




INTELIGENCIA ARTIFICIAL

**CONCEPTOS, AGENTES Y
REPRESENTACIÓN DEL CONOCIMIENTO**



Structuralia

Este documento es de uso único e intransferible para el alumno matriculado en el curso. Cualquier reproducción física o digital del documento sin permiso de los autores vulnera los derechos de propiedad intelectual de los mismos.



INDICE

INDICE.....	2
1. INTRODUCCIÓN.....	4
2. INTELIGENCIA ARTIFICIAL DÉBIL Y FUERTE.....	5
3. EL TEST DE TURING.....	6
4. AGENTES.....	7
4.1 Agentes reactivos.....	7
4.2 Agentes deliberativos.....	8
5. ARQUITECTURAS DE AGENTES.....	10
5.1 Autómatas finitos.....	10
5.2 Arquitecturas BDI.....	13
6. REPRESENTACIÓN DEL CONOCIMIENTO.....	14
6.1 El proceso de Ingeniería del Conocimiento.....	14
6.2 Ejemplo de ontología.....	16
6.3 Aplicando la lógica de primer orden.....	17
7. REFERENCIAS.....	20

1. INTRODUCCIÓN

Antes de ahondar en cada una de las áreas de la inteligencia artificial existen una serie de conceptos básicos, transversales a todas ellas, que facilitarán el estudio y comprensión de las mismas. En los siguientes apartados se presenta cada concepto junto con una breve explicación y su principal uso.

2. INTELIGENCIA ARTIFICIAL DÉBIL Y FUERTE

La inteligencia artificial en su sentido más abstracto siempre se ha relacionado con el pensamiento [1]. Por ello, existen distintas corrientes, hasta cierto punto filosóficas, sobre qué significa que una máquina piense o tenga consciencia (y sus implicaciones).

- La inteligencia artificial fuerte es una corriente que defiende que las máquinas sí piensan (pensamiento real).
- La inteligencia artificial débil es una corriente que defiende que las máquinas actúan con inteligencia, pero no piensan por sí mismas (pensamiento simulado).

Actualmente, la mayoría de los investigadores son afines a la corriente de la inteligencia artificial débil, pues presumir de que las máquinas sean racionales es demasiado ambicioso. En cualquier caso, lo más importante es ser conscientes de las implicaciones éticas que tiene cualquier sistema de este tipo. Esto se tratará en profundidad en el último tema del presente curso.

3. EL TEST DE TURING

En el segundo tema ya se presentó a Alan Turing, el matemático impulsor de la computación y la inteligencia artificial. Una de sus grandes contribuciones fue “el test de Turing”. A pesar de que el trabajo se publicó en 1950, a día de hoy sigue vigente en el mundo académico a la hora de evaluar la inteligencia de cualquier sistema.

El test de Turing está compuesto de una serie de pruebas específicas para ver si el evaluador, ajeno al sistema inteligente, es capaz de distinguir si está interactuando con una persona o una máquina. El principal requisito del test es que el evaluador no tenga en ningún momento contacto visual con el sistema, pues sino la complejidad aumentaría muchísimo al tener que simular a un ser humano. El evaluador establecerá una serie de preguntas que la máquina deberá responder. Como el evaluador a lo largo del proceso desconoce qué hay al otro lado (máquina o humano) sólo podrá basar su criterio en la calidad de las respuestas obtenidas a sus preguntas. Se considera que un sistema ha pasado la prueba si el evaluador es incapaz de distinguir entre las respuestas del mismo y las de un humano.

Existe un segundo test, conocido como Test Global de Turing, que sí que incluye visión artificial para identificar objetos y manipulación para poder moverlos (que entraría dentro del área de la robótica).

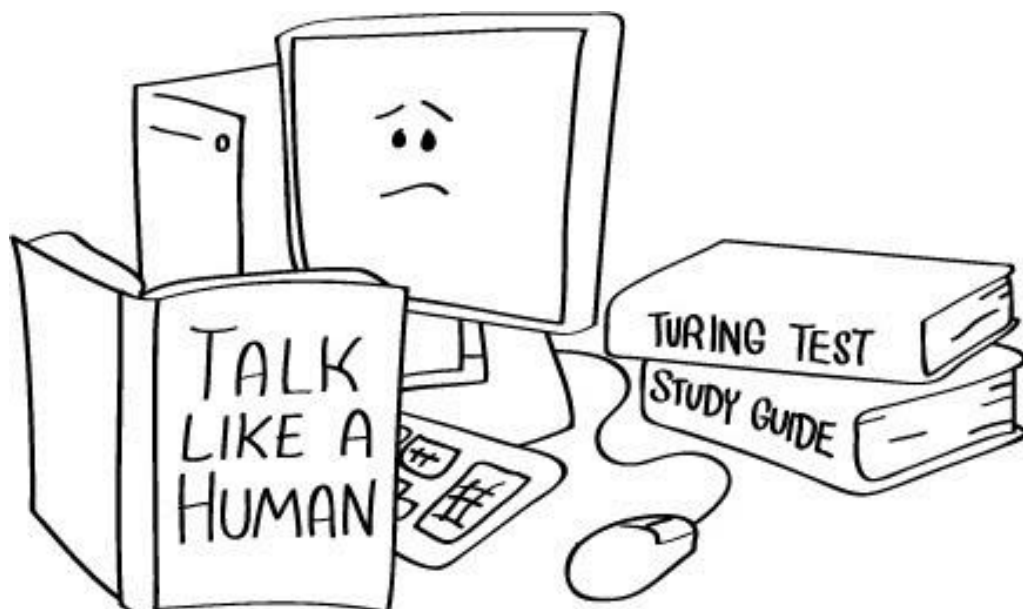


Ilustración 1: Viñeta cómica que ilustra el Test de Turing

4. AGENTES

En inteligencia artificial llamamos agente a cualquier ente o elemento físico o virtual que sea capaz de percibir qué sucede en el entorno, procesarlo y ejecutar cambios o interaccionar sobre el mismo.

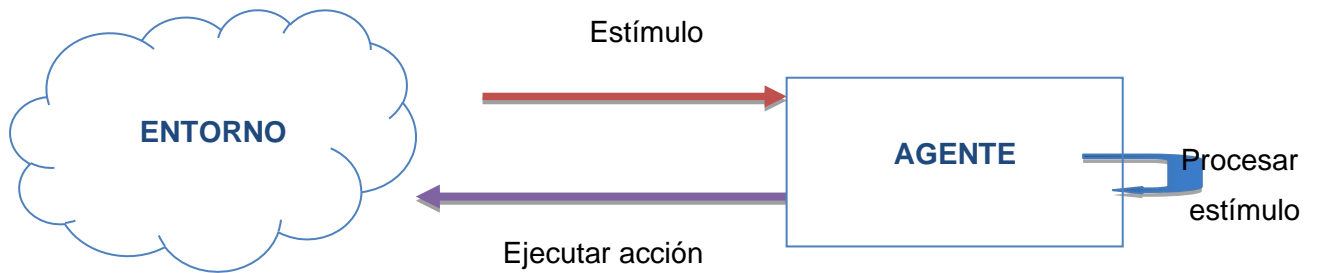
Desde un punto de vista computacional, el agente no es más que un conjunto de líneas de código integrado en un sistema o arquitectura. Ocasionalmente y sobre todo en el área de procesamiento del lenguaje natural, a los agentes se les identifica como bots. Cuando el agente tiene además un cuerpo físico, se le llama robot.

En el área de la inteligencia artificial y los agentes existen dos tipos de comportamiento: reactivo y deliberativo.

4.1 Agentes reactivos

El comportamiento reactivo es aquel que no tiene en cuenta el historial previo de acciones o situaciones anteriores y/o que no es capaz de desarrollar nuevas formas de actuar ante el mismo evento. Los agentes reactivos sólo son capaces de ejecutar comportamientos del estilo: condición → acción. Ejemplo: “si está lloviendo, entonces activar limpiaparabrisas”, “si botella vacía, entonces llenar con agua”.

Como consecuencia, el abanico de comportamientos que puede ejecutar un agente reactivo está limitado por la base de conocimiento que tenga disponible. Por otro lado, los comportamientos que se ejecuten nunca serán demasiado elaborados o complejos y se evitará en la medida de lo posible la interacción con humanos. Un agente reactivo de este tipo nunca será capaz de entender ni asimilar los cambios que genere en el entorno.



Existe una evolución de agente reactivo que basa sus decisiones en un modelo de conocimiento previamente construido en lugar de utilizar la forma condición \rightarrow acción. Llamamos modelo a una base de conocimiento organizada en diferentes niveles de condiciones. Los sistemas expertos de los 80 y 90 pertenecen a este tipo de agentes. Este tipo de agente es más consciente del entorno y sus cambios, aunque no lo entiendan desde el punto de vista de nuevo conocimiento.

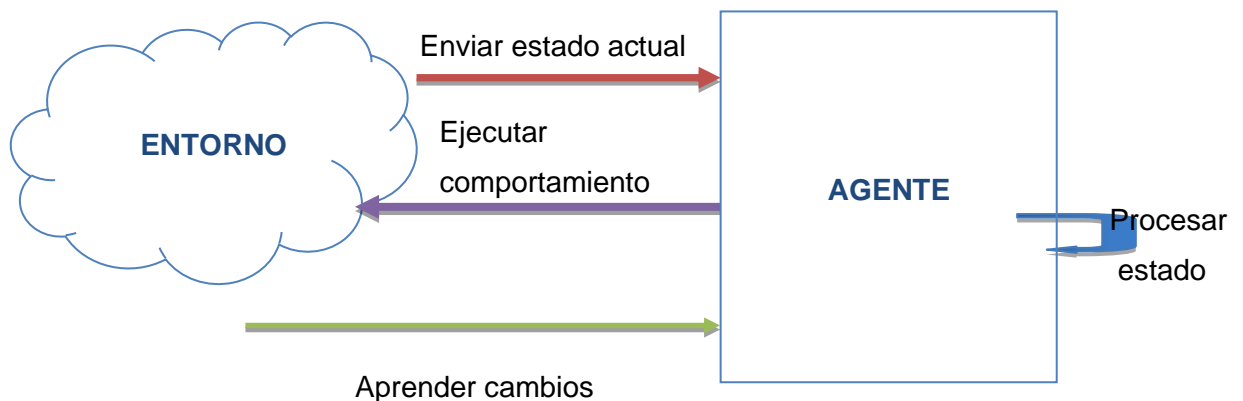
Ejemplo: Si el nivel de agua del depósito es bajo, mantener las compuertas abiertas hasta detectar nivel alto.

4.2 Agentes deliberativos

Por otro lado, el comportamiento deliberativo es aquel que conlleva un razonamiento lógico. El agente es plenamente consciente del entorno y sus cambios. Sabe desenvolverse ante dichos cambios al igual que aprende de sus acciones y percepciones.

Debido a ello, la complejidad para desarrollar un agente deliberativo aumenta considerablemente. La máxima expresión de este tipo de comportamiento serían los agentes autónomos, que son capaces de razonar ante cualquier tipo de situación.

Generalmente, un agente deliberativo actúa sobre el entorno según el siguiente diagrama:



Desde el punto de vista teórico, los agentes deliberativos pasan por cuatro fases o niveles clave:

- **Recopilación de información:** el agente observa el entorno y trata de identificar qué sucede, objetos y/o personas familiares, errores, etc. Todo lo que observa y es capaz de procesar se guarda en su base de conocimiento.
- **Exploración del mundo:** una vez se dispone de la base de conocimiento, el agente es capaz de mejorarla según se vaya moviendo o ejecutando cambios sobre el entorno. La mejora de la base de conocimiento está ligada a la capacidad de aprendizaje del agente.
- **Aprendizaje:** el proceso de aprendizaje comienza tras la ejecución de las acciones del agente sobre el entorno. Hay diferentes formas de llevarlo a cabo, (1) a través de una función de evaluación propia para medir el éxito, (2) esperando la puntuación del humano, (3) incorporando directamente el nuevo conocimiento sin evaluación.
- **Autonomía:** el agente alcanzará un grado de autonomía proporcional a las habilidades que pueda desarrollar y su adaptabilidad al entorno. Esto viene dado por la combinación de base de conocimiento + capacidad de aprendizaje.

5. ARQUITECTURAS DE AGENTES

En la sección anterior se han explicado los dos diferentes tipos de agentes y sus formas de actuar sobre el entorno. Aún queda un paso más: cómo se modela y controla ese comportamiento. Los científicos han implementado diferentes modelos o arquitecturas que permiten trabajar de forma sencilla con agentes. Para los agentes reactivos utilizaremos autómatas finitos; para los deliberativos la arquitectura BDI.

5.1 Autómatas finitos

Un autómata finito es un modelo computacional que cada vez que recibe una entrada realiza cálculos o acciones de forma automática producir una salida. La idea de los autómatas la construyó Alan Turing sobre “The Turing Bombe” y sirvió para sentar las bases de la computación. Existen dos tipos de autómatas finitos: deterministas y no deterministas.

5.1.1. Autómatas finitos deterministas

El modelo teórico de autómata finito determinista, como se puede ver en la figura a continuación, está compuesto de:

- **Estados (q_0 , q_1 , q_2 ...):** puede representar desde un número o letra hasta el estado concreto de un agente. En el modelo teórico, el estado inicial se representa señalado con una flecha (q_0) y el estado final o estado meta con un doble círculo (q_f). Ambos estados pueden o no coincidir. Puede haber varios estados meta pero sólo habrá un estado inicial por autómata.
- **Transiciones:** indican el estado siguiente al cuál transitar una vez se recibe el valor (número, letra, acción...) como entrada.
- **Valores sobre las transiciones:** indican los valores (números, letras, acciones...) que el autómata sabe procesar del entorno.

Se le llama autómata determinista porque desde un estado y escogiendo un valor se transita a otro estado diferente. Ninguna transición con origen en el mismo estado y destino en dos estados diferentes tiene el mismo valor asociado (lo llamaríamos autómata no determinista).

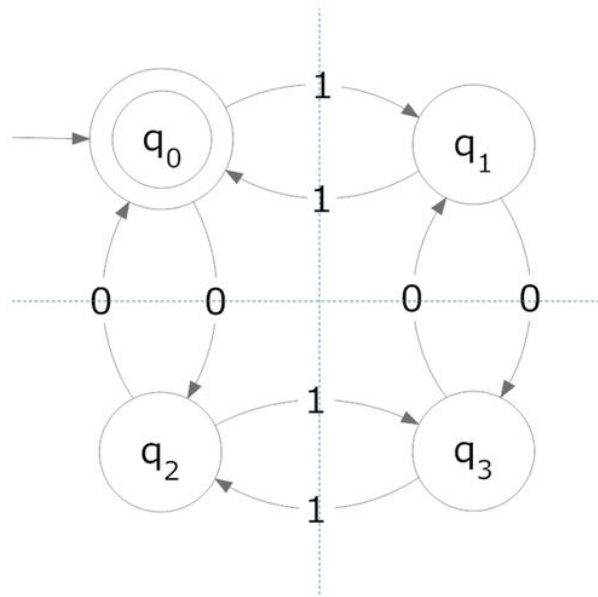


Ilustración 2: Autómata finito determinista. El estado q_0 es inicial y final

5.1.2. Autómatas finitos no deterministas

El modelo teórico de un autómata finito no determinista es el mismo que el explicado anteriormente con una diferencia. Las transiciones con un mismo valor asignado pueden dirigirse a más de un estado.

En el ejemplo a continuación disponemos de un autómata con tres estados. El estado inicial es q_0 y los tres estados son finales. Se puede observar cómo de q_0 con 'a' se transita a q_1 o q_2 . Lo mismo sucede en q_1 con 'b'. Debido a estas incertidumbres el nombre que se les asigna es el de 'no determinista'.

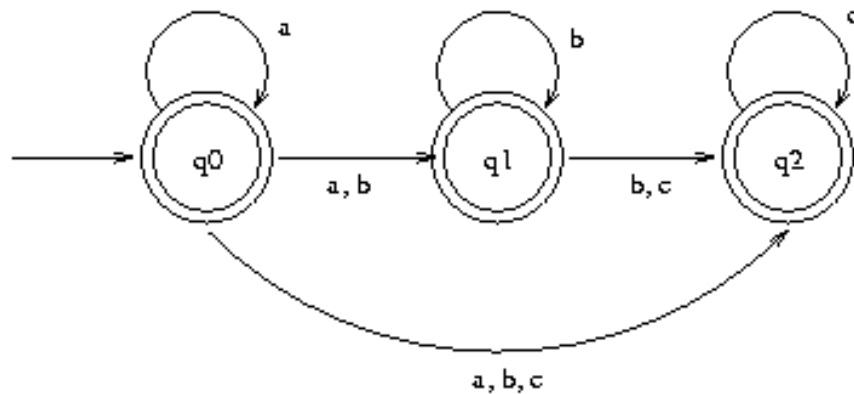


Ilustración 3: Autómata finito no determinista

5.1.3. Ejemplo de autómata aplicado a agente robótico

Cuando se trabaja con estructuras de control para agentes reactivos se utiliza un modelo de autómata más práctico, sobre todo si queremos que el agente realice acciones de forma repetitiva. Esto significa que no hay estado final como tal (en el modelo teórico del ejemplo sería S3). Las transiciones equivalen a acciones ejecutadas sobre el entorno.

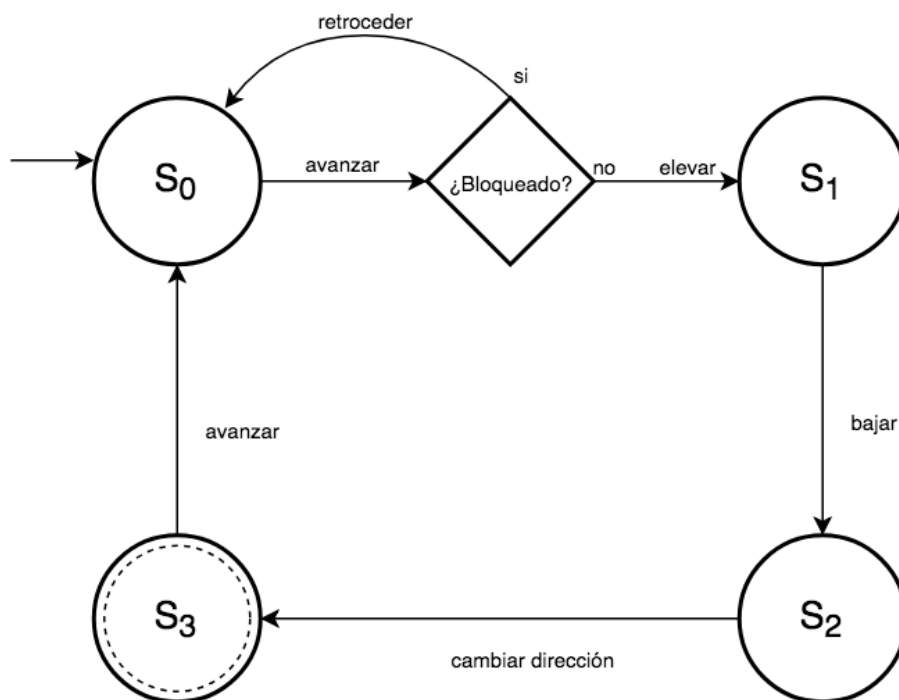


Ilustración 4: Sistema de control basado en un autómata

5.2 Arquitecturas BDI

A la hora de diseñar sistemas de control para agentes deliberativos, la arquitectura más conocida es la BDI [2], cuyas siglas vienen del inglés:

- **Beliefs (creencias):** Se refiere al conocimiento del agente sobre el entorno. El agente cree fervientemente en todos y cada uno de los hechos almacenados en su base de conocimiento.
- **Desires (deseos):** Equivalen al conjunto de metas a conseguir por el agente. Los deseos son las prioridades del agente sobre el entorno.
- **Intentions (intenciones):** Son las que conducen al agente hacia las metas, las encargadas de satisfacer los deseos del agente. Cualquier intención también podría afectar a la base de Beliefs del agente y actualizarla.

El sistema de control en combinación con estos tres elementos quedaría como se muestra en la siguiente figura:

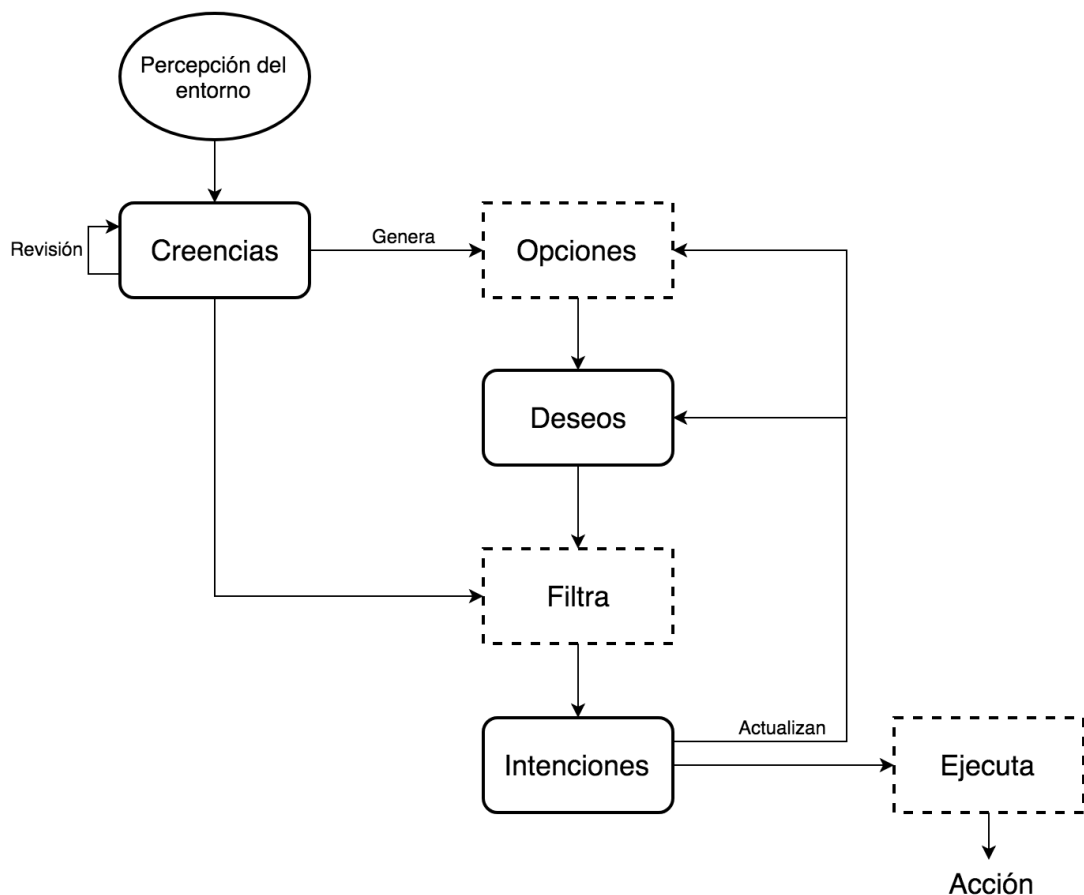


Ilustración 5: Ejemplo de arquitectura BDI

6. REPRESENTACIÓN DEL CONOCIMIENTO

Una de las partes clave de la inteligencia artificial es la correcta representación de la información (conocimiento) y del problema. La calidad de ambas afectará directamente a la eficiencia del agente, su inteligencia y posterior ejecución.

La mayoría de la información que se utiliza dentro de un sistema computacional inteligente se encuentra organizada en jerarquías y a su vez cada uno de sus elementos tiene algún tipo de relación o vinculación con otro/s. En la representación de la información, se acota la expresividad del propio lenguaje para reducirlo a los componentes esenciales y así simplificar el trabajo a la máquina que posteriormente deberá procesarlo.

Este tipo de lenguaje o definición de conocimiento está englobado dentro de la lógica de primer orden y engloba los siguientes elementos:

- **Hechos:** combinación generalmente verdadera de entidad/es + relación.
- **Entidades:** elementos del dominio o entorno que sabemos que existen.
- **Relaciones:** generalmente se dan entre objetos del mismo o diferente tipo.

Los hechos presentes en lógica de primer orden deberán tomar el valor de verdadero o falso. Verdadero equivale a que el hecho se cumple/verifica en el entorno. Falso, lo contrario.

Una vez se establece la jerarquía de objetos, las relaciones entre los mismos y los hechos que inicialmente son verdaderos en el entorno, se dice que se ha desarrollado una ontología.

6.1 El proceso de Ingeniería del Conocimiento

En las ciencias de la computación se llama ontología a una definición de tipos, propiedades y relaciones entre entidades que existen para un dominio o entorno en particular. De forma práctica, los hechos, entidades y relaciones se representan con predicados, objetos y constantes. Cuando la representación es compleja se utilizan niveles o jerarquías para clasificar el conocimiento. Una ontología o un conjunto de ontologías forma lo que se llama base de conocimiento.

El proceso de ingeniería del conocimiento para la elaboración de ontologías o bases de conocimiento consiste en los siguientes pasos:

- **Identificar la tarea a resolver:** como responsables de la elaboración de la ontología tendremos que definir cuál es la información que queremos modelar. Generalmente existe un término que engloba a toda la jerarquía.

El entorno en el que se va a desarrollar dicho sistema también ayuda a acotar la tarea. Por ejemplo, una base de conocimiento podría contener todos los términos y relaciones entre miembros de la familia o bien contener lenguaje técnico-científico sobre huesos porque se va a utilizar posteriormente para el análisis automático de radiografías.

Ejemplo: elaborar una base de conocimiento para predecir si una persona será solvente o no a la hora de concederle un préstamo.

- **Recopilar el conocimiento relevante:** una vez definida la tarea, se deben buscar y recopilar fuentes de información sobre elementos clave que intervengan en la misma o que pudiesen afectar. Si es necesario, también se podría entrevistar a expertos para tener presentes distintos casos de uso o nuevo conocimiento que no se contemplaba en un inicio.

Ejemplo: analizar indicadores financieros, nómina del trabajador, tipo de contrato, lista de morosos etc.

- **Decidir el vocabulario de los componentes:** cuando se elabora una base de conocimiento hay que tener en cuenta quién va a ser nuestro cliente final. Generalmente se utiliza un vocabulario técnico, relacionado con el área para el cual estamos elaborando la base de conocimiento. Cuanto más específico sea el tema más exigente será el cliente final. Las entrevistas con expertos también ayudan a solventar este punto.

Ejemplo: para trabajar en el sector financiero/banca es necesario incluir en la base de conocimiento términos como %T.A.E, interés fijo, interés variable, capacidad de crédito, aval, SWIFT, etc.

- **Modelar el conocimiento del dominio:** después de realizar las tres tareas anteriores hemos finalizado con el proceso de investigación y entendemos el entorno de nuestra base de conocimiento. El siguiente paso será ordenar toda esa información, términos y relaciones entre ellos. Decidiremos cuáles son los elementos principales, cuáles son los atributos de esos elementos, cómo están interconectados, qué acciones se pueden llevar a cabo etc.

Ejemplo: siguiendo con los préstamos bancarios habrá que agrupar el conocimiento en aquellos elementos que interesan al banquero y aquellos que evalúan la solvencia de una persona. Las acciones en su mayoría serían cálculos de indicadores, consulta de históricos y listas negras.

- **Modelar el conocimiento de un problema específico del dominio:** en el momento en el que se decide que la base de conocimiento está completa hay que evaluarla antes de dársela al cliente final. Para ello, se definen distintos casos de uso y se define uno o varios problemas en el ámbito de esa tarea inicial. A la hora de resolver cada uno de los problemas podremos verificar si la información presente en la base de conocimiento es suficiente o si, por el contrario, hay que ampliarla o profundizar en algunos aspectos. También puede suceder que falte alguna relación entre elementos y a consecuencia no se pueda llevar a cabo una acción.

Ejemplo: principalmente habría que modelar el caso de uso en el que la persona es solvente y el caso en el que no lo es. Adicionalmente podrían modelarse casos que estuviesen en el punto medio para ver cómo se comporta el sistema.

- **Evaluar el procedimiento de inferencia:** después de iterar sobre la base de conocimiento y disponer de una versión definitiva junto con todos esos escenarios sobre los que se ha evaluado la misma, se procede a evaluar el sistema con expertos. Ellos llevarán a cabo la tarea para la cual la base de conocimiento ha sido elaborada. También se les podría pedir que resolviesen cada uno de los casos de uso de la fase previa. Nuestro papel será apuntar las opiniones de cada uno de los expertos, los problemas a los que se han enfrentado, si han identificado alguna carencia y si hay que perfeccionar algún término o relación.

Ejemplo: para evaluar la base de conocimiento se pueden elaborar una serie de reglas del estilo “si entonces” que lleven a los expertos desde el inicio “el cliente quiere solicitar un préstamo X” hasta su concesión o denegación.

- **Depurar la base de conocimiento:** con toda esta información previa se procede a depurar la base de conocimiento y llegado a este punto ya estará operativa para que la utilice el cliente final. Debido a que las áreas de conocimiento, especialmente las científico-técnicas, evolucionan bastante rápido, será conveniente revisar y actualizar la base de conocimiento periódicamente.

6.2 Ejemplo de ontología

En la siguiente figura se muestra un ejemplo gráfico de cómo quedaría (parcialmente) modelada una base de conocimiento sobre el árbol y su denominación científica.

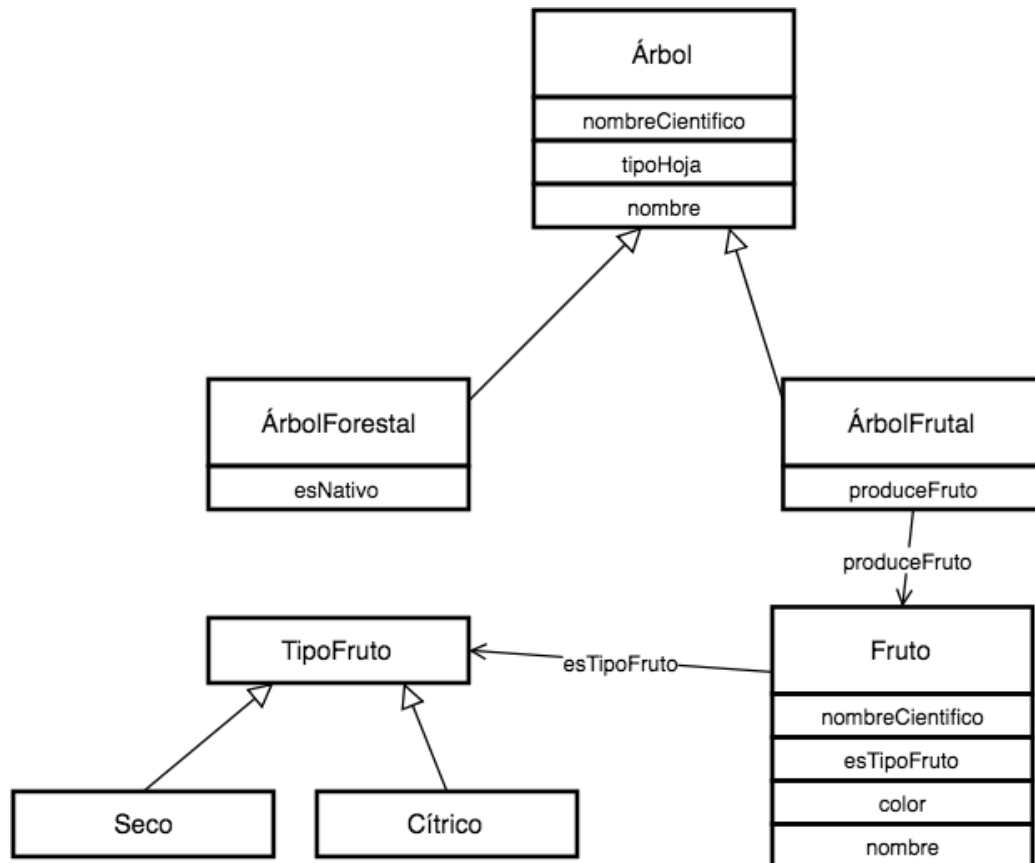


Ilustración 6 Ejemplo básico de ontología con la familia de los árboles

Dicha ontología podría aplicarse a tareas como “análisis y viabilidad de nuevas plantaciones” o “identificación temprana de árboles”.

6.3 Aplicando la lógica de primer orden

Finalmente, en este último apartado del presente tema se explica una de las bases de la representación del conocimiento: la lógica de primer orden [1].

Desde el nacimiento de la Inteligencia Artificial, la lógica siempre ha estado en un lugar muy cercano a este campo debido a su origen matemático en la teoría de conjuntos y el álgebra de Boole.

No obstante, los investigadores de la computación demandan una forma más descriptiva de representar el conocimiento, de tal forma que los sistemas inteligentes sean más fáciles de interpretar por los humanos. Así que se remitieron a finales de 1800 cuando un grupo de matemáticos definió por primera vez la lógica de primer orden. Existen otro tipo de lógicas como la matemática, la proposicional etc.

La lógica de primer orden es un sistema formal que permite estudiar y representar la inferencia de un conjunto de variables, predicados y funciones. Como consecuencia, la lógica de primer orden también se conoce como lógica de predicados.

Llamaremos término a cualquier objeto, persona o elemento de la sentencia. Llamaremos predicado a las palabras que representan relaciones entre los términos. Las relaciones se establecen a través de los argumentos del predicado, lo cuáles siempre serán términos.

Una sentencia lógica básica podría ser Curar (Herida, Pomada) o Abuelo (Julio, Miranda). Esto se leería como “la herida se cura con pomada” o “Julio es el abuelo de Miranda”. El orden de lectura siempre será 1er término, predicado, 2º término.

A continuación, se muestra una lista de símbolos que se utilizan para representar y dotar de mayor expresividad a las sentencias lógicas.

Símbolo	Tipo	Significado
\exists	Cuantificador existencial	Existe
\forall	Cuantificador universal	Para todo
\neg	Conectiva	No
\wedge	Conectiva	Y
\vee	Conectiva	O
\Rightarrow	Conectiva	entonces
\Leftrightarrow	Conectiva	equivale

La ventaja de poder utilizar este tipo de cuantificadores y conectores es la generalización de las sentencias lógicas. Cuando previamente hemos descrito:

Abuelo(Julio, Miranda)

Podemos convertir a Julio y a Miranda en variables que pueden representar cualquier nombre propio. Por ejemplo vamos a llamarlas persona1 y persona2.

Abuelo(persona1, persona2) = “persona1 es abuelo de persona2”

Dando un paso más allá, se podría representar la siguiente frase “para todo nieto existe un abuelo”, la cual ya albergaría en su definición la sentencia lógica anterior. Esto se representaría de la siguiente manera:

$\forall \text{ nieto } \exists \text{ abuelo} = \forall x, \exists y \text{ Abuelo}(x, y)$

No obstante esta representación nos aporta poco, pues no contiene ningún tipo de inferencia de conocimiento, que es el objetivo de este método. Como se ha mencionado anteriormente, los predicados crean relaciones y los conectores son capaces de unir dos o más relaciones. Entonces ya podríamos representar sentencias como la siguiente:

$\forall x, \exists y \text{ Madre}(x, y) \Leftrightarrow \text{Hija}(y, x)$; “todo x tiene una madre y, entonces y es hija de x”.

7. REFERENCIAS

- [1] S. J. Russell and P. Norvig, Artificial Intelligence: A Modern Approach 3rd Ed. (1994) <http://aima.cs.berkeley.edu/>.
- [2] Rao, A. S., & Georgeff, M. P. (1995, June). BDI agents: from theory to practice. In ICMAS (Vol. 95, pp. 312-319). <http://www.aaai.org/Papers/ICMAS/1995/ICMAS95-042.pdf>