



INTELIGENCIA ARTIFICIAL

www.postparaprogramadores.com

Contenido

AI - Inicio

AI - Descripción general

AI - Sistemas inteligentes

AI - Áreas de investigación

AI - Agentes y entornos

AI - Algoritmos de búsqueda populares

AI - Sistemas de lógica difusa

AI - Procesamiento de lenguaje natural

AI - Sistemas expertos

AI - Robótica

AI - Redes neuronales

AI - Problemas

AI - Terminología

Inteligencia artificial - Descripción general

Desde la invención de las computadoras o máquinas, su capacidad para realizar diversas tareas continuó creciendo exponencialmente. Los humanos han desarrollado el poder de los sistemas informáticos en términos de sus diversos dominios de trabajo, su velocidad creciente y la reducción de tamaño con respecto al tiempo.

Una rama de la informática llamada *Inteligencia Artificial* persigue la creación de computadoras o máquinas tan inteligentes como los seres humanos.

¿Qué es la inteligencia artificial?

Según el padre de Inteligencia Artificial, John McCarthy, es *"La ciencia y la ingeniería de hacer máquinas inteligentes, especialmente programas informáticos inteligentes"*.

La Inteligencia Artificial es una forma de **hacer que una computadora, un robot controlado por computadora o un software piensen de manera inteligente**, de la misma manera que piensan los humanos inteligentes.

La IA se logra al estudiar cómo piensa el cerebro humano y cómo los humanos aprenden, deciden y trabajan mientras intentan resolver un problema, y luego usan los resultados de este estudio como base para desarrollar software y sistemas inteligentes.

Filosofía de la IA

Mientras explota el poder de los sistemas informáticos, la curiosidad del ser humano lo lleva a preguntarse: *"¿Puede una máquina pensar y comportarse como lo hacen los humanos?"*

Por lo tanto, el desarrollo de la IA comenzó con la intención de crear una inteligencia similar en las máquinas que encontramos y consideramos alta en los humanos.

Objetivos de la IA

- **Para crear sistemas expertos** : los sistemas que exhiben un comportamiento inteligente, aprenden, demuestran, explican y aconsejan a sus usuarios.
- **Implementar la inteligencia humana en las máquinas** : crear sistemas que entiendan, piensen, aprendan y se comporten como los humanos.

Descarga más libros de programación GRATIS [click aquí](#)

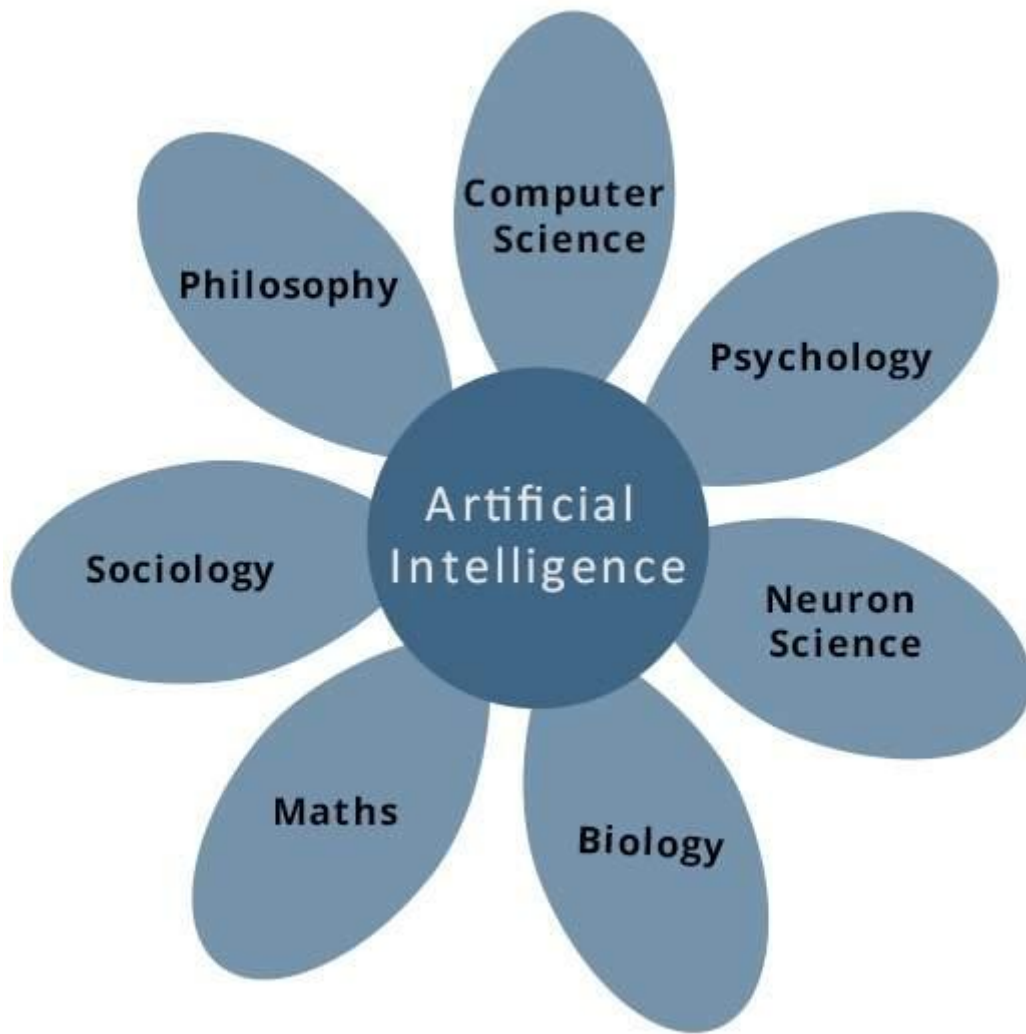


Síguenos en Instagram para que estés al tanto de los nuevos libros de programación. [Click aqui](#)

¿Qué contribuye a la IA?

La inteligencia artificial es una ciencia y tecnología basada en disciplinas como la informática, la biología, la psicología, la lingüística, las matemáticas y la ingeniería. Uno de los principales objetivos de la IA es el desarrollo de funciones informáticas asociadas con la inteligencia humana, como el razonamiento, el aprendizaje y la resolución de problemas.

De las siguientes áreas, una o varias áreas pueden contribuir a construir un sistema inteligente.



Programación sin y con IA

La programación sin y con AI es diferente de las siguientes maneras:

Programación sin IA	Programación con IA
Un programa de computadora sin IA puede responder las preguntas específicas que debe resolver.	Un programa de computadora con IA puede responder las preguntas genéricas que debe resolver.
La modificación en el programa conduce a cambios en su estructura.	Los programas de IA pueden absorber nuevas modificaciones al reunir piezas de información altamente independientes. Por lo tanto, puede modificar incluso una pequeña pieza de

	información del programa sin afectar su estructura.
La modificación no es rápida y fácil. Puede llevar a afectar negativamente al programa.	Modificación rápida y fácil del programa.

¿Qué es la técnica de IA?

En el mundo real, el conocimiento tiene algunas propiedades no deseadas:

- Su volumen es enorme, casi inimaginable.
- No está bien organizado o bien formateado.
- Sigue cambiando constantemente.

La técnica de IA es una manera de organizar y usar el conocimiento de manera eficiente de tal manera que:

- Debe ser perceptible por las personas que lo proporcionan.
- Debe ser fácilmente modificable para corregir errores.
- Debería ser útil en muchas situaciones, aunque sea incompleta o inexacta.

Las técnicas de IA elevan la velocidad de ejecución del complejo programa con el que está equipado.

Aplicaciones de IA

La IA ha sido dominante en varios campos, tales como:

- **Juegos** : la IA juega un papel crucial en juegos estratégicos como el ajedrez, el póker, el tic-tac-toe, etc., donde la máquina puede pensar en una gran cantidad de posibles posiciones basadas en el conocimiento heurístico.
- **Procesamiento del lenguaje natural** : es posible interactuar con la computadora que comprende el lenguaje natural hablado por los humanos.
- **Sistemas expertos** : hay algunas aplicaciones que integran máquina, software e información especial para impartir razonamiento y asesoramiento. Proporcionan explicaciones y consejos a los usuarios.
- **Sistemas de visión** : estos sistemas entienden, interpretan y comprenden la información visual en la computadora. Por ejemplo,
 - Un avión espía toma fotografías, que se utilizan para descubrir información espacial o un mapa de las áreas.
 - Los médicos usan el sistema clínico experto para diagnosticar al paciente.
 - La policía usa un software de computadora que puede reconocer la cara del criminal con el retrato almacenado hecho por el artista forense.
- **Reconocimiento de voz** : algunos sistemas inteligentes son capaces de escuchar y comprender el lenguaje en términos de oraciones y sus significados mientras un

humano habla con él. Puede manejar diferentes acentos, palabras de argot, ruido de fondo, cambios en el ruido humano debido al frío, etc.

- **Reconocimiento de escritura a mano:** el software de reconocimiento de escritura a mano lee el texto escrito en papel con un bolígrafo o en la pantalla con un lápiz óptico. Puede reconocer las formas de las letras y convertirlas en texto editable.
- **Robots inteligentes :** los robots pueden realizar las tareas asignadas por un humano. Tienen sensores para detectar datos físicos del mundo real, como luz, calor, temperatura, movimiento, sonido, golpes y presión. Tienen procesadores eficientes, múltiples sensores y una gran memoria para exhibir inteligencia. Además, son capaces de aprender de sus errores y pueden adaptarse al nuevo entorno.

Historia de la IA

Aquí está la historia de la IA durante el siglo XX:

Año	Hito / Innovación
1923	La obra de Karel Čapek llamada "Rossum's Universal Robots" (RUR) se abre en Londres, primer uso de la palabra "robot" en inglés.
1943	Fundamentos para redes neuronales establecidas.
1945	Isaac Asimov, un ex alumno de la Universidad de Columbia, acuñó el término <i>Robótica</i> .
1950	Alan Turing presentó Turing Test para la evaluación de inteligencia y publicó <i>Computing Machinery and Intelligence</i> . Claude Shannon publicó <i>Análisis detallado del juego de ajedrez</i> como una búsqueda.
1956	John McCarthy acuñó el término <i>Inteligencia Artificial</i> . Demostración del primer programa de IA en ejecución en la Universidad Carnegie Mellon.
1958	John McCarthy inventa el lenguaje de programación LISP para IA.
1964	La disertación de Danny Bobrow en el MIT mostró que las computadoras pueden entender el lenguaje natural lo suficientemente bien como para resolver problemas de álgebra de forma correcta.
1965	Joseph Weizenbaum del MIT construyó <i>ELIZA</i> , un problema interactivo que lleva a cabo un diálogo en inglés.
1969	Los científicos del Instituto de Investigación de Stanford desarrollaron <i>Shakey</i> , un

	robot, equipado con locomoción, percepción y resolución de problemas.
1973	El grupo Assembly Robotics de la Universidad de Edimburgo construyó <i>Freddy</i> , el famoso robot escocés, capaz de usar la visión para ubicar y ensamblar modelos.
1979	Se construyó el primer vehículo autónomo controlado por computadora, Stanford Cart.
1985	Harold Cohen creó y demostró el programa de dibujo, <i>Aaron</i> .
1990	<p>Grandes avances en todas las áreas de la IA:</p> <ul style="list-style-type: none"> • Demostraciones significativas en el aprendizaje automático. • Razonamiento basado en casos • Planificación multiagente • Planificación • Minería de datos, rastreador web • comprensión y traducción del lenguaje natural • Visión, realidad virtual • Juegos
1997	El programa Deep Blue Chess supera al entonces campeón mundial de ajedrez, Garry Kasparov.
2000	Las mascotas robot interactivas están disponibles comercialmente. MIT muestra a <i>Kismet</i> , un robot con una cara que expresa emociones. El robot <i>Nomad</i> explora regiones remotas de la Antártida y localiza meteoritos.

Inteligencia Artificial - Sistemas Inteligentes

Mientras estudias inteligencia artificial, necesitas saber qué es la inteligencia. Este capítulo cubre la idea de inteligencia, tipos y componentes de inteligencia.

¿Qué es la inteligencia?

La capacidad de un sistema para calcular, razonar, percibir relaciones y analogías, aprender de la experiencia, almacenar y recuperar información de la memoria, resolver problemas, comprender ideas complejas, usar el lenguaje natural con fluidez, clasificar, generalizar y adaptar nuevas situaciones.

Tipos de inteligencia

Según lo descrito por Howard Gardner, un psicólogo del desarrollo estadounidense, la inteligencia viene en múltiples fases:

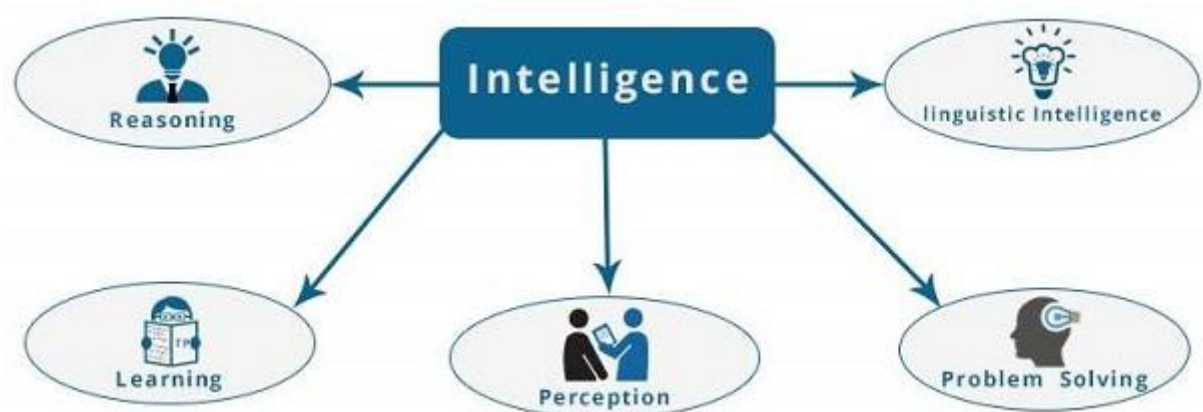
Inteligencia	Descripción	Ejemplo
Inteligencia lingüística	La capacidad de hablar, reconocer y usar mecanismos de fonología (sonidos del habla), sintaxis (gramática) y semántica (significado).	Narradores, oradores
Inteligencia musical	La capacidad de crear, comunicarse y comprender significados hechos de sonido, comprensión de tono, ritmo.	Músicos, cantantes, compositores
Inteligencia lógica-matemática	La capacidad de usar y comprender las relaciones en ausencia de acciones u objetos. Comprender ideas complejas y abstractas.	Matemáticos, científicos
Inteligencia espacial	La capacidad de percibir información visual o espacial, cambiarla y recrear imágenes visuales sin referencia a los objetos, construir imágenes en 3D y moverlas y rotarlas.	Lectores de mapas, astronautas, físicos
Inteligencia corporal-kinestésica	La capacidad de usar todo o parte del cuerpo para resolver problemas o productos de moda, controlar las habilidades motoras finas y gruesas, y manipular los objetos.	Jugadores, bailarines
Inteligencia intrapersonal	La capacidad de distinguir entre los propios sentimientos, intenciones y motivaciones.	Gautam Buddha
Inteligencia interpersonal	La capacidad de reconocer y hacer distinciones entre los sentimientos, creencias e intenciones de otras personas.	Comunicadores de masas, entrevistadores

Se puede decir que una máquina o un sistema es **artificialmente inteligente** cuando está equipado con al menos una y, como máximo, todas las inteligencias que contiene.

¿De qué está compuesta la inteligencia?

La inteligencia es intangible. Se compone de -

- Razonamiento
- Aprendizaje
- Resolución de problemas
- Percepción
- Inteligencia lingüística



Veamos brevemente todos los componentes:

- **Razonamiento** : es el conjunto de procesos que nos permite proporcionar una base para el juicio, la toma de decisiones y la predicción. En general, hay dos tipos:

Razonamiento inductivo	Razonamiento deductivo
Realiza observaciones específicas para hacer declaraciones generales amplias.	Comienza con una declaración general y examina las posibilidades de llegar a una conclusión específica y lógica.
Incluso si todas las premisas son verdaderas en una declaración, el razonamiento inductivo permite que la conclusión sea falsa.	Si algo es cierto para una clase de cosas en general, también es cierto para todos los miembros de

	esa clase.
Ejemplo: "Nita es maestra. Nita es estudiosa. Por lo tanto, todos los maestros son estudiosos".	Ejemplo: "Todas las mujeres mayores de 60 años son abuelas. Shalini tiene 65 años. Por lo tanto, Shalini es abuela".

- **Aprendizaje** : es la actividad de adquirir conocimiento o habilidad al estudiar, practicar, aprender o experimentar algo. El aprendizaje mejora la conciencia de los sujetos del estudio.

Los humanos, algunos animales y los sistemas habilitados para IA poseen la capacidad de aprender. El aprendizaje se clasifica como -

- **Aprendizaje auditivo** : es aprender escuchando y escuchando. Por ejemplo, estudiantes que escuchan conferencias de audio grabadas.
- **Aprendizaje episódico** : para aprender recordando secuencias de eventos que uno ha presenciado o experimentado. Esto es lineal y ordenado.
- **Aprendizaje motor** : es el aprendizaje mediante el movimiento preciso de los músculos. Por ejemplo, escoger objetos, escribir, etc.
- **Aprendizaje observacional** : para aprender observando e imitando a otros. Por ejemplo, el niño trata de aprender imitando a su padre.
- **Aprendizaje perceptivo** : es aprender a reconocer los estímulos que uno ha visto antes. Por ejemplo, identificar y clasificar objetos y situaciones.
- **Aprendizaje relacional** : implica aprender a diferenciar entre varios estímulos sobre la base de propiedades relacionales, en lugar de propiedades absolutas. Por ejemplo, agregar "un poco menos" de sal al momento de cocinar las papas que salieron saladas la última vez, cuando se cocinan agregando, digamos una cucharada de sal.
- **Aprendizaje espacial** : es aprender a través de estímulos visuales, como imágenes, colores, mapas, etc. Por ejemplo, una persona puede crear una hoja de ruta en mente antes de seguir la ruta.
- **Aprendizaje de estímulo-respuesta** : es aprender a realizar un comportamiento particular cuando está presente un cierto estímulo. Por ejemplo, un perro levanta la oreja al escuchar el timbre.
- **Resolución de problemas** : es el proceso en el que uno percibe e intenta llegar a una solución deseada a partir de una situación actual tomando un camino que está bloqueado por obstáculos conocidos o desconocidos.

La resolución de problemas también incluye la **toma de decisiones** , que es el proceso de seleccionar la mejor alternativa adecuada entre las múltiples alternativas disponibles para alcanzar la meta deseada.

- **Percepción** : es el proceso de adquisición, interpretación, selección y organización de información sensorial.

La percepción supone **sentir** . En los humanos, la percepción es ayudada por los órganos sensoriales. En el dominio de la IA, el mecanismo de percepción reúne los datos adquiridos por los sensores de manera significativa.

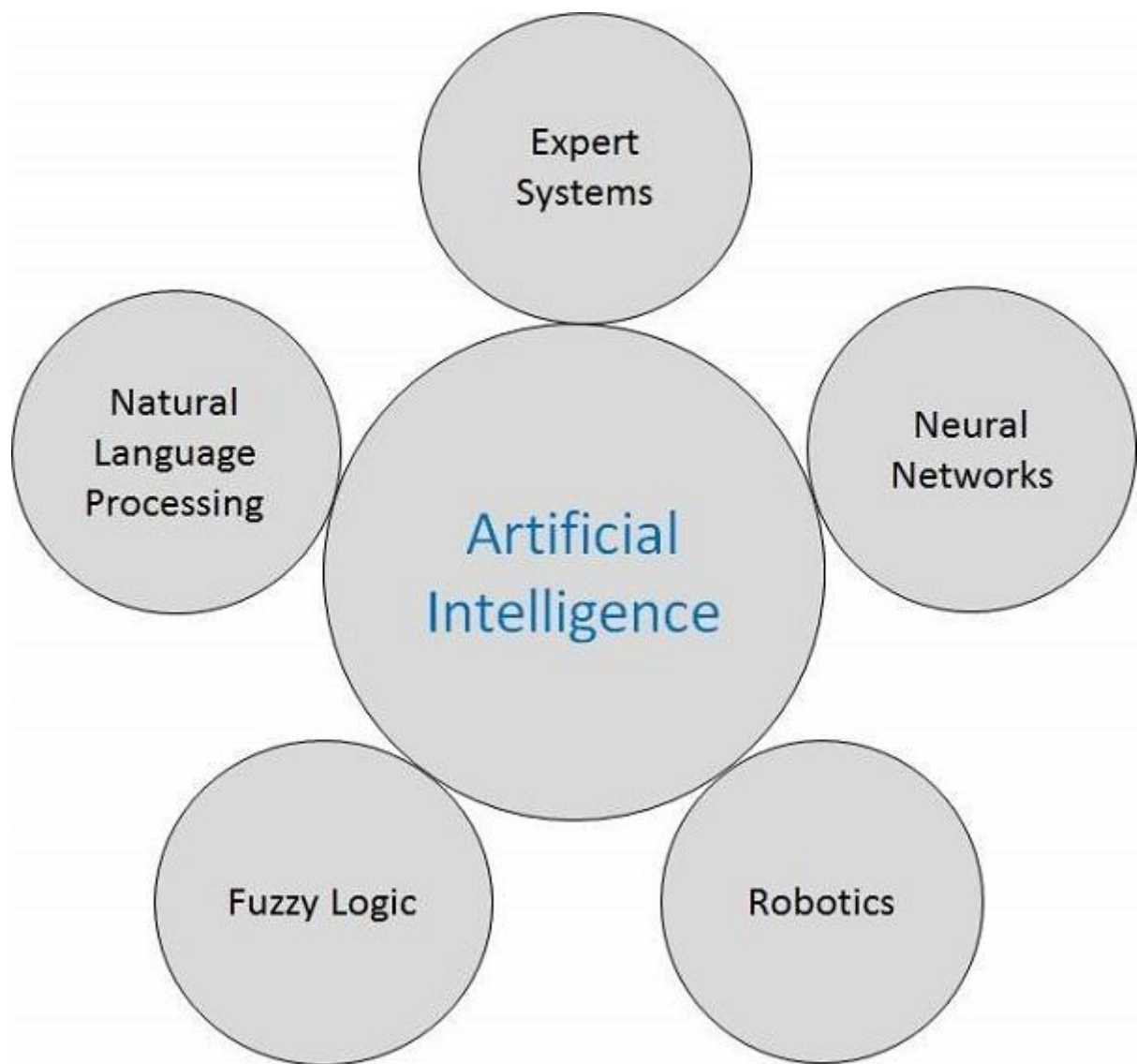
- **Inteligencia lingüística** : es la capacidad de uno para usar, comprender, hablar y escribir el lenguaje verbal y escrito. Es importante en la comunicación interpersonal.

Diferencia entre inteligencia humana y máquina

- Los humanos perciben por patrones, mientras que las máquinas perciben por conjunto de reglas y datos.
- Los humanos almacenan y recuperan información por patrones, las máquinas lo hacen buscando algoritmos. Por ejemplo, el número 40404040 es fácil de recordar, almacenar y recuperar, ya que su patrón es simple.
- Los humanos pueden descubrir el objeto completo incluso si falta una parte o está distorsionado; mientras que las máquinas no pueden hacerlo correctamente.

Inteligencia artificial - Áreas de investigación

El dominio de la inteligencia artificial es enorme en amplitud y anchura. Mientras avanzamos, consideramos las áreas de investigación ampliamente comunes y prósperas en el dominio de la IA:



Reconocimiento de voz y voz

Estos dos términos son comunes en robótica, sistemas expertos y procesamiento de lenguaje natural. Aunque estos términos se usan indistintamente, sus objetivos son diferentes.

Reconocimiento de voz	Reconocimiento de voz
El reconocimiento de voz tiene como objetivo comprender y comprender QUÉ se habló.	El objetivo del reconocimiento de voz es reconocer a la OMS que está hablando.
Se utiliza en computación manos libres, mapa o navegación de menús.	Se utiliza para identificar a una persona analizando su tono, tono de voz y acento, etc.
La máquina no necesita capacitación para el	Este sistema de reconocimiento necesita

reconocimiento de voz, ya que no depende del hablante.	capacitación, ya que está orientado a la persona.
Los sistemas de reconocimiento de voz independientes del hablante son difíciles de desarrollar.	Los sistemas de reconocimiento de voz dependientes del hablante son relativamente fáciles de desarrollar.

Funcionamiento de sistemas de reconocimiento de voz y voz

La entrada del usuario hablada en un micrófono va a la tarjeta de sonido del sistema. El convertidor convierte la señal analógica en señal digital equivalente para el procesamiento del habla. La base de datos se utiliza para comparar los patrones de sonido para reconocer las palabras. Finalmente, se da una retroalimentación inversa a la base de datos.

Este texto en el idioma de origen se convierte en entrada al Motor de traducción, que lo convierte en el texto del idioma de destino. Son compatibles con GUI interactiva, gran base de datos de vocabulario, etc.

Aplicaciones de la vida real de áreas de investigación

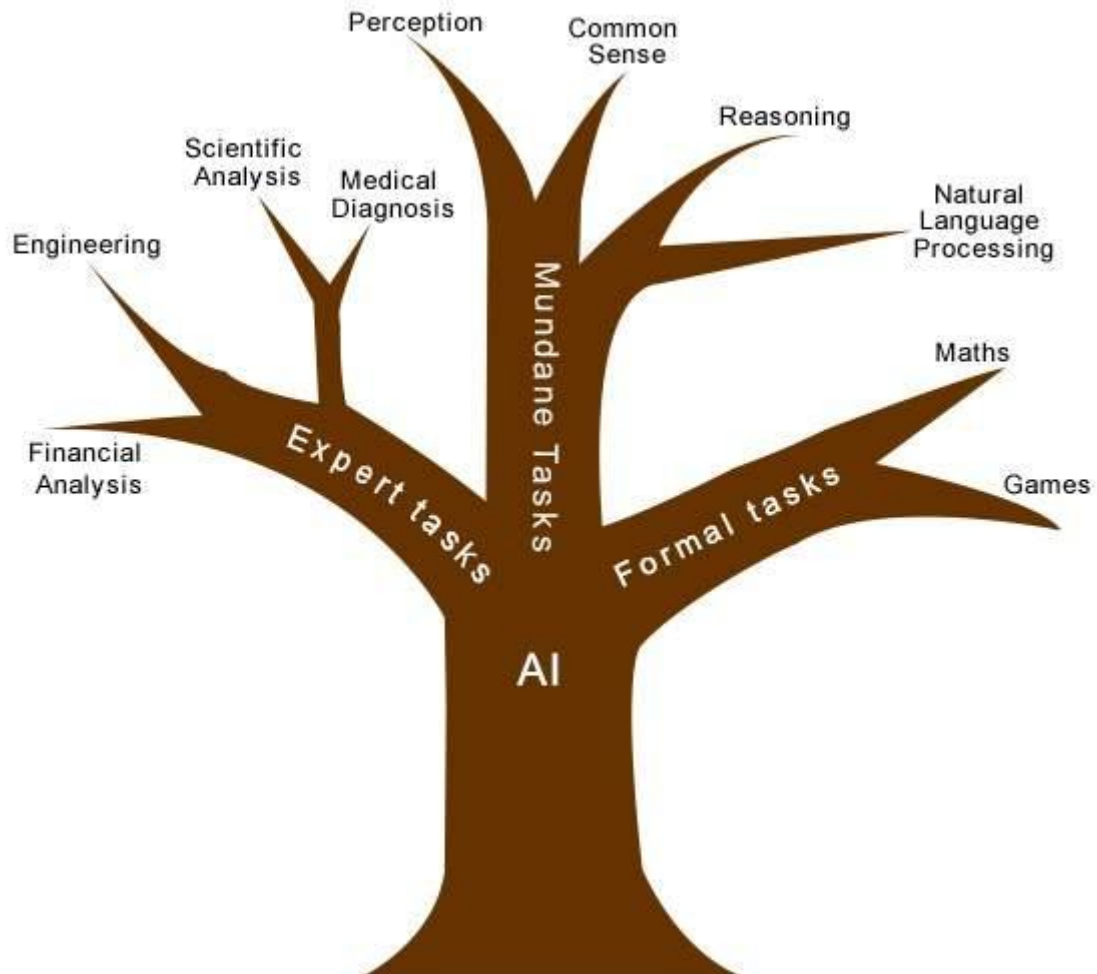
Hay una gran variedad de aplicaciones en las que la IA está sirviendo a personas comunes en su vida cotidiana:

No Señor.	Areas de investigacion	Aplicación de la vida real
1	Sistemas expertos Ejemplos: sistemas de seguimiento de vuelos, sistemas clínicos.	
2	Procesamiento natural del lenguaje Ejemplos: función Google Now, reconocimiento de voz, salida de voz automática.	

3	Redes neuronales Ejemplos: sistemas de reconocimiento de patrones como el reconocimiento facial, el reconocimiento de caracteres, el reconocimiento de escritura a mano.	
4 4	Robótica Ejemplos: robots industriales para mover, pulverizar, pintar, verificar con precisión, taladrar, limpiar, recubrir, tallar, etc.	
5 5	Sistemas de lógica difusa Ejemplos: electrónica de consumo, automóviles, etc.	

Clasificación de tareas de AI

El dominio de la IA se clasifica en **tareas formales**, **tareas mundanas** y **tareas de expertos**.



Dominios de tareas de inteligencia artificial		
Tareas mundanas (ordinarias)	Tareas formales	Tareas de expertos
Percepción <ul style="list-style-type: none"> Visión por computador Discurso, voz 	<ul style="list-style-type: none"> Matemáticas Geometría Lógica Integración y diferenciación 	<ul style="list-style-type: none"> Ingeniería Localización de averías Fabricación Supervisión
Procesamiento natural del lenguaje <ul style="list-style-type: none"> Comprensión Generación de lenguaje Traducción de idiomas 	Juegos <ul style="list-style-type: none"> Vamos Ajedrez (azul profundo) Ckeckers 	Análisis científico
Sentido común	Verificación	Análisis financiero

Razonamiento	Prueba de teorema	Diagnostico medico
Cepillado		Creatividad
Robótica <ul style="list-style-type: none"> • Locomotora 		

Los humanos aprenden **tareas mundanas (ordinarias)** desde su nacimiento. Aprenden mediante la percepción, el habla, el uso del lenguaje y las locomotoras. Aprenden Tareas formales y Tareas de expertos más tarde, en ese orden.

Para los humanos, las tareas mundanas son más fáciles de aprender. Lo mismo se consideró cierto antes de intentar implementar tareas mundanas en máquinas. Anteriormente, todo el trabajo de IA se concentraba en el dominio de la tarea mundana.

Más tarde, resultó que la máquina requiere más conocimiento, representación de conocimiento compleja y algoritmos complicados para manejar tareas mundanas. Esta es la razón **por la cual el trabajo de IA es más próspero en el dominio de Tareas Expertas** ahora, ya que el dominio de tareas experto necesita conocimiento experto sin sentido común, que puede ser más fácil de representar y manejar.

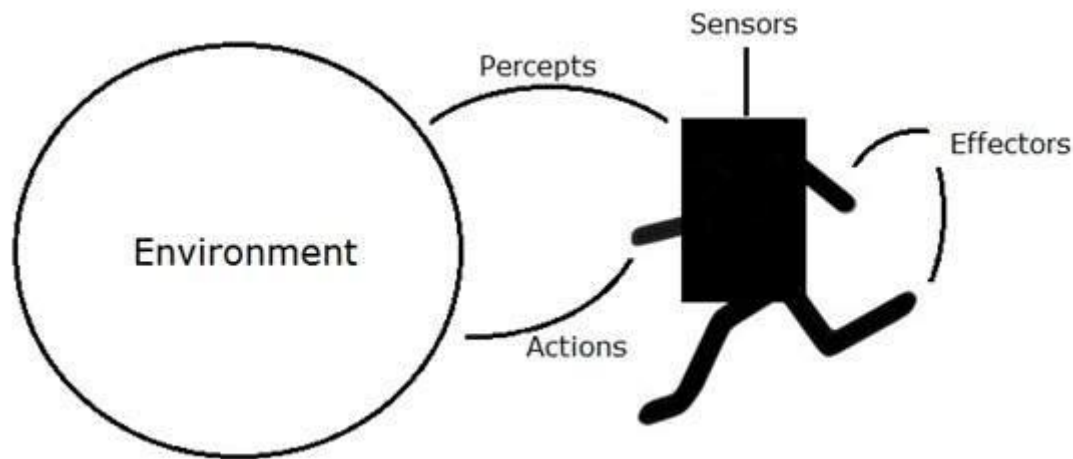
AI - Agentes y entornos

Un sistema de IA está compuesto por un agente y su entorno. Los agentes actúan en su entorno. El medio ambiente puede contener otros agentes.

¿Qué son Agente y Medio Ambiente?

Un **agente** es cualquier cosa que puede percibir su entorno a través de **sensores** y actúa sobre ese entorno a través de **efectores**.

- Un **agente humano** tiene órganos sensoriales como ojos, oídos, nariz, lengua y piel paralelos a los sensores, y otros órganos como manos, piernas, boca, para efectores.
- Un **agente robótico** reemplaza cámaras y buscadores de rango infrarrojo para los sensores, y varios motores y actuadores para efectores.
- Un **agente de software** ha codificado cadenas de bits como sus programas y acciones.



Terminología del agente

- **Medida de rendimiento del agente** : es el criterio que determina el éxito de un agente.
- **Comportamiento del agente** : es la acción que realiza el agente después de cualquier secuencia de percepciones.
- **Percepción** : son las entradas perceptivas del agente en una instancia determinada.
- **Percepción de secuencia** : es la historia de todo lo que un agente ha percibido hasta la fecha.
- **Función del agente** : es un mapa de la secuencia de preceptos a una acción.

Racionalidad

La racionalidad no es más que el estado de ser razonable, sensible y tener buen sentido del juicio.

La racionalidad tiene que ver con las acciones y resultados esperados dependiendo de lo que el agente haya percibido. Realizar acciones con el objetivo de obtener información útil es una parte importante de la racionalidad.

¿Qué es el agente racional ideal?

Un agente racional ideal es el que es capaz de realizar las acciones esperadas para maximizar su medida de rendimiento, sobre la base de:

- Su secuencia de percepción
- Su base de conocimiento incorporada

La racionalidad de un agente depende de lo siguiente:

- Las **medidas de rendimiento** , que determinan el grado de éxito.
- **Secuencia de percepción** del agente hasta ahora.

- El **conocimiento previo** del agente **sobre el medio ambiente** .
- Las **acciones** que el agente puede llevar a cabo.

Un agente racional siempre realiza la acción correcta, donde la acción correcta significa la acción que hace que el agente sea más exitoso en la secuencia de percepción dada. El problema que resuelve el agente se caracteriza por la Medida de rendimiento, el entorno, los actuadores y los sensores (PEAS).

La estructura de los agentes inteligentes

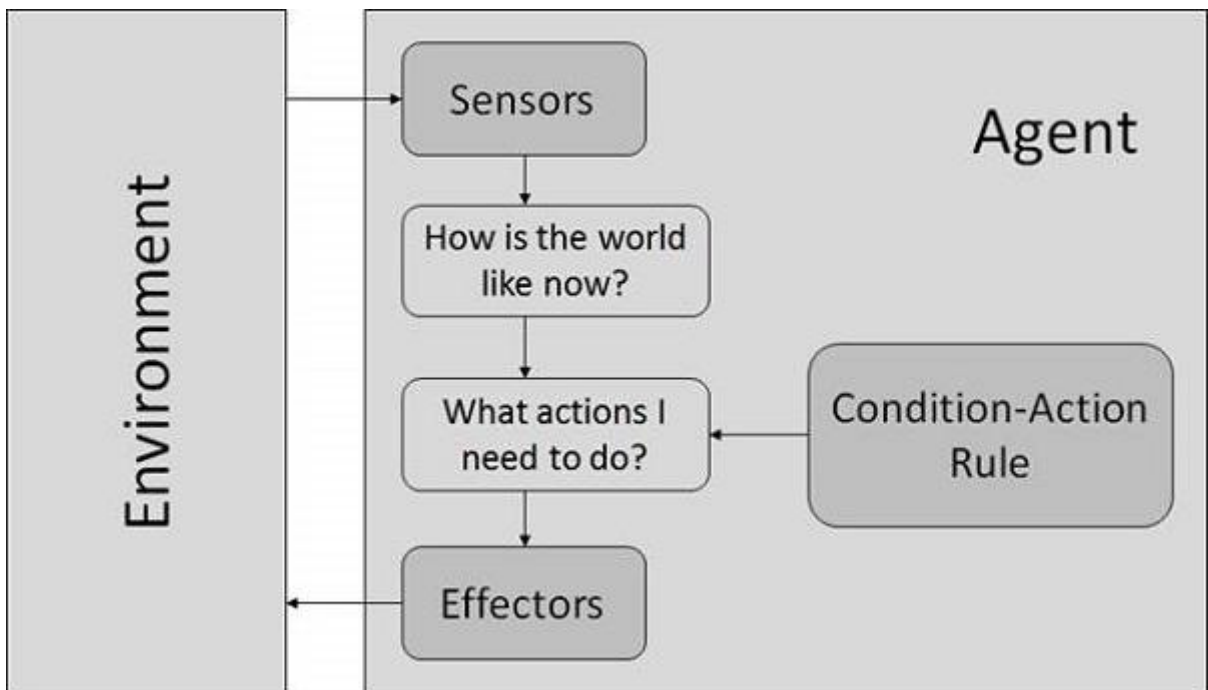
La estructura del agente se puede ver como:

- Agente = Arquitectura + Programa de agente
- Arquitectura = la maquinaria en la que se ejecuta un agente.
- Programa de agente = una implementación de una función de agente.

Agentes reflejos simples

- Eligen acciones solo en función de la percepción actual.
- Son racionales solo si se toma una decisión correcta sobre la base del precepto actual.
- Su entorno es completamente observable.

Regla de condición-acción : es una regla que asigna un estado (condición) a una acción.



Agentes reflejos basados en modelos

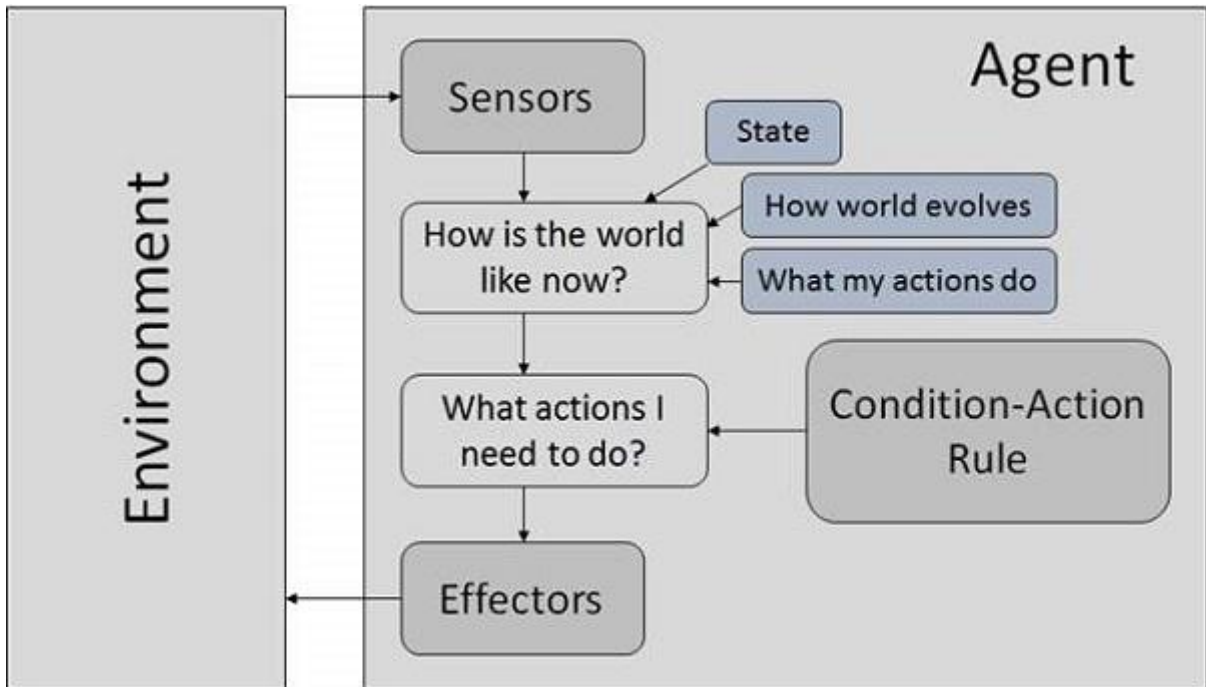
Usan un modelo del mundo para elegir sus acciones. Mantienen un estado interno.

Modelo : conocimiento sobre "cómo suceden las cosas en el mundo".

Estado interno : es una representación de aspectos no observados del estado actual según el historial de percepción.

Actualizar el estado requiere la información sobre:

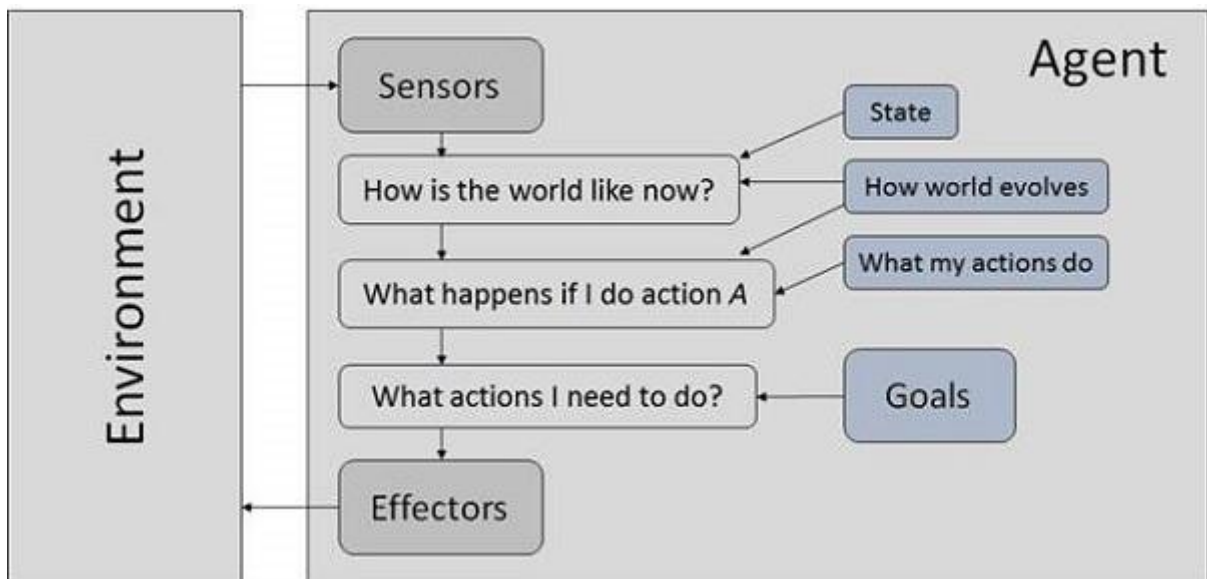
- Cómo evoluciona el mundo.
- Cómo las acciones del agente afectan al mundo.



Agentes basados en objetivos

Eligen sus acciones para lograr objetivos. El enfoque basado en objetivos es más flexible que el agente reflejo, ya que el conocimiento que respalda una decisión se modela explícitamente, lo que permite modificaciones.

Objetivo : es la descripción de situaciones deseables.

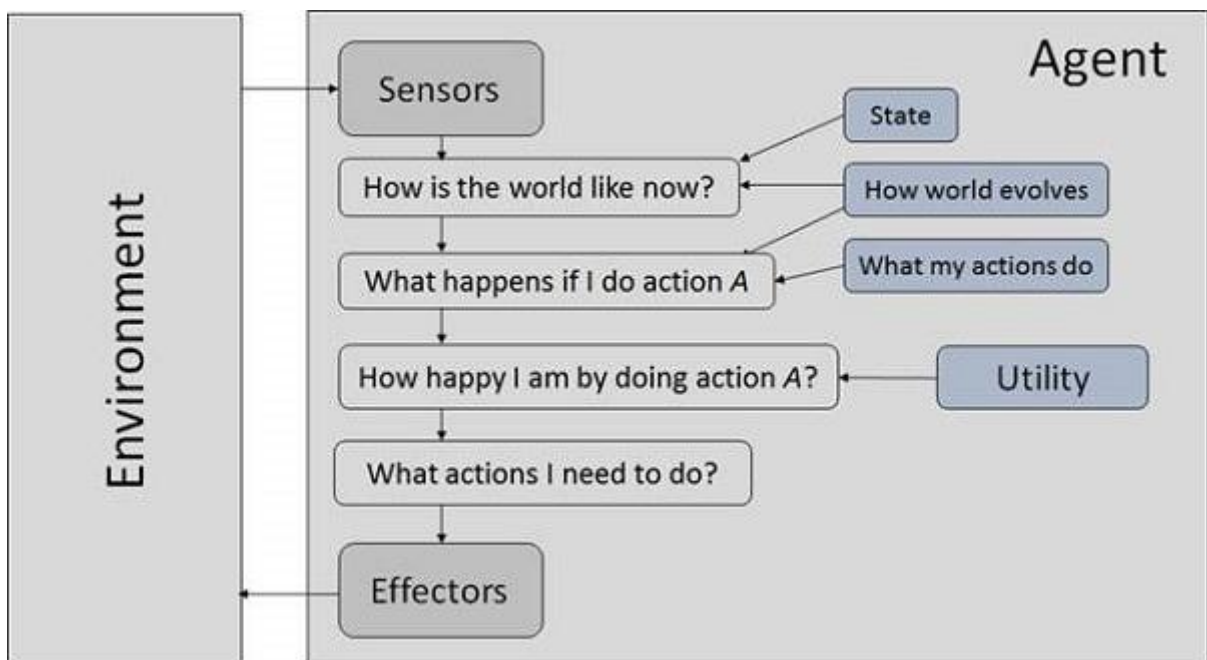


Agentes basados en servicios públicos

Eligen acciones basadas en una preferencia (utilidad) para cada estado.

Los objetivos son inadecuados cuando:

- Hay objetivos en conflicto, de los cuales solo unos pocos se pueden lograr.
- Los objetivos tienen cierta incertidumbre de ser alcanzados y debe sopesar la probabilidad de éxito con la importancia de un objetivo.



La naturaleza de los ambientes

Algunos programas operan en **un entorno** completamente **artificial** limitado a la entrada del teclado, la base de datos, los sistemas de archivos de computadora y la salida de caracteres en una pantalla.

En contraste, algunos agentes de software (robots de software o softbots) existen en dominios de softbots ricos e ilimitados. El simulador tiene un **entorno complejo muy detallado**. El agente de software debe elegir entre una gran variedad de acciones en tiempo real. Un softbot diseñado para escanear las preferencias en línea del cliente y mostrar elementos interesantes para el cliente funciona tanto en un entorno **real** como **artificial**.

El **entorno artificial** más famoso es el **entorno de prueba de Turing**, en el que un agente artificial real y otro se prueban en igualdad de condiciones. Este es un entorno muy desafiante ya que es muy difícil que un agente de software funcione tan bien como un ser humano.

Prueba de Turing

El éxito de un comportamiento inteligente de un sistema se puede medir con Turing Test.

Dos personas y una máquina a evaluar participan en la prueba. De las dos personas, una desempeña el papel del probador. Cada uno de ellos se sienta en habitaciones diferentes. El probador no sabe quién es la máquina y quién es un humano. Él interroga las preguntas escribiendo y enviándolas a ambas inteligencias, a las que recibe respuestas escritas.

Esta prueba tiene como objetivo engañar al probador. Si el probador no puede determinar la respuesta de la máquina a partir de la respuesta humana, entonces se dice que la máquina es inteligente.

Propiedades del medio ambiente

El medio ambiente tiene propiedades múltiples:

- **Discreto / continuo** : si hay un número limitado de estados del entorno distintos, claramente definidos, el entorno es discreto (por ejemplo, ajedrez); de lo contrario es continuo (por ejemplo, conducir).
- **Observable / parcialmente observable** : si es posible determinar el estado completo del entorno en cada punto de tiempo desde las percepciones, es observable; de lo contrario, solo es parcialmente observable.
- **Estático / Dinámico** : si el entorno no cambia mientras un agente está actuando, entonces es estático; de lo contrario es dinámico.
- **Agente único / agentes múltiples** : el entorno puede contener otros agentes que pueden ser del mismo tipo o de un tipo diferente al del agente.
- **Accesible / Inaccesible** : si el aparato sensorial del agente puede tener acceso al estado completo del entorno, el agente tendrá acceso al entorno.
- **Determinista / no determinista** : si el siguiente estado del entorno está completamente determinado por el estado actual y las acciones del agente, entonces el entorno es determinista; de lo contrario no es determinista.
- **Episódico / No episódico** : en un entorno episódico, cada episodio consiste en que el agente perciba y luego actúe. La calidad de su acción depende solo del episodio en sí. Los episodios posteriores no dependen de las acciones en los episodios anteriores. Los entornos episódicos son mucho más simples porque el agente no necesita pensar en el futuro.

AI - Algoritmos de búsqueda populares

La búsqueda es la técnica universal de resolución de problemas en IA. Hay algunos juegos para un solo jugador, como juegos de fichas, Sudoku, crucigramas, etc. Los algoritmos de búsqueda te ayudan a buscar una posición particular en dichos juegos.

Problemas de búsqueda de agente único

Los juegos como los acertijos 3X3 de ocho fichas, 4X4 de quince fichas y 5X5 de veinticuatro fichas son desafíos para encontrar la ruta de un solo agente. Consisten en una matriz de mosaicos con un mosaico en blanco. El jugador debe organizar las fichas deslizando una ficha vertical u horizontalmente en un espacio en blanco con el objetivo de lograr algún objetivo.

Los otros ejemplos de problemas de búsqueda de agente único son el Problema del vendedor ambulante, el Cubo de Rubik y la Prueba de teorema.

Buscar terminología

- **Espacio del problema** : es el entorno en el que se realiza la búsqueda. (Un conjunto de estados y un conjunto de operadores para cambiar esos estados)
- **Instancia del problema** : es el estado inicial + el estado del objetivo.
- **Gráfico del espacio del problema** : representa el estado del problema. Los estados se muestran mediante nodos y los operadores se muestran mediante aristas.
- **Profundidad de un problema** : longitud de una ruta más corta o secuencia más corta de operadores desde el estado inicial hasta el estado objetivo.
- **Complejidad espacial** : el número máximo de nodos que se almacenan en la memoria.
- **Complejidad de tiempo** : la cantidad máxima de nodos que se crean.
- **Admisibilidad** : propiedad de un algoritmo para encontrar siempre una solución óptima.
- **Factor de ramificación** : el número promedio de nodos secundarios en el gráfico del espacio del problema.
- **Profundidad** : longitud de la ruta más corta desde el estado inicial hasta el estado objetivo.

Estrategias de búsqueda de fuerza bruta

Son más simples, ya que no necesitan ningún conocimiento específico de dominio. Funcionan bien con un pequeño número de estados posibles.

Requerimientos -

- Descripción del estado

- Un conjunto de operadores válidos.
- Estado inicial
- Descripción del estado del objetivo

Breadth-First Search

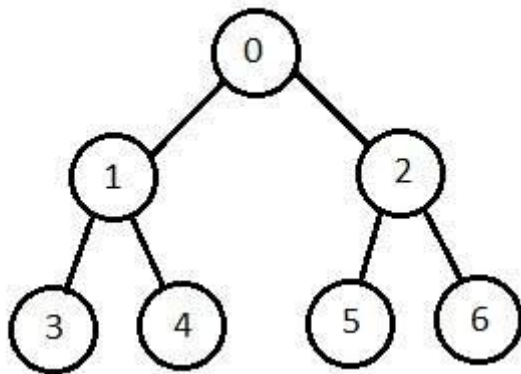
Comienza desde el nodo raíz, primero explora los nodos vecinos y avanza hacia los vecinos del siguiente nivel. Genera un árbol a la vez hasta que se encuentra la solución. Se puede implementar utilizando la estructura de datos de cola FIFO. Este método proporciona el camino más corto a la solución.

Si el **factor de ramificación** (número promedio de nodos secundarios para un nodo dado) = b y profundidad = d , entonces el número de nodos en el nivel $d = b^d$.

El número total de nodos creados en el peor de los casos es $b + b^2 + b^3 + \dots + b^d$.

Desventaja : dado que cada nivel de nodos se guarda para crear el siguiente, consume mucho espacio de memoria. El espacio requerido para almacenar nodos es exponencial.

Su complejidad depende del número de nodos. Puede verificar nodos duplicados.



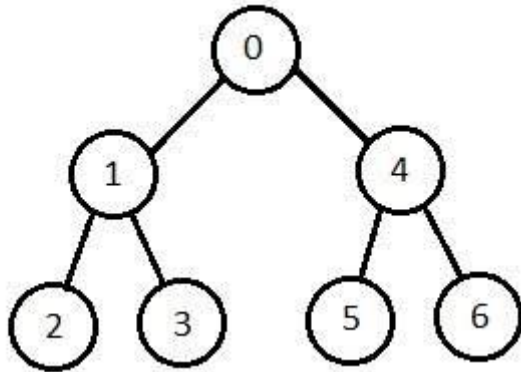
Profundidad-Primera búsqueda

Se implementa en recursividad con la estructura de datos de la pila LIFO. Crea el mismo conjunto de nodos que el método Breadth-First, solo en un orden diferente.

Como los nodos en la ruta única se almacenan en cada iteración desde el nodo raíz hasta el nodo hoja, el requisito de espacio para almacenar nodos es lineal. Con el factor de ramificación b y la profundidad como m , el espacio de almacenamiento es bm .

Desventaja : este algoritmo puede no terminar y continuar infinitamente en una ruta. La solución a este problema es elegir una profundidad de corte. Si el corte ideal es d , y si el corte elegido es menor que d , entonces este algoritmo puede fallar. Si el límite elegido es mayor que d , entonces el tiempo de ejecución aumenta.

Su complejidad depende del número de caminos. No puede verificar nodos duplicados.



Búsqueda bidireccional

Busca hacia adelante desde el estado inicial y hacia atrás desde el estado objetivo hasta que ambos se encuentran para identificar un estado común.

La ruta desde el estado inicial se concatena con la ruta inversa desde el estado objetivo. Cada búsqueda se realiza solo hasta la mitad de la ruta total.

Búsqueda uniforme de costos

La clasificación se realiza al aumentar el costo de la ruta a un nodo. Siempre expande el nodo de menor costo. Es idéntico a Breadth First search si cada transición tiene el mismo costo.

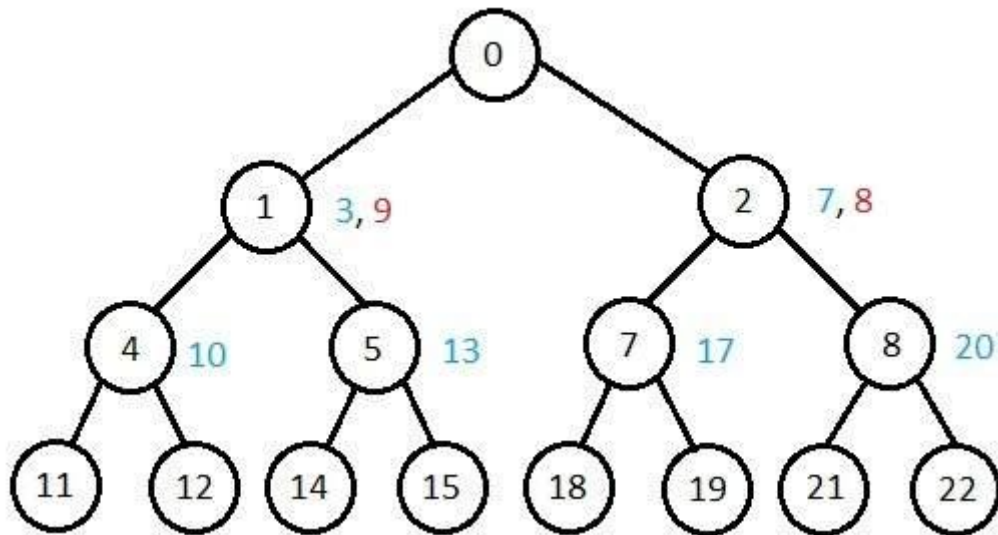
Explora caminos en el creciente orden de costo.

Desventaja : puede haber múltiples rutas largas con el costo $\leq C^*$. La búsqueda de costos uniformes debe explorarlos a todos.

Profundidad de profundización iterativa: primera búsqueda

Realiza una búsqueda de profundidad primero al nivel 1, comienza de nuevo, ejecuta una búsqueda completa de profundidad primero al nivel 2 y continúa de esta manera hasta que se encuentra la solución.

Nunca crea un nodo hasta que se generan todos los nodos inferiores. Solo guarda una pila de nodos. El algoritmo termina cuando encuentra una solución en profundidad d . El número de nodos creados en la profundidad d es b^d y en la profundidad $d-1$ es b^{d-1} .



Comparación de diversas complejidades de algoritmos

Veamos el rendimiento de los algoritmos basados en varios criterios:

Criterio	Anchura primero	Profundidad primero	Bidireccional	Costo uniforme	Profundización interactiva
Hora	b^d	b^m	$b^{d/2}$	b^d	b^d
Espacio	b^d	b^m	$b^{d/2}$	b^d	b^d
Óptima	si	No	si	si	si
Lo completo	si	No	si	si	si

Estrategias de búsqueda informadas (heurísticas)

Para resolver grandes problemas con una gran cantidad de estados posibles, se necesita agregar conocimiento específico del problema para aumentar la eficiencia de los algoritmos de búsqueda.

Funciones de evaluación heurística

Calculan el costo de la ruta óptima entre dos estados. Una función heurística para los juegos de fichas deslizantes se calcula contando el número de movimientos que realiza cada ficha desde su estado objetivo y agregando esta cantidad de movimientos para todas las fichas.

Búsqueda heurística pura

Expande los nodos en el orden de sus valores heurísticos. Crea dos listas, una lista cerrada para los nodos ya expandidos y una lista abierta para los nodos creados pero no expandidos.

En cada iteración, se expande un nodo con un valor heurístico mínimo, todos sus nodos secundarios se crean y se colocan en la lista cerrada. Luego, la función heurística se aplica a los nodos secundarios y se colocan en la lista abierta de acuerdo con su valor heurístico. Los caminos más cortos se guardan y los más largos se eliminan.

Una búsqueda

Es la forma más conocida de Best First search. Evita expandir rutas que ya son caras, pero primero expande las rutas más prometedoras.

$f(n) = g(n) + h(n)$, donde

- $g(n)$ el costo (hasta ahora) para llegar al nodo
- $h(n)$ costo estimado para llegar del nodo a la meta
- $f(n)$ costo total estimado de la ruta a través de n hasta la meta. Se implementa usando la cola de prioridad aumentando $f(n)$.

Greedy Best First Search

Expande el nodo que se estima más cercano al objetivo. Expande los nodos en función de $f(n) = h(n)$. Se implementa utilizando la cola de prioridad.

Desventaja : puede atascarse en bucles. No es optimo.

Algoritmos de búsqueda local

Empiezan desde una solución prospectiva y luego pasan a una solución vecina. Pueden devolver una solución válida incluso si se interrumpe en cualquier momento antes de que finalicen.

Búsqueda de escalada

Es un algoritmo iterativo que comienza con una solución arbitraria a un problema e intenta encontrar una mejor solución cambiando un solo elemento de la solución de forma incremental. Si el cambio produce una mejor solución, se toma un cambio incremental como una nueva solución. Este proceso se repite hasta que no haya más mejoras.

función Hill-Climbing (problema), devuelve un estado que es un máximo local.

```
inputs: problem, a problem
local variables: current, a node
                 neighbor, a node
current <- Make_Node(Initial-State[problem])
loop
  do neighbor <- a highest_valued successor of current
  if Value[neighbor] ≤ Value[current] then
```

```

    return State[current]
    current <- neighbor
end

```

Desventaja : este algoritmo no es completo ni óptimo.

Búsqueda de haz local

En este algoritmo, contiene k número de estados en un momento dado. Al principio, estos estados se generan aleatoriamente. Los sucesores de estos k estados se calculan con la ayuda de la función objetivo. Si alguno de estos sucesores es el valor máximo de la función objetivo, entonces el algoritmo se detiene.

De lo contrario, los estados (k iniciales y k número de sucesores de los estados = 2k) se colocan en un grupo. El grupo se ordena numéricamente. Los k estados más altos se seleccionan como nuevos estados iniciales. Este proceso continúa hasta que se alcanza un valor máximo.

La función BeamSearch (*problema*, *k*), devuelve un estado de solución.

```

start with k randomly generated states
loop
    generate all successors of all k states
    if any of the states = solution, then return the state
    else select the k best successors
end

```

Recocido simulado

El recocido es el proceso de calentar y enfriar un metal para cambiar su estructura interna y modificar sus propiedades físicas. Cuando el metal se enfría, se agarra su nueva estructura y el metal conserva sus propiedades recién obtenidas. En el proceso de recocido simulado, la temperatura se mantiene variable.

Inicialmente configuramos la temperatura alta y luego permitimos que se 'enfrie' lentamente a medida que avanza el algoritmo. Cuando la temperatura es alta, el algoritmo puede aceptar soluciones peores con alta frecuencia.

comienzo

- Inicializar $k = 0$; L = número entero de variables;
- Desde $i \rightarrow j$, busque la diferencia de rendimiento Δ .
- Si $\Delta \leq 0$, acepte otra cosa si $\exp(-\Delta / T(k)) > \text{random}(0,1)$ luego acepte;
- Repita los pasos 1 y 2 para los pasos $L(k)$.
- $k = k + 1$;

Repita los pasos del 1 al 4 hasta que se cumplan los criterios.

Final

Problema de vendedor ambulante

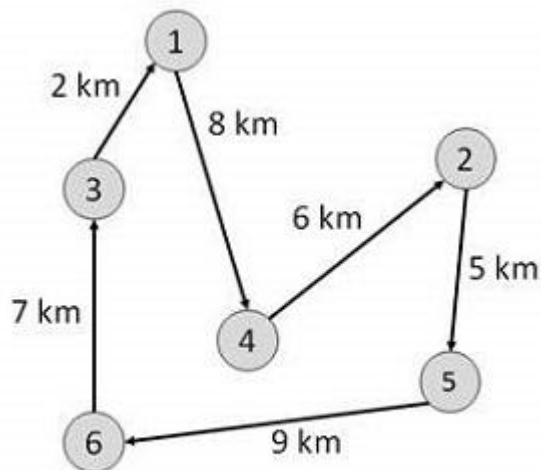
En este algoritmo, el objetivo es encontrar un recorrido de bajo costo que comience desde una ciudad, visite todas las ciudades en ruta exactamente una vez y termine en la misma ciudad de inicio.

Start

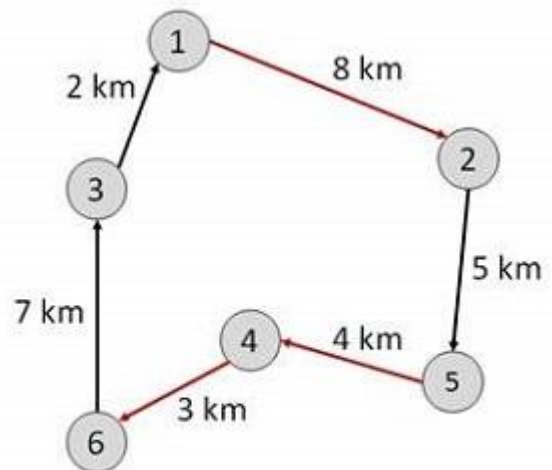
Find out all $(n - 1)!$ Possible solutions, where n is the total number of cities.

Determine the minimum cost by finding out the cost of each of these $(n - 1)!$ solutions.

Finally, keep the one with the minimum cost.
end



Total Distance = 37km



Total Distance = 31km

Inteligencia artificial - Sistemas de lógica difusa

Los sistemas de lógica difusa (FLS) producen resultados aceptables pero definidos en respuesta a entradas incompletas, ambiguas, distorsionadas o inexactas (difusas).

¿Qué es la lógica difusa?

Fuzzy Logic (FL) es un método de razonamiento que se asemeja al razonamiento humano. El enfoque de FL imita la forma de toma de decisiones en humanos que involucra todas las posibilidades intermedias entre los valores digitales SÍ y NO.

El bloque lógico convencional que una computadora puede entender toma una entrada precisa y produce una salida definida como VERDADERO o FALSO, que es equivalente al SÍ o NO del ser humano.

El inventor de la lógica difusa, Lotfi Zadeh, observó que, a diferencia de las computadoras, la toma de decisiones humanas incluye una gama de posibilidades entre SÍ y NO, como:

CIERTAMENTE SI
POSIBLEMENTE SÍ
NO SE PUEDE DECIR
POSIBLEMENTE NO
CIERTAMENTE NO

La lógica difusa funciona en los niveles de posibilidades de entrada para lograr la salida definida.

Implementación

- Se puede implementar en sistemas con varios tamaños y capacidades que van desde pequeños microcontroladores hasta grandes sistemas de control basados en estaciones de trabajo en red.
- Se puede implementar en hardware, software o una combinación de ambos.

¿Por qué Fuzzy Logic?

La lógica difusa es útil para fines comerciales y prácticos.

- Puede controlar máquinas y productos de consumo.
- Es posible que no proporcione un razonamiento preciso, sino un razonamiento aceptable.
- La lógica difusa ayuda a lidiar con la incertidumbre en la ingeniería.

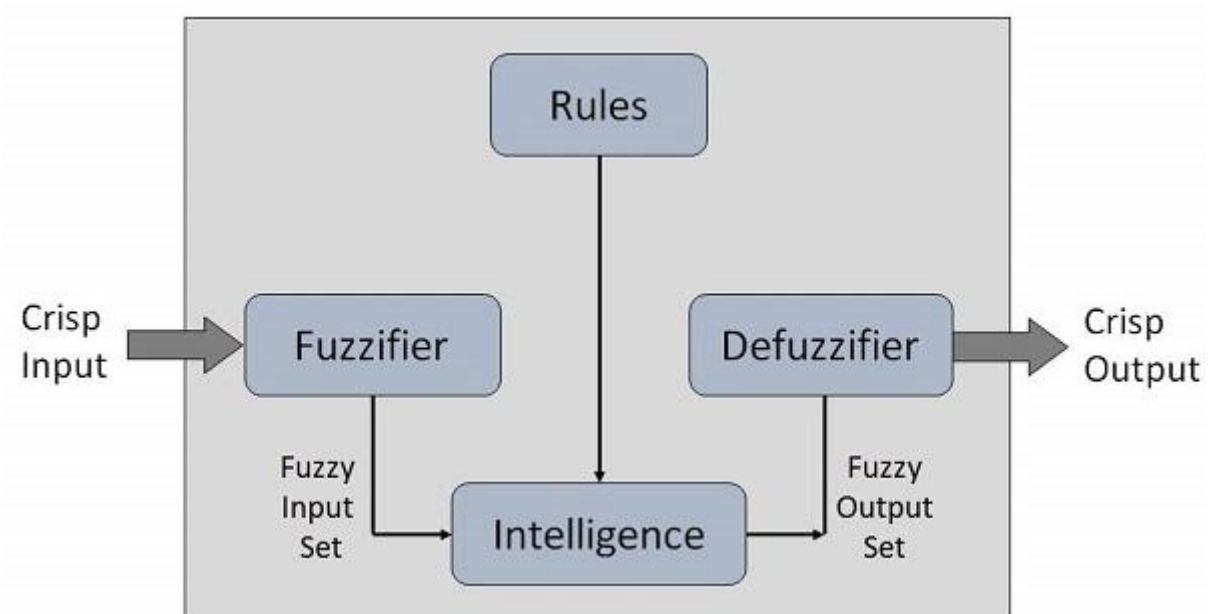
Arquitectura de sistemas de lógica difusa

Tiene cuatro partes principales como se muestra:

- **Módulo de fuzzificación** : transforma las entradas del sistema, que son números nítidos, en conjuntos difusos. Divide la señal de entrada en cinco pasos, tales como:

LP	x es grande positivo
MP	x es medio positivo
S	x es pequeño
Minnesota	x es medio negativo
LN	x es grande negativo

- **Base de conocimiento** : almacena las reglas IF-THEN proporcionadas por expertos.
- **Motor de inferencia** : simula el proceso de razonamiento humano al hacer una inferencia difusa en las entradas y las reglas IF-THEN.
- **Módulo de defuzzificación** : transforma el conjunto difuso obtenido por el motor de inferencia en un valor nítido.



Las **funciones de membresía funcionan en** conjuntos difusos de variables.

Función de la membresía

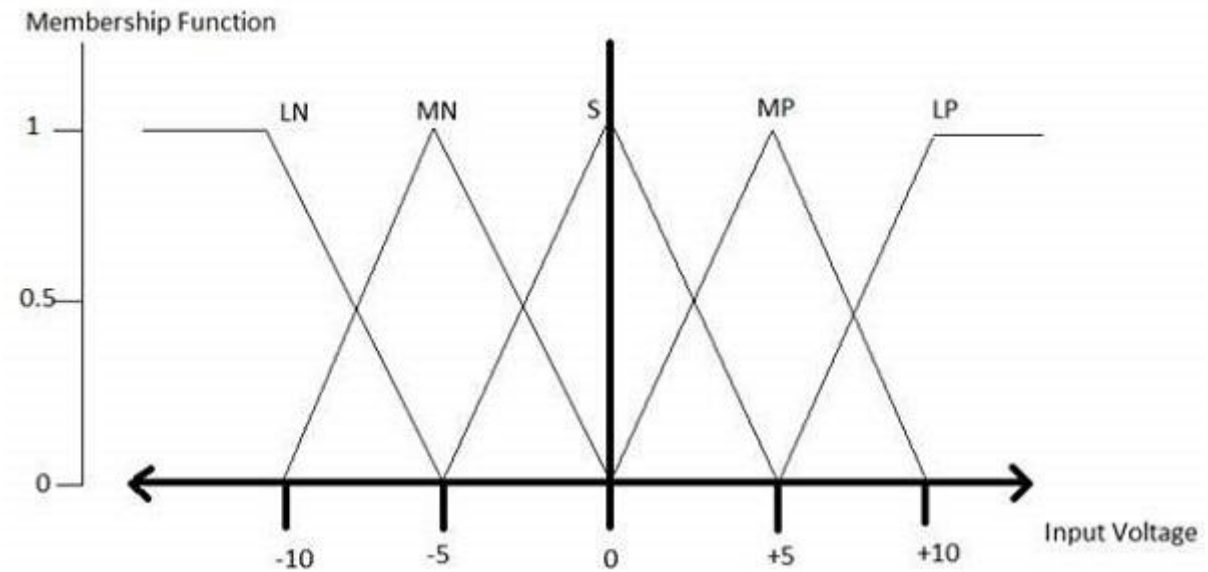
Las funciones de membresía le permiten cuantificar el término lingüístico y representar un conjunto difuso gráficamente. Una **función de pertenencia** para un *conjunto* difuso A en el universo del discurso X se define como $\mu_A : X \rightarrow [0, 1]$.

Aquí, cada elemento de X se asigna a un valor entre 0 y 1. Se llama **valor de membresía** o **grado de membresía**. Se cuantifica el grado de pertenencia del elemento en X al conjunto difuso A .

- El eje x representa el universo del discurso.
- El eje y representa los grados de membresía en el intervalo $[0, 1]$.

Puede haber múltiples funciones de membresía aplicables para difuminar un valor numérico. Las funciones de membresía simples se utilizan ya que el uso de funciones complejas no agrega más precisión en la salida.

Todas las funciones de membresía para **LP, MP, S, MN y LN** se muestran a continuación:

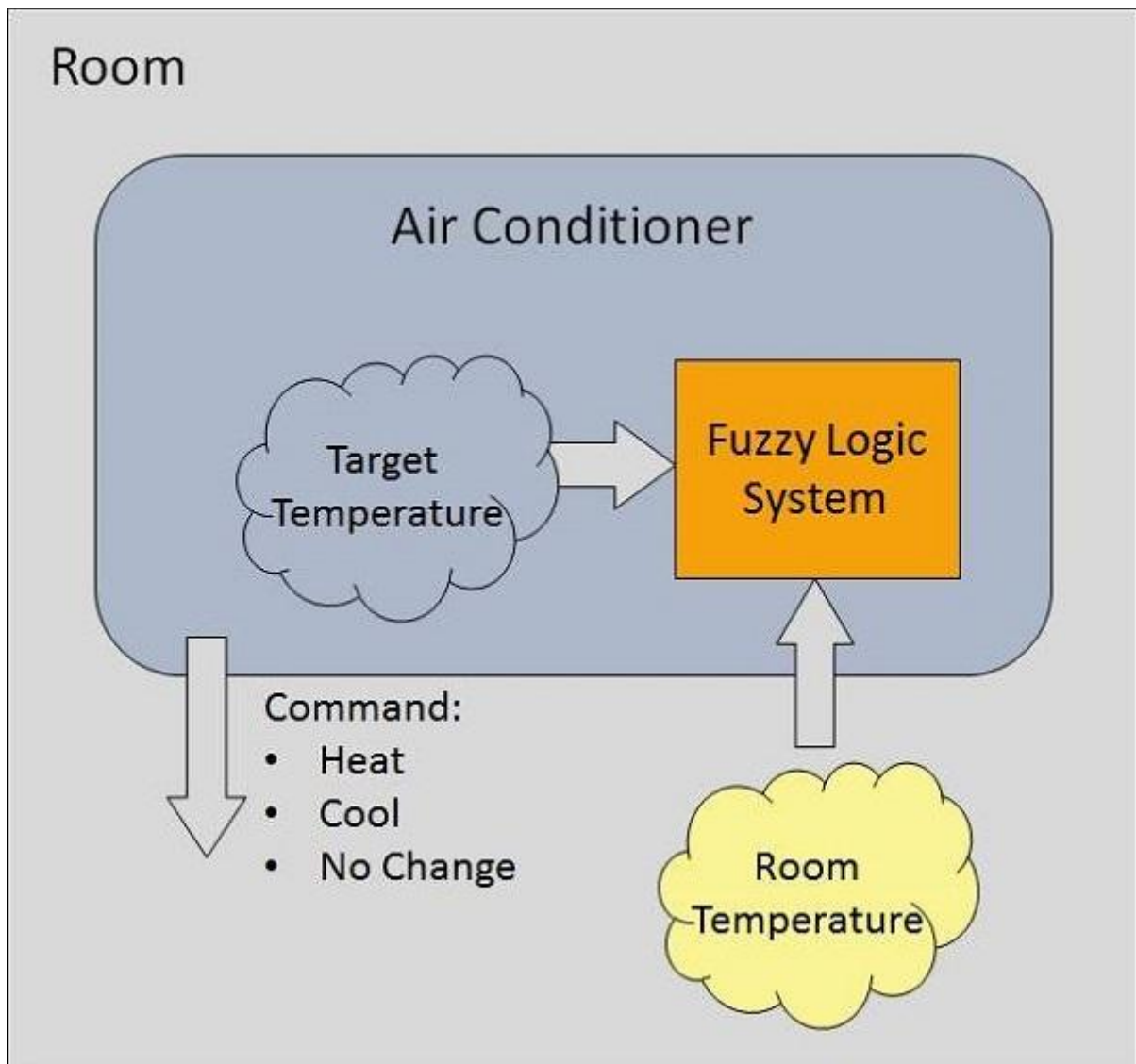


Las formas de función de membresía triangular son más comunes entre varias otras formas de función de membresía como trapezoidal, singleton y gaussiana.

Aquí, la entrada al fuzzificador de 5 niveles varía de -10 voltios a +10 voltios. Por lo tanto, la salida correspondiente también cambia.

Ejemplo de un sistema de lógica difusa

Consideremos un sistema de aire acondicionado con un sistema de lógica difusa de 5 niveles. Este sistema ajusta la temperatura del aire acondicionado comparando la temperatura ambiente y el valor de temperatura objetivo.



Algoritmo

- Definir variables y términos lingüísticos (inicio)
- Construir funciones de membresía para ellos. (comienzo)
- Construir una base de conocimiento de reglas (inicio)
- Convierta datos nítidos en conjuntos de datos difusos utilizando funciones de membresía. (fuzzificación)
- Evaluar reglas en la base de reglas. (Máquina de inferencia)
- Combina los resultados de cada regla. (Máquina de inferencia)
- Convierta los datos de salida en valores no difusos. (despuzzificación)

Desarrollo

Paso 1 - Definir variables y términos lingüísticos

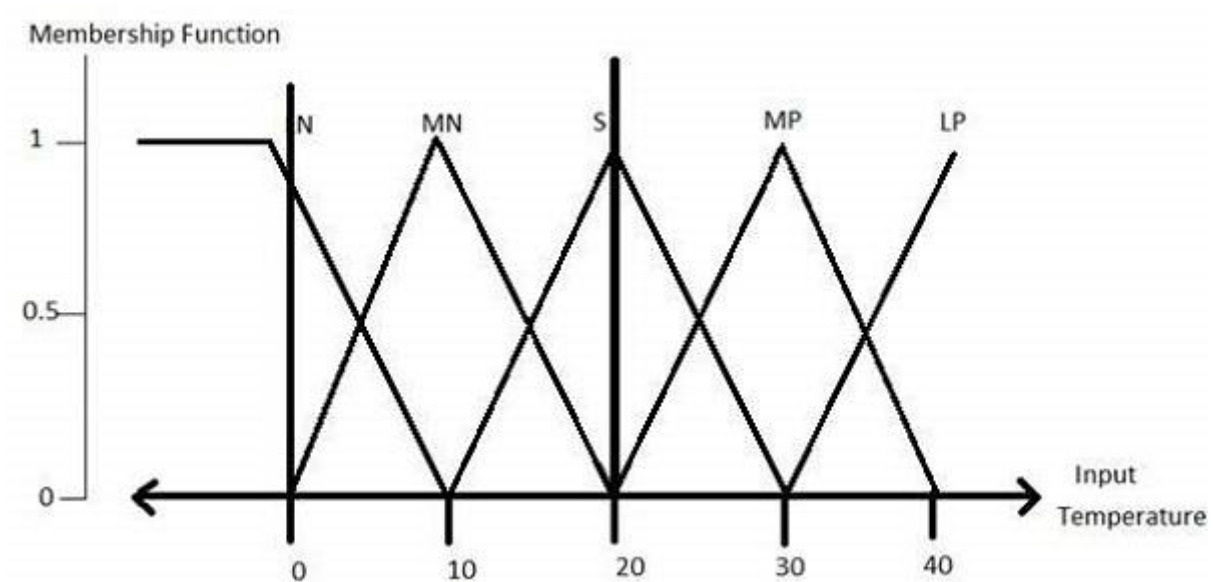
Las variables lingüísticas son variables de entrada y salida en forma de palabras u oraciones simples. Para temperatura ambiente, frío, cálido, caliente, etc., son términos lingüísticos.

Temperatura (t) = {muy frío, frío, cálido, muy cálido, caliente}

Cada miembro de este conjunto es un término lingüístico y puede cubrir una parte de los valores de temperatura general.

Paso 2: construya funciones de membresía para ellos

Las funciones de pertenencia de la variable de temperatura son como se muestran:



Paso 3 - Construir reglas de base de conocimiento

Cree una matriz de valores de temperatura ambiente versus valores de temperatura objetivo que se espera que proporcione un sistema de aire acondicionado.

Temperatura ambiente. /Objetivo	Muy frío	Frío	Calentar	Caliente	Muy_caliente
Muy frío	Ningún cambio	Calor	Calor	Calor	Calor
Frío	Frio	Ningún cambio	Calor	Calor	Calor
Calentar	Frio	Frio	Ningún cambio	Calor	Calor
Caliente	Frio	Frio	Frio	Ningún cambio	Calor

Muy_caliente	Frio	Frio	Frio	Frio	Ningún cambio
--------------	------	------	------	------	---------------

Construya un conjunto de reglas en la base de conocimiento en forma de estructuras IF-THEN-ELSE.

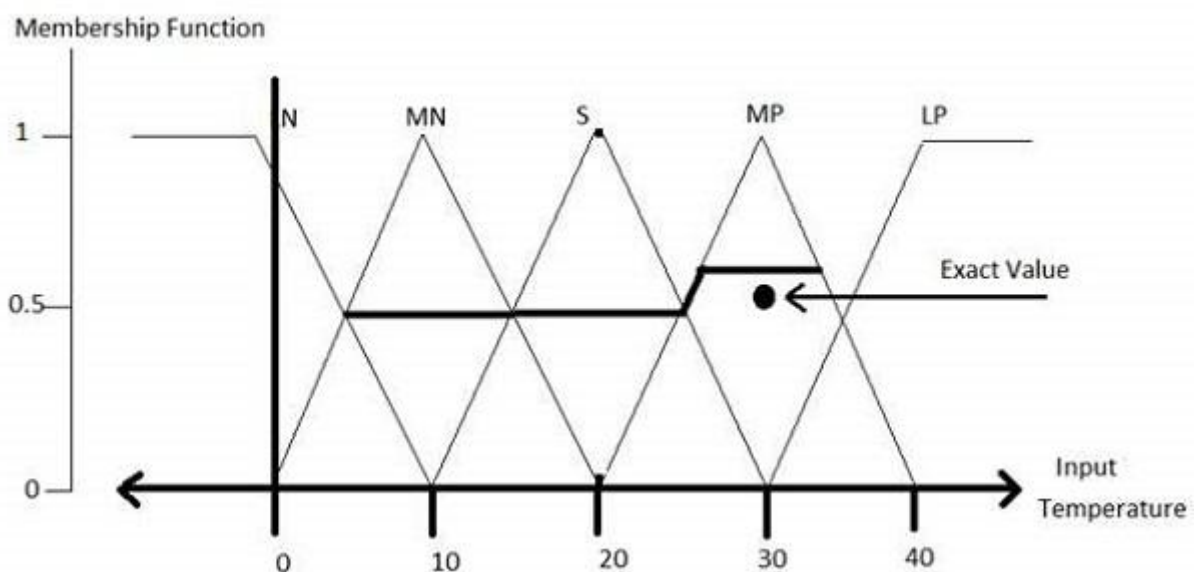
No Señor.	Condición	Acción
1	SI temperatura = (frío o muy frío) Y objetivo = tibio ENTONCES	Calor
2	SI temperatura = (Caliente O Muy_Caliente) Y objetivo = Caliente ENTONCES	Frio
3	SI (temperatura = Cálido) Y (objetivo = Cálido) ENTONCES	Ningún cambio

Paso 4: obtener un valor difuso

Las operaciones de conjuntos difusos realizan una evaluación de las reglas. Las operaciones utilizadas para OR y AND son Max y Min respectivamente. Combine todos los resultados de la evaluación para formar un resultado final. Este resultado es un valor difuso.

Paso 5 - Realice la defuzzificación

La defuzzificación se realiza según la función de pertenencia para la variable de salida.



Áreas de aplicación de Fuzzy Logic

Las áreas de aplicación clave de la lógica difusa son las siguientes:

Sistemas automotrices

- Cajas de cambios automáticas
- Dirección en las cuatro ruedas
- Control del entorno del vehículo

Bienes electrónicos de consumo

- Sistemas de alta fidelidad
- Fotocopiadoras
- Cámaras fijas y de video
- Televisión

Bienes nacionales

- Hornos de microondas
- Refrigeradores
- Tostadoras
- Aspiradoras
- Lavadoras

Control ambiental

- Aires Acondicionados / Secadoras / Calentadores
- Humidificadores

Ventajas de FLS

- Los conceptos matemáticos dentro del razonamiento difuso son muy simples.
- Puede modificar un FLS simplemente agregando o eliminando reglas debido a la flexibilidad de la lógica difusa.
- Los sistemas de lógica difusa pueden tomar información de entrada imprecisa, distorsionada y ruidosa.
- Los FLS son fáciles de construir y comprender.
- La lógica difusa es una solución a problemas complejos en todos los campos de la vida, incluida la medicina, ya que se asemeja al razonamiento humano y la toma de decisiones.

Desventajas de los FLS

- No existe un enfoque sistemático para el diseño de sistemas difusos.
- Son entendibles solo cuando son simples.
- Son adecuados para los problemas que no necesitan alta precisión.

AI - Procesamiento de lenguaje natural

El procesamiento del lenguaje natural (PNL) se refiere al método de inteligencia artificial para comunicarse con sistemas inteligentes que utilizan un lenguaje natural como el inglés.

El procesamiento del lenguaje natural es necesario cuando desea que un sistema inteligente como un robot funcione según sus instrucciones, cuando desea escuchar la decisión de un sistema de expertos clínicos basado en el diálogo, etc.

El campo de la PNL implica hacer que las computadoras realicen tareas útiles con los lenguajes naturales que usan los humanos. La entrada y salida de un sistema PNL puede ser:

- Habla
- Texto escrito

Componentes de PNL

Hay dos componentes de PNL según lo dado:

Comprensión del lenguaje natural (NLU)

La comprensión implica las siguientes tareas:

- Mapear la entrada dada en lenguaje natural en representaciones útiles.
- Analizando diferentes aspectos del lenguaje.

Generación de lenguaje natural (NLG)

Es el proceso de producir frases y oraciones significativas en forma de lenguaje natural a partir de alguna representación interna.

Implica

- **Planificación de texto** : incluye la recuperación del contenido relevante de la base de conocimiento.
- **Planificación de oraciones** : incluye elegir las palabras requeridas, formar frases significativas, establecer el tono de la oración.
- **Realización de texto** : está mapeando el plan de oración en la estructura de la oración.

La NLU es más difícil que la NLG.

Dificultades en NLU

NL tiene una forma y estructura extremadamente rica.

Es muy ambiguo. Puede haber diferentes niveles de ambigüedad:

- **Ambigüedad léxica** : está en un nivel muy primitivo, como el nivel de palabras.
- Por ejemplo, ¿trata la palabra "tablero" como sustantivo o verbo?

- **Nivel de sintaxis ambigüedad** : una oración se puede analizar de diferentes maneras.
- Por ejemplo, "Levantó el escarabajo con gorra roja". ¿Utilizó la gorra para levantar el escarabajo o levantó un escarabajo que tenía gorra roja?
- **Ambigüedad referencial** - Refiriéndose a algo usando pronombres. Por ejemplo, Rima fue a Gauri. Ella dijo: "Estoy cansada". ¿Exactamente quién está cansada?
- Una entrada puede significar diferentes significados.
- Muchas entradas pueden significar lo mismo.

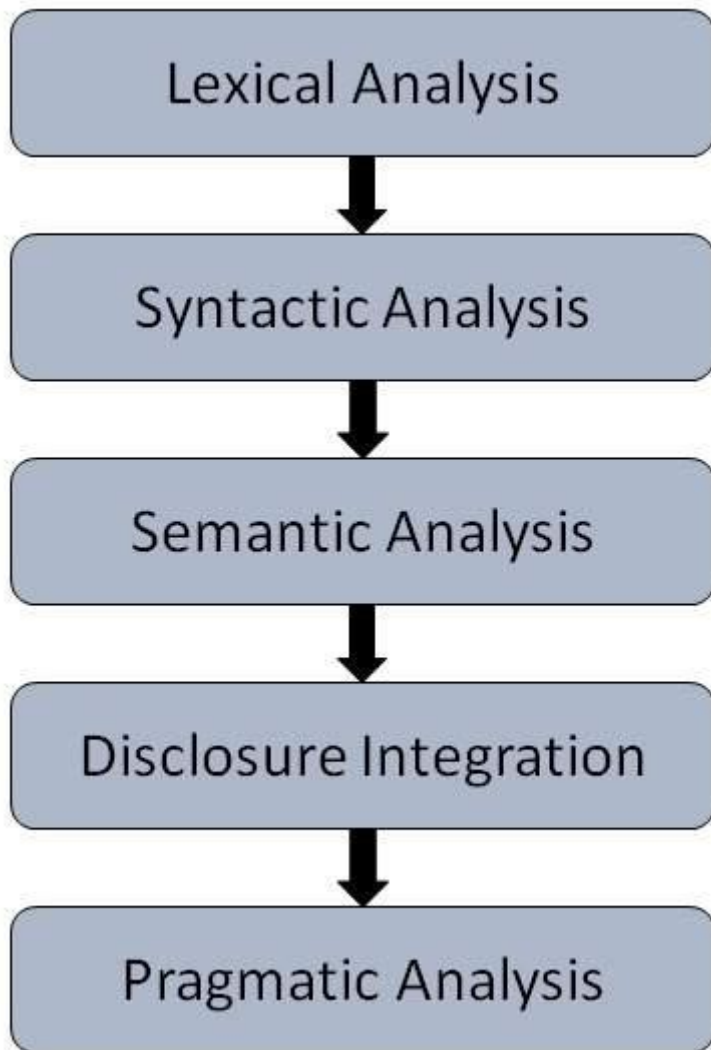
Terminología de PNL

- **Fonología** - Es el estudio de organizar el sonido sistemáticamente.
- **Morfología** : es un estudio de construcción de palabras a partir de unidades primitivas significativas.
- **Morfema** : es la unidad primitiva de significado en un idioma.
- **Sintaxis** : se refiere a organizar palabras para formar una oración. También implica determinar el papel estructural de las palabras en la oración y en las frases.
- **Semántica** : se ocupa del significado de las palabras y de cómo combinar palabras en frases y oraciones significativas.
- **Pragmática** : se trata de usar y comprender oraciones en diferentes situaciones y cómo se ve afectada la interpretación de la oración.
- **Discurso** : trata de cómo la oración que precede inmediatamente puede afectar la interpretación de la siguiente oración.
- **Conocimiento mundial** : incluye el conocimiento general sobre el mundo.

Pasos en PNL

Hay cinco pasos generales:

- **Análisis léxico** : implica identificar y analizar la estructura de las palabras. El léxico de un idioma significa la colección de palabras y frases en un idioma. El análisis léxico está dividiendo todo el fragmento de texto en párrafos, oraciones y palabras.
- **Análisis sintáctico (análisis)** : implica el análisis de las palabras en la oración para la gramática y la organización de las palabras de una manera que muestre la relación entre las palabras. El analizador sintáctico inglés rechaza la frase como "La escuela va al niño".



- **Análisis semántico** : extrae el significado exacto o el significado del diccionario del texto. Se verifica que el texto sea significativo. Se realiza mediante el mapeo de estructuras sintácticas y objetos en el dominio de la tarea. El analizador semántico no tiene en cuenta frases como "helado caliente".
- **Integración del discurso** : el significado de cualquier oración depende del significado de la oración justo antes de ella. Además, también produce el significado de una oración inmediatamente posterior.
- **Análisis pragmático** : durante esto, lo que se dijo se reinterpreta sobre lo que realmente significaba. Implica derivar aquellos aspectos del lenguaje que requieren conocimiento del mundo real.

Aspectos de implementación del análisis sintáctico

Existen varios algoritmos que los investigadores han desarrollado para el análisis sintáctico, pero consideramos solo los siguientes métodos simples:

- Gramática sin contexto
- Analizador de arriba hacia abajo

Vamos a verlos en detalle.

Gramática sin contexto

Es la gramática que consiste en reglas con un solo símbolo en el lado izquierdo de las reglas de reescritura. Creemos gramática para analizar una oración:

"El pájaro picotea los granos"

Artículos (DET) - a | una | el

Sustantivos - pájaro | pájaros | grano | granos

Frase sustantiva (NP) - Artículo + Sustantivo | Artículo + Adjetivo + Sustantivo

= DET N | DET ADJ N

Verbos - picotazos | picoteando | picoteado

Frase verbal (VP) - NP V | V NP

Adjetivos (ADJ) - hermoso | pequeño | piar

El árbol de análisis descompone la oración en partes estructuradas para que la computadora pueda entenderla y procesarla fácilmente. Para que el algoritmo de análisis construya este árbol de análisis, es necesario construir un conjunto de reglas de reescritura que describan qué estructuras de árbol son legales.

Estas reglas dicen que cierto símbolo puede expandirse en el árbol mediante una secuencia de otros símbolos. De acuerdo con la regla lógica de primer orden, si hay dos cadenas Frase sustantiva (NP) y Frase verbal (VP), entonces la cadena combinada por NP seguida de VP es una oración. Las reglas de reescritura de la oración son las siguientes:

S → NP VP

NP → DET N | DET ADJ N

VP → V NP

Lexocon -

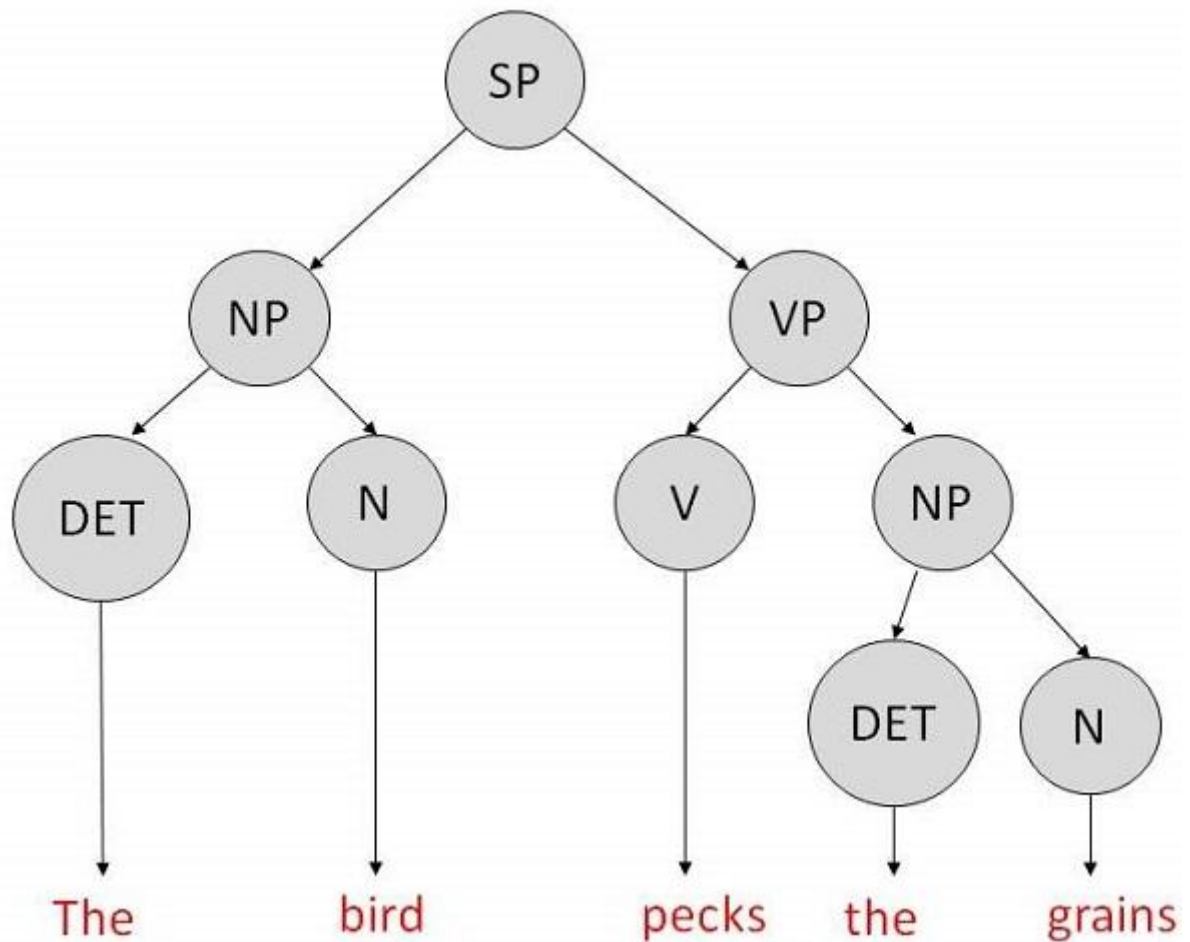
DET → a | el

ADJ → hermosa | encaramado

N → pájaro | pájaros | grano | granos

V → picotear | picotazos | picoteo

El árbol de análisis se puede crear como se muestra:



Ahora considere las reglas de reescritura anteriores. Dado que V puede ser reemplazado por ambos, "picotear" o "picotear", se pueden permitir erróneamente oraciones como "El pájaro picotea los granos". es decir, el error de acuerdo sujeto-verbo se aprueba como correcto.

Mérito : el estilo de gramática más simple, por lo tanto, ampliamente utilizado.

Deméritos -

- No son muy precisos. Por ejemplo, "Los granos picotean al pájaro", es un sintácticamente correcto según el analizador, pero incluso si no tiene sentido, el analizador lo toma como una oración correcta.
- Para resaltar la alta precisión, se deben preparar múltiples conjuntos de gramática. Puede requerir un conjunto completamente diferente de reglas para analizar variaciones singulares y plurales, oraciones pasivas, etc., lo que puede conducir a la creación de un gran conjunto de reglas que son inmanejables.

Analizador de arriba hacia abajo

Aquí, el analizador comienza con el símbolo S e intenta reescribirlo en una secuencia de *símbolos terminales* que coincida con las clases de las palabras en la oración de entrada hasta que consista completamente en símbolos terminales.

Luego se verifican con la oración de entrada para ver si coincide. Si no, el proceso se inicia nuevamente con un conjunto diferente de reglas. Esto se

repite hasta que se encuentre una regla específica que describa la estructura de la oración.

Mérito : es fácil de implementar.

Deméritos -

- Es ineficiente, ya que el proceso de búsqueda debe repetirse si se produce un error.
- Baja velocidad de trabajo.

Inteligencia artificial - Sistemas expertos

Los sistemas expertos (ES) son uno de los dominios de investigación destacados de la IA. Es presentado por los investigadores de la Universidad de Stanford, Departamento de Informática.

¿Qué son los sistemas expertos?

Los sistemas expertos son las aplicaciones informáticas desarrolladas para resolver problemas complejos en un dominio particular, a nivel de inteligencia y experiencia humana extraordinaria.

Características de los sistemas expertos

- Alto rendimiento
- Comprensible
- De confianza
- Altamente sensible

Capacidades de sistemas expertos

Los sistemas expertos son capaces de:

- Asesoramiento
- Instruir y ayudar a los humanos en la toma de decisiones.
- Demostrando
- Derivando una solución
- Diagnosticando
- Explicando
- Interpretación de entrada
- Predecir resultados
- Justificando la conclusión
- Sugerir opciones alternativas a un problema

Son incapaces de ...

- Sustitución de tomadores de decisiones humanos
- Poseer capacidades humanas

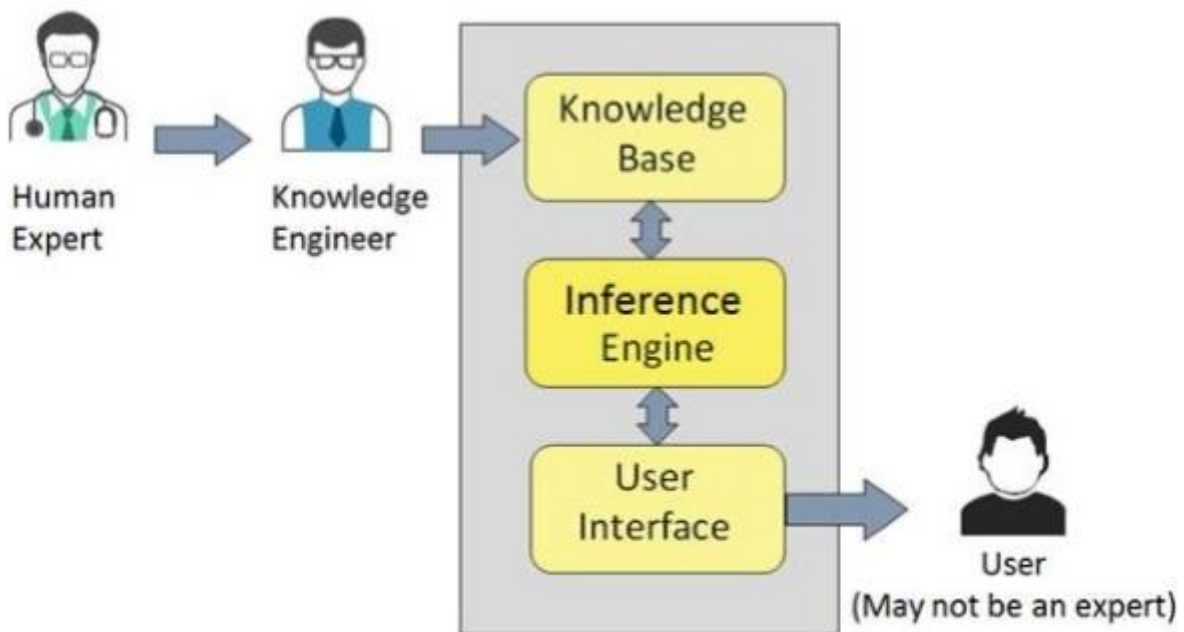
- Producir resultados precisos para una base de conocimiento inadecuada
- Refinando su propio conocimiento

Componentes de sistemas expertos

Los componentes de ES incluyen:

- Base de conocimientos
- Máquina de inferencia
- Interfaz de usuario

Vamos a verlos uno por uno brevemente.



Base de conocimientos

Contiene conocimiento de dominio específico y de alta calidad.

Se requiere conocimiento para exhibir inteligencia. El éxito de cualquier ES depende principalmente de la recopilación de conocimiento altamente preciso y preciso.

¿Qué es el conocimiento?

Los datos son recopilación de hechos. La información se organiza como datos y hechos sobre el dominio de la tarea. **Los datos, la información y la experiencia pasada** combinados se denominan conocimiento.

Componentes de la base de conocimiento

La base de conocimiento de un ES es una reserva de conocimiento tanto factual como heurístico.

- **Conocimiento fáctico** : es la información ampliamente aceptada por los ingenieros de conocimiento y académicos en el dominio de la tarea.
- **Conocimiento heurístico** : se trata de la práctica, el juicio preciso, la capacidad de evaluación y la adivinación.

Representación del conocimiento

Es el método utilizado para organizar y formalizar el conocimiento en la base de conocimiento. Tiene la forma de reglas IF-THEN-ELSE.

Adquisición de conocimientos

El éxito de cualquier sistema experto depende principalmente de la calidad, integridad y precisión de la información almacenada en la base de conocimiento.

La base de conocimiento está formada por lecturas de varios expertos, académicos y los **ingenieros de conocimiento** . El ingeniero de conocimiento es una persona con las cualidades de empatía, aprendizaje rápido y habilidades de análisis de casos.

Adquiere información del experto en la materia al grabarlo, entrevistarle y observarlo en el trabajo, etc. Luego, clasifica y organiza la información de manera significativa, en forma de reglas IF-THEN-ELSE, para ser utilizada por la máquina de inferencia. El ingeniero de conocimiento también supervisa el desarrollo de la ES.

Máquina de inferencia

El uso de procedimientos y reglas eficientes por parte del motor de inferencia es esencial para deducir una solución correcta e impecable.

En el caso de ES basado en el conocimiento, el motor de inferencia adquiere y manipula el conocimiento de la base de conocimiento para llegar a una solución particular.

En caso de ES basado en reglas, se -

- Aplica reglas repetidamente a los hechos, que se obtienen de la aplicación de reglas anteriores.
- Agrega nuevos conocimientos a la base de conocimientos si es necesario.
- Resuelve conflictos de reglas cuando varias reglas son aplicables a un caso particular.

Para recomendar una solución, el motor de inferencia utiliza las siguientes estrategias:

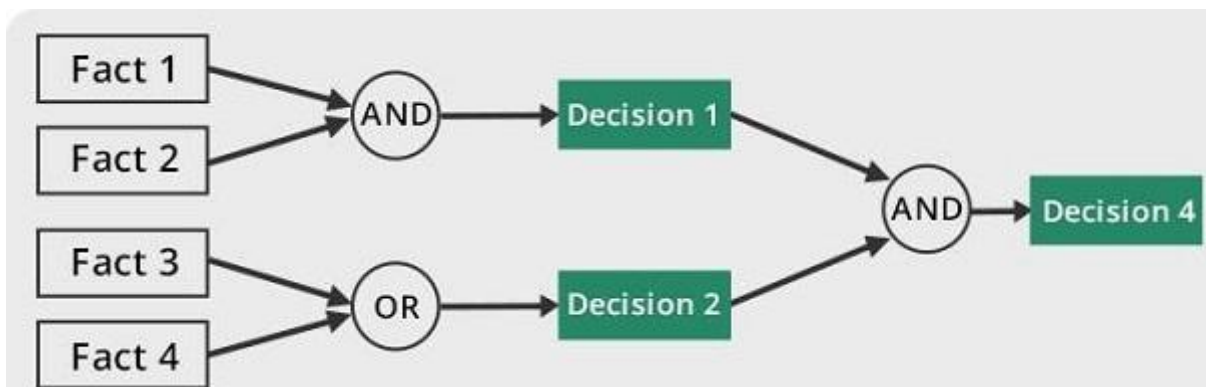
- Encadenamiento Adelante
- Encadenamiento hacia atrás

Encadenamiento Adelante

Es una estrategia de un sistema experto para responder la pregunta, "**¿Qué puede pasar después?**"

Aquí, el motor de inferencia sigue la cadena de condiciones y derivaciones y finalmente deduce el resultado. Considera todos los hechos y reglas, y los ordena antes de llegar a una solución.

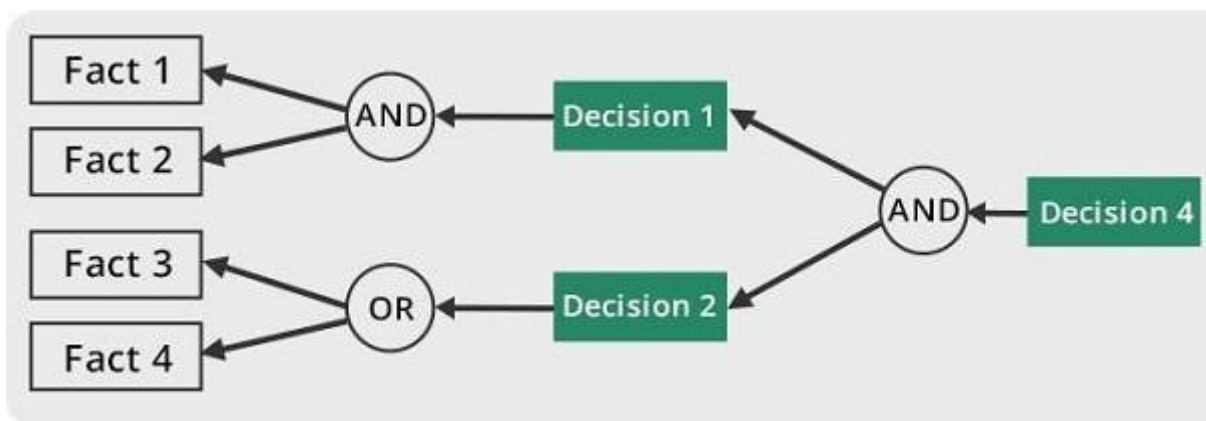
Esta estrategia se sigue para trabajar en la conclusión, resultado o efecto. Por ejemplo, la predicción del estado del mercado de acciones como efecto de los cambios en las tasas de interés.



Encadenamiento hacia atrás

Con esta estrategia, un sistema experto descubre la respuesta a la pregunta, "**¿Por qué sucedió esto?**"

Sobre la base de lo que ya sucedió, el motor de inferencia intenta averiguar qué condiciones podrían haber sucedido en el pasado para este resultado. Esta estrategia se sigue para descubrir la causa o la razón. Por ejemplo, diagnóstico de cáncer de sangre en humanos.



Interfaz de usuario

La interfaz de usuario proporciona interacción entre el usuario del ES y el propio ES. Generalmente es el procesamiento del lenguaje natural para ser utilizado por el usuario que está bien versado en el dominio de la tarea. El

usuario de ES no necesita ser necesariamente un experto en Inteligencia Artificial.

Explica cómo el ES ha llegado a una recomendación particular. La explicación puede aparecer en las siguientes formas:

- Lenguaje natural que se muestra en la pantalla.
- Narraciones verbales en lenguaje natural.
- Listado de números de regla que se muestran en la pantalla.

La interfaz de usuario facilita el seguimiento de la credibilidad de las deducciones.

Requisitos de la interfaz de usuario eficiente de ES

- Debería ayudar a los usuarios a lograr sus objetivos de la manera más corta posible.
- Debe estar diseñado para funcionar para las prácticas laborales existentes o deseadas del usuario.
- Su tecnología debe ser adaptable a los requisitos del usuario; No al revés.
- Debe hacer un uso eficiente de la entrada del usuario.

Limitaciones de sistemas expertos

Ninguna tecnología puede ofrecer una solución fácil y completa. Los sistemas grandes son costosos, requieren un tiempo de desarrollo significativo y recursos informáticos. Los ES tienen sus limitaciones que incluyen:

- Limitaciones de la tecnología
- Difícil adquisición de conocimiento
- Los ES son difíciles de mantener
- Altos costos de desarrollo

Aplicaciones del sistema experto

La siguiente tabla muestra dónde se puede aplicar ES.

Solicitud	Descripción
Dominio de diseño	Diseño de lentes de cámara, diseño de automóviles.
Dominio medico	Sistemas de diagnóstico para deducir la causa de la enfermedad a partir de datos observados, operaciones médicas de conducción en humanos.

Sistemas de monitoreo	Comparar datos continuamente con el sistema observado o con el comportamiento prescrito, como el monitoreo de fugas en tuberías de petróleo largas.
Sistemas de control de procesos	Control de un proceso físico basado en monitoreo.
Dominio del conocimiento	Descubrir fallas en vehículos, computadoras.
Finanzas / Comercio	Detección de posibles fraudes, transacciones sospechosas, negociación en bolsa, programación de aerolíneas, programación de carga.

Tecnología de sistema experta

Hay varios niveles de tecnologías ES disponibles. Las tecnologías de sistemas expertos incluyen:

- **Entorno de desarrollo de sistema experto** : el entorno de desarrollo de ES incluye hardware y herramientas. Ellos son
 - Estaciones de trabajo, minicomputadoras, mainframes.
 - Lenguajes de programación simbólicos de alto nivel tales como **LIS t P** rogramming (LISP) y **PRO** gramation en **LOG** ique (PROLOG).
 - Grandes bases de datos.
- **Herramientas** : reducen el esfuerzo y el costo involucrados en el desarrollo de un sistema experto en gran medida.
 - Potentes editores y herramientas de depuración con múltiples ventanas.
 - Proporcionan prototipos rápidos
 - Tener definiciones incorporadas de modelo, representación del conocimiento y diseño de inferencia.
- **Shells** : un shell no es más que un sistema experto sin base de conocimientos. Un shell proporciona a los desarrolladores la adquisición de conocimiento, el motor de inferencia, la interfaz de usuario y la facilidad de explicación. Por ejemplo, a continuación se dan algunas conchas:
 - Java Expert System Shell (JESS) que proporciona una API Java completamente desarrollada para crear un sistema experto.
 - *Vidwan* , un shell desarrollado en el Centro Nacional de Tecnología de Software, Mumbai en 1993. Permite la codificación del conocimiento en forma de reglas IF-THEN.

Desarrollo de sistemas expertos: pasos generales

El proceso de desarrollo de ES es iterativo. Los pasos para desarrollar el ES incluyen:

Identificar el dominio del problema

- El problema debe ser adecuado para que un sistema experto lo resuelva.
- Encuentre los expertos en el dominio de tareas para el proyecto ES.
- Establecer la rentabilidad del sistema.

Diseña el sistema

- Identificar la tecnología ES
- Conocer y establecer el grado de integración con los otros sistemas y bases de datos.
- Date cuenta de cómo los conceptos pueden representar mejor el conocimiento del dominio.

Desarrollar el prototipo

De Knowledge Base: El ingeniero de conocimiento trabaja para:

- Adquirir conocimiento de dominio del experto.
- Representarlo en forma de reglas If-THEN-ELSE.

Probar y refinar el prototipo

- El ingeniero de conocimiento utiliza casos de muestra para probar el prototipo en busca de deficiencias en el rendimiento.
- Los usuarios finales prueban los prototipos de ES.

Desarrollar y completar el ES

- Probar y garantizar la interacción del ES con todos los elementos de su entorno, incluidos los usuarios finales, las bases de datos y otros sistemas de información.
- Documente bien el proyecto ES.
- Entrenar al usuario para usar ES.

Mantener el sistema

- Mantenga actualizada la base de conocimientos mediante revisiones y actualizaciones periódicas.
- Atiende nuevas interfaces con otros sistemas de información, a medida que estos sistemas evolucionan.

Beneficios de los sistemas expertos

- **Disponibilidad** : están fácilmente disponibles debido a la producción en masa de software.

- **Menos costo de producción:** el costo de producción es razonable. Esto los hace asequibles.
- **Velocidad :** ofrecen una gran velocidad. Reducen la cantidad de trabajo que realiza un individuo.
- **Menos tasa de error:** la tasa de error es baja en comparación con los errores humanos.
- **Reducción del riesgo :** pueden funcionar en un entorno peligroso para los humanos.
- **Respuesta constante :** funcionan de manera constante sin moverse, tensarse o fatigarse.

Inteligencia Artificial - Robótica

La robótica es un dominio de la inteligencia artificial que se ocupa del estudio de la creación de robots inteligentes y eficientes.

¿Qué son los robots?

Los robots son los agentes artificiales que actúan en el entorno del mundo real.

Objetivo

Los robots están destinados a manipular los objetos percibiendo, escogiendo, moviendo, modificando las propiedades físicas del objeto, destruyéndolo, o para tener un efecto que libere a la mano de obra de realizar funciones repetitivas sin aburrirse, distraerse o agotarse.

¿Qué es la robótica?

La robótica es una rama de la IA, que se compone de Ingeniería Eléctrica, Ingeniería Mecánica e Informática para el diseño, construcción y aplicación de robots.

Aspectos de la robótica.

- Los robots tienen **una construcción mecánica** , forma o forma diseñada para realizar una tarea en particular.
- Tienen **componentes eléctricos** que alimentan y controlan la maquinaria.
- Contienen algún nivel de **programa de computadora** que determina qué, cuándo y cómo un robot hace algo.

Diferencia en el sistema de robot y otros programas de IA

Aquí está la diferencia entre los dos:

Programas de IA	Robots
Usualmente operan en mundos estimulados por computadora.	Operan en el mundo físico real
La entrada a un programa de IA está en símbolos y reglas.	Las entradas a los robots son señales analógicas en forma de forma de onda de voz o imágenes.
Necesitan computadoras de uso general para operar.	Necesitan hardware especial con sensores y efectores.

Robot Locomoción

La locomoción es el mecanismo que hace que un robot sea capaz de moverse en su entorno. Hay varios tipos de locomociones:

- Patas
- Con ruedas
- Combinación de locomoción de patas y ruedas
- Deslizamiento / deslizamiento

Locomoción patas

- Este tipo de locomoción consume más energía al demostrar caminar, saltar, trotar, saltar, subir o bajar, etc.
- Se requiere más cantidad de motores para realizar un movimiento. Es adecuado para terrenos irregulares y lisos donde la superficie irregular o demasiado lisa hace que consuma más energía para una locomoción con ruedas. Es poco difícil de implementar debido a problemas de estabilidad.
- Viene con la variedad de una, dos, cuatro y seis patas. Si un robot tiene múltiples piernas, entonces la coordinación de las piernas es necesaria para la locomoción.

El número total de **pasos** posibles (una secuencia periódica de eventos de elevación y liberación para cada uno de los tramos totales) que un robot puede viajar depende del número de sus tramos.

Si un robot tiene k patas, entonces el número de posibles eventos $N = (2k-1)$.

¡En el caso de un robot de dos patas ($k = 2$), el número de eventos posibles es $N = (2k-1)! = (2 * 2-1)! = 3! = 6$.

Por lo tanto, hay seis posibles eventos diferentes:

- Levantando la pierna izquierda
- Liberando la pierna izquierda
- Levantando la pierna derecha

- Liberando la pierna derecha
- Levantando ambas piernas juntas
- Liberando ambas piernas juntas

En el caso de $k = 6$ patas, hay 39916800 posibles eventos. Por lo tanto, la complejidad de los robots es directamente proporcional al número de patas.



Locomoción de ruedas

Requiere menos motores para realizar un movimiento. Es poco fácil de implementar ya que hay menos problemas de estabilidad en caso de que haya más ruedas. Es eficiente en comparación con la locomoción de patas.

- **Rueda estándar** : gira alrededor del eje de la rueda y alrededor del contacto
- **Rueda giratoria**: gira alrededor del eje de la rueda y la junta de dirección desplazada.
- **Ruedas suecas de 45 ° y 90 ° suecas** : Omni-wheel, gira alrededor del punto de contacto, alrededor del eje de la rueda y alrededor de los rodillos.
- **Bola o rueda esférica**: **rueda** omnidireccional, técnicamente difícil de implementar.



Deslizamiento / Deslizamiento Locomoción

En este tipo, los vehículos usan orugas como en un tanque. El robot se dirige moviendo las pistas con diferentes velocidades en la misma dirección o en

dirección opuesta. Ofrece estabilidad debido a la gran área de contacto de la pista y el suelo.



Componentes de un robot

Los robots se construyen con lo siguiente:

- **Fuente de alimentación** : los robots funcionan con baterías, energía solar, hidráulica o neumática.
- **Actuadores** : convierten la energía en movimiento.
- **Motores eléctricos (CA / CC)** : son necesarios para el movimiento de rotación.
- **Músculos neumáticos de aire** : se contraen casi el 40% cuando se aspira aire.
- **Cables musculares** : se contraen un 5% cuando la corriente eléctrica pasa a través de ellos.
- **Motores piezoeléctricos y ultrasónicos** : lo mejor para robots industriales.
- **Sensores** : proporcionan conocimiento de información en tiempo real sobre el entorno de la tarea. Los robots están equipados con sensores de visión para calcular la profundidad en el entorno. Un sensor táctil imita las propiedades mecánicas de los receptores táctiles de las yemas de los dedos humanos.

Visión por computador

Esta es una tecnología de IA con la que los robots pueden ver. La visión por computadora juega un papel vital en los dominios de seguridad, salud, acceso y entretenimiento.

La visión por computadora extrae, analiza y comprende automáticamente información útil de una sola imagen o de una serie de imágenes. Este proceso implica el desarrollo de algoritmos para lograr la comprensión visual automática.

Hardware del sistema de visión por computadora

Esto implica:

- Fuente de alimentación
- Dispositivo de adquisición de imagen como cámara

- Un procesador
- Un software
- Un dispositivo de visualización para monitorear el sistema
- Accesorios como soportes de cámara, cables y conectores.

Tareas de visión por computadora

- **OCR** : en el dominio de las computadoras, Optical Character Reader, un software para convertir documentos escaneados en texto editable, que acompaña a un escáner.
- **Detección de rostros** : muchas cámaras de última generación vienen con esta característica, que permite leer el rostro y tomar la fotografía de esa expresión perfecta. Se utiliza para permitir que un usuario acceda al software en la coincidencia correcta.
- **Reconocimiento de objetos** : se instalan en supermercados, cámaras, automóviles de alta gama como BMW, GM y Volvo.
- **Posición estimada** : es la posición estimada de un objeto con respecto a la cámara como la posición del tumor en el cuerpo humano.

Dominios de aplicación de visión artificial

- Agricultura
- Vehículos autónomos
- Biometría
- Reconocimiento de caracteres
- Forense, seguridad y vigilancia
- Inspección de calidad industrial
- Reconocimiento facial
- Análisis de gestos
- Geociencia
- Imágenes medicas
- Monitoreo de la contaminación
- Control de procesos
- Sensores remotos
- Robótica
- Transporte

Aplicaciones de la robótica.

La robótica ha sido fundamental en varios dominios como:

- **Industrias** : los robots se utilizan para manipular material, cortar, soldar, revestir con color, perforar, pulir, etc.
- **Militar** : los robots autónomos pueden alcanzar zonas inaccesibles y peligrosas durante la guerra. Un robot llamado *Daksh* , desarrollado por la Organización de Investigación y Desarrollo de Defensa (DRDO), está en función de destruir objetos que amenazan la vida de forma segura.

- **Medicina** : los robots son capaces de realizar cientos de pruebas clínicas simultáneamente, rehabilitar a personas con discapacidad permanente y realizar cirugías complejas, como tumores cerebrales.
- **Exploración** : los robots escaladores de rocas utilizados para la exploración espacial, los drones submarinos utilizados para la exploración oceánica son solo algunos.
- **Entretención** : los ingenieros de Disney han creado cientos de robots para hacer películas.

Inteligencia artificial - Redes neuronales

Otra área de investigación en IA, las redes neuronales, está inspirada en la red neuronal natural del sistema nervioso humano.

¿Qué son las redes neuronales artificiales (ANN)?

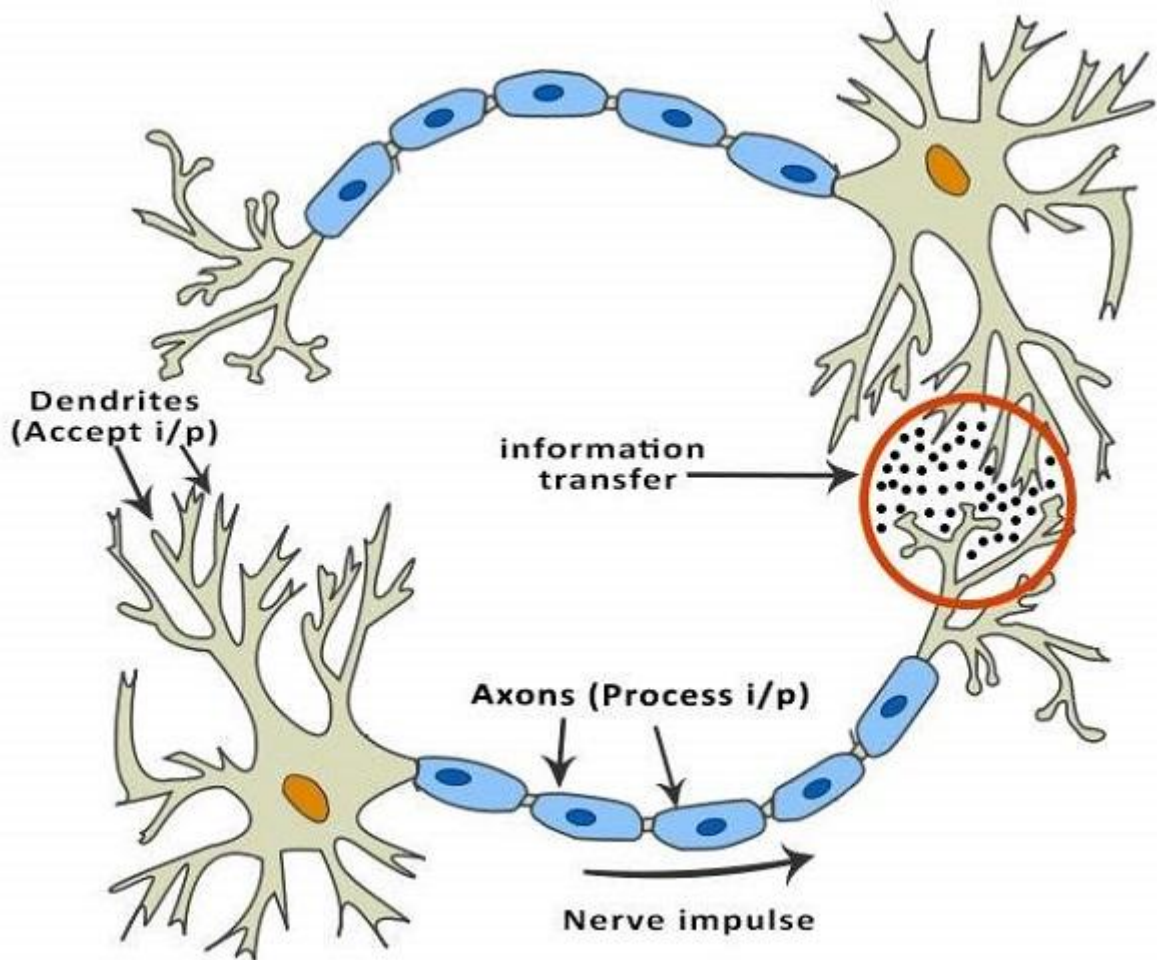
El inventor del primer neuroordenador, el Dr. Robert Hecht-Nielsen, define una red neuronal como:

"... un sistema informático compuesto por una serie de elementos de procesamiento simples y altamente interconectados, que procesan la información por su respuesta de estado dinámico a las entradas externas".

Estructura básica de las ANN

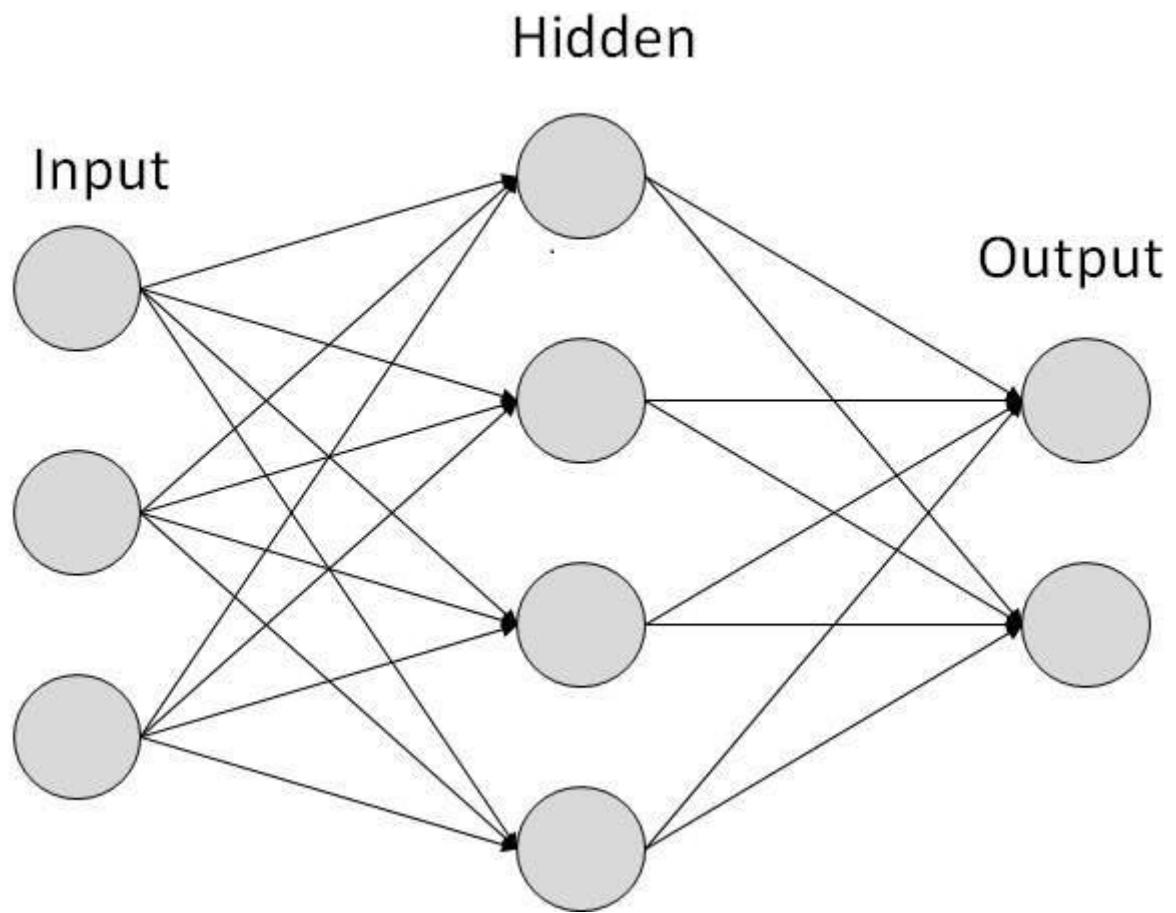
La idea de ANN se basa en la creencia de que el funcionamiento del cerebro humano al hacer las conexiones correctas puede imitarse utilizando silicio y cables como **neuronas** y **dendritas** vivas .

El cerebro humano está compuesto por 86 mil millones de células nerviosas llamadas **neuronas**. Están conectados a otras mil células por **Axons**. Los estímulos del entorno externo o los aportes de los órganos sensoriales son aceptados por las dendritas. Estas entradas crean impulsos eléctricos, que viajan rápidamente a través de la red neuronal. Una neurona puede enviar el mensaje a otra neurona para manejar el problema o no lo reenvía.



Los ANN están compuestos por múltiples **nodos**, que imitan las **neuronas** biológicas del cerebro humano. Las neuronas están conectadas por enlaces e interactúan entre sí. Los nodos pueden tomar datos de entrada y realizar operaciones simples en los datos. El resultado de estas operaciones se pasa a otras neuronas. La salida en cada nodo se llama su **activación** o **valor de nodo**.

Cada enlace está asociado con el **peso**. Las ANN son capaces de aprender, lo que ocurre al alterar los valores de peso. La siguiente ilustración muestra un simple ANN -

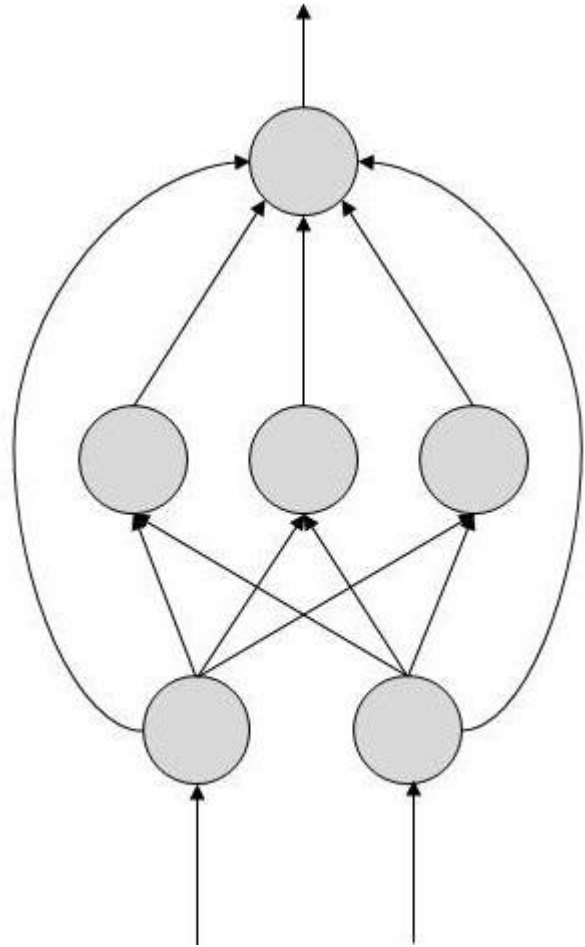
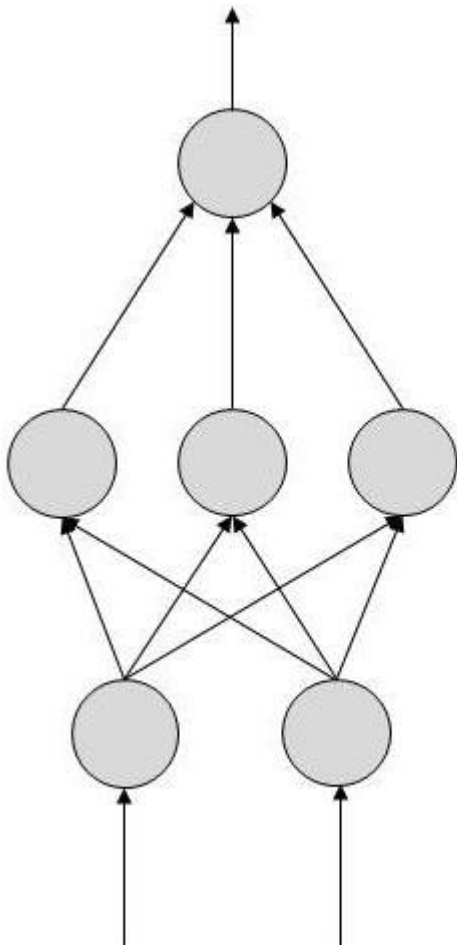


Tipos de redes neuronales artificiales

Hay dos topologías de redes neuronales artificiales: **FeedForward** y **Feedback**.

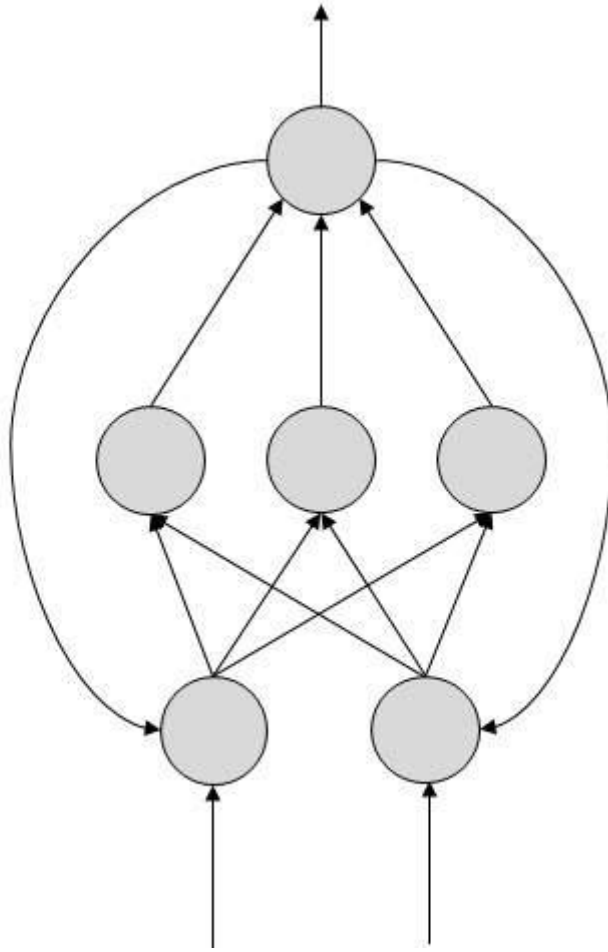
FeedForward ANN

En este ANN, el flujo de información es unidireccional. Una unidad envía información a otra unidad de la que no recibe ninguna información. No hay bucles de retroalimentación. Se utilizan en la generación / reconocimiento / clasificación de patrones. Tienen entradas y salidas fijas.



FeedBack ANN

Aquí, se permiten bucles de retroalimentación. Se usan en memorias direccionables de contenido.



Trabajo de ANNs

En los diagramas de topología mostrados, cada flecha representa una conexión entre dos neuronas e indica la vía para el flujo de información. Cada conexión tiene un peso, un número entero que controla la señal entre las dos neuronas.

Si la red genera una salida "buena o deseada", no hay necesidad de ajustar los pesos. Sin embargo, si la red genera una salida "pobre o no deseada" o un error, entonces el sistema altera los pesos para mejorar los resultados posteriores.

Aprendizaje automático en ANN

Los ANN son capaces de aprender y necesitan ser entrenados. Hay varias estrategias de aprendizaje:

- **Aprendizaje supervisado** : involucra a un maestro que es académico más que el ANN. Por ejemplo, el maestro alimenta algunos datos de ejemplo sobre los cuales el maestro ya conoce las respuestas.

Por ejemplo, el reconocimiento de patrones. Al ANN se le ocurren conjeturas mientras reconoce. Luego, el profesor proporciona a la ANN las

respuestas. Luego, la red compara las conjeturas con las respuestas "correctas" del profesor y realiza ajustes de acuerdo con los errores.

- **Aprendizaje no supervisado** : se requiere cuando no hay un conjunto de datos de ejemplo con respuestas conocidas. Por ejemplo, buscando un patrón oculto. En este caso, la agrupación, es decir, la división de un conjunto de elementos en grupos de acuerdo con algún patrón desconocido, se lleva a cabo en función de los conjuntos de datos existentes.
- **Aprendizaje de refuerzo** : esta estrategia se basa en la observación. La ANN toma una decisión observando su entorno. Si la observación es negativa, la red ajusta sus pesos para poder tomar una decisión requerida diferente la próxima vez.

Algoritmo de propagación hacia atrás

Es el algoritmo de entrenamiento o aprendizaje. Aprende con el ejemplo. Si envía al algoritmo el ejemplo de lo que desea que haga la red, cambia los pesos de la red para que pueda producir la salida deseada para una entrada particular al finalizar el entrenamiento.

Las redes de propagación hacia atrás son ideales para tareas simples de reconocimiento de patrones y mapeo.

Redes Bayesianas (BN)

Estas son las estructuras gráficas utilizadas para representar la relación probabilística entre un conjunto de variables aleatorias. Las redes bayesianas también se denominan **redes de creencias** o **redes bayesianas**. BN razona sobre el dominio incierto.

En estas redes, cada nodo representa una variable aleatoria con proposiciones específicas. Por ejemplo, en un dominio de diagnóstico médico, el nodo Cáncer representa la proposición de que un paciente tiene cáncer.

Los bordes que conectan los nodos representan dependencias probabilísticas entre esas variables aleatorias. Si de dos nodos, uno está afectando al otro, entonces deben estar conectados directamente en las direcciones del efecto. La fuerza de la relación entre variables se cuantifica por la probabilidad asociada con cada nodo.

Hay una única restricción en los arcos en un BN que no puede volver a un nodo simplemente siguiendo los arcos dirigidos. Por lo tanto, los BN se denominan Gráficos Acíclicos Dirigidos (DAG).

Los BN son capaces de manejar variables de valores múltiples simultáneamente. Las variables BN se componen de dos dimensiones:

- Rango de preposiciones
- Probabilidad asignada a cada una de las preposiciones.

Considere un conjunto finito $X = \{X_1, X_2, \dots, X_n\}$ de variables aleatorias discretas, donde cada variable X_i puede tomar valores de un conjunto finito, denotado por $Val(X_i)$. Si hay un enlace dirigido de la variable X_i a la

variable, X_j , entonces la variable X_i será un padre de la variable X_j mostrando dependencias directas entre las variables.

La estructura de BN es ideal para combinar conocimiento previo y datos observados. BN puede usarse para aprender las relaciones causales y comprender varios dominios problemáticos y para predecir eventos futuros, incluso en el caso de que falten datos.

Construyendo una Red Bayesiana

Un ingeniero de conocimiento puede construir una red bayesiana. Hay varios pasos que el ingeniero de conocimiento debe tomar mientras lo construye.

Problema de ejemplo : *cáncer de pulmón*. Un paciente ha estado sufriendo de disnea. Visita al médico, sospechando que tiene cáncer de pulmón. El médico sabe que, salvo el cáncer de pulmón, existen otras enfermedades posibles que el paciente podría tener, como tuberculosis y bronquitis.

Recopilar información relevante del problema

- ¿El paciente es fumador? Si es así, entonces hay altas posibilidades de cáncer y bronquitis.
- ¿Está el paciente expuesto a la contaminación del aire? En caso afirmativo, ¿qué tipo de contaminación del aire?
- Tomar una radiografía positiva con rayos X indicaría TB o cáncer de pulmón.

Identificar variables interesantes

El ingeniero de conocimiento intenta responder las preguntas:

- ¿Qué nodos representar?
- ¿Qué valores pueden tomar? ¿En qué estado pueden estar?

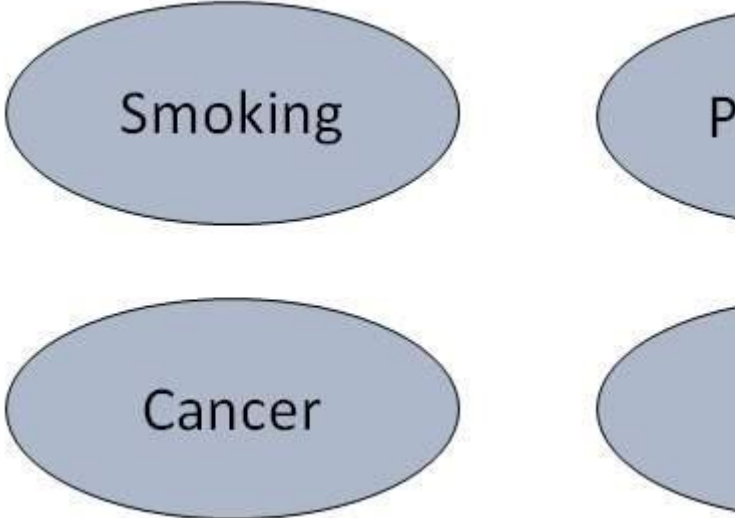
Por ahora consideremos nodos, con solo valores discretos. La variable debe tomar exactamente uno de estos valores a la vez.

Los tipos comunes de nodos discretos son :

- **Nodos booleanos** : representan proposiciones, tomando valores binarios VERDADERO (T) y FALSO (F).
- **Valores ordenados** : una *contaminación* de nodo puede representar y tomar valores de {bajo, medio, alto} que describe el grado de exposición de un paciente a la contaminación.
- **Valores integrales** : un nodo llamado *Edad* podría representar la edad del paciente con posibles valores de 1 a 120. Incluso en esta etapa temprana, se están tomando decisiones de modelado.

Posibles nodos y valores para el ejemplo de cáncer de pulmón:

Nombre del nodo	Tipo	Valor	Creación de nodos

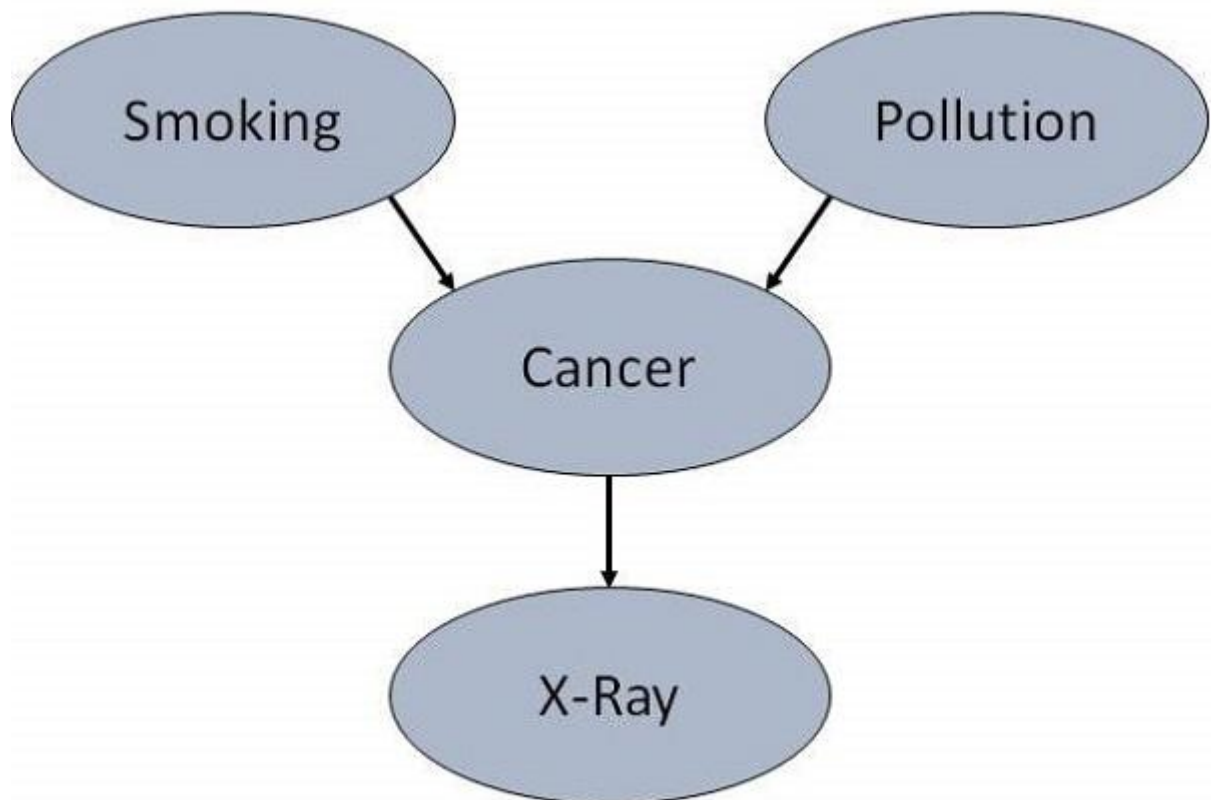
Contaminación	Binario	{BAJO, ALTO, MEDIO}	
Fumador	Booleano	{VERDADERO, FÁCIL}	
Cáncer de pulmón	Booleano	{VERDADERO, FÁCIL}	
Radiografía	Binario	{Positivo negativo}	

Crear arcos entre nodos

La topología de la red debe capturar las relaciones cualitativas entre las variables.

Por ejemplo, ¿qué causa que un paciente tenga cáncer de pulmón? - Contaminación y tabaquismo. Luego agregue arcos desde el nodo *Contaminación* y el nodo *Fumador* al nodo *Cáncer de pulmón*.

Del mismo modo, si el paciente tiene cáncer de pulmón, el resultado de la radiografía será positivo. Luego agregue arcos desde el nodo *Lung-Cancer* al nodo *X-Ray*.



Especificar topología

Convencionalmente, los BN se disponen de modo que los arcos apunten de arriba a abajo. Los padres (X) proporcionan el conjunto de nodos principales de un nodo X.

El *cáncer de pulmón* nodo tiene dos padres (razones o causas): *Contaminación* y *fuma*, mientras que el nodo *Fumador* es un **antepasado** de nodo *X-Ray*. Del mismo modo, la *radiografía* es un niño (consecuencia o efectos) del *cáncer de pulmón* de nodo y **sucesor** de los nodos *fumador* y *contaminación*.

Probabilidades condicionales

Ahora cuantifique las relaciones entre los nodos conectados: esto se realiza especificando una distribución de probabilidad condicional para cada nodo. Como aquí solo se consideran variables discretas, esto toma la forma de una **Tabla de probabilidad condicional (CPT)**.

Primero, para cada nodo necesitamos observar todas las combinaciones posibles de valores de esos nodos principales. Cada combinación de este tipo se denomina **instanciación** del conjunto primario. Para cada instanciación distinta de los valores de los nodos principales, debemos especificar la probabilidad que tomará el hijo.

Por ejemplo, los padres del nodo de *cáncer de pulmón* son la *contaminación* y el *tabaquismo*. Toman los valores posibles = {(H, T), (H, F), (L, T), (L, F)}. El CPT especifica la probabilidad de cáncer para cada uno de estos casos como <0.05, 0.02, 0.03, 0.001> respectivamente.

Cada nodo tendrá una probabilidad condicional asociada de la siguiente manera:

Smoking	Pollution
$P(S = T)$	$P(P = L)$
0.30	0.90

Lung-Cancer		
P	S	P (C = T P, S)
H	T	0.05
H	F	0.02
L	T	0.03
L	F	0.001

X-Ray	
C	X = (Pos C)
T	0.90
F	0.20

Aplicaciones de redes neuronales

Pueden realizar tareas que son fáciles para un humano pero difíciles para una máquina:

- **Aeroespacial** - Aviones de piloto automático, detección de fallas de aeronaves.
- **Automotriz** : sistemas de guía para automóviles.
- **Militar** : orientación y dirección de armas, seguimiento de objetivos, discriminación de objetos, reconocimiento facial, identificación de señal / imagen.
- **Electrónica** : predicción de secuencia de código, diseño de chip IC, análisis de falla de chip, visión artificial, síntesis de voz.
- **Financiero** : evaluación de bienes inmuebles, asesor de préstamos, evaluación de hipotecas, calificación de bonos corporativos, programa de negociación de cartera, análisis financiero corporativo, predicción del valor de la moneda, lectores de documentos, evaluadores de solicitudes de crédito.
- **Industrial** : control de procesos de fabricación, diseño y análisis de productos, sistemas de inspección de calidad, análisis de calidad de soldadura, predicción de calidad de papel, análisis de diseño de productos químicos, modelado dinámico

de sistemas de procesos químicos, análisis de mantenimiento de máquinas, licitación, planificación y gestión de proyectos.

- **Médico** : análisis de células cancerosas, análisis EEG y ECG, diseño protésico, optimizador de tiempo de trasplante.
- **Voz** : reconocimiento de voz, clasificación de voz, conversión de texto a voz.
- **Telecomunicaciones** : compresión de imágenes y datos, servicios de información automatizados, traducción de idiomas hablados en tiempo real.
- **Transporte** : diagnóstico del sistema de frenos de camiones, programación de vehículos, sistemas de enrutamiento.
- **Software** - Reconocimiento de patrones en reconocimiento facial, reconocimiento óptico de caracteres, etc.
- **Predicción de series de tiempo** : los ANN se utilizan para hacer predicciones sobre stocks y calamidades naturales.
- **Procesamiento de señal** : las redes neuronales se pueden entrenar para procesar una señal de audio y filtrarla adecuadamente en los audífonos.
- **Control** : los ANN se utilizan a menudo para tomar decisiones de dirección de vehículos físicos.
- **Detección de anomalías** : como los ANN son expertos en reconocer patrones, también pueden ser entrenados para generar una salida cuando ocurre algo inusual que no se ajusta al patrón.

Inteligencia artificial - Problemas

La IA se está desarrollando con una velocidad tan increíble, a veces parece mágica. Existe una opinión entre los investigadores y desarrolladores de que la IA podría crecer tan inmensamente fuerte que sería difícil de controlar para los humanos.

Los humanos desarrollaron sistemas de IA al introducirles toda la inteligencia posible que pudieran, por lo que los humanos mismos ahora parecen amenazados.

Amenaza a la privacidad

Un programa de inteligencia artificial que reconoce el habla y comprende el lenguaje natural es teóricamente capaz de comprender cada conversación por correo electrónico y teléfono.

Amenaza a la dignidad humana

Los sistemas de IA ya han comenzado a reemplazar a los seres humanos en pocas industrias. No debe reemplazar a las personas en los sectores donde ocupan puestos dignos que pertenecen a la ética, como enfermería, cirujano, juez, oficial de policía, etc.

Amenaza a la seguridad

Los sistemas de IA que se mejoran a sí mismos pueden llegar a ser tan poderosos que los humanos que podrían ser muy difíciles de detener para lograr sus objetivos, lo que puede tener consecuencias no deseadas.

Inteligencia Artificial - Terminología

Aquí está la lista de términos de uso frecuente en el dominio de AI:

No Señor	Término y significado
1	Agente Los agentes son sistemas o programas de software capaces de ser autónomos, con un propósito y un razonamiento dirigido hacia uno o más objetivos. También se les llama asistentes, corredores, bots, droides, agentes inteligentes y agentes de software.
2	Robot autónomo Robot libre de control o influencia externa y capaz de controlarse de forma independiente.
3	Encadenamiento hacia atrás Estrategia de trabajar hacia atrás por Razón / Causa de un problema.
4 4	Pizarra Es la memoria dentro de la computadora, que se utiliza para la comunicación entre los sistemas expertos que cooperan.
5 5	Ambiente Es la parte del mundo real o computacional habitado por el agente.
6 6	Encadenamiento Adelante Estrategia de trabajar hacia adelante para la conclusión / solución de un problema.
7 7	Heurística Es el conocimiento basado en prueba y error, evaluaciones y experimentación.
8	Ingeniería del conocimiento Adquirir conocimiento de expertos humanos y otros recursos.

9 9	Percepciones Es el formato en el que el agente obtiene información sobre el entorno.
10	Poda Anular consideraciones innecesarias e irrelevantes en los sistemas de IA.
11	Regla Es un formato de representación de la base de conocimiento en Expert System. Está en la forma de IF-THEN-ELSE.
12	Cáscara Un shell es un software que ayuda a diseñar el motor de inferencia, la base de conocimientos y la interfaz de usuario de un sistema experto.
13	Tarea Es el objetivo que el agente intenta lograr.
14	Prueba de Turing Una prueba desarrollada por Allan Turing para probar la inteligencia de una máquina en comparación con la inteligencia humana.