Laboratorio 14:

Prof. Heider Sanchez
ACLs: Ana María Accilio, Sebastián Loza

El objetivo de esta actividad es aplicar técnicas de indexación de texto completo en MongoDB, utilizando datos no estructurados provenientes de dos fuentes distintas. A través del laboratorio, se reforzarán los conocimientos sobre la ejecución de consultas eficientes en grandes volúmenes de texto dentro de bases de datos NoSQL. Además, el laboratorio se desarrollará sobre una configuración de Sharding, lo que permitirá explorar estrategias de distribución y escalabilidad para optimizar el rendimiento de las búsquedas.

1. (6 pts) Configuración del Sharding

- Siguiendo el tutorial configurar en su propio entorno local con Dockers un Sharding cond dos Shards.
- Crea un nueva conexión en MongoDB Compass para conectarse al Sharding y asignarle su propio nombre (ejemplo "Cluster_Heider").
- Ejecutar el comando rs.status() y tomar un screenshot de su configuración y guardarlo como "sharding.jpg". Debe ser notorio el uso de varios servidores.
- Trabajar el resto de la actividad en dicho cluster.

2. (14 pts) Búsqueda de Texto en Grandes Volúmenes de Datos

Se le pide aplicar técnicas de indexación de texto completo en MongoDB, utilizando datos no estructurados provenientes de dos fuentes distintas.

Datasets a utilizar:

Deberás trabajar con los siguientes conjuntos de datos:

1. Drake Lyrics Datase
   Contiene letras de canciones del artista Drake.

2. News Headlines About President Milei
   Colección de titulares de noticias referidas al presidente argentino Javier Milei, con información adicional como fecha de publicación, medio, y categoría.

Actividades requeridas

1. **Carga de datos a MongoDB**
   - Crear una base de datos llamada `text_analysis`.
   - Crear dos colecciones: `drake_lyrics` y `milei_news`.
   - Cargar los datasets en sus respectivas colecciones utilizando `pymongo`.

2. **Creación de índices de tipo texto**

- Crear índices de texto sobre los campos relevantes para realizar búsquedas textuales:
  - Para `drake_lyrics`, en el campo `lyrics`.
  - Para `milei_news`, en el campo `title` y `summary`.

3. **Consultas en lenguaje natural**
   A partir de un input en lenguaje natural, plantear **al menos tres consultas distintas** usando `$text` para realizar búsquedas eficientes sobre los textos cargados. A modo de ejemplo:

   - **Consulta 1:** Buscar letras de canciones que hablen sobre "love and money".
   - **Consulta 2:** Encontrar titulares que mencionen la palabra "inflación".
   - **Consulta 3:** Buscar noticias que incluyan simultáneamente los términos "Milei" y "dólar", pero no "inflación".

4. **Visualización y análisis de resultados**

   - Mostrar los resultados de cada consulta con una visualización básica (puede ser una tabla simple con `pandas.DataFrame`).
   - Incluir una breve reflexión sobre la relevancia de los resultados obtenidos y cómo el uso de índices mejora el rendimiento de la búsqueda.
   - ¿Será posible mejorar mejorar la estructura de indexación para obtener los documentos más relevantes a una consulta textual de lenguaje natural?

Entregable

Parte 1: Configuración del Sharding (6 pts)

- Screenshot "sharding.jpg" con la ejecución de `rs.status()`
- Captura de MongoDB Compass conectado al cluster
- Explicación breve del proceso de configuración

Parte 2: Búsqueda de Texto Completo (12 pts)

- **Notebook Jupyter** completo con:
  - Conexión al cluster y carga de datasets
  - Creación de índices de texto
  - Tres consultas usando `$text`
- **Screenshots** de colecciones, índices y resultados
- **Análisis** con visualización en pandas y reflexión sobre rendimiento

> **Importante:** Código documentado y capturas legibles.