



TECNOLÓGICO
NACIONAL DE MÉXICO



Tecnológico Nacional de México

Campus Culiacán

Carrera:

Ingeniería en Sistemas Computacionales

Nombre de la materia:

Inteligencia Artificial

Tarea 4: Paradigmas IA

Alumnos:

Aguilar Recio Jesús Octavio

Flores Fernández Emily Karely

Nombre del maestro:

Zuriel Dathan Mora Félix

Grupo:

9:00 – 10:00

Paradigma Simbólico

El paradigma simbólico, también conocido como IA basada en reglas o IA simbólica, se basa en la manipulación de símbolos y reglas lógicas para representar el conocimiento y tomar decisiones. Utiliza estructuras como **sistemas expertos**, **ontologías** y **lógica proposicional**, permitiendo razonar sobre problemas específicos mediante reglas definidas previamente.

Ejemplo:

Chatbot Basado en Reglas

Un chatbot basado en reglas, como los primeros asistentes virtuales (ej. ELIZA), es diseñado para interactuar con los usuarios mediante respuestas predefinidas en función de patrones de texto.

Aplicación:

El chatbot sigue un conjunto de reglas estructuradas, como:

- Si el usuario menciona "me siento triste", entonces responder "¿Quieres hablar de ello?".
- Si el usuario dice "hola", entonces responder "¡Hola! ¿Cómo puedo ayudarte hoy?".

Cada entrada del usuario es analizada mediante coincidencias de patrones y reglas lógicas preestablecidas para generar una respuesta adecuada.

Beneficios:

- Alta interpretabilidad y transparencia en las decisiones.
- Implementación relativamente sencilla y rápida.
- Adecuado para tareas estructuradas con respuestas predecibles.

Limitaciones:

- Falta de flexibilidad para manejar entradas fuera de las reglas predefinidas.
- No aprende ni mejora con el tiempo sin intervención manual.
- Difícil de escalar a aplicaciones más complejas con lenguaje natural avanzado.

Paradigma Conexionista

El paradigma conexionista se basa en redes neuronales artificiales inspiradas en el funcionamiento del cerebro humano. En lugar de reglas explícitas, este modelo aprende patrones y relaciones a partir de grandes volúmenes de datos mediante algoritmos de aprendizaje profundo.

Ejemplo:

Reconocimiento de Imágenes con Redes Neuronales

Un sistema de reconocimiento de imágenes basado en **Redes Neuronales Convolucionales (CNN)**, como el modelo **ResNet**, es utilizado para identificar objetos en fotografías.

Aplicación:

Este sistema es entrenado con millones de imágenes etiquetadas para aprender a distinguir entre diferentes objetos. La red neuronal ajusta sus pesos mediante **backpropagation** y optimización con gradiente descendente, permitiendo reconocer patrones complejos sin reglas predefinidas.

Beneficios:

- Capacidad de generalización y adaptación a nuevos datos sin necesidad de reglas explícitas.
- Alto rendimiento en tareas complejas como visión por computadora y procesamiento de lenguaje natural.
- Aprendizaje autónomo sin necesidad de programación manual extensa.

Limitaciones:

- Opacidad en la interpretación de los resultados (problema de "caja negra").
- Requiere grandes cantidades de datos y capacidad computacional.
- Posibilidad de sesgos en los datos de entrenamiento.

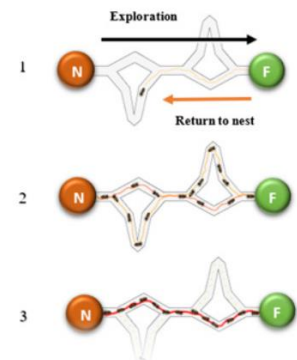
Paradigma Bioinspirado

El diseño bioinspirado (BID) es un método que transfiere principios, funciones y estrategias de los sistemas biológicos para resolver problemas complejos del mundo real. El proceso BID normalmente comprende cuatro etapas clave: definición del problema, exploración biológica, abstracción biológica y aplicación de la solución. La definición del problema y la exploración biológica implican inicialmente la identificación y definición de un problema de diseño, y luego la recuperación y exploración de analogías biológicas relevantes que podrían proporcionar soluciones.

Ejemplo:

Algoritmo de colonias de hormigas (ACO – Ant Colony Optimization)

Técnica de optimización metaheurística de colonias de hormigas basada en el comportamiento de las hormigas. Fue desarrollada a principios de los años 90 por Dorigo. Cuando buscan alimento, las hormigas exploran inicialmente el área que rodea su nido de forma aleatoria. Si una hormiga encuentra una fuente de alimento, la evalúa y devuelve algo de comida al nido. Durante el viaje de regreso, la hormiga deposita un rastro de feromonas en el camino seguido. La feromona depositada depende de la cantidad y calidad del alimento, y esto guía a otras hormigas hacia esta fuente de alimento. Los rastros de feromonas representan una comunicación indirecta (este sistema de comunicación se conoce como estigmergia) entre hormigas, lo que les permite encontrar los caminos más cortos entre su nido y las fuentes de alimento.



Aplicación:

Por ejemplo, si representamos a las hormigas en código “Hormigas Virtuales” estas exploran diferentes rutas en un grafo. Cada hormiga elige un camino basándose en la cantidad de feromonas depositadas y la distancia entre los nodos. Las rutas más cortas acumulan más feromonas, atrayendo a más hormigas. La relación con el enfoque bioinspirado es que el algoritmo imita el comportamiento natural de las hormigas, que usan feromonas para comunicarse y encontrar caminos eficientes.

Beneficios:

- Es eficiente para resolver problemas de optimización combinatoria, como el TSP.
- Puede encontrar soluciones cercanas al óptimo en problemas complejos.
- Búsqueda distribuida ya que exploran el espacio de soluciones de manera paralela.
- Esto permite una exploración más amplia y diversa del espacio de búsqueda.

Limitaciones:

- Puede ser lento en problemas con muchos nodos (ciudades).
- Requiere ajuste de parámetros, como la tasa de evaporación de feromonas.

Paradigma Computacional

Se centra en el diseño de sistemas informáticos inteligentes que imitan la naturaleza y el razonamiento lingüístico humano para resolver problemas complejos. También podemos decir que se centra en la resolución de problemas mediante algoritmos eficientes y técnicas de búsqueda heurística. Este enfoque es especialmente útil en problemas de planificación, optimización y toma de decisiones, donde se requiere encontrar soluciones en un tiempo razonable.

Ejemplo:

Algoritmo de búsqueda A*

El algoritmo de A* es un método de búsqueda heurística utilizado para encontrar la ruta más corta entre dos puntos en un mapa, como en sistemas de navegación GPS.

Aplicación:

El algoritmo combina la búsqueda informada, lo que significa que aprovecha una función heurística para guiar su búsqueda hacia el objetivo. Esta función heurística estima el coste de alcanzar el objetivo desde un nodo determinado, lo que permite al algoritmo dar prioridad a los caminos prometedores y evitar explorar los innecesarios.

Beneficios:

- Es eficiente y garantiza encontrar la ruta más corta si la heurística es admisible.
- Es ampliamente utilizado en aplicaciones de planificación de rutas y juegos.
- Garantía de optimalidad si la heurística es admisible (nunca sobreestima el costo real), A* garantiza encontrar la ruta más corta entre el nodo inicial y el objetivo.

Limitaciones:

- Requiere una buena heurística para ser eficaz.
- Puede consumir mucha memoria en problemas con espacios de búsqueda grandes.
- Garantía de optimalidad si la heurística es admisible (nunca sobreestima el costo real), A* garantiza encontrar la ruta más corta entre el nodo inicial y el objetivo.

Referencias

<https://www.sciencedirect.com/topics/engineering/ant-colony-optimization>

<https://www.cs.us.es/~fsancho/Blog/posts/PSO.md>

<https://www.datacamp.com/es/tutorial/a-star-algorithm>

<https://cibernetica.wordpress.com/2020/03/17/paradigmas-y-tendencias-en-la-investigacion-de-la-ia/>

<https://maslibertad.com.co/los-paradigmas-cientificos-de-la-inteligencia-artificial/>