ML Canvas – Scania Trucks Air Pressure System Failure Prediction

En esta oportunidad, se desarrollará un producto analítico para abordar un problema de mantenimiento predictivo en camiones detectado en una empresa de transporte.

El sistema de aire comprimido (APS por sus siglas en inglés) se considera un factor clave en el mantenimiento, ya que el sistema de frenos requiere un suministro constante de aire comprimido dentro de una presión máxima y mínima predeterminada a fin de funcionar correctamente.

Entonces, se desarrollará un modelo predictivo de Machine Learning, de aprendizaje supervisado, que identificará fallas en componentes del APS antes de que ocurran. Los datos de entrenamiento se generan a partir de sensores instalados en el camión y son provistos por Scania (fabricante).

A continuación, se desglosan todos los detalles acerca del modelo:

Bloque 1: Propuesta de valor

¿A qué **objetivo de negocio** estamos sirviendo?
Con la implementación exitosa del modelo, se estará contribuyendo al objetivo de negocio de optimizar los costos de mantenimiento de la flota de camiones.

¿Quién es el **usuario final** de nuestro sistema?
El usuario final será el área de Taller, encargada de planificar y efectuar el mantenimiento de los vehículos.

¿Cual es el **problema** que estamos intentando resolverle?
El mantenimiento predictivo del APS de los camiones no se encuentra performando bien, ya que está pronosticando fallas erróneamente, dando falsos positivos (el APS no fallará y sí lo hace) y falsos negativos (fallas pronosticadas del APS que nunca ocurren).

¿Por qué es **importante** hacerlo?
Para evitar fallas durante la operación de los camiones.

¿Qué **valor** le entrega el proyecto de ML al usuario final?
Para el área de Taller quien necesita mejorar el proceso de mantenimiento predictivo, el modelo de ML permite identificar fallas en el APS, evitando fallas durante la operación y minimizando así los costos de mantenimiento.

¿**Cómo** va a usar nuestras salidas o predicciones?
A través de un dashboard, en la herramienta de AWS Quicksight, donde se podrán ver las predicciones para cada unidad de la flota.

Bloque 2: Fuentes de datos

¿Cuáles son sus **fuentes de datos**?
La principal fuente de datos será la base de datos de Scania, a la cual los usuarios de la empresa acceden a través de un sitio web. También se dispondrá de datos históricos acerca de las tareas de mantenimiento realizadas a cada camión, viajes realizados, distancias recorridas, etc.

Bases de datos internas/externas:

1. Sistema transaccional ERP de la empresa (MySQL), donde se registran todos los viajes;
2. APIs REST para acceder de manera automática a los últimos registros de la base de datos de Scania;

¿Cuáles son los **costos ocultos** de la aplicación de ML?

¿Cuánto podrían **crecer** los datos?
Podrían crecer considerablemente, sobre todo el almacenamiento de la data cruda. Debemos considerar que los algoritmos que predicen la aparición de fallos funcionan mejor con una cantidad muy grande de datos, por lo que su rendimiento se optimiza si disponen de información en tiempo real y casi continua del estado de las máquinas. Por ello, el modelo se alimenta de 170 atributos para realizar una predicción.

¿Que tan costoso se puede volver el **almacenamiento** de los datos?

¿Deberíamos **comprar** datos externos para resolver el problema?
No, es suficiente con los datos que ya se disponen.

Bloque 3: Tarea de predicción

¿Es **supervisada** o no supervisada?
El algoritmo será supervisado.

¿Es detección de **anomalías**?
Podría decirse que sí. El algoritmo identifica anormalidades en distintos parámetros, a partir de lo cual induce una falla.

¿Necesitamos predecir un valor **continuo**?
No, se trata de un algoritmo de clasificación binaria, con etiquetas positivas y negativas.

¿Qué **categoría** debe ser predicha?
Los datos de etiqueta positiva se recopilaron de camiones con fallas de componentes en el APS, mientras que los datos de etiqueta negativa pertenecen a camiones sin fallas en el APS. El set de entrenamiento contiene 60000 registros en total (98% negativos y 2% positivos), por lo cual estamos hablando de un conjunto de datos muy desbalanceado.

¿Necesitamos **agrupar** nuestra data?
No.

¿Cuál sería la definición de un **sample**?
Un sample comprendería todos los registros de las features. Es decir, cada vez que se genera un dato en las 170 variables se genera un nuevo sample.

¿Será un modelo de Machine Learning o de Deep Learning?
Machine Learning.

Bloque 4: Ingeniería de features

¿Cómo se **extraen** los features de las fuentes crudas?
La data será extraída desde las BD de Scania, a través de una API Rest, con herramientas de AWS. Dicha extracción se realizará de manera automática con AWS Glue, que aplicará un job ETL y almacenará la data cruda en un bucket de S3.

¿Hay que **cruzar** muchas fuentes de datos?

- ¿Hay que hacer un procesamiento complejo para que sean **útiles**?
  Sí, será necesario un procesamiento previo, el cual consistirá en reemplazar los ceros y NaN y remover los outliers usando intercuartiles.

- ¿Hay **personal** o **herramientas** para hacer esta labor?

Como se mencionó anteriormente, se utilizarán las herramientas provistas por AWS Cloud para el pipeline de datos completo.

- ¿Se considera incluir a **expertos** del dominio para especificar qué aspectos de la data son los más importantes para esta tarea de ML en particular?
  Sí, se considera la contratación de un Data Scientist para que mantenga el pipeline de datos en correcto funcionamiento.

Bloque 5: Evaluación offline

Antes de implementar cualquier tipo de modelo debemos especificar y establecer la metodología y métrica para evaluar el **sistema completo**

- Métricas **específicas del dominio** que justifiquen el desarrollo del modelo.
  Una métrica importante para este caso es la reducción de costos de mantenimiento. Para ello, el modelo buscará minimizar la siguiente función de costos:

  $$Costo = 10 * FP + 500 * FN \quad \textbf{(1)}$$

  donde:
  - ➔ FP: falsos positivos (se pronostica un fallo que en realidad no termina ocurriendo);
  - ➔ FN: falsos negativos (no se detecta un fallo del sistema de aire comprimido).
  - ➔ La distinta ponderación para los FP y FN radica en que es preferible revisar un camión que no tuvo fallas (FP) antes que no identificar una falla (FN) que podría interrumpir la operación del camión y, aún peor, ocasionar un accidente.

- Esa meta del negocio tiene que ser medible con la metodología **S.M.A.R.T** (Specific, Measurable, Achievable, Relevant, and Time-bound)

- ¿Cuál será la **performance mínima** con la que se autorizará la puesta productiva del modelo?

Para poner en marcha el modelo, Macro F1-Score debe ser al menos 0,65. Esta métrica hace un promedio de las métricas F1 de las dos clases, tratando a ambas por igual, lo cual resulta apropiado para este caso de clases desbalanceadas.

- ¿Cuales son las consecuencias medibles en los **errores de predicción** del modelo como los Falsos Positivos y los Falsos Negativos?
  El costo de una clasificación errónea es muy alto, dado que una falla no detectada de componentes del APS puede provocar fallas del camión durante la operación y, por lo tanto, un aumento en el costo de mantenimiento. Como se indica en la fórmula (1), la empresa considera que el costo de un FN es 50 veces más caro que el costo de un FP.

Bloque 6: Toma de decisiones

Luego de definir la tarea de predicción, ingeniería de features y evaluación offline.

- ¿Cómo **interactúa** el usuario final u otro sistema con las predicciones del modelo?
  El usuario final puede ver por sistema cuales son las unidades de la flota en las cuales se pronosticó una falla.

- ¿Qué **decisiones** se toman con las predicciones del sistema?
  A partir de las predicciones del sistema, si el mismo arroja una posible falla en el APS, el personal del Taller programará la revisión de un camión.

- ¿Cuáles son los costos ocultos en la toma de decisiones como los **human-in-the-loop**?
  No hay costos ocultos en la toma de decisiones.

Bloque 7: Realizando predicciones

Este bloque nos permite saber cuando hacer las predicciones basadas en nuevas entradas.

- ¿**Cuándo** deberían estar las predicciones disponibles?

En primer lugar, antes de iniciar un viaje necesariamente deben estar disponibles las predicciones, de manera tal que el usuario del Taller pueda introducir la patente del tractor en el sistema y corroborar si el mismo tiene fallas identificadas.
Por otra parte, también deben estar disponibles mientras que el camión está en operación, a fin de que a través del modelo predictivo pueda alertarse una posible falla a ocurrir.

- ¿Nuevas predicciones son hechas **a demanda**?
  No.

- ¿Nuevas predicciones son **calendarizadas**?
  No, serán realizadas en tiempo real.

- ¿Las predicciones son hechas **sobre la marcha** en **cada punto de datos** o para un **batch** de datos de entrada?
  Las predicciones son hechas sobre la marcha cada vez que se registra un dato de todos los sensores.

- ¿Hay un human-in-the-loop en la **salida** de estas predicciones?
  No.

- ¿Se dispone de **hardware** para predecir?
  No.

- ¿Se utiliza algún servicio de **Cloud** para predecir?
  Sí, al igual que las instancias de extracción y almacenamiento de los datos, para la predicción también se utilizaran los servicios de AWS, precisamente SageMaker Studio.

Bloque 8: Recolectando datos

Relacionado al punto anterior, esta sección recolecta información sobre **nuevos datos** que deben ser recolectados de manera de reentrenar el modelo.

- ¿Se dispone de datos para hacer un **entrenamiento inicial**?
  Sí, el set de entrenamiento es provisto por Scania.

- ¿Cómo **etiquetamos** los datos nuevos?

- ¿Hay que procesar datos **multimedia** de tipo imagen, sonido o video?
  No.

- ¿Hay un human-in-the-loop en la **limpieza manual** y el etiquetado de la **data entrante**?
  No, la limpieza estará incluida en el procesamiento previo de los datos y estará automatizado para cada registro nuevo que ingrese.

Bloque 9: Construyendo modelos

Relacionado a la sección anterior, pero esta vez referido a la **actualización** del modelo, dado que diferentes tareas requieren diferentes frecuencias de reentrenamiento:

- ¿Qué tan **seguido** debería ser reentrenado el modelo?
  El modelo se reentrena constantemente, a medida que ingresan nuevos puntos de datos, es decir, que los sensores arrojan nuevos valores.

- ¿Cómo vamos a lidiar con los asuntos de **escala** en la medida que la operación se vuelva más compleja y costosa?
  En tal caso, se evaluará la posibilidad de disminuir la frecuencia de reentrenamiento del modelo, sin la necesidad de tener que estar ejecutando las predicciones permanentemente. Un ejemplo podría ser, pasar de real time a batch, y hacer una predicción por día para cada camión.

- ¿Cuál es el **stack** de tecnologías usado?
  No entiendo a qué se refiere.

- ¿Qué hacemos si aparece un modelo que **supera ampliamente** el que estamos desarrollando?
  En ese caso, se analizará la posibilidad de implementar ese nuevo modelo, ya que el mantenimiento de la flota de camiones de la empresa podría verse claramente beneficiado.

Bloque 10: Monitoreo y evaluación en vivo

- ¿Cómo vamos a hacer seguimiento de la **performance** del sistema?
  Para hacer seguimiento de la performance del sistema, en la instancia de monitoreo, se controlará la fórmula (1) de costos. Cuanto menor sea el resultado de esta, querrá decir que el modelo está siendo más preciso, arrojando menos FP y/o FN.

- ¿Cómo evaluamos la **creación de valor**?
  La creación de valor podría medirse con la cantidad de averías de las unidades durante la operación, o también con la cantidad de veces que se revisa un camión por una supuesta falla que no existió.