

1.1. Definición de Inteligencia Artificial

¿QUE ES LA INTELIGENCIA ARTIFICIAL?

La inteligencia artificial es nieta de la ciencia y tecnología de la computación e hija de la vida artificial. Se encarga del estudio de la tecnología y la ciencia del diseño de máquinas o agentes que realizan tareas que normalmente se asocian con la inteligencia humana o de los animales con sistemas nerviosos con algo de "inteligencia".

Se la puede definir como:

- una ciencia de lo artificial y
- como un conjunto de tecnologías computacionales que se interesan en cómo se manifiesta
 - ❖ la adaptación al ambiente,
 - ❖ el procesamiento del lenguaje natural,
 - ❖ la representación del conocimiento,
 - ❖ el razonamiento,
 - ❖ la resolución de problemas y la toma de decisiones,
 - ❖ el autoaprendizaje,
 - ❖ la percepción por sentidos que en biología son fisiológicos, pero que pueden ser otros,
 - ❖ la autorreflexión.

La IA es el estudio de técnicas de resolución de problemas de complejidad exponencial mediante el uso de conocimiento sobre el campo de aplicación del problema.

Es una de las áreas de las ciencias computacionales encargadas de la creación de hardware y software que tenga comportamientos inteligentes.

La inteligencia artificial es la parte de la informática que concierne al diseño de sistemas inteligentes para el computador, esto es, sistemas que presentan características asociadas con el comportamiento humano inteligente.

La I.A. es la parte de la informática concerniente al diseño de sistemas de computación inteligente, es decir, de sistemas que exhiban o simulen las características que son comúnmente asociadas con la inteligencia en la conducta humana.

DEFINICIONES DE VARIOS AUTORES

“La interesante tarea de lograr que las computadoras piensen... máquinas con mente, en su amplio sentido literal.” **(Haugeland, 1985).**

“[La automatización de actividades que vinculamos con procesos de pensamiento humano, actividades tales como toma de decisiones, resolución de problemas, aprendizaje...” **(Bellman, 1978).**

“El arte de crear máquinas con capacidad de realizar funciones que realizadas por personas requieren de inteligencia.” **(Kurzweil, 1990).**

“El estudio de cómo lograr que las computadoras realicen tareas que, por el momento, los humanos hacen mejor.” **(Rich y Knight, 1991).**

“El estudio de las facultades mentales mediante el uso de modelos computacionales.” **(Charniak y McDermott, 1985).**

“El estudio de los cálculos que permiten percibir, razonar y actuar.” **(Winston, 1992).**

“Un campo de estudio que se enfoca a la explicación y emulación de la conducta inteligente en función de procesos computacionales.” **(Schalkoff, 1990).**

“La rama de la ciencia de la computación que se ocupa de la automatización de la conducta inteligente.” **(Luger y Stubblefield, 1993).**

OTRAS DEFINICIONES DE I.A.

- ✚ .IA es el estudio de cómo lograr que las computadoras hagan cosas que, de momento, las personas hacen mejor (Nilsson).
- ✚ Son las técnicas de diseño de sistemas informáticos que presentan las características de la inteligencia humana.
- ✚ .Estudio de las ideas que permiten al computador realizar acciones que hacen al ser humano inteligente (Winston).
- ✚ .La IA es la ciencia de hacer máquinas que hacen cosas, que realizadas por el hombre requieren el uso de inteligencia (Minsky).
- ✚ IA es la parte de la ciencia de la computación que investiga procesos simbólicos, razonamientos no algorítmicos, y representación simbólica del conocimiento (Buchanan).

- ✚ La IA es todo lo que no se ha hecho hasta ahora (Tesler).
- ✚ La IA es todo aquello que las computadoras aún no saben hacer (Hofstadter).
- ✚ La IA tiene mucho de A y poco de I. (Rich).

La IA (Inteligencia Artificial) es una de las disciplinas más nuevas. Formalmente se inicia en 1956 cuando se acuñó este término, sin embargo el estudio de la inteligencia contemplada como el razonamiento humano viene siendo estudiado por los filósofos hace más de 2 milenios.

La inteligencia artificial es la ciencia que enfoca su estudio a lograr la comprensión de entidades inteligentes. Es evidente que las computadoras que posean una inteligencia a nivel humano (o superior) tendrán repercusiones muy importantes en nuestra vida diaria.

1.1.1. Diferentes enfoques de Inteligencia Artificial

Se considera generalmente que el origen remoto de la "Inteligencia Artificial" se remonta a los intentos por crear autómatas, que simulasen la forma y las habilidades de los seres humanos.

Aunque el nacimiento de la Inteligencia Artificial se sitúa en una reunión celebrada en el Dartmouth College (Hanover, EEUU) en 1956, en la que se planteó la posibilidad de construir máquinas inteligentes. Llama la atención que en esa fecha la informática apenas se había desarrollado, y ya se planteaba la posibilidad de diseñar mecanismos inteligentes.

En aquella reunión se encontraban entre otros, Claude Shannon, padre de la Teoría de la Información; Marvin Minsky, que más tarde demostraría las limitaciones de ciertos modelos de redes neuronales; Herbert Simon, premio Nobel de Economía quien además desarrolló el primer programa de Inteligencia Artificial y un largo etcétera de investigadores.

Fue en los años 60 cuando Alan Newell y Herbert Simon, que trabajando la demostración de teoremas y el ajedrez por computador logran crear un programa llamado GPS (General Problem Solver: solucionador general de problemas).

Este era un sistema en el que el usuario definía un entorno en función de una serie de objetos y los operadores que se podían aplicar sobre ellos. Este programa era capaz de trabajar con las torres de Hanoi, así como con criptoaritmética y otros problemas similares, operando, claro está, con microcosmos formalizados que representaban los parámetros dentro de los cuales se podían resolver problemas.

Lo que no podía hacer el GPS era resolver problemas ni del mundo real, ni médicos ni tomar decisiones importantes.

Haciendo una reseña de la evolución de la inteligencia artificial se divide en etapas las cuales son:

EVOLUCIÓN HISTÓRICA DE LA INTELIGENCIA ARTIFICIAL.

PRIMERA ETAPA: LA REUNIÓN DE DARTMOUTH.

Verano de 1956. Diez prestigiosos científicos se reúnen en el Dartmouth College (USA.) para discutir las maneras de simular el comportamiento humano mediante la simulación con computador.

Nace la Inteligencia Artificial.

- ✓ “Demostrador” de teoremas lógicos
- ✓ Jugador de damas con capacidad de aprendizaje
- ✓ “Demostrador” de teoremas de geometría plana
- ✓ Reconocedor caracteres escritos a mano
- ✓ GPS. Solucionador de problemas
- ✓ Modelos psicológicos del comportamiento humano
- ✓ Interactivos en lenguaje natural

Es en los años 50 cuando se logra realizar un sistema que tuvo cierto éxito, se llamó el Perceptrón de Rossenblatt. Éste era un sistema visual de reconocimiento de patrones en el cual se aunaron esfuerzos para que se pudieran resolver una gama amplia de problemas, pero estas energías se diluyeron enseguida.

INVESTIGACIONES 1ª ETAPA (56/70):

- ✓ Traducción automática.
- ✓ Cálculo simbólico.
- ✓ Juegos.
- ✓ Proceso de imágenes y reconocimiento de modelos.
- ✓ Resolución de Problemas y demostración de teoremas.
- ✓ Desarrollo de teorías psicológicas y modelos de comportamiento.

Fue en los años 60 cuando Alan Newell y Herbert Simon, que trabajando la demostración de teoremas y el ajedrez por computador logran crear un programa llamado GPS (General Problem Solver: o solucionador general de problemas).

Éste era un sistema en el que el usuario definía un entorno en función de una serie de objetos y los operadores que se podían aplicar sobre ellos.

Este programa era capaz de trabajar con las torres de Hanoi, así como con criptoaritmética y otros problemas similares, operando, claro está, con microcosmos formalizados que representaban los parámetros dentro de los cuales se podían resolver problemas.

Lo que no podía hacer el GPS era resolver problemas ni del mundo real, ni médicos ni tomar decisiones importantes.

El GPS manejaba reglas heurísticas (aprender a partir de sus propios descubrimientos) que la conducían hasta el destino deseado mediante el método del ensayo y el error.

FRACASOS (56/70):

- ✓ Traducción automática.
- ✓ Sistemas autoorganizados.
- ✓ Resolución de Problemas mediante lógica
- ✓ Búsqueda heurística.

2ª ETAPA (70-80)

- ✓ Poca confianza en las aplicaciones prácticas de la I.A.
- ✓ Desarrollo de técnicas de búsqueda y representación del conocimiento
- ✓ Prototipos en visión artificial y comprensión y proceso del lenguaje.
- ✓ Interacción de la I.A. con otras disciplinas
- ✓ Interés por la adquisición y por la representación del conocimiento:
INGENIERÍA DEL CONOCIMIENTO

En los años 70, un equipo de investigadores dirigido por Edward Feigenbaum comenzó a elaborar un proyecto para resolver problemas de la vida cotidiana o que se centrara, al menos, en problemas más concretos. Así es como nació el sistema experto.

El primer sistema experto fue el denominado Dendral, un intérprete de espectrograma de masa construido en 1967, pero el más influyente resultaría ser el Mycin de 1974.

El Mycin era capaz de diagnosticar trastornos en la sangre y recetar la correspondiente medicación, todo un logro en aquella época que incluso fueron utilizados en hospitales (como el Puff, variante de Mycin de uso común en el Pacific Medical Center de San Francisco, EEUU).

Ya en los años 80, se desarrollaron lenguajes especiales para utilizar con la Inteligencia Artificial, tales como el LISP o el PROLOG. Es en esta época cuando se desarrollan sistemas expertos más refinados, como por el ejemplo el EURISKO.

Este programa perfecciona su propio cuerpo de reglas heurísticas automáticamente, por inducción.

3ª ETAPA (80-90) (1):

- ✓ “Éxito” Comercial.
- ✓ Creación de Empresas I.A.
- ✓ Desarrollo en grandes empresas de sistemas propios

INVESTIGACIONES EN LA 3ª ETAPA (80-90)

- ✓ Sistemas de acceso a bases de datos en lenguaje natural.
- ✓ Sistemas de visión simplificados
- ✓ Sistemas de síntesis y reconocimiento de voz.

Algunos Sistemas Actuales

- ✓ PEGASUS - hace reservas de transporte por teléfono [Zue et al. 94]
- ✓ MARVEL - sistema experto en tiempo real que maneja la información de Voyager y ayuda a diagnosticar fallas [Schwuttker 92]
- ✓ Sistema de diagnóstico de medicina interna con capacidad de explicar sus razones a los expertos [Heckerman 91]
- ✓ NAVLAB - vehículo autónomo que viaja en forma automática de un extremo a otro de EUA [Pomerleau 93]
- ✓ Sistema de visión que monitorea el tráfico en París y reconoce accidentes con los cuales llama a servicios de emergencia [Koller 94]
- ✓ DEEP BLUE - máquina paralela que juega ajedrez y primera en derrotar al campeón mundial [IBM 97]

FUNDAMENTOS DE LA INTELIGENCIA ARTIFICIAL

Durante más de 2000 años de tradición en filosofía han surgido diversas teorías del razonamiento y del aprendizaje simultáneamente con el punto de vista de que la mente se reduce al funcionamiento de un sistema físico.

Durante más de 400 años de matemáticas, han surgido teorías formales relacionadas con la lógica, probabilidad, teoría de decisiones y la computación.

La psicología ofrece herramientas que permiten la investigación de la mente humana, así como un lenguaje científico para expresar las teorías que se van obteniendo.

La lingüística ofrece teorías sobre la estructura y significado del lenguaje. La ciencia de la computación de la que se toman herramientas que permiten que la inteligencia artificial sea una realidad.

Los filósofos (desde el año 400 AC) permitieron el poder pensar en la inteligencia artificial al concebir a la mente con maneras diversas como una máquina que funciona a partir del conocimiento codificado en un lenguaje interno y al considerar que el pensamiento servía para determinar cuál era la acción correcta que había que emprender.






Las matemáticas proveyeron las herramientas para manipular las aseveraciones de certeza lógica así como las inciertas de tipo probabilista. Así mismo prepararon el terreno para el manejo del razonamiento con algoritmos.

Los psicólogos reforzaron la idea de que los humanos y otros animales podían ser considerados como máquinas para el procesamiento de información.

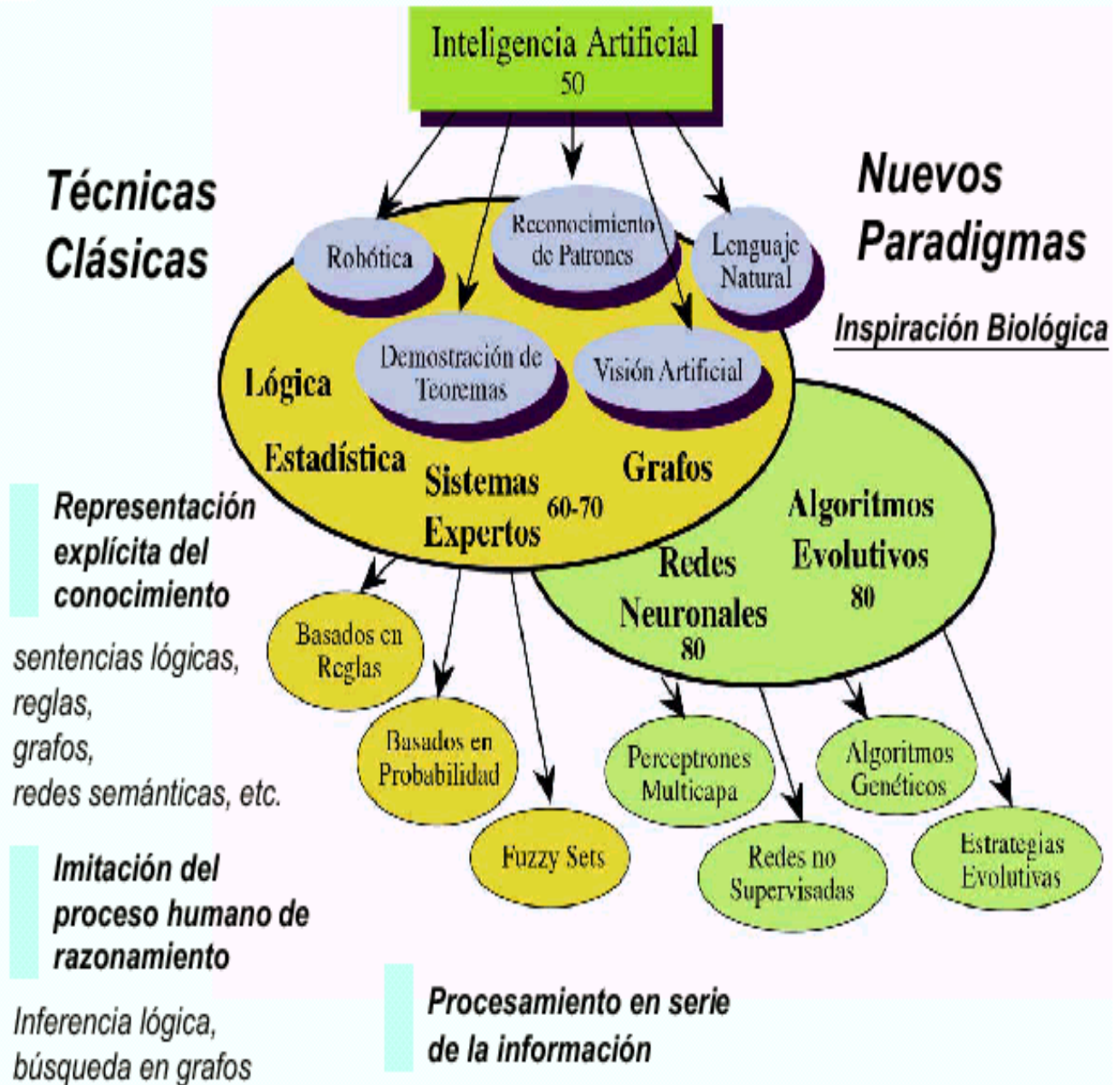
Los lingüistas demostraron que el uso de un lenguaje se ajusta dentro de este modelo.

La ingeniería de cómputo ofreció el dispositivo que permite hacer realidad las aplicaciones de la inteligencia artificial. Los programas de inteligencia artificial por general son extensos y no funcionarían sin los grandes avances de velocidad y memoria apartadas por la industria de cómputo.

En conclusión, las ciencias que aportan a la Inteligencia Artificial son:

-  Filosofía
-  Matemáticas
-  Psicología
-  Lingüística
-  Ciencias de la Computación

Paradigmas en Inteligencia Artificial



TÉCNICAS EN LA INTELIGENCIA ARTIFICIAL

¿Qué es una Técnica de IA?

- La Inteligencia necesita Conocimiento, el cual posee las siguientes propiedades:
 - ✓ es voluminoso
 - ✓ es difícil de caracterizar con exactitud
 - ✓ cambia constantemente
 - ✓ se distingue de los datos en que se organiza de tal forma que se corresponde con la forma en que va a ser utilizado.
- Una Técnica de IA es un método que utiliza conocimiento representado de forma que:
 - ✓ represente las generalizaciones
 - ✓ sea comprensible
 - ✓ pueda modificarse fácilmente
 - ✓ Puede utilizarse en muchas situaciones, aún no siendo totalmente preciso o completo.
 - ✓ puede utilizarse para superar su propio volumen (acotar posibilidades)

a) ÁREAS Y APLICACIONES

- Procesamiento del lenguaje natural.
 - ✓ Comprensión de textos: sintaxis y semántica.
 - ✓ Traducción.
- Recuperación inteligente de la información.
 - ✓ Recuperación de información no explícitamente representada.
 - ✓ Bases de datos deductivas.
 - ✓ Procesos inferenciales. Sentido común.
 - ✓ Interfaz natural.
- SBC: Sistemas Expertos.
 - ✓ Ingeniería del conocimiento: Adquisición, representación, tratamiento y validación.
 - ✓ Representación del conocimiento. Mecanismos inferenciales.
 - ✓ Otras facilidades: explicación, justificación, aprendizaje, etc.
- Demostración automática de teoremas.
 - ✓ Demostración o refutación de teoremas.
- Métodos deductivos
- Robótica.
 - ✓ Percepción. Capacidad locomotiva y manipulativa.
 - ✓ Razonamiento: planificación.
 - ✓ Aprendizaje
- Programación automática.
 - ✓ Compilador de un lenguaje de alto nivel.
 - ✓ Verificador de programas.
- Problemas combinatorios y de planificación.

INTELIGENCIA ARTIFICIAL.

- ✓ Problemas NP: restricción de dominios, heurística.
- ✓ Planificación.
- Percepción
 - ✓ Visión y Habla.
 - ✓ Reconocimiento de formas
 - ✓ Aprendizaje
 - ✓ Aplicaciones en documentación:
 - ✓ Aplicación de los S.E. a la gestión documental:
 - ✓ Obtención Automática de Resúmenes.
 - ✓ Indexación Automatizada.
 - ✓ Herramientas de catalogación.
 - ✓ Búsqueda Inteligente
 - ✓ SOM / WEBSOM.
 - ✓ Agentes de Información.

INTELIGENCIA ARTIFICIAL EN LAS EMPRESAS

Actualmente hay un amplio rango de aplicaciones tecnológicas en las que la computadora juega un papel fundamental, tanto en las áreas científica, educativa, como en los negocios de producción, de servicio y de entretenimiento por mencionar algunos.

Ese uso comprende el acceso a y el intercambio de información a través de nuevas tecnologías como bancos de datos públicos, correo electrónico, Internet e intranet, pero también comprende el uso en las áreas de producción, control, o administración de las empresas.

Los aspectos anteriormente mencionados han planteado nuevos retos a resolver, como es el desarrollo de sistemas de cómputo más flexibles y autónomos, organizados en redes que posibiliten la cooperación entre ellos para contender con volúmenes de información cada vez mayores, de contenido diverso e impreciso.

En particular, la relación entre usuario y computadora demanda una nueva forma de interacción en donde ésta última deje de jugar un papel pasivo y receptor, y se convierta en un participante activo que coopere con el usuario en la solución de problemas.

Ante estos retos, las técnicas y metodologías de la Inteligencia Artificial (IA) han iniciado un repunte como soluciones posibles.

En este artículo se presenta una breve historia de la IA, una reseña general de uno de los pioneros en el campo Herbert Simon, su influencia en el destino de la IA, su contribución. Finalmente se presentan las dos etapas de la IA y se delinean algunas de las perspectivas del campo en las empresas.

Nilsson [2001] propone como meta de estudio de la inteligencia artificial, el comportamiento inteligente de las máquinas, que supone percibir, razonar, aprender, comunicarse y actuar en entornos complejos.

La Inteligencia Artificial, interesada en la síntesis de sistemas que exhiban un comportamiento inteligente, constituye una alternativa promisorio y viable para el desarrollo de las arquitecturas de cómputo requeridas, aptas para resolver los problemas generados por esta nueva cultura informática.

Esto no quiere decir que la disciplina, a pesar de su juventud, no tenga aplicaciones reales. Por el contrario, las soluciones de la IA están presentes en un sin número de aplicaciones, como veremos más adelante.

b) ANTECEDENTES

La búsqueda de los principios que rigen la inteligencia para incorporarlos en una máquina, es antigua. Tenemos constancia de mecanismos físicos y ficticios con los griegos o en otras culturas como la china o incluso la maya.

Revisando la historia de la humanidad, encontramos dos vertientes: una que se inclina por la búsqueda de principios abstractos y otra que se inclina por la construcción de máquinas que tengan, representen o emulen alguna característica del pensamiento humano.

Desde que los griegos buscaban explicar y representar estilos de razonamiento, como es el caso de Aristóteles y sus silogismos. El caso de Gottfried Leibnitz que buscaba encontrar un álgebra capaz de representar todo el conocimiento incluyendo las verdades morales y metafísicas que pudieran lograr un sistema deductivo.

Lo que logró fue la invención de un sistema abstracto llamado cálculo filosófico o raciocinador, faltaba tecnología para implantar las ideas. George Boole desarrolla los principios de la lógica proposicional con la que pretendía recoger algunos fundamentos relativos a la naturaleza y a la constitución de la mente humana, su obra se llamaba “Una investigación sobre las leyes del pensamiento, sobre las cuales están fundamentadas las teorías matemáticas de la lógica y de la probabilidad” (An investigation of the laws of thought on which are founded the mathematical theories of logic and probabilities). G. Frege propuso sistema de notación para el razonamiento mecánico, que es el antecedente del cálculo de predicados, lo denominó “escritura de conceptos”.

Turing (1950) propuso el primer artículo moderno sobre la posibilidad de mecanizar la inteligencia. La inteligencia artificial nace como tal a mediados de los años cincuenta (1956). John McCarthy había coeditado un libro junto con Claude Shannon sobre los fundamentos matemáticos de la teoría de autómatas, orientada a la formalización de las ideas sobre la inteligencia.

Pero la inteligencia era algo más que principios matemáticos así que McCarthy resolvió organizar una conferencia llevada a cabo en Darmouth, con la finalidad de ampliar el panorama de estudios de la inteligencia y la mente humana.

Se pensaron varios nombres para esa conferencia tales como: procesamiento de información compleja, inteligencia maquinística, programación heurística o cognología pero el que se quedó y ha perdurado es el de Inteligencia Artificial.

En esa conferencia se presentaron los trabajos que delinearon el campo de la Inteligencia Artificial. En los inicios, los temas de las investigaciones se orientaban a explorar a encontrar formalismos y mecanismos para representar los problemas, para encontrar métodos automáticos para resolver problemas y se desarrollaban programas de computadora para demostrar teoremas, resolver problemas rompecabezas simples, recuperar información.

Entre estos trabajos destacan los de Allen Newell, Herbert Simon y Cliff Shaw, quienes desarrollaron un programa capaz de demostrar teoremas en lógica proposicional; un programa general solucionador de problemas o GPS (General Problem Solver).

Los trabajos de ellos contribuyeron al desarrollo de la Inteligencia Artificial que consideraba a la búsqueda heurística como uno de los métodos fundamentales para resolver problemas y al sistema de símbolos físicos para representar los problemas y sus soluciones.

Herbert Simon (1916-2001) es conocido en el ámbito de las ciencias de la administración por sus contribuciones a la economía y a la administración que le valieron el premio Nóbel de economía en 1984.

H. Simon propuso la hipótesis del símbolo físico como una forma de representar y modelar el pensamiento humano, esta propuesta fue la línea que siguió el desarrollo de la inteligencia artificial en su primera etapa.

La investigación realizada por Simon abarcaba varios campos tanto de la computación como de la psicología cognitiva (uno de los creadores de ella), la administración y la economía.

Su interés se orientaba a los procesos que usamos las personas tanto para resolver problemas como para tomar decisiones y las implicaciones de estos procesos en las instituciones sociales que formamos. Para ello usaba a la computadora como herramienta para modelar y simular el pensamiento humano.

Esto dio lugar a su teoría de la racionalidad limitada. Después de la publicación de los trabajos publicados del “Teórico lógico” y el “solucionador general de problemas”, Simon anunció (1957) que había máquinas que podían aprender y crear y por tanto en 10 años una computadora podría derrotar a un maestro de ajedrez, probar

nuevos teoremas matemáticos y escribir música de un alto valor estético. Con ello se creo una gran expectativa sobre las posibles aplicaciones de la IA.

Las investigaciones se orientaron a buscar métodos para planear, para razonar, para aprender, para jugar contra un adversario, en particular porque se consideraba que un rasgo distintivo de la inteligencia humana era la capacidad para jugar ajedrez, resolver rompecabezas, problemas que presentaban retos para su representación, para encontrar heurísticas y estrategias de gane, porque las estrategias encontradas eran problemas intratables computacionalmente hablando, es decir, demandaban tiempos de procesamiento extraordinarios o representaban problemas indecidibles, o sin solución óptima.

c) DECAIMIENTO DE LA INTELIGENCIA ARTIFICIAL

Desde el punto de vista de los negocios se esperaban más resultados debido a que se tenían sistemas expertos que emulaban el razonamiento de expertos para diagnosticar, predecir, planear.

Algunos de ellos lograban dar diagnósticos para enfermedades como la meningitis, la prospección de petróleo, optimización de plantas químicas, análisis de bitácoras de búsquedas de petróleo.

Estos avances de la primera ola se vieron restringidos por la incapacidad de los sistemas para adaptarse a entornos inciertos e imprecisos. Y por la limitación de la lógica y los sistemas propuestos para representar el conocimiento asociado con el sentido común, su limitada capacidad de percepción.

E. Dyson, en los 80's predijo que la IA no estaba madura y por tanto no podría ser importante desde el punto de vista comercial hasta que no quedara inmersa en sistemas estratégicamente importantes.

Esa predicción no se cumplió sino hasta mediados de los años 90 por lo que algunas personas consideraron que la IA estaba muerta.

La primera etapa de la IA dejó algunas lecciones del uso de la IA, entre ellas se encuentran: Reconocimiento de que la computadora y la IA no reemplazan a la inteligencia humana, la complementan.

El capital más importante en una empresa son las personas, la experiencia y el conocimiento que han acumulado con el transcurso de los años. La mejor inversión se obtiene en el uso de nuevas tecnologías y en la capacitación del personal

Los sistemas inteligentes ayudarán a mejorar significativamente la computación personal, en el desarrollo de interfaces y de sistemas operativos El uso de la IA en los negocios, se debe enfocar a resolver los problemas de los clientes

d) LA NUEVA IA

Uno de los enfoques más recientes de la Inteligencia Artificial es el denominado: Basado en el Comportamiento. Este enfoque proporciona una noción amplia del concepto de inteligencia, al suponer que ésta depende de la forma en que el comportamiento de un sistema contribuye a realizar sus funciones.

La inspiración más cercana a la Inteligencia Artificial basada en el comportamiento se encuentra en la naturaleza, pues los seres vivos son sistemas de auto-mantenerse, auto-regularse, sus mecanismos internos y su comportamiento están encaminados a preservar su propia existencia, inmersos en medios ambientes dinámicos e imprevisibles.

La Inteligencia Artificial basada en el Comportamiento concibe a los sistemas como grandes rompecabezas integrados por componentes simples que, incrementalmente y siguiendo reglas de interacción no-lineales, conforman sistemas complejos.

Este enfoque nos indica que la construcción de sistemas inteligentes debe hacerse incrementalmente, es decir, que su implementación inicie con la construcción de subsistemas de relativa simplicidad que gradualmente nos permitan generar sistemas mejores.

Lo anterior se ve reforzado por los éxitos hasta ahora alcanzados acerca de la construcción de sistemas (no solo computacionales), basados en el comportamiento, benévolos en la resolución de problemas intratables o costosos por métodos tradicionales, como la búsqueda y recuperación de información en la supercarretera de la información, la comunicación y cooperación entre sistemas, el desarrollo de asistentes personales y la navegación robótica, entre otras.

e) EL REPUNTE DE LA IA

En 1997, en el MIT (Massachusetts Institute of Technology) se llevó a cabo una reunión entre expertos de la IA y compañías vinculadas con grupos de investigación y desarrollo.

El resultado de la reunión fue una evaluación del repunte de la IA y delinear algunas perspectivas del impacto de la IA en los negocios. Con el uso de las técnicas tradicionales, con las nuevas técnicas basadas en redes neuronales, algoritmos genéticos, redes bayesianas, aprendizaje.

La Inteligencia Artificial tiene un repunte. Sony, General Electric, Microsoft tienen grupos de investigación que desarrollan y mejoran productos basados en técnicas de IA.

INTELIGENCIA ARTIFICIAL.

La IA proporciona nuevas formas de comunicación, de acceso y de conexión de las personas con las computadoras, con el conocimiento, con el mundo físico y con otras personas.

- f) El avance de la IA se basa en el avance del hardware y en las tecnologías de infraestructura como las comunicaciones, el Internet.
- g) La promoción de la IA como una opción de inversión se basa en el desarrollo de sistemas que contribuyan a:
 - ✓ El ahorro de dinero mediante la distribución de recursos, la detección de fraudes, la minería de datos y la capacitación
 - ✓ El incremento de la competitividad a través de aplicaciones que ofrezcan ayuda en línea, capturen el conocimiento a bajo costo.
 - ✓ A la creación de nuevas capacidades

En los campos nuevos de la IA que tiene un uso directo o impacto potencial en las empresas, se pueden señalar:

El uso del aprendizaje virtual para la capacitación de personas, para resolver los problemas de actualización, de conservación de la información y del conocimiento.

El desarrollo de la computación emotiva y su uso para generar mejores interfaces, para auxiliar en la recuperación de enfermedades, para posibilitar el uso por personas con capacidades diferentes.

Encontramos que la computación está omnipresente en los aparatos electrodomésticos, vehículos, la ropa (el vestido). Ahora se pretende conjuntar con técnicas de la IA para mejorar la calidad de vida de las personas.

La idea es hacer a las personas más listas a través de proporcionar el acceso y facilitar la presentación de la información.

Mejorar las computadoras para hacerlas menos tontas o lo suficientemente listas para que sean un ayudante (chalán) no un estorbo de las personas. Para facilitar la colaboración, posibilitar la creación de entornos de aprendizaje interactivos, explotar la posibilidad de la tele-presencia.

El trágico ataque a las torres gemelas ha sido un catalizador para el uso de técnicas que están en los laboratorios que no habían salido a los mercados, como el reconocimiento de caras, la minería de datos, entre otros, para mejorar la infraestructuras de comunicaciones, de servicios bancarios, de salud, legales.

La incorporación de esas nuevas tecnologías no implica un olvido de las antiguas formas de trabajo, hay que guardar un balance entre la nueva y la vieja tecnología.

Evitar el espejismo del control, se requiere maneras de capturar la comprensión intuitiva y el conocimiento tácito de los trabajadores y expertos tradicionales.

La inteligencia humana abarca muchas otras habilidades que difícilmente son igualadas por una máquina: la habilidad para percibir y analizar nuestras percepciones tanto táctiles, auditivas, visuales, olfativas; la habilidad para comprender o generar un lenguaje.

Aún cuando hay avances importantes en estos campos, todavía tenemos limitaciones. A la fecha las personas seguimos teniendo ventajas en cuanto a las habilidades de las máquinas, somos más rápidos para extraer y procesar información incompleta, incierta y contradictoria.

1.2. Diferencia entre computación Algorítmica y computación heurística

Una definición informal (no se considera aquí una definición formal, aunque existe): conjunto finito de reglas que dan una secuencia de operaciones para resolver todos los problemas de un tipo dado. De forma más sencilla, podemos decir que un algoritmo es un conjunto de pasos que nos permite obtener un dato. Además debe cumplir estas condiciones:

- *Finitud*: el algoritmo debe acabar tras un número finito de pasos. Es más, es casi fundamental que sea en un número razonable de pasos.
- *Definibilidad*: el algoritmo debe definirse de forma precisa para cada paso, es decir, hay que evitar toda ambigüedad al definir cada paso. Puesto que el lenguaje humano es impreciso, los algoritmos se expresan mediante un lenguaje formal, ya sea matemático o de programación para un computador.
- *Entrada*: el algoritmo tendrá cero o más entradas, es decir, cantidades dadas antes de empezar el algoritmo. Estas cantidades pertenecen además a conjuntos especificados de objetos. Por ejemplo, pueden ser cadenas de caracteres, enteros, naturales, fraccionarios, etc. Se trata siempre de cantidades representativas del mundo real expresadas de tal forma que sean aptas para su interpretación por el computador.
- *Salida*: el algoritmo tiene una o más salidas, en relación con las entradas.
- *Efectividad*: se entiende por esto que una persona sea capaz de realizar el algoritmo de modo exacto y sin ayuda de una máquina en un lapso de tiempo finito.

A menudo los algoritmos requieren una organización bastante compleja de los datos, y es por tanto necesario un estudio previo de las estructuras de datos fundamentales. Dichas estructuras pueden implementarse de diferentes maneras, y es más, existen algoritmos para implementar dichas estructuras. El uso de estructuras de datos adecuadas pueden hacer trivial el diseño de un algoritmo, o un algoritmo muy complejo puede usar estructuras de datos muy simples.

Uno de los algoritmos más antiguos conocidos es el algoritmo de Euclides. El término algoritmo proviene del matemático Muhammad ibn Musa al-Khwarizmi, que vivió aproximadamente entre los años 780 y 850 d.C. en la actual nación Iraní. El describió la realización de operaciones elementales en el sistema de numeración decimal. De al-Khwarizmi se obtuvo la derivación algoritmo.

- Clasificación de algoritmos

- * Algoritmo determinista: en cada paso del algoritmo se determina de forma única el siguiente paso.
- * Algoritmo no determinista: deben decidir en cada paso de la ejecución entre varias alternativas y agotarlas todas antes de encontrar la solución.

Todo algoritmo tiene una serie de características, entre otras que requiere una serie de recursos, algo que es fundamental considerar a la hora de implementarlos en una máquina. Estos recursos son principalmente:

- El tiempo: período transcurrido entre el inicio y la finalización del algoritmo.
- La memoria: la cantidad (la medida varía según la máquina) que necesita el algoritmo para su ejecución.

Obviamente, la capacidad y el diseño de la máquina pueden afectar al diseño del algoritmo.

En general, la mayoría de los problemas tienen un parámetro de entrada que es el número de datos que hay que tratar, esto es, N . La cantidad de recursos del algoritmo es tratada como una función de N . De esta manera puede establecerse un tiempo de ejecución del algoritmo que suele ser proporcional a una de las siguientes funciones:

- **1** : Tiempo de ejecución constante. Significa que la mayoría de las instrucciones se ejecutan una vez o muy pocas.
- **$\log N$** : Tiempo de ejecución logarítmico. Se puede considerar como una gran constante. La base del logaritmo (en informática la más común es la base 2) cambia la constante, pero no demasiado. El programa es más lento cuanto más crezca N , pero es inapreciable, pues $\log N$ no se duplica hasta que N llegue a N^2 .
- **N** : Tiempo de ejecución lineal. Un caso en el que N valga 40, tardará el doble que otro en que N valga 20. Un ejemplo sería un algoritmo que lee N números enteros y devuelve la media aritmética.
- **$N \cdot \log N$** : El tiempo de ejecución es $N \cdot \log N$. Es común encontrarlo en algoritmos como Quick Sort y otros del estilo divide y vencerás. Si N se duplica, el tiempo de ejecución es ligeramente mayor del doble.

- **N^2** : Tiempo de ejecución cuadrático. Suele ser habitual cuando se tratan pares de elementos de datos, como por ejemplo un bucle anidado doble. Si N se duplica, el tiempo de ejecución aumenta cuatro veces. El peor caso de entrada del algoritmo Quick Sort se ejecuta en este tiempo.
 - **N^3** : Tiempo de ejecución cúbico. Como ejemplo se puede dar el de un bucle anidado triple. Si N se duplica, el tiempo de ejecución se multiplica por ocho.
 - **2^N** : Tiempo de ejecución exponencial. No suelen ser muy útiles en la práctica por el elevadísimo tiempo de ejecución. El problema de la mochila resuelto por un algoritmo de fuerza bruta -simple vuelta atrás- es un ejemplo. Si N se duplica, el tiempo de ejecución se eleva al cuadrado.
- * Algoritmos polinomiales: aquellos que son proporcionales a N^k . Son en general factibles.
- * Algoritmos exponenciales: aquellos que son proporcionales a k^N . En general son infactibles salvo un tamaño de entrada muy reducido.

- Notación O-grande

En general, el tiempo de ejecución es proporcional, esto es, multiplica por una constante a alguno de los tiempos de ejecución anteriormente propuestos, además de la suma de algunos términos más pequeños. Así, un algoritmo cuyo tiempo de ejecución sea $T = 3N^2 + 6N$ se puede considerar proporcional a N^2 . En este caso se diría que el algoritmo es del orden de N^2 , y se escribe $O(N^2)$. Los grafos definidos por matriz de adyacencia ocupan un espacio $O(N^2)$, siendo N el número de vértices de éste.

La notación O-grande ignora los factores constantes, es decir, ignora si se hace una mejor o peor implementación del algoritmo, además de ser independiente de los datos de entrada del algoritmo. Es decir, la utilidad de aplicar esta notación a un algoritmo es encontrar un límite superior del tiempo de ejecución, es decir, el peor caso.

A veces ocurre que no hay que prestar demasiada atención a esto. Conviene diferenciar entre el peor caso y el esperado. Por ejemplo, el tiempo de ejecución del algoritmo Quick Sort es de $O(N^2)$. Sin embargo, en la práctica este caso no se da casi nunca y la mayoría de los casos son proporcionales a $N \cdot \log N$. Es por ello que se utiliza esta última expresión para este método de ordenación.

Una definición rigurosa de esta notación es la siguiente:

Una función $g(N)$ pertenece a $O(f(N))$ si y sólo si existen las constantes c_0 y N_0 tales que:

$|g(N)| \leq |c_0 \cdot f(N)|$, para todo $N \geq N_0$.

- Clasificación de problemas

Los problemas matemáticos se pueden dividir en primera instancia en dos grupos:

- * Problemas indecidibles: aquellos que no se pueden resolver mediante un algoritmo.
- * Problemas decidibles: aquellos que cuentan al menos con un algoritmo para su cómputo.

Sin embargo, que un problema sea decidable no implica que se pueda encontrar su solución, pues muchos problemas que disponen de algoritmos para su resolución son inabordables para un computador por el elevado número de operaciones que hay que realizar para resolverlos. Esto permite separar los problemas decidibles en dos:

- * intratables: aquellos para los que no es factible obtener su solución.
- * tratables: aquellos para los que existe al menos un algoritmo capaz de resolverlo en un tiempo razonable.

Los problemas pueden clasificarse también atendiendo a su **complejidad**. Aquellos problemas para los que se conoce un algoritmo polinómico que los resuelve se denominan **clase P**. Los algoritmos que los resuelven son deterministas. Para otros problemas, sus mejores algoritmos conocidos son no deterministas. Esta clase de problemas se denomina **clase NP**. Por tanto, los problemas de la clase **P** son un subconjunto de los de la clase **NP**, pues sólo cuentan con una alternativa en cada paso.

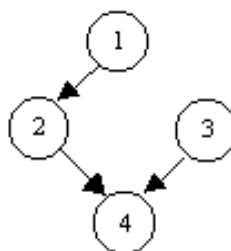
He aquí un ejemplo de aplicación con computación algorítmica; se puede pensar que el conocimiento de estos algoritmos y estructuras de datos no tienen una aplicación práctica inmediata. Sin embargo, su conocimiento y correcta aplicación sirven para producir programas mejores, en el sentido de que aprovechan mejor la memoria del sistema, son más rápidos, eficientes, robustos y tolerantes a fallos.

Las aplicaciones de estos algoritmos en algunos casos son inmediatas; por ejemplo, hallar el trayecto más corto entre dos estaciones es algo que interesa a muchos viajeros del metro y se pueden obtener aproximaciones bastante buenas del mundo real utilizando algunos de los algoritmos que obtienen distancias mínimas. Otros algoritmos sirven para procesar cadenas, lo cual sirve de base para analizadores léxicos o algoritmos criptográficos, por ejemplo.

Además, tener conocimientos adecuados de algoritmia y estructuras de datos facilita el poder pasar de un lenguaje de programación a otro con mucha mayor dinámica puesto que ya se tiene la base, sólo hace falta superar las dificultades técnicas particulares de cada lenguaje.

Veamos un ejemplo con aplicación de computación algorítmica

Definimos una red como un conjunto de elementos más un conjunto de arcos que unen pares de elementos diferentes; consideraremos que los elementos son números naturales de 1 a una cierta cota N . Por ejemplo, una red puede constar de los naturales $\{1, 2, 3, 4\}$ y del conjunto de arcos $\{(1, 2), (2, 4), (3, 4)\}$. Una red puede representarse gráficamente:



Destacamos que los arcos están dirigidos, es decir, los arcos $(1, 2)$ y $(2, 1)$ son diferentes.

Diremos que una secuencia C de naturales está ordenada respecto una red R si se cumplen las tres condiciones siguientes:

- C no contiene elementos repetidos.
- Los elementos de C y de R coinciden.
- Si existe un arco (a, b) en R , entonces a aparece antes que b en C .

Así, las secuencias $1\ 2\ 3\ 4$, $1\ 3\ 2\ 4$ y $3\ 1\ 2\ 4$ están ordenadas respecto la red del ejemplo anterior, pero no sucede lo mismo con las secuencias $1\ 2\ 3\ 2\ 4$, $1\ 2\ 3$ ni $1\ 2\ 4\ 3$, pues violan alguna de las tres condiciones establecidas.

Se pide un programa capaz de determinar si una secuencia está ordenada respecto una red. Dicho programa debe procesar un fichero de entrada, de nombre "RED1.DAT", que contendrá la configuración de la red y diversas secuencias. El formato exacto es el siguiente:

Línea 1: un carácter numérico (dígito) que representa el valor de N (comprendido entre 1 y 9).

Líneas siguientes: cada línea contendrá una o más parejas de valores naturales, representando cada una de ellas un arco entre dos elementos de la red. Cada natural se representa mediante un dígito; dos naturales consecutivos están siempre separados por un único espacio en blanco.

Una línea que contiene únicamente una aparición del carácter '0'.

Líneas siguientes: cada línea representa una secuencia a examinar y contendrá diversos valores naturales, separados por un único espacio en blanco, y representados cada uno de ellos por un dígito.

INTELIGENCIA ARTIFICIAL.

La salida debe almacenarse en el fichero "RED1.RES" y debe contener tantas líneas como secuencias a examinar. Cada línea contendrá o bien la palabra 'SI' o bien la palabra 'NO' (en mayúsculas) dependiendo de si la secuencia correspondiente está ordenada respecto la red o no.

A continuación, se codifica la red del ejemplo anterior y algunas secuencias respecto ella, así como el resultado esperado:

<u>RED1.DAT</u>	<u>RED1.RES</u>
4	SI
1 2 3 4 2 4	NO
0	NO
1 2 3 4	SI
1 2 3 2 4	NO
1 2 3	SI
1 3 2 4	
2 1 3 4	
3 1 2 4	

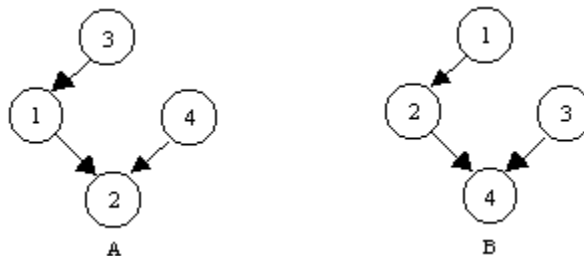
Para simplificar el problema, podéis suponer en este apartado y en todos los que siguen, que no habrá errores de formato en la entrada.

Ahora vamos a considerar el siguiente escenario:

Decimos que dos redes A y B son similares si se cumplen las condiciones siguientes:

- Tienen los mismos elementos.
- B tiene los mismos arcos que A, utilizando una reenumeración de los elementos de A.

Por ejemplo, la red A con elementos {1, 2, 3, 4} y arcos {(3, 1), (4, 2), (1, 2)} es similar a la red del apartado anterior (llamémosla B) si reenumeramos los elementos de A de la forma siguiente: 1, como 2; 2, como 4; 3, como 1; y 4, como 3. La similitud queda patente dibujando ambas redes una junto a otra:



Se pide construir un programa que, dadas dos redes, determine si son similares o no. Puede suponerse que, en caso de ser similares, habrá una sola reenumeración de los elementos de la primera de las redes que permita obtener la condición de similitud. El programa debe procesar el fichero de entrada, de nombre "RED2.DAT", que contendrá la configuración de ambas redes. El formato exacto es el siguiente:

Línea 1: un carácter numérico (dígito) que representa el valor de N para la red A.

Líneas siguientes: cada línea contendrá una o más parejas de valores naturales, representando cada una de ellas un arco entre dos elementos de la red A. Cada natural se representa mediante un dígito; dos naturales consecutivos están siempre separados por un único espacio en blanco.

Una línea que contiene únicamente una aparición del carácter '0'. Línea siguiente: un carácter numérico (dígito) que representa el valor de N para la red B.

Líneas siguientes: cada línea contendrá una o más parejas de valores naturales, representando cada una de ellas un arco entre dos elementos de la red B. Cada natural se representa mediante un dígito; dos naturales consecutivos están siempre separados por un único espacio en blanco.

La salida debe almacenarse en el fichero "RED2.RES" y debe contener una única línea. Dicha línea contendrá la palabra 'NO' (en mayúsculas) si las redes no son similares, mientras que si lo son deberá contener la reenumeración (única) que asegura dicha condición. La reenumeración se escribirá como una secuencia de valores naturales, separados por un único espacio en blanco, y representados cada uno de ellos mediante un dígito; el valor que aparece en posición k en la línea representa la reenumeración del elemento k de A. Así, para el ejemplo anterior, el fichero "RED2.RES" debe contener los valores 2 4 1 3, en este orden:

<u>RED2.DAT</u>	<u>RED2.RES</u>
4	2 4 1 3
4 2 3 1 1 2	
0	
4	
1 2 3 4 2 4	

Debemos considerar que en computación, dos objetivos fundamentales son encontrar algoritmos con buenos tiempos de ejecución y buenas soluciones, usualmente las óptimas. Una **heurística** es un algoritmo que abandona uno o ambos objetivos; por ejemplo, normalmente encuentran buenas soluciones, aunque en ocasiones no hay pruebas de que la solución no pueda ser

arbitrariamente errónea; o se ejecuta razonablemente rápido, aunque no existe tampoco prueba de que deba ser así.

A menudo, pueden encontrarse instancias concretas del problema donde la heurística producirá resultados muy malos o se ejecutará muy lentamente. Aún así, estas instancias concretas pueden ser ignoradas porque no deberían ocurrir nunca en la práctica por ser de origen teórico, y el uso de heurísticas es muy común en el mundo real.

Heurísticas para encontrar el camino más corto

Para problemas de búsqueda del camino más corto el término tiene un significado más específico. En este caso una *heurística* es una función matemática, $h(n)$ definida en los nodos de un árbol de búsqueda, que sirve como una estimación del coste del camino más económico de un nodo dado hasta el nodo objetivo. Las heurísticas se usan en los algoritmos de búsqueda informada como la búsqueda egoísta. La búsqueda egoísta escogerá el nodo que tiene el valor más bajo en la función heurística. A* expandirá los nodos que tienen el valor más bajo para $g(n) + h(n)$, donde $g(n)$ es el coste (exacto) del camino desde el estado inicial al nodo actual. Cuando $h(n)$ es *admisible*, esto es si $h(n)$ nunca sobrestima los costes de encontrar el objetivo; A* es probablemente óptimo.

Un problema clásico que usa heurísticas es el puzzle-n. Contar el número de casillas mal colocadas y encontrar la suma de la distancia Manhattan entre cada bloque y su posición al objetivo son heurísticas usadas a menudo para este problema.

Efecto de las heurísticas en el rendimiento computacional

En cualquier problema de búsqueda donde hay b opciones en cada nodo y una profundidad d al nodo objetivo, un algoritmo de búsqueda ingenuo deberá buscar potencialmente entre b^d nodos antes de encontrar la solución. Las heurísticas mejoran la eficiencia de los algoritmos de búsqueda reduciendo el factor de ramificación de b a (idealmente) una constante b^* .

Aunque cualquier heurística admisible devolverá una respuesta óptima, una heurística que devuelve un factor de ramificación más bajo es computacionalmente más eficiente para el problema en particular. Puede demostrarse que una heurística $h_2(n)$ es mejor que otra $h_1(n)$, si $h_2(n)$ domina $h_1(n)$, esto quiere decir que $h_1(n) < h_2(n)$ para todo n .

Heurísticas en la Inteligencia Artificial

Muchos algoritmos en la inteligencia artificial son heurísticos por naturaleza, o usan reglas heurísticas. Un ejemplo reciente es Spam Assassin que usa una amplia variedad de reglas heurísticas para determinar cuando un correo

electrónico es spam. Cualquiera de las reglas usadas de forma independiente pueden llevar a errores de clasificación, pero cuando se unen múltiples reglas heurísticas, la solución es más robusta y creíble. Esto se llama alta credibilidad en el reconocimiento de patrones (extraído de las estadísticas en las que se basa). Cuando se usa la palabra heurística en el procesamiento del lenguaje basado en reglas, el reconocimiento de patrones o el procesamiento de imágenes, es usada para referirse a las reglas.

1.3. Lenguajes proposicionales

La lógica proposicional es una rama de la lógica clásica que estudia las proposiciones o sentencias lógicas, sus posibles evaluaciones de verdad y en el caso ideal, su nivel absoluto de verdad.

La lógica proposicional se preocupa por la manera de representar las cosas.

Proposición: se define una proposición como un enunciado declarativo que puede ser verdadero o falso, pero no ambos a la vez. Las proposiciones se representan mediante variables proposicionales simbolizadas mediante letras.

1.4. Estructuras básicas de los lenguajes proposicionales

Los patrones o expresiones de la lógica proposicional se construyen a partir de un alfabeto que consta de los siguientes símbolos:

- Las constantes lógicas **Verdadero** (T) y **Falso** (\perp). También pueden ser V o F
- Los símbolos de variables tales como **P** y **Q**.
- Los conectivos lógicos \neg , \cup , \cap , \rightarrow , y \leftrightarrow
- Símbolos de puntuación: paréntesis (), corchetes [] y llaves { } para evitar ambigüedades

Todas las oraciones se forman combinando los símbolos anteriores mediante ciertas reglas.

- Las constantes lógicas *Verdadero* y *Falso* constituyen oraciones en sí mismas
- Las variables proposicionales P, Q, R,... son oraciones
- Encerrar entre paréntesis una oración produce también una oración, por ejemplo

$(P \cup Q)$.

Combinar oraciones con los conectadores lógicos siguientes forma una oración

Símbolo	Ejemplo	Nombre	Nombre de la oración
\wedge	$P \wedge Q$	Y	Conjunción
\vee	$P \vee Q$	O	Disyunción
\neg	$\neg P$	No	Negación
\rightarrow	$P \rightarrow Q$	Implicancia	implicancia
\leftrightarrow	$P \leftrightarrow Q$	equivalencia	Equivalencia (bicondicional)

Oraciones: son Un conjunto de palabras con sentido gramatical.

- La oración es la mínima unidad comunicacional, con significado completo.
- La oración en la lógica, es la unidad de análisis fundamental.
- **Conjunción (\wedge) (y).** A la oración cuyo conector principal es \wedge (y) se le llama conjunción, y a sus partes se les llama coyuntos.
- **Disyunción (\vee) (o).** A la oración cuyo conector principal es \vee (o) se le llama disyunción, y a sus partes se les llama disyuntos.
- **Implicación (\rightarrow).** Una oración como $P \rightarrow R$ se conoce como **implicación** (o condicional), su **premisa** o **antecedente** es P y su **conclusión** o **consecuente** es R . A las implicaciones también se les llama **reglas** o aseveraciones **si-entonces**.
- **Premisas.** Son los antecedentes de una implicación.
- **Equivalencia.**
 - Dos sentencias α y β son equivalentes lógicamente si es que son verdaderas con el mismo conjunto de hechos.
- **Negación (\neg) (no).**
 - A una oración como $\neg P$ se le llama **negación** de P . \neg es el único de los conectores que funcionan como una sola oración.

FORMALIZAR LOS RAZONAMIENTOS:

1. " Si el resultado obtenido es superior al previsto en 5 unidades, será debido a no haber realizado el proceso a la temperatura adecuada o a la existencia de errores en los cálculos finales."

Solución

p = Resultado obtenido menor al previsto en 5 unidades.

q = Haber realizado el proceso a la temperatura adecuada.

r = Existencia de errores en los cálculos finales.

q r p

2) " El análisis realizado, innecesario si nos dejamos llevar por la precipitación, se torna necesario si nos paramos a reflexionar sobre el mensaje que se pretende transmitir."

solución

p = Análisis realizado es necesario.

q = Nos dejamos llevar por la precipitación.

r = Nos paramos a reflexionar sobre el mensaje que se pretende transmitir.

q p r p

3)" El cáncer no logrará curarse a no ser que se logre determinar su causa y se consiga encontrar fármacos adecuados o bien para prevenirlo o para curarlo."

solución

p = El cáncer logrará curarse.

q = Se logra determinar su causa.

r = Se consigue encontrar fármacos adecuados para prevenirlo.

s = Se consigue encontrar fármacos adecuados para curarlo.

q r s p

SEMÁNTICA DE LA LÓGICA DEL PROPOSICIONAL

- Una interpretación asocia cada variable proposicional con una proposición sobre el mundo. Porque las proposiciones son o verdades o falso, podemos también especificar una interpretación asignando los valores de verdad (VERDAD y FALSO) directamente a las variables proposicionales, sin importar qué proposición cada uno denota.

- Cada conector lógico es definido por una tabla de verdad
Dado una interpretación de las variables proposicionales, nosotros podemos utilizar una tabla de verdad para calcular el valor de verdad de cualquier oración bajo esa interpretación

En términos generales, una *semántica* permite atribuir un significado a las expresiones del lenguaje simbólico considerado. En el caso de un lenguaje de programación como C, esta semántica es procedural y consiste en describir el efecto que produce el programa sobre sus estructuras de datos. **Para un lenguaje de representación, lo que interesa es capturar una descripción del universo modelado.** La lógica permite hacer esto asignando un valor de verdad a cada expresión del lenguaje.

La semántica de un lenguaje proposicional depende de

1. La interpretación de los conectivos lógicos, que tienen el mismo significado en todos los dominios,
2. Los valores de verdad asignados a las variables proposicionales, distintos según la situación reflejada

TABLAS DE VERDAD

Se emplean en la lógica para determinar los posibles valores de verdad de una expresión o proposición. O si un esquema de inferencia, como argumento, es formalmente válido mostrando que, efectivamente, es una tautología.

La tabla de verdad de una sentencia es una tabla en la que se presentan todas las posibles interpretaciones de las variables proposicionales que constituyen la sentencia y el valor de verdad de la sentencia para cada interpretación.

Dado que en el cálculo proposicional se opera sólo sobre dos valores de verdad, para cualquier expresión existe un número finito de valuaciones posibles que se pueden tabular.

La tabla de verdad de una expresión con n variables proposicionales tiene 2^n filas

Semántica

- **Negación** Consiste en cambiar el valor de verdad de una variable proposicional.

INTELIGENCIA ARTIFICIAL.

p	$\neg p$
V	F
F	V

- **Disyunción:** La sentencia será verdadera cuando una o ambas variables proposicionales sean verdaderas.

p	q	$p \vee q$
V	V	V
V	F	V
F	V	V
F	F	F

- **Conjunción :**La sentencia será verdadera sólo cuando ambas variables proposicionales sean verdaderas.

p	q	$p \wedge q$
V	V	V
V	F	F
F	V	F
F	F	F

- **Condicional**

La sentencia será verdadera cuando se cumpla si es válido p entonces lo es q.

p	q	$p \rightarrow q$
V	V	V
V	F	F
F	V	V
F	F	V

- **Bicondicional**

La sentencia será verdadera cuando ambas variables proposicionales sean iguales.

p	q	$p \leftrightarrow q$
V	V	V
V	F	F
F	V	F
F	F	V

- **Disyunción exclusiva**

La sentencia será verdadera sólo cuando sólo una de las dos variables proposicionales sea verdadera, pero no las dos.

P	q	$p \oplus q$

V	V	F
V	F	V
F	V	V
F	F	F

EQUIVALENCIA LÓGICA

Dos formulas A ; B se dicen equivalentes (se denota por B ó AB) si para toda interpretación I , se cumple que $V_i(A) = V_i(B)$

Teorema : $A \equiv B$ si y sólo si la fórmula $A \equiv B$ es válida

A continuación se presenta una tabla con una serie de equivalencias de uso común

1. Supresión de Implicación:

1.1

2. Contraposición:

2.1

3. Supresión de Doble Implicación:

3.1

4. Absorción:

- 1.
- 2.

5. Elemento neutro (identidad)

1. $A \vee A$
2. $A \wedge A$

5.3 $A \wedge F$

5.4 A V V

6. Complementario- Contradicción

6.1 A A F

6.2 A A V

F V

V F

7. Idempotencia

8. Commutativa

9. Asociativa

1.

2.

10. Distributiva

1.

11. De Morgan

12. Doble Negación

VALIDEZ E INFERENCIA

Los términos "razonamiento" e "inferencia" son utilizados para referirse a cualquier proceso mediante el que se obtienen conclusiones.

Las tablas de verdad sirven no solo para definir los conectores, sino también para probar la validez de las oraciones. Si se desea considerar una oración, se construye una tabla de verdad con una hilera por cada una de las posibles combinaciones de valores de verdad correspondientes a los signos propositivos de la oración. Se calcula el valor de verdad de toda la oración, en cada una de las hileras. Si la oración es verdadera en cada una de las hileras. La oración es válida.

Las tablas nos manifiestan los valores de verdad de cualquier proposición, así como el análisis de los mismos, encontrándonos con los siguientes casos:

- **Tautología o validez:**

Se entiende por proposición tautológica, o tautología, aquella proposición que en todos los casos posibles de su tabla de verdad su valor siempre es V.

- **Contradicción:**

Se entiende por proposición contradictoria, o contradicción, aquella proposición que en todos los casos posibles de su tabla de verdad su valor siempre es F

- **Contingencia (verdad indeterminada)**

Se entiende por verdad contingente, o verdad de hecho, aquella proposición que puede ser verdadera o falsa, o no se tiene suficiente información para llegar a una conclusión

- **Satisfabilidad.**

Si en la tabla de verdad se obtiene al menos una VERDAD

EJEMPLO:

1. Determinar La Validez De La Siguiete oración compleja

$$((P \vee H) \wedge \neg P) \Rightarrow P$$

Solución

P	H	$P \vee H$	$(P \vee H) \wedge \neg P$	$((P \vee H) \wedge \neg P) \Rightarrow P$
F	F	F	F	V
F	V	V	F	V
V	F	V	V	V
V	V	V	F	V

Respuesta: sí es valida

2. Compruébese si los siguientes razonamientos son correctos o no:

- o. Si no llueve salgo al campo. Si salgo al campo respiro. Por tanto, respiro si y sólo si no llueve."

Respuesta:

NO es válido, puedo salir al campo, lloviendo y respirar. Luego no se deduce que respire si y solo si no llueve.

- Si ha nevado será difícil conducir. Si no es fácil conducir llegaré tarde si no salgo temprano. Ha nevado. Luego saldré temprano.

Respuesta

El razonamiento NO es válido porque puede darse el caso de NO salir temprano y llegar tarde habiendo nevado y siendo difícil conducir. Cumpliéndose todas las premisas.

REGLAS DE INFERENCIA

- Existen ciertos patrones de inferencia que se presentan una y otra vez, lo que permite establecer de una vez por todas su confiabilidad.
- La regla permite evitar pasar por las tablas de verdad.

3. Modus ponens o implicación-Eliminación:

A partir de una implicación y la premisa de la implicación, se puede inferir la conclusión.

$$\frac{A \Rightarrow B, A}{B}$$

➤ **Y- Eliminación: (eliminación de ^)**

A partir de una conjunción se puede inferir cuales son los coyuntos(elementos)

$$\frac{A1 \wedge A2 \wedge A3..... \wedge An}{A1}$$

➤ **Y- Introducción (Introducción del ^)**

A partir de una lista de oraciones es posible inferir su conjunción

$$\frac{A1, A2, A3, \dots, An}{A1 \wedge A2 \wedge A3 \dots \wedge An}$$

➤ **O- Introducción (Introducción del \vee)**

A partir de una oración es posible inferir su disyunción con todo lo demás.

$$\frac{A1}{A1 \vee A2 \vee A3 \dots \vee An}$$

➤ **Eliminación de la doble negación:**

A partir de una oración doblemente negada, es posible inferir una oración positiva

$$\frac{\neg \neg A}{A}$$

➤ **Resolución unitaria**

A partir de una disyunción, si uno de los disyuntos es falso, entonces se puede inferir que el otro es verdadero.

$$\frac{A \neg B, B, \neg B}{A}$$

➤ **resolución:**

Es la mas difícil. Puesto que B no puede ser al mismo tiempo verdadera ni falsa, uno de los otros disyuntos debe ser en una de las premisas. O también, que la implicación es transitiva.

$$\frac{A \vee B, \neg B \vee A}{A \vee B} \quad \circ \text{ equivalente } \frac{\neg A \Rightarrow B, B \Rightarrow A}{\neg A \Rightarrow A}$$

EJEMPLO:

- Utilice la tabla de verdad para determinar para demostrar que la siguiente oración es valida y que por lo tanto la equivalencia es correcta

$$P \wedge (q \vee r) \vee (p \wedge r)$$

p	q	r	$P \wedge (q \vee r) \vee (p \wedge r)$									
V	V	V	V	V	V	V	V	F	F	V	V	V
V	V	F	V	F	F	V	V	F	F	F	F	F
V	F	V	V	V	V	V	V	V	V	V	V	V
V	F	F	V	V	V	V	V	V	V	V	V	F
F	V	V	F	F	V	V	F	F	F	F	F	F
F	V	F	F	F	F	V	F	F	F	F	F	F
F	F	V	F	F	V	V	F	F	V	F	F	F
F	F	F	F	F	V	V	F	F	V	F	F	F



TAUTOLOGIA

Por tanto: $P \wedge (q \vee r) \vee (p \wedge r)$, Es válida y equivalente

- Haciendo uso de la lógica equivalente simplificar las siguiente proposición
- $(P \wedge q)$

$$(P \wedge q)$$

$$P \wedge (q \vee r) \dots\dots\dots R. Distributiva(10.2)$$

$$P \wedge (V) \dots\dots\dots R. Complementaria (6.2)$$

PR. Identidad (5.1)

- Haciendo uso de las reglas de inferencia Demostrar que :

$p \vee q, q \vdash p$

1. $p \vee q$ **Premisa**
2. q Regla. Eliminación de \vee (1)
3. p Regla. Eliminación de \vee (1)
4. p Regla. Introducción del 2,3)

1.5. Programas en LISP Y PROLOG

COMPONENTES DE UN SISTEMA LISP.

Un componente importante de un sistema LISP es la gestión dinámica de la memoria. El sistema administrará el espacio en la memoria para las listas en constante modificación, sin que el usuario lo deba solicitar.

Libera los espacios de memoria que ya no son necesarios y los pone a disposición de usos posteriores. La necesidad de este proceso se deriva de la estructura básica de LISP, las listas, que se modifican de forma dinámica e ilimitada.

Además un sistema LISP abarca bastante más que el solo intérprete del lenguaje LISP.

Consta de algunos cómodos módulos que ofrecen ayuda en el desarrollo y control del progreso en programas, como son el Editor, el File-System y el Trace. Por supuesto estos módulos sólo están en versiones de LISP que contengan la conocida interfaz gráfica IDE típica de los modernos lenguajes visuales.(IDE = entorno de desarrollo integrado).

LISP

El nombre LISP es la abreviatura de List-Processing, ya que el LISP fue desarrollado para el procesamiento de listas. La lista es la estructura más importante de LISP.

El lenguaje LISP fue diseñado ya a finales de los años 50 por McCarthy. A lo largo de los últimos años se han desarrollado muchos dialectos, por ejemplo MACLISP, COMMONLISP, INTERLISP, ZETALISP, donde el COMMONLISP se está imponiendo cada vez más como estándar.

Se presenta ahora ya el LISP como lenguaje propio, dejando ya atrás su historia. En LISP se dan los siguientes conceptos característicos:

Listas y Átomos: La estructura más importante es la lista. Los átomos pueden subordinarse a cualidades.

La Función: Cada función LISP y cada programa LISP tiene estructura de lista. Los programas no pueden distinguirse sintácticamente de los datos. LISP ofrece sus propias funciones básicas.

Forma de Trabajo: LISP es un lenguaje funcional. Ofrece la posibilidad de realizar definiciones recursivas de funciones. La unión de procedimientos se realiza de forma dinámica, es decir en plena ejecución, y no como en otros lenguajes de programación.

El sistema realiza automáticamente una gestión dinámica de memoria.

La estructura más importante en LISP es la lista por lo que, se muestra un pequeño ejemplo:

Ejemplo :

(A (B C) D) es una lista con tres elementos
A átomo
 (B C) lista de átomos B y C
D átomo

También está permitida una lista vacía, "()" ó " NULL", que significa lo mismo.

Con esta estructura podemos configurar estructuras de cualquier complejidad, tan grandes como queramos.

Los átomos son números, cadenas de caracteres o símbolos. Un símbolo puede tener varios valores, al igual que una variable en otros lenguajes de programación, como por ejemplo un número, o también puede ser el nombre de una función, o incluso ambos.

Además a un símbolo pueden subordinarse cualidades, que además del valor del símbolo, contienen información adicional. Estas cualidades también reciben el nombre de atributos.

PROLOG

PROLOG es la abreviatura de **PRO**gramación **LOG**ica, con lo que hacemos mención a la procedencia del lenguaje: Es una realización de lógica de predicados, como lenguaje de programación.

En la actualidad, el PROLOG se aplica como lenguaje de desarrollo en aplicaciones de Inteligencia Artificial en diferentes proyectos de Europa. En los Estados Unidos, el LISP está más extendido que el PROLOG. Pero para la mayoría de los terminales de trabajo de Inteligencia Artificial se ofrece también el PROLOG.

Como una especie de estándar se han establecido el DECsystem-10 PROLOG de Edimburgo y el PROLOG descrito en el libro "PROGRAMMING IN PROLOG" de W. F. Clocksin y C. S. Melish. La mayoría de los dialectos PROLOG se basan en este y contienen el DECsystem-10 PROLOG en su ámbito lingüístico.

Al contrario que el LISP (y otros lenguajes), en el PROLOG los programas son confeccionados de forma distinta. A los interesados en pueden leer a Kowalski que escribió un artículo con el título "Algorithm = Logic + Control". Con esto pretende decirnos que los algoritmos pueden ser divididos en su lógica y en sus mecanismos de control.

La lógica se representa en forma de predicados. Estos predicados aparecen en tres formas distintas: como hechos, como reglas y como preguntas. La lógica formulada

como hechos y reglas se define como base de conocimientos. A esta base de conocimientos se le pueden formular preguntas.

Los mecanismos importantes del PROLOG son: recursividad, instanciación, verificación, unificación, backtracking e inversión.

La Recursividad representa la estructura más importante en el desarrollo del programa. En la sintaxis del PROLOG no existen los bucles FOR ni los saltos; los bucles WHILE son de difícil incorporación, ya que las variables sólo pueden unificarse una sola vez.

La recursión es más apropiada que otras estructuras de desarrollo para procesar estructuras de datos recursivas como son las listas y destacan en estos casos por una representación más sencilla y de mayor claridad.

La Instanciación es la unión de una variable a una constante o estructura. La variable ligada se comporta luego como una constante.

La Verificación es el intento de derivar la estructura a comprobar de una pregunta desde la base de conocimientos, es decir, desde los hechos y reglas. Si es posible, la estructura es verdadera, en caso contrario es falsa.

La Unificación es el componente principal de la verificación de estructuras. Una estructura estará comprobada cuando puede ser unificada con un hecho, o cuando puede unificarse con la cabecera de una regla y las estructuras del cuerpo de dicha regla pueden ser verificadas.

EL PROGRAMA PROLOG TIENE CARACTERÍSTICAS COMO:

Consta de una secuencia de oraciones, que guardan una relación implícita de conjunción. Todas las variables tienen cuantificación universal implícita y cuando las variables aparecen en creaciones distintas se consideran como diferentes,

Se aceptan únicamente oraciones en forma de cláusula de Horn. Es decir, las oraciones son atómicas, o bien una implicación que no tiene antecedentes negados y un consecuente atómico

Los términos pueden ser signos de constante, variables o términos funcionales. En vez de utilizar antecedentes negados en sus implicaciones, en Prolog se emplea un operador de negación como falla: una meta no P se considera demostrada si el sistema no logra demostrar P.

Todos los términos diferentes desde el punto de vista sintáctico se consideran como que refieren a objetos diferentes. Es decir, no es posible afirmar que:

$A = B$ o que $A = F(x)$, en donde A es una constante.

Es posible afirmar que:

$x=B$ o que $x=F(y)$, en donde x es una variable.

Se cuenta con una gran cantidad de predicados incorporados para aritmética, de entrada/salida y diversas funciones del sistema y de la base de conocimientos.

Las literales que hacen uso de estos predicados se "demuestran" mediante la ejecución del código en vez de hacerlo por inferencia adicional

COMPILACIÓN DE PROGRAMAS LÓGICOS

En la interpretación de programas de cualquier lenguaje, Prolog necesariamente resulta más lento que la ejecución de código compilado.

La razón es que el intérprete se comporta siempre como si nunca hubiese visto el programa antes. El intérprete de Prolog debe recuperar las bases de datos para encontrar oraciones que satisfagan la meta y un análisis de la estructura de la oración para decidir que submetas generar.

Toda la parte medular, para trabajo pesado, de la programación de Prolog se realiza mediante código compilado. La gran ventaja de la compilación es que cuando llega el momento de ejecutar el procedimiento de inferencia, se utilizan rutinas de inferencia, específicamente diseñadas para las oraciones de base de conocimientos.

Prolog básicamente lo que hace lo que hace es generar un demostrador de teoremas en miniatura para cada predicado diferente con lo que elimina mucho exceso de interpretación

El conjunto de instrucciones de las computadoras actuales resulta muy pobre en relación con la semántica de Prolog, por lo que la mayoría de los compiladores realizan una compilación en un lenguaje intermedio en vez de hacerlo directamente en lenguaje de máquina.

El lenguaje intermedio más popular es el Warren Abstract Machine o WAM, nombrado así en honor de David H. D. Warren, uno de los autores de la implantación del primer compilador de Prolog.

El WAM es un conjunto de instrucciones abstractas utilizable en Prolog, y que se puede interpretar o traducir a lenguaje de máquina. Existen otros compiladores que traducen Prolog a un lenguaje de alto nivel tal como "Lisp" o "C", y utilizan dicho compilador para traducir a lenguaje máquina.

SISTEMAS EXPERTOS BASADOS EN PROLOG

La programación lógica considera al programa y a sus entradas como aseveraciones lógicas acerca del mundo, y al procedimiento de hacer explícitas las consecuencias como un proceso de inferencia.

La relación entre lógica y algoritmo está resumida en la denominada “ecuación de Robert Kowalski”:

Algoritmo= Lógica + Control.

Prolog ha sido hasta ahora el lenguaje de programación lógica que más se ha utilizado para permitir escribir algoritmos al complementar las oraciones lógicas con información para control del procedimiento de inferencia.

Debemos considerar a sus usuarios que ascienden a cientos de miles, se le utiliza fundamentalmente como un lenguaje rápido para elaborar prototipos y en tareas donde hay que manejar signos, como el diseño de compiladores y el análisis gramatical del lenguaje natural. También se le ha utilizado en el diseño de aplicaciones de sistemas expertos en las áreas legal, médica, financiera y otras.

UNIDAD II

SISTEMAS EXPERTOS

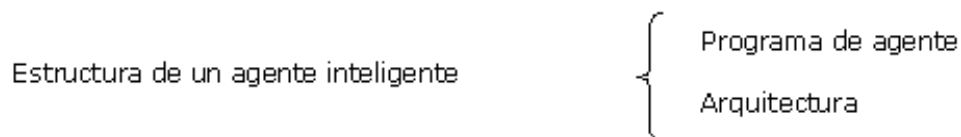
2.1. Definición de Agentes Inteligentes

Agente, del latín agere, es **"el que hace"**. En el ámbito de los negocios, un agente es "aquel que tiene por oficio gestionar negocios ajenos".

Un agente inteligente es un programa de computador que es capaz de hacer lo que haríamos nosotros si tuviéramos tiempo, o de forma más precisa **"un tipo de programa informático que, por encargo de un usuario u otro programa, realiza de forma autónoma tareas que requieren cierto grado de inteligencia y aprendizaje"**.

2.1.1. Estructura de Agentes Inteligentes

La base de la Inteligencia Artificial es el diseño de un programa de agente: Una función que permita implantar el mapeo del agente para pasar de percepciones a acciones. Este programa se ejecutará en algún tipo de dispositivo de cómputo al que se denominará arquitectura. La arquitectura puede ser una computadora sencilla o un hardware especial.



$$\text{agente} = \text{arquitectura} + \text{programa}$$

Elementos básicos que se consideran en la elección de los tipos de agente:

<i>Tipo de agente</i>	<i>Percepciones</i>	<i>Acciones</i>	<i>Metas</i>	<i>Ambiente</i>
Sistema diagnóstico médico	-Sistemas -Evidencias -Respuestas del paciente	-Preguntas -Pruebas -Tratamientos	Paciente saludable	Paciente hospital
Robot clasificador de partes	Pixels de intensidad variable	Recoger partes y clasificarlas	Poner las partes en el bote correspondiente	Banda transportadora

Tipos de Agente y sus descripciones PAMA

PAMA (Percepciones, Acciones, Metas y Ambiente)

Softbots (Agentes de software o robots de software)

Programas de Agente

function SKELETON-AGENT (percept) **returns** Action

función ESQUELETO-AGENTE(percepción) **responde con una** acción

estática: memoria (la memoria del mundo del agente)

memoria → ACTUALIZACION-MEMORIA(*memoria*,
percepción)

acción → ESCOGER-LA-MEJOR-ACCION(*memoria*)

memoria → ACTUALIZACION-MEMORIA(*memoria*, *acción*)

responde con una acción

El esqueleto de un agente. Cada vez que así se solicite, se actualiza la memoria para que refleje la nueva percepción, se escoge la mejor acción y también se consigna en la memoria la acción emprendida. La memoria persiste de una solicitud a otra.

function TABLE-DRIVEN-AGENT (percept) **returns** action

función AGENTE-CONDUCTIVO-MEDIANTE-TABLA (percepción)
responde con una acción

estático: percepciones, una secuencia, originalmente está vacía

tabla: una tabla indexada mediante secuencia de percepciones originalmente especificada en su totalidad

añadir la percepción al final de todas las percepciones

acción → CONSULTA(*percepciones*, *tabla*)

devolver acción

Un agente basado en una tabla de consulta previamente especificada. Se mantiene al tanto de la secuencia de percepciones y se limita a definir cuál es la mejor acción. Carecen de autonomía, pero son válidos.

Antes de proceder al diseño de un programa de agente es necesario contar con una idea bastante precisa de las posibles percepciones y acciones que intervendrán, qué metas o medidas de desempeño se supone lleve a cabo el agente, así como del tipo de ambiente en que tal agente operará, para tal efecto, denominamos a la matriz PAMA (percepciones, acciones, metas y ambientes).

2.1.2. Tipos de Agentes Inteligentes

- **Agentes de reflejo simple:** Este tipo de agente no contiene internamente estados y sus procesos o acciones que realiza son respuestas a la entrada de percepciones, a esta conexión entre percepciones y acciones se las denomina reglas de condición-acción. Ejemplo: **Si** el carro de adelante está frenando **entonces** empiece a frenar.

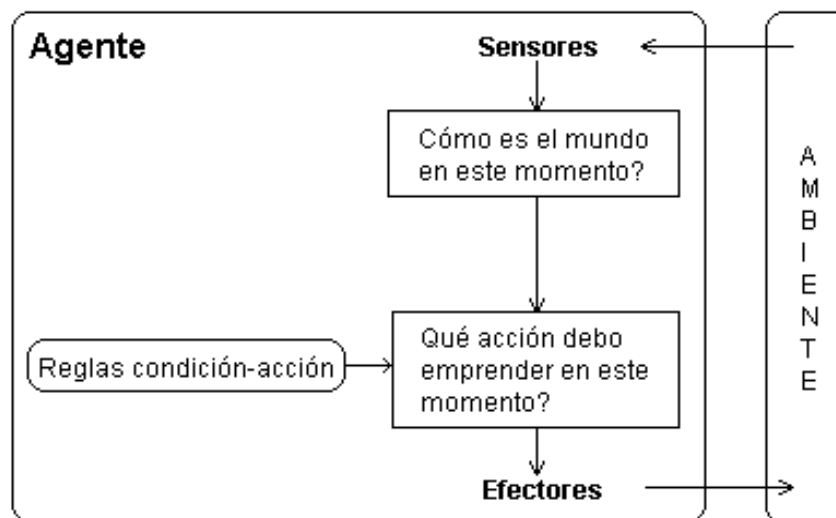
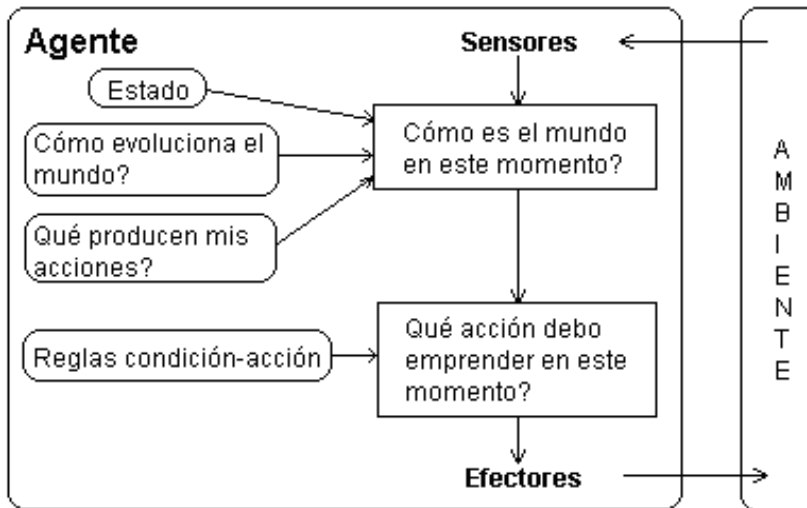


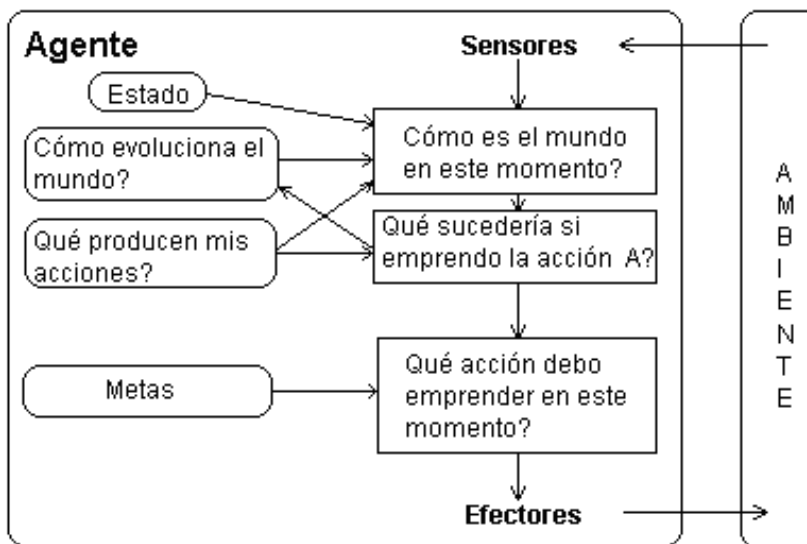
Diagrama esquematizado de un agente reflejo simple

Agentes bien informados de todo lo que pasa: Este tipo de agente guarda estados internos lo que nos sirve sin consideración para ejecutar una acción. Los sensores no nos pueden informar a la vez de todos los estados que maneja nuestro ambiente, es por este caso que el agente necesita actualizar algo de información en el estado interno. Esto le permite discernir que entre estados del ambiente que generan la misma entrada de percepciones pero, sin embargo; para cada uno de los estados se necesitan acciones distintas.



Un agente reflejo con un estado interno

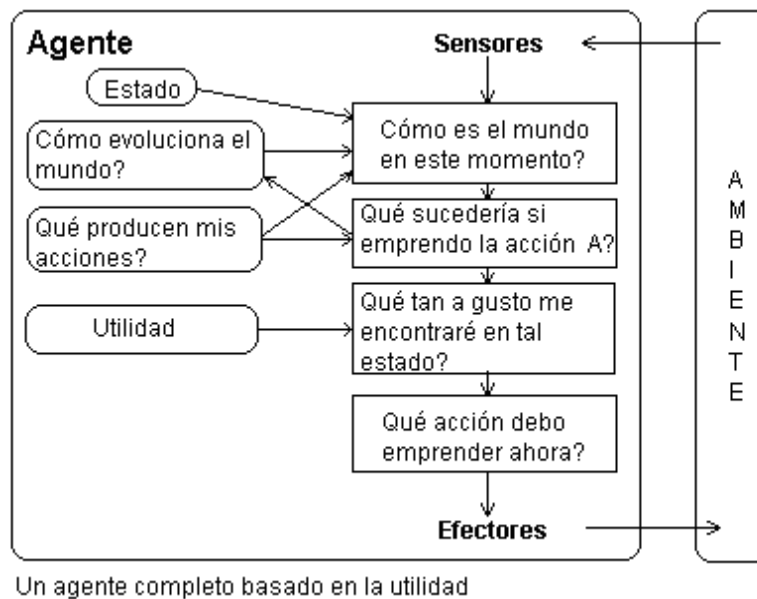
- **Agentes basados en metas:** Además de los estados, los agentes necesitan cierto tipo de información sobre sus metas. Estas metas van a detallar las situaciones a las que se desea llegar de este modo, el programa de agente puede combinar las metas con la información de los resultados (acciones) que emprenda y de esta manera poder elegir aquellas acciones que permitan alcanzar la meta.



Un agente con metas explícitas

Agentes basados en utilidad: Las metas por sí solas me garantizan la obtención de una conducta de alta calidad. En mi programa de agente se podría tener un conjunto de metas pero la obtención de éstas no me garantizan distinciones entre estados felices e infelices, mediante una medida de desempeño se podría establecer una comparación entre los

diversos estados del mundo (ambientes) para poder encontrar el estado de felicidad para el agente. Este estado ofrecerá una mayor utilidad al agente.



2.2. Definición de Ambientes

a) Representante virtual

Se comunican en lenguaje natural y suplen a los comerciales.

Charla con un representante virtual financiero

- En el de la **Coca Cola** (<http://questions.coca-cola.com>) pregunta: When do you pay dividends? o Can you mail me an annual report?
- Charla con la atractiva agente virtual de **Caja Madrid** (<http://www.cajamadrid.es/CajaMadrid/Home/puente?pagina=3447>) o **IKEA** (http://www.ikea.com/ms/es_ES/local_home/alcorconn.html)

- En mensajería instantánea (<http://www.error500.net/bots-mensajeria-instantanea>).

b) Como asistentes personales

Nos ayudan como lo haría un ayudante, una especie de "mayordomo virtual".

- Agenda inteligente. Si le digo que el sábado voy a la playa, me recuerda que tengo que llevar la toalla. Por ejemplo estos agentes del **MIT** (<http://agents.media.mit.edu/projects/tasks/calendar.jpg>) (<http://agents.media.mit.edu/projects/tasks/calendar.wmv>).
- Asesor financiero. En función de nuestra "manera de ser", por ejemplo aversos al riesgo nos va recomendando unos valores u otros, **MIT** (<http://agents.media.mit.edu/projects/investing>)
- Auditor. **REMS** (http://www.cica.ca/index.cfm/ci_id/1606/la_id/1.htm). Si hay un encargo nuevo, captura la información del cliente que hay en bases de datos financieras y nos presenta el cuestionario de auditoría más apropiado. Por ejemplo, si es un cliente que no exporta se eliminan las cuestiones relativas a exportación.

c) Negociador en mercados electrónicos

Localiza una subasta en Internet, aprende cómo va la puja y realiza la compra por nosotros. O mira en las tiendas más baratas.

- Ejemplos en (http://www.botspot.com/BOTSPOT/Windows/Shopping_Bots/Auction_Bots)

d) Agente de búsqueda de información o rastreador

Rastrear en las redes de computadoras en busca de la información solicitada. Son parametrizables por el usuario o aprenden de sus hábitos, rastrean la red e informan por correo electrónico de novedades que consideran pueden ser de interés para el usuario.

- **Maimai** (<http://maimai.com>) agiliza la búsqueda de los anuncios clasificados tras haber estudiado el comportamiento de sus visitantes durante varios meses. De esta forma, si un navegante está buscando un Audi A6 y en ese momento no hay ningún modelo disponible, el agente de inteligencia relaciona rápidamente este vehículo con otros de características similares, y así ofrece alternativas razonables. Por otra parte, este programa informático ofrece la posibilidad de avisar al interesado, mediante un sistema de alerta al correo electrónico o al teléfono móvil, cuando disponga de alguna oferta que hubiera sido solicitada previamente.
- **Googlealert** (<http://www.googlealert.com>)
- **Copernic** (<http://www.copernic.com/en/products/agent/download.html>) y (<http://www.copernic.com/en/products/agent/screenshots.html>) [[local versión vieja](#)]
- **Spypress** (<http://www.spypress.com>)
- **Tracerlock** (<http://www.tracerlock.com>)
- **Trademarkbots** (<http://www.trademarkbots.com>) ¿Qué dicen de mi y de mi empresa? buscan en 11 sitios (blogs, noticias, foros...).

- **Google News** (<http://news.google.com>). Podemos crear alertas (<http://www.google.com/newsalerts>), de forma que nos avisen cuando haya una noticia sobre la palabra solicitada. Podemnos confeccionar un periódico a medida.



Periódico a medida, con Google News

Contaba un ex-Presidente del Gobierno que una de las cosas que más le llamaba la atención cuando llegó a ser Presidente del Gobierno era que ya no podía leer los periódicos por la mañana, sino que un equipo de personas leían la prensa cada mañana y le recortaban las noticias de forma que tenía un "periódico a medida". Hoy en día, con Internet todos podemos conseguirlo.

Ejercicio: Cree en **Google News** (<http://news.google.com>) su periódico personal. También podemos crear alertas (<http://www.google.com/newsalerts>), de forma que nos avisen cuando haya una noticia sobre la palabra solicitada.

e) Agente secreto o espía

Monitorizan una página web identificada previamente por el usuario -una página con la información financiera de una empresa, una web oficial con normativa europea, la página con ofertas de empleo de una empresa, etc.- e informan cuando se producen cambios en dicha página.

- **Changedetection** (<http://www.changedetection.com>)
- **Zaptxt** (<http://zaptxt.com>)
- **Spyweb** (<http://www.spyweb.com/spyweb.php>)
- **Pubsub** (<http://www.pubsub.com/edgar.php>) It lets you "subscribe" to a particular search, after which the engine will continue to retrieve new information related to your search as it appears on blogs, in newsgroup discussions or in SEC filings, automatically refreshing your search results so they are ready (one click away from the home page) when you want them.
- **Accoona** (<http://accoona.com>). This engine uses artificial intelligence to "understand" the meaning of the keywords in your query in an effort to deliver more relevant results. Drop-down menus allow you to set some search parameters, such as business type and location; run a News search for recent articles, Business search for facts and figures, or a Web search for relevant links.



¿Cómo mantenerse informados?

Una vez que hemos encontrado un servidor interesante, (una página oficial con normativa europea, una página personal de alguien que va añadiendo recursos, la página con ofertas de empleo de una empresa, etc.) podemos estar interesados en recibir un aviso cada vez que se modifique la página. En este caso, lo más intuitivo es acceder periódicamente al servidor para comprobar si ha habido alguna novedad. Pero nos encontramos con el inconveniente de que, en ocasiones accedemos al servidor y no ha cambiado nada. Por el contrario otras veces comprobaremos con disgusto que ha habido una novedad y no nos hemos enterado a tiempo. Es mucho más sensato utilizar un agente inteligente como **Changedetection** (<http://www.changedetection.com>) que nos avisa cuando cambia una pagina, enviándonos un correo electrónico.

Hi!

At your request we have been monitoring the following page for changes:

<http://www.ciberconta.unizar.es/comercioelectronico/inicio.html>

A change was detected at:

Thu Mar 13 06:22:57 PST 2003

EJERCICIO: Configure un agente para que nos avise cuando haya una lección nueva en Ciberconta.

2.2.1. Propiedades

La mayoría de los agentes poseen las siguientes tres características: comunicación, inteligencia y autonomía.

- **Comunicación.** El agente puede comunicarse con el usuario, con otros agentes y con otros programas. Con el usuario se comunica con un interfaz amigable, mediante el que personaliza sus preferencias. Algunos agentes permiten comunicarse en lenguaje natural, algo típico de los chatbots.
- El grado de **inteligencia** varía mucho de unos agentes a otros, que suelen incorporar módulos con tecnologías procedentes de la Inteligencia Artificial. Los más sencillos se limitan a recoger las preferencias del usuario, quien debe personalizarlos. Un ejemplo son los agentes inteligentes basados en tecnología de redes neuronales especializados en identificar mensajes de correo electrónico sospechosos de contener **spam** -mensajes no deseados-. En una primera fase el

usuario debe marcarlos como spam, el agente va aprendiendo a identificar los rasgos que caracterizan a estos mensajes y posteriormente los filtra.

- **Autonomía.** Un agente no sólo debe ser capaz de hacer sugerencias al usuario sino de actuar. En el ejemplo anterior, el agente que filtra el spam no puede estar continuamente alertando al usuario en cada mensaje de correo que llega sobre la posibilidad de que sea un mensaje no deseado y su verdadera utilidad surge cuando elimina de forma autónoma dichos mensajes.

2.3. Definición de Sistemas Expertos

SISTEMAS EXPERTOS

Los sistemas expertos son la técnica de Inteligencia Artificial que primero se utilizó en los sistemas de análisis del riesgo de crédito. No obstante, muchos sistemas expertos están plenamente vigentes y en servicio en diversas instituciones, y sus posibilidades se han visto enriquecidas con los nuevos enfoques que han aportado otras técnicas de aparición más reciente, como los sistemas de inducción.

Es una forma de programación que emula artificialmente ciertos aspectos del razonamiento de un especialista humano en un ámbito restringido y limitado de conocimiento.

CONCEPTO Y TIPOS DE SISTEMAS EXPERTOS.

Los sistemas expertos son programas que capturan el conocimiento de un experto e imitan sus procesos de razonamiento al resolver los problemas de un determinado dominio. Al igual que las bases de datos contienen conocimiento, pero en las bases de datos ese conocimiento es únicamente declarativo (hechos).

En contraposición, los sistemas expertos incorporan experiencia, que consiste tanto en conocimiento de tipo declarativo, como conocimiento de tipo procedimental (pautas de actuación), lo que les permite emular los procesos de razonamiento de los expertos humanos.

Existen distintos tipos de sistemas expertos, según la forma de representar el conocimiento incluido en ellos. Los más comúnmente utilizados en el ámbito del análisis de la solvencia son los basados en reglas, siendo la estructura de cada una de las reglas incluidas en los mismos:

SI	premisa
ENTONCES	conclusión.

Sobre esta estructura básica se pueden buscar variantes combinando diferentes premisas mediante operadores lógicos (y/o).

Otro tipo de sistemas expertos de más reciente desarrollo son los sistemas basados en casos (*Case Based Reasoning Systems*). Éstos constituyen modelos de

razonamiento consistentes en resolver un problema determinado a través de analogías con situaciones pasadas, de tal manera que se buscará el caso almacenado en la base de conocimientos que más se parezca al problema a resolver y se adaptará la decisión tomada para ese caso a la situación actual.

En este tipo de sistemas se produce aprendizaje cuando nuevos casos son resueltos y pasan a formar parte de la base de conocimientos.

2.3.1. Características y propiedades

Su característica principal es que están basados en reglas. Es decir, contienen un juego predefinido de conocimientos que se utiliza para tomar sus decisiones. Los sistemas expertos pueden resolver con mucha facilidad ciertos problemas complejos (no abordables mediante cálculo puro) que sólo puede llevarse a cabo por un número limitado de personas expertas intensamente entrenadas.

Tal vez su característica más fascinante es su capacidad para enfrentar problemas del mundo real, por medio de la aplicación de procesos que emulan el juicio y la intuición humanas. Para ello, es necesario dotar a la computadora tanto de una “base de conocimientos” como de un “motor de inferencia”.

La base contiene el conocimiento del sistema con respecto a un tema específico para el que se diseña el sistema. El motor de inferencia, en cambio, es el que combina los hechos y las preguntas particulares, utilizando la base de conocimientos, seleccionando los datos y los pasos apropiados para presentar los resultados.

Posteriormente este núcleo central fue ampliado con la incorporación de otros muchos procesos cognitivos vinculados con el pensamiento: aprendizaje automático, razonamiento basado en casos, lógica difusa, etc.

Su campo de aplicación es muy amplio utilizándoselos, entre otras cosas, para diagnosticar enfermedades, planificar la síntesis de complejos compuestos de química orgánica, analizar circuitos electrónicos o resolver ecuaciones diferenciales en forma simbólica.

2.3.2. Objetivos y perspectivas

El campo del análisis de decisiones cuyo desarrollo se produjo entre la década de los años 50 y los 60, estudia la aplicación de la teoría de decisiones en los problemas de decisión reales.

Sirvió para la toma de decisiones en dominios importantes en donde están en juego importantes intereses, como el mundo empresarial, gobierno, campo legal, estrategia militar, diagnóstico médico y salud pública, diseño de ingeniería y administración de recursos.

El proceso implica un cuidadoso estudio de acciones y resultados posibles así como de las preferencias por cada resultado. Es frecuente que en el análisis de decisiones figuren los protagonistas: el tomador de decisiones, quien define resultados son posibles de obtener, y el analista de decisiones quién enumera las posibles decisiones, decide cuál es la acción idónea.

Hasta principios de la década de los ochenta, era muy limitado el empleo de la computadora en el análisis de decisiones; se consideraba que el principal objetivo del análisis era auxiliar a los humanos para tomar decisiones que auténticamente reflejarán sus preferencias.

En las primeras investigaciones sobre sistemas expertos, el énfasis fue puesto en responder preguntas, más que en tomar decisiones.

Tales sistemas, concentrados en la recomendación de acciones, en vez de ofrecer opiniones sobre determinados hechos, generalmente lo hacían basándose en reglas condición-acción, en vez de recurrir a representaciones explícitas de resultados y preferencias.

Cuando finalmente aparecieron las redes de creencias fue posible construir sistemas a gran escala mediante los que fue posible obtener inferencias probabilísticas confiables a partir de evidencias.

La incorporación de las redes de decisión permitió el diseño de sistemas expertos de manera que éstos fuesen capaces de recomendar decisiones óptimas, que tomaran en cuenta las preferencias del usuario y que reflejaran también la evidencia disponible.

Son muchas las ventajas adicionales obtenidas con la inclusión de modelos y cálculos de utilidad explícita en el marco de los sistemas expertos. El experto aprovecha la posibilidad de representar de manera explícita sus preferencias (o las de su cliente), el sistema realiza automáticamente el procedimiento de selección de acciones así como la deducción de conclusiones a partir de la evidencia.

Un sistema que cuente con utilidad, permitirá evitar uno de los errores más comunes relacionados con el procedimiento de consulta: posibilidad de que se produzcan confusiones de importancia respectiva.

Por ejemplo, en los primeros sistemas expertos para medicina una estrategia muy utilizada consistía en dar prioridad a los diagnósticos por orden de posibilidad, y ofrecer como respuesta aquellos que tenían más posibilidades.

Desafortunadamente, el método anterior podría resultar desastroso al pasar por alto casos de enfermedades relativamente raras, aunque tratables, cuyos síntomas fácilmente podrían confundirse con los de enfermedades más comunes. Un ejemplo clásico es cuando se confunde la enfermedad de Hodgkin (una forma de cáncer) con la mononucleosis (infección viral leve y muy común).

(En la consulta, el diagnóstico más probable para la mayoría de los pacientes es “no tiene nada”.)

Desde luego, un plan de pruebas de laboratorio o de tratamiento deberá construirse con base tanto en probabilidades; como utilidades.

Resumiendo, contar con información sobre la utilidad es un apoyo para la ingeniería del conocimiento y procedimiento de consulta, como se explica a continuación.

2.3.3. Aplicaciones

REDES NEURONALES ARTIFICIALES

Inspiradas en el funcionamiento de las neuronas biológicas, están compuestas por una multitud de procesadores paralelos interconectados, cada uno de los cuales es capaz de efectuar sólo un pequeño número de operaciones simples y transmitir sus resultados a sus vecinos.

A las redes neuronales no se les “inculca” ningún tipo de regla, sino que son capaces de aprender a reconocer patrones (como las imágenes digitalizadas o los comportamientos monetarios en el mercado de cambio), a partir de un proceso de entrenamiento basado en el análisis automático y sistemático de una suficiente cantidad de ejemplos diferentes.

Desde este punto de vista, simulan con bastante aproximación algunos procesos perceptuales humanos, como, por ejemplo, la identificación y el aprendizaje de símbolos (letras, números, íconos) o de imágenes (rostros, mapas, configuraciones), aunque todavía la tecnología se encuentra lejos de alcanzar la eficacia de los mecanismos perceptivos naturales.

La habilidad de manipular datos imprecisos, incompletos, con ruidos y hasta compuestos de ejemplos contradictorios, hacen que las redes neuronales sean extremadamente eficaces en problemas de clasificación, reconocimiento de patrones, modelación y previsión de sistemas complejos, así como en tareas donde los especialistas no están a disposición o no puede formularse fácilmente un conjunto de reglas claras.

Sin embargo, no son apropiadas para las aplicaciones en donde es fundamental la explicación del razonamiento llevado a cabo.

LÓGICA DIFUSA

Proporciona un conjunto de técnicas matemáticas para que las computadoras operen con datos que no tienen una precisión definida y concreta.

Se trata de una generalización de la lógica tradicional que no utiliza sólo dos variables antagónicas como “negativo” o “positivo”, sino que puede manipular valores intermedios como “muy negativo, medio negativo, poco negativo, aproximadamente cero, poco positivo, medio positivo, muy positivo”. Se trata, en definitiva, de una forma de razonamiento que permite incorporar a la máquina esquemas de pensamientos típicamente humanos.

Los sistemas basados en lógica difusa han mostrado gran utilidad en una variedad de operaciones de control industrial y en tareas de reconocimiento de patrones que se extienden desde el reconocimiento de texto manuscrito, hasta a validación de crédito financiero.

Una de las principales potencialidades de esta técnica, cuando se la compara con las redes neuronales, es que sus bases de conocimiento -que se representan en forma de reglas- son fáciles de examinar y entender, lo cual también simplifica su mantenimiento y actualización.

Por este motivo, existe un creciente interés en utilizar la lógica difusa en conjunto con los sistemas expertos, con el fin de tornarlos más flexibles.

ALGORITMOS GENÉTICOS

Son algoritmos matemáticos eficientes que se inspiran en los mecanismos de la evolución biológica propuestos por Darwin. Han producido excelentes soluciones en problemas complejos de optimización que poseen un gran número de parámetros.

Las áreas en donde se han aplicado estos algoritmos incluyen: planeamiento industrial, layout de circuitos de muy alta escala de integración, control y optimización en la distribución de energía, optimización de ruteamiento y tráfico de datos, etc.

El objetivo principal de un algoritmo genético es el de evolucionar a partir de una población inicial de soluciones para un determinado problema, intentando producir nuevas generaciones de soluciones que sean mejores que las anteriores.

Esos algoritmos operan a través de un ciclo simple que consiste en la generación al azar de la población inicial y la evaluación, selección y reproducción de las más aptas (con respecto a un objetivo predefinido).

Este último proceso incorpora la recombinación y mutación del material “genético” de las soluciones. El ciclo se repite hasta llegar a una solución aceptable o al determinarse el óptimo de una función.

Tratamiento del lenguaje natural: sería muy conveniente para el ser humano poder decirle a una máquina -en lenguaje natural- lo que tiene que hacer y recibir de ésta un comentario hablado sobre la factibilidad de la orden.

Una de las principales tareas de la IA es potenciar a la computadora para que sea capaz de identificar la voz humana en general y de responder en forma hablada, pero -desgraciadamente y por el momento- ha demostrado ser más difícil de lo que originalmente se pensaba.

Esto se debe a que interviene una gran cantidad de conocimiento: además de tener en cuenta el significado de cada palabra de la oración, la estructura de la frase y el contexto en que se pronuncia, hay que disponer del conocimiento específico de la materia sobre la cual se habla así como también considerar los cambios de pronunciación y de entonación, las pausas e, incluso, el ruido ambiental.

Hoy en día existen algunos programas que pueden comprender la voz de una persona (aunque no la de varias al mismo tiempo), otros que traducen de un idioma humano a otro (pero no son tan buenos como los traductores humanos) y otros capaces de tomar dictados (pero no entienden lo que escriben).

Sin embargo, estos sistemas están bastante limitados: se debe hablar despacio y pronunciar bien las palabras. Asimismo actualmente hay computadoras que pueden sintetizar voces en varios idiomas, con diferentes entonaciones fonéticas (hombre, mujer, niño, viejo) y en diferentes estados psicológicos (irritación, ansiedad, miedo), así como sistemas que -luego de “leer” un texto- son capaces de resumirlo o responder a diferentes preguntas, extrayendo inferencias acerca del argumento, los personajes y las motivaciones.

No obstante, estos sistemas se descarrilan fácilmente por modismos, metáforas, o expresiones no gramaticales.

Su aplicación principal se encuentra en el control de sistemas informáticos o automáticos: teléfonos que “oyen” y marcan el número que se le indica, microscopios y videocámaras controlados por voz, computadoras que reconocen el dictado, etc.

Visión artificial

Puede interpretarse como una casa, pero también como un vehículo. Es un conocimiento más profundo del campo de aplicación el cual nos permite inferir que un pequeño rectángulo situado en medio de una superficie monótona tiene muchas probabilidades de ser un establecimiento rural o en otras palabras,

Trata de traducir el mundo visual (que las personas interpretan como real) un código binario de forma tal que facilite a un sistema robótico interactuar con su entorno.

Este procesamiento se desarrolla en diferentes niveles: digitalización de la imagen, detección de contornos, reconocimiento de objetos y técnicas de interpretación.

A pesar de tener resuelto el problema de imitar funcionalmente al ojo humano (con la invención de la cámara fotográfica y, posteriormente, de la cámara de video), la interpretación de una imagen dista de ser un ejercicio trivial.

En efecto, la tarea de transformar una imagen bidimensional en un objeto tridimensional identificado es uno de los retos más desafiantes que se hayan encarado.

A fin de lograr una “comprensión” real de una escena y tener en cuenta el contexto, es necesario disponer en los sistemas computacionales -al igual que en el procesamiento del lenguaje- de un conjunto de conocimientos previos afín al campo de aplicación que se encara.

La ventaja de estos sistemas reside en su habilidad para realizar tareas repetitivas y tediosas con alto grado de eficiencia; características que los capacitan para la detección de fallas, análisis microscópico, control de calidad, selección de componentes, etc.

Así, por ejemplo, los sistemas actuales reconocen objetos artificialmente diseñados (piezas mecánicas, componentes electrónicos o planos de circuitos electrónicos) e, incluso, son capaces de distinguir los colores de pequeñísimas piezas iguales que se desplazan a velocidades imposibles de percibir por el ojo humano, así como medirlas con una precisión increíble o clasificarlas por color. Sin embargo, tienen dificultades cuando “ven” rostros, árboles, animales o paisajes.

ROBÓTICA

Estudia los mecanismos de control que permiten a un ente mecánico moverse en un medio físico y/o manipular elementos físicos con cierto grado de autonomía. No obstante, para que un sistema alcance una verdadera autonomía en su funcionamiento debe ser capaz de identificar los cambios del entorno y tomar sus propias decisiones, modificando por sí mismo su programación en función de los objetivos de la tarea que ha de realizar.

Aquí es muy importante, no sólo incorporar a la máquina sentidos del tipo humano (vista, tacto, oído, olfato y la relación de esa información) para que comprendan su entorno, sino extraer de los datos sólo la información pertinente, para lo cual necesita disponer -como en el caso del habla y la visión- de “modelos mentales”.

INTELIGENCIA ARTIFICIAL.

En ningún otro campo es más visible la tecnología de la IA que en el campo de la robótica: la percepción (visual, auditiva o táctil), el reconocimiento de patrones, la ingeniería del conocimiento, la toma de decisiones, la comprensión del lenguaje natural, el habla, el aprendizaje, el planeamiento...; todo se une en los robots.

Quizás por eso, muchos autores no consideran a la robótica como parte de la IA, sino a ésta como un ingrediente de aquélla.

En el futuro, los robots serán más eficientes, fuertes, durables y rápidos que los humanos. Pero su función técnica no será reemplazar al ser humano, sino relevarlo y aliviarlo de los trabajos arduos y pesados, de las tareas repetitivas y de las obligaciones insalubres, peligrosas y desagradables.

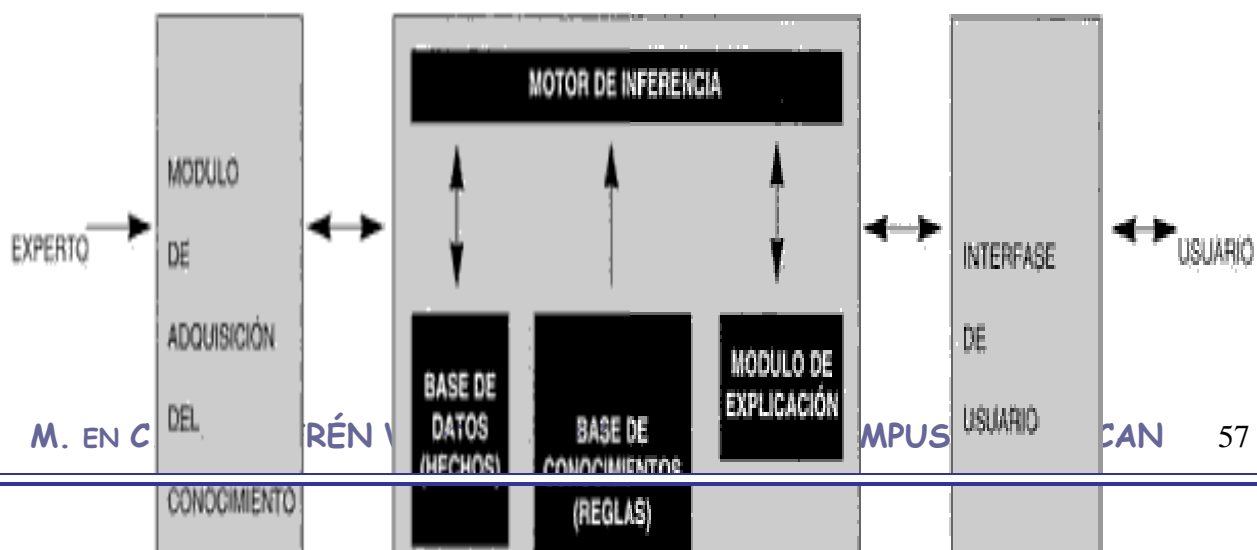
Es que cualquier cosa que pueda hacer un robot es impropia para una persona; en otras palabras, si un ser humano hace el trabajo que puede hacer un robot, se está desaprovechando todas sus capacidades y habilidades.

Actualmente se encuentran en funcionamiento sistemas que se usan ampliamente en plantas de montaje, pero sólo son capaces de tareas muy limitadas. Los robots tienen grandes dificultades para identificar objetos basados en la apariencia o en el tacto, y todavía se mueven y manipulan objetos torpemente.

Características generales de un sistema experto.

- Alto desempeño
- Tiempo de respuesta adecuado
- Confiabilidad
- Comprensible

Donde la base del conocimiento es fundamental pues es la que contiene el conocimiento proporcionado por el experto humano en el campo del conocimiento hacia el cual se orienta el sistema y que lo mismo puede ser un medico virtual o un sacerdote virtual.



COMPONENTES DE UN SISTEMA EXPERTO

Una característica decisiva de los Sistemas Expertos es la separación entre conocimiento (reglas, hechos) por un lado y su procesamiento por el otro.

A ello se añade una interface de usuario y un componente explicativo.

A continuación se muestra una breve descripción de cada uno de los componentes, pudiendo seleccionar cada uno de ellos individualmente para verlos en profundidad.

Componentes:

- 1- La Base de Conocimientos de un Sistema Experto contiene el conocimiento de los hechos y de las experiencias de los expertos en un dominio determinado.
- 2- El Mecanismo de Inferencia de un Sistema Experto puede simular la estrategia de solución de un experto.
- 3- El Componente Explicativo explica al usuario la estrategia de solución encontrada y el porqué de las decisiones tomadas.
- 4- La Interface de Usuario sirve para que éste pueda realizar una consulta en un lenguaje lo más natural posible.
- 5- El Componente de Adquisición ofrece ayuda a la estructuración e implementación del conocimiento en la base de conocimientos.

Los sistemas expertos que cuentan con información relacionada con la utilidad están en ventaja respecto de los sistemas de inferencia puros. Además de poder tomar decisiones, también pueden tomar la decisión de allegarse información dependiendo del valor que ésta represente y pueden calcular la sensibilidad de sus decisiones respecto a ligeras variaciones en la probabilidad y evaluaciones de utilidad.

La naturaleza de la resolución de problemas durante la década de la investigación en IA residía en un mecanismo de búsqueda de propósito general en el que se entrelazaban pasos de razonamiento elementales para encontrar así soluciones completas.

A estos procedimientos se les a denominado métodos débiles debido a que la información sobre el dominio que cuenta es débil. La única forma de remediarlo es

recorrer al uso del conocimiento adecuado a pasos de razonamiento más amplios, y resolver casos recurrentes en limitadas áreas de la experiencia.

METODOLOGÍA PARA ELABORAR SISTEMAS EXPERTOS

El procedimiento de la ingeniería del conocimiento que se emplea para elaborar y utilizar sistemas expertos teóricos y de decisión es el siguiente:

➤ Definir el alcance del problema

Definir cuáles son las posibles acciones, resultados y evidencias que deberán tomarse en cuenta. Por lo general, el analista deberá consultar uno o varios expertos del dominio de interés con el fin de saber cuáles son los factores más importantes.

➤ Definir la topología

Una vez que se sabe que se sabe cuáles son los factores importantes, ahora hay que saber qué factores influyen sobre otros. Es particularmente importante saber que aspectos del estado de resultado son los que definen su utilidad.

➤ Asignación de probabilidades.

En las redes de decisión, las probabilidades condicionales reflejan no sólo las influencias causales que existen entre las variables aleatorias, sino también ,los efectos de dichas acciones.

➤ Asignación de Utilidades

Es frecuente utilizar las técnicas descritas anteriormente para calcular funciones de utilidad.

Existen programas de computación mediante los cuales obtenemos de manera automatizada las preferencias correspondientes a diversas loterías y se construye sus respectiva función de utilidad.

El poder identificar la estructura de preferencia de funciones de utilidad de atributos múltiples es también determinante para reducir las dimensiones del problema de cálculo.

Permite así mismo, disminuir de manera exponencial las preguntas que es necesario formular.

➤ Introducir toda la evidencia de que se disponga

Por cada caso concreto en el que sea utilizado el sistema, puede haber alguna evidencia inicial.

➤ **Evaluar el programa**

Calcular la acción óptima con base en la evidencia existente.

➤ **Obtener otras evidencias más**

Calcular el valor de la información, comparándola con los costos de su obtención y realizar las observaciones que correspondan, si así es el caso.

Hay que tener presente que los sistemas expertos meramente inferenciales, pero que no disponen de utilidades, no están en posibilidad de saber que nuevas tendencias deberán obtener.

➤ **Efectuar un análisis de sensibilidad.**

Este es un paso muy importante para saber si la decisión considerada como la mejor es sensible a ligeras variaciones de las probabilidades y utilidades asignadas; para ello se varían de manera sistemática estos parámetros y se efectúa la correspondiente evaluación.

Si una ligera variación produce una gran variación en las decisiones, quizás valga la pena dedicar más recursos para obtener mejores datos. Si todas estas variaciones producen la misma decisión, el usuario podrá estar más seguro de que ésta es la correcta.

El análisis de la sensibilidad resulta especialmente importante ya que uno de los principales argumentos en contra de la aplicación de los métodos probabilistas en los sistemas expertos es que es muy difícil evaluar la cantidad de probabilidades numéricas necesarias.

Gracias a un análisis de sensibilidad es posible saber que muchas de las cantidades sólo necesitan de una ligera aproximación, por ejemplo, 0.2 del valor que se obtendrá a través de un análisis exhaustivo.

En algunos de los sistemas es posible especificar las probabilidades como si fueran rangos, mediante los cuales obtenemos el rango de utilidad de las acciones.

Si el rango de una acción domina sobre los rangos de los demás, no hay necesidad de otra evaluación de probabilidad.

Por ejemplo, podría darse el caso de que la ubicación óptima de un aeropuerto sea insensible al tráfico aéreo previsto a lo largo de un amplio rango de valores, con base en las creencias del sistema en cuanto a otros factores relevantes, por lo que

el usuario deberá seguir tranquilo por su falta de experiencia en la predicción del tráfico.

Los sistemas expertos siguen una filosofía diferente a los programas clásicos. Esto queda reflejado en la tabla que se presenta, la cuál resume las diferencias entre ambos tipos de procesamiento.

SISTEMA CLÁSICO	SISTEMA EXPERTO
Conocimiento y procesamiento combinados en un programa	Base de conocimiento separada del mecanismo de procesamiento
No contiene errores	Puede contener errores
No da explicaciones, los datos sólo se usan o escriben	Una parte del sistema experto la forma el módulo de explicación
Los cambios son tediosos	Los cambios en las reglas son fáciles
El sistema sólo opera completo	El sistema puede funcionar con pocas reglas
Se ejecuta paso a paso	La ejecución usa heurísticas y lógica
Necesita información completa para operar	Puede operar con información incompleta
Representa y usa datos	Representa y usa conocimiento

Comparación entre un sistema clásico de procesamiento y un sistema experto

OBJETIVOS.

Los sistemas expertos son programas que reproducen el proceso intelectual de un experto humano en un campo particular, pudiendo mejorar su productividad, ahorrar tiempo y dinero, conservar sus valiosos conocimientos y difundirlos más fácilmente.

Antes de la aparición del computador, el hombre ya se preguntaba si se le arrebataría el privilegio de razonar y pensar. En la actualidad existe un campo dentro de la inteligencia artificial al que se le atribuye esa facultad: el de los sistemas expertos.

Estos sistemas permiten la creación de máquinas que razonan como el hombre, restringiéndose a un espacio de conocimientos limitado. En teoría pueden razonar siguiendo los pasos que seguiría un experto humano (médico, analista, empresario, etc.) para resolver un problema concreto. Este tipo de modelos de conocimiento por computador ofrece un extenso campo de posibilidades en resolución de problemas y en aprendizaje. Su uso se extenderá ampliamente en el futuro, debido a su importante impacto sobre los negocios y la industria.

El objetivo de este artículo es enseñar de forma práctica el funcionamiento de un sistema experto.

La función de un Sistema Experto es la de aportar soluciones a problemas , como si de humanos se tratara, es decir capaz de mostrar soluciones inteligentes. Y os preguntareis ¿Como es posible? .

Es posible gracias a que al sistema lo crean con expertos (humanos), que intentan estructurar y formalizar conocimientos poniéndolos a disposición del sistema, para que este pueda resolver una función dentro del ámbito del problema, de igual forma que lo hubiera hecho un experto.

Acceder a los conocimientos adquiridos por experiencia es lo más difícil, ya que los expertos, al igual que otras personas, apenas los reconocen como tales. Son buscados con mucho esfuerzo y cuidado siendo descubiertos de uno en uno, poco a poco.

USOS DE UN SISTEMA EXPERTO

Un sistema experto es muy eficaz cuando tiene que analizar una gran cantidad de información, interpretándola y proporcionando una recomendación a partir de la misma.

Un ejemplo es el análisis financiero, donde se estudian las oportunidades de inversión, dependiendo de los datos financieros de un cliente y de sus propósitos.

Para detectar y reparar fallos en equipos electrónicos, se utilizan los sistemas expertos de diagnóstico y depuración, que formulan listas de preguntas con las que obtienen los datos necesarios para llegar a una conclusión.

Entonces recomiendan las acciones adecuadas para corregir los problemas descubiertos. Este tipo de sistemas se utilizan también en medicina (ej. MYCIN y PUFF), y para localizar problemas en sistemas informáticos grandes y complejos.

Los sistemas expertos son buenos para predecir resultados futuros a partir del conocimiento que tienen. Los sistemas meteorológicos y de inversión en bolsa son ejemplos de utilización en este sentido. El sistema PROSPECTOR es de este tipo.

La planificación es la secuencia de acciones necesaria para lograr una meta. Conseguir una buena planificación a largo plazo es muy difícil. Por ello, se usan sistemas expertos para gestionar proyectos de desarrollo, planes de producción de fábricas, estrategia militar y configuración de complejos sistemas informáticos, entre otros.

Cuando se necesita controlar un proceso tomando decisiones como respuesta a su estado y no existe una solución algorítmica adecuada, es necesario usar un sistema experto.

Este campo comprende el supervisar fábricas automatizadas, factorías químicas o centrales nucleares. Estos sistemas son extraordinariamente críticos porque normalmente tienen que trabajar a tiempo real.

El diseño requiere una enorme cantidad de conocimientos debido a que hay que tener en cuenta muchas especificaciones y restricciones. En este caso, el sistema experto ayuda al diseñador a completar el diseño de forma competente y dentro de los límites de costes y de tiempo. Se diseñan circuitos electrónicos, circuitos integrados, tarjetas de circuito impreso, estructuras arquitectónicas, coches, piezas mecánicas, etc.

Por último, un sistema experto puede evaluar el nivel de conocimientos y comprensión de un estudiante, y ajustar el proceso de aprendizaje de acuerdo con sus necesidades.

El acceso al conocimiento y al juicio de un experto es extremadamente valioso en muchas ocasiones (prospecciones petrolíferas, manejo de valores bursátiles, diagnóstico de enfermedades, etc.), sin embargo, en la mayoría de los campos de actividad existen más problemas por resolver que expertos para resolverlos.

Para solucionar este desequilibrio es necesario utilizar un sistema experto.

En general, actuará como ayudante para los expertos humanos y como consultor cuando no se tiene otro acceso a la experiencia.

INTELIGENCIA ARTIFICIAL.

Un sistema experto, además, mejora la productividad al resolver y decidir los problemas más rápidamente. Esto permite ahorrar tiempo y dinero. A veces sin esa rapidez las soluciones obtenidas serían inútiles.

Los valiosos conocimientos de un especialista se guardan y se difunden, de forma que, no se pierden aunque desaparezca el especialista. En los sistemas expertos se guarda la esencia de los problemas que se intenta resolver y se programa cómo aplicar los conocimientos para su resolución.

Dicho de otra forma, simplemente ayudan a entender cómo se aplican los conocimientos para resolver un problema, si lo vemos de esta forma, esto es útil porque normalmente el especialista da por ciertos sus conocimientos y no analiza cómo los aplica.

Se pueden utilizar personas no especializadas para resolver problemas. Además si una persona utiliza regularmente un sistema experto aprenderá de él, y se aproximará a la capacidad del especialista.

Con un sistema experto se obtienen soluciones más fiables gracias al tratamiento automático de los datos, y más contrastadas, debido a que se suele tener informatizado el conocimiento de varios expertos.

Bajo el término de Sistemas Expertos se entiende un nuevo tipo de software que imita el comportamiento de un experto humano en la solución de un problema. Pueden almacenar conocimientos de expertos para un campo determinado y solucionar un problema mediante deducción lógica de conclusiones.

¿UN SISTEMA EXPERTO UTILIZA TÉCNICAS DE LA INTELIGENCIA ARTIFICIAL?

Los Sistemas Expertos son uno de los puntos que componen las investigaciones en el campo de la IA. Un sistema de computadoras que trabaje con técnicas de IA deberá estar en situación de combinar información de forma “inteligente”, alcanzar conclusiones y justificarlas (al igual que el resultado final).

Los Sistemas Expertos son una expresión de los sistemas basados en el conocimiento. Con la aplicación de técnicas de Inteligencia Artificial finaliza la transición del procesamiento de datos al procesamiento de conocimientos.

Los sistemas expertos se aplican por norma general en problemas que implican un procedimiento basado en el conocimiento.

Pues bien, nos preguntamos ¿que es eso?.

Un procedimiento de solución basado en el conocimiento comprende las siguientes capacidades:

INTELIGENCIA ARTIFICIAL.

- ✓ Utilización de normas o estructuras que contengan conocimientos y experiencias de expertos especializados.
- ✓ Deducción lógica de conclusiones.
- ✓ Capaz de interpretar datos ambiguos.
- ✓ Manipulación de conocimientos afectados por valores de probabilidad.

SU USO

- Los motivos mas importantes para utilizar a un Sistema Experto son:
 - ✓ Utilizar personal no especializado para resolver problemas que requieren especialidad
 - ✓ Obtención de soluciones mas rápidas
 - ✓ Obtención de soluciones mas fiables
 - ✓ Reducción de costos
 - ✓ Eliminación de operaciones incómodas o monótonas
 - ✓ Escasez de expertos humanos
 - ✓ Acceso del conocimiento a poblaciones mas amplias
- Además de las razones expuestas anteriormente, hay algunos casos en los que el uso de SE es recomendable por darse circunstancias especiales por ejemplo:
- Los expertos son difícilmente accesibles
- Los expertos son caros
- Se necesita una mejora del conocimiento
- El conocimiento es difícil de adquirir y se basa en reglas que solo la experiencia da a conocer
- Cuando la información disponible es pobre, parcial, incompleta o los problemas estan mal definidos
- Cuando hay un desconocimiento general entre los que necesitan el conocimiento
- Cuando el problema esta condicionado por normas legales que son muy cambiantes.

DESARROLLO DE LOS SISTEMAS EXPERTOS

1.-EL EQUIPO DE DESARROLLO

Las personas que componen un grupo o un equipo, como en todos los ámbitos deben cumplir unas características y cada uno de ellos dentro del equipo desarrolla un papel distinto. A continuación detallaremos cada componente del equipo dentro del desarrollo y cual es la función de cada uno :



1.1- EL EXPERTO

La función del experto es la de poner sus conocimientos especializados a disposición del Sistema Experto.



1.2- EL INGENIERO DEL CONOCIMIENTO

El ingeniero que plantea las preguntas al experto, estructura sus conocimientos y los implementa en la base de conocimientos.



1.3- EL USUARIO

El usuario aporta sus deseos y sus ideas, determinado especialmente el escenario en el que debe aplicarse el Sistema Experto.



En el desarrollo del Sistema Experto, el ingeniero del conocimiento y el experto trabajan muy unidos. El primer paso consiste en elaborar los problemas que deben ser resueltos por el sistema.

Precisamente en la primera fase de un proyecto es de vital importancia determinar correctamente el ámbito estrechamente delimitado de trabajo. Aquí se incluye ya el usuario posterior, o un representante del grupo de usuarios.

Para la aceptación y consecuencia para el éxito, es de vital y suma importancia tener en cuenta los deseos y las ideas del usuario.

Una vez delimitado el dominio, nos pondremos a "engrosar" nuestro sistema con los conocimientos del experto. El experto debe comprobar constantemente si su conocimiento ha sido transmitido de la forma más conveniente.

El ingeniero del conocimiento es responsable de una implementación correcta, pero no de la exactitud del conocimiento. La responsabilidad de esta exactitud recae en el experto.

A ser posible, el experto deberá tener comprensión para los problemas que depara el procesamiento de datos. Ello facilitará mucho el trabajo.

Además, no debe ignorarse nunca al usuario durante el desarrollo, para que al final se disponga de un sistema que le sea de máxima utilidad.

La estricta separación entre usuario, experto e ingeniero del conocimiento no deberá estar siempre presente. Pueden surgir situaciones en las que el experto puede ser también el usuario.

Cuando exista un tema muy complejo cuyas relaciones e interacciones deben ser determinadas una y otra vez con un gran consumo de tiempo. De esta forma el experto puede ahorrarse trabajos repetitivos.

La separación entre experto e ingeniero del conocimiento permanece, por regla general inalterada.

2.-MÉTODOS AUXILIARES EN EL DESARROLLO

La eficiencia en la creación de Sistemas Expertos puede aumentarse en gran medida con la aplicación de Shells.

Un Shell (de forma resumida) es un Sistema Experto que contiene una base de conocimientos vacía. Existen el mecanismo de inferencia, el componente explicativo y a veces también la interface de usuario.

Ya el mecanismo de inferencia depende del problema o grupos de problemas. No existe ningún Shell para todas las aplicaciones, sino que hay que buscar un Shell para cada aplicación.

También es posible que haya que desarrollar adicionalmente partes del mecanismo de inferencia. Según el tamaño de esta parte tendrá que pensar si la aplicación de un Shell determinado sigue siendo apropiada. Si el ingeniero del conocimiento conoce bien este Shell,

Si, por ejemplo, conoce exactamente cómo son procesadas las reglas, entonces sólo tendrá que concentrarse en la creación de la base de conocimientos. A menudo, el Shell contiene Frames.

Estos son marcos previamente preparados, en los que, por ejemplo, sólo se introduce en nombre del objeto, sus cualidades y los correspondientes valores.

Las relaciones entre los objetos se indican mediante señalización de los dos objetos y del tipo de relación que exista entre ellos. El trabajo de implementación debemos procurar, en la medida de lo posible, reducirlo al máximo.

Los Frames son componentes explicativos y/o el mecanismo de inferencia están dimensionados de forma distinta en los diferentes Shells.

3.-CAMPOS DE APLICACIÓN

La aplicación de Sistemas Expertos será adecuada allí donde los expertos dispongan de conocimientos complejos en un área muy delimitada, donde no existan algoritmos ya establecidos (o donde los existentes no puedan solucionar algunos problemas).

Otro campo de aplicación es allí donde encontremos teorías que resulten prácticamente imposibles de analizar todos los casos teóricamente imaginables mediante algoritmos y en un espacio de tiempo relativamente corto y razonable.

Debido a que la estructuración e implementación del conocimiento del experto requiere una gran cantidad de trabajo, sólo valdrá la pena realizar el esfuerzo de crear un Sistema Experto cuando un conocimiento sea válido durante un largo espacio de tiempo y vaya a ser utilizado por el mayor número de personas.

Los Sistemas Expertos ofrecen ayuda para:

- 1- Evitar fallos en labores rutinarias complejas
- 2- Ampliar de forma más rápida los conocimientos de los especialistas.
- 3- Diagnosticar los fallos con mayor rapidez y conseguir tareas de planificación más completas y consistentes.

En este sentido, el Sistema Experto supone una descarga del experto en el trabajo rutinario y, por lo tanto, la reducción de sus problemas. Cuando la labor del experto no está tan sobrecargada, se reducen las decisiones erróneas y se aceleran los procesos en la toma de decisiones.

Los Sistemas Expertos no deben considerarse como soluciones aisladas respecto a otros desarrollos de software. La aplicación del software convencional debe realizarse allí donde tenga sentido hacerlo.

Otro punto que afecta al sistema es el hardware disponible. En una situación en la que un sistema sea necesario en muchos puntos distintos, el Sistema Experto deberá poder funcionar, a ser posible, en el hardware allí disponible. En la actualidad hay aparatos de Inteligencia Artificial muy avanzados, pero que por cuestiones de coste (como siempre) no pueden ser aplicados en todos los puestos de trabajo de los usuarios.