Chatbots conversacionales inteligentes de texto

Conceptos básicos

- se entrenan
  - predecir palabra siguiente
  - secuencia de texto
- modelado de lenguaje
  - se basa en que
    - palabras
    - no se usan de forma aleatoria, sino que están relacionadas de manera predecible.
- grandes modelos de lenguaje (LLM)
  - se entrenan con
    - enormes conjuntos de datos
  - usan
    - redes neuronales profundas
- modelos conversacionales
  - modelos
  - preparados para conversar.
- chatbots de IA generativa
  - utilizan
  - (LLM) generar respuestas

Limitaciones

- Sesgos
  - pueden reflejar y perpetuar prejuicios,
- Alucinaciones
  - modelo genera información incorrecta
- Repeticiones
  - respuestas monótonas
- Capacidades Limitadas de Resolución de Problemas
  - dificultades
  - resolver problemas complejos que requieren pensamiento crítico.
- Comprensión del Contexto Limitada
  - dificultades mantener
  - contexto conversación prolongada o compleja.
- Ambigüedad e Interpretación
  - bots pueden luchar con
  - preguntas ambiguas
  - frases
  - múltiples significados.
- Dependencia de Entradas de Calidad
  - bots son tan buenos como datos con
  - que han sido entrenados.

Etapas para crear un modelo de IA

- •Etapa 1: preparar los datos.
  - 1. Se describen
  - 2. Se filtran
  - 3. eliminan duplicados.
- Etapa 2: entrenar el modelo.
  - 1. Elegimos arquitectura
  - 2. Se empareja pila de datos
  - 3. Tokenizamos pila de datos.
  - 4. Se inicia el proceso de entrenamiento
  - 5. Durante entrenamiento, modelo aprende cómo tokens
    - se relacionan
- Fase 3: Validación.
  - Evaluar el modelo
    - ejecutar modelo y comprobar rendimiento
  - métricas especializadas
    - Medir capacidad
      - predecir muestra de texto
    - Medir similitud
      - texto generado y referencia.
    - Medir porcentaje
      - predicciones que coinciden con referencia.
    - Medir calidad
      - traducciones automáticas.
- Etapa 4: Puesta a punto.
  - desarrollador se involucra
    - dar instrucciones provoquen buen rendimiento
    - proporcionar datos locales
  - ajuste fino
    - adaptarlo tarea específica
  - adaptación nuevos datos
    - mantener modelo actualizado
    - información reciente
- Etapa 5: Despliegue del modelo:
  - desplegarse
    - nube pública
    - aplicación o servicio
  - balance de carga,
    - sistema pueda manejar solicitudes simultáneas.
  - optimizar los recursos
    - respuesta sea eficiente.

Componentes de un chatbot

- Interfaz del usuario:
- Motor de PLN :
  - preprocesamiento de texto:
    - divide texto en tokens,
    - remueve stop words,
    - maneja puntuación,
    - prepara el input
  - Reconoce la intención
  - reconocimiento de entidades:
    - extrae entidades claves
    - genera texto
- Modelo grande de lenguaje:
  - comprende y mantiene el contexto.
- Componente de gestión del diálogo:
  - crucial para
  - conversaciones coherentes
- Integración con aplicaciones:
  - modelo se accede a través de APIs

gestión del diálogo

- involucra:
  - historia de la conversación
  - estado del diálogo actual
  - gestión de diálogo
  - aprendizaje de políticas

Diferencias

- (MLPG)
  - no diseñado para
    - interacciones conversacionales.
  - carece de
    - gestión de diálogo integrada:
  - Cada prompt
    - se trata de forma independiente,
- IA conversacional
  - Incorpora
    - estrategias gestión del diálogo
    - para mantener
      - contexto
      - coherencia
  - puede
    - seguir estado conversación,
    - reconocer
      - intenciones usuario
    - gestionar el contexto

Arquitectura de transformers

- permite
  - procesamiento paralelo datos
- especialmente efectiva para
  - (NLP).
- mecanismos de autoatención
  - permiten
    - enfocarse diferentes partes entrada,
  - mejora
    - comprensión del contexto
    - relaciones entre palabras.
- componentes
  - codificador
    - procesa entrada
  - decodificador
    - genera salida basada en representación
    - codificador.
- genera texto
  - prediciendo una palabra a la vez,
  - usando contexto
    - palabras anteriores.

Entrenamiento de un transformer

- aprendizaje supervisado
  - Entrada:
    - Se proporciona oración pero oculta la última palabra.
  - Predicción:
    - predecir palabra que falta
  - Comparación:
    - predicción se compara con palabra real
  - Cálculo de Error:
    - Se calcula diferencia predicción y la palabra real.
  - Retropropagación:
    - error se utiliza ajustar
    - pesos del modelo
- patrones aprendidos
  - durante entrenamiento son fundamentales
  - predecir siguiente palabra
  - patrones aprendidos
    - no se almacenan en
    - base de datos en cambio, se reflejan en parámetros del modelo.

Fases del decodificador

- Entrada Inicial:
- Cálculo de Autoatención:
- Atención Cruzada:
- Transformación a través de la Capa Feed-Forward:
- Generación Final:
  - ocurre capa final decodificador,
    - generación de la siguiente palabra
  - es seleccionado como siguiente palabra.
    - token mayor probabilidad

Capas del decodificador

- autoatención:
  - preste atención palabras generadas previamente
    - permite que
- atención cruzada:
  - permite acceder representación generada por codificador.
- alimentación directa:
  - transforma salida capas anteriores utilizando red neuronal
  - Se usa para
    - enriquecer representaciones antes
    - de siguiente capa.
- Codificación posicional:
  - permitir decodificador comprenda orden palabras generadas.
- Capa lineal:
  - representación final contexto generada
    - toma
  - puntuación cada palabra posible
    - la transforma en
  - palabra mayor puntuación.
    - Elige entre ellas

El decodificador

- permiten
  - generar salidas a partir de representaciones contextuales codificadores.

Fases del codificador

- Entrada de Texto:
  - texto es tokenizado
  - Cada token
    - se convierte vector numérico
    - (embedding).
- Aplicación de codificación posicional:
  - agrega información sobre posición cada palabra
- Cálculo de autoatención:
  - evalúe relevancia cada palabra respecto a las demás
  - asigna diferentes pesos a
    - palabras según importancia contextual.
- La capa de alimentación directa:
  - transforma
    - salida capa autoatención
    - para mejorar capacidad aprender patrones complejos.
  - Se genera
    - nueva representación cada palabra, enriqueciendo contexto.

codificador

- permiten
  - procesar
  - entender
  - secuencias de texto.
- a partir del texto
  - generará representaciones contextuales que capturan
    - significado
    - relaciones entre las palabras.
- resultado de un codificador
  - serie de vectores
  - representan cada palabra del prompt.
- está compuesto por
  - capas apiladas,
  - funciones específicas.