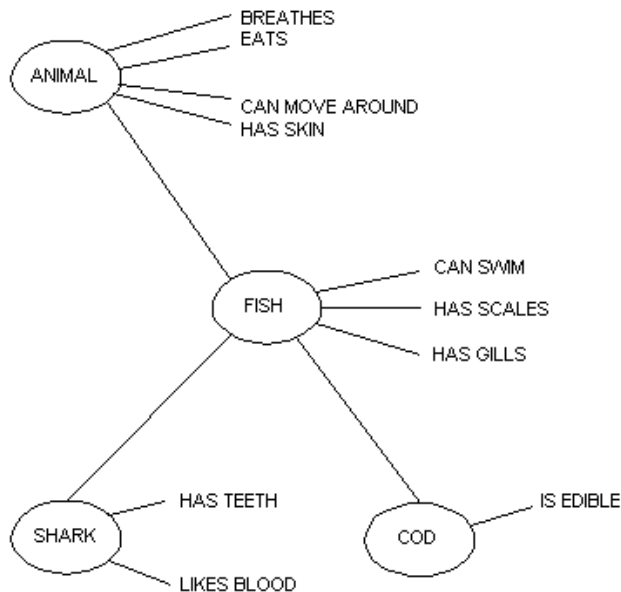


Tema 5. Comportamiento inteligente: Representación del Conocimiento e inferencia basados en lógica



Objetivos

- Adquirir las habilidades básicas para construir sistemas capaces de resolver problemas mediante técnicas de IA.
- Entender que la resolución de problemas en IA implica definir una representación del problema y un proceso de búsqueda de la solución.
- Comprender la necesidad de representar el conocimiento y realizar inferencia para que un sistema pueda exhibir comportamiento inteligente.
- Conocer los fundamentos de la representación del conocimiento en lógica proposicional y de predicados y sus mecanismos de inferencia asociados.
- Aplicar los aspectos de representación basada en la lógica y mecanismos de inferencia, mediante técnicas y herramientas de programación lógica.

Estudia este tema en ...

- Nils J. Nilsson, “*Inteligencia Artificial: Una nueva síntesis*”, Ed. Mc Graw Hill, 2000. pp. 215-284

Contenido

- Representación del conocimiento en IA
- El cálculo proposicional
- Resolución en el cálculo proposicional
- Cálculo de predicados
- Resolución en el cálculo de predicados
- Introducción a los Sistemas Basados en el Conocimiento

Principio del Conocimiento

“Un sistema exhibe un comportamiento inteligente, debido principalmente al conocimiento que puede manejar: conceptos, hechos, representaciones, métodos, modelos, metáforas y heurísticas en su dominio de actuación”

Representación del conocimiento en IA

- Hemos estudiado varias formas de modelar el mundo de un agente, entre ellas:
 - **Representaciones icónicas:** Simulaciones del mundo que el agente podía percibir.
 - **Representaciones descriptivas:** Valores binarios que describían aspectos ciertos o falsos sobre el mundo.
- Las representaciones descriptivas tienen ciertas ventajas sobre las icónicas:
 - Son más sencillas.
 - Son fáciles de comunicar a otros agentes.
 - Se pueden descomponer en piezas más simples.

Representación del conocimiento en IA

- Además, hay información del entorno del agente que no se puede representar mediante modelos icónicos, tales como:
 - **Leyes generales.** “Todas las cajas azules pueden ser cogidas”.
 - **Información negativa.** “El bloque A no está en el suelo”, sin decir dónde está el bloque A.
 - **Información incierta.** “O bien el bloque A está sobre el bloque C, o bien el bloque A está sobre el bloque B”.
- Sin embargo, este tipo de información es fácil de formular como conjunto de restricciones sobre los valores de las características binarias del agente.
- Estas restricciones representan **conocimiento sobre el mundo.**

Representación del conocimiento en IA

- A menudo, este **conocimiento sobre el mundo** puede utilizarse para razonar sobre él y hallar nuevas características del mismo.

Ejemplo:

- El conocimiento que se tiene es “Todos los pájaros vuelan”; y “Piolín es un pájaro”.
- Se puede *razonar*, por tanto, que “Piolín vuela”.
- **Otro Ejemplo:** Un robot sólo puede levantar un bloque si tiene suficiente batería y el bloque es elevable. Entonces, el conocimiento sobre el mundo es: “Si el bloque es elevable y hay suficiente batería, entonces es posible levantar el bloque”.
- El robot “sabrá” si es capaz de levantar el bloque a partir de este **conocimiento** sobre su entorno.

Representación del conocimiento en IA

- Conocimiento: Representación simbólica de “aspectos” de un cierto dominio o “universo” de discurso.

Los “aspectos” pueden ser hechos, relaciones o procedimientos.

- Información: Cualquier estímulo capaz de alterar el estado o respuesta de un sistema.

Niveles de conocimiento

- Conocimiento de los elementos del dominio o conocimiento factual. Hechos del entorno. (Ej. un coche falla si se estropea el sistema de inyección)
- Heurísticas del entorno sobre el conocimiento, jerarquizan los conocimientos factuales. (Ej. las averías mecánicas son mas frecuentes que averías eléctricas)
- Metaconocimiento, conocimiento acerca del conocimiento pero independiente del mismo. (Ej: la aparición de una sola avería es mas frecuente que la aparición simultanea de dos).

Modelos de representación inspirados por la Psicología

1.- Modelos de conducta: Reglas de Producción o reglas IF__THEN

Si A entonces B (A: condición; B: Acción)

2.- Modelos de razonamiento: Lógica (s).

Lenguaje formal para representar conocimiento y formular inferencia.

3.- Modelos de memoria: Almacenaje y representación.

3.1 Modelos simples: Ficheros y Bases de datos.

3.2 Modelos de conocimiento heredable. Almacenamiento jerarquizado:

Redes Semánticas,

Frames, guiones,

Representaciones orientadas a objetos...

Representación del conocimiento en IA

- Estudiaremos 2 tipos básicos para representar el conocimiento y razonar sobre él:
 - Cálculo proposicional.
 - Cálculo de predicados.

Cálculo Proposicional

- Distinguiremos entre los siguientes elementos del lenguaje del cálculo proposicional:
 - **Átomos:** **V** (verdadero) y **F** (falso), las proposiciones (en mayúscula) **P, Q, P1, P2, SOBRE_A_B**, etc.
 - **Conectivas:** “y”, “o”, “implica”, “no”, representadas como:
 \wedge (**y**), \vee (**o**), \supset (**implica**), \neg (**no**)
 - **Sentencias (Fórmulas Bien Formadas, FBF):**
 - Cualquier átomo es una FBF.
 - Si A y B sobre FBF, entonces también lo son:

$$\mathbf{A \wedge B, A \vee B, A \supset B, \neg A}$$

Reglas de inferencia

- Las **reglas de inferencia** nos permiten producir nuevas FBFs a partir de las que ya existen, Algunas de las más comunes son:
 - Q puede inferirse a partir de P y $P \supset Q$ (*modus ponens*)
 - $P \wedge Q$ se puede inferir a través de la conjunción de P y Q
 - $Q \wedge P$ se puede inferir desde $P \wedge Q$ (*conmutatividad*)
 - P (también Q) se puede inferir desde $Q \wedge P$
 - $P \vee Q$ se puede inferir bien desde P , bien desde Q
 - P se puede inferir desde $\neg(\neg P)$

Resolución en el cálculo proposicional

- **Claúsulas como FBFs:**

- Un **Literal** es un átomo o su negación: Ejemplo: P , $\neg P$
- Una **cláusula** Σ es un conjunto de literales. Ejemplo:
 $\Sigma = \{P, Q, \neg R\}$
- Una cláusula es equivalente a la disyunción de sus elementos.
Por tanto, la cláusula Σ anterior equivale a: $P \vee Q \vee \neg R$

- **Idea general de la resolución:**

- Siendo λ un literal y Σ_1, Σ_2 dos cláusulas, a partir de $\{\lambda\} \cup \Sigma_1$ y $\{\neg \lambda\} \cup \Sigma_2$ se puede inferir $\Sigma_1 \cup \Sigma_2$.
- Ejemplo: Resolviendo $R \vee P, \neg P \vee Q$ se obtiene $R \vee Q$
- Denominaremos **Nil** a la cláusula vacía $\Sigma = \{\}$

El Cálculo de Predicados

- **El cálculo proposicional es limitado.** Supongamos nuestro mundo de bloques. Para decir que el bloque **A** está sobre el bloque **B**, deberíamos establecer una interpretación **SOBRE_A_B**.
- Para representar esta situación con todos los bloques usando cálculo proposicional, necesitaríamos tantos literales como posibilidades.
- Además, supongamos dos literales **P** y **Q**, con la semántica asociada **P \equiv SOBRE_A_B**, **Q \equiv SOBRE_B_C**.
- En lenguaje natural y mediante el conocimiento que tenemos del problema, nosotros (diseñadores, personas, etc.) *sabemos* que **A** está sobre **B**, y que **B** está sobre **C**. Por tanto, **C** está por debajo de **A**. Sin embargo, necesitaríamos más proposiciones y más complejas para implementar este conocimiento utilizando únicamente cálculo proposicional.

El Cálculo de Predicados

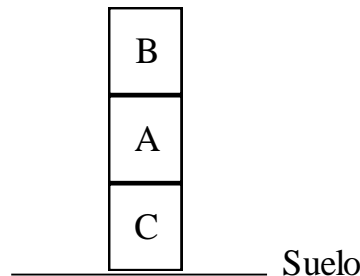
- Sería de gran utilidad un lenguaje que permitiese definir objetos y relaciones entre ellos.
- El **cálculo de predicados** nos permite esta opción y, además solventa los problemas planteados en la diapositiva anterior.
- Ejemplo: Para decir que $SOBRE_B_C \supset \neg LIBRE_C$, para cualquier bloque, el cálculo de predicados nos evita tener que reescribir todas las proposiciones del cálculo proposicional de las situaciones que pueden darse. Podemos abstraer los objetos a variables y escribir:

$$SOBRE(x, y) \supset \neg LIBRE(y)$$

- El significado sería “*cuando un objeto x esté sobre otro y , entonces y no estará libre*”.

Modelos e interpretaciones

- **Ejemplo:** En el mundo de los bloques, definimos **A, B, C, Suelo**, y les asignamos su interpretación con los bloques reales y el suelo. Definimos también las relaciones **Sobre(x,y)** y **Libre(x)**, indicando la primera que **x** está sobre **y**, y la segunda que **x** está libre (no tiene nada encima).
- Supongamos que la relación **Sobre** se da para: **Sobre(B, A)**, **Sobre(A, C)**, **Sobre(C, Suelo)**, y que **Libre** se da para **Libre(B)**. Estas relaciones definen el estado del mundo actualmente representada como sigue:



Resolución para demostrar teoremas

- Se hace de forma análoga como lo hacíamos en el cálculo proposicional.
- **Ejemplo:** La Base de Conocimiento de un agente contiene los siguientes elementos:

$$(\forall x, y)[\{Paquete(x) \wedge Paquete(y) \wedge \\ \wedge EnHabitacion(x, H27) \wedge EnHabitacion(y, H28)\} \supset MasPequeño(x, y)]$$

Paquete(A)

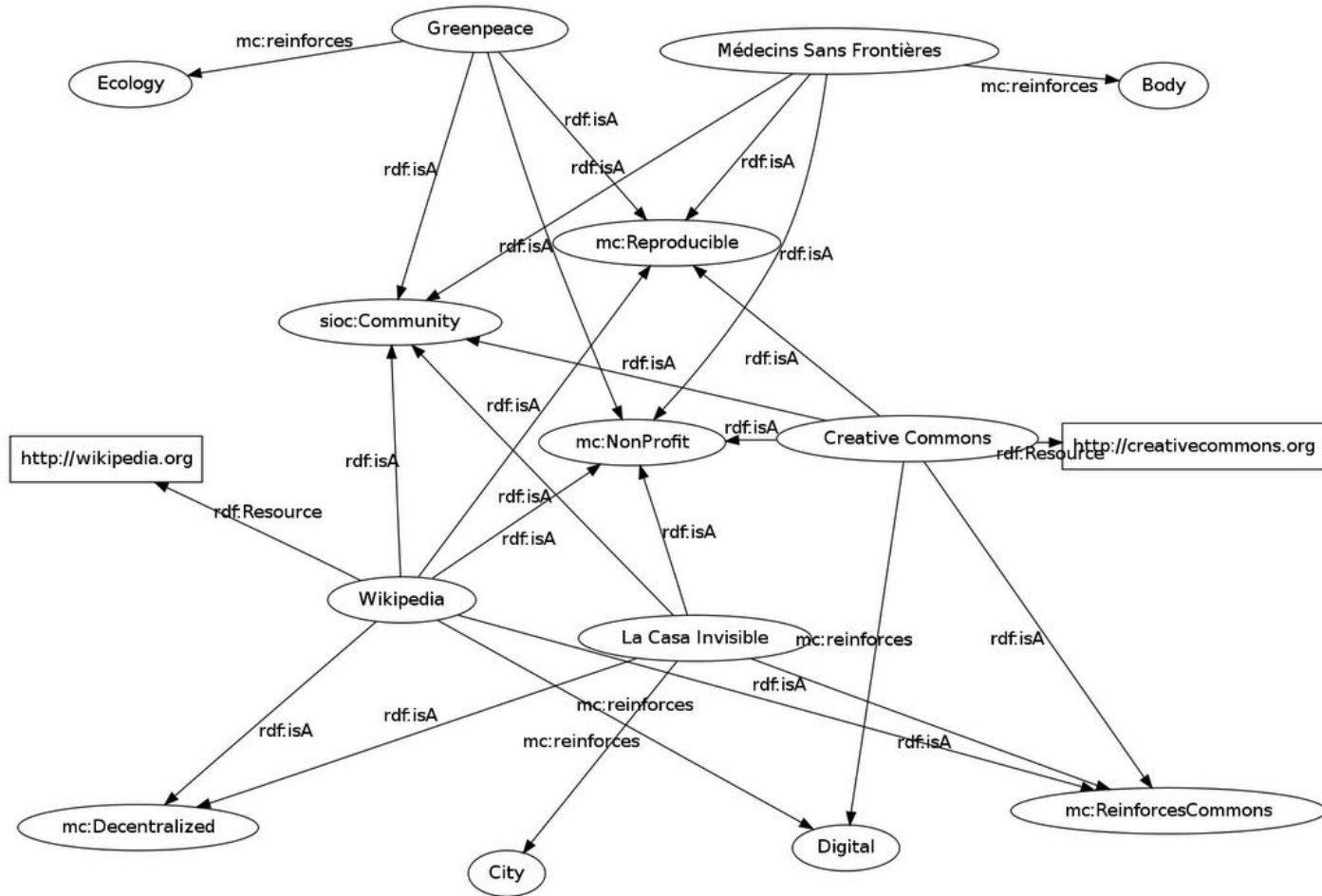
Paquete(B)

EnHabitacion(B, H27)

MasPequeño(B, A)

- **Pregunta:** ¿Dónde está el paquete A?

Red semántica



Sistemas Basados en el Conocimiento

- Una gran cantidad de aplicaciones reales de la IA se basan en la existencia de una gran masa de conocimiento:
 - Diagnóstico médico.
 - Diseño de equipos.
 - Sistemas de Recomendación.
 - Etc.
- Este tipo de sistemas se denominan **Sistemas Basados en el Conocimiento**, ya que este ocupa la parte central de la solución al problema a resolver.

Sistemas Basados en el Conocimiento

- Un **Sistema Basado en el Conocimiento (SBC)** necesita 3 componentes básicas:
 - Una **Base de Conocimiento (BC)**, que contenga el conocimiento experto necesario sobre el problema a resolver. Puede ser:
 - **Estática**, si la BC no varía a lo largo del tiempo.
 - **Dinámica**, cuando se añaden nuevos hechos o reglas, o se modifican las existentes a lo largo del tiempo.
 - Un **Motor de Inferencia**, que permite razonar sobre el conocimiento de la BC y los datos proporcionados por un usuario.
 - Una **interfaz de usuario** para entrada/salida de datos.
-

Tareas en la construcción de un SBC

- La adquisición del conocimiento y como representarlo de una forma abstracta efectiva.
- La representación del conocimiento en términos de un formalismo que una máquina pueda procesar.
- La realización de inferencias o como hacer uso de esas estructuras abstractas para generar información útil en cada caso específico.

Principales formalismos de representación

- lógica
- reglas de producción
- redes asociativas (semánticas)
- marcos

Cada formalismo de representación usa un método de inferencia específico:

- Resolución en Lógica
- Razonamiento hacia a delante y hacia atrás en sistemas de reglas
- Herencia en sistemas de frames

Sistemas Basados en el Conocimiento

- Ejemplos de **SBC** que incorporan conocimiento específico sobre algún tema:
 - **Anna** (Agente conversacional de IKEA). Su Base de Conocimiento está descrita en lenguaje AIML (*Artificial Intelligence Markup Language*).



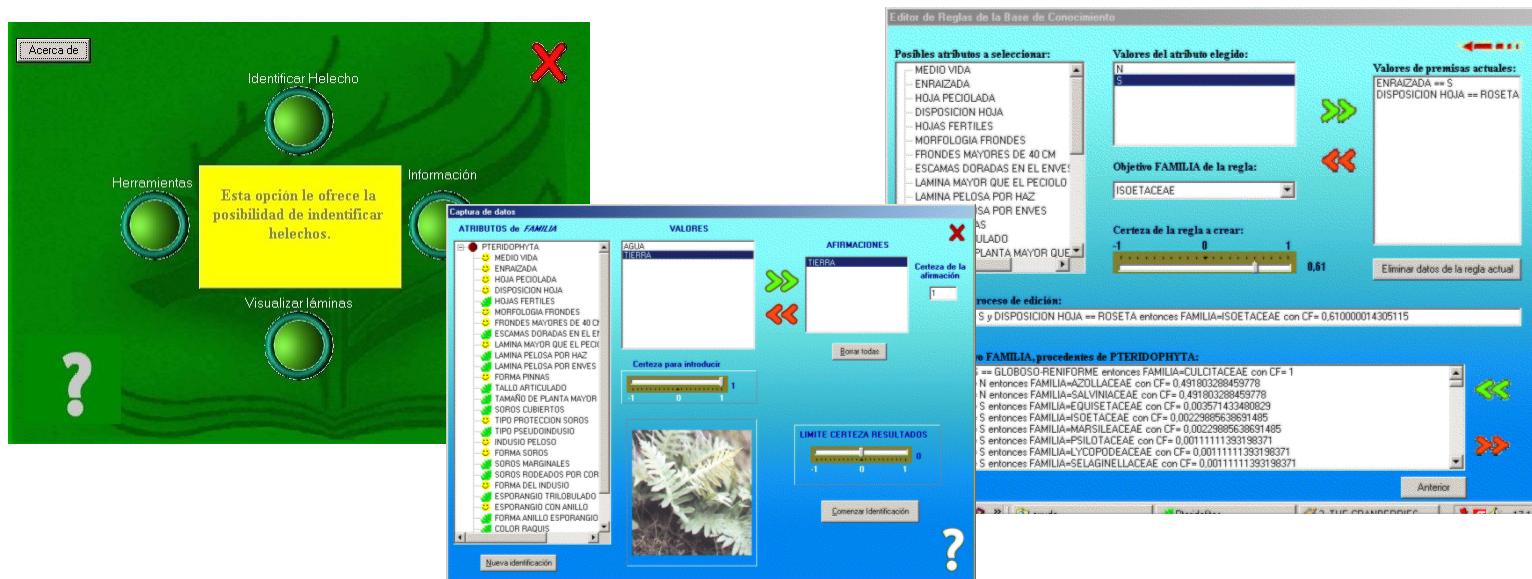
Sistemas Basados en el Conocimiento

- Ejemplos de **SBC** que incorporan conocimiento específico sobre algún tema:
 - El **Paciente Simulado Virtual (PSV)** es un agente que simula enfermedades. Se utiliza para la formación de especialistas en medicina: El médico pregunta y/o visualiza síntomas que el paciente presenta y debe decidir un diagnóstico válido.



Sistemas Basados en el Conocimiento

- Ejemplos de **SBC** que incorporan conocimiento específico sobre algún tema:
 - **Pteridophita** es un experto en helechos que, de forma interactiva con un usuario, permite identificar tipos de helechos y proporcionar información sobre los mismos.



Sistemas Expertos basados en Reglas (SEBR)

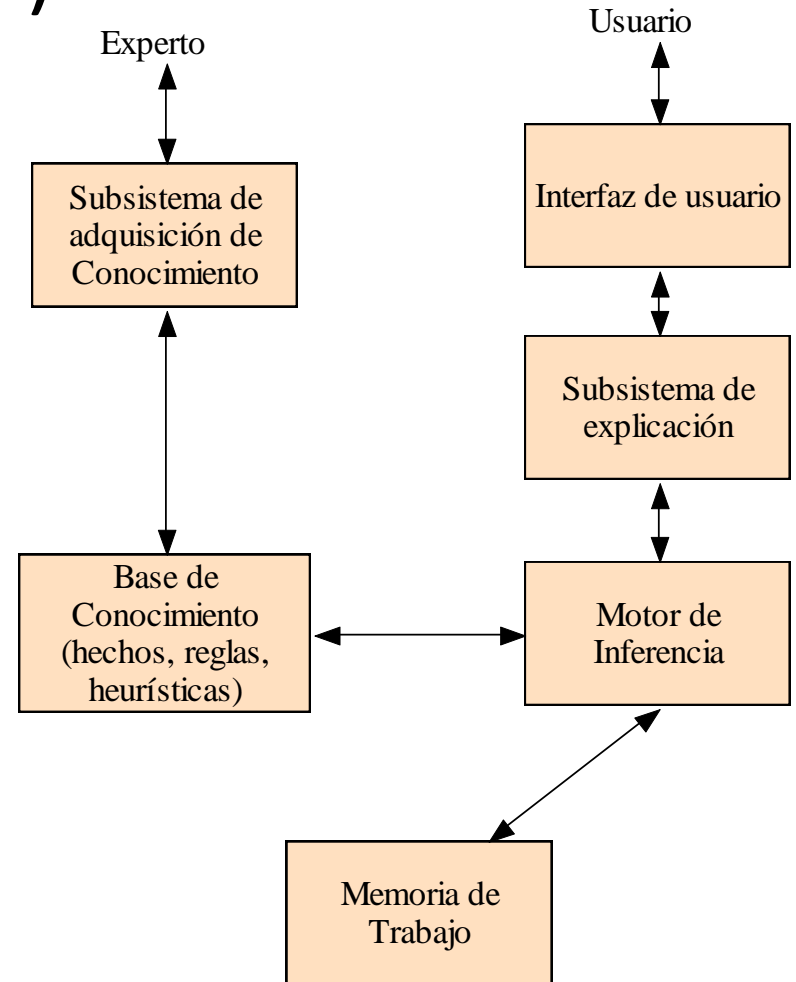
- Un **SEBR** es un **SBC** donde el conocimiento se incluye en forma de reglas y hechos.
- Estas reglas y hechos pueden implementarse, por ejemplo, mediante el **cálculo de predicados**.
- El proceso de construcción de un SEBR es el siguiente:
 - Se **extrae el conocimiento** experto (bibliografía, entrevistas a expertos reales, etc.).
 - Se **modela y se adquiere el conocimiento**, utilizando un lenguaje adecuado (cálculo de predicados, otras lógicas más avanzadas, etc.)
 - Se **crea la Base de Conocimiento** con el conocimiento adquirido.

Sistemas Expertos basados en Reglas (SEBR)

- Por otra parte, también se necesita:
 - Una **interfaz de usuario**, para poder utilizar el sistema y adquirir/enviar datos.
 - Un **subsistema de explicación**, para los casos en los que sea necesario indicar al usuario porqué se llega a las conclusiones que se llegan.
 - Un **Motor de Inferencia**, para razonar sobre la Base de Conocimiento y los datos proporcionados por el usuario.

Sistemas Expertos basados en Reglas (SEBR)

- El esquema general de diseño de un SEBR es el siguiente:
- La **memoria de trabajo** contiene la información relevante que el Motor de Inferencia está usando para razonar las respuestas para el usuario.



Extracción del conocimiento en SEBR

- La forma más sencilla de representar los datos de un experto para la posterior extracción de conocimiento es mediante Bases de Datos relacionales y tablas relacionales. Ejemplo (Sistema Pteridophita). El objetivo es conocer la familia del helecho que se tiene

FAMILIA	MEDIO VIDA	DISPOSICIÓN HOJA	FRONDES > 40 CM	LAMINA > PECIOLO
ADIANTACEAE	TIERRA	INDIVIDUAL	S	S
PTERIDACEAE	TIERRA	INDIVIDUAL	S	S
ISOETACEAE	TIERRA	ROSETA	N	S
ADIANTACEAE	TIERRA	INDIVIDUAL	N	S/N
ADIANTACEAE	TIERRA	INDIVIDUAL	S	N
PTERIDACEAE	TIERRA	INDIVIDUAL	N	S

- Claramente, de la tabla se puede inferir visualmente que se extraerá la regla entre otras.

$$DisposicionHoja(ROSETA) \supset Familia(ISOETACEAE)$$

Extracción del conocimiento en SEBR

- Los métodos para extraer el conocimiento se estudian en el área de **aprendizaje automático** dentro de la IA. En SEBR, la extracción del conocimiento también se conoce como **extracción de reglas** o **aprendizaje de reglas**.
- Además, normalmente, la percepción del mundo por parte de un agente no es perfecta.
- Del mismo modo, es posible que una regla no sea aplicable siempre (aunque sí en un gran número de casos). Este hecho no permite que la regla sea admitida en un sistema de cálculo proposicional o de predicados, dado que daría lugar a sistemas inconsistentes.
- Se hace necesario establecer mecanismos para **tratar con incertidumbre**.

Tema 6: Introducción al Aprendizaje Automático



Objetivos

- Conocer el concepto de aprendizaje automático.
- Conocer los fundamentos, necesidad y utilidad de agentes capaces de aprender.
- Entender las técnicas básicas de aprendizaje automático. Conocer las técnicas necesarias para el aprendizaje de árboles de decisión. Saber resolver problemas en los que sea necesaria la aplicación de esta técnica.

Estudia este tema en ...

- S. Russell, P. Norvig, *“Inteligencia Artificial: Un enfoque moderno”*, Ed. Prentice Hall, 2ª edición, 2004, pp.649-789
- T. Mitchell, *Machine Learning*, Ed. Mac Graw-Hill, 1998, pp. 1-390.
- Nils J. Nilsson, *“Inteligencia Artificial: Una nueva síntesis”*, Ed. Mc Graw Hill, 2000. pp. 255-268

Contenido

- Distintos tipos de aprendizaje
- Modelos inductivos sobre árboles de decisión

Aprendizaje Automático (Machine Learning)

El aprendizaje es una capacidad fundamental de la inteligencia humana, que nos permite:

- Adaptarnos a cambios de nuestro entorno.
- Desarrollar una gran variedad de habilidades.
- Adquirir experiencia en nuevos dominios.

¿Para qué?

- El aprendizaje automático cubre una amplia gama de fenómenos como:
 - El perfeccionamiento de la habilidad.
 - La adquisición del conocimiento.
- El aprendizaje es esencial en entornos desconocidos.
- Programa de IA (Búsqueda, SBC, Planificación, ...):
 - Su límite está en el conocimiento que se les ha proporcionado.
 - No resuelven problemas mas allá de esos límites.
- El aprendizaje modifica el mecanismo de decisión del agente para mejorar su comportamiento.
- Aprendizaje automático: programas que mejoran su comportamiento con la experiencia.

¿Qué aprender?

Does Machine Learning Really Work?

Tom Mitchell. AI Magazine 1997

- ¿Donde y para que se puede usar el aprendizaje automático?
 - Tareas difíciles de programar (reconocimiento de caras, voz, ...)
 - Aplicaciones auto adaptables (interfaces inteligentes, spam killers, sistemas recomendadores, ...)
 - Minería de datos (análisis de datos inteligente)

Definición

- Un programa de ordenador se dice que aprende de la experiencia E con respecto a alguna clase de tareas T y a alguna medida de comportamiento P , si su comportamiento en tareas de T , medido a través de P , mejora con la experiencia E .

Ejemplos

- Aprendizaje de damas:
 - T: jugar a las damas.
 - P: % de juegos ganados.
 - E: partidas jugadas contra una copia de si mismo.

Ejemplos

- Aprendizaje de reconocimiento de caracteres de escritos a mano:
 - T: reconocer y clasificar palabras escritas a mano a través de imágenes.
 - P: % de palabras correctamente clasificadas.
 - E: una base de datos de palabras escritas a mano con su correspondiente clasificación.

Ejemplos

- Aprendizaje de un sistema para conducir:
 - T: conducir por una carretera usando sensores de visión.
 - P: distancia promedio antes de que se produzca un error (juzgado por un humano).
 - E: secuencia de imágenes y comandos de conducción registrados en la observación de un conductor humano.

Ejemplos del sector financiero y seguros

<http://www.daedalus.es/mineria-de-datos/>

- Estimación de Riesgos en la Concesión de Seguros de Crédito
- Detección y Control de Fraude en el Uso de Tarjetas de Crédito
- Segmentación de Clientes de Entidades Financieras

Ejemplos del sector sanitario y farmacéutico

- Predicción de Ventas de Productos Farmacéuticos
- Diagnóstico de Accidentes Cerebro-Vasculares Agudos
- Supervisión de Calidad del precultivo en el Cultivo Industrial de Antibióticos
- Supervisión de la Evolución de Cultivos en la Fabricación de Antibióticos

Ejemplos del sector industria

- Optimización de Centrales Eléctricas
- Control de Trenes de Laminado en la Industria del Acero
- Optimización de Altos Hornos
- Optimización de la Producción de Cartón en la Industria Papelera
- Gestión de Alarmas en Plantas Petroquímicas
- Control de Calidad en la Fabricación de Electrodomésticos
- Optimización del Proceso de Producción de Cemento
- Control de Calidad de Materiales Fabricados Industrialmente
- Control Adaptativo para Optimización de Trayectorias de Robots Industriales
- Control de Calidad en la Fabricación de Cajas de Cambio en la Industria del Automóvil
- Diagnóstico Automático de Componentes de Sistemas hi-fi para Automóviles

Tipos de aprendizaje

- Uno de los puntos clave para el aprendizaje es el tipo de realimentación disponible en el proceso:
 - Aprendizaje supervisado: Aprender una función a partir de ejemplos de sus entradas y salidas.
 - Aprendizaje no supervisado: Aprender a partir de patrones de entradas para los que no se especifican los valores de su salidas.
 - Aprendizaje por refuerzo: Aprender a partir del refuerzo que devuelve el entorno.

Ejemplo

- Supermercado: se desea clasificar los clientes entre buenos y malos clientes
- Base de datos: información acerca de los clientes y forma de pago de los mismos.

Id	Casado	N-hijos	Sexo	Pago	Buen-cliente
1	si	3	m	Tarjeta	sí
2	no	0	h	Tarjeta	sí
3	no	1	m	Efectivo	no
4	si	4	m	Crédito	sí
5	si	2	h	Efectivo	no
6	no	1	m	Tarjeta	no
7	no	0	h	Efectivo	sí
8	no	0	h	Crédito	sí
9	no	1	h	Tarjeta	no
10	si	4	m	Crédito	sí

Si N-hijos > 2 ENTONCES Buen-cliente=si

Si Casado=no Y sexo=h Y N-hijos=0 ENTONCES Buen-cliente=sí

Ejemplo

- Ejemplo: modelado de la probabilidad de fallo de una máquina.
- Clase: la máquina fallará / la máquina no fallará.
- Atributos: conjunto de medidas:
 - Temperatura.
 - Nivel de vibraciones.
 - Horas de funcionamiento.
 - Meses desde la última revisión.
- Instancias: ejemplos pasados (situaciones conocidas).
- Hipótesis: relación entre las medidas y la probabilidad de fallo:

Si nivel_vibraciones = alto Y temperatura = alta ENTONCES fallará.

Espacio de hipótesis

- Las hipótesis se pueden expresar de diversas formas:
 - Árboles de decisión.
 - Reglas.
 - Redes neuronales.
 - Modelos bayesianos o probabilísticos.
 - etc.
- Los árboles de decisión y las reglas son algunos de los modelos más usados en aprendizaje automático.

WEKA



- Algoritmos clásicos como CART, ID3, C4.5 o CN2.
- También existen sistemas de inducción de reglas o árboles de inducción integrados en paquetes de minería de datos o aplicaciones estadísticas.
- WEKA es un conjunto de librerías JAVA para la extracción del conocimiento desde bases de datos.
- Fue desarrollado por la Universidad de Waikato bajo licencia GPL.
- WEKA contiene una gran cantidad de métodos y un entorno de experimentación para comparar el rendimiento de los algoritmos utilizando técnicas estadísticas.
- <http://www.cs.waikato.ac.nz/ml/weka/>