

Tema-5.pdf



Blublu__3



Inteligencia Artificial



2º Grado en Ingeniería Informática



Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación
Universidad de Granada

**QUIERES
15€?**

TRAER A TU CRUSH
DE APUNTES ♡



si juegas con
fuego te fiegas

WUOLAH



QUIERES
CONSEGUIR
15€??

TRÁENOS A TU
CRUSH DE APUNTES
ANTES DE QUE
LOS QUEME



Tema 5

5.1 Introducción.

Hemos estudiado varias formas de modelar un mundo de un agente entre ellas:

- Representaciones icónicas: simulaciones lo más acotadas posibles al mundo que un agente puede percibir. Por ejemplo: las matrices que hemos representado para almacenar un mapa del mundo.
- Representaciones descriptivas: valores binarios que describen aspectos ciertos o falsos sobre el mundo. Estos aspectos son características o propiedades o atributos de objetos que nos interesan representar. Por ejemplo el bikini y las zapatillas en la práctica 2.

Aunque las representaciones icónicas son eficientes las representaciones descriptivas tienen una serie de ventajas por si mismas:

- Son más sencillas de representar.
- Son fáciles de comunicar a otros agentes.
- Se pueden descomponer en piezas más simples. Esto nos interesa porque queremos comunicar sobre solo ciertas partes y no sobre todo el mundo

Además es importante tener en cuenta que hay información del entorno del agente que no vamos a poder representar con modelos icónicos como por ejemplo:

- Leyes generales: "Todas las cajas azules pueden ser cogidas".
- Información negativa: "El bloque A no está en el suelo".
- Información incierta: "O bien el bloque A está sobre el bloque B o bien está sobre el bloque C".

Sin embargo este tipo de información es fácil de formular en modelos descriptivos, es decir, a partir de un conjunto de restricciones sobre los valores de esas características binarias de esa gente. Este tipo de restricciones es lo que representa el conocimiento sobre el mundo y esto es lo que nosotros vamos a intentar ver cómo se puede representar.

Además no solo nos va a interesar representar el conocimiento, nos va a interesar razonar sobre él y hallar nuevas características del mismo. Entonces si nosotros somos capaces de manejar y razonar este conocimiento vamos a ver como lo podemos hacer sobre máquinas o agentes. Para ello vamos a estudiar dos tipos básicos para representar el conocimiento y razonar sobre él:

- Cálculo proposicional
- Cálculo de predicados.



si
consigues
que suba
apuntes, te
llevas 15€ +
5 Wuolah
Coins para
los sorteos

5.2 Cálculo Proposicional

Elementos básicos de representación: proposiciones y conectivas.

$$\wedge(\text{y}), \vee(\text{o}), \rightarrow(\text{implica}), \neg(\text{no})$$

Mecanismos de inferencia: serie de reglas que vamos a usar para hacer deducciones a partir de estas.

Una de las ventajas fundamentales de este esquema de conocimiento es que es muy general y además es decidible es decir en tiempo finito podemos encontrar un algoritmo que sea capaz de decidir si una proposición es deducible de la información de la que disponemos.

El principal problema del calculo proposicional es que si queremos razonar sobre conjuntos de cosas va a ser muy ineficiente o muy complicado de representar.

5.2.1 Demostración

Una demostración es una secuencia de pasos para llegar a una conclusión partiendo de unas premisas conocidas. Las premisas las vamos a representar como un conjunto llamado delta (Δ).

Las fórmulas bien formadas (FBFs) son fórmulas que se construyen las reglas del lenguaje. Y a la conclusión la vamos a denominar teorema y vamos a decir que el teorema se puede demostrar desde delta mediante una demostración con la siguiente notación: $\Delta \vdash w$.

5.2.2 Reglas de inferencia

Podemos aplicar las siguientes reglas de inferencia:

- Reglas de equivalencia:
 - $\neg(p \wedge q) = (\neg p) \vee (\neg q)$
 - $\neg(p \vee q) = (\neg p) \wedge (\neg q)$
 - $\neg\neg p = p$
- Inferencia (o deducción)
 - p, q : por tanto $p \wedge q$
 - Modus ponens ($p \rightarrow q$ y p ; por tanto q). Si sé que p implica q y conozco p , por tanto, puedo obtener q .

5.2.3 Regla de resolución.

Para conocer en qué consiste esta regla necesitamos conocer el concepto de:

- Cláusula: una clausula es un conjunto de literales donde un literal puede ser una proposición o su negación y este conjunto es una forma de representar una disyunción de todos los literales que pertenecen al conjunto. Por tanto una cláusula es una FBF.
- Refutación: es útil para demostrar que la negación de una cláusula es inconsistente en el sistema, quedando así demostrada, por tanto, la veracidad de dicha cláusula.

La regla de la resolución la voy a utilizar como una especie de regla de cálculo que me va a permitir resolver dos cláusulas. Dos cláusulas se pueden resolver si una de ellas tiene un literal y la otra contiene su negación. Por lo tanto el resultado de resolver es la unión de las dos clausulas sin ese literal. Al resultado de esta operación se le llama resolvente y al proceso resolución.

$$\begin{array}{ccc} \lambda \vee \Delta 1 & \neg \lambda \vee \Delta 2 & \rightarrow \Delta 1 \Delta 2 \\ & \text{resolución} & \rightarrow \text{resolvente} \end{array}$$

QUIERES
CONSEGUIR
15€??

TRÁENOS A TU
CRUSH DE APUNTES
ANTES DE QUE
LOS QUEME



si consigues que suba apuntes, te llevas 15€ + 5
Wuolah Coins para los sorteos

WUOLAH

La resolución en el cálculo proposicional es un algoritmo que tiene una secuencia de pasos que me va a permitir determinar si una determinada sentencia puede ser cierta o no a partir de cierto conocimiento que ya tengo. Los pasos de este algoritmo son:

1. Convertir las FBF que tengamos en conjunciones de cláusulas.
2. Convertir la negación de la sentencia que queremos demostrar como una conjunción de cláusulas.
3. Unir el resultado de los pasos 1 y 2 en un único conjunto en el que están todas las cláusulas que vamos a manejar.
4. Aplicar distintas etapas de resolución de manera iterativa ante la selección de distintas parejas de cláusulas de manera que vamos a proceder hasta que no haya nada más que resolver o se llegue a nil (el conjunto vacío).

Finalmente, como podemos observar el procedimiento de refutación que hemos seguido mediante la aplicación de varias resoluciones consiste en la selección en cada paso de una pareja de cláusulas. Hemos visto que la selección de cláusulas para su resolución de forma manual es relativamente sencilla, evidentemente con grandes conjuntos de cláusulas vamos a llevar un proceso de búsqueda en el que vamos a tener que utilizar distintas estrategias que nos van a permitir determinar la selección de las parejas de cláusulas más adecuadas para resolver y para conseguir una mayor eficiencia.

Aunque el cálculo proposicional es un mecanismo de representación de conocimiento general y prácticamente cualquier esquema de conocimiento podría representarse como un conjunto de proposiciones tiene varias dificultades, especialmente cuando lo que queremos representar son dominios en los que tenemos objetos, estos tienen propiedades y relaciones entre ellos.

5.3 Cálculo de Predicados

El cálculo de predicados lo vamos a utilizar cuando tengamos conjuntos de objetos y queremos representar propiedades de objetos y relaciones entre ellos. Los elementos del lenguaje de la lógica de predicados son:

- Términos constantes, Variables y funciones.
- Fórmulas atómicas: predicados definidos sobre términos
- Fórmulas bien formadas (fbf): fórmulas atómicas unidas por conectivas ($\wedge, \vee, \neg, \rightarrow$) y cuantificadas (\exists, \forall).

5.3.1 Reglas de inferencia

En el caso de la lógica de predicados vamos a tener las mismas reglas que teníamos en la lógica proposicional más una regla especial que se relaciona con las variables que se llama instanciación universal.

La instanciación universal me va a permitir deducir nuevo conocimiento sobre objetos concretos a partir de la definición de predicados generales sobre variables y el cuantificador asociado.

Hasta el momento lo único que hemos hecho es representar información pero esto lo podríamos representar en una base de datos de forma más sencilla. Sin embargo, si lo representamos como una base de conocimiento podemos establecer un procedimiento automático para que aparezcan nuevas relaciones en nuestra base de conocimiento sin que nosotros tengamos que utilizarlo.

QUIERES
CONSEGUIR
15€??

TRÁENOS A TU
CRUSH DE APUNTES
ANTES DE QUE
LOS QUEME



5.3.2 Deducción hacia atrás

Este proceso me va a permitir respuestas a preguntas que yo pueda plantearme. Estos son procesos de deducción regresivos. Los pasos de esta deducción son los siguientes:

- Nos realizamos una pregunta y la representamos en el lenguaje como una fórmula de lógica de predicados en la que hay variables que no están instanciadas. Por ejemplo: matriculado-en(anaMorales,Y)
- Se busca una implicación lógica en la que aparezca matriculado-en (V,V1)
- Si se puede unificar, se intentan deducir los literales que aparezcan en la parte izquierda de la implicación.
- Si existe alguna asignación de valor a las variables de la parte izquierda que me permitan deducir como ciertas las condiciones, se podrá deducir la pregunta de diferentes formas.

5.3.3 Características

Ventajas:

- Nos proporciona una representación de tipo general más rica que la proposicional.
- Características de un sistema de razonamiento lógico:
 - Solidez: nos permite determinar si estamos seguros de que una conclusión inferida por un algoritmo de razonamiento es cierta.
 - Completitud: nos permite asegurarnos de que cuando existe una conclusión para una inferencia tarde o temprano se va a producir esa conclusión
 - Decidibilidad: es una propiedad que vamos a usar para asegurarnos de que, en efecto, la inferencia que queremos realizar es factible
- La refutación mediante resolución es sólida y completa.

Problema:

- El cálculo de predicados es semidecidible y además en los casos en los que la refutación mediante resolución termina, el procedimiento es NP-duro.

5.3.4 PROLOG y otras lógicas

En cualquier caso hay una solución a este problema de semidecidibilidad y alto coste computacional y es intentar representar el conocimiento basándonos en la lógica de predicados a partir de subconjuntos decidibles de lógica de predicados que es mediante las cláusulas de Horn pero extendiéndolas al concepto de cláusula no proposicional sino de cláusula de predicados.

Una cláusula de Horn es una cláusula (disyunción de literales) con, como máximo, un literal positivo.

El lenguaje de programación PROLOG es un lenguaje de programación lógico que permite crear y ejecutar programas en lógica de predicados que están representadas como cláusulas de horn. En PROLOG estas cláusulas se van a representar como reglas en las que el consecuente está en la parte izquierda y el antecedente está en la parte derecha. Por ejemplo:

conectados(X,Y) :- conectados(Y,X)

En PROLOG una consulta se representa como una cláusula que solo tiene antecedente

:- Encima(A,C)

si
consigues
que suba
apuntes, te
llevas 15€ +
5 Wuolah
Coins para
los sorteos



Los hechos se representan con cláusulas que solo tienen antecedente

Sobre(A,B) :-

El proceso de deducción que aplica PROLOG sobre las distintas cláusulas puede representarse como un árbol Y/O.

Finalmente, comentar que la lógica proposicional y la lógica de predicados no son las únicas lógicas que se utilizan en IA para representar conocimiento. Nos podemos encontrar con:

- Lógicas de segundo orden (o de orden superior) que nos permiten representar predicados como argumentos de los predicados
- Lógicas modales y temporales: son lógicas importantísimas para el razonamiento temporal. Nos van a permitir expresar que determinados hechos o reglas o son necesarios que se cumplan o es posible que se cumplan dando lugar a razonamiento contingente.
- Lógica difusa: en esta no existe un grado de certeza de cierto y falso para cada sentencia sino que hay grados de verdad.
- Otras: lógicas multivaluadas en las que de nuevo los valores de verdad pueden ser múltiples, lógicas no monótonas, lógicas cuánticas etc..

La lógica difusa es una extensión de la lógica clásica diseñada para permitir el razonamiento sobre hechos imprecisos cuando hay incertidumbre representada como imprecisión. Uno de los éxitos de la lógica difusa está dentro de los sistemas de control difusos que se usan, por ejemplo, en el control del movimiento de las articulaciones de un robot.

5.4 Sistemas Basados en el Conocimiento

Estos sistemas todavía a día de hoy son los que representan la mayor cantidad de aplicaciones reales y con éxito de la IA y se basan en que tienen que gestionar una gran cantidad de conocimiento. Todos estos sistemas se llaman así porque el conocimiento ocupa la parte central y fundamental en la solución al problema a resolver.

La arquitectura de estos sistemas se componen de tres elementos básicos:

1. Una base de conocimiento(BC) que es el almacén donde va a estar el conocimiento necesario para resolver el problema. Este conocimiento va a ser normalmente conocimiento experto, es decir, proporcionado por una persona experta en un determinado dominio. Una base de conocimiento puede ser:
 - Estática: si no varía a lo largo del tiempo
 - Dinámica: cuando permite que se añadan nuevos hechos o reglas o que se modifiquen las existentes a lo largo del tiempo (son las más comunes).
2. Un motor de inferencia que es el que tiene representado el algoritmo de razonamiento que nos va a permitir inferir nuevo conocimiento o razonar en general sobre el conocimiento de la base de conocimiento y los datos proporcionados por un usuario que nos van a representar situaciones concretas o problemas concretos que tenemos que solucionar.
3. Una interfaz de usuario para poder interactuar con el sistema.

5.4.1 Sistemas Expertos basados en Reglas

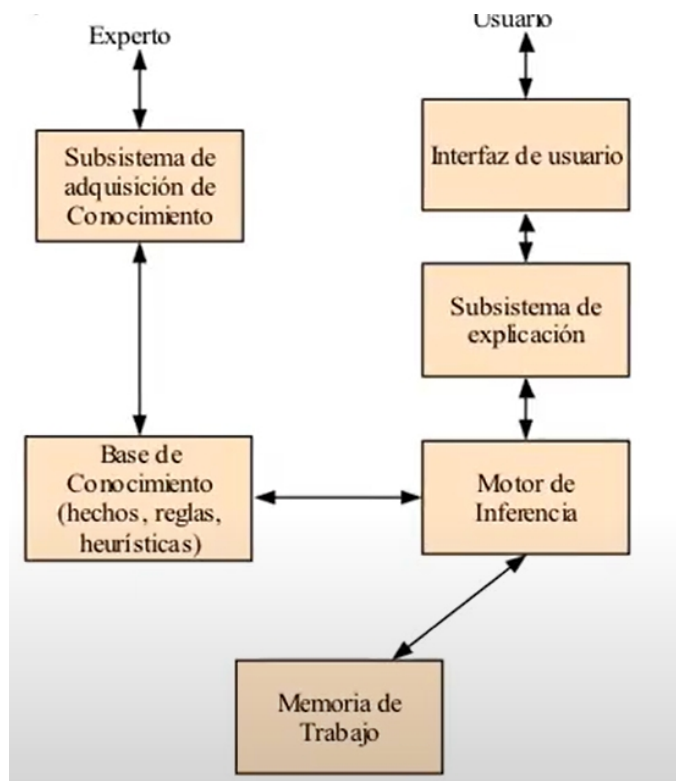
Un tipo muy común y específico de sistema basado en conocimiento son los sistemas expertos basados en reglas en los que la principal característica es que el conocimiento que está representado en la BC tiene la forma de reglas y hechos. Las reglas y los hechos pueden implementarse, por ejemplo, mediante el cálculo de predicados.

El sistema de construcción de un SEBR ha sido objeto de estudio durante mucho tiempo y ha recibido mucho esfuerzo tanto de investigación como de empresas y ha dado lugar a una disciplina que se llama la ingeniería del conocimiento. La ingeniería del conocimiento se estudian los distintos métodos que podemos tener para representar conocimiento dentro de una BC y, en general, cualquier metodología de ingeniería de conocimiento tiene 3 etapas generales:

- Extracción de conocimiento.
- Modelado y adquisición en el que utilizamos un lenguaje adecuado para representar el conocimiento que se ha abstraído en la fase anterior.
- Creación de la BC en la que se almacena físicamente el conocimiento que se ha adquirido.

Puesto que los SEBR están muy destinados a ayudar en la toma de decisiones a expertos en cualquier dominio. A parte de la interfaz de usuario y del motor de inferencia estos sistemas requieren de un sistema de explicación para los casos en los que sea necesario indicar al usuario porqué se llega a las conclusiones que se llegan.

El esquema general de diseño de un SEBR es el siguiente:



QUIERES
CONSEGUIR
15€?? →

TRÁENOS A TU
CRUSH DE APUNTES
ANTES DE QUE
LOS QUEME 🔥



Experto: es el que va a permitir que se almacene conocimiento y se cree dentro de la BC.

Base de conocimiento: va a aportar conocimiento al motor de inferencia y este a su vez con las conclusiones lo puede modificar.

Motor de inferencia: va a utilizar el subsistema de explicación que, a través de la interfaz de usuario, se va a comunicar con el usuario.

Memoria de trabajo: contiene la información relevante que el motor de inferencia está usando para razonar las respuestas para el usuario.

5.5 Otros modelos/problemas de representación del conocimiento

5.5.1 Organización jerárquica del conocimiento

Esta organización está muy relacionada con la idea de jerarquía de clases en la PDOO. Uno de los aspectos interesantes de esta organización es que podemos establecer mecanismos para la herencia de propiedades.

5.5.2 Redes semánticas

Son estructuras gráficas (estructuras de grafos) que nos van a permitir representar el conocimiento taxonómico (de categorías) sobre objetos y propiedades de estos. Vamos a poder representar propiedades como nodos etiquetados con constantes de relación y objetos como nodos etiquetados con constantes de objetos y los arcos que unen los distintos nodos del grafo tienen también una interpretación diferente. Me puedo encontrar arcos que me permitan establecer una relación jerárquica entre nodos, relaciones de pertenencia o arcos que me permitan representar una función.

5.5.3 Razonamiento temporal

Las lógicas temporales lo que tratan de representar es que hay una serie de procesos y eventos que ocurren en el tiempo, es decir, a parte de representar objetos y sus propiedades me permiten representar eventos o sucesos que ocurren en intervalos de tiempo.



si
consigues
que suba
apuntes, te
llevas 15€ +
5 Wuolah
Coins para
los sorteos