**1. ¿Cuáles son las diferencias fundamentales entre los modelos encoder-only, decoder-only y encoder-decoder en el contexto de los chatbots conversacionales? Explique qué tipo de modelo sería más adecuado para cada caso de uso y por qué.**

**Diferencias fundamentales:**

* **Encoder-only**: Este tipo de modelo solo se utiliza para tareas de comprensión. El encoder toma una entrada (como un texto) y la codifica en una representación interna de alta dimensión. Los modelos como BERT (Bidirectional Encoder Representations from Transformers) son ejemplos de modelos encoder-only. Estos modelos son excelentes para tareas de clasificación y etiquetado de secuencias, pero no son ideales para generar texto. En chatbots, este tipo de modelo no genera respuestas, sino que puede usarse para entender la intención de la consulta del usuario.
* **Decoder-only**: En este tipo de modelo, solo se utiliza la parte del decoder del transformer para generar texto a partir de un contexto dado. El modelo toma una secuencia de tokens (en un formato autoregresivo) y predice el siguiente token de la secuencia. GPT (Generative Pretrained Transformer) es un ejemplo típico de modelo decoder-only. Este tipo de modelo es adecuado para tareas de generación de texto, como en los chatbots conversacionales, donde el modelo necesita generar una respuesta a partir de un prompt o contexto dado.
* **Encoder-decoder**: Este tipo de modelo combina tanto el encoder como el decoder. Se usa principalmente en tareas de traducción, resumen o cualquier tarea donde se necesite tanto comprender como generar texto. Modelos como T5 (Text-to-Text Transfer Transformer) o BART son ejemplos de modelos encoder-decoder. En estos modelos, el encoder procesa la entrada (como una pregunta del usuario) y el decoder genera una respuesta basada en la representación codificada. Estos modelos son útiles cuando hay un mapeo claro entre la entrada y la salida (por ejemplo, traducción o generación de resúmenes).

**Adecuación para casos de uso:**

* **Encoder-only**: Se utilizaría cuando el objetivo es **comprender el contexto** o la intención del usuario sin necesidad de generar texto, como en sistemas de clasificación de preguntas o chatbots que simplemente clasifican la intención de un mensaje.
* **Decoder-only**: Es el modelo más adecuado para **generación de texto** como en los chatbots conversacionales. Al usar un modelo como GPT, el chatbot puede generar respuestas contextualmente coherentes y fluidas a partir de un prompt dado. Ideal para sistemas de conversación en tiempo real, generación de respuestas en un diálogo.
* **Encoder-decoder**: Este modelo es más adecuado para tareas **de entrada y salida transformacional**, como traducción automática o resúmenes. Aunque también puede ser útil en algunos chatbots que requieren comprensión profunda y generación de texto, su uso es más específico para tareas donde la entrada no es solo un texto para clasificar, sino que necesita ser transformada antes de generar una salida.

**2. Explique el concepto de "temperatura" en la generación de texto con LLMs. ¿Cómo afecta al comportamiento del chatbot y qué consideraciones debemos tener al ajustar este parámetro para diferentes aplicaciones?**

**Concepto de temperatura:**  
La **temperatura** es un parámetro utilizado en la generación de texto con modelos de lenguaje (LLMs) que controla la **aleatoriedad** de las predicciones del modelo. En un modelo autoregresivo, como GPT, durante la generación de cada palabra, el modelo calcula una distribución de probabilidad sobre todas las palabras posibles que podrían seguir, y la temperatura ajusta esa distribución.

* **Baja temperatura (cerca de 0)**: El modelo tiende a generar respuestas más deterministas, seleccionando las palabras más probables en cada paso. Esto produce respuestas más predecibles y coherentes, pero puede ser monótono o repetitivo.
* **Alta temperatura (por ejemplo, 1.0 o más)**: El modelo se vuelve más **aleatorio** en sus elecciones, lo que permite una mayor diversidad en las respuestas generadas. Esto puede hacer que las respuestas sean más creativas y variadas, pero también puede aumentar la probabilidad de generar respuestas sin sentido o menos coherentes.

**Impacto en el comportamiento del chatbot**:

* Con **temperatura baja**, el chatbot generará respuestas más predecibles y coherentes, lo que es ideal para aplicaciones donde se requiere alta precisión y consistencia, como en tareas de atención al cliente o soporte técnico.
* Con **temperatura alta**, el chatbot se vuelve más creativo, lo que puede ser útil en escenarios donde se desea una mayor **variedad** de respuestas, como en generadores de texto, chatbots de entretenimiento o aplicaciones creativas.

**Consideraciones al ajustar la temperatura**:

* Para aplicaciones que requieren precisión y coherencia, como servicios de soporte, se debe **reducir la temperatura** para que las respuestas sean más predecibles y orientadas a hechos.
* Para chatbots de entretenimiento o aquellos que buscan generar conversaciones más fluidas y variadas, se puede **aumentar la temperatura**, lo que permitirá que el modelo genere respuestas más creativas y dinámicas.

**3. Describa las técnicas principales para reducir el problema de "alucinaciones" en chatbots basados en LLMs. ¿Qué estrategias podemos implementar a nivel de inferencia y a nivel de prompt engineering para mejorar la precisión factual de las respuestas?**

**El problema de las "alucinaciones"**: En el contexto de LLMs, las **alucinaciones** son respuestas generadas por el modelo que son **incorrectas o inventadas**. Aunque el modelo puede generar texto de manera fluida y coherente, a menudo inventa hechos, citas o información que no está basada en datos reales.

**Técnicas para reducir las alucinaciones**:

1. **Uso de fuentes externas o conocimiento actualizado**:
   * **Integración con bases de datos externas**: Usar sistemas de recuperación de información (retrieval-based systems) donde el chatbot consulta una base de conocimiento antes de generar la respuesta. Esto asegura que la respuesta esté basada en hechos reales y verificables.
   * **Incorporación de mecanismos de búsqueda**: Permitir que el modelo consulte en tiempo real fuentes externas (como artículos, bases de datos o páginas web) para obtener información precisa y actualizada.
2. **Fine-tuning con datos específicos**:
   * **Entrenamiento en un conjunto de datos específico**: Realizar fine-tuning del modelo en un conjunto de datos curado que esté enfocado en el dominio de interés (por ejemplo, un modelo entrenado solo con información científica, médica, etc.). Esto reduce las alucinaciones, ya que el modelo será más confiable dentro de ese dominio específico.
3. **Mejora en el diseño de los prompts (prompt engineering)**:
   * **Instrucciones claras y restrictivas en el prompt**: Incluir instrucciones específicas en el prompt que orienten al modelo a ser preciso y limitar la generación de información falsa. Por ejemplo: "Responde solo con hechos verificables" o "Si no sabes la respuesta, di que no lo sabes".
   * **Refuerzo de contexto y validación**: Dar al modelo ejemplos específicos dentro del prompt de cómo debe generar respuestas basadas en hechos, y dar ejemplos de respuestas incorrectas o alucinaciones para que el modelo evite generarlas.
4. **Técnicas de control de la generación**:
   * **Reescritura de la salida con modelos adicionales**: Usar modelos adicionales para verificar o corregir la respuesta generada, por ejemplo, un modelo de clasificación que pueda identificar respuestas "falsas" o inexactas.
   * **Temperature y Top-p tuning**: Ajustar los parámetros de generación como la temperatura y el top-p para hacer que el modelo sea más determinista y menos propenso a generar información errónea o inventada.

**Estrategias para mejorar la precisión factual**:

* **A nivel de inferencia**: Asegurarse de que el modelo consulte fuentes externas actualizadas y utilice datos precisos en tiempo real antes de generar respuestas.
* **A nivel de prompt engineering**: Diseñar los prompts de manera que fomenten respuestas precisas y fundamentadas, y entrenar al modelo para manejar situaciones en las que no tiene suficiente información, pidiendo que se abstenga de generar respuestas especulativas.