

Práctica No. 7

Un sistema experto en Prolog.

Introducción

¿Qué es un sistema experto?

Los sistemas expertos forman parte de un firme y verdadero avance en la inteligencia artificial simbólica. Es el ejemplo más exitoso de un Sistema Basado en Conocimiento. Los sistemas expertos pueden incorporar el “expertise” de algún experto humano de un campo específico en forma de miles de reglas. Para una persona sería una experiencia casi "traumática" el realizar una búsqueda de reglas posibles al completado de un problema y concordar estas con las posibles consecuencias, mientras que se sigue en un papel los trazos de un árbol de búsqueda. Los sistemas expertos realizan amablemente esta tarea; mientras que la persona responde a las preguntas formuladas por el sistema experto, este busca recorriendo las ramas más interesantes del árbol, hasta dar con la respuesta afín al problema, o en su falta, la más parecida a esta. Los sistemas expertos tienen la ventaja frente a otros tipos de programas de Inteligencia Artificial, de proporcionar gran flexibilidad a la hora de incorporar nuevos conocimientos. Para ello solo tenemos que introducir la nueva regla que deseamos hacer constar y ya está, sin necesidad de cambiar el funcionamiento propio del programa. Los sistemas expertos son "auto explicativos", al contrario que en los programas convencionales, en los que el conocimiento como tal está encriptado junto al propio programa en forma de lenguaje de ordenador. Los expertos de I.A. dicen que los sistemas expertos tienen un conocimiento declarativo, mientras que los demás programas tienen un conocimiento procedural.

Prolog puede ser visto como un lenguaje prescriptivo dado que un programa en Prolog consiste de un conjunto de cláusulas, donde cada cláusula es, o un hecho o una regla acerca de como puede inferirse alguna solución para los hechos dados. Además, Prolog tiene implementada cierta “inteligencia” para resolver el problema (esta “inteligencia” es un mecanismo de razonamiento automático llamado “motor de inferencia” basado en un algoritmo que se conoce como resolución). Estas características de Prolog hacen que sea relativamente sencillo desarrollar un pequeño sistema experto en este lenguaje. El sistema experto que usaremos en esta práctica consta de 2 partes: Un “shell” del sistema experto, que se encarga de interactuar con el usuario y manipular la Base de Conocimientos del sistema y una base de conocimientos, con reglas del tipo: (situación1, [condicion1, condicion2, ..., condicionN]). Este sistema experto puede justificar su diagnóstico y explicar porqué hace ciertas preguntas.

Actividades

1.- Captura el “shell” del sistema experto y guarda el archivo con el nombre **experto.pl**

```
/*      Sistema Experto: experto.pl
      Trata los sintomas como una lista. La cabeza es el diagnostico y la "cola" son
      los sintomas.
      Utiliza assert/1 para cambiar dinamicamente la base de conocimientos.
      Determina la verdad y falsedad de los sintomas conocidos.
      Puede contestar a las preguntas 'porque' e incluye capacidad de explicacion.
      Elimina dinamicamente las aseveraciones agregadas despues de cada consulta.
*/
:- dynamic conocido/1.
consulta:-
    haz_diagnostico(X),
    escribe_diagnostico(X),
    ofrece_explicacion_diagnostico(X),
    clean_scratchpad.
consulta:-
    write('No hay suficiente conocimiento para elaborar un diagnostico.'),
    clean_scratchpad.

haz_diagnostico(Diagnosis):-
    obten_hipotesis_y_sintomas(Diagnosis, ListaDeSintomas),
```

```

prueba_presencia_de(Diagnosis, ListaDeSintomas).

obten_hipotesis_y_sintomas(Diagnosis, ListaDeSintomas):-
    conocimiento(Diagnosis, ListaDeSintomas).

prueba_presencia_de(Diagnosis, []).
prueba_presencia_de(Diagnosis, [Head | Tail]):- prueba_verdad_de(Diagnosis, Head),
    prueba_presencia_de(Diagnosis, Tail).

prueba_verdad_de(Diagnosis, Sintoma):- conocido(Sintoma).
prueba_verdad_de(Diagnosis, Sintoma):- not(conocido(is_false(Sintoma))),
    pregunta_sobre(Diagnosis, Sintoma, Reply), Reply = si.

pregunta_sobre(Diagnosis, Sintoma, Reply):- write('Es verdad que '),
    write(Sintoma), write('? '),
    read(Respuesta), process(Diagnosis, Sintoma, Respuesta, Reply).

process(Diagnosis, Sintoma, si, si):- asserta(conocido(Sintoma)).
process(Diagnosis, Sintoma, no, no):- asserta(conocido(is_false(Sintoma))).
process(Diagnosis, Sintoma, porque, Reply):- nl,
    write('Estoy investigando la hipotesis siguiente: '),
    write(Diagnosis), write('.'), nl, write('Para esto necesito saber si '),
    write(Sintoma), write('.'), nl, pregunta_sobre(Diagnosis, Sintoma, Reply).
process(Diagnosis, Sintoma, Respuesta, Reply):- Respuesta \== no,
    Respuesta \== si, Respuesta \== porque, nl,
    write('Debes contestar si, no o porque.'), nl,
    pregunta_sobre(Diagnosis, Sintoma, Reply).

escribe_diagnostico(Diagnosis):- write('El diagnostico es '),
    write(Diagnosis), write('.'), nl.

ofrece_explicacion_diagnostico(Diagnosis):-
    pregunta_si_necesita_explicacion(Respuesta),
    actua_consecuentemente(Diagnosis, Respuesta).

pregunta_si_necesita_explicacion(Respuesta):-
    write('Quieres que justifique este diagnostico? '),
    read(RespuestaUsuario),
    asegura_respuesta_si_o_no(RespuestaUsuario, Respuesta).

asegura_respuesta_si_o_no(si, si).
asegura_respuesta_si_o_no(no, no).
asegura_respuesta_si_o_no(_, Respuesta):- write('Debes contestar si o no.'),
    pregunta_si_necesita_explicacion(Respuesta).

actua_consecuentemente(Diagnosis, no).
actua_consecuentemente(Diagnosis, si):- conocimiento(Diagnosis, ListaDeSintomas),
    write('Se determino este diagnostico porque se encontraron los siguientes
sintomas: '), nl,
    escribe_lista_de_sintomas(ListaDeSintomas).

escribe_lista_de_sintomas([]).
escribe_lista_de_sintomas([Head | Tail]):-
    write(Head), nl, escribe_lista_de_sintomas(Tail).

clean_scratchpad:- retract(conocido(X)), fail.
clean_scratchpad.
conocido(_):- fail.
not(X):- X,!,fail.
not(_).

```

2.- Para poder utilizar este shell debemos definir una base de conocimientos. Vamos a definir la primera base de conocimientos para un sistema de diagnóstico médico. Captura las siguientes reglas y salva el archivo con el nombre BaseConocimientos1.pl:

```
/* BaseConocimientos1: diagnosticos y sintomas
   Dominio: diagnostico medico. Trata los sintomas como una lista multi-elementos
*/
conocimiento('sarampion',
             ['el paciente esta cubierto de puntos', 'el paciente tiene temperatura alta',
              'el paciente tiene ojos rojos','el paciente tiene tos seca']).

conocimiento('influenza',
             ['el paciente tiene dolor en las articulaciones', 'el paciente tiene mucho
              estornudo','el paciente tiene dolor de cabeza']).

conocimiento('malaria',
             ['el paciente tiene temperatura alta','el paciente tiene dolor en las
              articulaciones', 'el paciente tiembla violentamente', 'el paciente tiene
              escalofrios']).

conocimiento('gripe',
             ['el paciente tiene cuerpo cortado', 'el paciente tiene dolor de cabeza', 'el
              paciente tiene temperatura alta']).

conocimiento('tifoidea',
             ['el paciente tiene falta de apetito', 'el paciente tiene temperatura alta', 'el
              paciente tiene dolor abdominal', 'el paciente tiene dolor de cabeza', 'el paciente
              tiene diarrea']).
```

3.- Para poder utilizar el sistema experto con esta base de conocimientos, dentro de Prolog debemos teclear:

```
consult('experto.pl').
consult('BaseConocimientos1.pl').
```

3.1.- Para poder determinar una enfermedad, se teclea `consulta.<enter>` y se contestan las preguntas que va haciendo el sistema. En cualquier momento se puede constestar con un **porque** en lugar de **si o no**. Entonces el sistema explicará cuál es el diagnóstico que trata de probar. Al proporcionar un diagnóstico, el sistema preguntará si se quiere una justificación para su diagnóstico. Si se contesta afirmativamente el sistema mostrará los hechos que lo llevaron al diagnóstico presentado.

3.2.- Usando el sistema experto, trata de determinar las enfermedad de las personas que cursan los siguientes sintomas:

i) temperatura alta, dolor de cabeza y cuerpo cortado.

 Diagnostico: _____

 Justificación: _____

ii) dolor en las articulaciones, temblor violento y escalofrios.

 Diagnostico: _____

 Justificación: _____

iii) dolor de cabeza, estornudos y dolor en las articulaciones

 Diagnostico: _____

 Justificación: _____

4.- Considera la siguiente base de conocimientos para un sistema de diagnóstico automotriz:

Diagnóstico: Banda del alternador defectuosa
Síntomas: Tiene grietas, está cristalizada, y esta floje.

Diagnóstico: Banda del alternado defectuosa
Síntomas: La banda está engrasada y rechina al girar.

Diagnóstico: Batería defectuosa
Síntomas: Luces y ventilador encendidos y Motor apagado y Voltaje de batería menor a 10.5Volts.

Diagnóstico: Regulador defectuoso
Síntomas: Si motor en marcha y luces encendidas y se intensifican al acelerar.

Diagnóstico: Alternador defectuoso
Síntomas: Si la batería esta en buenas condiciones y luz de advertencia permanece encendida y motor de arranque gira lentamente.

Diagnóstico: Sistema de frenos defectuoso
Síntomas: Pedal del freno duro y al frenar se produce un tirón lateral y ruedas rechinan al frenar.

4.1.- Representa esta base de conocimientos en Prolog y captúralos en un archivo llamado
BaseConocimientos2.pl Base capturada: OK ____

4.2.- Desde SWI-Prolog, ahora consulta la BaseConocimientos2.pl Base cargada en Prolog: OK ____

4.3.- Realiza 4 consultas al sistema de diagnóstico automotriz, reporta las consultas que hiciste y el resultado de la consulta.

Consulta 1: _____

Diagnóstico 1: _____

Consulta 2: _____

Diagnóstico 2: _____

Consulta 3: _____

Diagnóstico 3: _____

Consulta 4: _____

Diagnóstico 4: _____

5.- Considera ahora la siguiente base de conocimientos para un sistema experto que ayuda a identificar animales:

Diagnóstico: cheeta

Sintómas: Si es mamifero, carnivor, tiene color leonado y puntos negros.

Diagnóstico: tigre

Sintómas: Si es mamifero, es carnivor, tiene color leonado y tiene rayas negras.

Diagnóstico: jirafa

Sintómas: Si es ungulado, tiene cuellos largo y piernas largas.

Diagnóstico: zebra

Sintómas: Si es ungulado y tiene rayas negras.

Diagnóstico: aveztruz

Sintómas: Si es un pájaro, no vuela y tiene cuello largo,

Diagnóstico: pingüino

Sintómas: Es un pájaro, no vuela, sabe nadar y es blanco con negro.

Diagnóstico: albatros

Sintómas: Es un pájaro, aparece en historias marinas y vuela bien.

5.1.- Representa esta base de conocimientos en Prolog y captúralos en un archivo llamado

BaseConocimientos3.pl

Base capturada: OK ____

5.2.- Desde SWI-Prolog, ahora consulta la BaseConocimientos3.pl Base cargada en Prolog: OK ____

5.3.- Realiza 4 consultas al sistema de identificación de animales, reporta las consultas que hiciste y el resultado de la consulta.

Consulta 1: _____

Diagnóstico 1: _____

Consulta 2: _____

Diagnóstico 2: _____

Consulta 3: _____

Diagnóstico 3: _____

Consulta 4: _____

Diagnóstico 4: _____

