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.

1. **Definición del Problema:**
   El objetivo es desarrollar un sistema de clasificación automatizada de correos electrónicos en cuatro categorías:

   - Compras Cementos
   - Compras Energía
   - Compras Concretos
   - Correos Generales (otros)

   Dado que se cuenta con un dataset histórico etiquetado, se puede abordar este requerimiento como un problema de aprendizaje supervisado (clasificación multiclase).

2. **Metodología (CRISP-DM)**
   Se podría seguior la metodología CRISP-DM (Cross-Industry Standard Process for Data Mining), que consta de:

   - Alinear el modelo con las necesidades del área de compras.
   - Un análisis exploratorio (EDA) para identificar patrones en los correos (texto, metadatos, etc.).
   - Limpieza, tokenización, vectorización y balanceo (si es necesario).
   - Selección y entrenamiento de algoritmos de NLP.
   - Validación con métricas de clasificación.
   - Integración en la bandeja de entrada (ej: API + automatización con Power Automate o solución similar).

3. **Pipeline de Procesamiento de Datos**
   - Eliminación de stopwords, URLs, caracteres especiales.
   - Lematización/Stemming).
   - Detección de entidades como empresas, materiales y números de pedido.
   - TF-IDF: Para capturar importancia de términos.
   - Word Embeddings o modelos contextuales (BERT) si se requiere mayor precisión.
   - Técnicas como oversampling (SMOTE) o weighted loss en el modelo.

4. **Algoritmos y Modelos Propuestos:**
   Se pueden evaluar los siguientes modelos:

Modelo	Ventajas	Desventajas
Regresión Logística	Interpretable, buen baseline.	Limitado con texto no lineal.
Random Forest	Maneja bien features no lineales.	Menos eficiente en NLP.
SVM	Efectivo en espacios de alta dimensión.	Costoso computacionalmente.
XGBoost/LightGBM	Buen rendimiento en datos estructurados.	Requiere fine-tuning.
Redes Neuronales (LSTM)	Captura contexto secuencial.	Necesita más datos y recursos.
Transformers (BERT, DistilBERT)	Máximo rendimiento.	Requiere GPU y mayor complejidad.

5. **Arquitectura del Proyecto**

- Lenguaje: Python (scikit-learn, Transformers).
- Plataforma Cloud: Azure (servicios clave abajo).
- Pipeline en Azure
- Ingestión:
  - Azure Logic Apps (monitoreo de bandeja de entrada en Exchange Online).
- Preprocesamiento:
  - Azure Functions (serverless, limpieza de texto y vectorización).
- Modelado/Inferencia:
  - Azure Machine Learning (entrenamiento con AutoML o BERT en GPU, despliegue como API).
- Almacenamiento:
  - Azure SQL DB (registro de predicciones para auditoría).
- Automatización:
  - Power Automate (clasificación en subcarpetas según etiqueta).

6. **Evaluación y Validación**
   - Accuracy
   - F1-Score
   - Matriz de Confusión
   - Validación Cruzada
   - A/B Testing

7. **Riesgos y Mitigación**

- Sesgo en Datos: Si los correos históricos no son representativos, se solicitará muestreo adicional.

- Cambio de Distribución: Implementar drift detection para monitorear cambios en los correos entrantes.
- Privacidad: Anonimización de datos sensibles.

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.

⇒ **¿Puede ser Drift?**

- Sí, es probable. Dos tipos de drift pueden afectar el modelo:
    - Data Drift: Que se refiere a un cambio en la distribución de las variables de entrada.
    - Concept Drift: Que se refiere a un cambio en la relación entre features y target.

⇒ **Validación del Drift**

- KS-test o Earth Mover's Distance para comparar distribuciones (datos actuales vs. entrenamiento).
- Alertas en desviaciones de métricas (MAE, RMSE) o drift en features clave.
- Segmentar datos recientes y evaluar performance por periodo.

⇒ **Corrección (si hay Drift)**

- Reentrenamiento:
    - Fine-tuning con datos recientes.
    - Nuevo entrenamiento con dataset actualizado.
- Adaptación en producción:
    - Desplegar nuevo modelo en paralelo antes de reemplazar el actual.
    - Liberación gradual a un subconjunto de usuarios.

⇒ **Causas Alternativas a Validar**

- Problemas en datos: Missing values, errores en pipelines ETL.
- Sesgo en nuevas poblaciones: Datos no vistos durante entrenamiento.
- Solución prioritaria: Implementar monitoreo automatizado de drift para prevenir futuros casos.

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?

Ha tener en cuenta los siguientes ítems.

⇒ **Prompt Engineering (Clave)**
- Hay que dar instrucciones claras y restrictivas en el *system prompt*:
  - Ejemplo: "Eres un asistente especializado en recopilar [información X]. Solo pregunta o responde sobre este tema. Si el usuario se desvía, redirígelo amablemente."
- Hay que instaurar ejemplos de interacciones válidas (*few-shot learning*) para guiar el comportamiento.

⇒ **Contexto Dinámico**
- Mantener un buffer de contexto con:
  - Objetivo actual "Recopilar datos de contacto".
  - Datos ya obtenidos (evita repeticiones).

⇒ **Post-Procesamiento**
- Validar respuestas con reglas de negocio (regex para correos/teléfonos).
- Si la respuesta no cumple con el tema, reiniciar la conversación o pedir clarificación.

⇒ **Fine-Tuning**
- Entrenar una versión customizada de GPT-3.5 con ejemplos de conversaciones ideales.

⇒ **Fallback Controlado**
- Si el chatbot no puede obtener la info requerida: dar una respuesta predefinida como "Por ahora solo puedo ayudarte con [tema X]. ¿Podrías compartir [datos en específico]?"