



INSTITUTO TECNOLÓGICO DE CULIACÁN

Ingeniería en Sistemas Computacionales

Inteligencia artificial

Hora: 09:00 am – 10:00 am

Tarea:

Proceso de aprendizaje automático

Equipo:

Peña López Miguel Ángel
Robles Rios Jacquelin

Docente:

Zuriel Dathan Mora Félix

Culiacán, Sinaloa

05/03/2024

Paradigmas de la inteligencia artificial

Paradigma basado en Conexión

Se basa en redes neuronales artificiales inspiradas en el cerebro humano. Este enfoque es capaz de aprender de datos y adaptarse a nuevas situaciones. Se ha utilizado con éxito en aplicaciones como el reconocimiento de imágenes y el procesamiento del lenguaje natural. Sin embargo, puede ser difícil entender cómo funcionan las redes neuronales.

Reconocimiento facial en redes sociales (Facebook usa redes neuronales para etiquetar personas en fotos automáticamente).

Paradigma Bioinspirado

Este paradigma toma ideas de la naturaleza y la biología para resolver problemas computacionales. Se basa en procesos como la evolución, la adaptación y el comportamiento colectivo de los seres vivos.

Optimización de rutas en algoritmos genéticos: Empresas como Amazon o FedEx usan técnica para mejorar sus sistemas de logística, También como ejemplo el comportamiento de bichos como las abejas, cuando hacen exploración una de ellas tiene un peso mas grande para encontrar una fuente de comida optima

Paradigma Simbólico

Este paradigma considera que la inteligencia puede representarse a través de símbolos y reglas lógicas. Se basa en la manipulación de estructuras de datos simbólicas mediante reglas predefinidas.

Ejemplo: MYCIN, este ayudaba a diagnosticar infecciones bacterianas, no sin antes haber recaudado información de expertos y sus diagnósticos para poder determinar si es una infección o no.

Paradigma Computacional

Es un enfoque fundamental en la IA que se basa en algoritmos y estructuras de datos para resolver problemas. Se utiliza en una amplia variedad de aplicaciones, como la planificación, el razonamiento y la búsqueda. Sin embargo, puede ser computacionalmente costoso para resolver problemas complejos.

Entre los ejemplos de planificación y búsqueda están como ejemplo el A* que es un algoritmo de búsqueda entre nodos de un árbol y seleccionando el que tenga un menor peso entre los demás que se agreguen a una cola, También podríamos decir que el algoritmo de dijkstra es otro que se encarga de determinar el mejor camino posible

desde un punto A, hacia un punto B, igual metiendo a los nodos vecinos de donde comienza y determinando su menor peso para determinar el camino con menor peso.

Paradigma computacional

El paradigma computacional en Inteligencia Artificial (IA) se basa en el uso de modelos matemáticos, estadísticos y algoritmos computacionales para resolver problemas y tomar decisiones. En lugar de depender de reglas explícitas o simulaciones biológicas, este enfoque utiliza cálculos numéricos y análisis de datos para aprender patrones, predecir resultados y optimizar procesos.

Principios del paradigma computacional

- ❖ Modelos matemáticos y estadísticos
- ❖ Aprendizaje a partir de datos
- ❖ Optimización y toma de decisiones
- ❖ Basado en estadística y probabilidad

Técnica	Descripción	Ejemplo
Redes Bayesianas	Modelos probabilísticos para la toma de decisiones en situaciones inciertas	Diagnostico medico basado en síntomas
Regresión Lineal/Logística	Modelos estadísticos para predecir valores o clasificar datos	Predicción de precios de viviendas
Arboles de decisión	Modelos que dividen datos en ramas basadas en condiciones lógicas	Sistemas de recomendación (Netflix, Amazon)
Máquinas de soporte vectorial	Algoritmo que encuentra la mejor separación entre categoría de datos	Clasificación de correos electrónicos (spam/no spam)
Algoritmos de optimización	Métodos para encontrar soluciones optimas en problemas complejos	Algoritmo de Dijkstra para calcular rutas eficientes en Google maps

Beneficios:

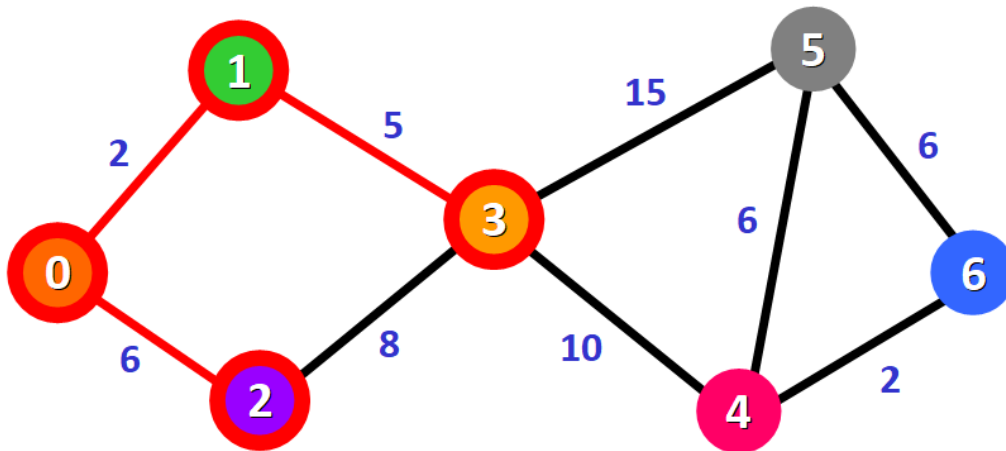
- ❖ Puede manejar grandes volúmenes de datos con alta precisión
- ❖ Se basa en modelos matemáticos sólidos, lo que permite realizar predicciones confiables
- ❖ Es utilizado en aplicaciones del mundo real, como la optimización logística y la inteligencia empresarial

Limitaciones:

- ❖ Requiere un gran poder computacional para procesar grandes cantidades de datos

- ❖ En algunos casos, los modelos pueden ser difíciles de interpretar (Redes neuronales)
- ❖ La calidad de los resultados depende en gran medida de la calidad de los datos utilizados para el entrenamiento

Ejemplo: El algoritmo Dijkstra



El de Dijkstra es el algoritmo de referencia para encontrar el camino más corto entre dos puntos de una red, que tiene muchas aplicaciones. Es fundamental en informática y teoría de grafos

Esta habilidad es transferible a otros algoritmos de optimización. Puede que el algoritmo de Dijkstra no sea el más eficiente en todos los escenarios, pero puede ser una buena referencia para resolver problemas de "distancia más corta".

Algunos ejemplos son:

- Los sistemas de navegación GPS encuentran la ruta más rápida
- Enrutamiento de paquetes de datos en redes informáticas
- Los servicios de reparto optimizan las rutas para ser más eficientes
- Redes sociales (sugerir conexiones)
- Finanzas (encontrar vías óptimas de inversión)
- Gestión de proyectos (encontrar el flujo de trabajo más eficaz)

Paradigma simbolico

El paradigma simbólico propone que la mente es un sistema de procesamiento compuesto por un conjunto de estructuras organizadas que operan siguiendo reglas precisas para tratar y transformar la información.

Se centra en el procesamiento y la manipulación de símbolos o conceptos, en lugar de datos numéricos. El objetivo de la IA simbólica es construir sistemas inteligentes que puedan razonar y pensar como los humanos, representar y manipular el conocimiento y el razonamiento en función de reglas lógicas.

Beneficios:

- Interpretabilidad: la IA simbólica proporciona transparencia en el proceso de razonamiento, lo cual facilita la comprensión de cómo un sistema ha llegado a una conclusión.
- Representación del conocimiento: la IA simbólica puede representar conocimientos complejos de manera formal y estructurada, lo que permite manipularlos y razonar con facilidad.
- Flexibilidad: la IA simbólica es muy flexible y puede adaptarse a distintos ámbitos al modificar las reglas y la base de conocimientos.

Limitaciones:

- Conocimiento incompleto: la IA simbólica requiere un conocimiento completo y bien definido para funcionar correctamente. En dominios en los que el conocimiento es incompleto, la IA simbólica puede no ser eficaz.
- Escalabilidad: la IA simbólica puede volverse costosa computacionalmente a medida que aumenta el número de símbolos y reglas, lo que dificulta su ampliación a grandes dominios.
- Dificultad para manejar información incierta o ambigua: la IA simbólica se basa en representaciones precisas e inequívocas del conocimiento, lo que limita su capacidad de razonar eficazmente con datos inciertos o ambiguos.
- Capacidad limitada para aprender y adaptarse: los sistemas simbólicos de IA requieren programación manual y pueden no permitir el aprendizaje y la adaptación en tiempo real

¿Cómo funciona?

Funciona mediante el procesamiento de símbolos, que representan objetos o conceptos del mundo, y sus relaciones. El enfoque principal de la IA simbólica es utilizar la programación basada en la lógica, en la que se utilizan reglas y axiomas para hacer inferencias y deducciones.

Ejemplo: Diagnostico de afecciones médicas en función de los síntomas

Los sistemas expertos médicos son programas que ayudan a los doctores a diagnosticar enfermedades en función de los síntomas ingresados por el paciente. Estos sistemas utilizan una base de conocimientos con reglas predefinidas para sugerir posibles enfermedades y recomendar tratamientos.

El paradigma simbólico se basa en el uso de reglas lógicas y manipulación de símbolos para la toma de decisiones. En este caso, el sistema experto utiliza reglas del tipo "SI-ENTONCES" para analizar los síntomas.

Enlace al ejemplo en prolog: <https://swish.swi-prolog.org/p/SistemaM.pl>

```
% Posibles diagnósticos
hipotesis(resfriado) :- resfriado.
hipotesis(gripe) :- gripe.
hipotesis(faringitis) :- faringitis.
hipotesis(sinusitis) :- sinusitis.
hipotesis(desconocido).

% Sin diagnóstico

% Reglas para cada enfermedad
resfriado :-
    verificar(dolor_cabeza),
    verificar(nariz_congestionada),
    verificar(estornudos),
    verificar(dolor_garganta).

gripe :-
    verificar(fiebre),

faringitis :-
    verificar(dolor_garganta),
    verificar(dificultad_para_tragar),
    verificar(fiebre),
    verificar(ronquera).

sinusitis :-
    verificar(dolor_cabeza),
    verificar(congestion_nasal),
    verificar(dolor_facial),
    verificar(secrecion_nasal_espesa).
```

Descripción:

El programa sigue una serie de pasos estructurados:

1. **El usuario responde preguntas** sobre síntomas con "si." o "no."
2. **El sistema verifica los síntomas** y determina la enfermedad más probable.
3. **Se muestra el diagnóstico y una serie de recomendaciones** médicas.
4. **El sistema limpia la memoria** de respuestas para futuras consultas.

¿Qué es el aprendizaje automático?

El aprendizaje automático (ML) es un subconjunto de inteligencia artificial (AI) que permite que los sistemas aprendan de los datos sin ser explícitamente programados. En lugar de depender de la programación basada en reglas, los algoritmos de ML detectan patrones en los datos y toman predicciones o decisiones basadas en datos. ML es cada vez más crucial en varias industrias debido a su capacidad para analizar grandes conjuntos de datos, identificar patrones y tomar predicciones o decisiones con mayor precisión.

Los procesos de aprendizaje automático se han convertido en una parte importante de MLOps. Al seguir un proceso de aprendizaje automático bien definido, las organizaciones pueden reducir el tiempo de lanzamiento al mercado y garantizar la confiabilidad y escalabilidad de sus soluciones de AI.

¿Qué es un proceso de aprendizaje automático?

Un proceso de ML es una secuencia de pasos interconectados que transforman los datos sin procesar en modelos de ML capacitados e implementados. Cada paso del proceso realiza una tarea específica, como el preprocesamiento de datos, la ingeniería de características, la capacitación de modelos, la evaluación, la implementación y el mantenimiento. El resultado de un paso sirve como aporte al siguiente, creando un flujo de trabajo optimizado para desarrollar e implementar modelos de aprendizaje automático.

Etapas del proceso de aprendizaje automático

El proceso de aprendizaje automático abarcan varias etapas, cada una de las cuales desempeña un papel fundamental en la transformación de datos sin procesar en un modelo de aprendizaje automático, estas etapas son la siguientes:

1. Adquisición o ingesta de datos

La adquisición de datos La adquisición de datos es el primer paso en el proceso de aprendizaje automático. Se refiere al acto de reunir y recopilar información relevante de diversas fuentes. Las fuentes de datos pueden incluir bases de datos empresariales, archivos CSV o JSON, APIs de servicios web, sensores IoT, redes sociales, encuestas, registros históricos, entre otros.

Dependiendo del caso de uso, la adquisición de datos puede ser un proceso continuo (flujo de datos en tiempo real) o un procedimiento puntual para obtener un conjunto de datos estático.

Uno de los aspectos clave en esta fase es garantizar que los datos recopilados sean relevantes, completos y de alta calidad, ya que la precisión del modelo de aprendizaje automático dependerá en gran medida de la calidad de los datos con los que se entrene.

2. Preprocesamiento de datos

Antes de entrenar un modelo, los datos deben someterse a una serie de procesos para asegurar su calidad y utilidad. El **preprocesamiento de datos** abarca tareas clave como la **limpieza, transformación y normalización**, garantizando que la información esté en un formato adecuado para el análisis y modelado.

- Limpieza de datos

Consiste en identificar y corregir errores, inconsistencias y valores faltantes en los conjuntos de datos, ya que datos incorrectos pueden afectar negativamente el rendimiento del modelo.

- Transformación y normalización

Los datos sin procesar se formatean y convierten en una estructura accesible para el algoritmo de aprendizaje automático. Además, este paso ayuda a manejar valores atípicos e inconsistencias, mejorando la estabilidad del modelo.

3. Entrenamiento del modelo

El entrenamiento del modelo es el proceso en el que un modelo ajusta sus parámetros internos para aprender patrones a partir de un conjunto de datos de entrenamiento. Durante este proceso, el modelo mejora su capacidad para hacer predicciones precisas minimizando el error entre las predicciones y los valores reales. El objetivo es desarrollar un modelo que generalice bien y sea capaz de hacer predicciones precisas sobre datos no vistos.

4. Evaluación del modelo

Una vez que la versión del modelo está lista, es hora de evaluar varias métricas para asegurarnos de que está preparado para la producción. Probaremos el modelo con un conjunto de datos de prueba e involucraremos a expertos para detectar errores en las predicciones. Para poder evaluar un modelo correctamente, separe una muestra de datos que han sido etiquetados con el destino (dato real) de la fuente de datos de entrenamiento. También verificaremos que cumplimos con los marcos éticos, industriales y legales al desarrollar soluciones de IA.

Finalmente, compararemos los resultados con las métricas de éxito esperadas y decidiremos si implementar el modelo, registrando y versionando cada paso para garantizar calidad y reproducibilidad.

5. Implementación del modelo

La implementación del modelo constituye el paso donde se materializa el valor del aprendizaje automático. Esto implica la integración del modelo en sistemas de producción, el desarrollo de APIs o interfaces para consumir las predicciones, el monitoreo continuo del rendimiento y la actualización periódica con nuevos datos. Esta fase requiere consideraciones de escalabilidad, mantenimiento y gobernanza de modelos para garantizar su correcto funcionamiento a largo plazo

Conclusión

Implementar sistemas de aprendizaje automático requiere mucho más que solo entrenar modelos. Necesita de un proceso que abarca desde la obtención y procesamiento inicial de los datos hasta la evaluación, puesta en producción y su mantenimiento después de ello.

Cada fase resulta esencial para garantizar que el modelo sí funcione con precisión, eficiencia y aplicabilidad práctica. Además es muy importante evaluar su rendimiento mediante métricas adecuadas, cumplir con las normativas, e implementar sistemas de monitoreo y mejora continua. Finalmente el potencial de los modelos de aprendizaje automático se realiza cuando son escalables, sostenibles a largo plazo y generan impacto en la toma de decisiones o la automatización de tareas.

BIBLIOGRAFÍA

- Mrukwa, G., Mrukwa, G., & Netguru. (2024, July 2). Machine learning development process: from data collection to model deployment. *Netguru*. <https://www.netguru.com/blog/machine-learning-development-process>
- Content Studio. (2024, March 29). *¿Qué es un proceso de aprendizaje automático?* Pure Storage. <https://www.purestorage.com/la/knowledge/what-is-machine-learning-pipeline.html>
- 2.2 Paradigmas de la Inteligencia Artificial – Sistemas e Informatica. (s. f.). <https://sistemasinformatica.istmo.tecnm.mx/cursos/inteligencia-artificial/2-representacion-del-conocimiento/2-2-paradigmas-de-la-inteligencia-artificial/#:~:text=Enfoque%20Conexionista,el%20procesamiento%20del%20lenguaje%20natural>.
- Alejandro. (2021, 13 mayo). *Modelos de la inteligencia artificial. El conexionista*. La Era de la Inteligencia Artificial. <https://cibernetica.wordpress.com/2021/05/13/modelos-de-la-inteligencia-artificial-el-conexionista/>
- Azcoitia, S. S. (2023, 29 junio). MYCIN, El comienzo de la Inteligencia Artificial en el mundo de la medicina. *Telefónica Tech*. <https://telefonicatech.com/blog/mycin-el-comienzo-de-la-inteligencia>
- Awan, A. A. (2024, March 12). Explicación del ciclo de vida del machine learning. <https://www.datacamp.com/es/blog/machine-learning-lifecycle-explained>
- Evaluación de modelos de ML - Amazon Machine Learning. (n.d.). https://docs.aws.amazon.com/es_es/machine-learning/latest/dg/evaluating_models.html
- Tuychiev, B. (2024, July 30). Implementación del Algoritmo de Dijkstra en Python: Un tutorial paso a paso. <https://www.datacamp.com/es/tutorial/dijkstra-algorithm-in-python>