Suplemento de Estudio Avanzado: AWS AI Practitioner

IA Responsable, Arquitecturas de Transformadores y MLOps

Material Basado en Guías Oficiales de AWS

1. Uso Responsable de la Inteligencia Artificial

AWS define la IA Responsable como un área en constante evolución que debe integrarse en todo el ciclo de vida del sistema. No es una solución final, sino un proceso continuo.

1.1. Las Tres Fases del Ciclo de Vida Responsable

- **Fase 1: Diseño y Desarrollo:**
  - **Evaluación de impacto:** Identificar si el sistema afecta derechos humanos o seguridad (ej. vehículos autónomos vs. recomendación de películas).
  - **Equipos Diversos:** Incluir variedad de géneros, razas, edades y disciplinas (técnicos, abogados, expertos en ética) para evitar sesgos desde el origen.
  - **Explicabilidad:** Evaluar la necesidad de entender *cómo* el modelo llega a una conclusión, especialmente en casos de alto riesgo.
- **Fase 2: Despliegue:**
  - **Aviso y Transparencia:** Informar a los usuarios cuando interactúan con una IA (ej. chatbots).
  - **Educación del Usuario:** Capacitar a los revisores humanos sobre las limitaciones del sistema y el significado de los puntajes de confianza.
- **Fase 3: Operación:**
  - **Retroalimentación:** Implementar mecanismos para que los usuarios reporten errores o resultados injustos.
  - **Validación Continua:** Vigilar el *Concept Drift* (cambio en el comportamiento del modelo por cambios en el entorno).

1.2. Mitigación de Sesgos (Bias)

El sesgo ocurre cuando los resultados están sesgados a favor o en contra de un grupo.

- **Causas:** Datos insuficientes, ruidosos o que no representan el mundo real.

- **Técnicas de Balance:** Uso de *under-sampling* (quitar datos de la clase mayoritaria) o *over-sampling* (duplicar datos de la minoritaria).
- **Human-in-the-loop:** Integrar la revisión humana mediante **Amazon Augmented AI (A2I)** para auditar predicciones de baja confianza.

2. Arquitectura de Transformadores

Los transformadores superaron a las Redes Neuronales Recurrentes (RNN) al permitir el procesamiento en paralelo y gestionar dependencias de largo alcance en el texto.

2.1. Mecanismo de Autoatención (Self-Attention)

Es el núcleo del transformador. Permite que el modelo analice toda la secuencia simultáneamente y asigne ïmportancia.ª diferentes palabras según su contexto, sin importar su posición.

2.2. Componentes Técnicos

- **Embeddings (Incrustaciones):** Conversión de tokens en vectores numéricos en un espacio n-dimensional. Palabras con significados similares quedan çerca"matemáticamente.
- **Codificación Posicional:** Como el transformador procesa todo en paralelo, necesita señales matemáticas adicionales para entender el orden de las palabras.
- **Softmax:** Capa final que convierte los puntajes internos (*logits*) en una distribución de probabilidad (0 a 1) para elegir la palabra más probable.

3. Modelos de Lenguaje de Gran Tamaño (LLM)

Los LLMs son modelos de aprendizaje profundo con miles de millones de parámetros entrenados con datos masivos de Internet (Wikipedia, Common Crawl).

3.1. Tipos de Modelos

- **BERT:** Bidireccional. Útil para entender el contexto completo de una oración.
- **GPT:** Autorregresivo. Predice la siguiente palabra en una secuencia.
- **ViT (Vision Transformers):** Aplica la lógica de los transformadores a imágenes, dividiéndolas en "parchesçomo si fueran palabras.

4. Sobreajuste (Overfitting) y Subajuste (Underfitting)

- **Sobreajuste:** El modelo aprende el ruido"de los datos de entrenamiento. Funciona bien con datos conocidos, pero falla con datos nuevos.

    - **Solución:** Regularización, poda (*pruning*), detención temprana y aumento de datos.

- **Subajuste:** El modelo es demasiado simple para aprender la tendencia. Error alto en entrenamiento y prueba.
- **Validación Cruzada (k-fold):** Técnica para detectar sobreajuste dividiendo los datos en k grupos y probando el modelo en cada uno de ellos de forma rotativa.

5. MLOps: Operaciones de Machine Learning

MLOps une el desarrollo de modelos con la operación del sistema, tratando los activos de ML como software tradicional (CI/CD).

- **Nivel 0 (Manual):** Todo es manual, desde la preparación de datos hasta el despliegue. Desconexión entre científicos de datos e ingenieros.
- **Nivel 1 (Automatización de Pipeline):** El entrenamiento es continuo. Se despliega una canalización que re-entrena el modelo automáticamente con nuevos datos.
- **Nivel 2 (CI/CD Completo):** Ideal para empresas tecnológicas. Automatiza la creación, prueba y despliegue de múltiples canalizaciones de forma recurrente.