Inteligencia de Negocios: Proyecto 1 etapa 2

Estudiantes

Alejandro Hoyos

Luis Alberto Pinilla

Julián Andrés Bolívar

**Sección 2. (15%) Proceso de automatización del proceso de preparación de datos,**

**construcción del modelo, persistencia del modelo y acceso por medio de API**

Sección 2 – Proceso e implementación.

El proceso de automatización se diseñó para permitir que el ciclo completo del modelo desde la preparación de los datos hasta su despliegue y reentrenamiento pueda ejecutarse de manera controlada y reproducible a través de una API REST desarrollada con **FastAPI**. Se compone de tres cpas principales, el preprocesamiento, entrenamiento/reentrenamiento, y servicios (en el api). Cada una de estas funcionalidades tiene un archivo responsable en la carpeta src del backend. En la que se responsabiliza a cada módulo de una tarea específica. Estas tareas se compactan en un pipeline que permita a la entrada de la información ser procesada de manera automática y ágil. El flujo consiste en preprocesar el texto, vectorizar la información con (BOW) y finalmente entrenar el clasificador. Así mismo se implementaron módulos encargados de las métricas y a el almacenamiento de data, modelos, documentos y logs.

Sección 2 – Investigación de tres métodos de reentrenamiento.

Cuando queremos utilizar un método de reentrenamiento en un modelo de machine learning debemos tomar en cuenta tres criterios fundamentales según las reglas del negocio, sus necesidades, su disponibilidad de recursos, la naturaleza de los datos que está manejando y según el caso. Para este caso tuvimos en cuenta los siguientes criterios: la facilidad de implementar el modelo de reentrenamiento, la escalabilidad según los datos, el riesgo o necesidad de tomar en cuenta datos viejos en el entrenamiento del modelo.

Batch retraining:

En esencia este método de reentrenamiento se basa esencialmente en la reconstrucción del modelo desde cero cada vez que se dispone de datos nuevos. En este caso se reincorporan los datos nuevos con los datos viejos y se vuelve a entrenar el modelo con los datos nuevos de esa manera. Este modelo asegura que el modelo considere toda la información disponible. Además es un método ampliamente usado, estable y compatible con la mayoría de algoritmos supervisados como Naive Bayes (que fue nuestro mejor modelo de la etapa anterior). El problema de este método radica en su exigencia computacional ya que entrenar los datos constantemente y periódicamente puede ser bastante costoso, en ese sentido, no es ideal si el negocio requiere actualizar datos en tiempo real o muy seguido.

Incremental Learning:

En esta estrategia, el modelo se actualiza de manera continua a medida que recibe nuevos datos, sin necesidad de reentrenar desde cero. Algunos algoritmos permiten realizar esta actualización mediante métodos como partial\_fit(), que ajustan gradualmente los parámetros del modelo. Una de las grandes ventajas es que permite entrenar los datos de manera más veloz y eficiente, ya que permite añadir especificad al modelo implementado sin incurrir en muchos costos computacionales extra. Además, es escalable para contextos donde los datos llegan en flujo (de manera constante). El problema de esta implementación, es que no todos los modelos permiten actualizaciones incrementales y además puede presentarse un fenómeno de “olvido” esto significa que el modelo deja de recordar patrones antiguos, al incorporar información nueva. Evitar ese problema implica un cuidadoso uso y manejo de los datos que se le incorporan.

Sliding Windown Retraining:

Este método en esencia mantiene una ventana con los datos más recientes, descartando los caos más antiguos a medida que se incorpora nuevos. El modelo se entrena de nuevo usando los datos que contiene esta ventana. Este modelo es súper flexible y permite adaptar el modelo a cambios recientes. Permite adaptar el modelo a cambios más recientes y además controla el tamaño de entrenamiento, de esta manera le permite al individuo controlar qué tantos datos se usan en qué momentos según la disponibilidad de recursos computacionales. El problema de esta estrategia es que al eliminar datos antiguos puede perder información relevantes para contextos actuales. Elegir el tamaño de la ventana puede afectar el rendimiento (velocidad del modelo) como también puede afectar su exactitud y desempeño.

Conclusiones:

En el contexto del proyecto, el modelo tiene como objetivo clasificar textos relaionados con los objetivos de desarrollo sostenible, esto implica que los temas a tratar van a ser iguales sin importar la época con a lo mejor pequeñas variaciones. Es decir, la novedad de los datos no es importante para la exactitud y aplicabilidad del modelo, podríamos entrenar los los modelos con datos actuales o con datos falsos y los resultados deberán dar similares. Puesto que es un código diseñado para la ONU, los recursos computacionales tampoco deberían ser un problema en caso en el que el modelo escale mucho. Además estos modelos no tienen que ser actualizados con mucha frecuencia, pues para el funcionamiento del negocio, no es muy importante que los datos sean los más recientes. Por estas razones decidimos implementar el algoritmo bajo la técnica de **batch retraining**.