ISBN	55	Mala	-	Mala
M2	Books	ISBN	80	Muy buena	Muy buena	Muy buena
M3	Books	Tupla	95	Excelente	-	Excelente
M4	Books	Tabla	78	-	Muy buena	Muy buena
M5	Books	Tupla	95	Excelente	-	-
M6	Books	Columna	91	Excelente	-	Muy buena

Cuadro 43: Ejecución de la evaluación según los umbrales definidos

Los resultados de la ejecucion (Cuadro 43) son almacenados en BDDQ.

3.4. Salidas

Esta etapa tiene como salida la Subsubsección 3.3.2 y la Subsubsección 3.3.3, donde ambos conforman el reporte de evaluación de CD.

4. DQ Improvement phase

La 3 fase esta enfocada en la mejora de los datos, está no sera ejecutada en este proyecto. sin embargo, se analizara los reportes de CD y todos los artefactos generados hasta ahora para sugerir un breve plan de mejora.

La fase consta de 3 etapas, la primera, la etapa 7, recibe como entrada el reporte de evaluación de CD, elaborado en la fase anterior. Se analizan las causas que ocasionan los problemas de calidad y se genera como salida el reporte con los problemas de CD seleccionados y priorizados de acuerdo a las causas.

La próxima etapa, la etapa 8, recibe como entrada la salida de la etapa anterior y genera un reporte de análisis de costos, así como un plan de mejora de la CD.

Finalmente, la etapa 9 recibe como entrada las salidas de la etapa 8 y genera como salidas un reporte de ejecución del plan de mejora de CD y los data at hand mejorados.

4.1. Stage 7 resumida

Los resultados de algunas métricas resultan mejor de lo esperado por las estimaciones iniciales. Los mayores problemas ocurren en la tabla *Ratings*, donde la métrica es Mala para ambos usuarios interesados. Esto es esperable dado el volumen y la relación con *Books*. El origen de este problema se centra en la integración de los datasets de ambas librerías, donde el datasets de ratings significativamente mayor utiliza un identificador que en su mayoría no cumple con los requisitos para ser un ISBN.

Dada las reglas de negocio, el dominio y las tareas que los usuarios desempeñan, el trabajo sobre la tabla *Ratings* o aquello que pueda mejorar sus métricas es lo prioritario. Mejorar la calidad de estos datos mejorara recomendaciones de libros, análisis de los usuarios sobre ellos, publicidad más efectiva, etc. Para que una calificación sea útil es necesario que este asociado al libro correcto.

4.2. Stage 8 resumida

Para mejorar la integridad de la relación entre *Ratings* y *Books* es necesario mejorar la calidad del ISBN. Mejorando este atributo se puede relacionar correctamente libros con calificaciones y por ende usuarios con libros. En el Cuadro 44 se analizan acciones que pueden mejorar y su costo asociado.

Acción	**Costo**	**Beneficio potencial**	**Justificación**
Definición de claves primarias y foráneas	Bajo	Medio	Poco impacto en los datos actuales, alto impacto en los datos futuros.
Corrección de sintaxis de ISBN	bajo	bajo	Aquellos ISBN que tengan el formato adecuado y sean recuperables mediante un algoritmo podrían ayudar a recuperar información útil manualmente
Consulta en base de datos externa	Muy alto	Desconocido	Muy útil para corregir semántica de los atributos identificadores, sin embargo es una operación costosa y no se evaluó la sintaxis o semántica de todos ellos.

Cuadro 44: Acciones de mejora, costo y beneficio estimado.

Dado el análisis de costo se sugiere el siguiente plan de mejora:

1. Definir claves primarias y foráneas.

2. Intentar corregir sintaxis de aquellos ISBN que pueden ser recuperados con su digito verificador.

Considerando las limitaciones y los costos se propone eventualmente profundizar en el análisis de los datos y tomar la siguiente acción: aplicar la consulta de base de datos externa en lotes de libros que tengan ISBN correcto pero baja densidad en el resto de atributos identificadores, así como aquellos que el autor y el titulo son desconocidos o su sintaxis es incorrecta. Priorizar las tuplas de *Books* que tienen mayor cantidad de tuplas asociadas en *Ratings*, pues es probable que estos libros sean mas populares y tengan mayor impacto en el negocio.

4.3. Stage 9 resumida

Para ejecutar el plan de mejora se realizara las acciones 1 y 2. La acción 3 se realizara utilizando la API de *Google Books*, por ser gratuita, conociendo sus limitaciones. En el futuro se evaluara utilzar la API de isbndb para mejorar el impacto de esta acción.

5. Conclusiones

Durante el desarrollo de la fase 1[4] se elaboro un modelo de contexto acertado, priorizando los requerimientos de los usuarios y contemplando las reglas de negocio y dominio. La estimación de calidad generada durante esta fase era mala. Dado el origen de los datos y las características de la integración era esperable encontrar contradicciones, duplicaciones y problemas graves para identificar libros y usuarios.

Durante el transcurso de la fase 2 se priorizo abordar aquellos problemas relacionados a la identificación de los libros, ya que estos son la base de muchos de los problemas restantes y en caso de ser correctos es sencillo recuperar la información sobre los atributos restantes. Como fue estimado, la tabla *Ratings* tiene una calidad mala para los atributos evaluados. Sin embargo, la tabla *Books* tiene mayor calidad de la prevista para las métricas definidas. Dado el alcance del proyecto, no se evaluó la calidad de la relación entre usuarios y calificaciones así como otros atributos interesantes para el negocio.

La priorización y el plan descrito en la fase 3, pretenden aprovechar las métricas evaluadas y maximizar el impacto que tengan las correcciones aplicadas. Sin embargo, es necesario seguir evaluando la calidad del resto de datos para lograr un plan efectivo.

6. Reflexión sobre el proyecto y la metodología

Durante el desarrollo del proyecto nos enfrentamos a desafíos variados e interesantes. Pudimos comprender el impacto real que tiene un atributo mal definido o nulo cuando se trabaja

[4] Las siguientes conclusiones son de la metodología en general e integran lo elaborado en las tareas anteriores desde la perspectiva actual.

con millones de datos. Buscar estrategias para resolver estos problemas y explorar las herramientas disponibles resultó muy interesante, especialmente al aplicarlas en un contexto real".

La metodología CaDQM nos resultó fácil de utilizar. En general, todas sus fases, etapas y actividades tienen sentido y aplicación directa. Algunos artefactos generados por la metodología parecían redundantes o sin aplicación clara, pero la mayoría fue ganando protagonismo en distintos momentos del proyecto. Otros, sin duda, cobrarían más sentido en un contexto organizacional real que en un proyecto académico.

Los problemas abordados en cada tarea estuvieron bien nivelados. Sin embargo, los datasets provistos y la necesidad de integrarlos antes de comenzar el desarrollo del proyecto representaron un desafío en sí mismo que no forma parte explícita de CaDQM. Esto generó que, en las primeras etapas, los conceptos trabajados fueran más difíciles de asimilar debido a la carga extra de integración.

7. Anexos

7.1. Herramientas

Las principales herramientas utilizadas fueron:

- Kettle (Pentaho): Integración de datasets
- DataCleaner: Para realizar el data analysis
- PosgreSQL: Para manejar las bases de datos
- Python análisis de datos, con las bibliotecas psycopg3 para consultas SQL y matplotlib para graficar datos
- Google Books API: consultas para los ISBN

Parte del código generado y estructuras de las bases de datos se encuentran disponible en el Repositorio de GitHub.

7.2. Integración

Para generar la tabla de autores e integrar los libros se utilizo Kettle. A partir de las fuentes de datos con información de libros, se extrajo la columna de autores y se procesaron sus valores para obtener una lista de autores únicos, evitando duplicados. En particular, en el archivo books_data.csv, los nombres de los autores estaban entre comillas y corchetes, por lo que se normalizó el formato eliminando estos caracteres. Además, cuando había más de un autor en una misma fila, se separaron en distintas filas. Finalmente, a cada autor se le asignó un identificador único y la información fue insertada en la base de datos PostgreSQL. La Figura 32 muestra las transformaciones realizadas para generar este proceso.

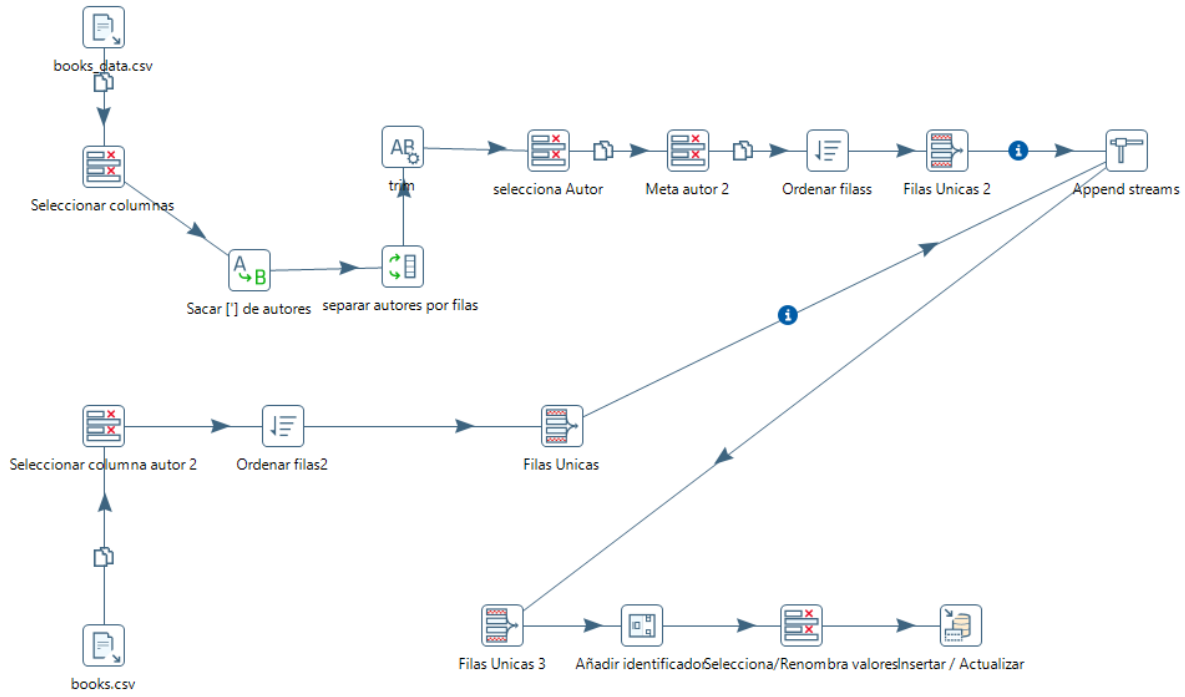


Figura 32: Transformación de autores

En la Figura 33 se muestra la estrategia utilizada para integrar la información de libros. A partir del archivo books_ratings, se obtuvo el campo ISBN y se lo vinculó con el archivo books_data mediante una unión por el campo title. Además, se aplicó un proceso similar al realizado con los autores, de forma que se mantenga la correspondencia con las filas que se generarán para la tabla de autores.

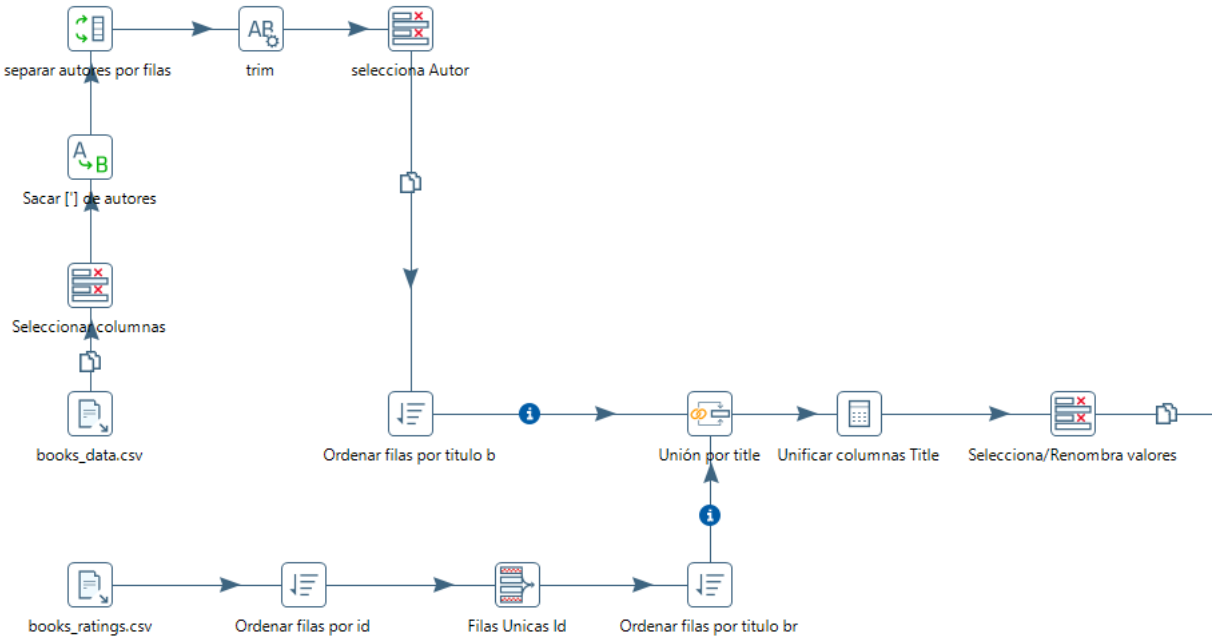


Figura 33: Transformación de autores

La Figura 34 muestra la siguiente etapa del proceso de integración, donde se realiza la unión con la tabla books.csv utilizando el atributo ISBN como clave principal para identificar los libros. Como resultado de esta unión, se generan columnas duplicadas que contienen la misma información. Para resolverlo, se unificaron dichas columnas, conservando los valores no nulos o asignando NULL en caso de que ambas estuvieran vacías. Finalmente, a partir de la tabla de autores ya creada, se asocia el identificador correspondiente a cada instancia, y la información resultante se inserta en la base de datos.

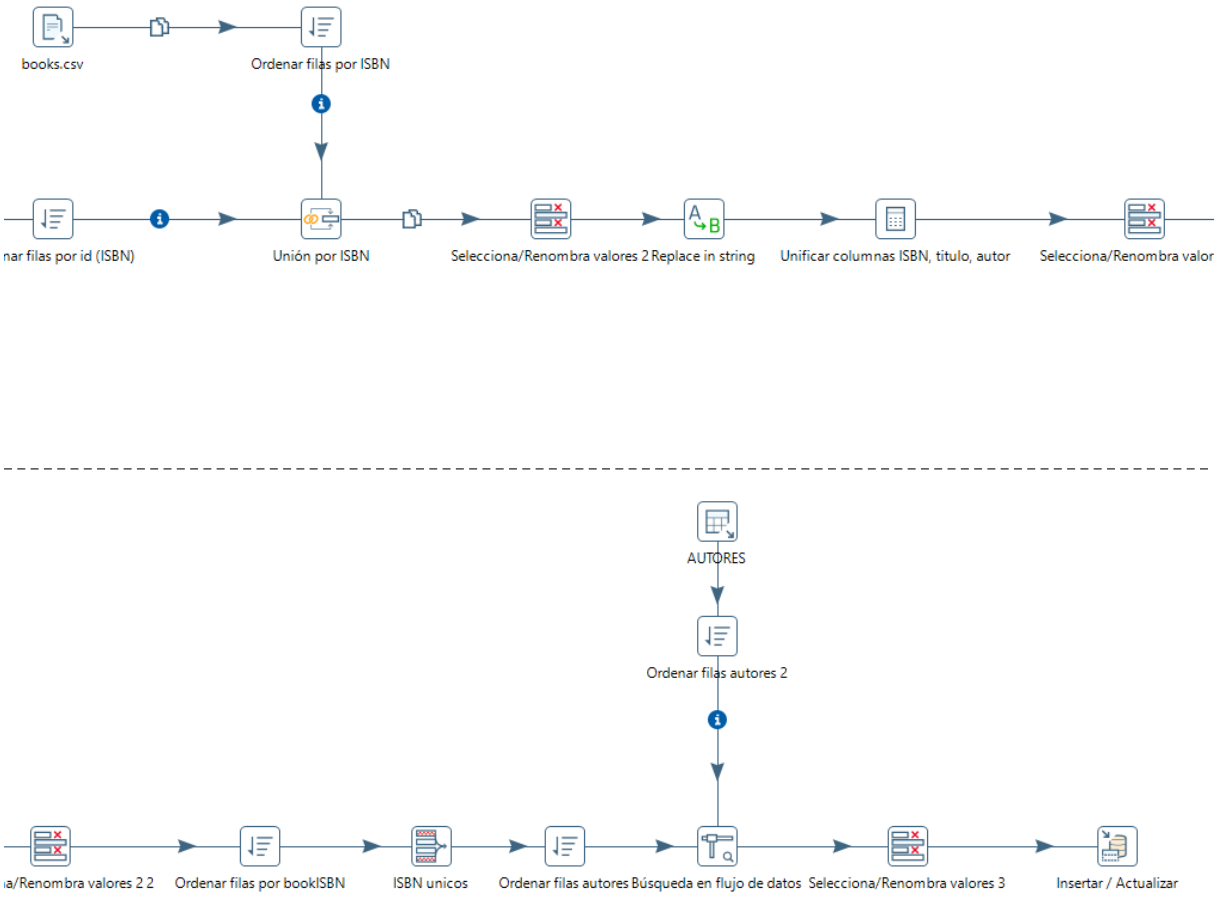


Figura 34: Transformación de autores