Informe N°4 Base de datos avanzadas

Dante Hortuvia
Diego Lara

1 Introducción

En este laboratorio se explora el uso de una nueva plataforma de bases de datos, específicamente Elasticsearch. A lo largo del ejercicio, se lleva a cabo una serie de tareas que incluyen la descarga de letras de canciones y el desarrollo de un programa capaz de cargar hasta dos gigabytes de datos en la base de datos. Posteriormente, se importan archivos CSV y se implementa una función de búsqueda que permite encontrar canciones que contengan una palabra o frase específica. Además, se utiliza la herramienta Discover de Elasticsearch para crear tres gráficos requeridos por el laboratorio, lo que facilita la visualización y análisis de los datos cargados. Este laboratorio proporciona una experiencia práctica en la gestión y análisis de grandes volúmenes de datos, así como en la utilización de herramientas avanzadas de búsqueda y visualización.

2 Desarrollo

Lo primero en realizarse es la creación del código que permite el ingreso de dos gigabytes de datos el cual lee archivos CSV de una carpeta específica y carga sus datos en índices, se emplea utilizando un cliente de Elasticsearch configurado con credenciales. luego busca en la carpeta output un archivo .csv para de este saca el nombre para convertirlo en indice, luego de encontrar el csv este lo envia a la funcion Ingresar Datos la cual crea una lista en formato elastic para que Bulk (funcion de elastic) pueda leerlo sin ninguna complicacion, donde el indice del archivo seleccionado queda como "music-nombre del csv", esto genera 83 inidices en total (son 81 archivos pero elastic crea 2 mas adicionales, sin ningun tipo de informcion), esto nos genera en kibana la cantidad 1.380.485 datos.

```python
from elasticsearch import Elasticsearch, helpers
import os
import csv
# Password for the 'elastic' user generated by Elasticsearch
ELASTIC_PASSWORD = "ei2WV2OvzahwS7deT512uKPR"

# Found in the 'Manage Deployment' page
CLOUD_ID = "Tareabases:dXMtY2VudHJhbDEuZ2NwLmNsb3VkLmVzLmlvOjQ0MyRmNmU4MzAyMjFhOTQ0ZTE1YWIyNTE1NTY4MDcyNDFjYSRjYjI2Mzk1NGE0MDM0YmZhOT81NjdlZTczOWZhZGE0Mw=="

# Create the client instance
client = Elasticsearch(
    cloud_id=CLOUD_ID,
    basic_auth=("elastic", ELASTIC_PASSWORD)
)
#
def ingresar_datos(Ruta_archivo,Nombre_indice):
    with open(Ruta_archivo, 'r', encoding='ISO-8859-1') as archivo:
        data = csv.DictReader(archivo, delimiter='|')
        Datos = [
            {
                "_index": Nombre_indice,
                "_source": {
                    "Url":fila['Url'],
                    "Artist":fila['Artist'],
                    "Title":fila['Title'],
                    "Lyric":fila['Lyric']
                }
            }
            for fila in data
        ]
        helpers.bulk(client, Datos)

def main():
    carpeta = "C:/Users/Dante hortuvia/Desktop/Tarea_4/output"
    carp = os.listdir(carpeta)
    for archivo in carp:
        if archivo.endswith('.csv'):
            ruta_completa = os.path.join(carpeta, archivo)
            nombre_archivo = "music-" + os.path.splitext(archivo)[0]
            ingresar_datos(ruta_completa,nombre_archivo)
            print(f'Se creo {nombre_archivo}')
    print("Terminado")
if __name__ == "__main__":
    main()
```

Figura 1: Código ingreso de datos

Index name	Index health	Docs count	Ingestion name	Ingestion method	Ingestion status	Actions
metrics-endpoint.metadata_current_default	green	0		API	Connected	
music-alternativo-indie	green	27402		API	Connected	
music-axe	green	24149		API	Connected	
music-bachata	green	2819		API	Connected	
music-blues	green	16022		API	Connected	
music-bolero	green	6622		API	Connected	
music-bossa-nova	green	19493		API	Connected	
music-brega	green	13675		API	Connected	
music-classico	green	15913		API	Connected	
music-corridos	green	4548		API	Connected	
music-country	green	38279		API	Connected	
music-cuarteto	green	5388		API	Connected	
music-cumbia	green	9073		API	Connected	
music-dance	green	14767		API	Connected	

Figura 2: Algunos Índices creados

Luego se creó un segundo código, en este caso para la busqueda sobre coincidencias de las letras con palabras o frases especificas, que en el contexto de nuestro laboratorio se pide que sea la frase "ciudad de la furia".Para esto se creo un código que busca documentos donde el campo Lyric contiene exactamente la frase "ciudad de la furia". Esto hara que se busque en el texto para obtener palabra objetivo lo que hace que sea una búsqueda de texto estandar, si quisieramos usar Lyric.Keyword tendriamos que haber definido la palabra a buscar, en el mapeo del índice, al no haber hecho esto se debio buscar de manera estandar

```
GET _search
{
  "query": {
    "match": {
      "Lyric": "ciudad de la furia"
    }
  }
}
```

Figura 3: Código busqueda

```
"hits": {
  "total": {
    "value": 10000,
    "relation": "gte"
  },
  "max_score": 30.920124,
  "hits": [
    {
      "_index": "music-progressivo",
      "_id": "Cx-Wp48BkbTy7aD_cquV",
      "_score": 30.920124,
      "_ignored": [
        "Lyric.keyword"
      ],
      "_source": {
        "Url": "https://www.letras.mus.br/elfonia/447991/maquina-print.html",
        "Artist": "Elfonia",
        "Title": "MÃ¡quina",
        "Lyric": "    GuÃame, dentro de la mÃ¡quina    el ruido es frio, la voz se apaga;    no hay signos
          de libertad.        La furia del mar    jamÃ¡s se detiene;    detrÃ¡s de sus muros    la luna me
          espera.        Cambio a merced de la ciudad    y el espacio que soÃ±Ã© se encoge.    Â¿HabrÃ¡
          caminos mÃ¡s allÃ¡?   "
      }
    },
    {
      "_index": "music-rock-alternativo",
      "_id": "HyGap48BkbTy7aD_Lt3O",
      "_score": 30.092674,
      "_ignored": [
        "Lyric.keyword"
      ],
      "_source": {
        "Url": "https://www.letras.mus.br/aterciopelados/en-la-ciudad-de-la-furia/en-la-ciudad-de-la-furia
          -print.html",
        "Artist": "Aterciopelados",
        "Title": "En La Ciudad de La Furia",
        "Lyric": "    Me verÃ¡s volar    Por la ciudad de la furia    Donde nadie sabe de mÃ    Y yo soy
          parte de todos        Nada cambiarÃ¡    Con un aviso de curva    En sus caras veo el temor    Ya no
          hay fÃ¡bulas en la ciudad de la furia        Me verÃ¡s caer    Como un ave de presa    Me verÃ¡s
          caer    Sobre terrazas desiertas    Te desnudarÃ©    Por las calles azules    Me refugiarÃ©
          Antes que todos despierten        Me dejarÃ¡s dormir al amanecer    Entre tus piernas    Entre tus
          piernas    SabrÃ¡s ocultarme bien y desaparecer    Entre la niebla    Entre la niebla        Un
          hombre alado extraÃ±a la tierra    Un hombre alado protege la tierra        Me verÃ¡s volar    Por
          la ciudad de la furia    Donde nadie sabe de mi    Y yo soy parte de todos        Con la luz del
```


Figura 4: Algunos resultados de la busqueda

Posteriormente, utilizando la herramienta Kibana en la opcion discover, se crearon tres gráficos para el análisis de datos. El primer gráfico fue un gráfico circular que muestra la distribución de canciones por artista. El segundo gráfico fue un gráfico de barras que ilustra la cantidad de canciones por género, esto se hace solo para las 10 primeros artistas y generos musicales. El tercer gráfico, también de barras, muestra las palabras que mas se repiten en los titulos(title.keyword) , donde se muestran las 1000 pabras mas utilizadas en los titulos(title.keyword).

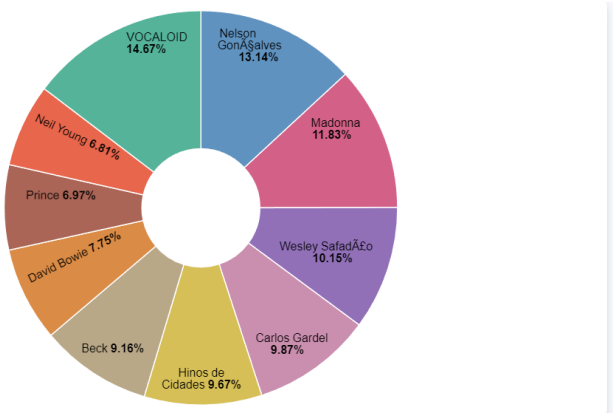

Figura 5: Gráfico circular

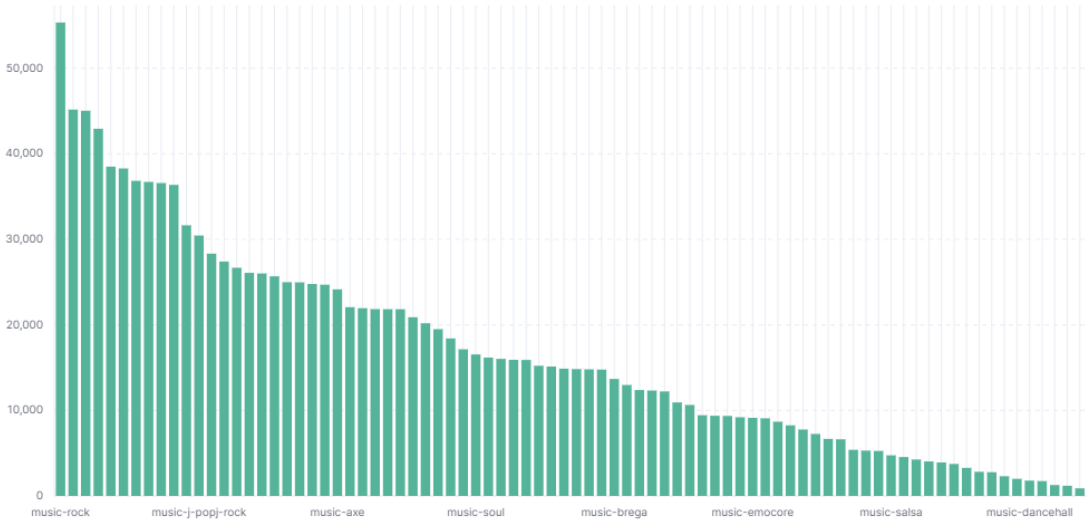

Figura 6: Primer gráfico de barras

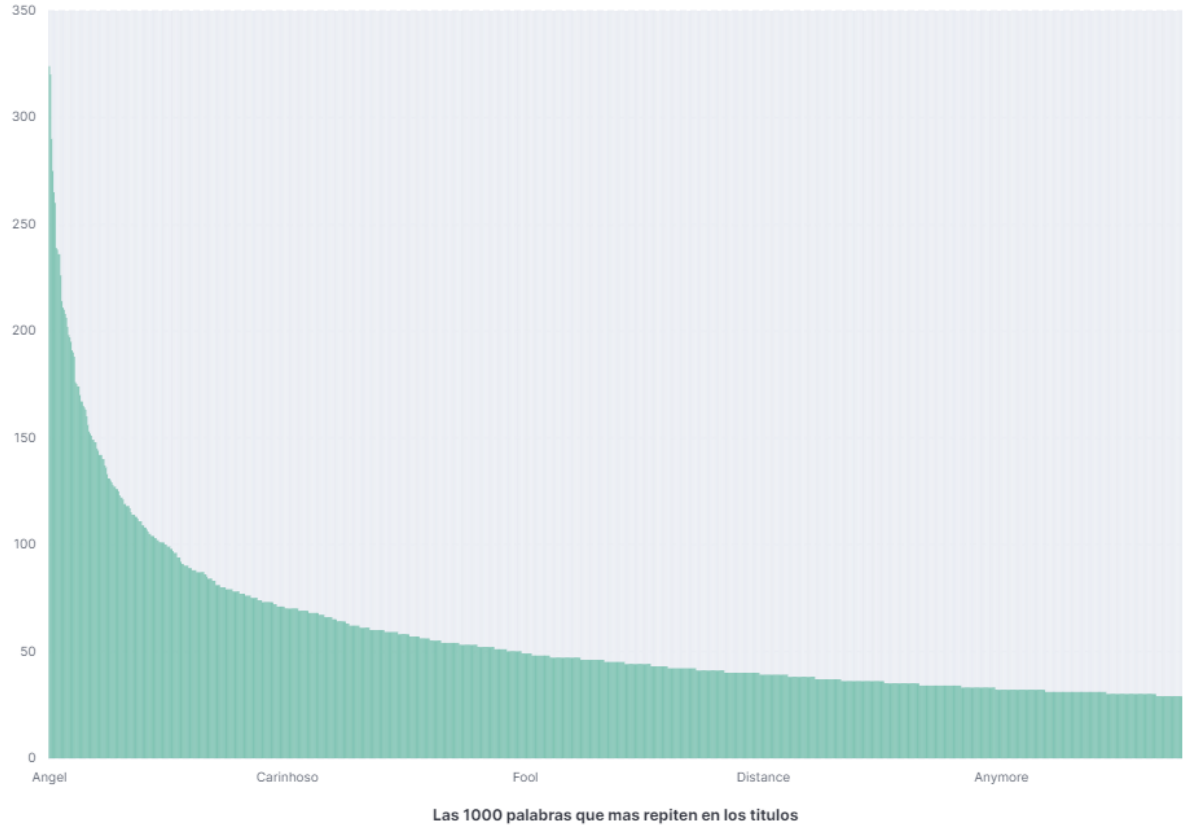

Figura 7: Segundo gráfico de barras

3 Conclusión

En este laboratorio, se exploró el uso de Elasticsearch como plataforma de bases de datos para la gestión y análisis de grandes volúmenes de datos. A lo largo del ejercicio, se realizaron varias tareas clave que permitieron una comprensión profunda de las capacidades de Elasticsearch y su integración en aplicaciones de análisis de datos.

Primero, se desarrolló un programa para cargar datos desde archivos CSV a la base de datos de Elasticsearch. Este proceso incluyó la creación de un cliente de Elasticsearch utilizando credenciales específicas y la implementación de una función para leer los archivos CSV y cargar sus datos en índices de Elasticsearch. Esta actividad no solo facilitó la ingesta de datos en grandes volúmenes, sino que también destacó la eficiencia de Elasticsearch en la manipulación y almacenamiento de datos estructurados.

Luego, se creó un segundo programa para realizar búsquedas en los datos cargados. Específicamente, se implementó una función de búsqueda que permite encontrar canciones que contienen una palabra o frase específica, como "ciudad de la furia". Esta búsqueda se realizó utilizando consultas match de Elasticsearch, demostrando cómo se pueden realizar búsquedas de texto precisas y eficaces en grandes conjuntos de datos.

Finalmente, se utilizó la herramienta Discover de Elasticsearch para crear visualizaciones de los datos. Se generaron tres gráficos: un gráfico circular que muestra la distribución de canciones por artista, un gráfico de barras que ilustra la cantidad de canciones por género, y otro gráfico de barras que muestra las canciones que más se repiten. Estas visualizaciones facilitaron una comprensión clara y visual de las tendencias y patrones en los datos, resaltando la capacidad de Elasticsearch para no solo almacenar y buscar datos, sino también para analizarlos y presentarlos de manera intuitiva.

A través de este laboratorio, se adquirió una experiencia práctica en la gestión de bases de datos con Elasticsearch, la manipulación de grandes volúmenes de datos, la ejecución de búsquedas avanzadas y la creación de visualizaciones informativas. Este conocimiento es fundamental para el análisis de datos a gran escala y proporciona una base sólida para el uso de herramientas avanzadas de búsqueda y visualización en proyectos futuros.

Referencias

"Elasticsearch: The Definitive Guide" de Clinton Gormley y Zachary Tong.

"Mastering Elasticsearch" de Bahaaldine Azarmi.