UNIVERSIDAD AUTÓNOMA DE SINALOA

UNIDAD ACADÉMICA FACAULTAD DE INFÓRMATICA CULIACÁN

LICENCIATURA EN INFÓRMATICA

ASIGNATURA:

Sistemas Distribuidos

TRABAJO:

Proyecto Final

PROFESOR:

Jesús Humberto Abundis Patiño

ALUMNO:

CANO CORVERA JESÚS ENRIQUE

NÚÑEZ AVENA HÉCTOR DE JESUS

GRUPO:

5-3

7 de diciembre de 2025

Funcionamiento Interno

1. Procesamiento Masivo de Datos (Backend con PySpark): El núcleo analítico del proyecto utiliza **PySpark**, una herramienta estándar en la industria para el procesamiento de Big Data. En lugar de procesar los libros uno por uno de manera secuencial, el sistema carga todos los archivos de texto ubicados en el directorio books/ en un DataFrame distribuido. El sistema implementa un *Pipeline* de Machine Learning que transforma el texto crudo en representaciones matemáticas. Primero, el texto se "tokeniza" (se divide en palabras individuales). Luego, se aplica un filtro de StopWordsRemover para eliminar palabras comunes que no aportan significado (como "el", "la", "y"). Finalmente, se utiliza la técnica **TF-IDF (Term Frequency - Inverse Document Frequency)**. Este algoritmo asigna un peso numérico a cada palabra, destacando aquellas que son únicas y representativas de cada libro, mientras penaliza las que aparecen en todos los textos. El resultado es que cada libro se convierte en un vector numérico de alta dimensión.

2. Cálculo de Similitud Matemática: Una vez que todos los libros han sido convertidos a vectores numéricos, el sistema exporta estos datos a **Pandas** y **Scikit-Learn** para calcular la **Similitud del Coseno**. Esta métrica mide el ángulo entre dos vectores; si dos libros tienen vectores con un ángulo muy cerrado (cercano a 0 grados), significa que utilizan un vocabulario muy similar y, por tanto, tratan temas parecidos. El resultado de este proceso es una "Matriz de Similitud" que se guarda en el disco duro (matriz.npy). Esto permite que, en el futuro, las consultas sean instantáneas sin tener que volver a procesar todos los libros.

3. Generación de Resúmenes con Deep Learning: Para la fase de consulta, el sistema no solo recupera los libros más similares basándose en la matriz precalculada, sino que también integra un modelo de **Deep Learning** de la librería Transformers de Hugging Face. Específicamente, utiliza el modelo facebook/bart-large-cnn, una red neuronal avanzada entrenada específicamente para tareas de "resumir" (summarization). El código extrae un fragmento significativo del libro (omitiendo licencias y encabezados irrelevantes) e inyecta este texto en la red neuronal. La IA procesa el contexto y genera una síntesis abstractiva, la cual es recortada programáticamente a 20 palabras para ofrecer una descripción rápida y directa al usuario.

Requisitos para correr el proyecto

1. Estructura de Carpetas Necesaria (Mapa de ejemplo)

Proyecto/

│

├── ProyectoSD.py

└── books/

  ├── 1.txt

  ├── 2.txt

  └── ...

2. Instalación de Librerías

pip install numpy pandas pyspark scikit-learn transformers torch

(Nota: torch es necesario para que funcione la librería transformers).

Guía de Ejecución

El programa tiene dos modos de uso que deben ejecutarse en un orden específico.

PASO 1: Configuración Inicial (Setup)

Este paso es obligatorio ejecutarlo la primera vez o cada vez que se agreguen nuevos libros a la carpeta books/.

Este proceso entrena el modelo y genera la matriz matemática.

Comando:

python ProyectoSDCompleto.py --setup

Al ejecutar este comando se verá una barra de progreso en la consola indicando etapas como "Pipeline NLP", "Tokenización" y "Guardando matriz". Al finalizar, se habrá creado una carpeta oculta con los cálculos llamada matriz_similitud.

PASO 2: Realizar una Consulta

Una vez completado el setup, se pueden pedir recomendaciones sobre un libro específico que exista en la carpeta. Se debe usar el nombre exacto del archivo.

Comando (Ejemplo):

python ProyectoSD.py 11.txt

Lo que sucederá:

1. El sistema cargará la matriz guardada.
2. Mostrará una lista de los 10 libros más similares ordenados por relevancia.
3. Descargará (si es la primera vez) el modelo de IA y generará un resumen inteligente de 20 palabras sobre el libro consultado.