

## **PRUEBA TÉCNICA GESTOR SR CON ÉNFASIS EN IA.**

**AIRAM KATIZA COBO SOLIS**

### **TABLA DE CONTENIDO**

<b>RESPUESTA 1.</b>	<b>1</b>
<b>RESPUESTA 2:</b>	<b>3</b>
<b>RESPUESTA 3.</b>	<b>4</b>

### **TEORÍA**

1. 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.

Explique: ¿cómo resolvería este problema?, metodología, algoritmos, modelos, arquitectura del proyecto etc.

### **RESPUESTA 1.**

#### **Proyecto de Clasificación y Organización de Correos Electrónicos de la Empresa GA**

##### **Objetivo:**

Clasificar y ordenar los correos electrónicos que llegan a la bandeja de entrada en cuatro tipos de correo (Compras cementos, Compras energía, Compras concretos y Correos generales o de otra índole).

##### **Solución del problema:**

Para solucionar el problema de la clasificación y organización de correos en las 4 categorías, se propone el uso de la nube Azure, que iría desde el almacenamiento de los correos históricos, la limpieza, el modelo de entrenamiento, el testing, el despliegue del modelo y el mantenimiento de este.

## Herramientas y recursos:

### Recursos Humanos:

- Un analista de datos para interpretar patrones de clasificación.
- Un ingeniero de machine learning para entrenar el modelo de IA.
- Un especialista en Azure para integración y despliegue.

### Arquitectura:

- Azure Cognitive Services: Procesamiento de lenguaje natural para entender los correos.
- Azure Machine Learning: Entrenamiento de modelos personalizados para clasificación.
- Azure Key Vault: Para almacenar de forma segura las credenciales de la cuenta de correo.
- Azure Logic Apps: Automatización del flujo de correos.
- Azure Storage & SQL Databases: Para almacenar datos históricos y resultados clasificados.
- Power BI Service.

### Metodología

1. Preparación de Infraestructura y Seguridad.
  - Repositorio (en Azure DevOps).
  - Configurar un Azure Key Vault para almacenar credenciales del correo.
  - Configurar las carpetas en el correo, Crear 4 carpetas (“Compras cemento”, “Compras energía”, “Compras concreto” y “General/Otros”) que luego recibirán los correos clasificados.
2. Monitorear la Bandeja de Entrada.
  - Usar Azure Logic App para que esté en monitoreo continuo de la bandeja de entrada y notifique.
3. Análisis de Texto y Clasificación (NLP).
  - Extraer asunto y cuerpo del correo para analizar el contenido de este.
4. Almacenamiento de Resultados y Logs.
  - Usar base de datos relacional. Guardar en Azure SQL Database (o Azure Database for PostgreSQL/MySQL).
5. Movimiento y Organización de Correos.
  - Finalmente clasificar los correos en la carpeta correspondiente.
    - Compras cementos.
    - Compras energía.
    - Compras concretos.
    - Generales/Otras indoles.
6. Monitoreo y Mantenimiento Continuo.
  - Conectar Power BI a la Azure SQL Database para visualizar, con el fin de analizar los datos y analizar si el modelo esta cumpliendo con lo solicitado o necesita reentrenamiento.

### Modelo

- CLU (Conversational Language Understanding).
- Text Analytics Custom Classification (bajo la misma licencia de Azure Cognitive Services).



2. 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.

## RESPUESTA 2.

### ¿Cree que el modelo esté sufriendo Drift?

Si, ya que no se están ajustando como se esperaría y está perdiendo precisión, esto se puede dar debido a cambios entre las variables del modelo, ya que han pasado seis meses y las variables iniciales pueden cambiar debido a cambios en los usuarios, actualizaciones de los datos que no se han tenido en cuenta, puede que el modelo se haya creado con datos en un determinado tiempo y no se hayan realizado modificaciones, etc.

### ¿Cómo puede validarlo?

1. Validar que parámetros estadísticos han cambiado con el tiempo (Media, Mediana, Desviación estándar).
2. Comparar los datos actuales con los resultados de hace seis meses.
3. Evaluar las métricas de desempeño.

### ¿De ser así, que haría usted para corregir esto? Explique sus respuestas.

1. Obtener y recopilar nuevos datos.
2. Reentrenar el modelo con datos recientes.
3. Ajustar las variables ya que puede que algunas hayan perdido relevancia con el tiempo.

4. Crear alertas para que avise cuando algunos de esos parámetros se desvíe del umbral establecido.
  5. Implementar un plan de revisiones continuas que repita la validación, el reentrenamiento y despliegue.
- 3. 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?**

### RESPUESTA 3.

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, se recomiendan las siguientes estrategias:

1. primero se debe ir a la documentación del modelo GPT-3.5 con el fin de obtener información relevante del modelo, como, por ejemplo, el comportamiento por defecto del modelo, las mejores prácticas para realizar prompts y sus parámetros de configuración (temperature, max\_tokens, stop\_sequences). (<https://platform.openai.com/docs/models/gpt-3.5-turbo>)
2. En los modelos de IA es fundamental la manera en la que diseñas el Prompt, ya que permitirá que el modelo tenga respuestas claras y se cumplan con los objetivos. Para lograr esto se recomienda que los prompts tengan:
  - Instrucciones claras y breves.
  - Roles y contexto bien definidos.
  - Objetivos claros y 2 o 3 ejemplos que permitan reforzar los roles de comportamiento.
3. El modelo GPT-3.5 ofrece varios parámetros que regulan la respuesta, para evitar el texto aleatorio.
  - Fijar la temperature=0 esto reduce la aleatoriedad y enfoca al modelo a seguir el patrón que se le indico.
  - max\_tokens: Funciona para limitar la longitud, generando que el modelo no se extienda en explicaciones demasiado largas con contenidos innecesarios.
  - stop\_sequences: Hace que el modelo se detenga tras una sola pregunta, y espere a la respuesta del usuario sin generar texto extra.
4. Otra forma de evitar texto aleatorio es evaluar continuamente los resultados del modelo, después de cada prompt siempre debemos observar que las respuestas tengan coherencia con el objetivo, que cada respuesta tenga el nivel de detalle y claridad que el usuario necesita y, por último, llevar un registro de los prompts que generan este tipo de desviaciones con el fin de detectar patrones.
5. Si el modelo sigue presentando problemas por último se recomienda realizar un fine-tuning supervisado ya que permite adaptar un modelo de lenguaje pre-entrenado para que realice tareas de

una manera más efectiva según las necesidades. Este procediendo se encuentra en (<https://platform.openai.com/docs/guides/fine-tuning>) y tiene como ventajas:

- Introducir datos de entrenamiento personalizado.
- Ahorro de tokens y latencia.

#### **Pasos para un fine-tuning supervisado con GPT-3.5:**

**Paso 1: Recolección y preparar los datos:** Recopilar ejemplos reales donde el usuario y el asistente interactúen con claridad y siempre generando preguntas predefinidas.

**Paso 2: Subir Archivos a la OpenIA:** Subir los archivos .jsonl a la cuenta.

**Paso 3: Crear un trabajo de fine-tuning:** Una vez los archivos se suben a la OpenIA esta generara un nuevo modelo.

**Paso 4: Usar un modelo fine-tuneado:** Una vez el fine-tune finalice, obtén el ID del modelo entrenado y con este modelo “preajustado” ya podrás realizar las preguntas específicas que tú diseñaste.

**Paso 5. Evaluar los resultados:** Prueba la nueva versión con casos reales. Si observas que el chatbot aún se sale del guion, aumenta el número de ejemplo y revisa con claridad los prompts que se diseñaron.

Con la combinación de prompts bien estructurados, parámetros ajustados y un fine-tuning, puedes asegurar que el modelo GPT-3.5 se mantenga enfocado en recopilar información específica y evitas que se desvíe a temas aleatorios.

**Dimensiones dataset, modelos evaluados, validación cruzada, optimización de hiperparámetros, evaluación del modelo con los datos de prueba.**