

Proyecto N°2

Sistema basado en conocimiento con encadenamiento inverso

Integrantes: Ricardo Garcia R.
Profesor: Claudio Held
Auxiliar: Cristóbal Garrido Cáceres
Leonardo Causa M.
Ayudantes: Jhon Pilataxi

Fecha de realización: 22 de junio de 2022
Fecha de entrega: 22 de junio de 2022
Santiago, Chile

Resumen

En este documento se explica el procedimiento llevado a cabo para la realización de un Sistema Basado en Conocimiento (SBC) que permite comprobar una serie de hipótesis evaluando una base de hechos, base de reglas y consultando al usuario sobre valores de certeza de hipótesis utilizando encadenamiento inverso.

Una vez diseñado el modelo del SBC, se procede a construir una interfaz gráfica que le permita al usuario manejar de mejor las respuestas que le entrega al SBC y mostrar de forma justificada los resultado que entrega.

La realización de este proyecto permite efectivamente cumplir los objetivos de familiarizarse con el diseño e implementación de un SBC, y verificar su efectividad, su maleabilidad y el potencial que tiene en el área de diagnóstico de hipótesis. Además de poner a prueba la capacidad de diseñar e implementar modelos computacionales e interfaces gráficas que le faciliten la tarea al usuario.

Índice de Contenidos

1. Descripción del proyecto	1
2. Resultados	5
3. Salida de consola	5
4. Discusión y conclusiones	10

Índice de Figuras

1. Vista de bienvenida de la interfaz gráfica.	5
2. Vista de la pregunta animal tiene pelo.	7
3. Resultado de Jirafa	7

Índice de Códigos

1. Output de la consola mostrando las reglas utilizadas.	5
--	---

1. Descripción del proyecto

El presente proyecto tiene por objetivo la implementación de un Sistema Basado en el Conocimiento (SBC) con encadenamiento inverso o *Back Chaining*. Un SBC es una base de conocimiento para razonar y resolver problemas complejos. Para estos tipos de sistemas, la interfaz gráfica y la justificación de las conclusiones obtenidas son de extrema importancia, pues el especialista que está usándolo requiere de fundamentos para tomar una decisión y estos deben ser obtenidos a través del SBC. Existen dos tipos de encadenamiento para la inferencia de conclusiones en un sistema experto: Encadenamiento Directo y Encadenamiento Inverso.

- Encadenamiento Directo (Forward Chaining): Consiste en obtener conclusiones a partir de las premisas recorriendo las reglas $R : F \wedge S \rightarrow D$ de izquierda a derecha.
- Encadenamiento Inverso (Inverse Chaining): se trata de determinar el grado de certeza de un diagnóstico D determinado. Se selecciona una regla que tenga a D como consecuente, $R1 : F \wedge S \rightarrow D$. Si F y S pueden ser probados, el grado de certeza de D se calcula mediante R1. A su vez, F y S pueden ser probados por otras reglas, y así sucesivamente, hasta llegar a las premisas que son datos conocidos del problema.

Por otra parte, el sistema consta de distintas variables globales que se actualizan en tiempo real para obtener conclusiones. Estas se describen a continuación:

- Base de Hechos: Es el conjunto de hechos que caracteriza al actual problema, con sus condiciones iniciales (si existen), y sus conclusiones parciales e intermedias. En esta aplicación en particular, se trata de toda proposición a la que se haya asociado un valor de incerteza para el caso en estudio.
- Base de reglas: Es el conjunto de expresiones condicionales que definen el conocimiento en el dominio del problema.
- Conjunto de hipótesis: Son las metas de alto nivel que hay que alcanzar o demostrar.

Inicialmente, la base de hechos se inicializa vacía, pues no se tiene conocimiento a priori del problema. Por su parte, la base de reglas se detalla a continuación:

- Rule R1: animal tiene pelo:
 - Hypothesis: animal es mamífero with 0.8 certain
 - Hypothesis: animal es ave with -1.0 certain
 - Hypothesis: animal es reptil with -1.0 certain
- Rule R2: animal da leche:
 - Hypothesis: animal es mamífero with 1.0
 - certain Hypothesis: animal es ave with -1.0
 - certain Hypothesis: animal es reptil with -1.0 certain
- Rule R3: animal pone huevos, animal tiene piel dura

- Hypothesis: animal es mamífero with -1.0 certain
- Hypothesis: animal es ave with -1.0 certain
- Hypothesis: animal es reptil with 1.0 certain
- Rule R4: animal pone huevos, animal puede volar
 - Hypothesis: animal es ave with 1.0 certain
 - Hypothesis: animal es reptil with -1.0 certain
- Rule R5: animal tiene plumas
 - Hypothesis: animal es mamífero with -1.0 certain
 - Hypothesis: animal es ave with 1.0 certain
 - Hypothesis: animal es reptil with -1.0 certain
- Rule R6: animal come carne
 - Hypothesis: animal es carnívoro with 1.0 certain
- Rule R7: animal tiene garras
 - Hypothesis: animal es carnívoro with 0.8 certain
- Rule R8: animal es mamífero', 'animal tiene pezuñas
 - Hypothesis: animal es ungulado with 1.0 certain
- Rule R9: animal es mamífero, animal es rumiante
 - Hypothesis: animal es ungulado with 0.75 certain
- Rule R10: animal vive con personas
 - Hypothesis: animal es doméstico with 0.9 certain
- Rule R11: animal vive en el zoológico
 - Hypothesis: animal es doméstico with -0.8 certain
- Rule R12: animal es mamífero, animal es carnívoro, animal tiene manchas oscuras
 - Hypothesis: animal es cheetah with 0.9 certain
- Rule R13: animal es mamífero, animal es carnívoro, animal tiene rayas negras
 - Hypothesis: animal es tigre with 0.85 certain
- Rule R14: animal es mamífero, animal es carnívoro, animal es doméstico
 - Hypothesis: animal es perro with 0.9 certain
- Rule R15: animal es reptil, animal es doméstico
 - Hypothesis: animal es tortuga with 0.7 certain

- Rule R16: animal es mamífero, animal es ungulado, animal tiene cuello largo
 - Hypothesis: animal es jirafa with 1.0 certain
- Rule R17: animal es mamífero, animal es ungulado, animal tiene rayas negras
 - Hypothesis: animal es cebra with 0.95 certain
- Rule R18: animal es mamífero, animal puede volar, animal es feo
 - Hypothesis: animal es murciélago with 0.9 certain
- Rule R19: animal es ave, animal vuela bien
 - Hypothesis: animal gaviota with 0.9 certain
- Rule R20: animal es ave, animal corre rápido
 - Hypothesis: animal es avestruz with 1.0 certain
- Rule R21: animal es ave, animal es parlanchín
 - Hypothesis: animal es loro with 0.95 certain
- Rule R22: animal es mamífero, animal es grande, animal es ungulado, animal tiene trompa
 - Hypothesis: animal es elefante with 0.9 certain

Por otra parte, el conjunto de hipótesis de alto nivel que se quieren demostrar son las siguientes:

- Animal es perro
- Animal es murciélago
- Animal es tigre
- Animal es elefante
- Animal es cebra
- Animal es jirafa
- Animal es tortuga
- Animal es cheetah
- Animal es gaviota
- Animal es avestruz
- Animal es loro

Para demostrar las hipótesis de alto nivel, se van evaluando en orden y utilizando las reglas relevantes para demostrarla se efectúa el siguiente algoritmo iterativo:

- Revisar la base de hechos: si se revisa una regla relevante, se revisa la base de hechos para demostrarla. Si es que existen las premisas con suficiente valor de certeza, se aceptan las acciones como hechos.
- Cuestionar la base de reglas: En el caso de que la hipótesis que se quiere demostrar no puede ser demostrada para el caso anterior se revisan las reglas relevantes iterativamente para poder demostrarla.
- Preguntar al usuario: Si la hipótesis no puede ser demostrada por la base de hechos y no existen reglas relevantes para realizar la inferencia, se le pregunta al usuario el valor de certeza.

La forma en que se combinan las reglas para realizar la inferencia se detalla a continuación:

- Conjunción: para calcular el valor de certeza de una regla, se ocupa el mínimo de los valores de certeza de las premisas.
- Propagación: para calcular el valor de certeza de una conclusión, se realiza el producto entre el valor de certeza la premisa por el de la conclusión.
- Disyunción: Si es que se obtiene más de un valor de certeza para una conclusión, se utiliza el valor máximo de las reglas gatilladas.

Los parámetros de control utilizados en la inferencia son:

- α : parámetro para determinar cuando una hipótesis de alto nivel ha sido demostrada en forma satisfactoria, de manera que no sea necesario analizar otra hipótesis.
- β : parámetro para determinar cuando un hecho (o su negación) tiene un grado de certidumbre suficiente como para ser útil en una inferencia.
- γ : parámetro para determinar cuando un hecho (o su negación) tiene un grado de certidumbre suficiente como para no requerir mayor esfuerzo para mejorar su grado de certidumbre.
- δ : parámetro para determinar cuando una premisa tiene un grado de certidumbre suficiente como para gatillar un regla. $\delta(regla) \in [\beta, \frac{\beta}{\epsilon}]$.
- ϵ : parámetro para determinar cuando una regla tiene un grado de certidumbre suficiente como para inferir una hipótesis (o conclusión intermedia) dada, suponiendo que la premisa esté totalmente satisfecha.

Los valores utilizados para los parámetros son: $\alpha = 0.7$, $\beta = 0.2$, $\gamma = 0.85$, $\epsilon = 0.5$ y $\delta = \frac{0.2}{vc(regla)}$.

2. Resultados

El sistema que se desarrolló contempla una interfaz gráfica, en la cual se tiene una ventana de bienvenida que le permite al usuario configurar los valores de los parámetros a utilizar para la inferencia. Esta vista se muestra en la Figura 1. Además, otro opcional que se programó fue compilar las reglas y mostrar las conclusiones que tienen. Esto se muestra en consola cuando se inicia el programa y el resultado se aprecia en el Código 1.

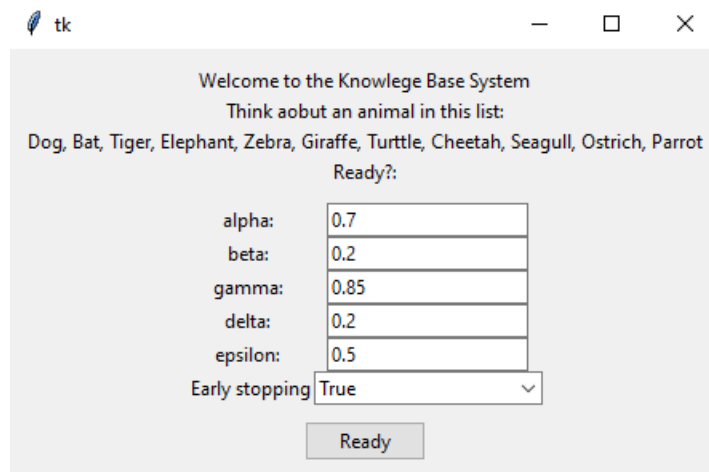


Figura 1: Vista de bienvenida de la interfaz gráfica.

3. Salida de consola

Código 1: Output de la consola mostrando las reglas utilizadas.

```

1 Rule R1:
2   ['animal tiene pelo']
3   [Hypothesis: animal es mamífero with 0.8 certain, Hypothesis: animal es ave with -1.0
   ↪ certain, Hypothesis: animal es reptil with -1.0 certain]
4 Rule R2:
5   ['animal da leche']
6   [Hypothesis: animal es mamífero with 1.0 certain, Hypothesis: animal es ave with -1.0
   ↪ certain, Hypothesis: animal es reptil with -1.0 certain]
7 Rule R3:
8   ['animal pone huevos', 'animal tiene piel dura']
9   [Hypothesis: animal es mamífero with -1.0 certain, Hypothesis: animal es ave with -1.0
   ↪ certain, Hypothesis: animal es reptil with 1.0 certain]
10 Rule R4:
11  ['animal pone huevos', 'animal puede volar']
12  [Hypothesis: animal es ave with 1.0 certain, Hypothesis: animal es reptil with -1.0 certain
   ↪ ]
13 Rule R5:
14  ['animal tiene plumas']

```



```
15      [Hypothesis: animal es mamífero with -1.0 certain, Hypothesis: animal es ave with 1.0
      ↪ certain, Hypothesis: animal es reptil with -1.0 certain]
16 Rule R6:
17     ['animal come carne']
18     [Hypothesis: animal es carnívoro with 1.0 certain]
19 Rule R7:
20     ['animal tiene garras']
21     [Hypothesis: animal es carnívoro with 0.8 certain]
22 Rule R8:
23     ['animal es mamífero', 'animal tiene pezuñas']
24     [Hypothesis: animal es ungulado with 1.0 certain]
25 Rule R9:
26     ['animal es mamífero', 'animal es rumiante']
27     [Hypothesis: animal es ungulado with 0.75 certain]
28 Rule R10:
29     ['animal vive con personas']
30     [Hypothesis: animal es doméstico with 0.9 certain]
31 Rule R11:
32     ['animal vive en el zoológico']
33     [Hypothesis: animal es doméstico with -0.8 certain]
34 Rule R12:
35     ['animal es mamífero', 'animal es carnívoro', 'animal tiene manchas oscuras']
36     [Hypothesis: animal es cheetah with 0.9 certain]
37 Rule R13:
38     ['animal es mamífero', 'animal es carnívoro', 'animal tiene rayas negras']
39     [Hypothesis: animal es tigre with 0.85 certain]
40 Rule R14:
41     ['animal es mamífero', 'animal es carnívoro', 'animal es doméstico']
42     [Hypothesis: animal es perro with 0.9 certain]
43 Rule R15:
44     ['animal es reptil', 'animal es doméstico']
45     [Hypothesis: animal es tortuga with 0.7 certain]
46 Rule R16:
47     ['animal es mamífero', 'animal es ungulado', 'animal tiene cuello largo']
48     [Hypothesis: animal es jirafa with 1.0 certain]
49 Rule R17:
50     ['animal es mamífero', 'animal es ungulado', 'animal tiene rayas negras']
51     [Hypothesis: animal es cebra with 0.95 certain]
52 Rule R18:
53     ['animal es mamífero', 'animal puede volar', 'animal es feo']
54     [Hypothesis: animal es murciélago with 0.9 certain]
55 Rule R19:
56     ['animal es ave', 'animal vuela bien']
57     [Hypothesis: animal es gaviota with 0.9 certain]
58 Rule R20:
59     ['animal es ave', 'animal corre rápido']
60     [Hypothesis: animal es avestruz with 1.0 certain]
61 Rule R21:
62     ['animal es ave', 'animal es parlanchín']
63     [Hypothesis: animal es loro with 0.95 certain]
64 Rule R22:
65     ['animal es mamífero', 'animal es grande', 'animal es ungulado', 'animal tiene trompa']
```

Posteriormente, cuando se presiona el botón “Ready”, el algoritmo comienza a realizar las preguntas pertinentes para demostrar las hipótesis. La vista de una pregunta se muestra en la Figura 2. En ella se puede apreciar que se tiene una etiqueta indicando que se está haciendo una pregunta para una hipótesis y una barra deslizante cuyo valor $\in [-1, 1]$ para *No* y *Yes*, respectivamente. Además se tiene un botón *Next Question* con el cual se confirma el valor de certeza y se pasa a la siguiente pregunta.

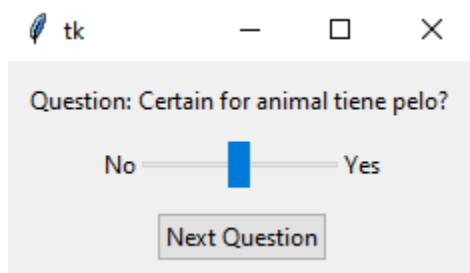


Figura 2: Vista de la pregunta animal tiene pelo.

Finalmente, cuando se consigue un resultado la vista que se obtiene es como la que se muestra en la Figura 3. Esta contiene una etiqueta que indica la hipótesis que se ha demostrado con el nivel de certeza obtenido, junto con una imagen del animal. Por otra parte, se tienen botones que permiten al usuario realizar otro intento o finalmente cerrar el programa.

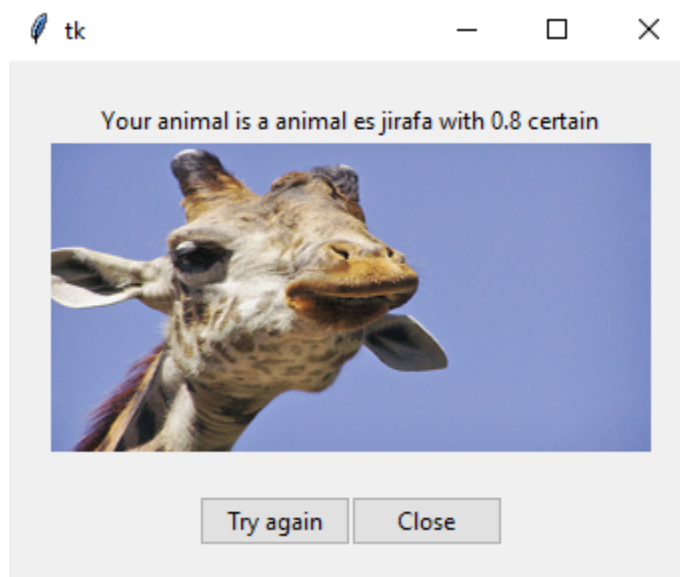


Figura 3: Resultado de Jirafa

Un ejemplo de inferencia es el que se muestra a continuación. Se muestran las preguntas que se realizan y las respuestas obtenidas para el caso en que se utiliza *Early Stopping* y no. Se quiere adivinar el animal **Perro**:

Usando *Early Stopping*

- ¿Animal tiene pelo? Sí
- ¿Animal da leche? Sí
- ¿Animal pone huevos? No
- ¿Animal tiene piel dura? No
- ¿Animal tiene plumas? No
- ¿Animal come carne? Sí
- ¿Animal tiene garras? Sí
- ¿Animal vive con personas? Sí
- ¿Animal vive en el zoológico? No
- Resultado: Animal es perro con certeza 0.72

Sin utilizar *Early stopping*

- ¿Animal tiene pelo? Sí
- ¿Animal da leche? Sí
- ¿Animal pone huevos? No
- ¿Animal tiene piel dura? No
- ¿Animal tiene plumas? No
- ¿Animal come carne? Sí
- ¿Animal tiene garras? Sí
- ¿Animal vive con personas? Sí
- ¿Animal vive en el zoológico? No
- ¿Animal puede volar? No
- ¿Animal es feo? No
- ¿Animal tiene rayas negras? No
- ¿Animal es grande? No
- ¿Animal tiene pezuñas? No
- ¿Animal es rumiante? No
- ¿Animal tiene trompa? No
- ¿Animal tiene cuello largo? No

- ¿Animal tiene manchas oscuras? No
- ¿Animal vuela bien? No
- ¿Animal corre rápido? Sí
- ¿Animal es parlanchín? No
- Animal es perro con 0.72 de certeza

Notar que se consideraron sólo los casos límites para la ejemplificación en donde 1 es Sí y -1 es No.

4. Discusión y conclusiones

Se tiene que el sistema puede “adivinar” el animal en el que uno está pensando de manera satisfactoria. También se puede observar que utilizar *Early Stopping* puede ser de gran utilidad para ahorrar tiempo en el diagnóstico, ya que si se ha demostrado una hipótesis con un gran grado de confianza y asumiendo que estas son mutuamente excluyentes, no tiene sentido seguirle preguntando al usuario, pues en principio no se debería obtener otra con mayor grado de certeza.

Por otra parte, se tiene que las reglas son muy importantes para el funcionamiento correcto del sistema. De hecho el orden puede influir significativamente en el rendimiento del sistema. Por ejemplo, existen reglas que tienen premisas sensibles que no pueden ser demostradas por otras reglas y que además no están presentes en otras reglas. Esto ocurre con la premisa “tiene rayas negras”, “es feo”, “tiene trompa”, “tiene cuello largo”, “es parlanchín”. De esta forma, se podría alterar el orden de consultas de las reglas para que estas premisas sean preguntadas con anterioridad y no se tengan que esperar demasiado en estos casos sensibles que pueden ser optimizados fácilmente.

Además, se comprobó que el uso de una interfaz gráfica facilita el funcionamiento del sistema, ya que le permite al usuario despreocuparse de todo lo que ocurre por detrás y sólo responder las preguntas relevantes para la inferencia.

Finalmente, se pudo comprobar que la aplicación de encadenamiento inverso para este tipo de aplicaciones es muy útil. Esto es porque se parte de la premisa de que hay un diagnóstico que hay que buscar, por lo tanto sólo se exploran las reglas relevantes para la inferencia de la hipótesis. En adición, se puede mencionar al no tener que computar todas las premisas por adelantado, se reduce el tiempo de cómputo lo que es muy muy deseable para cuando se tienen sistemas con más reglas.