

Algoritmo de predicción de probabilidad en arribo en tiempo de vuelos



HERRAMIENTAS DE SOFTWARE PARA BIG DATA

FACULTAD DE INGENIERÍA ORT URUGUAY

Ingeniero Yader Coca

Montevideo 2019



Resumen

- Las aerolíneas alrededor del mundo generan una infinidad de datos que representan las estadísticas de los vuelos que operan. Estos van desde los tiempos en que se retrae el tren de aterrizaje hasta los horarios ,las tardanzas y los aeropuertos en los cuales se mueven. Es interesante dado un algoritmo de machine learning predecir que vuelos dado un origen y destino pudieran llegar temprano.
- Palabras Claves: machine learning,random forest,,Spark Hadoop,Hue.



Objetivos

Objetivo General:

- Desarrollar algoritmo Random Forest para los vuelos domésticos de Estados Unidos.

Objetivos Específicos:

- Definir ingeniería de atributos, seleccionar el conjunto de datos, correlación de los atributos y establecer los datos para hacer una recomendación.
- Seleccionar el algoritmo y el conjunto de datos de entrenamiento y de prueba.
- Matriz de confusión para establecer el rendimiento del algoritmo seleccionado
- Automatizar el algoritmo.



Metodología Científica

- Para la realización de este trabajo me base en la bibliografía consultada y los materiales propuesto ,llegando a las conclusiones basadas en los objetivos planteados.



Problema

- Se necesita saber teniendo los datos históricos de varias aerolíneas dentro de Estados Unidos cual es la probabilidad de que dada una aerolínea una fecha de salida, un punto de salida y de llegada que llegue con demora.



Justificación y Alcance

Justificación:

- Apoyo a la logística de los aeropuertos para así poder mover los pasajeros de acuerdo a la carga y demora de los vuelos.

Alcance:

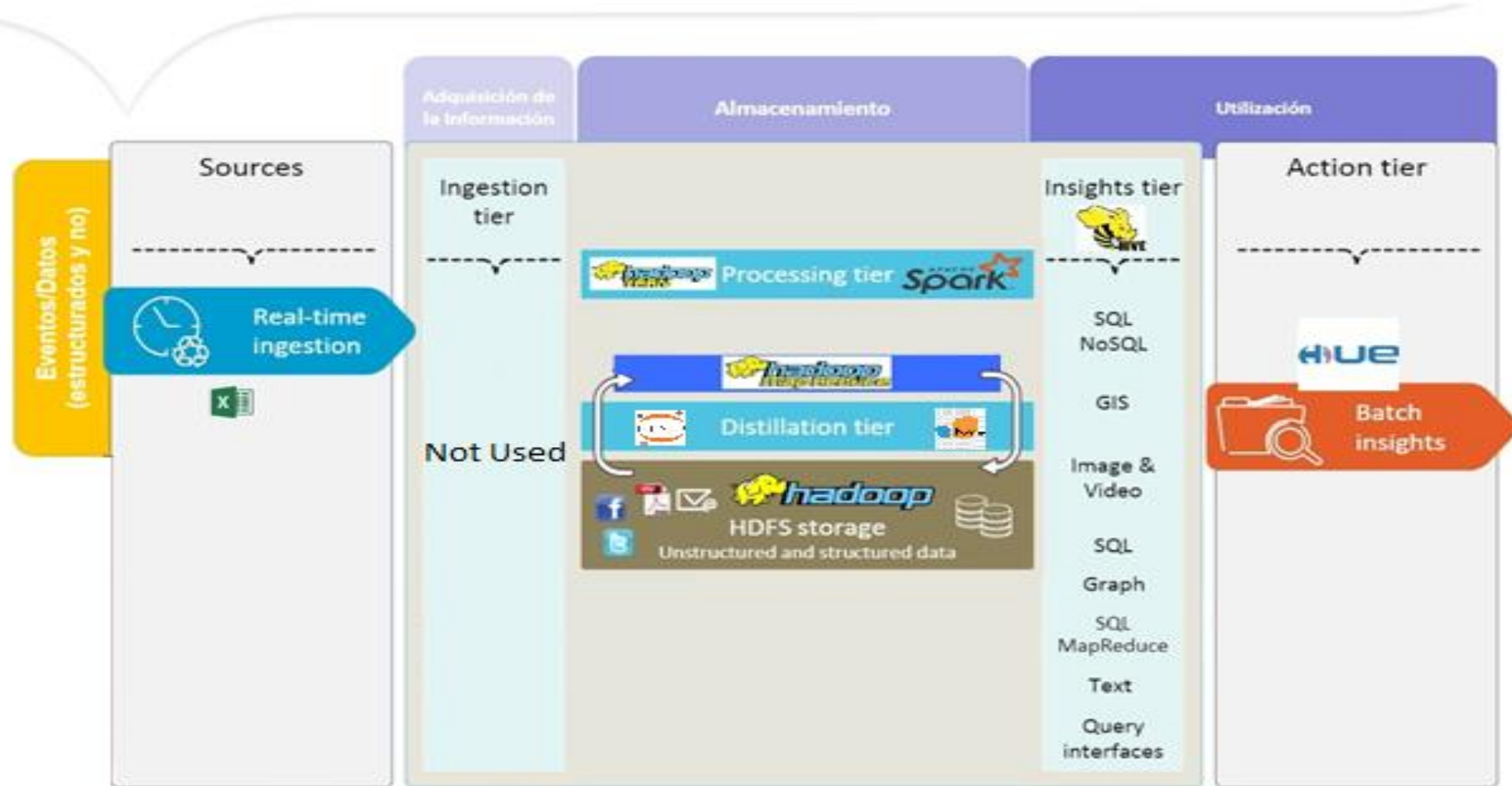
- Recopilar información de aeropuertos, vuelos y aerolíneas sobre los vuelos en Estados Unidos en el año 2015. Introducir estos datos en un sistema de archivos distribuidos (HDFS). Leer estos archivos haciendo uso de Hive. Utilizar Spark para levantar la información de las tablas en formato Parquet. Implementar ingeniería de atributos y utilizar el algoritmo de Random Forest para encontrar la predicción sobre si llegara temprano.



Limitaciones

- Entre las dificultades que se encontraron para este trabajo se encuentra la capacidad de computo a la hora de realizar el algoritmo de machine learning que hace complicado el mejoramiento de los resultados dado al tiempo de demora en realizar cada corrida. También con el uso de máquinas virtuales dependientes de los recursos de la máquina *host* .



Propuesta







Conclusiones

- Uno de los aspectos mas importantes de la preparación del dataset fue la selección de los features que son relevantes a nuestros resultados. Filtré las columnas que no afectaran los resultados para así evitar sesgo y multicolinealidad. Otra tarea impórtate para la ingeniería de atributos lo constituyó la eliminación de valores vacíos o faltantes y donde estaban.
- Al no contar con una columna que me permitirá efectivamente clasificar el resultado de si llegaba tarde el vuelo o no, agregué una columna llamada *late* que contiene el valor de 1 o 0 dependiendo de si la columna *arrival delay* es mayor que cero.

- 
- 
- Sobre esta columna evalué la predicción. Si llega tarde se evalúa la columna en 1 y si no se evalúa en cero. La idea es evaluar la probabilidad de un vuelo dado llega temprano a su destino.
 - Luego me era importante agrupar las fechas a la hora mas cercana para que no estuvieran fraccionadas y así facilitar el trabajo de predicción.
 - Aquí los datos quedaron listos para utilizar un modelo sobre ellos. Para el modelo elegí hacer una clasificación usando la librería **scikit-learn** que incluye una amplia variedad de clases para implementar modelos de machine learning comunes. También se seleccionaron las particiones de entrenamiento y prueba usando la mencionada librería que nos facilita con una función simple este trabajo.

- 
- 
- Este algoritmo debe correr como una tarea programada ya sea usando un *cron job* o una tarea programada dependiendo del sistema operativo del servidor todos los días y así reentrenar el modelo y devolver las predicciones en demanda.



Recomendaciones

- Se recomienda ampliar el entrenamiento y el *tunning* de el algoritmo de predicción así como implementar una ingesta de datos propiamente en tiempo real y así probar la capacidad de aprendizaje y el manejo de la carga sobre datos no conocidos.

Bibliografía

■ Bibliografía

- Bonillo, P. B. (s.f.). Propuesta de una Arquitectura de Gestión de Grandes Volúmenes de Datos para la Analítica en Tiempo Real bajo Software Libre. Recuperado 30 noviembre, 2019, de <https://www.linkedin.com/pulse/propuesta-de-una-arquitectura-gesti%C3%B3n-grandes-datos-la-bonillo-ramos>
- Chakrabarty, N. C. (2019, 15 marzo). A Data Mining Approach to Flight Arrival Delay Prediction for American Airlines. Recuperado 1 diciembre, 2019, de <https://deepai.org/publication/a-data-mining-approach-to-flight-arrival-delay-prediction-for-american-airlines>
- Domínguez, J. D. (2019, 1 noviembre). De Lambda a Kappa: evolución de las arquitecturas Big Data. Recuperado 30 noviembre, 2019, de <https://www.paradigmadigital.com/techbiz/de-lambda-a-kappa-evolucion-de-las-arquitecturas-big-data/>
- Zorrilla, M. Z., García, D. G., & Saiz, S. (2017, enero). Arquitecturas y tecnologías para el bigdata [Documento]. Recuperado 30 noviembre, 2019, de <https://ocw.unican.es/pluginfile.php/2396/course/section/2473/tema%203.2%20Arquitecturas%20y%20tecnologi%C3%A1as%20para%20el%20big%20data.pdf>