

Prueba técnica Gestor SR con énfasis en IA.

Teoría:

1. Clasificación de Correos en la Empresa GA

Descripción del Problema: En la empresa GA, en el área de compras necesitan CLASIFICAR y organizar los correos que llegan a la bandeja de entrada entre 4 tipos de correos (Compras cementos, Compras energía, Compras concretos y correos generales o de otra índole). Esta tarea se le encomienda a usted, como es el Gestor SR en temas de analítica e IA puede solicitar al área interesada los recursos humanos que necesite para llevar a cabo este proyecto, también puede solicitar en tecnología todo lo que necesite, además tiene las bandejas de entrada de correos históricos de los analistas que reciben estas solicitudes con aproximadamente: 5500 correos de compras cementos, 2700 correos de compras de energía, 1100 correos de compras concretos y 12876 correos generales o de otra índole.

Metodología:

- **Recolección de Datos:** Obtener acceso a los correos históricos de los analistas.
- **Preprocesamiento de Datos:**
 - Limpieza de datos: Eliminar duplicados, correos irrelevantes y correos basura.
 - Tokenización: Descomponer los correos en palabras o tokens.
 - Filtrado de palabras comunes (stop words) y normalización de texto.
- **Exploración de Datos:**
 - Análisis exploratorio de datos, para entender la distribución de las categorías de correos.
 - Visualización de datos para identificar patrones y relaciones.
- **Modelado:**
 - **Algoritmos:** Considerar algoritmos de clasificación como Naive Bayes, Support Vector Machines (SVM), y modelos basados en redes neuronales (LSTM, BERT).
 - **Entrenamiento:** Dividir los datos en conjuntos de entrenamiento y prueba. Entrenar varios modelos y evaluar su rendimiento.
 - **Evaluación:** Utilizar métricas como precisión, recall, F1-score y la matriz de confusión para evaluar los modelos.
- **Implementación:**
 - Desplegar el modelo seleccionado en un entorno de producción.

- Integrar el modelo con la bandeja de entrada de correos para clasificar automáticamente los correos entrantes.
- **Arquitectura del Proyecto:**
 - **Entrada de Datos:** Servicio de ingesta de correos electrónicos (API o ETL).
 - **Preprocesamiento:** Scripts de limpieza y normalización de texto.
 - **Modelado:** Frameworks de machine learning (scikit-learn, TensorFlow, PyTorch).
 - **Implementación:** Despliegue en la nube (AWS, Azure, GCP) o servidores internos.

2. Identificación y Corrección de Drift en Modelos de Regresión

Descripción del Problema: Seis meses después de haber desplegado un modelo de regresión en producción, los usuarios se dan cuenta que las predicciones que este está dando no son tan acertadas, se le encarga a usted como Gestor SR en temas de IA que revise que puede estar sucediendo.

¿Cree que el modelo esté sufriendo Drift?

Sí, es posible que el modelo esté experimentando drift. El drift puede ocurrir cuando las características del entorno o los datos cambian con el tiempo, respecto a las iniciales entrenadas, afectando la precisión del modelo.

¿Cómo puede validarlo?

1. Monitoreo de Métricas:

- Revisar las métricas de rendimiento del modelo, desde su despliegue hasta la fecha actual.
- Comparar estas métricas con las obtenidas durante el entrenamiento y validación inicial del modelo.

2. Análisis de Datos:

- Analizar si las distribuciones de los datos de entrada han cambiado significativamente.

¿De ser así, qué haría usted para corregir esto?

➤ Recopilación de Nuevos Datos:

- Recopilar datos más recientes y relevantes que reflejen los cambios en el entorno.

➤ Reentrenamiento del Modelo:

- Reentrenar el modelo utilizando los nuevos datos recopilados.

- Implementar un pipeline de reentrenamiento automático si es necesario.

➤ **Actualización del Modelo:**

- Desplegar el nuevo modelo reentrenado en producción.
- Asegurarse de realizar validaciones periódicas para prevenir futuros problemas de drift.

3. Implementación de un Chatbot con GPT-3.5 para Información Específica

Descripción del Problema: Su equipo de trabajo está trabajando en un chatbot con generación de texto utilizando el modelo GPT-3.5, según cómo funciona este modelo, ¿cómo haría usted para hacer que las respuestas del chatbot estén siempre relacionadas a conseguir cierta información particular del usuario y no empiece a generar texto aleatorio sobre cualquier tema?

Propuesta:

- **Prompt Engineering:**
 1. Utilizar técnicas de ingeniería de prompts para guiar al modelo GPT-3.5 a generar respuestas relevantes.
 2. Incluir instrucciones explícitas en los prompts para que el modelo entienda el contexto y la finalidad de la conversación.
- **Entrenamiento Adicional:**
 1. Realizar un fine-tuning en GPT-3.5 con un conjunto de datos específicos relacionados con el tema de interés.
- **Estrategias de Configuración Hiper-parámetros del Modelo**
 1. Temperatura Baja: La temperatura controla la aleatoriedad de las respuestas generadas por el modelo. Para hacer que las respuestas sean más deterministas y menos creativas.
 2. Max Tokens: Controlar la longitud máxima de las respuestas generadas. Limitar el número de tokens generados a un valor razonable que cubra la información necesaria sin irse por ramas.
 3. Validación de otros parámetros del modelo.