



MATERIA
SISTEMAS EXPERTOS

“Proyecto”

Stephanie Vianney Aguila Rivera 22310212

PROF. Mauricio Alejandro Cabrera Arellano

Ingeniería en Mecatrónica

Grupo 7F

08 de Diciembre de 2025

Introducción

El Sistema Experto de Recomendación Literaria (SERL) es una aplicación que emula el razonamiento de un bibliotecario o asesor experto para sugerir lecturas personalizadas. Este sistema se basa en la arquitectura tradicional de la Inteligencia Artificial (IA) simbólica, utilizando reglas de inferencia y Factores de Certeza (FC).

Objetivo

El SERL busca ofrecer una o dos recomendaciones de libros con la **mayor afinidad posible** a las preferencias declaradas por el usuario, mediante la ponderación de sus elecciones frente a una base de conocimiento especializada.

Justificación del Enfoque

La selección de un libro puede ser un proceso complejo. Un sistema experto puede actuar como un bibliotecario o asesor que aplica el conocimiento literario. Nuestro **Sistema Experto Literario** se alimenta con las características de 30 libros (género, tema, estilo, estructura narrativa). Cuando un usuario introduce sus preferencias ("Me gusta la ciencia ficción, pero no las distopías, prefiero los personajes femeninos fuertes"), el sistema infiere y recomienda un libro específico que cumpla con esas reglas. Este enfoque en reglas es fundamental, ya que los sistemas de recomendación avanzados (aunque hoy usan Machine Learning) comenzaron con estructuras de **sistemas basados en reglas**.

Arquitectura y Componentes

El SERL se compone de tres elementos clave que interactúan para producir la recomendación final.

A. Base de Hechos (Base de Conocimiento)

La Base de Hechos contiene los datos estáticos sobre los que el motor de inferencia trabajará.

- Implementación: Archivo [base_conocimiento.json](#).
- Contenido: 30 registros de libros, cada uno con los siguientes atributos utilizados para la clasificación:
 - Título, Autor, Ruta_Imagen.

- Atributos de Inferencia: **Género, Ritmo, Complejidad, Motivación, Compromiso.**
- **Rating_Base:** Un puntaje inicial (ej. el rating promedio de la crítica) que sirve como base para la acumulación de la afinidad.

B. Base de Reglas (Conocimiento Experto)

La Base de Reglas codifica el conocimiento del experto (el bibliotecario) sobre cómo las preferencias del usuario se relacionan con los atributos de los libros.

- Implementación: Clases **ReglaInferencia** en **modelo_conocimiento.py** y su carga en **motor_inferencia.py**.
- Estructura: Cada regla sigue la sintaxis SI (Antecedente) ENTONCES (Consecuente) CON FC.
 - Ejemplo de Regla: SI el usuario elige *Motivación_Emocional* ENTONCES el libro debe tener el atributo *Emocional* CON FC 0.80.

C. Motor de Inferencia

El Motor de Inferencia es el núcleo de razonamiento que evalúa la compatibilidad de cada libro con las respuestas del usuario.

- Implementación: Método **inferir_recomendaciones()** en **motor_inferencia.py**.
- Mecanismo: Encadenamiento Hacia Adelante (Forward Chaining) y Ponderación.
 1. El motor toma las 5 respuestas del usuario como hechos de entrada.
 2. Por cada respuesta, recorre la Base de Reglas y activa las reglas cuyo antecedente (respuesta del usuario) coincide.
 3. El Factor de Certeza (FC) de la regla activada se suma al puntaje total del libro que cumple con el consecuente (el atributo del libro).
 4. El libro con el puntaje acumulado más alto es la recomendación final.

Demostración y Funcionamiento

A continuación, se detalla una corrida de ejemplo, demostrando la interacción del usuario y la lógica de inferencia del sistema.

1. Interfaz de Usuario (UI)

El usuario navega a través de 5 preguntas utilizando la interfaz gráfica construida con `customtkinter` (`main.py`).

Pregunta	Respuesta del Usuario (Hecho de Entrada)
P1	Ciencia Ficción
P2	Ritmo Lento
P3	Complejidad Alta
P4	Motivación Evadir
P5	Compromiso Largo

Demostración y Funcionamiento

A continuación, se detalla una corrida de ejemplo, demostrando la interacción del usuario y la lógica de inferencia del sistema.

1. Interfaz de Usuario (UI)

El usuario navega a través de 5 preguntas utilizando la interfaz gráfica construida con `customtkinter` (`main.py`).

Pregunta	Respuesta del Usuario (Hecho de Entrada)
P1	Ciencia Ficción
P2	Ritmo Lento
P3	Complejidad Alta
P4	Motivación Evadir
P5	Compromiso Largo

2. Trazabilidad de la Inferencia (Forward Chaining)

El Motor de Inferencia procesa las 5 respuestas. A continuación, se muestra la traza de cómo se activan las reglas y se actualiza el puntaje de afinidad de los libros:

Regla Activada (Respuesta)	Atributo Buscado (Consecuente)	FC	Libro afectado	Puntaje Inicial (Rating_Base)	Puntaje Acumulado
Ciencia Ficción	Ciencia Ficción	0.95	Dune	4.70	5.65
Ritmo Lento	Lento	0.80	Dune	5.65	6.45
Complejidad Alta	Alta	0.90	Dune	6.45	7.35
Motivación Evadir	Evadir	0.70	Dune	7.35	8.05
Compromiso Largo	Largo	0.60	Dune	8.05	8.65
...	Otros Libros

3. Resultados Finales

El motor finaliza el proceso, ordena los puntajes de afinidad y presenta los dos libros mejor clasificados al usuario.

Resultado de la Recomendación:

1. Libro Mejor Clasificado: [Título del libro con el puntaje más alto, por ejemplo: Dune]
 - Afinidad Total: [Puntaje final, por ejemplo: 8.65]
2. Segunda Opción: [Título del segundo libro]
 - Afinidad Total: [Puntaje final]

Conclusión

El Sistema Experto de Recomendación Literaria (SERL) demuestra la aplicación efectiva de la Inteligencia Artificial Simbólica para la toma de decisiones. Al utilizar el Encadenamiento Hacia Adelante y los Factores de Certeza, el sistema logra transformar las preferencias cualitativas del usuario en una recomendación cuantitativa y ponderada, cumpliendo así con el objetivo de actuar como un bibliotecario digital experto.

https://github.com/vianneyaguila/SE_AguilaRivera_7F/tree/main/Proyecto

<https://youtu.be/7yVTi3AaAvQ>