Procesamiento de Grandes Volúmenes de Datos

Análisis de movilidad urbana con datos de tráfico vial

Integrantes:

Amalia González

Diego Puentes

Universidad de La Habana

Octubre 2025

Descripción del proyecto

Objetivo central: Analizar y optimizar el tráfico urbano mediante el procesamiento y análisis de grandes volúmenes de datos geoespaciales, con énfasis en patrones de movilidad vehicular, identificación de zonas críticas y estimación de tiempos de viaje.

Dataset seleccionado: Road Traffic Dataset, publicado por Arash Nic en Kaggle (https://www.kaggle.com/datasets/arashnic/road-trafic-dataset). Formato CSV.

Justificación del dataset:

- Volumen: Contiene 4 337 136 registros, lo que permite realizar análisis de Big Data sobre datos reales a gran escala.
- Características: El dataset posee estructura tabular, variables geoespaciales y temporales, identificadores de carreteras, categorías de vehículos y volúmenes de tráfico.
- Pertinencia: Permite analizar patrones de movilidad urbana, identificar zonas congestionadas y realizar análisis histórico de la red vial.

Columnas del primer CSV (tráfico horario detallado)

- `count_point_id`: Identificador único del punto de conteo.
- `direction_of_travel`: Dirección del flujo vehicular (E, W, N, S).
- `year`: Año del registro.
- `count_date`: Fecha exacta de conteo.
- `hour`: Hora de conteo.
- `region_id`, `region_name`: Identificadores y nombres de región.
- `local_authority_id`, `local_authority_name`: Autoridad local.
- `road_name`, `road_type`: Nombre y tipo de vía.
- `start_junction_road_name`, `end_junction_road_name`: Tramos de inicio y fin.
- `easting`, `northing`: Coordenadas proyectadas.
- `latitude`, `longitude`: Coordenadas geográficas.
- `link_length_km`, `link_length_miles`: Longitud del tramo.
- `pedal_cycles`: Conteo de bicicletas.
- `two_wheeled_motor_vehicles`: Motocicletas y similares.
- `cars_and_taxis`: Autos y taxis.
- `buses_and_coaches`: Autobuses.
- `lgvs`: Vehículos ligeros de carga.

- `hgvs_*`: Diferentes categorías de camiones pesados (según ejes y tipo).
- `all_hgvs`: Total de vehículos pesados.
- `all_motor_vehicles`: Total de vehículos motorizados.

Columnas del segundo CSV (agregado por autoridad local)

- `local_authority_id`, `local_authority_name`: Identificador y nombre de la autoridad local.
- `year`: Año.
- `link_length_km`, `link_length_miles`: Longitud total de los tramos viales bajo esa autoridad.
- `cars_and_taxis`: Total anual de autos y taxis.
- `all_motor_vehicles`: Total anual de vehículos motorizados.

Columnas del tercer CSV (agregado por región y categoría de vía)

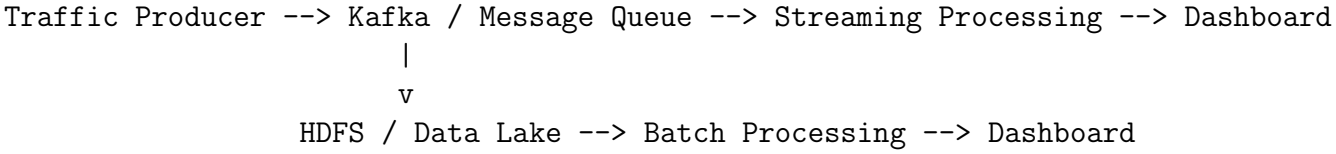
- `year`: Año.
- `region_id`, `region_name`: Identificadores y nombre de región.
- `road_category_id`, `road_category_name`, `road_category_description`: Clasificación de las carreteras.
- `total_link_length_km`, `total_link_length_miles`: Longitud total de tramos en esa categoría y región.
- `pedal_cycles`, `two_wheeled_motor_vehicles`, `cars_and_taxis`, `buses_and_coaches`, `lgvs`, `all_hgvs`, `all_motor_vehicles`: Volúmenes anuales agregados por tipo de vehículo.

Arquitectura propuesta

Enfoque: Combinación de *batch* y *streaming* para procesar grandes volúmenes de datos de tráfico en tiempo real y análisis histórico.

- **Traffic Producer:** Genera datos continuos simulando sensores de tráfico, GPS o registros históricos. Esto permite probar el pipeline en condiciones de tiempo real.
- **Kafka / Message Queue:** Actúa como buffer escalable, desacoplando la producción de datos de su procesamiento, asegurando tolerancia a fallos y soporte a altos volúmenes de eventos.
- **Procesamiento en streaming (Spark Streaming / Flink):** Calcula métricas en tiempo real, como densidad vehicular, rutas congestionadas y flujos de vehículos. También produce información lista para visualización inmediata.
- **Almacenamiento histórico (HDFS / Data Lake):** Guarda datos crudos y agregados para análisis batch, entrenamientos de modelos predictivos y consultas históricas.
- **Procesamiento batch (Spark):** Limpieza, transformación y agregación de datos históricos para generar KPIs, métricas por región, autoridad local y categoría de carretera.
- **Dashboard / Visualización:** Integra datos en tiempo real y agregados históricos mostrando mapas de calor, flujos de vehículos, rutas congestionadas y evolución temporal de la movilidad urbana.

Diagrama conceptual del pipeline:



Explicación del flujo:

- Los datos generados por el Traffic Producer fluyen hacia Kafka, donde se almacenan temporalmente para garantizar procesamiento confiable.
- Spark Streaming consume los datos de Kafka para métricas en tiempo real, como mapas de calor, flujos de vehículos y alertas de congestión.
- Paralelamente, los datos se almacenan en HDFS para procesamiento batch con Spark, permitiendo análisis histórico, agregaciones y generación de KPIs.
- El dashboard final combina la información en tiempo real con los datos históricos, proporcionando una visión completa para optimización de tráfico y rutas urbanas.

Generación de datos sintéticos y flujo de procesamiento

Para simular y analizar el tráfico urbano, se ha implementado un **Producer de datos sintéticos** capaz de generar grandes volúmenes de información consistente con los patrones del dataset histórico.

Flujo general del pipeline

- **Traffic Producer:**
  - Basado en perfiles estadísticos obtenidos a partir del dataset histórico limpio.
  - Genera registros con atributos como región, autoridad local, tipo de carretera, hora del día, día de la semana, longitud de tramo, densidad vehicular, proporción de vehículos pesados, coordenadas geográficas y número total de vehículos.
  - Cada registro es consistente con las distribuciones observadas en el dataset real, incluyendo correlaciones entre variables como densidad y tipo de carretera.
- **Almacenamiento histórico (HDFS):**
  - Recibe todos los registros generados y sirve como fuente para análisis batch.
  - Facilita consultas agregadas y generación de KPIs históricos.
- **Procesamiento batch (Spark):** limpieza, agregación y análisis histórico de datos para entrenar modelos predictivos y evaluar patrones de tráfico a largo plazo.

Funcionamiento del Producer

El producer se basa en los perfiles estadísticos obtenidos con Spark a partir del dataset limpio de tráfico:

- **Distribuciones de probabilidad:** regiones, autoridades locales, tipo de carretera, horas del día y días de la semana.
- **Perfiles numéricos:** longitud de tramo, densidad vehicular, número total de vehículos y proporción de vehículos pesados, modelados mediante distribuciones logarítmicas con percentiles 5 % y 95 %.
- **Generación aleatoria controlada:** para cada registro se realiza un muestreo ponderado de las distribuciones y perfiles, asegurando que los datos sintéticos respeten las estadísticas del dataset original.
- **Coherencia geoespacial:** coordenadas y autoridades locales se generan con media y desviación estándar derivadas del dataset real.

Validez y utilidad de los datos sintéticos

- Los registros reflejan las características estadísticas observadas en el dataset real de 4,337,136 filas, garantizando que las simulaciones sean representativas.
- Se conservan correlaciones relevantes entre variables clave (densidad, tipo de carretera, proporción de vehículos pesados), lo que permite análisis válidos de movilidad urbana.
- El producer puede generar grandes volúmenes de datos de manera continua (˜1500 registros por minuto), demostrando escalabilidad y capacidad para pruebas de procesamiento *Big Data*.
- Los datos son directamente útiles para alimentar pipelines de streaming y dashboards interactivos, así como para entrenar modelos predictivos o realizar análisis históricos.

Diagrama conceptual actualizado

```
Traffic Producer --> Spark Streaming --> Dashboard
        |
        v
      HDFS --> Spark Batch --> Dashboard
```

Cada bloque está alineado con los objetivos del proyecto: optimización de tráfico y rutas urbanas, visualización en tiempo real, identificación de zonas críticas y predicción de tiempos de viaje.