

Comparativa de sistema experto basado en reglas, redes neuronales y probabilidades.

Comparison of rule-based expert system, neural networks and probability.

Raúl E. Huarote Zegarra

Universidad César Vallejo, Trujillo, Perú
raulhuarote@ucv.edu.pe

Recepción: 16-05-2013 / **Aceptación:** 13-07-2013

RESUMEN

El presente artículo "Comparativa de sistema experto basado en reglas, redes neuronales y probabilidades", muestra los diferentes modelos para resolver un caso de sistemas expertos, para ello nos agenciamos de dos áreas de la inteligencia artificial y una de la estadística y probabilidad para poder dar solución a este tipo de casos que requiera conocimiento. Considerando casos similares de tipo síntomas-enfermedad para las diferentes maneras de cómo resolverlos. Para el caso de probabilidades se tiene que crear un conjunto de reglas que refleja la interpretación del especialista en el área de evaluación (esto se encarga el ingeniero de conocimiento) formalizándolo en un grafo AND/OR y llevando a un lenguaje de tenga motor de inferencia para poder resolverlos. Otro caso es usando redes neuronales donde los patrones de entrada son los síntomas y las salidas son las enfermedades, para ello se tiene que ingresar casos para el proceso de aprendizaje, tomando como tipo de red neuronal el de propagación hacia atrás. Como último caso se usa el método probabilístico del teorema de bayes (tomando los casos históricos como antecedentes de pacientes).

Usamos estos métodos para dar solución a casos que se requiera un especialista, donde cada uno tiene su particularidad para dar soluciones (ya sea en el proceso de aprendizaje, de inferencia o de evaluación por antecedentes), entre otras cosas.

Palabras clave: redes neuronales, grado AND/OR, teorema de bayes.

ABSTRACT

This article "Comparison of rule-based expert system, neural networks and probability", shows the different models to solve a case of expert systems, agencies, therefore we are two areas of artificial intelligence and statistics and probability in order to solve this kind of case that requires knowledge. Considering similar cases-disease-like symptoms for different ways to solve them. In the case of chances you have to create a set of rules that reflects the interpretation of specialist assessment area (this is handled by the knowledge engineer) by executing a graph AND / OR and leading to language inference engine has to solve them. Another is using neural networks where input patterns are the symptoms and diseases are outputs, this will have to enter cases for the learning process, taking as the type of neural network back propagation. As a last case we use the probabilistic method of Bayes' theorem (taking the case history and background of patients). We use these methods to solve cases that require a specialist, where each has its particularity to give solutions (either in the process of learning, inference or evaluation by history), among other things.

Key words: Neural net, Node AND/OR, theorem of bayes.

1. INTRODUCCIÓN

Estos tres procedimientos tienen como objetivo comportarse como un especialista en un área específica, poder resolver un problema planteado. Cada uno de los modelos a presentar tiene su particularidad de poder resolver.

Para el sistema experto basado en reglas: Es un programa en computadora, inteligente que usa el conocimiento y procedimientos de inferencia para resolver problemas que son suficientemente difíciles para requerir significativa experiencia humana para su solución. Actúa sobre un dominio específico. [1]

Para el sistema experto basado en redes neuronales: Sistema que puede actuar de la misma forma que funciona el cerebro humano o simularlo. Pueden procesar muchas piezas de información y aprender a reconocer patrones. Aprendiendo por ensayo error (típico de la red neuronal de propagación hacia atrás). [2]

Para el sistema experto basado en probabilidades: Considerando que la mayor parte de aplicaciones maneja incertidumbres. Como si un paciente presenta un conjunto de síntomas médicos ¿Cuál es su enfermedad? El conocimiento no es terminantico puesto que la relación entre enfermedades y síntomas no son determinísticas, ya que un conjunto de síntomas puede estar asociado a diferentes enfermedades. [3]

2. CONTENIDO

Para el sistema experto se va a tomar como caso una enfermedad (como salida o diagnóstico) y los síntomas asociados a esta enfermedad. Se considera a un paciente que tiene la enfermedad anemia siempre y cuando tenga el síntoma de cansancio, palidez en la piel o en la parte interna de los párpados o uñas, sensación de frío, falta de apetito, debilidad muscular y somnolencia.

Mostrando este cuadro, el especialista (médico tratante) debe tener en cuenta los síntomas para poder dar su respuesta (o diagnosticar la enfermedad asociada a la enfermedad). La idea es que cada uno de los modelos responda (o

proporcione la respuesta) tal como lo hace el especialista.



Figura 1: Representación de la anemia.

3. METODOLOGIA

3.1. Sistema experto basado en reglas.

Para ello debemos de conocer la manera de como representar el cuadro de síntomas-enfermedades para cada uno de los casos a evaluar. Usaremos la manera de representar en grafos AND/OR. Primero lo representamos como cuantificadores:

$\forall enfermedad = anemia / (síntoma = (cansancio \wedge (palidez\ en\ la\ piel \wedge palidez\ en\ la\ parte\ interna\ de\ los\ párpados \vee en\ la\ parte\ interna\ de\ las\ uñas) \wedge sensación\ de\ frío \vee falta\ de\ apetito \wedge debilidad\ muscular \wedge somnolencia))$. Esto se representa gráficamente en grafo AND / OR tal como se muestra:

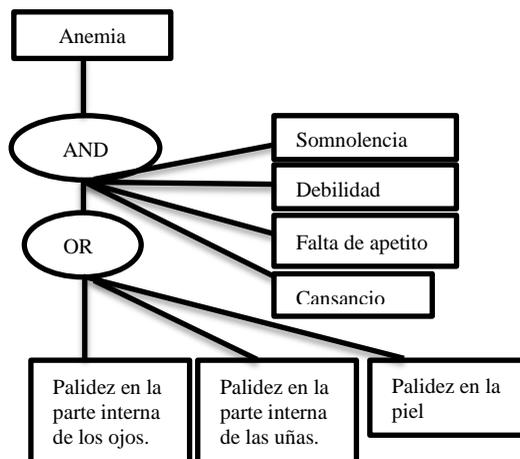


Diagrama 1: representación grafo AND/ OR.

Arquitectura de un sistema experto

La manera en el cual un especialista interactúa con su paciente es un método persona-persona y tanto la información proporcionada (los síntomas que adolece) como el especialista que le proporciona el diagnóstico (la enfermedad que adolece).

Lo ideal es que el sistema experto se acerque a la sensación de persona a persona (una aproximación) donde el sistema experto le proporciona los síntomas (como lista de opciones) para que marque si es que lo padece o no (en este caso marcado o desmarcado respectivamente).

Para ello se muestra un cuadro donde representamos el modelo de cómo trabaja el sistema experto basado en reglas internamente para proporcionar el resultado (diagnóstico) a un caso particular.

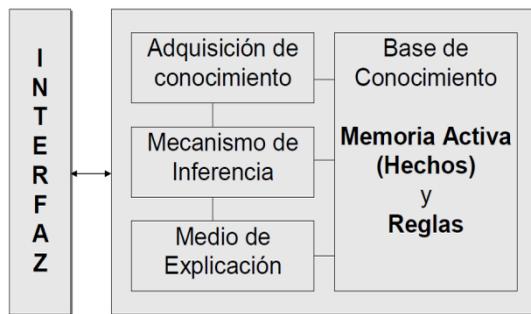


Figura 2: Representación de un sistema experto.

A. La interfaz

Es la parte donde el usuario interactúa con el sistema experto, lo que el usuario contempla respecto al sistema experto. Debe ser lo más amigable posible (sea lo más sencillo para el usuario, evitando la exageración de acciones a realizar hasta mostrar el resultado o diagnóstico).

Debe de contemplar una interfaz gráfica, colocándole imágenes de ayuda para saber de lo que se trata.

B. Adquisición del conocimiento.

Es la parte donde el ingeniero en conocimiento (la persona que se encarga de adquirir la información del especialista) obtiene la información, plasmándolo en un modelo lógico ya sea en un grafo AND/OR entre otros.

Por lo general se obtiene mediante entrevistas, bibliografías, ensayos, etc.

Debe tenerse en cuenta que el conocimiento adquirido no necesariamente es de un especialista; por ello el sistema experto puede alimentarse de conocimiento de varios especialistas, donde refleja en los resultados.

C. Mecanismo de inferencia.

Cuando se tiene el conocimiento (obtenido del(los) especialista) y formalizado este tiene que ser plasmado en un lenguaje de programación que cuente con motor de inferencia donde realizar una secuencia de pasos internos hasta encontrar una solución como respuesta encontrada. Para ello se usa un motor de inferencia que ofrece los lenguajes de programación lógica como el Swi-Prolog. Visual Prolog, etc.

D. Medio de explicación.

Dentro del conocimiento, debemos de mostrar como el motor de inferencia ha proporcionado un resultado y que validez tiene dicho resultado.

Como reportar los resultados (en el contenido que sea lo más sencillo posible para el usuario) esto se contempla puesto que para el usuario no conocedor del tema, tenga una explicación sencilla.

E. Base de conocimiento.

Es la información adquirida por el ingeniero de conocimiento (proveniente del especialista, bibliografía, videos, etc.). Para ello lo representamos en:

Hechos: Datos sueltos necesarios para el proceso de inferencia. Usa la regla de inferencia para obtener una respuesta de acuerdo a la cantidad de hechos proporcionados. Tal como cansancio, falta de apetito, palidez en la piel, anemia, etc.

Reglas: Es la relación que existe entre todos los hechos formulados, por ejemplo, si tienes palidez en la piel y falta de apetito, entre otros entonces tienes anemia.

De esta manera lo representamos en un lenguaje de programación lógica que maneja motor de inferencia. Swi-prolog.

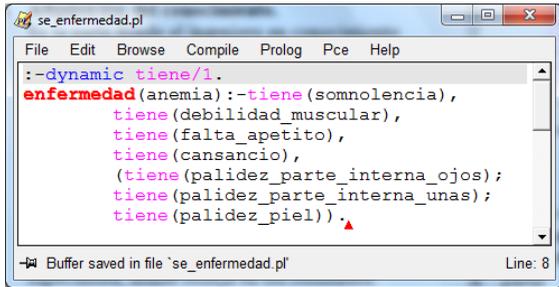


Figura 3: Programa en swi-prolog.

3.2. Sistema experto basado en red neuronal.

Es un algoritmo de aprendizaje que imita el proceso humano para poder aprender. Relaciona las neuronas y lo plasma en un modelo artificial. Se toma como modelo de red neuronal, al back propagation (propagación hacia atrás) por que se adecua a este caso. Mostramos la característica de una red neuronal para el proceso de aprendizaje:

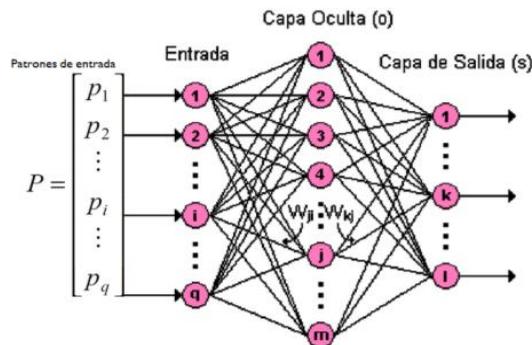


Figura 4: Representación de una red neuronal.

Donde el P_i viene a ser representados por los síntomas y las salidas están representadas por enfermedades, donde para cada P_i tiene el valor 0 ó

1 (no tiene el síntoma o si lo tiene respectivamente).

- Donde el valor neto obtenido por la capa oculta es:

$$n_j^o = \sum_{i=1}^q W_{ji}^o p_i + b_j^o$$

- El valor operado por la función de transferencia de la capa oculta para la entrada de la capa de salida es:

$$a_j^o = f^o \left(\sum_{i=1}^q W_{ji}^o p_i + b_j^o \right)$$

- El valor obtenido para como peso ponderado que son previos para la capa de salida es:

$$n_k^s = \sum_{j=1}^m W_{kj}^s a_j^o + b_k^s$$

- Calculo de error en cada unidad de la capa de salida:

$$\delta_k = (t_k - a_k^s)$$

Dónde:

a_k^s : Valor obtenido

t_k : Valor deseado

- Error medio cuadrático:

$$ep^2 = \frac{1}{2} \sum_{k=1}^s (\delta_k)^2$$

- Error total en el proceso de aprendizaje en una iteración luego de haber presentado los r patrones de entrenamiento:

$$e^2 = \sum_{p=1}^r ep^2$$

- Para la actualización de pesos de la capa oculta:

$$W_{ji}(t + 1) = W_{ji}(t) - 2 \alpha \delta_j^o p_i$$

$$b_j(t + 1) = b_j(t) - 2 \alpha \delta_j^o$$

Dónde:

$$\delta_j^o = \text{Sensibilidad de capa oculta}$$

$$\delta_j^o = f^0(n_j^o) \times \sum_{k=1}^j \delta_k^s w_{kj}^s$$

$w_{ji}(t + 1)$: Nuevo peso

$w_{ji}(t)$: Peso actual

α : Razón

p_i : Patrones de entrada

- Actualización de pesos de la capa de salida:

$$W_{kj}(t + 1) = W_{kj}(t) - 2 \alpha \delta_k^s$$

$$b_k(t + 1) = b_k(t) - 2 \alpha \delta_k^s$$

Dónde:

$$\delta_k^s = \text{Sensibilidad de capa salida}$$

$$\delta_k^s = (t_k - a_k^s) f'^s(n_k^s)$$

$w_{kj}(t + 1)$: Nuevo peso

$w_{kj}(t)$: Peso actual

α : Razón

- Función de transferencia:
Función que permite actualizar los pesos de manera secuencial.

$$\text{logsig: } f(n) = \frac{1}{1 + e^{-n}}$$

Entonces el proceso de aprendizaje de una red neuronal está dado por el historial de pacientes, donde la red neuronal tiene como patrones de entrada a cada caso (conjunto de síntomas) y salida por cada caso la respectiva enfermedad diagnosticada. Tal como lo muestra la Tabla 1.

La idea de resolver un sistema experto basado en redes neuronales es que aprenda los casos ya obtenidos (casos históricos o historial de casos) y poder responder por cercanía a un nuevo caso que se presente (para el mapeo).

Tabla 1: Historial de pacientes.

Paciente	Enfermedad	Síntomas		
		S1	S2	S3
	E			
1	e ₅	1	1	1
2	e ₃	1	0	1
3	e ₃	1	1	0
4	e ₅	0	0	1
5	e ₃	0	1	0
6	e ₁	1	1	0
7	e ₁	1	1	1
8	e ₃	1	0	0
9	e ₁	1	1	1
10	e ₅	1	0	1

3.2. Sistema experto basado en probabilidades.

La mayor parte de aplicaciones maneja incertidumbre. La certeza de ello se basa en la cantidad de información que tienes como antecedente y que tan probable es la respuesta (diagnóstico) que se responde.

El sistema experto basado en probabilidades toma en consideración los datos históricos o antecedentes ocurridos en el historial de los pacientes, tal como lo muestra en la Tabla 1.

Para la solución de un sistema experto basado en probabilidades, nos tenemos que agenciar del teorema de Bayes. Tal como se muestra en la siguiente ecuación:

$$p(x_i | x_1, \dots, x_k) = \frac{p(x_i | x_1, \dots, x_k)}{p(x_1, \dots, x_k)}$$

La ecuación mostrada anteriormente se representaría para nuestro caso como:

$$p(e_i | s_1, \dots, s_k) = \frac{p(e_i) p(s_1, \dots, s_k | e_i)}{\sum_{e_i} p(e_i) p(s_1, \dots, s_k | e_i)}$$

Dónde e = enfermedad, S= síntomas y p=probabilidad condicional.

Donde se expresa “cuál es la probabilidad de que tenga la enfermedad e_i, dado que tiene el síntoma s₁=1, s₂=1, s₃=0,.....”

La probabilidad condicional de todas las enfermedades a analizar se muestra en la tabla 2.

Tabla 2: Resultados de la evaluación de casos.

Enfermedad e	$p(e_i s_1, \dots, s_k)$
e_1	0.2
e_2	0.1
e_3	0.8 ← más probable
e_4	0.4
e_5	0.0 ← menos probable
e_6	0.7
...	...

Dónde e_3 representa como más probable a que tenga la enfermedad el e_3 y la menos probable es e_5 . También éste resultado se puede tomar como nuevo dato histórico a ser almacenado como parte del grupo de datos históricos.

4. PRUEBAS

Las pruebas realizadas son con un software para cada caso:

Sistema experto basado en reglas: Contempla las opciones para que el usuario ingrese los síntomas y le proporcione como resultado la enfermedad que adolece. Esto sucede cuando el lenguaje implementado (NetBeans) proporciona los síntomas al swi-prolog y a su vez esta le envía una respuesta que es la enfermedad que está asociada los síntomas seleccionados.

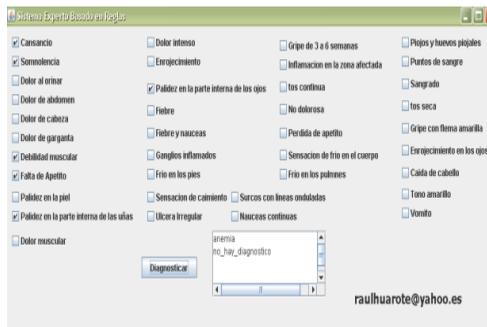


Figura 5: Implementación del sistema experto basado en reglas.

Sistema experto basado en Redes Neuronales: Los patrones de entrada son los síntomas y se espera una respuesta como salida (en el proceso de mapeo) y que la salida represente el proceso de aprendizaje

(de los casos o antecedentes ingresados para su proceso de aprendizaje).

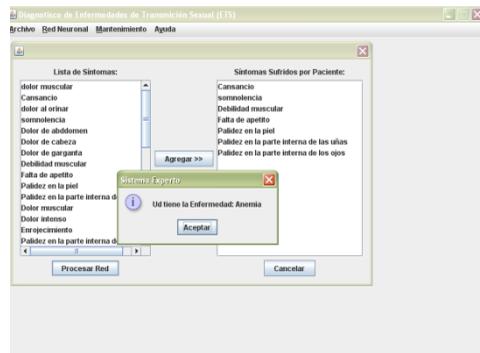


Figura 6: Implementación del sistema experto basado en redes neuronales.

Sistema experto basado en probabilidades:

Teniendo una base de datos de los historiales de los pacientes donde adolecen de una enfermedad y está asociada a un conjunto de síntomas, debe reportar el más probable que tenga una enfermedad a comparación de otra (ello se refleja en valores que estén cerca al 1) y rechazar los menos probables (valores que estén cerca al 0).

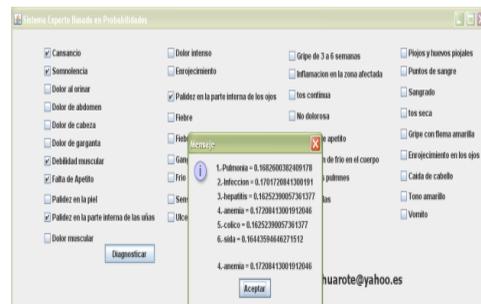


Figura 7: Implementación del sistema experto basado en probabilidades.

5. OBSERVACIONES

- El sistema experto basado en reglas, es necesario hacer la conexión entre el lado de la interfaz (NetBeans) con lenguaje de programación lógica (Swi-Prolog).
- El sistema experto basado en redes neuronales, no considera las operaciones lógicas OR, se toman como una representación de conexiones

conjuntivas a cada uno de los síntomas asociados a una enfermedad.

- El sistema experto basado en redes neuronales, en el proceso de diagnóstico, se evalúa de acuerdo a los pesos sinápticos que ha llegado a aprender la red.
- En el sistema experto basado en probabilidades, cuando se realiza el proceso de entrega de enfermedad diagnosticada, es necesario que lea todos los datos históricos para que emita su resultado.

6. CONCLUSIONES

- Fue posible el desarrollo el software de tipo sistema experto basado en reglas, redes neuronales y basadas en probabilidades con el fin de resolver problemas de tipo médico.
- Queda demostrado que las diferentes metodologías para el desarrollo de un sistema experto proporciona resultados de la misma manera que un especialista.
- El entorno de desarrollo para del sistema experto es de manera amigable para el usuario que padece el síntoma y desea ser diagnosticado.
- Las técnicas del sistema experto, permiten su interpretación para su desarrollo e implementación informática, conjuntamente con los usuarios conocedores de los métodos.
- Se implementó el software para el sistema experto con NetBeans 6.9 para el sistema experto basado en redes neuronales y basado en probabilidades, y para el sistema experto basado en reglas se usó adicionalmente el Swi-Prolog 5.10.1.
- Se presenta la versión funcional del software para el sistema experto mediante la programación orientada a objetos.
- La comparación lo podemos mostrar en la siguiente tabla.

Tabla 3: Cuadro comparativo de los sistemas expertos.

	Basado en reglas	Redes neuronales	Probabilidades
Base de conocimiento	Objeto, reglas y hechos.	Datos de entrada y suceso de salida.	Sucesos(casos históricos)

Motor de inferencia	Estrategias de inferencia.	(Aprendizaje) Actualización de los pesos sinápticos.	Probabilidad condicional.
Subsistema de explicación	Basado en reglas activas.	Modelo y características de la red neuronal (Back propagation)	Teorema de bayes.
Aprendizaje	Cambio en objetos y reglas.	Entrenamiento de la red neuronal.	Cambio en modelo probabilístico. Actualización y/o adición de sucesos.

7. RECOMENDACIONES

- Para la utilización del sistema experto se recomienda que cuenta con una base de datos de historial para poder acceder de manera inmediata al diagnóstico y resultado.
- Utilizar este software como base para el desarrollo de estudio de tipo sistema experto.
- Mejorar el proceso de aprendizaje en las redes neuronales para casos en que se consideren tipos de conectores lógicos asociados al conocimiento.
- Difundir este software para su utilización tanto como herramienta de trabajo como para el uso académico.

8. AGRADECIMIENTOS

Agradeciendo a la escuela de Informática de la Universidad Nacional de Trujillo – Perú. Por la formación y el conocimiento proporcionado por los docentes de excelente calidad académica y personal.

REFERENCIAS BIBLIOGRAFICAS

- [1] RUSSELT STUART. Inteligencia artificial: Un Enfoque Moderno. 1ra edición México. Prentice Hall, 1996.979p. ISBN: 0-13-790395-2.
- [2] JOSE RAMON HILERA GONZALES Y VICTOR JOSE MARTINEZ HERNANDO. Redes Neuronales Artificiales (2000).

Fundamentos, modelos y aplicaciones. Editorial Alfa y Omega.

Editorial RAMA 2006, 364p. ISBN: 84-7897-676-0.

[3] ENRIQUE CASTILLO, JOSE MANUEL GUTIERREZ Y ALI S. HADI. Sistemas expertos y modelos de redes probabilísticas. Universidad de Cantabria y Universidad de Cornell.

[5] Jacobson Ivar, Boch Grady. Rumbaugh James, El proceso Unificado de Desarrollo de Software. Pearson Educación, México 2002

[4] PAJARES MARTINEZ GONZALO. Inteligencia Artificial e ingeniería del conocimiento. 1era ed. Madrid (España).

[6] Análisis y Diseño Detallado de Aplicaciones Informáticas de Gestión Mario G. ,Editorial RA-MA. Septiembre de 2002.