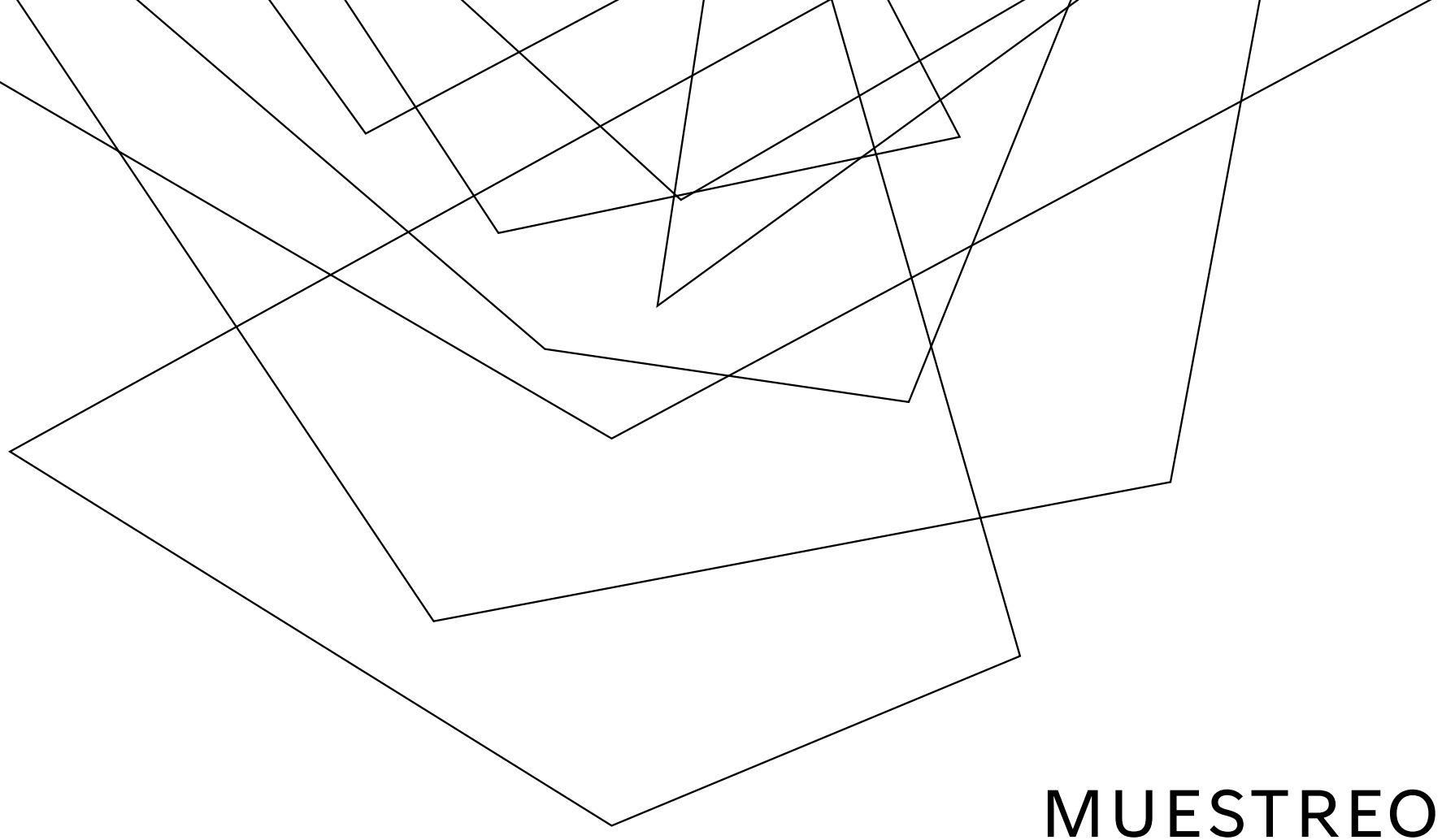
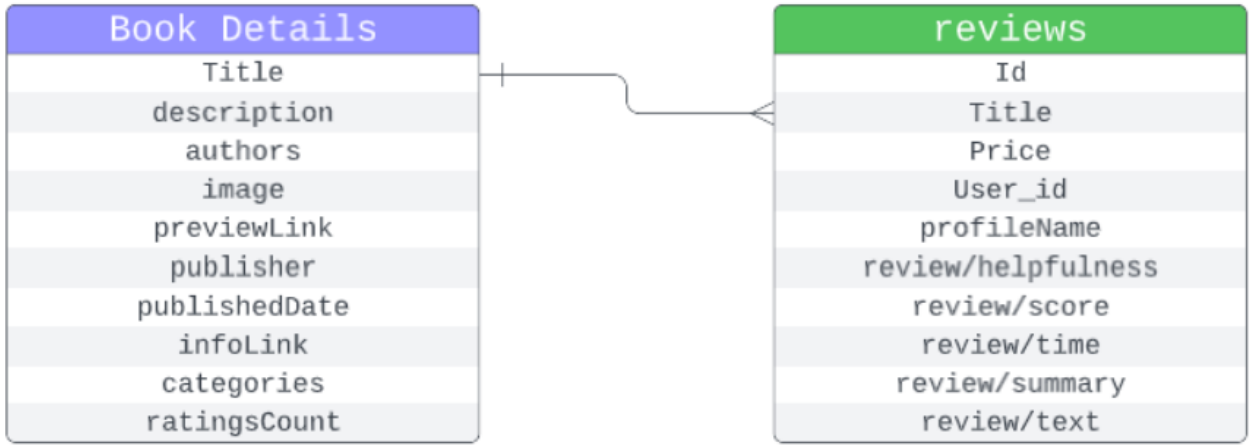


MUESTREO DE DATOS MASIVOS

Emmanuel Ramos

REVIEWS EN LIBROS DE AMAZON

El datasetobtenido de kaggle contiene información de reviews de 3M de libros, este se compone de los reviews actuales de cada uno de los libros y su metadata de cada uno de los libros, dada la cantidad de información se hara uso de la Plataforma databricks la cual haremos uso de un cluster de Spark para el procesamiento de la información. El siguiente sera el esquema de las tablas actuales y su relacion.



Los pasos para procesar la información fue subir los archivos a un blobcontainer de Azure los cuales vamos a leer y aplicaremos el esquema correspondiente a cada archivo, después de escribirán los datos del dataframe en formato parquet y los dejaremos en una tabla.

```
%run "../includes/configuration"
```

```
Command took 0.17 seconds -- by jesus.ramosdv@uanl.edu.mx at 6/29/2023, 10:33:01 AM on JESUS DAVILA's Cluster
```

Cmd 4

Python

```
df_ratings = spark.read.schema(reviews_schema).option("header", "true").option("multiLine", True).csv(f"{raw_folder_path}/Books_rating.csv")

```

```
df_ratings:  pyspark.sql.dataframe.DataFrame = [Id: string, Title: string ... 7 more fields]
Command took 2.54 seconds -- by jesus.ramosdv@uanl.edu.mx at 6/29/2023, 10:33:07 AM on JESUS DAVILA's Cluster
```

Cmd 5

```
df_books = spark.read.schema(book_details_schema).option("header", "true").option("multiLine", True).csv(f"{raw_folder_path}/books_data.csv")
```

```
df_books:  pyspark.sql.dataframe.DataFrame = [Title: string, description: string ... 8 more fields]
Command took 0.21 seconds -- by jesus.ramosdv@uanl.edu.mx at 6/29/2023, 10:33:10 AM on JESUS DAVILA's Cluster
```

```
df_ratings.write.mode("overwrite").format("parquet").saveAsTable("demo.book_reviews")
```

```
(1) Spark Jobs
  Job 0  View (Stages: 1/1)
Command took 2.30 minutes -- by jesus.ramosdv@uanl.edu.mx at 6/29/2023, 10:33:28 AM on JESUS DAVILA's Cluster
```

Cmd 11

```
df_books.write.mode("overwrite").format("parquet").saveAsTable("demo.book_details")
```

```
(1) Spark Jobs
Command took 10.10 seconds -- by jesus.ramosdv@uanl.edu.mx at 6/29/2023, 10:33:50 AM on JESUS DAVILA's Cluster
```

SAMPLING SIMPLE Y SESGADO

Para esta parte sesgaremos el sampling al solo usar libros que tengan un rating mayor a 3, **siendo la calificación más alta 5.** después de realizado este filtro se usará un método para realizar un sampling de manera simple tratando de solo obtener un 40%.

```
1  df_results = spark.sql('select d.Title, d.description, d.categories, d.ratingsCount , r.`review/summary` , r.`review/text`, r.`review/helpfulness`  from  demo.
   book_reviews r inner join demo.book_details d on r.Title = d.Title where d.ratingsCount > 3')
2  display(df_results)
```

Python

▸ (5) Spark Jobs

▸ df_results: pyspark.sql.dataframe.DataFrame = [Title: string, description: string ... 5 more fields]

Cmd 13

```
1  #Perform simple sample
2
3  print(df_results.count())
4  df_results_simple_sampling = df_results.sample(0.4).toPandas()
5
```

▸ (9) Spark Jobs

12967

2

Command took 9.08 seconds -- by jesus.ramosdv@uanl.edu.mx at 6/29/2023, 11:46:11 AM on JESUS DAVILA's Cluster

SAMPLING SELECTIVOS Y ESTRATIFICADO

Para este proceso de sampling usaremos el mismo query para obtener los datos a diferencia de que no usaremos ningún filtro de rating como en el proceso anterior, para el sampling estratificado tendremos que darle a cada estrato un porcentaje, en este caso será equivalente del 40% para cada uno de ellos, (uno de los pasos extra el cual aplicamos fue quitar categorías que en realidad no eran categorías eran links o descripciones breves del libro, estos datos se omitieron)

```
1  df_results_snd_sampling = spark.sql('select d.Title, d.description, d.categories, d.ratingsCount , r.`review/summary` , r.`review/text`, r.`review/helpfulness`
   from  demo.book_reviews r inner join demo.book_details d on r.Title = d.Title')
2  display(df_results_snd_sampling)
```

▸ (3) Spark Jobs

▸ df_results_snd_sampling: pyspark.sql.dataframe.DataFrame = [Title: string, description: string ... 5 more fields]

```
1  from pyspark.sql.functions import col,length
2  df_results_snd_sampling_filter = df_results_snd_sampling.filter(length(col('categories')) < 30)
3  fractions = df_results_snd_sampling_filter.rdd.map(lambda x:
4     (x[2])).distinct().map(lambda x:
5                            (x,0.4)).collectAsMap()
6  fractions
```

SAMPLING SELECTIVOS Y ESTRATIFICADO

Después de crear las fracciones para cada categoría , se aplica el método en Spark sampleBy.
Dada la cantidad de datos se decidió realizar un segundo sampling simple para reducir la cantidad de datos.

```
1  strat_sampling_df = df_results_snd_sampling_filter.sampleBy("categories", fractions)
```

```
strat_sampling_df: pyspark.sql.dataframe.DataFrame
  Title: string
  description: string
  categories: string
  ratingsCount: integer
  review/summary: string
  review/text: string
  review/helpfulness: string
```

```
ommand took 2.02 seconds -- by jesus.ramosdv@uanl.edu.mx at 6/29/2023, 12:02:09 PM on JESUS DAVILA's Cluster
```

Cmd 25

```
1  strat_df = strat_sampling_df.sample(frac =.007)
```

```
Command took 0.09 seconds -- by jesus.ramosdv@uanl.edu.mx at 6/29/2023, 12:26:48 PM on JESUS DAVILA's Cluster
```

Cmd 26

RESULTADOS

Se obtuvieron resultados un poco similares, como se esperaba cuando se realiza un sampling simple y sesgado se obtuvieron más sentimientos positivos que en el estratificado el cual nos podría indicar que los libros tienden a tener un sentimiento positivo, pero esto estaría sesgado desde un inicio.

Sampling selectivo y Estratificado

Sentimiento	Porcentaje
Positivo	50%
Negativo	41%
Neutro	9%

Sampling Simple y Sesgado

Sentimiento	Porcentaje
Positivo	54%
Negativo	38%
Neutro	8%

REFERENCIAS

Kaggle Dataset

https://www.kaggle.com/datasets/mohamedbakhet/amazon-books-reviews

Spark sampleBy()

https://www.skytowner.com/explore/pyspark_dataframe_sampleby_method

https://spark.apache.org/docs/3.1.1/api/python/reference/api/pyspark.sql.DataFrame.sampleBy.html