



Inteligencia Artificial

Unidad 3: Razonamiento basado en conocimiento

TEMA 4: Lógica computacional en la I.A.

Módulo 1: Lógica proposicional

Unidad 3

Razonamiento basado en conocimiento

TEMA 4: Lógica computacional en la I.A.

Semana 11

MÓDULO 1: Lógica proposicional



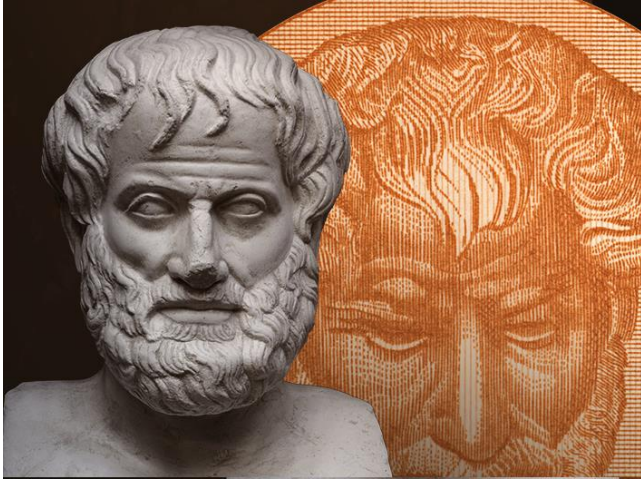
Contenido

1. Origen y evolución de la Lógica
2. Lógica y Computación
3. Agentes basados en conocimientos
4. Lenguajes y modelos lógicos
5. Ejemplos de aplicación



Preguntas

1. Origen y evolución de la Lógica



Aristóteles, reconocido como el padre de la lógica.

El estudio serio de la Lógica como disciplina independiente comenzó con el trabajo de Aristóteles (384-322 a. C.).

Aristóteles elaboró sofisticados escritos sobre lógica y aportó con dos principios de gran importancia en la lógica proposicional:

- **La Ley del Medio Excluido:** es el principio de que todo enunciado es verdadero o falso.
- **Ley de la Contradicción:** es el principio de que ningún enunciado es a la vez verdadero y falso.

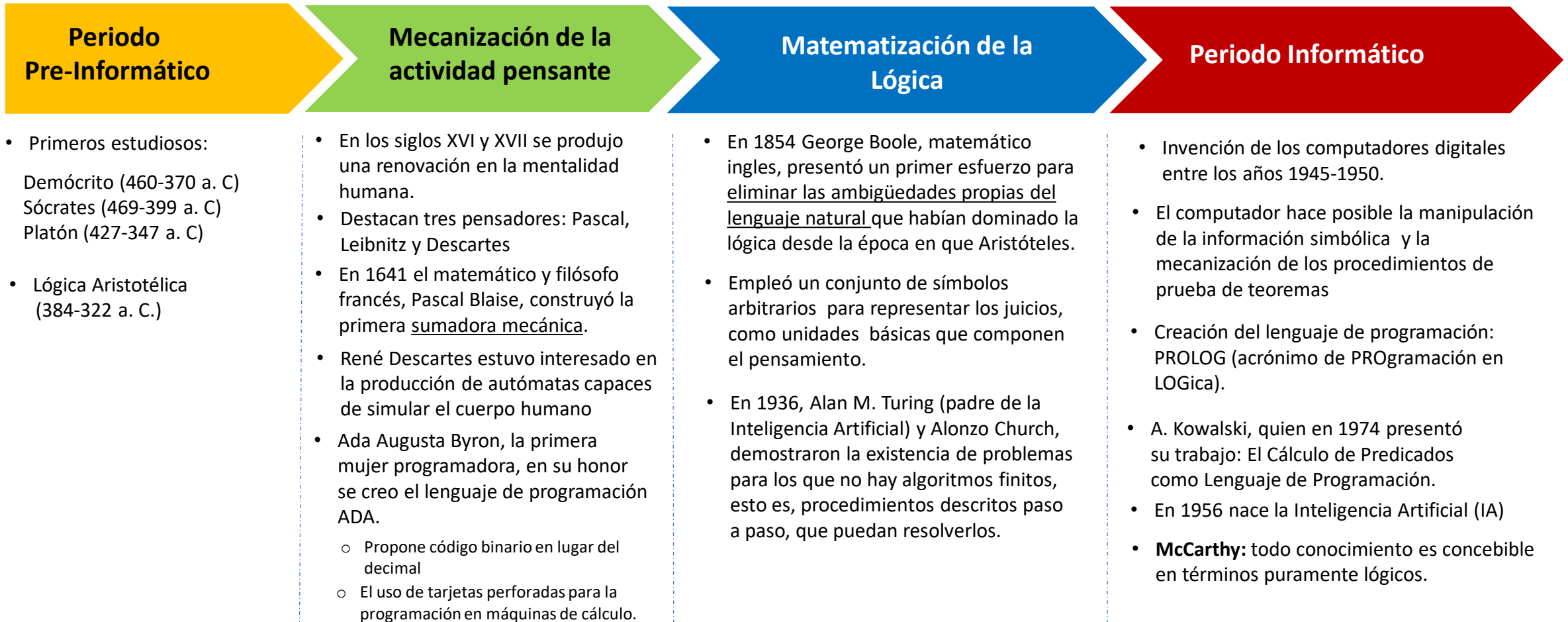
La lógica se puede definir como:

"El estudio de las verdades basado completamente en los significados de los términos que contienen."
Es un proceso para llegar a una conclusión.



1. Origen y evolución de la Lógica

Evolución del campo de estudio de la Lógica



2. Lógica y Computación

Inteligencia computacional

Los científicos han desarrollado formas de representar formalmente el conocimiento (lenguajes lógicos) y utilizar “métodos inteligentes” para resolver problemas con la ayuda de computadoras.

Lógica y Computación se enfocan en métodos y algoritmos lógicos para modelar, construir y analizar sistemas informáticos complejos e inteligentes.

Aplicaciones en:

- Lógica matemática.
- Procedimientos de decisión automatizados y algoritmos y su complejidad.
- Aplicaciones de métodos lógicos a bases de datos y conocimiento.
- Sistemas semánticos.
- Inteligencia artificial.



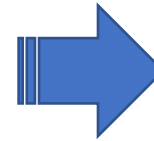
COMPUTATIONAL INTELLIGENCE

3. Agentes basados en conocimientos

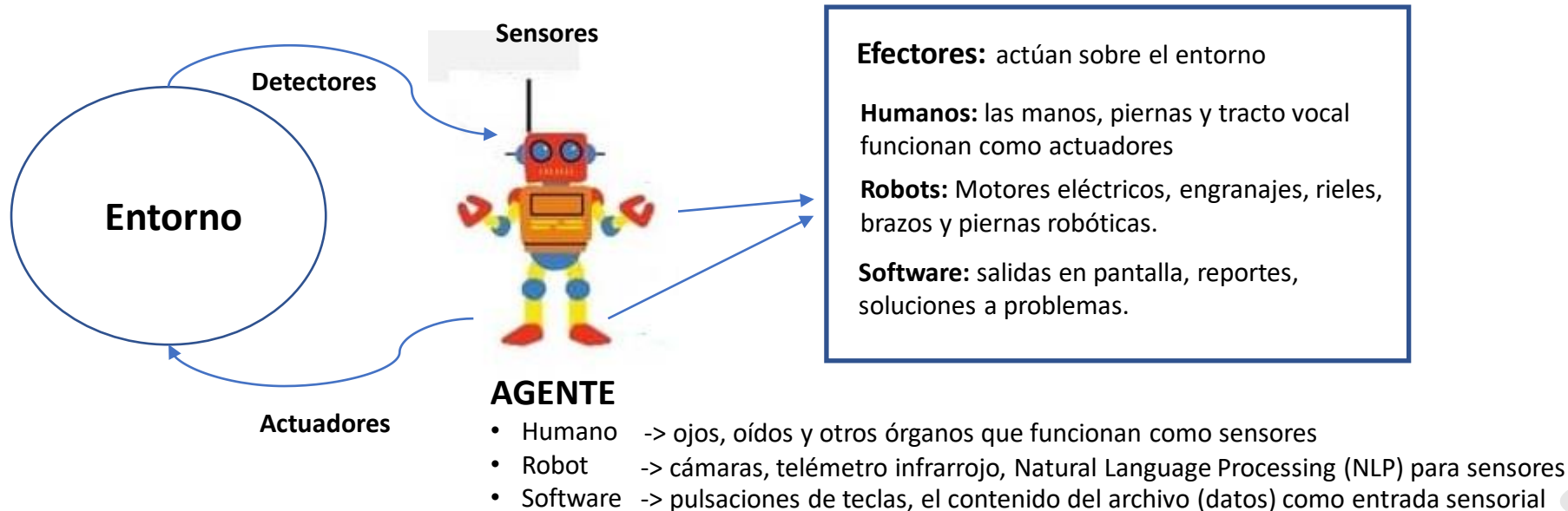
AGENTES INTELIGENTES

- Un **agente** puede ser cualquier cosa que perciba su entorno a través de sensores y actúe sobre ese entorno a través de actuadores.
- El ciclo de un Agente consiste en:
 - ❖ Percibir
 - ❖ Pensar
 - ❖ Actuar

¿Objetivo?



RESOLVER PROBLEMAS



¿Qué es un **AGENTE** en I.A.?



3. Agentes basados en conocimientos

AGENTES INTELIGENTES => AGENTES RACIONALES

Agente Inteligente

- Entidad autónoma que actúa sobre un entorno utilizando sensores y actuadores para lograr objetivos.
- Puede aprender del entorno para lograr sus objetivos.
- Reglas:

Regla 1: La capacidad de percibir el entorno.

Regla 2: La observación debe usarse para tomar decisiones.

Regla 3: La decisión debe resultar en una acción.

Regla 4: La acción realizada por un agente de IA debe ser una acción racional.

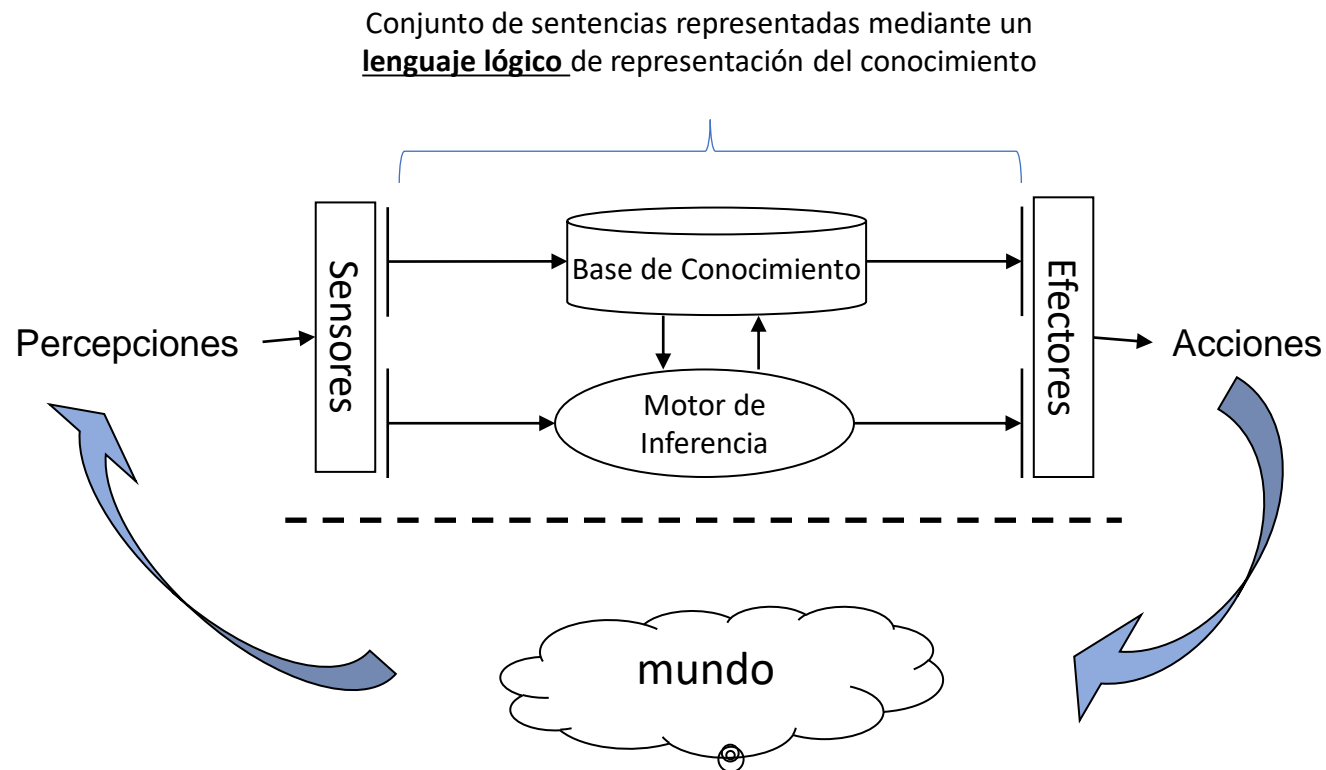


Agente Racional

- Es un agente inteligente que tiene una preferencia clara, resuelve la incertidumbre y actúa de manera que maximiza su medida de desempeño con todas las acciones posibles.
- Se espera que un agente racional realice las cosas correctas.
- **La IA se trata de crear agentes racionales** para usar en la teoría de juegos y la teoría de decisiones en varios escenarios del **mundo real**.

3. Agentes basados en conocimientos

AGENTES INTELIGENTES => AGENTES RACIONALES => AGENTES BASADOS EN CONOCIMIENTOS



- Un Agente Basado en Conocimiento (ABC) es aquel sistema que posee conocimiento de su mundo y que es capaz de razonar sobre las posibles acciones que puede tomar para cambiar el estado de su mundo.

EL ABC consiste principalmente en:

- **Base del conocimiento:**
 - Es un sistema de oraciones que representan hechos acerca del mundo, expresado en algún lenguaje de representación del conocimiento.
 - Cada hecho está representado por una sentencia u oración
- **Motor de inferencia**
 - Deduce nuevas oraciones o sentencias a partir de las oraciones almacenadas en la base de conocimiento y de las nuevas percepciones.
 - Adición de nuevo conocimiento.

3. Agentes basados en conocimientos

Ejecución de un Agente basado en conocimientos (ABC)

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

variables estáticas: BC, una base de conocimiento t, un contador,
al inicio=0, mide el tiempo

DECIR(BC, CONSTRUIR-SENTENCIA-DE-PERCEPCION(*percepción*, t))

acción \leftarrow PREGUNTAR(BC, PEDIR-ACCION(t))

DECIR(BC, CONSTRUIR-SENTENCIA-DE-ACCION(*acción*, t))

$t \leftarrow t + 1$

devolver *acción*

Proceso:

1. Informa a la Base de Conocimiento que el agente está percibiendo algo del ambiente.
2. Pregunta a la Base de Conocimiento cuál es la próxima acción que debe ser ejecutada.
3. Se realiza un proceso de raciocinio lógico sobre la Base de Conocimiento para que se decidan las acciones que deben ser ejecutadas.
4. Realiza la acción escogida e informa a la base de conocimiento sobre la acción que está siendo realizada

3. Agentes basados en conocimientos

Hipótesis de la representación del conocimiento en I.A.

Según los investigadores de la Inteligencia Artificial, para que un sistema sea "**artificial inteligente**" debe contener un componente que se puede entender como lingüístico (es decir, que puede ser expresado en algún lenguaje), tal que:

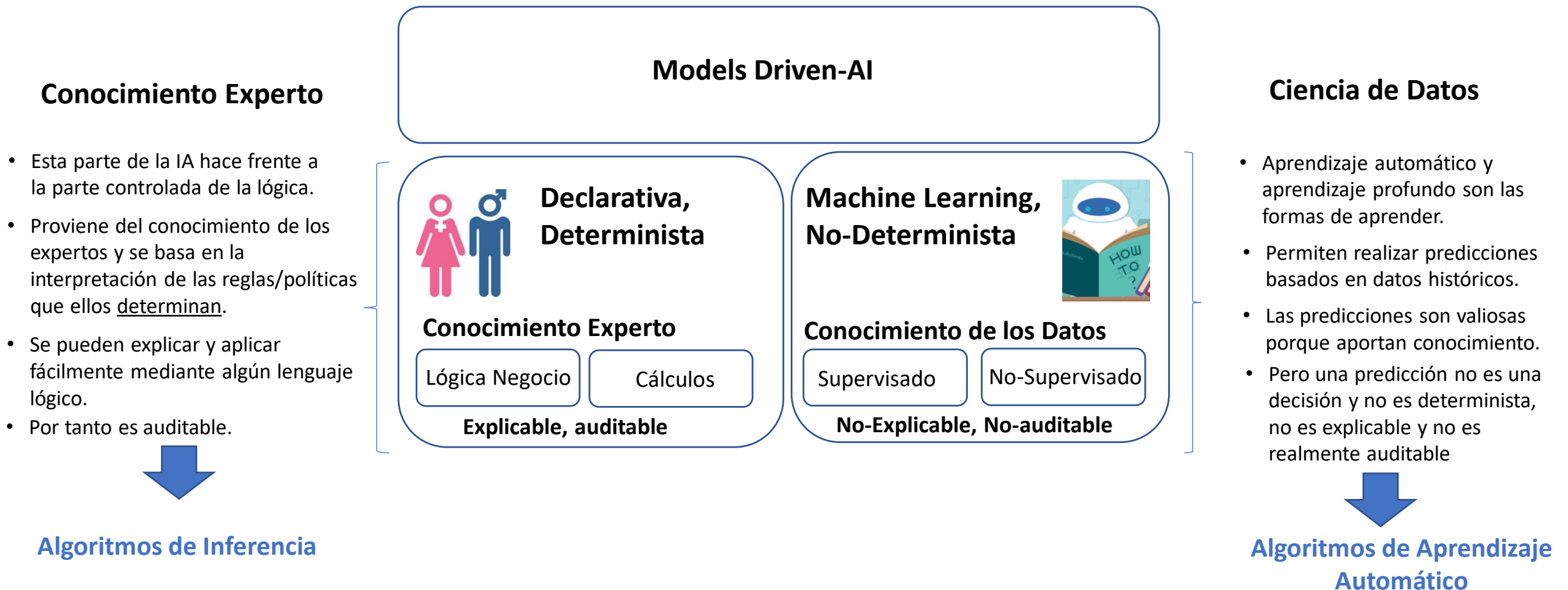
- Este componente contiene el conocimiento del sistema, y
- Este componente conduce el comportamiento inteligente del sistema



Estas condiciones la cumplen los agentes basados en conocimiento

4. Lenguajes y modelos lógicos

Modelos impulsados por la IA



4. Lenguajes y modelos lógicos

La **lógica**, entendida como el estudio de las verdades, se incorpora como **sentencias** dentro un **lenguaje lógico**.

Una lógica es un sistema formal para describir lo que esta sucediendo en un momento determinado y que consta de:

1. **Sintaxis:** Reglas que explican cómo construir oraciones o sentencias legales (correctas).

Ejemplo: $x+y=4$

2. **Semántica:** Especifica el significado de las sentencias. Define el valor de verdad de cada sentencia con relación a cada “mundo posible”.

Ejemplo: la sentencia “ $x+y=4$ ” es verdadera en un mundo en que $x = 2$ e $y = 2$, pero es falsa en un mundo en que $x = 1$ e $y = 1$

3. **Modelo:** Es un “mundo posible”, entorno real en el que el agente puede o no estar.

Ejemplo: La frase “ m es modelo de α ” indica que la sentencia α es verdadera en el modelo m

4. **Implicación o Teoría:** Reglas para inferir oraciones desde otras oraciones (conclusiones).

Ejemplo: $\alpha \models \beta$: Si α es verdadera, β también lo debe ser

4. Lenguajes y modelos lógicos

Es importante incluir la lógica en la **Inteligencia Artificial** porque:

- Queremos que nuestro agente (sistema) piense y actúe humanamente, y para ello debe ser capaz de tomar cualquier decisión en función de la situación actual.
- Si hablamos de comportamiento humano normal, entonces se toma una decisión eligiendo una opción entre las diversas opciones disponibles. Hay razones para seleccionar o rechazar una opción. Entonces, nuestro agente artificial también debería funcionar de esta manera.
- Al tomar una decisión, el agente debe proporcionar las razones específicas en función de las cuales se tomó la decisión.
- Y este razonamiento puede ser realizado por el agente solo si el agente tiene la capacidad de comprender la lógica.

Un **lenguaje lógico** será el idioma para hacer a los agentes “**artificialmente inteligentes**”



4. Lenguajes y modelos lógicos

Si la sintaxis y la semántica están definidas de manera precisa, se dice que el lenguaje es una lógica.

Lenguaje lógico = lenguaje + lógica



Lenguaje de representación del conocimiento

¿Por qué utilizar un
lenguaje lógico?



Porque el lenguaje lógico permite:

- Poder crear agentes.
- Poder expresar lo que el agente sabe a través de sentencias lógicas.
- Que el agente pueda agregar nuevas sentencias a su Base de Conocimiento mientras explora el ambiente (modelo).
- La creación de sistemas de forma declarativa (basado en reglas).

4. Lenguajes y modelos lógicos

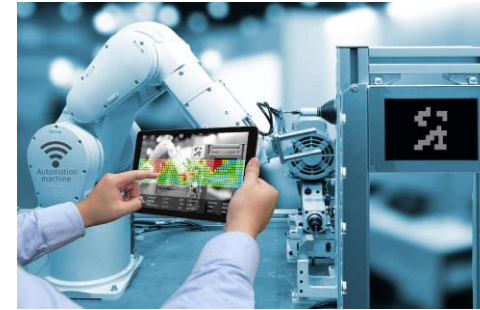
Analogía entre la Programación y los problemas de IA



PROGRAMACIÓN

1. Diseñar un algoritmo para resolver el problema.
2. Seleccionar un lenguaje de programación en el que se pueda codificar el algoritmo.
3. Capturar el algoritmo en un programa.
4. Ejecutar el programa.

Es el paso final que suele implicar la búsqueda.



INTELIGENCIA ARTIFICIAL

(Conocimiento Experto / Lenguaje lógico)

1. Identificar los conocimientos necesarios para resolver el problema.
2. Seleccionar un idioma en el que se pueda representar el conocimiento.
3. Anotar los conocimientos en el idioma.
4. Utilizar las consecuencias del conocimiento para resolver el problema.



4. Lenguajes y modelos lógicos

Modelos lógicos

Todos los lenguajes de representación del conocimiento (lenguajes lógicos) se basan, de cierta manera, en **lógica formal o modelos lógicos** como:

1. **Lógica proposicional**
2. **Lógica de primer orden (o lógica de predicados, cálculo o predicativa)**
3. Lógica temporal
4. Lógica difusa

4. Lenguajes y modelos lógicos

Modelo lógico: LOGICA PREPOSICIONAL

CARACTERÍSTICAS

- Utiliza una lógica sencilla.
- Una **proposición** es una declaración que es verdadera o falsa.
- Las **sentencias** son formadas por conectivos como: “**y**”, “**o**”, “**entonces**”.
- Es necesario definir:
 - ✓ **Sintaxis**: sentencias válidas.
 - ✓ **Semántica**: el cómo se determina la verdad de las sentencias.
 - ✓ **Implicación**: la relación entre una sentencia y otra que se sigue de ella.
 - ✓ Algoritmo para **inferencia lógica** (conclusión).

4. Lenguajes y modelos lógicos

Modelo lógico: LOGICA PREPOSICIONAL

SINTAXIS

SENTENCIAS ATOMICAS (simples)

- Constituidas por elementos sintácticos indivisibles (un único símbolo proposicional)

Ejemplos:

Vero es maestra
Vero es millonaria
Vero es cantante de rock

SIMBOLOS

- Nombres en letras mayúsculas (P, Q, R, ...) .
- Pueden asumir Verdadero o Falso

Ejemplos:

P = Vero es maestra
Q = Vero es millonaria
R = Vero es cantante de rock

SENTENCIAS COMPLEJAS (compuestas)

- Construidas a partir de sentencias más simples con el uso de conectivos lógicos:

\neg (no)	\Rightarrow (implica)
\wedge (y)	\Leftrightarrow (si y solo si)
\vee (o)	

Ejemplos:

No P = $\neg P$
P o Q = $P \vee Q$
R implica Q = $R \Rightarrow Q$

- Las reglas que se aplican a cada conectivo se resumen en una **tabla de verdad**.

4. Lenguajes y modelos lógicos

Modelo lógico: LOGICA PREPOSICIONAL

TABLA DE VERDAD PARA LOS CONECTIVOS

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
F	F	V	F	F	V	V
F	V	V	F	V	V	F
V	F	F	F	V	F	F
V	V	F	V	V	V	V

4. Lenguajes y modelos lógicos

Modelo lógico: LOGICA PREPOSICIONAL

OPERADORES O CONECTORES LÓGICOS

Permiten realizar operaciones lógicas con las preposiciones.



conectivo	Símbolo lógico	Expresión en lenguaje natural	Ejemplos
Negación	$\neg p$	No p No ocurre que p No es cierto que p Es falso que p	Hoy no hace calor No llegaré tarde Eso no es verdad
Conjunción ("y")	$p \wedge q$	p y q p aunque q p pero q p sin embargo q p no obstante q p a pesar de q	Vamos al cine y a cenar también Luis trabaja aunque estudia de noche Llegué a tiempo no obstante haber salido tarde
Disyunción ("o")	$p \vee q$	p o q o ambos O bien p o bien q Al menos p o q Como mínimo p o q	O vamos al cine o vamos a cenar O me saco un 7 o me saco un 8
Condicional implicación	$p \rightarrow q$	si p entonces q sólo si q entonces p p es suficiente para q q es necesaria para p No p a menos que q	Si saco 8 entonces mi promedio aprobatorio Si saco 8 en el parcial tendré el promedio aprobado Para tener el promedio aprobado debe de sacar 8
Bicondicional doble implicación	$p \leftrightarrow q$	p si y solo si q p necesario y suficiente para q	Voy de vacaciones si y solo si apruebo todas mis materias

4. Lenguajes y modelos lógicos

Modelo lógico: LOGICA PREPOSICIONAL

GRAMATICA

Sentencia \rightarrow Sentencia Atómica | Sentencia Compleja

Sentencia Atómica \rightarrow Verdadero | Falso | Símbolo

Símbolo \rightarrow P | Q | R | ...

Sentencia Compleja \rightarrow \neg Sentencia
| (Sentencia \wedge Sentencia)
| (Sentencia \vee Sentencia)
| (Sentencia \Rightarrow Sentencia)
| (Sentencia \Leftrightarrow Sentencia)

Operadores \rightarrow \neg (no), \wedge (y), \vee (o), \Rightarrow (implica), \Leftrightarrow (si y solo si)

A semejanza de la **Notación BNF**
(Backus Naur-Form)

BNF = metalenguaje

Metalenguaje: se le usa para describir otro lenguaje, es una nomenclatura que nos permite una descripción compacta y precisa de los constructores sintácticos usando ciertos símbolos y reglas.

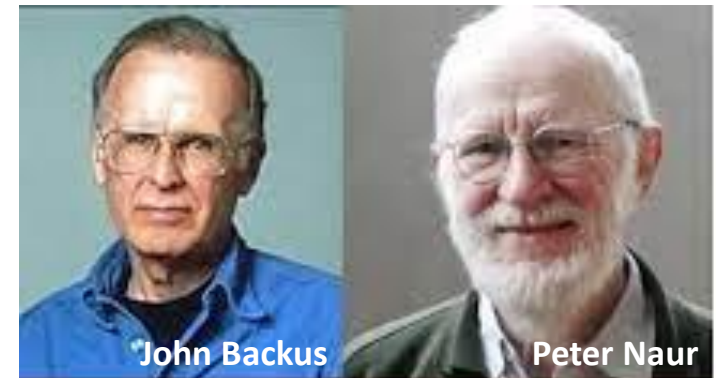
4. Lenguajes y modelos lógicos

Modelo lógico: LOGICA PREPOSICIONAL

GRAMATICA

Notación BNF (Backus Naur-Form)

- En 1959, la notación BNF fue desarrollada para representar la gramática del **Algol 60** por **John Backus** (Estados Unidos) y **Peter Naur** (Dinamarca).
- **Algol 60** es un lenguaje de "punto y coma como separador". Otros lenguajes similares son Algol 68 y Pascal. Esto difiere con lenguajes como C y PHP en los que el punto y coma son terminadores de instrucciones.



```
<expression> ::= <expression> + <term>
                | <expression> - <term>
                | <term>
<term>         ::= <term> * <factor>
                | <term> / <factor>
                | <factor>
<factor>       ::= number
                | name
```

4. Lenguajes y modelos lógicos

Modelo lógico: LOGICA PREPOSICIONAL

GRAMATICA

Sentencias válidas

P

Verdadero

$P \wedge Q$

$(P \vee Q) \Rightarrow S$

$(P \wedge Q) \vee R \Rightarrow S$

$\neg(P \vee Q)$

$\neg(P \vee Q) \Rightarrow R \wedge S$

Implicación Lógica (\Rightarrow)

$P \Rightarrow Q$

Si P es verdadero entonces Q también es verdadero

Ejemplo:

Si está lloviendo entonces las calles están mojadas.

Equivalencia Lógica (\Leftrightarrow)

$P \Leftrightarrow Q$

Si P es verdadero entonces Q también es verdadero.
Si Q es verdadero entonces P también es verdadero.

Ejemplo:

Si los dos lados de un triángulo son iguales entonces los dos ángulos de la base del triángulo son iguales

Es equivalente a dos sentencias de implicación:

$(P \Rightarrow Q) \wedge (Q \Rightarrow P)$

4. Lenguajes y modelos lógicos

Modelo lógico: LOGICA PREPOSICIONAL

EQUIVALENCIA (\Leftrightarrow)

Dos sentencias α y β son lógicamente equivalentes ($\alpha \Leftrightarrow \beta$) si tienen los mismos valores de verdad en el mismo conjunto de modelos:

$$\alpha \equiv \beta \text{ si y solo si } \alpha \models \beta \text{ y } \beta \models \alpha$$



$$\begin{aligned}(\alpha \wedge \beta) &\equiv (\beta \wedge \alpha) \\ (\alpha \vee \beta) &\equiv (\beta \vee \alpha) \\ (\alpha \wedge \beta) \wedge \gamma &\equiv \alpha \wedge (\beta \wedge \gamma) \\ (\alpha \vee \beta) \vee \gamma &\equiv \alpha \vee (\beta \vee \gamma) \\ \neg \neg \alpha &\equiv \alpha \\ (\alpha \Rightarrow \beta) &\equiv (\neg \beta \Rightarrow \neg \alpha) \\ (\alpha \Rightarrow \beta) &\equiv (\neg \alpha \vee \beta) \\ (\alpha \Leftrightarrow \beta) &\equiv ((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)) \\ \neg(\alpha \wedge \beta) &\equiv (\neg \alpha \vee \neg \beta) \\ \neg(\alpha \vee \beta) &\equiv (\neg \alpha \wedge \neg \beta) \\ (\alpha \wedge (\beta \vee \gamma)) &\equiv ((\alpha \wedge \beta) \vee (\alpha \wedge \gamma)) \\ (\alpha \vee (\beta \wedge \gamma)) &\equiv ((\alpha \vee \beta) \wedge (\alpha \vee \gamma))\end{aligned}$$

Conmutatividad de \wedge
Conmutatividad de \vee
Asociatividad de \wedge
Asociatividad de \vee
Eliminación de la doble negación
Contraposición
Eliminación de la implicación
Eliminación de la bicondicional
Ley de Morgan
Ley de Morgan
Distribución de \wedge respecto a \vee
Distribución de \vee respecto a \wedge

Nota: $\alpha \models \beta$ significa que β es verdadero en todas las estructuras en las que α es verdadero.

4. Lenguajes y modelos lógicos

Modelo lógico: LOGICA PREPOSICIONAL

VALIDEZ

- Una sentencia es válida si es verdadera en todos los modelos.
- Las sentencias válidas también se conocen como **tautologías**.

Teorema de la deducción

$\alpha \models \beta$ si y solo si la sentencia $(\alpha \Rightarrow \beta)$ es válida

Nota: $\alpha \models \beta$ significa que β es verdadero en todas las estructuras en las que α es verdadero.

4. Lenguajes y modelos lógicos

Modelo lógico: LOGICA PREPOSICIONAL

PRUEBA LOGICA

Es la aplicación de una secuencia de reglas de inferencias para derivar a una conclusión.

Reglas de Inferencia

¿Qué es una prueba lógica?



1. Modus Ponendo Ponens (PP)

$$\begin{array}{l} p \Rightarrow q \\ p \\ \hline \therefore q \end{array} \quad \begin{array}{l} p \rightarrow \sim q \\ p \\ \hline \therefore \sim q \end{array} \quad \begin{array}{l} \sim p \rightarrow q \\ \sim p \\ \hline \therefore q \end{array}$$

2. Modus tollendo tollens (TT)

$$\begin{array}{l} p \rightarrow q \\ \sim q \\ \hline \therefore \sim p \end{array} \quad \begin{array}{l} \neg p \rightarrow \neg q \\ q \\ \hline \therefore p \end{array}$$

3. Silogismo hipotético (SH)

$$\begin{array}{l} p \rightarrow q \\ q \rightarrow r \\ \hline \therefore p \Rightarrow r \end{array}$$

4. Tollendo ponens (TP)

$$\begin{array}{l} p \vee q \\ \sim q \\ \hline \therefore p \end{array} \quad \circ \quad \begin{array}{l} p \vee q \\ \sim p \\ \hline \therefore q \end{array}$$

5. Doble negación (DN)

$$\frac{\neg(\neg p)}{p}$$

6. Adjunción y simplificación

Adjunción (A)

$$\begin{array}{l} p \quad (\text{premisa}) \\ q \quad (\text{premisa}) \\ \hline \therefore p \wedge q \end{array}$$

Simplificación (S)

$$\frac{p \wedge q}{\therefore p} \quad \circ \quad \frac{p \wedge q}{\therefore q}$$

7. ley de la adición (LA)

$$\frac{p \quad (\text{premisa})}{\therefore p \vee q}$$

8. Silogismo disyuntivo (DS)

$$\begin{array}{l} p \vee q \\ p \rightarrow r \\ q \rightarrow s \\ \hline \therefore r \vee s \end{array}$$

OTRAS LEYES LOGICAS DE APOYO

10. Prueba del condicional

$$\frac{p \rightarrow q \quad p \quad (\text{se anexa})}{\text{se debe concluir } q}$$

11. Ley de absorción (abs.)

$$\frac{p \rightarrow q}{\therefore p \rightarrow (q \wedge p)}$$

12. Ley de contraposición (contraria)

$$\bullet \quad p \rightarrow q \equiv \neg q \rightarrow \neg p$$

13. Ley del condicional

$$\bullet \quad p \rightarrow q \equiv \neg p \vee q$$

14. Ley de exportación (LE)

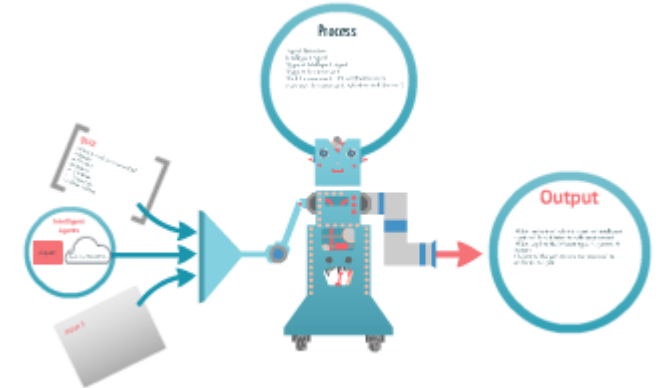
$$\bullet \quad [(p \wedge q) \rightarrow r] \equiv [p \rightarrow (q \rightarrow r)]$$

4. Lenguajes y modelos lógicos

Modelo lógico: LOGICA PREPOSICIONAL

LIMITACIONES

- La lógica proposicional es muy simple para representar algunos problemas del mundo real.
- En problemas complejos puede ser necesario usar un número muy grande de sentencias para crear un agente realmente inteligente.



5. Ejemplos de Aplicación

(1) LEYES DE MORGAN

- Son dos leyes que lógicas muy útiles cuando se quiere encontrar equivalentes para proposiciones que se obtienen por negación de proposiciones compuestas.



Augustus de Morgan

1. La negación de la conjunción es la disyunción de las negaciones.

$$\neg(\alpha \wedge \beta) \equiv (\neg\alpha \vee \neg\beta)$$

2. La negación de la disyunción es la conjunción de las negaciones.

$$\neg(\alpha \vee \beta) \equiv (\neg\alpha \wedge \neg\beta)$$

Ejemplo:

Escribir en castellano razonable la negación de la proposición $P + Q$, siendo:

P: "El numero 15 es par"

Q: "Hay un numero que, cuando se añade a 6, da una suma de 13"

$$\neg(P \vee Q) \equiv (\neg P \wedge \neg Q) = \text{"El numero 15 no es par y no hay un numero que, cuando se añade a 6, da una suma de 13"}$$

5. Ejemplos de Aplicación

(2) TAUTOLOGIA

Una forma proposicional que siempre es verdadera se llama **tautología**.

Considere $(p \vee q) \vee ((\neg p) \wedge (\neg q))$. Utilice la tabla de verdad siguiente:

p q	$\neg p$	$\neg q$	$p \vee q$	$\neg (p) \wedge (\neg q)$	$(p \vee q) \vee ((\neg p) \wedge (\neg q))$
T T	F	F	T	F	T
T F	F	T	T	F	T
F T	T	F	T	F	T
F F	T	T	F	T	T

5. Ejemplos de Aplicación

(3) CONTRADICCION

Una forma proposicional que siempre es FALSA se llama **contradicción**.

Considere $(p \wedge q) \wedge ((\neg p) \vee (\neg q))$. Utilice la tabla de verdad siguiente:

p q	$\neg p$	$\neg q$	$p \wedge q$	$\neg (p) \vee (\neg q)$	$(p \wedge q) \wedge ((\neg p) \vee (\neg q))$
T T	F	F	T	F	F
T F	F	T	F	T	F
F T	T	F	F	T	F
F F	T	T	F	T	F

PREGUNTAS

Dudas y opiniones