



**TECNOLÓGICO
NACIONAL DE MÉXICO**



Instituto Tecnológico de México

Campus Culiacán

Ingeniería en Sistemas Computacionales

Materia: Inteligencia Artificial

Hora: 11:00 - 12:00

Tarea 5: Paradigmas de IA y Aprendizaje Automatizado

Docente: Zuriel Dathan Mora Felix

Equipo:

-Yosef Emiliano Soto Iribé

-Cristian Andrea Ramírez Medina

Paradigmas de Inteligencia Artificial

Enfoque conexionista

El conexionismo, también conocido como procesamiento distribuido en paralelo, se refiere a un conjunto de enfoques en el campo de la inteligencia artificial que modela fenómenos mentales o conductuales como procesos emergentes de redes interconectadas de unidades simples. Estas redes interconectadas son similares a las redes neuronales y se utilizan ampliamente para simular procesos cognitivos. En el contexto de la inteligencia artificial y el aprendizaje automático, el conexionismo desempeña un papel fundamental en el modelado de sistemas complejos y ha producido avances significativos en diversas aplicaciones, como el reconocimiento de imágenes, el procesamiento del lenguaje natural y el análisis predictivo.

Hubo un tiempo que se utilizó el nombre de paradigma conexionista que, más representar los estados mentales, buscaba la representación de las conexiones que se establecen entre las diferentes de un organismo, ya sean neuronas (redes neuronales), agentes (agentes inteligentes), genes (algoritmos genéticos) y como de estas conexiones se genera un accionar inteligente. Ahora en lugar de buscar la representación del conocimiento humano se buscaba la representación de elementos no inteligentes (neuronas, agentes, genes) que son capaces de conectarse entre sí para darle solución a un problema. Conexiones que se van almacenando como un aprendizaje. Esta flexibilidad permite que estos sistemas no dependan de un ingeniero de conocimientos que los esté alimentando constantemente, además rompen con el esquema secuencial de paradigma simbólico que obliga a contar con una máquina de inferencia que vaya ejecutando las reglas, ahora estos entes (neuronas, agentes, genes) son capaces de conectarse entre sí y de ajustarse al problema e ir guardando su experiencia.

Todas tienen en común un nuevo enfoque: la inteligencia es producto de la asociación, combinación o conexión de entidades independientes más simples (agentes, genes o neuronas), las cuales pueden tener alguna inteligencia o ninguna y producto de su interacción, el sistema presentaría un comportamiento inteligente. Lo cual nos lleva a un nuevo paradigma basado en la copia de los sistemas emergentes y evolutivos de la naturaleza, que de cierta manera es un retorno a la cibernética, primera ciencia de carácter técnico inspirada en la naturaleza.

Redes neuronales artificiales (RNA)

Modelo conexionista

Los sistemas conexionistas no son incompatibles con la hipótesis simbolista (SSF) pero al contrario del simbólico, se trata de un modelo ascendente, ya que se basa en la hipótesis de que la inteligencia emerge a partir de la actividad distribuida de un gran número de unidades interconectadas que procesan información paralelamente. En la IA conexionista estas unidades son modelos muy aproximados de la actividad eléctrica de las neuronas biológicas.

Ventajas:

- Aprendizaje adaptativo y generalización a partir de ejemplos
- Capacidades de procesamiento paralelo y distribuido
- Resiliencia ante datos ruidosos e incompletos
- Representación eficaz de patrones y relaciones complejos

desventajas:

- Dependencia de datos de entrenamiento extensos
- Riesgo de sobreajuste en arquitecturas de redes neuronales complejas
- Interpretabilidad y transparencia de las decisiones del modelo

Ejemplos:

1. El ámbito del reconocimiento de imágenes y la visión artificial depende en gran medida de enfoques conexionistas para tareas como la detección de objetos, el reconocimiento facial y la comprensión de escenas. Al aprovechar las redes neuronales convolucionales (CNN) y las arquitecturas de aprendizaje profundo, los modelos conexionistas se destacan en la extracción de características jerárquicas de las imágenes, lo que permite una identificación robusta y precisa de objetos y patrones visuales. Los sistemas de reconocimiento de imágenes impulsados por el conexionismo se han implementado ampliamente en diversos dominios, que abarcan el diagnóstico médico, la automatización industrial y la navegación autónoma.
2. El conexionismo desempeña un papel fundamental en el desarrollo de vehículos autónomos, donde los algoritmos basados en redes neuronales permiten la toma de decisiones inteligente y capacidades de percepción. Al procesar datos de sensores del entorno del vehículo, los sistemas conexionistas pueden analizar escenarios de conducción complejos, detectar obstáculos y adaptarse a las condiciones dinámicas de la carretera. A través de la integración de los principios conexionistas, los vehículos autónomos demuestran una mayor conciencia situacional y razonamiento cognitivo, lo que fomenta avances en el ámbito de las tecnologías de conducción autónoma.
3. El software de texto predictivo, que predomina en los teléfonos inteligentes y las aplicaciones de procesamiento de texto, aprovecha los modelos conexionistas para anticipar y sugerir palabras o frases a medida que los usuarios escriben. A través del análisis de patrones y contexto en las entradas de texto, el software emplea algoritmos de redes neuronales para predecir la próxima palabra más probable, lo que mejora la velocidad y la precisión de la escritura. Esta aplicación del conexionismo muestra su capacidad para inferir el contexto y las relaciones semánticas, lo que proporciona a los usuarios capacidades de entrada de texto intuitivas y eficientes.

Enfoque simbólico

Los algoritmos de IA simbólica funcionan 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.

Supongamos que tenemos un sistema de IA simbólica diseñado para diagnosticar afecciones médicas en función de los síntomas comunicados por un paciente. El sistema tiene un conjunto de reglas y axiomas que utiliza para hacer inferencias y deducciones sobre el estado del paciente.

Por ejemplo, si el paciente informa de que tiene fiebre, el sistema podría utilizar la siguiente regla:

IF patient has a fever AND patient has a cough AND patient has difficulty breathing THEN patient may have pneumonia.

A continuación, el sistema comprobaría si el paciente también tiene tos y dificultad para respirar y, en ese caso, concluiría que puede tener neumonía.

Este enfoque es muy interpretable, ya que podemos rastrear fácilmente el proceso de razonamiento hasta las reglas lógicas que se aplicaron. También nos permite modificar y actualizar fácilmente las normas del sistema a medida que disponemos de nueva información.

La IA simbólica utiliza lenguajes formales como la lógica para representar el conocimiento. Este conocimiento lo procesan motores de razonamiento que utilizan algoritmos para manipular símbolos. Esto permite crear sistemas como los sistemas expertos y los sistemas de apoyo a la toma de decisiones, que pueden hacer deducciones e inferencias basadas en reglas y conocimientos predefinidos.

Ejemplos de aplicaciones de IA simbólica en el mundo real

La IA simbólica se ha aplicado en diversos campos, como el procesamiento del lenguaje natural, los sistemas expertos y la robótica. Algunos ejemplos concretos son:

- Siri y otros asistentes digitales utilizan la IA simbólica para comprender el lenguaje natural y dar respuestas.
- Los sistemas de diagnóstico médico utilizan la IA simbólica para proporcionar recomendaciones a los médicos basadas en los síntomas del paciente (como se ha demostrado antes).
- Los coches autónomos utilizan la IA simbólica para tomar decisiones basadas en el entorno, como reconocer las señales de stop y los semáforos.
- Los sistemas de visión por ordenador utilizan la IA simbólica para reconocer objetos y patrones en las imágenes.

¿Cuáles son las ventajas de la IA simbólica?

La IA simbólica tiene varias ventajas, entre ellas:

- **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.

¿Cuáles son las limitaciones de la IA simbólica?

La IA simbólica tiene algunas 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.

Ejemplo de IA Simbólica (Precio de una Casa):

Imagina que queremos predecir el precio de una casa utilizando Inteligencia Artificial simbólica. Un sistema experto estaría programado con reglas predefinidas por un experto en bienes raíces, como:

* Regla 1: Si el tamaño de la casa es mayor a 150 m² y está ubicada en el centro de la ciudad, entonces el precio es mayor a 300,000 €.

* Regla 2: Si la casa tiene más de 3 habitaciones y un jardín grande, entonces el precio aumenta en 50,000 €.

* Regla 3: Si la casa tiene más de 20 años de antigüedad, entonces el precio baja en 30,000 €.



Enfoque bioinspirado: De la misma forma que los algoritmos evolutivos están basados en las leyes de la selección natural, los algoritmos de inteligencia de enjambre se basan, en sus comienzos, en el estudio del comportamiento de insectos sociales como las hormigas o las abejas, ya que tienen una alta capacidad de organización siguiendo normas muy simples[6]. Este comportamiento social se caracteriza por ser autónomo, auto-regulable, y funcionar de una forma descentralizada. Posteriormente su aplicación se ha extendido a otros tipos de agrupaciones animales tales como las bandadas de pájaros, la comunicación entre luciérnagas o incluso se han llegado a basar en el funcionamiento del sistema inmune. Como ya hemos mencionado, estos algoritmos se basan en la inteligencia emergente de la interacción de agentes de comportamiento simple. Para conseguir esto, los algoritmos de inteligencia de enjambre siguen 5 principios básicos:

1) Principio de proximidad: La población debe ser capaz de hacer cálculos de espacio y tiempo sencillos.

2) Principio de la calidad: La población tiene que ser capaz de reconocer cambios de la calidad en el “ambiente” (es decir, en la nube de soluciones en las que se está ejecutando la población)

3) Principio de la respuesta diversa: La población no puede ejecutar su actividad en zonas de soluciones excesivamente pequeñas.

4) Principio de estabilidad: La población no debe cambiar su comportamiento cada vez que cambie el “ambiente”

5) Principio de la adaptabilidad: La población debe cambiar su comportamiento cuando merezca la pena el esfuerzo computacional requerido

Para imitar este comportamiento, se establece un enjambre de “partículas” (que son agentes que no tienen masa o volumen, a los que se les somete a velocidad y aceleración distintas para buscar una solución) que serían puntos en la nube de soluciones, capaces de hacer cálculos sencillos. Cada partícula tiene dos parámetros, la posición que es la solución de la nube en la que se encuentra actualmente, y la velocidad a la que se mueve por los ejes de la la nube de soluciones. (determinando así también la dirección de la partícula). Además cada partícula es capaz de “recordar” la posición con mejor fitness en la que se ha encontrado (pbest) y la posición con mejor fitness de todas las partículas vecinas (gbest). En cada ciclo la nueva posición de la partícula viene determinada por su posición anterior y por la velocidad, que depende de su velocidad anterior pero que también está influenciada con una intensidad aleatoria por pbest y gbest. De esta forma se guía a la partícula hacia una zona con mejor soluciones.

En su ejecución se siguen los siguientes pasos:

1) Iniciación, se coloca cada partícula del enjambre en una posición aleatoria en la nube de soluciones.

2) Evaluación del fitness de las partículas.

3) Comparación del valor del fitness de cada partícula con su pbest correspondiente. En el caso de ser mayor, se sustituye el pbest por la nueva posición.

4) Identificación de la partícula con mejor fitness sustituyendo gbest por el valor de su posición.

5) Modificación de los parámetros de cada partícula y cálculo de su nueva posición.

6) Repetición de los pasos hasta que se llegue al número de ciclos indicado o a un fitness aceptable.

Este método tiene varias ventajas:

Eficiente: se implementa muy fácilmente al ser muy sencillo, y además da unos buenos resultados.

Capacidad de memoria superior a los Algoritmos Evolutivos.

Aumento de variabilidad: ya que cada partícula usa su mejor resultado y el mejor resultado de todas para alcanzar mejores valores, en vez de eliminar los peores valores como ocurría en los Algoritmos Evolutivos. Así se consigue aumentar la variabilidad de la población.

Principalmente utilizado para optimización de funciones continuas no lineales de forma estocástica.

Este algoritmo se caracteriza por su alto rendimiento comparado con otros algoritmos así como capacidad para salir de óptimos locales y encontrar el óptimo global. Sin embargo, los parámetros de control que se le impongan al inicio pueden afectar su rendimiento y su velocidad. Se puede emplear eficazmente en problemas de optimización complejos, como multimodales y multivariantes. También ha demostrado su utilidad en aplicaciones industriales como problemas de ordenamiento de trabajos, es decir el orden en el que se deben encontrar los diferentes procesos de una cadena industrial para mejorar su eficacia y reducir el tiempo de producción.

Modelo general de la robótica de enjambre

El modelo de robótica de enjambre es un componente clave del algoritmo cooperativo que controla los comportamientos e interacciones de todos los individuos. En el modelo, los robots del enjambre deberían tener algunas funciones básicas, como detección, comunicación, movimiento, etc.

Módulo de intercambio de información

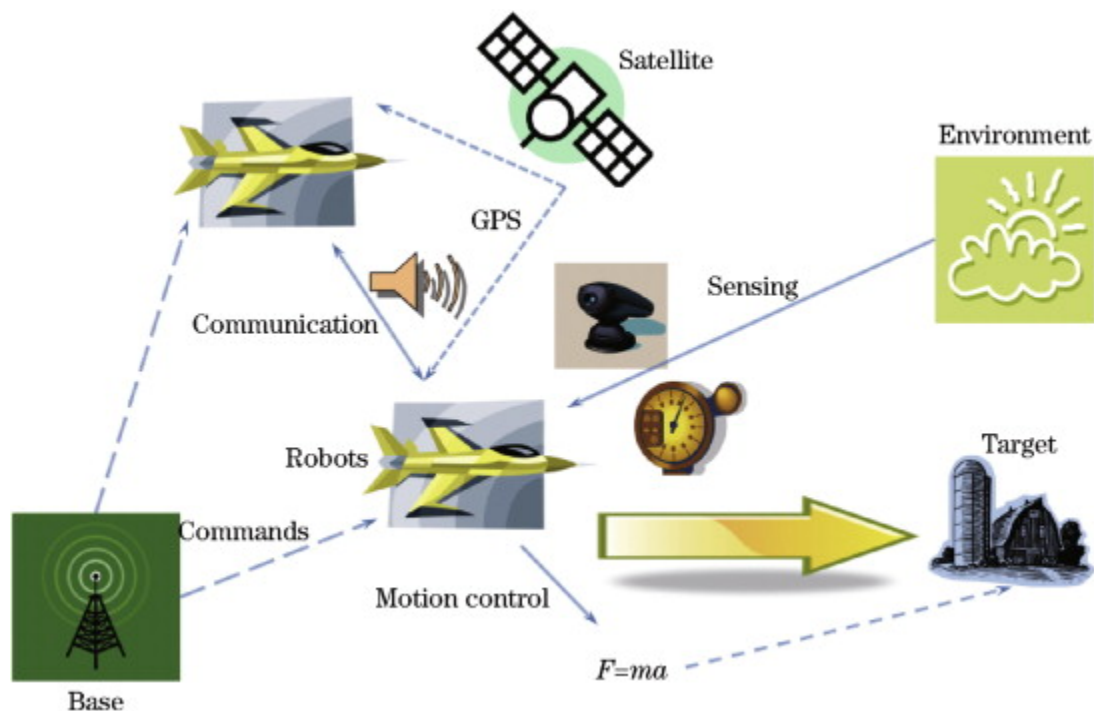
El intercambio de información es inevitable cuando los robots cooperan entre sí y es la parte fundamental para controlar el comportamiento del enjambre. Las principales funciones de los individuos involucrados en este módulo son la detección limitada y la comunicación local. El intercambio de información de un robot se divide en dos categorías: interacción con el robot o con el entorno. Las estrategias pueden ser iguales o diferentes para el enjambre debido a las diferentes aplicaciones.

Comunicación a través del entorno

El entorno puede actuar como intermediario para la interacción de los robots. Los robots dejan sus huellas en el entorno después de una acción para estimular a otros robots que pueden detectar la huella, sin comunicación directa entre individuos. De esta manera, las acciones posteriores tienden a reforzarse

y apoyarse mutuamente, lo que lleva al surgimiento espontáneo de actividades a nivel de enjambre. El enjambre se imita como hormigas o abejas e interactúa con la ayuda de feromonas virtuales. Este esquema interactivo está exento de la explosión exponencial de la población, pero tiene algunas limitaciones en el entorno para apoyar las feromonas.

Ejemplo:



Se muestra un modelo general de robótica de enjambre. Los robots se comunican entre sí. En algunos casos, se introducen comandos de posicionamiento global o centrales, pero el enjambre debería poder completar la tarea si se bloquea la comunicación global.

Enfoque 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.

Ejemplo 1: Algoritmos de Búsqueda en Juegos (Ajedrez)

Descripción:

Los motores de ajedrez como Stockfish y AlphaZero utilizan algoritmos de búsqueda para evaluar millones de posibles movimientos y seleccionar el mejor.

Aplicación del paradigma:

El paradigma computacional se aplica mediante el uso de algoritmos como *Minimax* con poda *Alpha-Beta* para evaluar jugadas futuras. En el caso de AlphaZero, se usa una combinación de redes neuronales y búsqueda de Monte Carlo.

Beneficios y limitaciones:

- Beneficios: Capacidad de analizar millones de jugadas por segundo y encontrar soluciones óptimas.
- Limitaciones: Requiere gran poder de cómputo y puede ser ineficiente en juegos con espacios de búsqueda demasiado grandes.

Ejemplo 2: Redes Neuronales en Reconocimiento de Imágenes**Descripción:**

Modelos como CNN (Convolutional Neural Networks) se usan para tareas como el reconocimiento facial en teléfonos o la detección de objetos en imágenes médicas.

Aplicación del paradigma:

Las redes neuronales artificiales aprenden a reconocer patrones en datos de imágenes, ajustando pesos y capas para mejorar la precisión de la clasificación.

Beneficios y limitaciones:

- Beneficios: Alto rendimiento en tareas de clasificación y reconocimiento de patrones.
- Limitaciones: Requiere grandes volúmenes de datos y poder computacional para entrenar modelos con precisión.

Enfoque Probabilístico:

Se basa en la teoría de la probabilidad para manejar la incertidumbre. Es útil para resolver problemas en los que la información es incompleta o inexacta. Se ha utilizado con éxito en aplicaciones como la planificación y el diagnóstico. Sin embargo, puede ser complejo modelar la incertidumbre de forma precisa.

Ejemplo 1: Filtros Bayesianos en Correos Electrónicos (Detección de Spam)

Descripción:

Los filtros de spam utilizan modelos probabilísticos para clasificar correos electrónicos como spam o no spam.

Aplicación del paradigma:

Se usa el **teorema de Bayes** para calcular la probabilidad de que un correo sea spam dado el contenido del mensaje. Palabras sospechosas aumentan la probabilidad de ser spam, mientras que otras pueden reducirla.

Beneficios y limitaciones:

- **Beneficios:** Adaptabilidad y precisión en la clasificación de correos no deseados.
- **Limitaciones:** Puede cometer errores si los correos de spam evolucionan y no se actualiza el modelo.

Ejemplo 2: Modelos de Markov Ocultos en Reconocimiento de Voz**Descripción:**

Asistentes de voz como Siri o Google Assistant usan modelos probabilísticos para interpretar el habla humana.

Aplicación del paradigma:

Los **Modelos de Markov Ocultos (HMM)** calculan la probabilidad de una secuencia de sonidos siendo una palabra específica, permitiendo convertir el audio en texto.

Beneficios y limitaciones:

- **Beneficios:** Maneja variaciones en la pronunciación y el ruido en el audio.
- **Limitaciones:** Puede confundirse con acentos o palabras similares si el modelo no ha sido bien entrenado.

Machine Learning o Aprendizaje Automático

¿Qué es el machine learning?

Es una disciplina de la inteligencia artificial la cual por medio de algoritmos puede hacer que una computadora pueda identificar patrones en datos masivos y hacer predicciones. Este aprendizaje permite a los computadores realizar tareas específicas de forma autónoma, es decir, sin necesidad de ser programados.

El término se utilizó por primera vez en 1959. Sin embargo, ha ganado relevancia en los últimos años debido al aumento de la capacidad de computación y al boom de los datos. Las técnicas de aprendizaje automático son, de hecho, una parte fundamental del Big Data.

El Machine Learning es uno de los pilares sobre los que descansa la transformación digital. En la actualidad, ya se está utilizando para encontrar nuevas soluciones en diferentes campos, entre los que cabe destacar:

Recomendaciones: permite hacer sugerencias personalizadas de compra en plataformas *online* o recomendar canciones. En su forma más básica analiza el historial de compras y reproducciones del usuario y lo compara con lo que han hecho otros usuarios con tendencias o gustos parecidos. Spotify, Youtube y las principales plataformas de streaming lo usan para recomendar nuevos contenidos que mantengan al usuario en la página durante más tiempo, por ejemplo.

Vehículos inteligentes: Los vehículos inteligentes ya son una realidad, con varios de ellos siendo testados en las carreteras. Gracias al aprendizaje automático, estos vehículos podrán ajustar la configuración interna (temperatura, música, inclinación del respaldo, etc.) de acuerdo a las preferencias del conductor e, incluso, mover el volante solos para reaccionar al entorno.

Redes sociales: X, por ejemplo, se sirve de algoritmos de *Machine Learning* para reducir en gran medida el spam publicado en esta red social mientras que Facebook, a su vez, lo utiliza para detectar tanto noticias falsas como contenidos no permitidos en retransmisiones en directo que bloquea automáticamente.

Búsquedas: los motores de búsqueda se sirven del aprendizaje automático para optimizar sus resultados en función de su eficacia, midiendo la misma a través de los clics del usuario.

Los 6 pasos del ciclo de vida estándar de machine learning son los siguientes:

1-Planificación:

La fase de planificación implica evaluar el alcance, la métrica de éxito y la viabilidad de la aplicación del ML. Tienes que entender el negocio y cómo utilizar el machine learning para mejorar el proceso actual. Por ejemplo: ¿Necesitamos machine learning? ¿Podemos conseguir peticiones similares con una programación sencilla?

También tienes que comprender el análisis coste-beneficio y cómo enviarán la solución en varias fases. Además, tienes que definir métricas de éxito claras y mensurables para el negocio, los modelos de machine learning (precisión, puntuación F1, AUC) y económicas (indicadores clave de rendimiento).

2-Preparación de datos

La sección de preparación de datos se divide a su vez en cuatro partes: obtención y etiquetado de datos, limpieza, gestión y procesamiento.

Recogida de datos y etiquetado

Primero tenemos que decidir cómo vamos a recopilar los datos: recopilando los datos internos, de código abierto, comparándolos a los vendedores o generando datos sintéticos. Cada método tiene pros y contras, y en algunos casos, obtenemos los datos de las cuatro metodologías.

Tras la recogida, tenemos que etiquetar los datos. Comprar datos limpios y etiquetados no es factible para todas las empresas, y también puede que tengas que hacer cambios en la selección de datos durante el proceso de desarrollo. Por eso no puedes comprarlos al por mayor y los datos pueden acabar siendo inútiles para la solución.

Limpieza de datos

A continuación, limpiaremos los datos imputando los valores que faltan, analizando los datos mal etiquetados, eliminando los valores atípicos y reduciendo el ruido. Crearás una canalización de datos para automatizar este proceso y realizar la verificación de la calidad de los datos.

Tratamiento de datos

La etapa de procesamiento de datos implica la selección de características, el tratamiento de las clases desequilibradas, la ingeniería de características, el aumento de datos y la normalización y escalado de los datos.

Gestión de datos

Por último, descubriremos soluciones de almacenamiento de datos, versionado de datos para su reproducibilidad, almacenamiento de metadatos y creación de canalizaciones ETL. Esta parte garantizará un flujo constante de datos para el entrenamiento del modelo.

3-Ingeniería de modelos

En esta fase, utilizaremos toda la información de la fase de planificación para construir y entrenar un modelo de machine learning. Por ejemplo: seguimiento de las métricas del modelo, garantía de escalabilidad y robustez, y optimización de los recursos de almacenamiento y computación.

4-Evaluación del modelo

Ahora que hemos finalizado la versión del modelo, es el momento de probar varias métricas.

Primero probaremos nuestro modelo en un conjunto de datos de prueba y nos aseguraremos de implicar a expertos en la materia para identificar el error en las predicciones.

También tenemos que asegurarnos de que seguimos los marcos industriales, éticos y legales para construir soluciones de IA.

Además, comprobaremos la solidez de nuestro modelo con datos aleatorios y del mundo real. Asegurarse de que el modelo infiere lo bastante rápido para aportar valor.

5-Implementación del modelo

En esta fase, implementamos modelos de machine learning en el sistema actual. Por ejemplo: introducir el etiquetado automático del almacén utilizando la forma del producto. Implementaremos un modelo de visión por ordenador en el sistema actual, que utilizará las imágenes de la cámara para imprimir las etiquetas.

Generalmente, los modelos pueden desplegarse en la nube y en el servidor local, navegador web, paquete como software y dispositivo de borde. Después, puedes utilizar la API, la aplicación web, los plugins o el panel de control para acceder a las predicciones.

En el proceso de despliegue, definimos el hardware de inferencia. Tenemos que asegurarnos de que tenemos suficiente RAM, almacenamiento y potencia de cálculo para producir resultados rápidos. Después, evaluaremos el rendimiento del modelo en producción mediante pruebas A/B, garantizando la aceptabilidad del usuario.

6-Supervisión y mantenimiento

Después de desplegar el modelo en producción, necesitamos supervisar y mejorar constantemente el sistema. Controlaremos las métricas del modelo, el rendimiento del hardware y el software, y la satisfacción del cliente.

La supervisión se realiza de forma totalmente automática, y se notifica a los profesionales las anomalías, la reducción del rendimiento del modelo y del sistema, y las malas críticas de los clientes.

Similitudes y diferencias entre los componentes del modelo cognitivo y las etapas del aprendizaje automatizado

Similitudes:

Procesamiento de información

- Ambos implican la recepción, transformación y almacenamiento de información para generar respuestas o acciones.
- En el modelo cognitivo, esto ocurre a través de la percepción, la memoria y el razonamiento.
- En ML, se da mediante la adquisición de datos, el entrenamiento y la inferencia.

Aprendizaje basado en experiencia

- Los humanos y las máquinas aprenden de la experiencia.
- En el modelo cognitivo, el aprendizaje ocurre a través de la observación, la práctica y el refuerzo.
- En ML, los algoritmos aprenden de conjuntos de datos mediante optimización y ajuste de parámetros.

Generalización y adaptación

- Ambos buscan encontrar patrones en los datos y generalizar el conocimiento para aplicarlo a nuevas situaciones.
- En el modelo cognitivo, esto se ve en la capacidad de resolver problemas nuevos basándose en conocimientos previos.
- En ML, los modelos generalizan a partir del entrenamiento y ajustan su desempeño con nuevas muestras.

Diferencias:

Fuente del conocimiento

- En el modelo cognitivo, el conocimiento proviene de la experiencia, la observación, la interacción social y la intuición.
- En el aprendizaje automático, el conocimiento se adquiere a partir de grandes volúmenes de datos estructurados mediante algoritmos matemáticos.

Flexibilidad y adaptación

- Los humanos pueden aprender de diversas fuentes, transferir conocimientos a nuevas áreas y adaptarse a entornos desconocidos.
- Los modelos de ML dependen de datos específicos y tienen dificultades para aplicar el conocimiento fuera de su dominio de entrenamiento.

Capacidad de abstracción y creatividad

- El modelo cognitivo permite la abstracción, el pensamiento conceptual, la imaginación y la creatividad.
- En ML, la abstracción es limitada y se basa en patrones dentro de los datos sin una verdadera comprensión conceptual.

Eficiencia en el aprendizaje

- Los humanos pueden aprender con pocas experiencias y generalizar rápidamente a nuevas situaciones.
- Los modelos de ML requieren grandes volúmenes de datos y múltiples iteraciones para lograr una buena generalización.

Referencias

- (N.d.-a). Iberdrola.com. Retrieved March 9, 2025, from <https://www.iberdrola.com/innovacion/machine-learning-aprendizaje-automatico>
- (N.d.-b). Datacamp.com. Retrieved March 9, 2025, from <https://www.datacamp.com/es/blog/machine-learning-lifecycle-explained>
- 2.2 *Paradigmas de la Inteligencia Artificial – Sistemas e Informatica*. (n.d.). Tecnm.mx. Retrieved March 9, 2025, from <https://sistemasinformatica.istmo.tecnm.mx/cursos/inteligencia-artificial/2-representacion-del-conocimiento/2-2-paradigmas-de-la-inteligencia-artificial/>
- Estratégica, P. y. C. (2024, 23 octubre). ¿Cómo la búsqueda visual con IA puede cambiar el proceso de compra? *Agencia de Comunicación SMiLE*. <https://smilecomunicacion.com/inteligencia-artificial/inteligencia-artificial-simbolica-vs-machine-learning/#ia-simbolica>
- 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/>
- Lark Editorial Team. (2023, 23 diciembre). *Connectionism*. https://www.larksuite.com/en_us/topics/ai-glossary/connectionism