

Tarea 3 (inferencia)

Materia: Sistemas Expertos

Juan Pablo Rizo Riestra 22110371

7E

Profesor: Mauricio Alejandro Cabrera Arellano

Jalisco CETI Colomos 19/03/2025



Motores de inferencia de la actualidad

Un motor de inferencia es el componente central de un sistema experto, encargado de razonar sobre el conocimiento almacenado en una base de conocimiento para llegar a conclusiones. Hoy en día, estos motores han evolucionado y se utilizan en diversos campos, como la inteligencia artificial, el aprendizaje automático y la toma de decisiones automatizada.

Es un software que aplica reglas lógicas y probabilísticas para deducir nueva información a partir de hechos y reglas preexistentes en una base de conocimiento.

◆ Funcionamiento

1. Entrada de datos: Recibe hechos o datos iniciales del usuario o del entorno.
2. Procesamiento: Aplica reglas de inferencia para deducir nueva información.
3. Salida: Presenta conclusiones o decisiones basadas en el razonamiento.

Los motores de inferencia pueden usar dos enfoques:

- Encadenamiento hacia adelante (Forward Chaining): parte de los datos y aplica reglas hasta llegar a una conclusión.
- Encadenamiento hacia atrás (Backward Chaining): parte de una hipótesis y busca pruebas para confirmarla.

Motores de Inferencia Modernos

◆ 1. CLIPS (C Language Integrated Production System)

- Tipo: Basado en reglas.
- Usos: Sistemas expertos en industria, defensa y automatización.
- Ventaja: Ligero y eficiente, con reglas expresadas en un lenguaje declarativo.

◆ 2. Prolog

- Tipo: Basado en lógica de predicados.

- Usos: IA, procesamiento del lenguaje natural, planificación.
 - Ventaja: Potente para resolver problemas complejos de inferencia lógica.
- ◆ 3. Drools (JBoss Rules)
- Tipo: Basado en reglas (Business Rules Management System - BRMS).
 - Usos: Sistemas empresariales, automatización de decisiones, industria financiera.
 - Ventaja: Integración con Java y escalabilidad en sistemas grandes.
- ◆ 4. TensorFlow y PyTorch (Motores de Inferencia en Aprendizaje Profundo)
- Tipo: Basado en redes neuronales y aprendizaje automático.
 - Usos: IA avanzada, procesamiento de imágenes, predicción en tiempo real.
 - Ventaja: Capacidad de manejar grandes volúmenes de datos y aprendizaje automático.
- ◆ 5. OpenAI GPT (Generative Pre-trained Transformer)
- Tipo: Basado en aprendizaje profundo y procesamiento de lenguaje natural (NLP).
 - Usos: Chatbots, generación de texto, asistentes virtuales.
 - Ventaja: Capacidad de generar respuestas inteligentes con razonamiento probabilístico.

Métodos de Inferencia en Sistemas Expertos

Los métodos de inferencia son estrategias utilizadas en los motores de inferencia para deducir nuevos conocimientos a partir de una base de reglas y hechos. Estos métodos permiten a un sistema experto razonar de manera automática y llegar a conclusiones fundamentadas.

1. Inferencia Basada en Reglas

Se basa en aplicar reglas del tipo "SI condición → ENTONCES acción" para deducir nueva información. Existen dos enfoques principales:

◆ 1.1. Encadenamiento hacia Adelante (Forward Chaining)

- Parte de los datos conocidos y aplica reglas hasta llegar a una conclusión.
- Se usa cuando se tienen hechos iniciales y se quiere descubrir nuevas consecuencias.
- Utilizado en sistemas de diagnóstico y planificación.

◆ 1.2. Encadenamiento hacia Atrás (Backward Chaining)

- Parte de una hipótesis o meta y busca pruebas en la base de conocimientos para confirmarla.
- Se usa cuando se tiene una pregunta específica y se busca una justificación.
- Utilizado en sistemas de diagnóstico médico y detección de fraudes.

2. Inferencia Basada en Redes Bayesianas

Este método usa probabilidades para razonar en presencia de incertidumbre. Se basa en la teoría de Bayes, permitiendo calcular la probabilidad de un evento en función de datos observados.

3. Inferencia Difusa (Lógica Difusa)

Este método maneja la incertidumbre y la ambigüedad en el conocimiento.

- Permite razonar con grados de verdad en lugar de valores absolutos (0 o 1).
- Se usa en controladores difusos, climatización inteligente, vehículos autónomos.

Modus Ponens y Modus Tollens en Inferencia Lógica

En lógica proposicional, **Modus Ponens** y **Modus Tollens** son reglas de inferencia fundamentales para razonar y llegar a conclusiones a partir de premisas. Son utilizadas en sistemas expertos, inteligencia artificial y programación lógica.

1. Modus Ponens (MP) - "Modo de Afirmación"

Si tenemos una implicación $P \rightarrow Q$ y sabemos que **P es verdadera**, entonces podemos concluir que **Q también es verdadera**.

Forma lógica:

- **Premisa 1:** $P \rightarrow Q$ (Si P, entonces Q)
- **Premisa 2:** P (P es verdadero)
- **Conclusión:** Q (Entonces Q es verdadero)

Ejemplo real:

- **Si llueve, entonces el suelo estará mojado.**
- **Está lloviendo.**
- **Conclusión:** El suelo está mojado.

Ejemplo en Python:

```
def modus_ponens(p, p_implica_q):  
    """Aplica Modus Ponens: si P es verdadero y  $P \rightarrow Q$  es verdadero, entonces Q es verdadero."""  
    if p and p_implica_q:  
        return True # Q es verdadero  
    return False # No se puede concluir Q  
  
# Ejemplo: Si llueve (P), entonces el suelo está mojado (Q)  
llueve = True # P  
implicacion = True #  $P \rightarrow Q$   
  
resultado = modus_ponens(llueve, implicacion)  
print(f"¿El suelo está mojado? {resultado}") # True
```

2. Modus Tollens (MT) - "Modo de Negación"

Si tenemos una implicación $P \rightarrow Q$ y sabemos que **Q es falsa**, entonces podemos concluir que **P también es falsa**.

Forma lógica:

- **Premisa 1:** $P \rightarrow Q$ (Si P, entonces Q)
- **Premisa 2:** $\neg Q$ (Q es falso)
- **Conclusión:** $\neg P$ (Entonces P es falso)

Ejemplo real:

- **Si el coche tiene gasolina, entonces puede arrancar.**
- **El coche no arranca.**
- **Conclusión:** No tiene gasolina.

Ejemplo en Python:

```
def modus_tollens(q, p_implica_q):
    """Aplica Modus Tollens: si Q es falso y  $P \rightarrow Q$  es verdadero, entonces P es falso."""
    if not q and p_implica_q:
        return False # P es falso
    return True # No se puede concluir que P es falso

# Ejemplo: Si el coche tiene gasolina (P), entonces puede arrancar (Q)
puede_arrancar = False # Q es falso
implicacion = True #  $P \rightarrow Q$ 

resultado = modus_tollens(puede_arrancar, implicacion)
print(f"¿El coche tiene gasolina? {resultado}") # False
```

Diferencias Clave

Regla	Premisas	Conclusión
Modus Ponens	$P \rightarrow Q, P$	Q
Modus Tollens	$P \rightarrow Q, \neg Q$	$\neg P$

Ambos son esenciales en inferencia lógica, usados en sistemas expertos, IA y pruebas matemáticas.

Conclusión

- Modus Ponens se usa cuando sabemos que P es verdadero y queremos concluir que Q también lo es.
- Modus Tollens se usa cuando sabemos que Q es falso y queremos concluir que P también lo es.
- Estas reglas son esenciales en razonamiento lógico, matemáticas, IA y sistemas expertos.