

UNIVERSIDAD NACIONAL DE INGENIERÍA
FACULTAD DE INGENIERÍA INDUSTRIAL Y DE SISTEMAS
SECCIÓN DE POSGRADO



**DESARROLLO DE UN PROTOTIPO DE SISTEMA EXPERTO
PARA EL DIAGNÓSTICO DE UNA ENFERMEDAD
GINECOLÓGICA**

TESIS

**PARA OPTAR EL GRADO ACADÉMICO DE MAESTRO
EN CIENCIAS CON MENCIÓN EN:
INGENIERÍA DE SISTEMAS**

VÍCTOR ANDRÉS MELCHOR ESPINOZA

**LIMA- PERÚ
2012**

DEDICATORIA

A la memoria de mi mamita María, de la que guardo los más bellos recuerdos por sus innumerables muestras de cariño y por su continuo afán de hacer cada día las cosas mejor.

A mi madre, Rosa Espinoza Hurtado, verdadero artífice de este trabajo. Nada ni nadie iguala el amor que te tengo.

A mi padre Efraín Melchor Guerrero, amigo y consejero de todos mis proyectos.

A mi hermana Rosa María, por su gran capacidad al tomar decisiones justo a tiempo, y a Charito y Andrea, que cada día me brindan momentos de alegría.

Realmente soy muy afortunado de pertenecer a una familia que siempre está dispuesta a dar su apoyo incondicional en lo profesional, académico y afectivo.

AGRADECIMIENTO

A los médicos especialistas Ronald Cabrera, Eddy Angles e Inés Sovero por su colaboración desinteresada en la elicitación de conocimientos.

Al MSc Celedonio Méndez, por el apoyo académico y profesional que obtuve durante las actividades realizadas y la amistad adquirida a lo largo de este periodo.

Al Mg. Josué Angulo, Jefe de la Sección de Posgrado por su gran labor como facilitador y por sus valiosas sugerencias de mejora para la tesis.

A mi amigo, doctor Jaime Osorio con el que tuve varias reuniones para compartir ideas acerca de la Inteligencia Artificial.

A todos los docentes del Posgrado de Sistemas por su incansable labor en la formación de profesionales que contribuyen a hacer un Perú con más oportunidades.

ÍNDICE

PORTADA.....	I
DEDICATORIA.....	II
AGRADECIMIENTO.....	III
INDICE.....	IV
LISTA DE ABREVIATURAS Y SIGLAS.....	VIII
LISTA DE FIGURAS.....	IX
LISTA DE TABLAS.....	XIII
DESCRIPTORES TEMÁTICOS.....	XV
RESUMEN.....	XVI
ABSTRACT.....	XVII
INTRODUCCIÓN.....	XVIII
CAPÍTULO 1 PROTOCOLO DE LA INVESTIGACIÓN	01
1.1 Problema de Investigación	01
1.1.1 Diagnóstico y Enunciado del Problema	01
1.1.2 Definición del Problema	02
1.1.2.1 Nominal	02
1.1.2.2 Operativa	03
1.2 Objetivos de la Investigación	03
1.2.1 Objetivo General	03
1.2.2 Objetivos Específicos	03
1.3 Marco Metodológico	04
1.3.1 Hipótesis	04
1.3.2 Variables	05
1.3.3 Operacionalización de las Variables	08
1.3.4 Tipo de Investigación	09
1.3.5 Etapas de la Investigación	10
1.4 Población y Muestra	13

1.5 Justificación de la Investigación	14
1.5.1 Importancia del Tema	14
1.5.2 Justificación	14
1.5.2.1 Teórica	14
1.5.2.2 Metodológica	15
1.5.2.3 Práctica	16
1.6 Alcances y Limitaciones	16
CAPÍTULO 2 MARCO TEÓRICO	17
2.1 Inteligencia Artificial	17
2.1.1 Introducción	17
2.1.2 Definición	18
2.1.3 Fundamentos de la Inteligencia Artificial	19
2.1.4 Síntesis Histórica de la Inteligencia Artificial	23
2.1.5 Historia	25
2.1.6 La Prueba de Turing	28
2.1.7 Génesis de la Inteligencia Artificial	30
2.1.8 Líneas de Investigación de la Inteligencia Artificial	32
2.1.8.1 IA Simbólica	33
2.1.8.2 IA Conexionista	33
2.1.8.3 IA Evolucionaria	34
2.1.8.4 IA Híbrida	35
2.1.9 Principales Aplicaciones	35
2.1.9.1 Procesamiento del Conocimiento	35
2.1.9.2 Procesamiento de Lenguaje Natural	36
2.1.9.3 Reconocimiento de Patrones	36
2.1.9.4 Robótica	36
2.1.9.5 Juegos	37
2.1.9.6 Base de Datos Inteligentes	37
2.1.9.7 Visión por Computador	37
2.1.9.8 Prueba de Teoremas	38
2.2 Sistemas Expertos	39
2.2.1 Sistemas Basados en Conocimiento	39
2.2.2 La Cadena DICS	39
2.2.3 Concepto de SBC	41
2.2.4 Introducción a los SE	43
2.2.5 Definición de SE	44
2.2.6 Primeros SE	46
2.2.7 Características de los SE	50
2.2.8 Tipología de los SE	50
2.2.9 Arquitectura de un SE	54
2.2.9.1 Base de Conocimiento	54
2.2.9.2 Memoria de Trabajo	55
2.2.9.3 Motor de Inferencia o de Deducción	55

2.2.9.4 Interfaz de Usuario	56
2.2.9.5 Interfaz de Adquisición del Conocimiento	56
2.2.9.6 Mecanismo de Explicación	57
2.2.10 Eventos Importantes en la Historia de SE	58
2.2.11 Antecedentes en la Medicina	60
2.2.11.1 Pasado	61
2.2.12 Fuentes de Conocimiento	66
2.2.13 Ingeniería del Conocimiento	67
2.3 Representación del Conocimiento	70
2.3.1 Características de las buenas representaciones	71
2.3.2 Conceptos de Abstracción	72
2.3.3 Técnicas para la representación del conocimiento	74
2.3.3.1 Representación en Lógica	75
2.3.3.1.1 Lógica	75
2.3.3.1.2 Lógica de Proposiciones	76
2.3.3.1.3 Lógica de Predicados de Primer Orden	77
2.3.3.1.4 Evaluación del uso de la Lógica	79
2.3.3.2 Sistemas de Producción	80
2.3.3.2.1 Razonamiento con Reglas de Producción	84
2.3.3.2.2 Métodos de Inferencia	84
2.3.3.2.2.1 Encadenamiento hacia adelante	85
2.3.3.2.2.2 Encadenamiento hacia atrás	88
2.3.3.2.3 Evaluación del uso de Reglas de Producción	92
2.3.3.3 Redes Semánticas	94
2.3.3.3.1 Evaluación del uso de Redes Semánticas	96
2.3.3.4 Tripletas Objeto Atributo Valor	97
2.3.3.5 Marcos	98
2.3.3.5.1 Evaluación del uso de Marcos	102
2.3.3.6 Árboles	103
2.3.3.7 Sistemas Probabilísticos	106
2.3.3.7.1 Probabilidad Clásica	106
2.3.3.7.2 Teoría de la Probabilidad	107
2.3.3.7.3 Probabilidades Condicionales	109
2.3.3.7.4 Teorema de Bayes	110
2.4 Personas involucradas en la construcción de Sistemas Expertos	111
2.5 Ventajas y desventajas de los Sistemas Expertos	115
CAPÍTULO 3 DOMINIO DEL PROBLEMA	119
3.1 Elementos del Diagnóstico Médico	119
3.1.1 Signo	119
3.1.2 Síntoma	120
3.1.3 Pruebas de Laboratorio	121
3.1.4 Síndrome	122

3.1.5 Enfermedad	123
3.1.6 Diagnóstico Diferencial	123
3.2 Infecciones de Transmisión Sexual	124
3.3 Definición de Infecciones de Transmisión Sexual	125
3.4 Causas de las Infecciones de Transmisión Sexual	126
3.5 Síndrome de Flujo Vaginal	128
3.5.1 Flujo vaginal normal	128
3.5.2 Definición de Síndrome de Flujo Vaginal	130
3.5.3 Etiología	131
3.5.4 Vaginitis Bacteriana	132
3.5.5 Candidiasis	134
3.5.6 Tricomoniasis	139
CAPÍTULO 4 MODELO PARA EL DIAGNÓSTICO DIFERENCIAL MEDIANTE EL CUADRO CLÍNICO	144
4.1 Introduciendo los SE basados en Probabilidad	144
4.1.1 El modelo de Síntomas Dependientes	146
4.1.2 El modelo de Síntomas Independientes	148
4.1.3 El modelo de Síntomas Relevantes Independientes	150
4.1.4 El modelo de Síntomas Relevantes Dependientes	152
CAPÍTULO 5 METODOLOGÍAS PARA DESARROLLAR SE	154
5.1 Metodologías para desarrollar un SE	154
5.1.1 Metodología de Weiss y Kulikowski	155
5.1.2 Metodología IDEAL	159
5.2 Desarrollo de la Metodología IDEAL	162
5.2.1 Fase I: Requerimientos, viabilidad y especificación técnica	162
5.2.2 Fase II: Desarrollo de los prototipos	165
CAPÍTULO 6 DESARROLLO E IMPLEMENTACIÓN DEL PROTOTIPO	185
6.1 Descripción del sistema desarrollado	185
CONCLUSIONES Y RECOMENDACIONES	210
Conclusiones	210
Recomendaciones	213
Trabajos Futuros	215
GLOSARIO DE TÉRMINOS DE INTELIGENCIA ARTIFICIAL	217
GLOSARIO DE TÉRMINOS MÉDICOS	220
REFERENCIA BIBLIOGRÁFICA	222
BIBLIOGRAFÍA COMPLEMENTARIA	227
EXTRACTO DE UNA ENTREVISTA A UN ESPECIALISTA	229

LISTA DE ABREVIATURAS Y SIGLAS

BC	Base de Conocimientos
CLIPS	C Language Integrated Production System
GPS	General Problem Solver
IA	Inteligencia Artificial
IAC	Inteligencia Artificial Conexionista
IAS	Inteligencia Artificial Simbólica
MI	Mecanismo de Inferencia
PB	Probabilidad Bayesiana
RB	Red Bayesiana
SE	Sistema Experto
SBC	Sistema Basado en Conocimientos
UNI	Universidad Nacional de Ingeniería

LISTA DE FIGURAS

Figura 2.1 – Doodle en homenaje al aniversario del nacimiento de Turing.....	28
Figura 2.2 – La prueba de Turing.....	30
Figura 2.3 – La cadena DICS	40
Figura 2.4 – Relación entre SBC y SE.....	43
Figura 2.5 – Un probable problema y su relación con el dominio del conocimiento	44
Figura 2.6 – Arquitectura de un SE basado en reglas.....	58
Figura 2.7 – Primeros Sistemas Expertos en Medicina.....	64
Figura 2.8 – Proceso de Adquisición del Conocimiento.....	68
Figura 2.9 – Encadenamiento hacia adelante.....	86
Figura 2.10 – Encadenamiento hacia atrás.....	90
Figura 2.11 – Red Semántica construida a partir de un texto.....	95
Figura 2.12 – Tripletas OAV.....	97
Figura 2.13 – Organización jerárquica usando marcos	100
Figura 2.14 – Representación mediante marcos	101
Figura 2.15 – Árbol Binario	104

Figura 2.16 – Árbol de decisión mostrando conocimiento acerca de animales	105
Figura 2.17 – Actores del equipo de desarrollo del SE.....	112
Figura 3.1 – Escala del pH.....	129
Figura 3.2 – Flujo Vaginal.....	131
Figura 3.3 – Células escamosas del cérvix cubiertas con la bacteria Gardnerella vaginalis (flechas).....	133
Figura 3.4 – Cultivo en una placa de agar de Candida albicans.....	137
Figura 3.5 – Epidemiología de Trichomoniasis.....	140
Figura 4.1 – Una ilustración gráfica del modelo de síntomas dependientes....	146
Figura 4.2 – Una ilustración gráfica del modelo de síntomas independientes.	148
Figura 4.3 – Una ilustración gráfica del modelo de síntomas relevantes independientes.....	150
Figura 4.4 – Una ilustración gráfica del modelo de síntomas relevantes dependientes.....	152
Figura 5.1 – Proceso aplicado por la Ingeniería del Conocimiento.....	155
Figura 5.2 – Etapas en el desarrollo de un Sistema Experto.....	158
Figura 5.3 – Implementación de hechos, plantillas y reglas en CLIPS.....	175
Figura 5.4 – Entorno Integrado de Desarrollo de Visual FoxPro.....	178
Figura 5.5 – Modelo inicial de las entidades en el diagnóstico médico.....	180
Figura 5.6 – Evolución de algunos lenguajes de programación.....	182
Figura 6.1 – Pantalla principal del prototipo de Sistema Experto.....	187
Figura 6.2 – Almacenamiento interno de los Síntomas.....	188
Figura 6.3 – Ingreso de los Síntomas.....	189

Figura 6.4 – Listado de los Síntomas.....	190
Figura 6.5 – Eliminación de Síntomas.....	191
Figura 6.6 – Almacenamiento interno de las Enfermedades.....	191
Figura 6.7 – Ingreso de las Enfermedades.....	192
Figura 6.8 – Listado de las Enfermedades.....	193
Figura 6.9 – Eliminación de Enfermedades.....	194
Figura 6.10 – Asociación de Síntomas a las Enfermedades.....	195
Figura 6.11 – Listado de Síntomas asociados a las Enfermedades.....	196
Figura 6.12 – Grabar base de datos en archivo.....	197
Figura 6.13 – Leer base de datos de archivo.....	197
Figura 6.14 – Carga de datos de enfermedades.....	198
Figura 6.15 – Carga de datos de síntomas.....	199
Figura 6.16 – Carga de síntomas de enfermedades.....	200
Figura 6.17 – Ingreso inicial de síntomas en la consulta.....	201
Figura 6.18 – Diagnóstico diferencial inicial.....	202
Figura 6.19 – Opciones de cuestionario.....	202
Figura 6.20 – Preguntas del cuestionario automático.....	203
Figura 6.21 – Segunda aproximación de Diagnóstico.....	203
Figura 6.22 – Tercera aproximación de Diagnóstico.....	204
Figura 6.23 – Cuarta aproximación de Diagnóstico.....	205
Figura 6.24 – Quinta aproximación de Diagnóstico.....	206
Figura 6.25 – Sexta aproximación de Diagnóstico.....	206
Figura 6.26 – Séptima aproximación de Diagnóstico.....	207

Figura 6.27 – Octava aproximación de Diagnóstico.....	208
Figura 6.28 –Diagnóstico Final.....	209

LISTA DE TABLAS

Tabla 1.1 – Operacionalización de las variables.....	09
Tabla 1.2 – Etapas en el desarrollo de la investigación.....	12
Tabla 2.1 – Hitos en el desarrollo de la IA.....	25
Tabla 2.2 – Tipos de Conocimiento.....	41
Tabla 2.3 – Eventos importantes en la historia de los SE.....	59
Tabla 2.4 – Sistemas de Inteligencia Artificial en Uso en Clínicas.....	66
Tabla 2.5 – Conectores Lógicos.....	77
Tabla 2.6 – Tabla de verdad de los conectores lógicos binarios.....	77
Tabla 2.7 – Tabla de verdad de la negación.....	77
Tabla 2.8 – Terminología usada en SE basados en reglas de producción.....	83
Tabla 2.9 – Comparación entre Encadenamiento Progresivo y Regresivo.....	92
Tabla 2.10 – Tabla OAV.....	98
Tabla 2.11 – Marco que describe a un vehículo.....	99
Tabla 3.1 –Signos asociados a una enfermedad	120
Tabla 3.2 –Síntomas asociados a una enfermedad	121
Tabla 3.3 –Prueba para descartar una enfermedad	122
Tabla 3.4 –Ejemplo de Síndrome	123
Tabla 3.5 –Comparación de calorías entre jugos y comida chatarra	135
Tabla 3.6 –Características del flujo vaginal según causa	143
Tabla 5.1 – Fases y etapas de la metodología IDEAL.....	160
Tabla 5.2 –Relación de patologías para el flujo vaginal	166

Tabla 5.3 –Relación de signos para el flujo vaginal	166
Tabla 5.4 –Relación de síntomas para el flujo vaginal	167
Tabla 5.5 –Relación de pruebas de laboratorio para el flujo vaginal	167
Tabla 5.6 –Enfermedades junto con sus probabilidades a priori	168
Tabla 5.7 –Verosimilitud de signos, síntomas y pruebas para la candidiasis	168
Tabla 5.8 –Verosimilitud de signos, síntomas y pruebas para la tricomoniasis	169
Tabla 5.9 –Verosimilitud de signos, síntomas y pruebas para la vaginosis	169

DESCRIPTORES TEMÁTICOS

1. **Inteligencia Artificial.**
2. **Sistemas Expertos.**
3. **Representación del Conocimiento.**
4. **Razonamiento Probabilístico.**
5. **Teorema de Bayes.**
6. **Informática Médica.**
7. **Diagnóstico Médico.**
8. **Síndrome de Flujo Vaginal.**

RESUMEN

Este trabajo de investigación presenta el desarrollo de un prototipo de Sistema Experto para el diagnóstico de patologías que corresponden al síndrome de flujo vaginal. El software utiliza en su base de conocimiento tanto las probabilidades a priori de las patologías de estudio como las verosimilitudes de los síntomas asociados a cada enfermedad, las mismas que han sido determinadas por el especialista de acuerdo a su experiencia profesional. Durante la ejecución, el prototipo permite realizar una serie de iteraciones en las cuales se va alimentando la base de hechos mediante el ingreso de los hallazgos obtenidos en la consulta médica como son los síntomas, signos y pruebas de laboratorio; luego de suministrados estos datos el motor de inferencia utiliza la técnica de propagación de probabilidades para emitir un diagnóstico diferencial en el que se presentan las posibles enfermedades clasificadas según la probabilidad obtenida. En cada iteración, mientras se sigan detectando más hallazgos en el paciente las probabilidades se irán mejorando hasta llegar a un punto en el que el diagnóstico se hace evidente.

PALABRAS CLAVE: Inteligencia Artificial. Sistemas Expertos Probabilísticos. Representación del Conocimiento. Razonamiento Probabilístico. Teorema de Bayes. Informática Médica. Diagnóstico Médico. Síndrome de Flujo Vaginal.

ABSTRACT

This research presents the development of a prototype expert system for diagnosis of diseases that correspond to the vaginal discharge syndrome. The software uses its knowledge base in both the prior probabilities of the pathologies of the study and the likelihood of symptoms associated with each disease, the same that have been determined by the specialist according to his professional experience. At runtime, the prototype allows a number of iterations in which feeding is the basis of facts through the entry of the findings in medical practice such as symptoms, signs and laboratory tests, after the data supplied inference engine uses the technique of spreading likely to make a differential diagnosis where possible diseases are classified according to the likelihood obtained. In each iteration, while still detecting more findings in the patient the odds will improve to a point at which the diagnosis is evident.

KEYWORDS: Artificial Intelligence. Probabilistic Expert Systems. Knowledge Representation. Probabilistic reasoning. Bayes theorem. Medical Informatics. Medical Diagnosis. Vaginal Discharge Syndrome.

INTRODUCCIÓN

Todo ser humano a lo largo de su existencia acumula vivencias y conocimientos aunque sólo un grupo muy reducido llega a desarrollar sus habilidades a un nivel tan alto, razón por la cual se les atribuye el nombre de especialistas.

Siempre ha sido de interés para la comunidad científica el elaborar máquinas que sean capaces de capturar y emular el conocimiento de un especialista en un dominio determinado. En mi caso debido a que poseo cierta experiencia laboral en el sector salud, decidí aplicarlo en la especialidad de Medicina.

En este trabajo cuyo dominio del problema elegido pertenece al campo de la Medicina intento dar respuesta a la siguiente interrogante: ¿Será posible desarrollar una herramienta informática que permita almacenar el conocimiento que un médico especialista posea en un dominio de aplicación específico de manera que al suministrarle el cuadro clínico de un paciente pueda inferir un diagnóstico diferencial que sea tan confiable (o más) como el que emitiría el mismo médico especialista?.

Luego de sostener una serie de entrevistas en diferentes escenarios y tiempos con médicos especialistas o de las conversaciones que se producen en los pasillos entre los residentes y especialistas en los que las frases como *“es muy probable que la enfermedad...”* o *“posiblemente el paciente padezca de ...”* son de uso cotidiano puedo concluir que en el proceso de toma de decisiones diagnóstica existe un gran nivel de incertidumbre lo que me lleva a afirmar que la forma de razonamiento más apropiada para este dominio es el razonamiento probabilístico.

Este trabajo de investigación está enfocado en el desarrollo de un prototipo de Sistema Experto probabilístico que contribuya a precisar la conclusión diagnóstica para mejorar la calidad en el servicio de un consultorio de Ginecología.

El ámbito de la investigación se concentra en el estudio de los Sistemas Expertos que se encuentran enmarcados en el campo de la Inteligencia Artificial. Para emprender el desarrollo de un Sistema Experto es de suma importancia analizar las diversas técnicas que existen para representar el conocimiento, entre estas cabe destacar a las reglas de producción, redes semánticas, tripletas OAV, marcos, árboles, redes bayesianas, entre otras. No se ha incluido en esta investigación la lógica difusa, propuesta por Lofti Zadeh, la cual considero que sería de gran utilidad en el estudio de otras patologías infecciosas como es el caso del HIV.

Por otro lado, se ha abordado como dominio de aplicación las patologías infecciosas en el aparato reproductor femenino que son producidas por agentes etiológicos como bacterias o microbios y cuya sintomatología común es el descenso o flujo vaginal anormal.

En cuanto a la metodología para el desarrollo del Sistema Experto sólo se han considerado las dos primeras fases de la metodología IDEAL que corresponden a la elaboración del prototipo.

Considero que los aportes de esta tesis son: el uso de razonamiento probabilístico como técnica para representar el conocimiento, la aplicación de una metodología para el desarrollo de sistemas expertos así como la implementación de un prototipo con características de Shell, que separa la base de conocimientos del motor de inferencia, lo cual permitirá al investigador concentrarse en el estudio de otra patología de cualquier especialidad aplicando el mismo Shell con enfoque probabilístico.

El presente trabajo está organizado en seis capítulos, de forma que la secuencia de la disposición de las informaciones en ese orden pueda ofrecer un mejor entendimiento de su contenido. Siendo así, la estructura del trabajo se presenta de la siguiente forma:

En el capítulo 1, Protocolo de la Investigación, se presenta el problema de investigación, su delimitación y la justificación e importancia para solucionarlo. Se establecen también los objetivos del trabajo, las hipótesis y variables así como el tipo de investigación, el método empleado, el universo y la muestra.

En el capítulo 2, Marco Teórico, se hará referencia a las bases sobre las cuales está fundamentada la investigación, abordando la Inteligencia Artificial y en particular el tema relativo a los sistemas expertos en medicina y de cómo son desarrollados. Se ha puesto especial énfasis en el estudio de las técnicas de representación del conocimiento que van acompañadas de ejemplos ilustrativos cuyo propósito es que sirvan de ayuda en futuras investigaciones.

En el capítulo 3, Dominio del Problema, se describe primero los elementos que conforman el proceso de diagnóstico médico para luego con un mayor entendimiento de la terminología médica pasar a revisar el Síndrome de Flujo Vaginal y su diagnóstico que es el tema de aplicación del presente trabajo de

investigación, para ello se explicará cómo el médico llega a un diagnóstico y luego se estudiarán las patologías que las conforman, de las que se describirá su etiología, el cuadro clínico y el diagnóstico.

En el capítulo 4, Modelo para el Diagnóstico Diferencial mediante el Cuadro Clínico, se caracterizan diversos modelos probabilísticos, según la relación que presenten los síntomas y enfermedades.

En el capítulo 5, Metodologías para desarrollar SE, se explicará algunas de las metodologías existentes para la confección de Sistemas Expertos, enfatizando en la Metodología IDEAL de la cual se describirá y desarrollará las fases que la conforman.

En el capítulo 6, Desarrollo e Implementación del Prototipo, se presenta como se implementará el prototipo de sistema probabilístico, explicando todas las interfaces de la herramienta y sus funciones así como los aspectos técnicos de la construcción del sistema de apoyo al diagnóstico, mostrando como cada etapa fue realizada. Además se presentan los resultados del trabajo en función de los objetivos específicos propuestos.

En Conclusiones y Recomendaciones se explican las conclusiones alcanzadas en la investigación realizada y se presentan algunas recomendaciones. Por último se dan algunas pautas para trabajos futuros que pueden ser desarrollados a partir del conocimiento proporcionado por este trabajo de investigación. Además he creído conveniente incluir un glosario con términos de inteligencia artificial y médicos que son de uso frecuente en el desarrollo de la tesis y en el Anexo el extracto de una entrevista a un Especialista.

CAPÍTULO I

PROTOCOLO DE LA INVESTIGACIÓN

1.1 PROBLEMA DE INVESTIGACIÓN

1.1.1 DIAGNÓSTICO Y ENUNCIADO DEL PROBLEMA

Las ITS (Infecciones de Transmisión Sexual) son uno de los males más graves de la humanidad pues no distingue en atacar a hombres, mujeres y niños de toda condición social en todo el mundo. En la actualidad es difícil controlar las ITS, por factores biológicos, de comportamiento, etc. y en los últimos años ha aumentado su frecuencia. Las ITS tienen una etiología bacteriana, viral, parasitaria o micótica y se caracterizan principalmente porque se contraen a través de relaciones sexuales.

La Organización Mundial de la Salud (OMS) estima que anualmente en el mundo ocurren 250 millones de casos nuevos de ITS. En el Perú 4 de cada 10 hombres y 5 de cada 10 mujeres tienen una ITS sin saberlo. Según datos del Instituto Nacional de Estadística publicados en 2004, el 5,4% de la población que ha mantenido relaciones sexuales refiere que ha sido diagnosticado de alguna ITS a lo largo de su vida.

Debido a la falta de infraestructura y de recursos de personal adecuados a nivel nacional el MINSA (Ministerio de Salud), a través del PROCETTS (Programa de Control de Enfermedades de Transmisión Sexual y Sida), ha editado el Manual de Doctrina, Normas y Procedimientos para el Control de las ETS y el Sida en el Perú, en el cual se enfoca el manejo clínico y terapéutico de las ITS por síndromes, que es necesario difundir.

En este trabajo proponemos el desarrollo de un sistema experto para el diagnóstico de una ITS en el cual buscaremos capturar, según las técnicas de la ingeniería del conocimiento, la experiencia de un médico especialista en su diagnóstico y tratamiento, asimismo nos basaremos en el documento del manejo sindrómico publicado por el MINSA.

En virtud de la gran cantidad de variables presentes en determinados procesos de la toma de decisión diagnóstica, humanamente es difícil hacer la distinción de que variables y valores interfieren de forma directa o indirecta en estos. Asimismo, algunas veces, el experto del dominio no consigue explicar, o hasta justificar, su conocimiento de forma clara o suficiente para colocarlo en la forma de reglas para el apoyo en el diagnóstico y consecuentemente en la implementación del sistema experto, haciendo que el subjetivismo impere en el proceso de la toma de decisión pudiendo generar conclusiones incorrectas sobre este.

1.1.2 DEFINICIÓN DEL PROBLEMA

1.1. 2.1 NOMINAL

Las estadísticas de los dos últimos años nos señalan que en la Clínica Médica el servicio de Ginecología es uno de los que más pacientes recibe. Al hacer las indagaciones en este servicio se pudo comprobar que la causa más frecuente de consultas de atención primaria corresponde a problemas de vulvovaginitis. Esta

afección no es exclusiva de pacientes adultas ya que puede ser contraída además por adolescentes, prepúberes e incluso niñas.

En las pacientes adultas es necesario realizar una detección temprana y precisa del problema ginecológico ya que de no actuar oportunamente este puede devenir en la adquisición de enfermedades como las ITS o virus como el HIV. Adicionalmente con frecuencia se presentan casos de pacientes asintomáticos, lo cual requiere otros criterios de evaluación diagnóstica a la vez de una mayor especialización del médico a cargo.

1.1.2.2 OPERATIVA

¿Hasta qué punto el desarrollo de un prototipo de Sistema Experto probabilístico contribuye a precisar la conclusión diagnóstica para mejorar la calidad en el servicio de un consultorio de ginecología?

1.2 OBJETIVOS DE LA INVESTIGACIÓN

1.2.1 OBJETIVO GENERAL

El objetivo general de este trabajo es desarrollar un prototipo de Sistema Experto probabilístico que contribuya a precisar la conclusión diagnóstica para mejorar la calidad en el servicio de un consultorio de Ginecología.

1.2.2 OBJETIVOS ESPECÍFICOS

A partir de lo expuesto se puede listar los siguientes objetivos específicos:

1. Realizar un levantamiento de información de la historia de la Inteligencia Artificial para comprender la importancia de las aplicaciones de los Sistemas Expertos en este campo.
2. Determinar las diversas formas de representación del conocimiento y establecer cuál es la más apropiada para el problema de diagnóstico propuesto.
3. Describir las manifestaciones clínicas asociadas a una enfermedad específica para comprender el proceso de diagnóstico consultando bibliografía especializada y entrevistando a médicos expertos.
4. Aplicar un prototipo de Sistema Experto de diagnóstico médico que provisto de una interfaz amigable proporcione al usuario final una fácil comprensión del caso de estudio.

1.3 MARCO METODOLÓGICO

1.3.1 HIPÓTESIS

Kerlinger[KER1971] dice “una hipótesis es una afirmación conjetural acerca de las posibles relaciones entre dos o más variables (...); las hipótesis constituyen oraciones declaratorias que relacionan en alguna forma ciertas variables con otras”. Luis Oblitas [OBL1987] cita a Mc Guigan al decir que “una hipótesis es una proposición comprobable y que podría ser la solución de un problema (...); una afirmación comprobable de una relación potencial entre dos variables”.

La hipótesis de trabajo de la presente tesis señala que: *El desarrollo de un prototipo de Sistema Experto probabilístico para la medicina clásica contribuirá a precisar la conclusión diagnóstica, de acuerdo a los síntomas que presente el paciente, lo cual permitirá mejorar la calidad en el servicio de un consultorio de Ginecología.*

1.3.2 VARIABLES

El disponer de un buen sistema de variables es importante en el proceso de investigación ya que facilita todo un diseño, desarrollo y posterior análisis estadístico de los resultados.

Las variables vienen a ser “un conjunto de datos relacionados unos con otros; permiten probar una subhipótesis o una parte de una hipótesis general y se define como un símbolo de un conjunto determinado de datos, el mismo que puede tomar un valor, cualquiera de ellos. Su dominio es el conjunto de datos que comprende una variable” [CAB1990].

Kerlinger señala que “una variable es un símbolo al que se le asignan numerales (...) Podemos llamar variable a una construcción hipotética”. [KER1971]

Oblitas cita a Arnau quien dice que “son constructos hipotéticos los cuales poseen varios valores o niveles(...), es un símbolo al que se le asignan valores ó números”.

En este trabajo de investigación se ha identificado las siguientes variables conceptuales y operacionales:

- Sistema Experto
- Medicina Clásica

- Entorno de Programación Delphi
- Sistema experto probabilístico para la Medicina Clásica

Variable 1: *Sistema Experto*. Representa una variable independiente que se usa para visualizar el espacio de diseño del conocimiento útil a nuestra propuesta.

Dimensiones

1.1 Conceptos

1.2 Atributos

1.3 Comprensibilidad

Indicadores

1.1 Número de conceptos

1.2 Número de atributos

1.3 Indicador de pruebas al sistema.

Variable 2: *Medicina Clásica*. Representa una variable independiente que se usará para elaborar tanto la base de conocimientos como la base de hechos del sistema experto.

Dimensiones

2.1 Síntoma

2.2 Enfermedad

2.3 Medicamento

Indicadores

2.1 Número de síntomas.

2.2 Grado de enfermedad.

2.3 Grado de cura.

Variable 3: *Sistema Experto para la Medicina Clásica.* Representa una variable dependiente que resulta de la asociación de una base de conocimiento que almacena el conocimiento del dominio necesario para resolver los problemas en base a probabilidades.

Dimensiones

3.1 Interfaz de usuario.

3.2 Mecanismo de inferencia.

3.3 Medio para la administración del conocimiento.

Indicadores

3.1 Uso de la interfaz.

3.2 Grado de mecanismo de inferencia.

3.3 Desempeño del formalismo de representación.

Variable 4: *Entorno de Programación Delphi.* Es una variable que interviene en el proceso de diseño del sistema experto. Más específicamente, es una herramienta que se utiliza para integrar a los principales componentes del sistema experto como son la base de conocimiento, el motor de inferencia y la interfaz de usuario.

Dimensiones

- 4.1 Manejo de la interfaz de usuario.
- 4.2 Administración de la base de conocimiento.
- 4.3 Administración del motor de inferencia.

Indicadores

- 4.1 Eficiencia en el manejo de la interfaz de usuario.
- 4.2 Eficiencia de la administración de la base de conocimiento.
- 4.3 Eficiencia en la administración del motor de inferencia

1.3.3 OPERACIONALIZACIÓN DE LAS VARIABLES

Para obtener la operacionalización de las variables de análisis se emplearán los instrumentos de investigación derivados de la metodología de la investigación científica. Las variables definidas en la sección anterior son operacionalizadas de acuerdo a lo que se muestra en la siguiente tabla:

VARIABLE	DIMENSION	INDICADORES	INSTRUMENTO
Sistema Experto	1.1 Conceptos	1.1 Número de conceptos	Ficha de observación
	1.2 Atributos	1.2 Número de atributos	Ficha de observación
	1.3 Comprensibilidad	1.3 Indicador de pruebas al sistema.	Experimento
Medicina Clásica	2.1 Síntoma	2.1 Número de síntomas.	Ficha de observación
	2.2 Enfermedad	2.2 Grado de enfermedad.	Experimento
	2.3 Medicamento	2.3 Grado de cura.	Experimento

Sistema Experto para la Medicina clásica	3.1 Interfaz de usuario. 3.2 Mecanismo de inferencia. 3.3 Medio para la administración del conocimiento.	3.1 Uso de la interfaz. 3.2 Grado de mecanismo de inferencia. 3.3 Desempeño del formalismo de representación.	Experimento Experimento Experimento
Entorno de Programación Delphi	4.1 Manejo de la interfaz de usuario. 4.2 Administración de la base de conocimiento. 4.3 Administración del motor de inferencia.	4.1 Eficiencia en el manejo de la interfaz de usuario. 4.2 Eficiencia de la administración de la base de conocimiento. 4.3 Eficiencia en la administración del motor de inferencia.	Observación directa del prototipo Observación directa del prototipo Observación directa del prototipo

Tabla 1.1: Operacionalización de las variables.

Fuente: [Elaboración Propia, 2012]

1.3.4 TIPO DE INVESTIGACIÓN

Esta investigación se caracteriza por ser inicialmente del tipo exploratoria pues uno de los objetivos propuestos es realizar el estudio de la IA enfatizando en el área de los SE. Si bien es cierto que existen dos trabajos de maestría anteriores (revisar VILCHEZ, Ricardo [VIL2003] y LEIVA, Jorge [LEI2010]) considero que no han abordado el estudio de la representación del conocimiento con la profundidad que se plantea en esta tesis. Por otro lado, en lo que se refiere al tipo de razonamiento utilizado, que es el probabilístico, luego de realizar el levantamiento de información he podido constatar que no ha sido tratado antes a nivel de posgrado UNI.

Luego, la tesis se enmarcó dentro de una investigación de carácter descriptivo, porque se tuvo que estudiar los factores etiológicos, los síntomas, signos y pruebas de laboratorio además del diagnóstico al que llega el médico para las tres patologías consideradas, a saber, candidiasis, tricomoniasis y vaginosis bacteriana así como la ingeniería del conocimiento.

Finalmente la investigación se caracteriza por ser cuasi experimental, debido a que no se aleatoriza o no se asigna al azar a la población sujeta a la intervención. En cierta forma se puede decir que la población se autoselecciona, obedeciendo a razones muy diversas y sobre las cuales el investigador no tiene ninguna influencia. Como producto de la investigación se ofrecerá un recurso informático, el sistema experto, a la cual la población va a acceder voluntariamente. Lo que se pretende es mostrar como el prototipo de sistema experto proporciona un diagnóstico adecuado en un dominio previamente delimitado, en relación con los médicos especialistas que se valen de diversos elementos heurísticos para la toma de decisiones.

1.3.5 ETAPAS DE LA INVESTIGACIÓN

A continuación se resumirá en una tabla las distintas etapas que ha seguido la investigación:

Etapa	Descripción	Actividades	Actores
Delimitación del problema de estudio	Definir cuál es el objeto de estudio y escoger una metodología adecuada al	Revisión a nivel nacional e internacional de tesis y artículos anteriores.	Investigador

	mismo		
Revisión teórica	Ubicar el objeto de estudio en el marco del conocimiento desarrollado en el área.	Consulta de fuentes bibliográficas accediendo a bibliotecas y centros de información y fuentes digitales realizando búsquedas en internet.	Investigador
Elaboración de instrumento	Establecer criterios organizados de los datos que se necesiten para la investigación	Elaboración de entrevistas no estructuradas	Investigador Médicos especialistas
Aplicación de instrumento	Acercarse a la realidad a través de la recolección de datos	Aplicar la entrevista	Investigador
Análisis de datos	Agrupar resultados de manera que permitan inferir la situación real	Estandarización y asociación de síntomas a enfermedades. Inferir diagnóstico mediante el modelo probabilístico propuesto.	Investigador

Redacción de conclusiones y elaboración del informe	de y del	Extraer resultados de investigación y ordenarlos en un todo coherente y comprensible	Elaboración y compaginación de la investigación final	Investigador
---	----------	--	---	--------------

Tabla 1.2: Etapas en el desarrollo de la investigación.

Fuente: [Elaboración Propia, 2012]

1.3.5.1 TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN

A continuación se describe los elementos más resaltantes que conforman esta fase de la investigación:

Dónde, cuándo y cómo obtendremos la información:

En nuestro caso se acordaron reuniones indicando lugar y hora con los médicos infectólogos y ginecólogos para la recopilación de la información.

Quién obtendrá los datos

Debido a cómo se llevó la investigación, los datos fueron recogidos por el propio investigador.

Con qué instrumento recogemos la información

En nuestro caso se usó la técnica de la entrevista personalizada no estructurada focalizada en la que se explicaba las hipótesis que se quería analizar.

1.3.5.1.1 MÉTODO

En esta investigación se utilizó el método hipotético deductivo en el que se parte de la observación de la medicina tradicional, se plantea una hipótesis y se hacen deducciones de conclusiones a partir de los conocimientos previos, apoyándonos en el paradigma de los sistemas basados en conocimiento y el entorno de programación Delphi.

1.3.5.1.2 TÉCNICA

Se utilizó la técnica de la entrevista personalizada no estructurada (ver Anexo A: Entrevista a un especialista), así como el llenado de formularios.

1.3.5.1.3 INSTRUMENTOS

Se utilizaron registros documentales. En estos registros se anotaron las características de las diferentes herramientas, técnicas y métodos de los dominios propuestos con la finalidad de efectuar un análisis que conducirá a la selección de las mejores características que ingresarán en el modelo que se propondrá como solución para la construcción del sistema experto probabilístico.

1.4 POBLACIÓN Y MUESTRA

1.4.1 POBLACIÓN

La población motivo de ésta investigación está conformada por todo el personal médico que labora en el departamento de Ginecología – Infectología de la Clínica Médica Cayetano Heredia.

1.4.2 MUESTRA

La muestra utilizada en la presente investigación, es del orden de 3 médicos especialistas, colaboraron preferentemente profesionales que sentían interés por participar en este proyecto interdisciplinario.

1.5 JUSTIFICACIÓN DE LA INVESTIGACIÓN

1.5.1 IMPORTANCIA DEL TEMA

La importancia de este trabajo se caracteriza por el hecho de que el prototipo de SE propuesto para el diagnóstico, puede ser visto como un auxilio precioso para los profesionales de salud pues no solo les sirve de apoyo sino como un medio de transmisión de conocimientos.

Una forma de hacer disponible el conocimiento y experiencia a personas consideradas novatas en un dominio de conocimiento es a través del desarrollo de un sistema experto. De este modo, es posible que personas sin la experiencia necesaria para la ejecución de una tarea especializada consigan obtener un resultado próximo o semejante al obtenido por un experto.

A través del desarrollo de este sistema experto, se hace posible la adquisición del conocimiento de un médico en lo que se refiere a la emisión de un diagnóstico, para que este conocimiento se logre difundir y esté accesible en casos donde no se dispone de un experto.

1.5.2 JUSTIFICACIÓN

1.5.2.1 TEÓRICA

La investigación busca por medio del marco teórico adquirido en asignaturas como Inteligencia Artificial y Estructuras Discretas así como conceptos básicos de matemática (probabilidades , inferencia lógica deductiva) y programación elaborar a mediano plazo un programa de computadora que sea capaz de inferir un

diagnóstico diferencial para una base de hechos conformada por la sintomatología presentada por el paciente.

Así, en un futuro, el investigador podrá contrastar los resultados devueltos por el sistema frente a lo diagnosticado por el médico especialista para evaluar el nivel de certeza y de ser el caso, hacer los ajustes correspondientes si es que la situación lo amerita. De alcanzarlo, se habría desarrollado una herramienta de gran aplicación en nuestro medio.

1.5.2.2 METODOLÓGICA

Con el fin de cumplir los objetivos expuestos, se llevarán a cabo entrevistas a diferentes profesionales de salud para elaborar un listado y estandarizar los síntomas más frecuentes que refiere un paciente al acudir a una consulta. Una vez recopilada esta información se procederá a desarrollar el motor de inferencia basado en la propagación de probabilidades. Para nuestra investigación nos apoyaremos en la Inteligencia Artificial convencional (Sistemas Expertos) .

Para la parte de programación se evaluarán como alternativas el uso de herramientas SHELL, que es un software especializado para el diseño de sistemas expertos, así como un lenguaje de programación que nos facilite un ambiente de desarrollo rápido a la vez que nos proporcione una interfaz agradable y finalmente se optará por uno de ellos.

Al culminar se debe pasar por un proceso de validación comparando los resultados inferidos por el programa con el diagnóstico definitivo proporcionado por el médico especialista.

1.5.2.3 PRÁCTICA

El resultado de esta investigación será de gran ayuda para instituciones de salud pues puede ser aplicable en los diversos servicios que la conforman como por ejemplo el área de emergencia, u hospitalización, además de acuerdo al grado de complejidad que se alcance ayudará a sugerir e incluso tomar decisiones en horarios críticos (como es el caso de feriados y días no laborables) pues como es evidente no hay un momento en que las instituciones de salud dejen de recibir pacientes. Este trabajo espera así contribuir a mejorar la organización y procedimientos actuales.

1.6 ALCANCES Y LIMITACIONES

El ámbito de la investigación se concentra en el estudio de los Sistemas Expertos probabilísticos. Previamente es de suma importancia revisar las diversas técnicas que existen para representar el conocimiento, entre estas cabe destacar a las reglas de producción, redes semánticas, tripletas OAV, marcos, árboles, redes bayesianas, entre otras, un conocimiento general nos permitirá determinar cuál de ellas resulta la más apropiada para el dominio de aplicación elegido. No se ha incluido en esta investigación la lógica difusa, desarrollada por Lofti Zadeh, la cual considero que sería de gran utilidad en el estudio de otras patologías infecciosas como es el caso del HIV.

Por otro lado, se ha abordado como dominio de aplicación las patologías infecciosas en el aparato reproductor femenino que son producidas por agentes etiológicos como bacterias o microbios y cuya sintomatología común es el descenso o flujo vaginal anormal.

CAPÍTULO II

MARCO TEÓRICO

2.1 INTELIGENCIA ARTIFICIAL

2.1.1 INTRODUCCIÓN

La Inteligencia Artificial (IA) busca estudiar las actividades mentales del ser humano para crear teorías y modelos cognitivos y proporcionar máquinas con la capacidad de realizar algunas de esas actividades. Las actividades realizadas por esas máquinas pueden comprender la senso-percepción (como tacto, audición, visión), las capacidades intelectuales (como aprendizaje de conceptos y de juicios, razonamiento deductivo y memoria), el lenguaje (como las orales y gráficas) y atención (decisión en el sentido de concentrar las actividades sobre un determinado estímulo).

2.1.2 DEFINICIÓN

La definición de IA ha sido motivo de mucha discusión entre los expertos en el tema. En el intento de definir con precisión lo que es la IA los especialistas adoptan las más diversas definiciones, haciendo la comprensión del asunto más compleja.

“Inteligencia Artificial es el resultado de la aplicación de técnicas y recursos especialmente de naturaleza no numérica, permitiendo la solución de problemas que exigirían del humano cierto grado de razonamiento y de experiencia. La solución de estos problemas con recursos típicamente numéricos es muy difícil ” [RAB1995].

Para WINSTON (1987), la IA es el estudio de conceptos que permiten a las computadoras ser más inteligentes.

La IA es una rama de las ciencias computacionales que se ocupa de los símbolos y métodos no algorítmicos para la resolución de problemas [PON2010].

ARNOLD Y BOWIE(1986) se remiten al diccionario, en el cual encuentran que Inteligencia es la capacidad de adquirir y de aplicar conocimientos, la facultad de pensar y de razonar. La tarea de acumular información. Y artificial es definido como aquello hecho por el hombre en vez de ocurrir en la naturaleza. Al asociar estas dos definiciones se consigue la siguiente definición de IA: *Capacidad de adquirir y aplicar conocimientos implementada por el humano.*

La IA es el estudio de cómo hacer que los computadores realicen tareas que, actualmente, son desempeñadas mejor por los seres humanos [RIC1993].

El campo de la IA intenta entender las entidades inteligentes mas al contrario de la filosofía y de la psicología que también se preocupan de la inteligencia, la IA se esfuerza para construir, así como entender, a las entidades inteligentes. [RUS1996].

IA es la ciencia e ingeniería de crear máquinas inteligentes, especialmente programas de computador inteligentes. Está relacionada a la tarea similar de usar computadoras para entender la inteligencia humana, pero IA no se limita a métodos que son biológicamente observables [MCA2007].

IA es el campo de conocimientos donde se estudian sistemas capaces de reproducir algunas de las actividades mentales humanas [NIL1995].

La investigación en IA se preocupa en desarrollar sistemas de computador que produzcan resultados que normalmente serían asociados a la inteligencia humana [HAR1988].

2.1.3 FUNDAMENTOS DE LA INTELIGENCIA ARTIFICIAL

La IA es heredera de diversas ideas, puntos de vista y técnicas de otras disciplinas. Entre sus fuentes tenemos a [RUS1996]

a) Filosofía

De los más de 2000 años de tradición de la filosofía, recibió teorías del conocimiento y del aprendizaje. A partir del siglo V a.c. la epistemología ha asumido tres vías:

1) La del **racionalismo** (de Platón, de Sócrates, de Descartes), con la primacía de lo abstracto, de lo absoluto, del a priori y de lo deductivo. Aristóteles trató de

explicar y codificar cierto tipo de razonamiento deductivo, que llamó silogismos. Formuló con mayor precisión las leyes que gobiernan la parte racional de la mente.

2) La del **constructivismo y del empirismo**(de los sofistas griegos, de Bacon, de Locke y Hume), representado por la percepción, por lo relativo, por el a posteriori, por la inducción: las reglas generales se obtienen por contacto con repetidas asociaciones entre sus elementos.

3) La del **dualismo**(de Aristóteles, Kant y Bertrand Russell), que sostiene que existe una parte de la mente que está al margen de la naturaleza, exenta de la influencia de las leyes físicas.

b) Matemática

De los más de 400 años de la matemática, la IA incluyó teorías de la lógica, de las probabilidades, toma de decisiones y de la computación.

El enfoque filosófico de la lógica se remonta a Aristóteles.

George Boole(1815-1864) introdujo un lenguaje formal que proporcionaba inferencias lógicas.

En 1879, Gottlob Frege estableció la lógica de predicados o lógica de primer orden, la cual constituye actualmente el sistema más básico de representación del conocimiento.

Kurt Godel (1906-1978), a través del teorema de la incompletez, demostró que, en cualquier lenguaje suficientemente capaz de describir las propiedades de los números naturales, existen afirmaciones cuya verdad no puede ser establecida por ningún algoritmo.

A mediados de 1960 Cobham introdujo la noción de intratabilidad, estableciendo la distinción entre problemas cuyo tiempo de resolución tiene un crecimiento polinomial o exponencial en función de la dimensión de las entidades involucradas. Un crecimiento exponencial implica la imposibilidad de resolver en un tiempo razonable casos de tamaño reducido.

En la misma década surgió en la teoría de la complejidad, el concepto de **reducción**, a través del cual es posible transformar problemas de una clase en otra, de manera que las soluciones de la primera clase puedan ser encontradas en la segunda, y, de ese modo, hacer posible la resolución de los problemas.

La noción de expresar un cálculo como un algoritmo formal tiene orígenes en Al-Jwarizmi, matemático árabe del siglo IX.

Thomas Bayes (1702-1761) propuso una regla para la actualización de las probabilidades subjetivas a la luz de las nuevas evidencias. La regla de Bayes constituye el fundamento sobre el que se apoya el enfoque moderno de abordar el razonamiento incierto en los sistemas de la IA.

La teoría de la decisión, iniciada por John Von Neumann(1944), combina la teoría de las probabilidades con la teoría de la utilidad.

c) Psicología

De la psicología, la IA aprovechó los procesos de investigación sobre la inteligencia humana. La psicología cognoscitiva afirma que el cerebro posee y procesa información.

d) Lingüística

De la lingüística, la IA recibió teorías de la estructura (la sintaxis) y del significado del lenguaje (la semántica). Por otro lado, ha existido una estrecha colaboración entre la lingüística y la IA en lo que se refiere a la representación del conocimiento.

e) Ciencia de la Computación

La IA ha recibido de la ciencia de la computación, el dispositivo y el software de desarrollo.

El computador fue reconocido como el dispositivo con la mayor capacidad de demostrar inteligencia. Hasta llegar en 1940 al primer computador operacional desarrollado por Alan Turing, lo antecedieron diversos instrumentos de cálculo.

El ábaco tiene una antigüedad de cerca de 7000 años. La máquina de Pascal, *la Pascalina*, que sumaba y restaba, fue creada en el siglo XVII.

El proyecto de Babbage de una máquina de computación programable (con memorias direccionables, almacenamiento de programas y saltos condicionales), fue concebido en el siglo XVIII. El primer programador del mundo es considerado Ada Lovelace, que diseñó programas para la máquina de Babbage.

La ciencia de la computación, en términos de software, proporcionó los sistemas operativos, los lenguajes de programación y las herramientas necesarias para escribir los programas actuales.

f) Ciencia Cognitiva

La ciencia cognitiva comenzó intentando explicar la inteligencia y el lenguaje humano de una forma racionalista, defendiendo que la representación simbólica es la base, necesaria y suficiente, de la inteligencia y considerando que el comportamiento inteligente es independiente del soporte en el que se desenvuelve (ya sea el cerebro o el computador) (Newell, Shaw, Simon, 1967). De acuerdo con esta teoría:

(1) Los sistemas inteligentes razonan sobre representaciones simbólicas del mundo real a través de procesos independientes del soporte donde son ejecutados.

(2) La resolución de problemas es el resultado de procesos de búsqueda en un espacio circunscrito de soluciones potenciales.

(3) La comunicación consiste en el intercambio de informaciones.

2.1.4 Síntesis Histórica de la Inteligencia Artificial

El siguiente cuadro presenta los hitos en el desarrollo de la IA a partir de Gödel.

[ERT2011]

1931	El austríaco Kurt Gödel demuestra que en la lógica de predicados de primer orden todos los enunciados verdaderos son derivables. En las lógicas de orden superior, por otro lado, hay ciertas declaraciones que son indemostrables Gödel demostró que la lógica de predicado extendida con los axiomas de la aritmética es incompleta.
1937	Alan Turing, señala los límites de las máquinas inteligentes, con el problema de la parada (halting problem) .
1943	McCulloch y Pitts modelan las <i>redes neuronales</i> y establecen la conexión a la lógica proposicional.
1950	Alan Turing define la inteligencia de la máquina con la <i>prueba de Turing</i> y escribe sobre el aprendizaje de las máquinas y los algoritmos genéticos.

1951	Marvin Minsky desarrolla una máquina de red neuronal. Con 3000 tubos de vacío simula 40 neuronas.
1955	Arthur Samuel (IBM) construye un programa de aprendizaje del ajedrez que juega mejor que su creador.
1956	McCarthy organiza una conferencia en la universidad de Dartmouth. Aquí, fue introducido por primera vez el nombre de <i>Inteligencia Artificial</i> . Newell y Simon de la Universidad Carnegie Mellon (CMU) presentan el Teórico Lógico, el primer programa de ordenador de procesamiento simbólico.
1958	McCarthy inventa en el MIT (Massachusetts Institute of Technology), el lenguaje de alto nivel LISP. El escribe programas que son capaces de modificarse a sí mismos.
1959	Gelemter (IBM) construye el demostrador de teoremas de geometría.
1961	El <i>Solucionador de Problemas Generales</i> (GPS) de Newell y Simon imita el pensamiento humano.
1963	McCarthy funda el Laboratorio de Inteligencia Artificial en la Universidad de Stanford.
1965	Robinson inventa el cálculo para la resolución de la lógica de predicados .
1966	El programa Eliza de Weizenbaum lleva a cabo el diálogo con personas en lenguaje natural.
1969	Minsky y Papert prueban en su libro <i>Perceptrones</i> que el perceptrón, una red neuronal muy simple, sólo puede representar funciones lineales.
1972	El científico francés Alain Colmerauer inventa el lenguaje de programación lógica PROLOG. Un médico británico de Dombal desarrolla un <i>sistema experto</i> para el diagnóstico de dolor abdominal agudo. Esto no pasa desapercibido en la corriente principal comunidad de inteligencia artificial del tiempo.
1976	Shortliffe y Buchanan desarrollan MYCIN, un sistema experto para el diagnóstico de enfermedades infecciosas, que es capaz de tratar con la incertidumbre.
1981	Japón comienza, a gran costo, el "Proyecto de Quinta Generación", con el objetivo de construir una máquina PROLOG poderosa.
1982	R1, el sistema experto para la configuración de ordenadores, ahorra a Digital Equipment Corporation (DEC) 40 millones de dólares por año.
1986	Renacimiento de las redes neuronales a través de Rumelhart, Hinton y Sejnowski entre otros. El sistema Ntalk aprende a leer textos en voz alta.
1990	Pearl, Cheeseman, Whittaker, Spiegelhalter introducen la teoría de probabilidades en la AI con las <i>redes bayesianas</i> . Los sistemas multiagente se vuelven populares.
1992	El programa TD-Gammon de Tesauro demuestra las ventajas de aprendizaje por refuerzo.
1993	Proyecto internacional <i>RoboCup</i> para promover la competencia de fútbol con robots autónomos.
1995	A partir de la teoría del aprendizaje estadístico, Vapnik desarrolla máquinas de soporte vectorial, que son muy importantes hoy en día.
1997	Computadora de ajedrez Deep Blue de IBM, derrota al campeón mundial

	de ajedrez Gary Kasparov. Primera competencia internacional RoboCup en Japón.
2003	Los robots en la <i>RoboCup</i> demuestran el nivel impresionante que la IA y la robótica son capaces de alcanzar.
2006	La robótica de Servicio se convierte en una importante área de investigación en IA.
2010	Los robots autónomos empiezan a aprender sus políticas.
2011	El programa "Watson", de comprensión del lenguaje natural y búsqueda de respuestas de IBM derrota a dos campeones humanos en concurso <i>Jeopardy!</i> de la televisión de EE.UU.

Tabla 2.1: Hitos en el desarrollo de la IA.

Fuente: [Ertel Wolfgang, 2011]

2.1.5 HISTORIA

La Inteligencia Artificial (IA) no surgió solamente como consecuencia del desarrollo de la informática. Su base teórica es resultado de siglos de estudios, donde la informática es un medio por el cual esos estudios pudieron ser aplicados y desarrollados.

Según el filósofo Hubert Dreyfus la historia de la Inteligencia Artificial pudo haber empezado por los 450 a.C. cuando Platón reportó el diálogo en que Sócrates pregunta a Eutidemo "Me gustaría saber cuáles son las características de la bondad que vuelven todas las acciones buenas... para que las utilice como un patrón a través del cual se pueda juzgar las acciones de otros hombres". O sea, Sócrates quería adquirir el conocimiento que Eutidemo poseía sobre la bondad de las acciones para que él también pudiese utilizar este conocimiento para distinguir

acciones buenas de las que no lo son. Esta es probablemente la más antigua descripción de un proceso de adquisición de conocimiento [RUS1996].

Los primeros estudios sobre el proceso de razonamiento son atribuidos a Aristóteles (384-322 a.C.), que intentó formular las leyes que gobiernan la parte racional de la mente, creando un sistema informal llamado *silogismos* que en principio permitían la generación mecánica de conclusiones dadas las premisas iniciales[PON2010].

Estudios más modernos sobre el razonamiento y el funcionamiento de la mente proporcionarán contribuciones importantes para la IA. René Descartes(1596-1650) afirmaba que todo entendimiento consiste en formar y utilizar representaciones simbólicas apropiadas. Decía sin embargo que los autómatas jamás se igualarían a los seres humanos en términos de habilidades mentales. A esto Kant adicionó la idea de que todos los conceptos son reglas, y Frege(1848-1925) mostró que las reglas podrían ser formalizadas para ser manipuladas sin intuición o interpretación. Leibniz, basado en la teoría materialista,(según la cual, todo el mundo, inclusive el cerebro y la mente, operan de acuerdo con leyes físicas), fue uno de los primeros en intentar crear un dispositivo mecánico con el objetivo de realizar operaciones mentales [RUS1996].

La idea de que el cerebro posee y procesa informaciones y su posible aplicación a la IA tiene un gran desarrollo con la psicología cognitiva. En 1943, Kenneth Craik especificó tres aspectos básicos de un agente basado en conocimiento: (1) El

estímulo debe ser traducido a una representación interna, (2) Esta representación se debe manipular por procesos cognitivos para derivar nuevas representaciones internas, y (3) estas son traducidas a una acción [RUS1996].

La formalización matemática también proporcionó un gran avance en la IA. George Boole(1815-1864), al formalizar la lógica de Aristóteles, proporcionó una herramienta que, al ser perfeccionada por Frege(1848-1825), se volvió la base de la mayoría de los sistemas de representación del conocimiento actuales. Otra gran contribución de la matemática a la IA es en el campo de la probabilidad, que proporciona teorías para el tratamiento de incertidumbre.

Para que la IA finalmente surgiese, fue necesario el surgimiento de un dispositivo que permitiese la aplicación de estas ideas. El desarrollo de este dispositivo comienza con el surgimiento del Ábaco, pasando por la máquina de suma y resta de Pascal y de la multiplicación de Leibniz, por la Máquina Analítica de Babbage, hasta llegar finalmente al computador moderno, en el inicio de la década del 40, con la construcción del primer computador operacional por Alan Turing, el *Heath Robinson*, construido con el objetivo de descifrar mensajes codificados alemanes durante la Segunda Guerra. [RUS1996]. Turing pasó a establecer una analogía entre el cerebro humano y los computadores. Esta analogía fue obtenida por psicólogos, neurofisiólogos e ingenieros electrónicos que percibieron que la forma como estaban dispuestas las células del cerebro (neuronas), ligadas a través de hilos nerviosos minúsculos, era semejante al circuito eléctrico de un computador. Al

finalizar la Segunda Guerra Mundial, los científicos ya tenían registrado importantes invenciones en el área electrónica y estudios sobre el cerebro humano desarrollados por médicos y psicólogos.

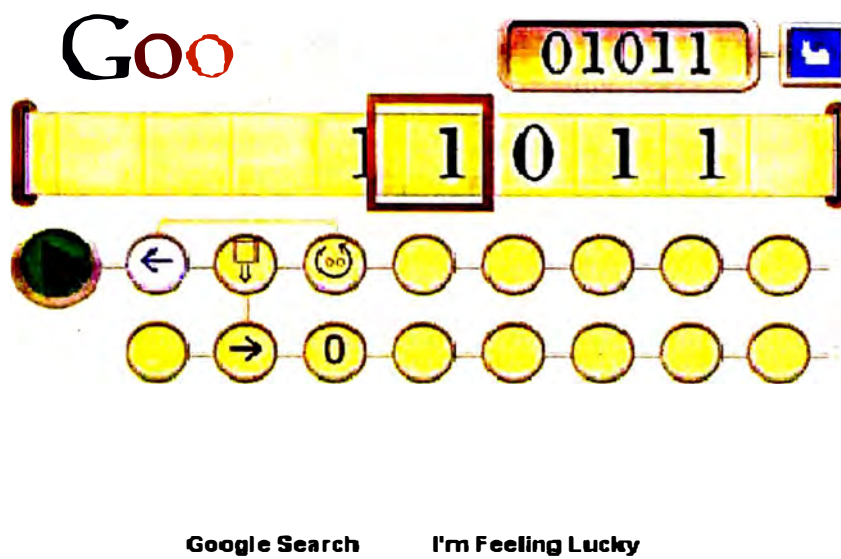


Figura 2.1: Doodle en homenaje al 100° aniversario del nacimiento de Turing.

Fuente: [Google,23 de Junio de 2012]

2.1.6 LA PRUEBA DE TURING

¿Cómo conoceremos cuando un programa de computador ha logrado una inteligencia equivalente a la de un humano?. **Alan Turing** (1912-1954), matemático inglés, fue uno de los fundadores de la Ciencia de Computación moderna e IA, cuyo logro intelectual hasta este día sigue asombrando en su extensión e

importancia. Fue el primero en ver al computador como un posible dispositivo inteligente, afirmando en mediados de la década del 40 que al finalizar el siglo XX existirían computadores inteligentes[PON2010].

Cuando llegó a examinar esta pregunta, él brillantemente resolvió el problema casi íntegramente. En su opinión, sería suficiente que un evaluador objetivo no pueda distinguir la diferencia en la conversación entre un humano y un computador para concluir que el computador es inteligente. Para eliminar cualquier parcialidad potencial del evaluador, la prueba de Turing (1950), ubica al evaluador en un cuarto equipado con un teletipo y a la máquina y al ser humano en otra, y hace que formule la prueba usando solo este dispositivo.

Si después de determinado tiempo el mediador no puede distinguir el computador del otro humano, entonces Turing percibe que se tiene que aceptar que el computador (o programa de computador en realidad) puede ser considerado inteligente [PON2010].

Otra consecuencia de la Prueba de Turing es que no refiere nada acerca de cómo uno construye un artefacto inteligente, así claramente evita discusiones sobre si el artefacto necesita imitar la estructura del cerebro humano o nuestro proceso cognitivo. En la mente de Turing no es realmente problema cómo el sistema fue construido. Su inteligencia sólo debería ser evaluada basada en su conducta [TUR1950].

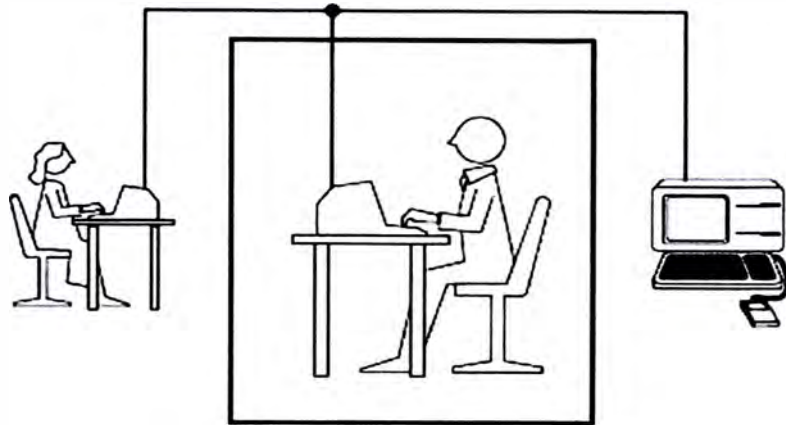


Figura 2.2: La prueba de Turing.

Fuente: [Turing,1950]

2.1.7 GÉNESIS DE LA INTELIGENCIA ARTIFICIAL

Se reconoce como el primer trabajo de Inteligencia Artificial el realizado en 1943 por *Warren McCulloch* y *Walter Pitts*, quienes propusieron un modelo de neurona artificial en el que cada una de ellas se caracterizaba por estar “encendida” o “apagada”. El “encendido” se daba como respuesta a la estimulación producida por una cantidad suficiente de neuronas vecinas. Estas neuronas nerviosas abstractas proporcionaron una representación simbólica de la actividad cerebral. Otro trabajo realizado por Frank Rosenblatt exploró el uso de redes llamadas perceptrones [RUS1996].

Más adelante, *Norbert Wiener* (1894-1964) elaboró estas ideas junto con otras, dentro del mismo campo, acuñando el término “cibernética” (tomado de un vocablo griego que significa el arte de gobernar un barco) para referirse *al control y comunicación en el animal y en la máquina*. La cibernética, la psicología cognitiva, la lingüística computacional y la teoría de control adaptable contribuyeron a la matriz intelectual de la Inteligencia Artificial [FON2007].

A inicio de los cincuenta *Claude Shannon* y *Allan Turing* se ocupaban de escribir programas de ajedrez para computadora al estilo de *Von Neumann*. Simultáneamente, *Marvin Minsky* y *Dean Edmonds* construyeron la SNARC, la primera computadora de red neuronal en 1951.

En el verano de 1956, Jhon McCarthy logró convencer a Minsky, Claude Shannon y Nathaniel Rochester para que le ayudaran a congregarse en el *Dartmouth College* (Estados Unidos) a investigadores interesados en la teoría de los autómatas, las redes neuronales y el estudio de la inteligencia. En total hubo diez asistentes que integraron el "*Grupo de Dartmouth*", que se reunió para realizar un estudio durante dos meses sobre el tópico de inteligencia artificial y discutir la posibilidad de construir máquinas "inteligentes". En ese taller, dos investigadores, Allen Newell y Herbert Simon acapararon la atención con un programa de razonamiento, el teórico lógico, TL, que era capaz de demostrar gran parte de teoremas de matemática.

En 1958, Jhon McCarthy¹ propuso utilizar el cálculo proposicional como un idioma para representar y utilizar el conocimiento en un sistema que denominó "*Advice Taker*" o "*Sistema Consejero*". A este sistema se le tenía que decir qué hacer en vez de ser programado.

Uno de los programas más influyentes fue el Solucionador General de Problemas, GPS, de Allen Newell, Cliff Shaw y Herbert Simon. El GPS posiblemente fue el primer programa que incorporó el enfoque de "*pensar como humano*".

A principios de los ochenta algunos programas que se desarrollaron contenían mayor capacidad y conocimientos para imitar el desempeño humano de expertos en varias tareas. El primer programa al que se le atribuye la demostración de la importancia de grandes cantidades de conocimiento y dominio específico es DENDRAL, un sistema de predicción de la estructura de las moléculas orgánicas

¹ Revisar **THE COMPUTER CHRONICLES**

que considera su fórmula química y el análisis de espectrograma de masa. Después se desarrollaron sistemas para diagnóstico médico, para configurar ordenadores y otros más.

En mayo de 1997 un programa de IBM llamado *Deep Blue* derrotó al campeón mundial de ajedrez *Garry Kasparov* por 3,5 a 2,5 en un encuentro a 6 partidas. Por otra parte, Larry Roberts desarrolló uno de los primeros programas de análisis de escena. Este trabajo fue seguido de una extensa cantidad de investigación en visión artificial. Otro proyecto que se puede mencionar es CYC, una de cuyas metas era recolectar e interpretar gran cantidad de información para su conocimiento. El interés por las redes neuronales iniciado por los trabajos pioneros de Frank Rosenblatt se reanudó con energía en los años ochenta.

Si proyectamos las tendencias actuales hacia el futuro, es razonable esperar un desarrollo de robots y *softbots*, que son agentes de software que recorren Internet buscando información que ellos creen que puede ser de interés para sus usuarios [NIL1995].

2.1.8 LÍNEAS DE INVESTIGACIÓN DE LA IA

Para resolver un problema es necesario tener algún conocimiento del dominio del problema y utilizar alguna técnica para obtener la solución.

Como métodos de enfoque para la solución de problemas se puede citar: IA Simbólica (IAS), IA Conexionista (IAC), IA Evolucionaria (IAE) y se puede también considerar una cuarta área que sería la Híbrida (IAH).

2.1.8.1 IA SIMBÓLICA

Este paradigma da énfasis a la forma como el ser humano razona. Trata acerca de la manipulación de símbolos que representan el conocimiento. Puede ser utilizada cuando el problema está bien definido, es decir cuando el modo de hallar el problema sea explícito. Este paradigma surgió gracias al interés de un grupo de científicos, en particular, psicólogos por buscar desarrollar programas de computador que simulasen el comportamiento humano.

Los proyectos de esta área se dividen en dos grandes grupos: aquellos con conocimiento basado en reglas, sistemas capaces de establecer conclusiones, basándose en las reglas utilizadas por expertos humanos, para resolver problemas objetivos y de dominio específico, y los basados en casos, en que el objetivo es retirar conocimiento a partir de ejemplos o casos paradigmáticos. La filosofía básica de ésta técnica es la de buscar la solución para una situación actual a través de la recuperación de la solución de una experiencia pasada semejante. El proceso consiste en: identificar la situación actual, buscar la experiencia más semejante en la memoria y aplicar el conocimiento de esta experiencia pasada en la situación actual. En el primer grupo los sistemas construidos son conocidos por una serie de denominaciones como: "Sistemas Basados en Reglas", "Sistemas Basados en Conocimiento", "Sistemas de Producción", "Sistemas de Inferencia Dirigidos a Patrones". En el segundo caso los sistemas son conocidos como "Razonamiento Basado en Casos".

2.1.8.2 IA CONEXIONISTA

La IA Conexionista se inició en 1943, cuando el neurofisiólogo Warren McCulloch y el lógico Walter Pitts desarrollaron el primer modelo matemático de una neurona. Se espera de la IAC un mejor desempeño que el de la IAS en problemas que no

estén bien definidos, donde falta el conocimiento específico de cómo realizar una tarea.

En la visión de la IA Conexionista se acredita que construyendo un sistema que simule la estructura cerebral humana, éste presentará inteligencia y será capaz de aprender, asimilar, errar y, con esta experiencia, adquirir nuevos conocimientos. De esta forma, las Redes Neuronales Artificiales, esquemas basados en la IA Conexionista, tienden a representar una estructura interconectada similar a la estructura del cerebro humano y con esto resolver situaciones que no son fáciles de resolver en la computación convencional, donde la informática es transferida lineal y sincronizadamente. Así, las Redes Neuronales se vuelven de fundamental importancia en el desarrollo de los sistemas de IA donde hay necesidad de resolución de problemas no lineales, así como en el reconocimiento de patrones (imagen, sonido, diagnóstico de fallas y control de procesos), en la optimización y en el control de movimiento de los miembros en robots.

2.1.8.3 IA EVOLUCIONARIA

La Inteligencia Artificial Evolucionaria (IAE), se deriva del modelo biológico de aprendizaje, trata con problemas bien definidos. Atribuye inteligencia al comportamiento de poblaciones que cambian sus características para adaptarse mejor al medio. Los elementos que poseen una performance débil son descartados, en cuanto los elementos más fuertes se multiplican, produciendo variaciones de sí mismos. Este paradigma posee una analogía directa con la teoría de Darwin, en la cual sobreviven los mejor adaptados al ambiente. La evolución biológica es un ejemplo de solución de problema bien definido de supervivencia de una especie en ambiente variable.

2.1.8.4 IA HÍBRIDA

La IAH utiliza los puntos positivos de todos los tipos de paradigmas de la IA para alcanzar una mejor solución para los problemas. Un tipo de aplicación de la IAH son los sistemas neuro-fuzzy, donde se unen los conocimientos de redes neuronales, expresadas a través de la interconexión de unidades de procesamiento simple, con la representación de conocimientos impresos obtenidos a través de los conjuntos difusos [BAR1997] .

2.1.9 PRINCIPALES APLICACIONES

Dentro de la computación, la Inteligencia Artificial es responsable por importantes aplicaciones que siempre exigirán una alta dosis de esfuerzo. Dentro de estas aplicaciones, se puede destacar: Procesamiento del Conocimiento, Procesamiento del Lenguaje Natural, Reconocimiento de Patrones, Robótica, Juegos, Base de Datos Inteligentes y Visión por Computador.

2.1.9.1 Procesamiento del Conocimiento

Es el que se refiere al almacenamiento y manipulación del conocimiento por la máquina, de tal forma que pueda ser utilizado para la resolución de problemas, como es hecho por los sistemas expertos. Estos son sistemas que proporcionan conclusiones expertas sobre asuntos especializados. Uno de los principales campos de aplicación de estos sistemas es la medicina, desarrollándose en ésta área el más conocido de los sistemas expertos, el Mycin en 1974, el cual diagnosticaba infecciones bacteriales de la sangre y sugería tratamiento. Tienen también aplicaciones en las demás áreas del conocimiento como en la matemática, química e ingeniería, entre otras [RAB1995] .

2.1.9.2 Procesamiento de Lenguaje Natural (PLN)

Es el entendimiento, por el computador, del lenguaje común usado por los humanos. El lenguaje escrito está dominado en ciertos aspectos sin embargo persisten serios problemas relativos al lenguaje figurado, doble interpretación y otros. El lenguaje hablado ya está siendo producido de forma razonable, aunque todavía hay problema muy serio en cuanto a la forma humana de hablar, corrigiendo las palabras unas a otras [RAB1995].

2.1.9.3 Reconocimiento de Patrones

El reconocimiento de patrones trata de tareas como reconocer una impresión digital, validar firmas de personas, leer y digitalizar textos escritos, etc. Esta es un área de la IA muy relacionada en otras áreas, como el lenguaje natural y en la robótica [RAB1995].

2.1.9.4 Robótica

La Robótica, aliada a la ingeniería, busca implementar las funciones de movimiento, percepción y control de la máquina. Existen robots² puramente mecánicos, que reproducen apenas tareas y movimientos implícitos en su construcción. Hay robots que complementan la parte mecánica con dispositivos electrónicos que constituyen una especie de cerebro, donde son almacenados conocimientos, los cuales pueden dar cierto grado de autonomía a estas máquinas. Este último tipo de robots, generalmente, están siendo utilizados en ambientes que son hostiles al hombre, como los viajes espaciales y actividades de prospección del petróleo. La parte del almacenamiento de conocimientos y su ejecución es semejante a la efectuada en sistemas expertos [RAB1995].

² La expresión robot viene del checo *robotá*, que significa trabajador, fue creada por *Karel Capek*, en 1917. El primer robot industrial del mundo, bautizado como *UNIMATE*, surgió en 1962.

2.1.9.5 Juegos

Los juegos tienen en general un número pequeño de reglas que varían poco, y una gran posibilidad de combinación y replicación de estas reglas, lo que los hace complejos.

Uno de los juegos más estudiados y experimentados en el computador es el ajedrez, donde hay áreas en que el computador ya es capaz de realizar análisis y pronósticos más profundos que el ser humano.

2.1.9.6 Base de Datos Inteligentes

Una Base de Datos Inteligente es aquella capaz de hacer razonamientos y producir resultados, propiedad que la mayoría de las bases de datos de los sistemas de información no poseen. El uso de técnicas de IA para la creación de Bases de Datos Inteligentes aumentará la productividad y funcionalidad de los sistemas de información, ya que las avanzadas técnicas de IA beneficiarían la recuperación de informaciones [RAB1995].

2.1.9.7 Visión por computador

El mundo está conformado de objetos tridimensionales, pero las entradas al ojo humano y las cámaras de computador son bidimensionales. Algunos programas útiles pueden trabajar únicamente en dos dimensiones, pero la visión por computador amplia requiere de información parcial tridimensional que no es sólo un conjunto de vistas bidimensionales. En la actualidad hay solo maneras limitadas de representar información tridimensional directamente y no son tan buenas como la que evidentemente usan los humanos.

2.1.9.8 Prueba de Teoremas

Es una aplicación típicamente matemática pero que es interesante como metodología de solución de problemas. No requiere sólo la habilidad de levantar hipótesis sino también exige experiencia intuitiva, buscando resultados probados anteriormente ayudando así en la prueba del teorema principal [RAB1995].

2.2 SISTEMAS EXPERTOS

2.2.1 SISTEMAS BASADOS EN CONOCIMIENTO

Los Sistemas Basados en Conocimiento (SBC) están siendo utilizados cada vez más tanto a nivel comercial como científico. Uno de los puntos clave de la construcción de los SBC está en la capacidad de estos sistemas de almacenar, aprovechar y hacer uso del conocimiento humano en el proceso de la toma de decisiones.

Actualmente existen diversas aplicaciones para SBC, entre ellas las áreas de ingeniería, ciencias, medicina, etc.

2.2.2 LA CADENA DICS

Datos, información, conocimiento y sabiduría son elementos importantes del proceso de pensamiento y razonamiento humano. Sin embargo hay marcadas diferencias entre estos. Los *datos* se refieren a la observación y hechos crudos. Ellos son inútiles sin un procesamiento adicional, vale decir, comparar, inferir, filtrar, etc. A los datos procesados se les conoce como *información*. Podemos concluir que el *conocimiento* es el resultado de procesos como la síntesis, el filtrado, la comparación y el análisis de la información disponible para generar resultados significativos. Con el tiempo, la experiencia, el juicio, los valores, leyes, etc. se van agregando para alcanzar la sabiduría. Esto se conoce como la *cadena datos – información – conocimiento - sabiduría* (DICS) . Esta cadena también se conoce como *pirámide de datos*. Estas entidades pueden ser dispuestas como se muestra en la siguiente figura [SAJ2010].

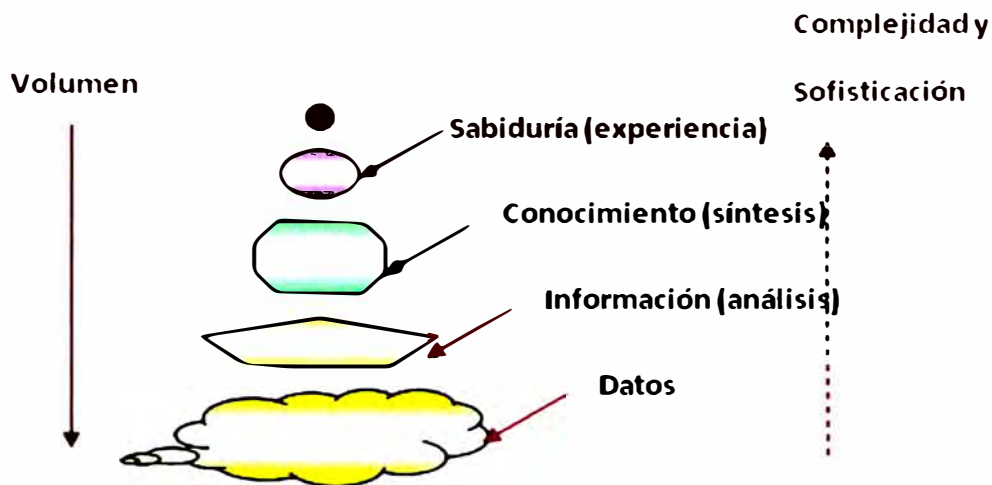


Figura 2.3: La cadena DICS.

Fuente: [Sajja, 2010]

El conocimiento se puede clasificar de diferentes maneras. En la siguiente tabla se presenta brevemente los distintos tipos de conocimiento.

Tipo de Conocimiento	Descripción
Conocimiento del dominio	El conocimiento del dominio es el conocimiento válido para un dominio específico. Especialistas y expertos desarrollan sus propios conocimientos del dominio y lo utilizan para la resolución de problemas
Meta Conocimiento	Meta Conocimiento se puede definir como el conocimiento acerca del conocimiento.
Conocimiento de sentido común	El conocimiento de sentido común es un conocimiento de propósito general que se espera esté presente en cada ser humano normal. Las ideas del sentido común tienden a relacionarse con los acontecimientos dentro de la experiencia humana.
Conocimiento heurístico	La heurística es una regla específica o un argumento derivado de la experiencia.
Conocimiento explícito	El conocimiento explícito se puede expresar fácilmente con palabras y números y se comparte en forma de datos, fórmulas científicas, especificaciones de productos, manuales, y los principios universales. Es más formal y sistemático.
Conocimiento tácito	El conocimiento tácito es el conocimiento

	<p>almacenado en el subconsciente de los expertos y no es fácil de documentar. Es muy personal y difícil de formalizar, y por lo tanto difícil de representar formalmente en el sistema.</p> <p>Percepciones subjetivas, intuiciones, emociones, modelos mentales, valores y acciones son ejemplos de conocimiento tácito</p>
--	---

Tabla 2.2: Tipos de Conocimiento.

Fuente: [Sajja, 2010]

2.2.3 CONCEPTO DE SBC

Un SBC es un sistema informático, que utiliza y genera conocimiento a partir de datos, información y conocimiento. Estos sistemas son capaces de entender la información en el proceso y pueden tomar decisiones en base a la información y/o conocimiento que se encuentre en el sistema, mientras que los sistemas informáticos tradicionales no conocen o no entienden los datos y/o información que procesan [SAJ2010].

Los SBC utilizan el conocimiento representado explícitamente para resolver problemas donde es necesaria una cantidad considerable de conocimiento humano y de especialización. En los SBC, el conocimiento y el proceso de resolución de problemas son dos aspectos claves en el desarrollo de los mismos.

El proceso de resolución de problemas posee dos tipos de operaciones: la primera es la capacidad de razonamiento, es decir, como se llega a ciertas conclusiones(o se genera un nuevo hecho) interpretando el conocimiento adquirido hasta el momento y la segunda es guiar el proceso de razonamiento. Para resolver problemas de una manera rápida y clara es necesario guiar el proceso de razonamiento de modo que sólo sean consideradas las conclusiones relevantes al problema en cuestión. Esta capacidad es denominada método para resolver

problemas y el razonamiento es conocido como *estrategia de razonamiento* o *estrategia de inferencia*.

Los SBC deben poseer la funcionalidad de cuestionar al usuario para reunir las informaciones que necesita, desarrollar una línea de razonamiento a partir de las informaciones recopiladas; explicar el modo de razonamiento y poseer un desempeño satisfactorio que compense sus posibles errores en caso sucedan.

De acuerdo a la clasificación de Tuthill & Levy(1991), existen cinco tipos de SBC [SAJ2010]

- (i) Sistemas Expertos,
- (ii) Sistemas de Manipulación de Hipertextos
- (iii) Sistemas Basados en Casos
- (iv) Bases de datos en conjunción con Interface de Usuario Inteligente y
- (v) Sistemas Tutores Inteligentes

Es necesario diferenciar los SBC de los Sistemas Expertos. Los SBC son capaces de resolver problemas usando conocimiento específico sobre el dominio de la aplicación, mientras que los SE son SBC que resuelven problemas que comúnmente son resueltos por un experto humano. Los SE requieren conocimiento sobre la habilidad, la experiencia y las heurísticas usadas por los expertos, siendo la interacción con los expertos en el proceso de desarrollo de fundamental importancia.



Figura 2.4: Relación entre SBC y SE
Fuente: [Rezende, 2003]

2.2.4 INTRODUCCIÓN A LOS SE

Los sistemas expertos son una rama de la IA que hace uso del conocimiento especializado para resolver problemas como un **especialista** humano. Éste es una persona que tiene experiencia desarrollada en cierta área; su habilidad es superior a la mayoría, puede resolver problemas que la mayoría no podría resolver o los resuelve con mucha mayor eficiencia. El conocimiento de los SE puede obtenerse por experiencia o consulta de los conocimientos ya sea en libros, revistas o con personas capacitadas [GIA1998].

Los términos *sistemas expertos*, *sistemas expertos basados en conocimiento* se usan como sinónimos. El conocimiento de un especialista se centra específicamente en el **dominio del problema**. Un dominio del problema es el área específica del problema como medicina, finanzas, ciencias, ingeniería, etc. en el que un especialista puede resolver problemas con facilidad. Los SE suelen diseñarse especializados en un dominio del problema.

Al conocimiento del especialista para resolver problemas específicos se le llama **dominio del conocimiento** del experto. Por ejemplo, un SE médico diseñado para

diagnosticar enfermedades infecciosas debe tener una gran cantidad de conocimiento acerca de los síntomas causados por este tipo de enfermedades. En este caso el dominio del conocimiento es la medicina y consta de las enfermedades, sus síntomas y tratamientos.



Figura 2.5: Un probable problema y su relación con el dominio del conocimiento

Fuente: [Giarratano y Riley, 1998]

Un SE médico normalmente no tiene conocimiento acerca de otras ramas de la medicina como cirugía o pediatría. En su dominio de conocimiento, el SE razona o hace inferencias de la misma forma en que un especialista humano inferiría la solución de un problema, es decir, dados algunos hechos, se infiere una conclusión [GIA1998].

2.2.1 DEFINICIÓN DE SE

El profesor *Edward Feigenbaum*³ de la Universidad de Stanford ha definido un Sistema Experto como “un programa inteligente para ordenador que utiliza conocimiento y procedimientos inferenciales en la resolución de problemas, problemas que son lo suficientemente difíciles como para que su solución requiera una experiencia humana importante” [HAR1988].

³ Ver THE COMPUTER CHRONICLES

A continuación presentamos otras definiciones de Sistemas Expertos:

- Los Sistemas Expertos son sistemas informáticos para aplicaciones especializadas que requieren experiencia, generalmente basada en técnicas de Inteligencia Artificial (IA) [MOH2000].
- Los Sistemas Expertos son sistemas computacionales que resuelven problemas de una manera bastante parecida que el especialista humano. Son sistemas con un conocimiento específico profundo sobre campos limitados del conocimiento [RAB1995].
- Un Sistema Experto es un sistema capaz de realizar una tarea que generalmente se considera que es difícil y que requiere cierto grado de experiencia humana [FRO1989].
- Un Sistema Experto es básicamente un conjunto de programas informáticos que aplica el proceso del razonamiento humano al conocimiento de un experto en la solución de tipos específicos de problemas. Son sistemas basados en reglas de producción u otros procesos de razonamiento [PAJ2006].
- Un Sistema Experto es un sistema computarizado que utiliza ampliamente el conocimiento basado en la experiencia en un determinado tema para solucionar tópicos de manera inteligente, o sea, es un programa de computador proyectado para modelar la habilidad de resolver problemas de un experto humano (Durkin, 1994).

2.2.6 PRIMEROS SISTEMAS EXPERTOS

Esta sección presenta una breve lista de algunos de los principales SE, que han tenido un impacto en (y llevado al desarrollo de) otros SE.

DENDRAL

Fue desarrollado en la Universidad de Stanford en 1965, donde *Edward Feigenbaum*, *Bruce Buchanan* (filósofo convertido en científico de la computación) y *Joshua Lederberg* (genetista ganador del premio nobel) colaboraron en la solución del problema de inferir una estructura molecular a partir de la información proporcionada por un espectrómetro de masas. El programa se alimentaba con la fórmula elemental de la molécula, el espectrómetro de masas informaba sobre las masas de los diversos fragmentos de la molécula que se producían después de ser bombardeada con un haz de electrones [RUS1996].

Dendral se considera el primer sistema experto ya que automatiza el proceso de toma de decisiones y facilitaba la inferencia de estructuras moleculares. Se compone de dos subprogramas: *Dendral Heurístico* y *Meta-Dendral*. Fue escrito en LISP (lenguaje de programación), considerado como el lenguaje de la IA. Muchos sistemas se obtuvieron a partir de Dendral incluyendo MYCIN, MOLGEN, MACSYMA, PROSPECTOR, XCON Y STEAMER [RUS1996].

La trascendencia de Dendral fue la de ser el primer sistema de conocimiento intensivo que lograba funcionar: sus conocimientos se basaban en importantes cantidades de reglas para propósitos especiales, sin embargo su implementación no separaba de forma explícita el conocimiento del motor de inferencia. Dendral originalmente provenía de *DENDRitic ALgorithm*, un procedimiento que era el corazón del programa Dendral [LIN1980].

MACSYMA

Desarrollado en 1969 por Martin y Moses, este sistema para manipulación de fórmulas matemáticas ilustra el proceso de desarrollar SE basados en reglas para dominios bien definidos.

MYCIN

Feigenbaum, Buchanan y el doctor Edward Shortliffe diseñaron en 1973 el programa MYCIN para el diagnóstico de las infecciones sanguíneas. Con un respaldo de 500 reglas, MYCIN era capaz de hacer diagnósticos tan buenos como los de un experto y desde luego, mejores que los de un médico recién graduado [RUS1996].

La experiencia de MYCIN radica en el dominio de las infecciones bacterianas. Los médicos, en general, inician un tratamiento con antibióticos en pacientes que tienen infecciones bacterianas sin saber exactamente qué organismo ocasiona la enfermedad. No hay tiempo para esperar las pruebas definitivas de un análisis de laboratorio, ya que estas se acumulan con excesiva lentitud. Para una persona muy grave, la terapia debe iniciarse en seguida y no dos días después. Esta necesidad ofrece dos alternativas: el médico puede recetar un medicamento de amplio espectro que cubra todas las posibilidades o bien recetar un medicamento mejor, específico para la enfermedad [WIN1994].

MYCIN ayuda al médico a recetar medicinas específicas para una enfermedad. MYCIN se informa sobre casos particulares, requiriendo información del médico acerca de los síntomas del paciente, su condición general, historia clínica y resultado de análisis de laboratorio que se puedan obtener fácil y rápidamente. En cada punto, la pregunta que hace MYCIN está determinada por las hipótesis actuales que posee y las respuestas que ha dado a todas las preguntas anteriores.

Por tanto el cuestionario comienza como si se tomara de una lista de verificación, pero las preguntas varían conforme se van generando las pruebas.

La fuente de conocimiento de MYCIN consiste en aproximadamente 500 reglas de antecedente – consecuente, que le dan la capacidad de reconocer alrededor de 100 causas de infecciones bacterianas. MYCIN es un sistema de encadenamiento regresivo porque los médicos prefieren pensar en una hipótesis a la vez. Al apegarse a las preguntas relevantes a una conclusión hipotética en particular se garantiza que el cuestionario seguirá siendo relevante para dicha hipótesis. Un sistema que se desplaza hacia adelante puede tener saltos, trabajar primero hacia una conclusión y después hacia otra como si obrara al azar [WIN1994].

MYCIN se distinguía del Dendral principalmente en dos aspectos. En primer lugar, a diferencia de las reglas del Dendral, no se contaba con un modelo teórico desde el cual pudieran deducirse las reglas de MYCIN. Era necesario obtenerla a partir de amplias consultas con los expertos, quienes las habían obtenido de su experiencia directa en diversos casos. En segundo lugar, las reglas deberían reflejar la incertidumbre inherente al conocimiento médico. En MYCIN, se contaba con un cálculo de incertidumbre denominado *factores de certeza*, que al parecer (en esa época) correspondía muy bien a la manera cómo los médicos ponderaban las evidencias al hacer un diagnóstico [RUS1996]. En 1980 se introdujeron varios shell de SE, entre ellos EMYCIN para diversas aplicaciones, que se obtiene por extracción MYCIN de su base de conocimiento médico [MOH2000].

HEARSAY-II

Desarrollado en 1973 por Reddy, este sistema de reconocimiento de voz propuso lo que ha llegado a conocerse como el "modelo de pizarra" con el uso de múltiples (distribuidos) fuentes de conocimiento (agentes) que cooperan para resolver una tarea compleja estructurada jerárquicamente [MOH2000].

PROSPECTOR

Desarrollado en 1976 por Duda, Gaschnig y Hart, este fue un sistema para exploración mineral que usaba una representación de conocimiento estructurada jerárquicamente, aplicando métodos probabilísticos (con hipótesis de independencia) para la toma de decisión práctica [MOH2000].

R1(XCON)

Desarrollado en 1977 por McDermott, este SE orientado por objetivo configuraba sistemas de computadora para Digital Equipment Corporation, ilustrando con mayor claridad los ahorros cuantificables (muchos minutos por orden, lo que se traduce en millones de dólares por año) que resultaban del uso de los SE.

Además del desarrollo de metodologías específicas, tales como programación basada en reglas, la contribución más importante de estos primeros sistemas es el concepto de separar el control del conocimiento.

Este es un principio muy valioso que hace que sea posible aplicar la misma metodología para un gran número de problemas, proporcionando a menudo plantillas de diseño para nuevos problemas. Lo que se debe evitar es el intento de resolver todos los problemas a partir de cero, desechando la experiencia obtenida anteriormente en la construcción de sistemas expertos para otras aplicaciones [MOH2000].

2.2.7 CARACTERÍSTICAS DE LOS SE

Un SE debe poseer las siguientes características generales [GIA1998]:

Alto desempeño: La capacidad de respuesta de un SE debe ser de un nivel de competencia igual o superior al de un especialista en el campo.

Tiempo de respuesta adecuado: El SE debe tomar una decisión en un tiempo razonable comparable o mejor al tiempo requerido por un especialista.

Confiabilidad: El SE no debe ser propenso a caídas, sino corre el riesgo de no ser usado.

Comprensible: El SE debe ser capaz de explicar los pasos de su razonamiento mientras se ejecutan, de modo análogo a como los especialistas pueden justificar como llegaron a cierta conclusión.

Cabe resaltar que en un SE el curso de la ejecución no es secuencial (el orden en el que se introducen las reglas al sistema no es necesariamente el mismo en que se ejecutarán). El SE actúa como un programa paralelo en que las reglas son procesadores de conocimiento independiente.

Flexibilidad: Un SE debe contar con un mecanismo eficiente para añadir, modificar y eliminar conocimiento.

2.2.8 TIPOLOGÍA DE SISTEMAS EXPERTOS

De acuerdo a los tipos de problemas que los SE trabajan pueden ser agrupados en las siguientes categorías:

Sistemas de Interpretación: Los sistemas de interpretación se utilizan principalmente para explicar los datos, proporcionando significados simbólicos adecuados y describiendo la situación y/o el estado que representan los datos. Esta categoría incluye la vigilancia, análisis de imágenes satelitales, interpretación geológica, la comprensión del habla, el análisis de la estructura química y la interpretación de señales.

Por ejemplo, *PROSPECTOR* es un sistema experto geológico, y *HEARSAY II* se refiere a la comprensión del habla [DAV1998].

Sistemas de Predicción: Se utilizan para inferir posibles consecuencias de una situación dada. Las aplicaciones incluyen la previsión meteorológica, las predicciones demográficas, las predicciones de tráfico, las estimaciones de los cultivos, y la previsión militar.

Por ejemplo, *SPERIL* se desarrolló en la Universidad de Purdue para analizar las estructuras por un posible daño del terremoto [DAV1998].

Sistemas de Diagnóstico: Deducen posibles problemas a partir de la evidencia observada o a partir de síntomas y signos [LAR1992]. Estos sistemas combinan el conocimiento de diseño de sistemas con conocimientos de diseño, implementación, o fallas de componentes potenciales para el diagnóstico de averías o recomiendan una mayor investigación. Las aplicaciones se encuentran en la medicina y en el diseño electrónico, mecánico, y de software.

Por ejemplo, *DART* fue desarrollado conjuntamente por la Universidad de Stanford e IBM para determinar los fallos informáticos, y *MYCIN* fue desarrollado en la Universidad de Stanford para diagnosticar infecciones en la sangre [DAV1998].

Sistemas de Diseño: Desarrollan las configuraciones que resuelven un problema ajustándose a las restricciones conocidas [LAR1992]. Estos sistemas incorporan a menudo comportamiento orientado a objetivos e intentan minimizar una función

objetivo y lograr la meta. Las aplicaciones incluyen el diseño de circuitos, diseño de edificios, y presupuestos.

Por ejemplo, *SYN* se ha desarrollado en el MIT para realizar el análisis de circuitos electrónicos [DAV1998].

Sistemas de Monitoreo: Comparan observaciones de comportamiento de sistemas para detectar posibles desviaciones de la norma y tomar medidas correctivas con toda oportunidad [LAR1992]. Las aplicaciones incluyen la gestión fiscal, reglamentación y cumplimiento, el monitoreo de enfermedades y control del tráfico aéreo.

Por ejemplo, *IMS* ha sido desarrollado por la Universidad Carnegie Mellon para realizar el seguimiento automatizado de la fábrica y *CALISTO* fue desarrollado por Digital Equipment Corporation para llevar a cabo actividades de gestión de proyectos [DAV1998].

Sistemas de Planeación: Idea acciones interrelacionadas para obtener el resultado deseado, descompone la tarea en un subconjunto de tareas. Incluye diseño de rutas, de proyectos, de experimentos, etc [LAR1992]. Las aplicaciones incluyen la programación automatizada, la robótica, el diseño de rutas, la comunicación de datos y la planificación militar.

Por ejemplo, *PECOS* se desarrolló en Stanford para modelar la exploración de petróleo e *ISIS-II* ha sido desarrollado por la Universidad Carnegie Mellon para programar el trabajo de la tienda [DAV1998].

Sistemas de Depuración: Se utilizan para prescribir remedios para el mal funcionamiento y/o soluciones para fallas [LAR1992]. Las aplicaciones incluyen la prescripción médica a un paciente, depuración de programas de computadora, desarrollo de software, edición de texto, sistemas de pregunta y respuesta, el procesamiento del lenguaje natural y la instrucción asistida por computadora [DAV1998].

Sistemas de Reparación: Estos sistemas desarrollan y ejecutan planes para corregir fallas de toda índole [LAR1992]. Las aplicaciones incluyen las redes de comunicaciones y el mantenimiento de computadoras. Los sistemas de reparación son relativamente nuevos, y las implementaciones de éxito están empezando a emerger [DAV1998].

Sistemas de Instrucción: Son los sistemas que ayudan en el aprendizaje, haciendo un diagnóstico de la situación de cada alumno. Dependiendo de la evolución del aprendiz aumenta la complejidad de las situaciones didácticamente hasta el nivel intelectual deseable. Enseñanza inteligente para que un estudiante pueda preguntar *Por qué, Cómo y Qué pasaría si*, como lo haría un maestro humano, por ejemplo educar un estudiante de medicina [LAR1992]. Las aplicaciones incluyen la enseñanza de los estudiantes y sistemas de explicación del comportamiento animal. Por ejemplo, GUIDON fue desarrollado por la Universidad de Stanford para proporcionar instrucción médica y WUMPUS fue desarrollado por el MIT para ofrecer instrucción de entrenamiento atlético [DAV1998].

Sistemas de Control: Sirven para regular un proceso, tiene como tarea el cumplimiento de un objetivo dado. Requieren interpretar una situación actual, predecir el futuro, diagnosticar las causas de los problemas que se puedan anticipar, formular un plan para remediar estas fallas y monitorear la ejecución de éste para asegurar el éxito [LAR1992]. Como posibles áreas de aplicación tenemos al control del tráfico aéreo, gestión de negocios, gestión de la batalla y control de la misión [DAV1998].

2.2.9 ARQUITECTURA DE UN SE

En el contexto de la IA, el término arquitectura se refiere a la ciencia de diseñar algo, la determinación de su estructura. La arquitectura de un SE depende de la forma de representación del conocimiento e implementación adoptadas.

Mientras que en la programación clásica se diferencia únicamente entre el propio programa y los datos, en el caso de los sistemas expertos se diferencian tres componentes principales [MOH2000] . Son los siguientes:

2.2.9.1 BASE DE CONOCIMIENTO (BC)

Feigenbaum en una conferencia sobre IA en 1977 decía: “La potencia de un SE deriva del conocimiento que él posee y no de los formalismos y de los esquemas específicos que él emplea”. La Base de Conocimiento contiene conocimiento bajo la forma de reglas de producción, marcos, redes semánticas o cualquier otra forma. La BC posee una sumatoria de hechos, de heurísticas y de creencias[RAB1995].

La BC es capaz de ciertos tipos de control sobre sí misma, pudiendo inclusive proporcionar algunas informaciones ausentes. Es especialmente esta última característica que la distingue de las tradicionales bases de datos. La BC es generalmente estática y cambia poco durante el funcionamiento del sistema. Esta es aplicable a todos los casos posibles que puedan surgir, y frecuentemente es obtenida en la *fase de adquisición del conocimiento*, en el desarrollo del SE.

Cuando el conocimiento de un SE es abstraído como un conjunto de reglas de producción la BC recibe el nombre de Memoria de Producción o Base de Reglas [GIA1998].

2.2.9.2 MEMORIA DE TRABAJO (MT)

Contiene conocimiento transitorio que cambia significativamente durante el uso del sistema. Se inicializa con los valores de los atributos específicos de interés para un caso particular que se está examinando, obtenido del usuario en cada interacción específica. En muchos sistemas, el conocimiento de un caso específico es codificado como un conjunto de hechos independientes y la MT es referida como una *Base de Hechos* [MOH2000].

2.2.9.3 MOTOR DE INFERENCIA O DE DEDUCCION (MI)

El motor de inferencia (o mecanismo de inferencia o máquina de inferencia) no es normalmente un único módulo de programa. El MI es el componente operacional que aplica la BC a la MT, cambiando la última. Esto genera resultados y explicaciones y es determinado generalmente por la elección de la herramienta de desarrollo de SE. Si se usa una Shell o herramienta de SE (tal como CLIPS) el usuario no tendrá que programar un nuevo motor de inferencia, pero debe aprender cómo usar una ya existente [MOH2000].

El MI es en general entendido como el interpretador de reglas y el programador de las reglas cuando el SE consiste de reglas de producción. El MI comprende tipos de solución ("forward", "backward",...) y la manipulación de las incertidumbres (lógica difusa, lógica bayesiana, lógica multivalores u otra). Su estructura está íntimamente ligada a la naturaleza del problema. Para la implementación del MI en algunos casos se pueden adoptar software disponible, en cuanto que en otros es necesario elaborarlos. Ambos casos tienen sus ventajas y desventajas. La adopción de un programa disponible comprende su aceptación como un todo, hasta en aspectos que acostumbran ser deficientes, como es el caso de la justificación de la respuesta. La opción por la elaboración de software implicará en

costos y tiempos adicionales, pudiendo sin embargo satisfacer más detalles y permitir una mejor adaptación al problema.

Se encuentra software en el mercado que permite al usuario disponer de los recursos fundamentales de inferencia para algunas formas de representación del conocimiento y además, que permiten hacer ciertas adaptaciones juzgadas convenientes [RAB1995].

Como con cualquier programa de computador, el SE debe comunicarse con los usuarios, requiriendo de una interfaz apropiada. Tres tipos de interfaz son necesarias (aunque estas pueden ser combinadas en algunas aplicaciones):

2.2.9.4 INTERFAZ DE USUARIO

El usuario es en general alguien que no participó de la elaboración del SE, siendo por tanto, natural que no conozca las estructuras que sostienen el sistema y que probablemente no esté familiarizado con las formas de representación del conocimiento adoptadas. Para que los usuarios potenciales puedan acceder con provecho y sin mayores dificultades al SE es preciso entonces dotarlo de recursos para consulta que son módulos explícitos o implícitos al sistema. Estos módulos establecen un lenguaje orientado al problema, pudiendo ser un subconjunto de la lengua nativa [RAB1995]. Mediante la IU se inicializa la MT enviando señales de control apropiadas al MI y recibiendo resultados de éste último [MOH2000].

2.2.9.5 INTERFAZ DE ADQUISICION DEL CONOCIMIENTO

Es un módulo que permite al usuario introducir conocimientos en el sistema, sin tener al ingeniero del conocimiento para que codifique éste en forma explícita. Aparece generalmente dotado de recursos para trabajar el conocimiento (editores, ordenadores, clasificadores, etc.) que ayudan en la difícil tarea de extraerlo y

aprovecharlo adecuadamente. En muchos sistemas es la única forma de aprendizaje.

La adquisición de conocimiento está logrando estatus propio en la IA, tendiendo a caracterizar áreas de investigación específicas en las universidades y en centros de investigación, generalmente ligadas a la ingeniería del conocimiento. Obtener conocimiento es sin duda, la parte más crítica de la construcción de un SE [RAB1995].

2.2.9.6 MECANISMO DE EXPLICACIÓN

Explica al usuario el razonamiento del sistema [GIA1998]. Permiten a los sistemas explicar o justificar sus conclusiones, y también posibilitan a los programadores verificar el funcionamiento de los propios sistemas. Los principales tipos de explicaciones son:

- **Por qué**, el cual es activado si el usuario cuestiona el por qué de la pregunta.
- **Como**, el cual se activa cuando el usuario desea saber cómo fueron obtenidas las conclusiones ofreciendo la secuencia de razonamiento llevada a cabo por el SBC.
- **Que sucede si**, el cual se activa si el usuario desea saber cuál sería el resultado alcanzado si una o más informaciones fueran modificadas.
- **Por qué no**, donde se analiza el proceso de inferencia y se descubre cual es la razón por la cual aquella conclusión no ha sido alcanzada.

Separando cada interfaz proporcionamos alguna protección contra la modificación de una BC por error o malicia.

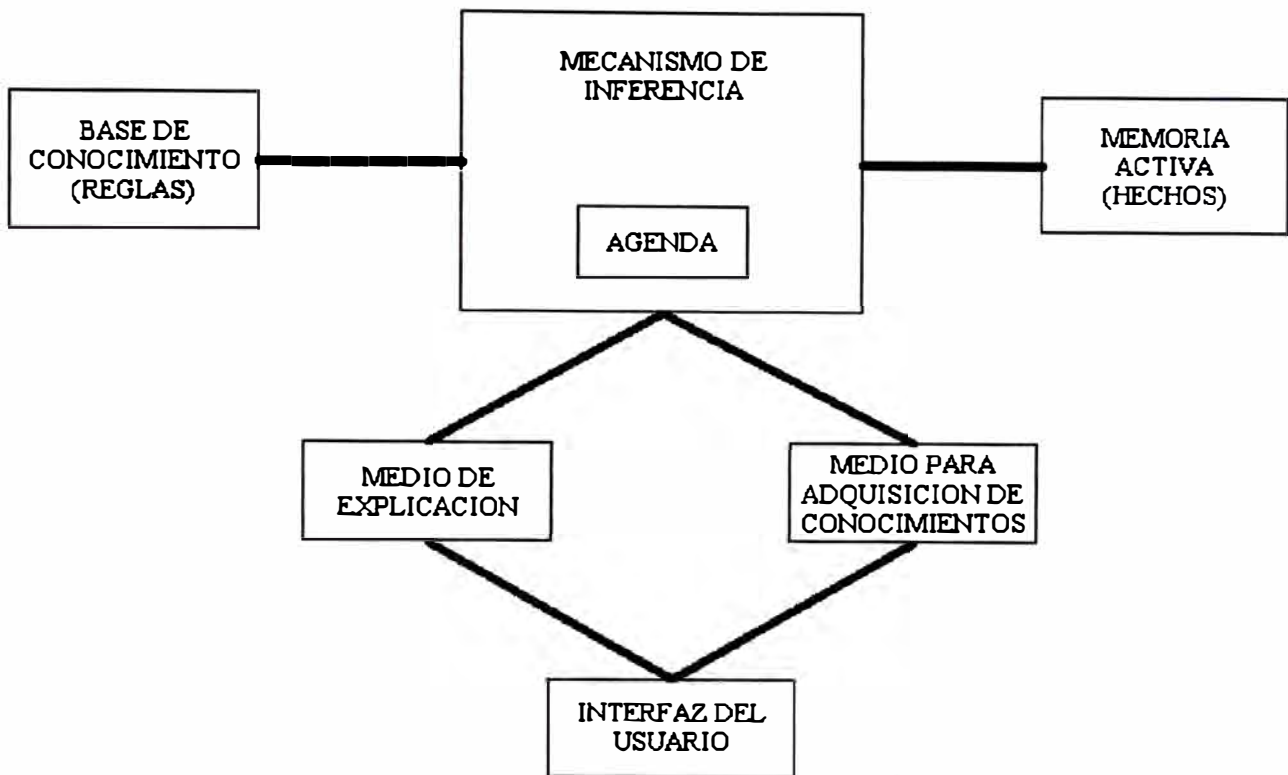


Figura 2.6 – Arquitectura de un SE basado en reglas

Fuente: [Giarratano y Riley, 1998]

2.2.10 EVENTOS IMPORTANTES EN LA HISTORIA DE SE

El desarrollo de la tecnología de SE tiene amplios antecedentes. La tabla presenta un breve resumen de los principales hitos en el desarrollo de los SE [GIA1998].

Año	Evento
1943	Reglas de producción de Post. Modelo neuronal de McCulloch y Pitts
1954	Algoritmo de Markov para controlar la ejecución de reglas
1956	Conferencia de Darmouth: Teórico Lógico; Búsqueda Heurística, se acuña el término "IA"
1957	Perceptrón inventado por Rosenblatt; Inicia el GPS (General Problem Solver, Solucionador General de Problemas) (Newell, Shaw y Simon).
1958	Lenguaje LISP de IA (McCarthy)
1962	Principios de Neurodinámica (Principles of Neurodynamics) en la percepción de Rosenblatt.
1965	Método de solución para la comprobación automática de teoremas (Robinson).

	Lógica difusa para el razonamiento acerca de objetos difusos (Zadeh). Inicia el trabajo en DENDRAL, el primer sistema experto (Feigenbaum, Buchanan)
1968	Redes semánticas, modelo de memoria asociativa (Quillian)
1969	MACSYMA, sistema experto en matemáticas (Martin y Moses)
1970	Inicia el trabajo en PROLOG (Colmerauer, Rousell)
1971	HEARSAY I, para reconocimiento del habla <i>Human Problem Solving</i> populariza las reglas (Newell y Simon)
1973	MYCIN, sistema experto para diagnóstico médico (Shortliffe), que conduce a GUIDON, tutoría inteligente (Clancey) TEIRESIAS, concepto de mecanismo de explicación (Davis), y EMYCIN, primer Shell (Van Melle, Shortliffe y Buchanan) HEARSAY II, modelo de pizarrón de cooperación múltiple entre especialistas.
1975	Marcos, representación del conocimiento (Minsky)
1976	AM (Artificial Mathematician, matemático artificial), descubrimiento creativo de conceptos matemáticos (Lenat) Teoría de la evidencia de Dempster – Shafer para el razonamiento bajo incertidumbre. Inicia el trabajo en el sistema experto PROSPECTOR, para la exploración mineral (Duda, Hart)
1977	OPS Shell (o armazón) de sistema experto (Forgy), usado en XCON / R1
1978	Inicia el trabajo en XCON / R1 (McDermott, DEC) para configurar los sistemas de computadora DEC. Meta-DENDRAL, metarreglas e inducción de reglas (Buchanan)
1979	Algoritmo Rete para correspondencia rápida de patrones (Forgy) Inicia la comercialización de la IA Se forma Inference Corp. (libera ART, herramienta del sistema experto en 1985)
1980	Se funda Symbolics LMI para fabricar máquinas LISP
1982	Sistemas expertos en matemáticas SMP; Hopfield Neural Net; Proyecto japonés de quinta generación para desarrollar computadoras inteligentes.
1983	KEE, herramienta de sistema experto (InteliCorp)
1985	CLIPS, herramienta de sistema experto (NASA)
1990	Sistemas neuronales artificiales
1993	Proyecto Lumiere (Microsoft)
1994	Sistemas Neuronales Artificiales Comerciales
1996	GUESS - Generically Used Expert Scheduling System
1998	Más de 12500 sistemas expertos (sólo en USA)
2000	Agentes inteligentes- Web, E-commerce, motores de búsqueda

Tabla 2.3: Eventos importantes en la historia de los SE

Fuente: [Giarratano y Riley, 1998]

Como podemos ver, la historia de los SE comienza en 1943 con las reglas de producción de Post. La tendencia inicial fue hacia sistemas inteligentes basados en métodos de razonamiento poderosos en vez de hacer uso de los conocimientos de

dominio. Ellos también fueron diseñados para cubrir cualquier dominio en lugar de una zona especializada [NOR1995].

El inconveniente de este enfoque fue que la máquina tenía que descubrir todo desde los primeros principios en cada nuevo dominio. En contraste, un experto humano se basaría en el conocimiento del dominio para un alto rendimiento.

2.2.11 ANTECEDENTES EN LA MEDICINA

Una discusión que se ha presentado desde sus inicios hasta el día de hoy es sobre el origen de la Inteligencia Artificial en medicina, unos dicen que se originó en las Ciencias de la Computación pero otros sostienen que es una nueva ciencia experimental. Sostienen que su origen está en las Ciencias de la Computación ya que los primeros laboratorios de investigación estaban en MIT, Rutgers, Stanford y la Universidad de Pittsburg; los métodos utilizados, la terminología y el ambiente en el que se desarrolló eran puramente informáticos, aunque siempre se contó con la presencia y participación de los médicos y miembros de esa rama de la ciencia.

Desde sus primeros días, la Inteligencia Artificial en la Medicina se ha basado en los problemas observados en la Biomedicina, pero sus verdaderos problemas eran a nivel de representación de la información, adquisición y manejo de la misma.

“Los investigadores de Inteligencia Artificial en Medicina están tratando de desarrollar herramientas para mejorar la efectividad en la toma de decisiones clínicas, no para reemplazar a los médicos u ofrecer consejos”.

2.2.11.1 PASADO

En la década del 50 se dio inicio al uso de las computadoras en la Bioingeniería. Las primeras aplicaciones que se desarrollaron eran cuestionarios automatizados para los pacientes.

En un artículo publicado por Ledley y Lusted en 1959 decían que el razonamiento médico contenía estrategias de inferencia entre las que se encontraba la lógica booleana, inferencias simbólicas y probabilidades bayesianas. Colocando como ejemplo particular el hecho de que el razonamiento para hacer un diagnóstico médico contenía las tres técnicas. La esperanza que surgió para muchos cuando se iniciaron estas investigaciones era que iban a obtener “Médicos perfectos en una caja” mas no iban a poder reemplazar al médico humano, pero lo que menos esperaban era que el camino a recorrer para lograrlo fuera tan largo y con obstáculos que ya se habían pasado o se estaban pasando en otras áreas de la Informática.

Una de esas primeras aplicaciones fue Eliza, desarrollada en la década del 60 en los laboratorios del MIT por el profesor Joseph Weizenbaum, Eliza simulaba el comportamiento de un médico psicoanalista, fue la primera aplicación psiquiátrica. Lo que hacía era repetirle al paciente sus respuestas en forma de pregunta cambiando ciertas palabras o frases. Funcionaba por medio de reconocimiento de patrones y reemplazo de palabras claves por frases diseñadas.

Otro sistema desarrollado en lo que se consideraría la primera generación de Sistemas de Inteligencia Artificial en la Medicina y que se considera como la mejor o al menos la pionera en su género es **Internist** que era una base de conocimiento y un programa de inferencia, cuyo objetivo era el diagnóstico en el área de medicina interna.

Fue desarrollado en la Universidad de Pittsburg como una colaboración entre el Dr. Jack Myers (médico) y el Profesor Harry Pople (Ciencias de la Computación); junto con ellos trabajo el Dr. Randy Miller en la década del 70 y luego se convirtió en el líder del proyecto en los 80's. Su método de razonamiento es de planteamiento de Hipótesis. Internist fue desarrollado en LISP, fue utilizado como plataforma para desarrollar métodos y extensas bases de conocimiento clínico. Internist dependía de una gran base de conocimiento que fue transferida a un sistema de información llamado **QMR** (Quick Medical Reference). [KUL2000]

La base de conocimiento de **Internist/QMR** asociaba enfermedades con síntomas usando dos números: una frecuencia de asociación y una fuerza de reaparición (evocación). Al comparar el desempeño de Internist/QMR con los de un médico y un experto o más bien comparar la cantidad de errores de diagnóstico de Internist con respecto a los otros, los resultados no estaban muy distantes; en ese estudio se observó que Internist cometió 29 errores de diagnóstico, el médico 28 y el experto 21. A pesar de ser un resultado muy positivo para los investigadores al sistema se le hicieron varias observaciones como el hecho de que no tenía un modelo probabilístico definido y no hacia diferenciaciones anatómicas ni temporales.

Uno de los pioneros en el desarrollo de la Inteligencia Artificial en la Medicina es **Edward Shortliffe** quien en 1983 dijo: *"la informática médica cubre más que aplicaciones de computadores para la medicina"*. Shortliffe era miembro de la Universidad de Stanford donde tenía un grupo de Investigación en el tema, grupo que desarrolló **Mycin**(1972-1980) y **Oncocin**. En el desarrollo de Mycin también participó Buchanan como uno de los investigadores principales junto con Shortliffe.

Este sistema introdujo nuevas características: utilización de conocimiento impreciso para razonar y posibilidad de explicar el proceso de razonamiento. Su función es la de aconsejar a los médicos en la investigación y determinación de diagnósticos en

el campo de las enfermedades infecciosas de la sangre. El sistema diagnosticaba la causa de la infección usando el conocimiento relativo a la infección de los microorganismos con historiales de pacientes, síntomas y los resultados de los test de laboratorio. MYCIN se basaba para realizar los razonamientos en factores de certeza y certidumbre

Uno de los aportes más importantes de MYCIN a la IA es que aparece claramente diferenciados motor de inferencia y la base de conocimientos. Al separar esas dos partes, se puede considerar el motor de inferencias aisladamente. Esto da como resultado un sistema vacío o Shell.

MYCIN supuso el punto de partida para una serie de sistemas expertos que fueron dando a su vez otros. Así surgió EMYCIN (MYCIN Esencial) con el que se construyó SACON, utilizado para estructuras de ingeniería, PUFF para estudiar la función pulmonar y GUIDON para elegir tratamientos terapéuticos.

El método de razonamiento de MYCIN es de encadenamiento hacia atrás, es decir, dirigido hacia un objetivo. MYCIN fue escrito en lenguaje LISP y sus reglas están formalmente representadas como expresión LISP.

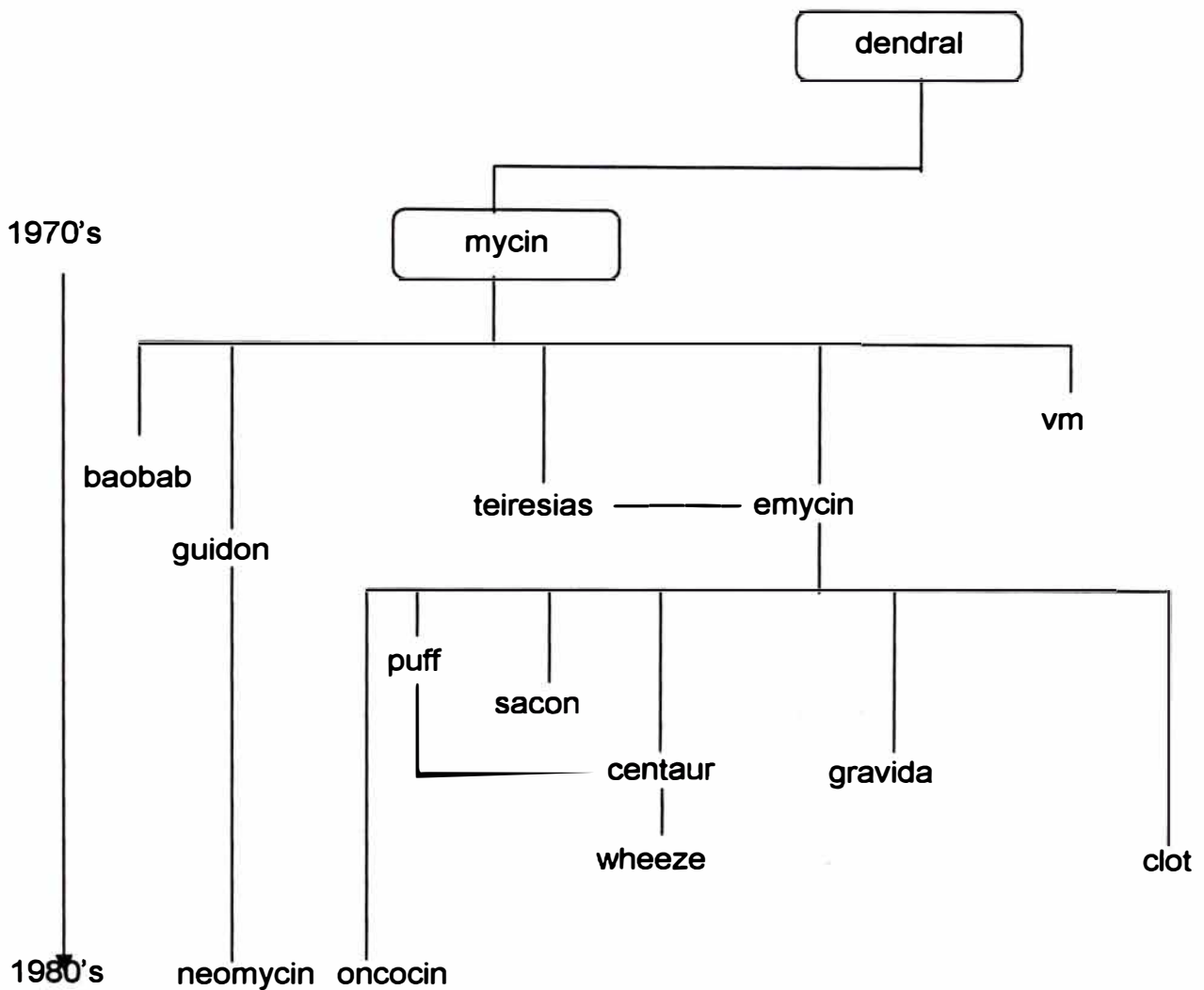


Figura 2.7: Primeros Sistemas Expertos en Medicina

Fuente: [WordPress, 2011]

Estos primeros sistemas fueron conocidos como **diagnosticadores**. Al evaluar estos sistemas su desempeño fue muy similar al de un médico humano. Aunque ahora el problema era uno que los Sistemas de Información tradicionales habían afrontado en sus primeros días: *“La apatía de los usuarios al sistema”*. Los usuarios finales de estos sistemas expertos por lo general eran enfermeras o médicos que obtenían la información de los pacientes; para esa época la información no estaba disponible en formato digital y era mucho lo que había que

pasar a formato digital y eran una serie de preguntas que resultaban engorrosas ya que su respuesta estaba en papel. Otro problema era la interfaz que era puramente texto.

Por estos problemas y otros que se observaron a través de estos años se volvieron importantes las formas de representación del conocimiento, la adquisición automática de la información, la formación de bases de datos (de hechos) y se vio la necesidad de una terminología estándar (en la cual ya se venía trabajando).

Muchos de estos sistemas por ser de los primeros no llegaron a ser usados mas allá de los laboratorios de desarrollo o como herramientas de estudio. Un ejemplo es **DXPlain** que fue desarrollado en el Hospital General de Massachussets alrededor de 1987 por el Dr. Barnett.

Usado para asistir en el proceso de diagnóstico, tomando información clínica como signos vitales, resultados de laboratorio y síntomas para generar un listado de diagnósticos ordenados por mayor similitud. Suministraba justificación para cada uno de los diagnósticos. Este sistema contenía una gran base de información médica de enfermedades y síntomas. En la actualidad es usada en varios hospitales y escuelas médicas con propósitos puramente educativos, es decir, como un texto guía electrónico.

A continuación se presentará una tabla con una lista de algunas aplicaciones que se han venido desarrollando en estos días, que tipo de sistemas son (de diagnóstico, monitoreo, etc.), una breve descripción, la fecha en la que fue creada o se hizo la última versión y el país donde fue creado.

Nombre del Sistema	Tipo	Descripción	Fecha y Lugar
Automedon	Administrador de ventilación	Sistema basado en conocimiento para la administración de ventilación mecánica en	2001 Francia

		la Unidad de Cuidados Intensivos.	
VIE-PNN	Nutrición Neonatal	Sistema experto para la nutrición de neonatos en cuidados intensivos.	1993 Austria
ATHENA	Administración de la hipertensión en cuidados primarios	Control de la hipertensión en cuidados primarios siguiendo lineamientos médicos y dando recomendaciones de cuidado y medicación.	2002 CA – EEUU
CEMS	Sistema de soporte de decisión de Salud Mental	Puede ser consultado sobre diagnóstico y tratamiento de pacientes y monitorea y da alertas sobre los métodos y los resultados	1993 EEUU
ERA	Sistema de soporte de decisiones y referencias de cáncer	Sistema de soporte de toma de decisiones interactivas en la identificación de pacientes con sospechas de cáncer.	2001 Reino Unido
LISA	Soporte a la toma de decisiones para niños con leucemia linfoblástica	De uso durante los periodos para la toma de decisiones sobre medicaciones, y almacenamiento y seguimiento del historial y estado de los pacientes.	2004 Reino Unido
PAIRS	Soporte para la toma de decisiones diagnósticas	Diseñado para ayudar a médicos en el diagnóstico de casos complicados.	2001 India
RetroGram	Soporte para la toma de decisiones para medicación	Genera regímenes de medicación usando historial médico e información genética de enfermos de VIH	1999 Reino Unido
TherapyEdge	Basado en Web para tratamiento de VIH y enfermedades crónicas	Sigue gráficamente y procesa automáticamente la información de pacientes (medicación, condición) con VIH y enfermedades crónicas.	2001 NC – EEUU
TxDENT	Experto diagnosticador de condiciones dentales	Para hacer seguimiento y recomendaciones a pacientes de tratamientos odontológicos.	1997 Canadá
Coulter® FACULTY™	Sistema Experto en Toxicología	Sistema basado en conocimiento cuyo fin es de consulta, asistir en el flujo de trabajo y servir como herramienta educacional en los laboratorios de hepatología.	1996 Reino Unido
HEPAXPERT I, II, III	Análisis e Interpretación de pruebas	La III es una versión basada en Web que analiza e interpreta pruebas para detectar hepatitis A, B, C y D	1991 Austria

Tabla 2.4: Sistemas de Inteligencia Artificial en Uso en las Clínicas.

Fuente: [Cywarlie, San Román y Uccello, 2004]

2.2.12 FUENTES DE CONOCIMIENTO

Existen dos tipos de fuentes de conocimiento:

Fuentes Primarias: acceso directo al conocimiento (experto humano, libros, textos, etc.).

Fuentes Secundarias: acceso indirecto a la información (referencias, publicaciones, etc.).

EL CONOCIMIENTO

Las tres primeras décadas de investigación en IA permitieron constatar que la inteligencia requiere conocimiento. El comportamiento inteligente de una persona o de un programa puede ser descrito por el conocimiento demostrado al ejecutarse la tarea que requiere inteligencia.

Se llama Conocimiento al entendimiento sobre cómo manipular los elementos que conforman el dominio para deducir conclusiones y soluciones a diferentes problemas. El conocimiento no es acumulación de información, sino capacidad de acción. En síntesis, el conocimiento es la unión de los hechos con las reglas y las heurísticas. Los hechos representan los elementos básicos y los objetos que describen y conforman el contexto, mientras que las reglas y las heurísticas determinan las acciones.

Existe conocimiento teórico y heurístico. El conocimiento teórico o factual lo forman las informaciones comprobadas y aceptadas por la comunidad científica. Está contenido en libros u otras publicaciones. El conocimiento heurístico lo forman reglas de "sentido común" obtenidas a través de la experiencia de pocos científicos considerados especialistas en alguna área de la Ciencia. Estas reglas resultan de la intuición de los especialistas que normalmente no tienen como comprobarlas científicamente. Este conocimiento es el menos riguroso, es más experimental.

2.2.13 INGENIERÍA DEL CONOCIMIENTO

La ingeniería del conocimiento se divide en la adquisición del conocimiento, representación, validación, inferencia, explicación y mantenimiento de las bases de conocimiento:

- **Adquisición del conocimiento** Se refiere a la obtención del conocimiento de sus diversas fuentes: libros, documentos, computadoras con principal énfasis, por su dificultad, a expertos humanos. El conocimiento se refiere a la solución de problemas en un dominio de aplicación e involucra los objetos del dominio, los procedimientos y la forma general de como el conocimiento es aplicado.

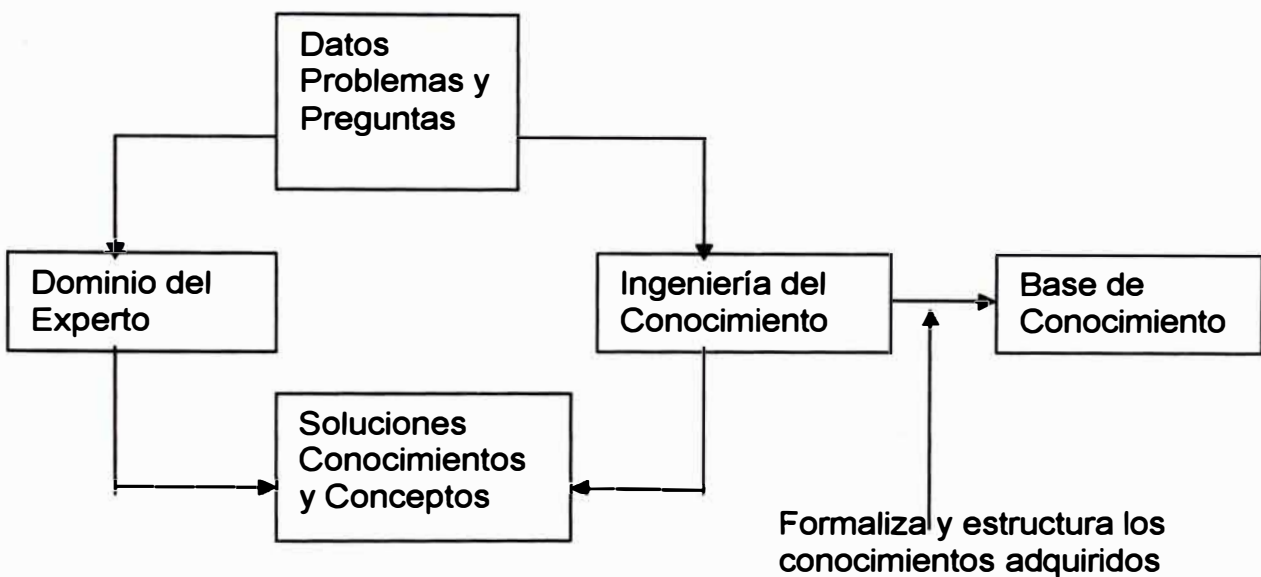


Figura 2.8 : Proceso de Adquisición del Conocimiento

Fuente: [Waterman, 1986]

En el contexto de SE, es el proceso de captar procedimientos, reglas, métodos es decir el razonamiento que sigue el experto para resolver el problema y luego transferirlo al sistema. En la figura queda clara la interacción existente entre el experto y el ingeniero de conocimiento en esta etapa.

- **Representación del conocimiento** Se refiere a la selección de una forma de representación que para un modelo del dominio y la codificación de la información adquirida en esa forma. Se refiere habitualmente al conocimiento declarativo. Tratar conocimiento con mecanismos artificiales, como las computadoras, no es fácil. Russel & Norvig aclaran que para manipular conocimiento es primordial que

se consiga formas de representarlo. La representación del conocimiento es una de las áreas más activas de la IA.

- **Validación del Conocimiento:** Verificación de la consistencia de la base de conocimiento.
- **Inferencia:** Definición de los procedimientos de manipulación y aplicación del conocimiento, con fines de implementación.
- **Explicación y Justificación:** Involucra la recuperación del razonamiento del sistema al alcanzar determinada conclusión y la definición de la forma de presentar esos caminos de inferencia para el usuario.

Etapas en la Adquisición del Conocimiento

La adquisición de conocimiento ocurre a lo largo de seis etapas:

- **Identificación y estudio del problema:** el experto, gerentes y usuarios del sistema definen cual es el problema que debe ser resuelto por el sistema. El problema es restringido para algo tratable computacionalmente y las entradas y salidas del sistema son definidas vagamente.
- **Entrevistas con el experto:** Se realizan entrevistas abiertas y después estructuradas con el experto para detallar cual conocimiento es aplicado en la solución del problema y de qué forma.
- **Análisis del conocimiento:** la información obtenida es analizada para evidenciar su consistencia, completitud y suficiencia para la solución del problema.
- **Propuesta de representación del conocimiento:** el conocimiento es estructurado siguiendo una de las formas de representación por computador.

- Implementación y prueba del sistema: el sistema es implementado, normalmente con la utilización de una herramienta de desarrollo de sistemas expertos y es probado en su funcionamiento.
- Validación junto al experto y usuarios: el sistema es validado por el experto, en cuanto al conocimiento que contiene, y por los usuarios en relación a la adecuación de la solución esperada y facilidad de uso.

2.3. REPRESENTACIÓN DEL CONOCIMIENTO

Para la Inteligencia Artificial, la *representación del conocimiento* es una combinación de estructuras de datos y procedimientos de interpretación que, si son utilizados correctamente por un programa, conducirán a un sistema a presentar una conducta inteligente. Así como una estructura de datos no es conocimiento, de la misma forma que una enciclopedia no lo es, si no tiene un lector dispuesto a comprenderla. El conocimiento puede ser considerado, por tanto, como la descripción de los objetos del mundo, sus relaciones con los otros objetos y la forma *como* esas informaciones pueden ser útiles para *resolver problemas*.

Los Psicólogos Cognitivistas han elaborado varias teorías para explicar cómo los humanos resuelven problemas. Estos trabajos describen como el conocimiento humano es usado comúnmente, cómo está mentalmente organizado y como es utilizado de manera eficiente para resolver problemas. Los investigadores de la IA

han usado estos trabajos para estudiar y desarrollar técnicas para representar mejor los diferentes tipos de conocimiento en el computador (Durkin,1994).

No existe una única teoría para explicar la organización del conocimiento humano o la mejor técnica para estructurar los datos en un computador convencional. Una de las mayores responsabilidades del Ingeniero del Conocimiento es escoger la mejor técnica de representación que se adapte a la aplicación.

2.3.1 CARACTERÍSTICAS DE LAS BUENAS REPRESENTACIONES DEL CONOCIMIENTO

A continuación se describen un conjunto de cualidades que caracterizan las representaciones del conocimiento:

- Las buenas representaciones explicitan las cosas importantes;
- Revelan restricciones naturales, facilitando algunas clases de cálculos;
- Son completas, pudiendo decir todo lo que deben ser capaces de decir;
- Son concisas, necesitando apenas de recursos mínimos y siendo al mismo tiempo aún eficientes cuando efectúan inferencias;
- Son transparentes, no usan criptografía, pudiéndose entender lo que quieren decir;
- Facilitan el cálculo en el sentido de que la información puede ser almacenada y recuperada rápidamente;
- Suprimen pormenores e informaciones raramente usadas, las cuales son abordadas y recuperadas apenas cuando son necesarias;

- Son computables por un procedimiento existente sin exigir el desarrollo de programación exclusiva;
- Permiten una adquisición fácil del conocimiento y son legibles por el experto, cuando fuera el caso;
- Permiten la aplicación de los mecanismos de inferencia necesarios;
- Presentan un acceso rápido y fácil al conocimiento;
- Permiten mantener todo el conocimiento coherente, utilizando, si es necesario, recursos para proteger el acceso.

2.3.2 CONCEPTOS DE ABSTRACCIÓN

Una *abstracción* es una descripción simplificada o una *especificación* de un sistema que enfatiza algunos detalles o propiedades en cuanto suprime aquellos juzgados irrelevantes para el objetivo en cuestión.

Una buena abstracción es aquella en la cual la información que es importante para el usuario es enfatizada y los detalles no importantes son suprimidos.

La tarea de abstracción se refiere a la representación de conocimiento, cuyo objetivo es el de modelar un “dominio del discurso” para almacenar y manipular conocimiento y realizar inferencias, bien como el modelo semántico de datos para la construcción de bases de datos. Podemos resumir las diferencias entre los dos objetivos como:

- Ambas disciplinas usan un proceso de abstracción para identificar propiedades comunes y aspectos importantes de los objetos del dominio en cuanto suprimen diferencias no significantes y detalles sin importancia.
- Ambas disciplinas suministran conceptos, restricciones, operaciones y lenguajes para definir datos y representar conocimiento.

- La representación de conocimiento posee un objetivo más amplio que el de modelos de datos semánticos. Diferentes formas de conocimiento como reglas, conocimiento incompleto y default, y conocimiento espacial y temporal, son representados en los esquemas de representación de conocimiento. Los modelos semánticos están siendo expandidos para incluir algunos de esos conceptos.
- Los esquemas de representación de conocimiento incluyen *mecanismos de razonamiento* que deducen hechos adicionales de aquellos almacenados en la base de datos, mientras que las bases de datos son capaces de responder sólo a preguntas directas sobre las informaciones contenidas en su base. La tecnología de BD está siendo expandida para incluir mecanismos de inferencia, como en las bases de datos deductivas.
- Los modelos de datos se concentran en la representación de los esquemas de bases de datos, o *metaconocimiento*, mientras que los modelos de conocimiento mezclan los esquemas con las propias instancias para obtener mayor flexibilidad en la representación. Eso hace que las implementaciones de las bases de conocimiento sean ineficientes cuando son comparadas con bases de datos, especialmente cuando necesitan ser almacenados grandes cantidades de hechos.

Los modelos describen el mundo a partir de cinco construcciones:

- Los objetos o entidades del dominio;
- Los atributos o propiedades de esos objetos;
- Las relaciones entre los objetos del dominio;
- Las operaciones sobre esos objetos que resultan en cambios en los objetos o en sus relaciones;
- Las restricciones que definen los conceptos y distinguen la realidad de este dominio de los demás.

La representación del dominio es realizada respetándose algunos axiomas de modelado:

- Todo lo que existe en el mundo real, incluyendo las entidades utilizadas para describir otras entidades son objetos del modelo. En ese caso, las diferencias entre los tipos del esquema de bases de datos (meta-información) y las instancias de la propia base no deben ser consideradas. Las diferencias entre tipos e instancias pueden ser representadas como objetos distintos, mas deberían ser tratados por el modelo del mismo modo.
- Se debe mantener una correspondencia uno a uno entre las entidades del mundo y los objetos del modelo, evitando que el concepto de un objeto sea distribuido entre diversas entidades.
- Cada objeto del mundo puede ser simple o compuesto (definido como una abstracción de otros objetos). Las abstracciones son por tanto expresadas como relaciones entre objetos con el objetivo de organizarlos de alguna forma.
- Las abstracciones pueden ser hechas sobre un conjunto de objetos simples para construir un objeto compuesto, siendo, en ese caso, una abstracción de un nivel. O pueden ser aplicadas recursivamente sobre objetos cada vez más complejos, formando relaciones n-nivel.

2.3.3 TÉCNICAS PARA LA REPRESENTACIÓN DEL CONOCIMIENTO

La selección de la técnica de representación del conocimiento está relacionada al grado de complejidad del conocimiento que debe ser representado, así como del tipo de aplicación donde será usado. Las técnicas de representación de

conocimiento más usadas son: lógica de predicados, reglas de producción, redes semánticas, tripletas objeto-atributo-valor, marcos y las basadas en probabilidad.

A continuación se realizará una descripción detallada de estas principales técnicas de representación de conocimiento.

2.3.3.1 REPRESENTACIÓN EN LÓGICA

2.3.3.1.1 LÓGICA

La lógica es el estudio de las reglas del razonamiento exacto. La aplicación de computadoras para realizar razonamientos ha tenido como resultado la programación lógica y el desarrollo de lenguajes basados en la lógica como PROLOG. La lógica tiene importancia primordial en los SE donde el mecanismo de inferencia razona a partir de los hechos para extraer conclusiones.

En el siglo IV a.C. *Aristóteles* (384 – 322 a.C.) desarrolló la lógica formal basada en silogismos como forma de representar el conocimiento, los mismos que se caracterizan por tener dos premisas y una conclusión que se infiere a partir de aquellas [GIA1998].

Ejemplo:

Premisa: Todos los hombres son mortales.

Premisa: Sócrates es un hombre.

Conclusión: Sócrates es mortal.

Su lógica permaneció inmutable hasta el siglo XIX con el desarrollo de las lógicas no clásicas. Fue *George Boole* (1815-1864) quien estructuró matemáticamente la lógica, publicando el libro ***Mathematical Analysis of Logic***, en la cual demuestra que puede ser manipulada algebraicamente, donde los resultados pueden ser

obtenidos a través de técnicas matemáticas. Este estudio dio soporte a la lógica simbólica usada en la computación.

2.3.3.1.2 LÓGICA DE PROPOSICIONES

Uno de los tipos más simples de la lógica formal es el silogismo. El término *formal* significa que la lógica se relaciona con la forma de las frases lógicas más que con su significado. Cualquier silogismo con la forma:

Premisa: Todos las X son Y.

Premisa: Z es una X.

Conclusión: Z es una Y.

Es válida sin importar qué se sustituye con X, Y y Z, sólo es importante la forma. Este concepto de separar la forma del significado convierte a la lógica en una herramienta poderosa. La lógica de proposiciones llamada también *cálculo de proposiciones* es una lógica simbólica para la manipulación de proposiciones.

A una afirmación en la que es posible determinar su valor de verdad se le llama **frase o proposición**. A las proposiciones en las que no se puede responder de manera absoluta se les llama **afirmaciones abiertas**. Una **afirmación compuesta** está formada por el uso de los conectores lógicos. Los conectores comunes de la lógica se muestran en la siguiente tabla:

Conector	Significado
\wedge	Y: conjunción
\vee	O: disyunción
\sim	NO: negación

\rightarrow	si..entonces: condicional
\leftrightarrow	si y sólo si: doble condicional

TABLA 2.5 – Conectores Lógicos

Fuente: [Giarratano y Riley, 1998]

Una **tautología** es una frase compuesta que siempre es verdadera. Una **contradicción** es una frase compuesta que es siempre falsa. Una frase **condicional** no es una tautología ni es una contradicción. Dos frases lógicamente equivalentes siempre tienen los mismos valores de verdad. Por ejemplo $p \equiv \sim \sim p$.

La tabla muestra los valores para los conectores binarios lógicos.

p	q	$p \wedge q$	$p \vee q$	$p \rightarrow q$	$p \leftrightarrow q$
V	V	V	V	V	V
V	F	F	V	F	F
F	V	F	V	V	F
F	F	F	F	V	V

Tabla 2.6: Tabla de verdad de los conectores lógicos binarios

Fuente: [Giarratano y Riley, 1998]

p	$\sim p$
V	F
F	V

Tabla 2.7: Tabla de verdad de la negación

Fuente: [Giarratano y Riley, 1998]

2.3.3.1.3 LÓGICA DE PREDICADOS DE PRIMER ORDEN

La lógica de predicados se desarrolló con el fin de analizar casos más generales. Su forma más simple es la lógica de predicados de primer orden, la base de lenguajes de programación lógicos como PROLOG. Los predicados son declaraciones respecto a los objetos o sobre las relaciones de los objetos entre sí.

La lógica de predicados se relaciona con la estructura interna de las afirmaciones, se relaciona con el uso de cuantificadores.

El cuantificador universal se representa por \forall y se lee “para cada uno” o “para todos”. Así por ejemplo, la afirmación “todos los triángulos son polígonos” se escribe como:

$$(\forall x)(x \text{ es un triángulo} \rightarrow x \text{ es un polígono})$$

Una forma más corta de escribir afirmaciones que incluyen predicados es utilizar las funciones de predicado para describir las propiedades del sujeto, así la afirmación lógica anterior también puede escribirse:

$$(\forall x)(\text{triángulo}(x) \rightarrow \text{polígono}(x))$$

El cuantificador universal también puede interpretarse como un conjunto de predicados acerca de casos. Por ejemplo, un perro llamado *Laika* es un caso particular de la clase perros y puede escribirse como:

$$\text{Perro}(\text{Laika})$$

Donde Perro es la función de predicado y Laika es un caso.

El cuantificador existencial describe una afirmación como verdadera por lo menos para un miembro del dominio. El cuantificador existencial se representa por \exists y se lee “Existe” [GIA1998].

Ejemplo:

$$(\exists x)(x \cdot x = 1)$$

$$(\exists x)(\text{elefante}(x) \wedge \text{nombre}(\text{Dumbo}))$$

2.3.3.1.4 EVALUACIÓN DEL USO DE LA LÓGICA COMO FORMA DE REPRESENTACIÓN DEL CONOCIMIENTO

Entre las ventajas tenemos:

- La expresión de algunos tipos de problemas en lógica parece ser la forma intuitiva de percibir el mundo, o sea es una representación natural y fácilmente comprendida.
- La lógica es precisa, se puede verificar el significado de una expresión utilizando procedimientos estandarizados.
- La lógica es flexible, se puede representar los hechos del mundo antes de definir con que forma de razonamiento ellos serán utilizados.
- La lógica es modular, las afirmaciones pueden ser construidas independientemente unas de otras sin afectar la consistencia de las deducciones.

Por otro lado, el hecho de que la forma de inferencia no esté asociada a la representación define una de las desventajas de la lógica, una vez que adiciona dificultades en la definición del razonamiento. Para problemas complejos, las formas de razonamiento tan genéricas como el *modus ponens* no son suficientes para la solución de problemas.

La segunda limitación del uso de la lógica para representar conocimiento se refiere a la granularidad de la representación. Los predicados representan muy poca información y la tarea de construir una base de conocimiento en dominios complejos puede ser inviable por la cantidad de informaciones a ser modeladas. En ese caso, las representaciones más estructuradas como redes semánticas o marcos son más adecuadas.

2.3.3.2 SISTEMAS DE PRODUCCIÓN

Fueron concebidos por el matemático *Emil Post* en 1943 como modelo "computacional" general de solución de problemas, cuando demostró que un procedimiento calculable puede ser modelado como un *sistema de producción*. En las décadas de los cincuenta y sesenta sirvieron de soporte para representar la forma humana de resolver problemas, especialmente en el juego de ajedrez y en la criptoaritmética. En la década del setenta sirvieron de soporte para el modelo mental.

El término "*Sistemas de Producción*" es usado actualmente para describir una familia de sistemas que tienen en común el hecho de estar constituidos por un conjunto de reglas, que reúnen condiciones y acciones. La condición (lado izquierdo, antecedente o premisa) está constituida por un patrón que determina la aplicabilidad de la regla, en cuanto la acción (lado derecho o consecuente) indica lo que será realizado cuando la regla fuera aplicada [RAB1995].

La sintaxis básica de una regla es:

SI <antecedente>
ENTONCES <consecuente>

Una regla puede tener múltiples antecedentes unidos por las palabras clave **Y** (conjunción), **O** (disyunción) o una combinación de ambos. Sin embargo, es un buen hábito evitar la mezcla de las conjunciones y disyunciones en la misma regla.

Las reglas pueden representar relaciones, recomendaciones, directrices, estrategias y heurísticas [NEG2005].

Relación

Si el "tanque de combustible" está vacío
ENTONCES el auto no arranca

Recomendación

SI la temporada es otoño
Y el cielo está nublado
Y el pronóstico es llovizna
ENTONCES el consejo es "llevar un paraguas"

Directiva

SI el auto no arranca
Y el "tanque de combustible" está vacío
ENTONCES la acción es "abastecer de combustible al auto"

Estrategia

Si el auto no arranca
ENTONCES la acción es "revisar el depósito de combustible"
Paso 1 es completo

SI Paso 1 es completo
Y el "tanque de combustible" está lleno
ENTONCES la acción es "revisar la batería"
Paso 2 es completo

Heurística

Si el flujo es líquido
Y el " pH del flujo " < 6
Y el "olor del flujo" es vinagre
ENTONCES el "material de flujo" es "ácido acético"

Un *sistema de producción* puede estar formado por una o más bases de reglas, separadas siguiendo las conveniencias de procesamiento. Complementa al *sistema*

de producción una *estrategia de control* que establece las prioridades en que las reglas serán aplicadas, así como *criterios de desempate* cuando hubiera más reglas candidatas a aplicación al mismo tiempo. Este último aspecto se denomina **resolución de conflictos** [RAB1995].

Dominios de aplicación de los Sistemas de Producción

Es cierto que los *sistemas de producción* tienen una amplia aplicación en Inteligencia Artificial, sin embargo, David y King (1977) proponen las siguientes:

1. Dominios en que el conocimiento es difuso, consistente de muchos hechos, como es el caso de la medicina, en oposición a los dominios donde ésta es una teoría concisa y unificada, por ejemplo en la Física.
2. Dominios en los cuales los procesos pueden ser representados como un conjunto de acciones independientes, como es el caso de un sistema médico de monitorización de pacientes, en oposición a dominios con subprocesos dependientes.
3. Dominios en que el conocimiento puede ser separado de su forma de uso, por ejemplo, en la taxonomía clasificatoria en botánica, en oposición a los casos en que la representación y control están íntimamente ligadas como ocurre en la receta médica.

Restricciones a la aplicación de sistemas de producción

Winston (1987) clasificó a los *sistemas de producción* como “eruditos idiotas” pues en su opinión, les faltan muchas características de los especialistas humanos. Sus principales deficiencias son [RAB1995]:

- No aprenden;
- No razonan en varios niveles;
- No examinan los problemas a partir de perspectivas diferentes;
- No saben cómo y cuándo violan sus propias reglas;

- No tienen acceso al razonamiento que está detrás de las reglas.

Frase	Significado
Base de conocimiento, memoria de producción	El conjunto de reglas en el programa
Regla	Tienen la forma si C entonces X, anotado posiblemente con un valor de prioridad
Elementos de condición	Elementos del lado izquierdo de una regla
Patrones	Elementos de condición simple a ser emparejados directamente con hechos; estos no contienen conectivos lógicos/aritméticos, ni predicados.
Acciones	Elementos del lado derecho de una regla
Hechos	Elementos de estado global, posiblemente de estructura de marcos
Plantilla de hechos	La estructura asociada con una clase de hechos
Memoria de trabajo	Colección de hechos presentes en un tiempo dado
Matching	Compara patrones de reglas con el estado global(elementos de la memoria de trabajo)
Agenda	Conjunto de instancias de regla que han sido emparejados con éxito y están listas para dispararse
Regla de activación	Ubica una instancia de una regla en la agenda
Regla de desactivación	Elimina una instancia de una regla de la agenda(debido a que otra regla fue activada, y su acción resulta en violar algún patrón para esta regla)
Motor de inferencia	Mecanismo de control que determina que regla se dispara.

Tabla 2.8: Terminología usada en sistemas expertos basados en reglas de producción.
Fuente: [Mohan, 2000]

2.3.3.2.1 RAZONAMIENTO CON REGLAS DE PRODUCCIÓN

- El mecanismo de inferencia busca validar las premisas de las reglas, para concluir la parte **ENTONCES** de la regla;
- Los resultados intermedios (premisas validadas y conclusiones deducidas) son almacenados en la memoria de trabajo;
- Cuando en el proceso de validación de una regla, hay tripletas que satisfacen apenas parte de las condiciones, es preciso volver atrás y buscar nuevas instanciaciones, en un proceso llamado *backtracking*;
- Si puede ser disparada o probada más de una regla en algún momento, debe ser previsto algún mecanismo de control para seleccionar la regla más promisoria, estableciendo las prioridades en que las reglas serán aplicadas, así como criterios de desempate. Este aspecto se denomina *resolución de conflictos*.
- Entre las estrategias para la resolución de conflictos tenemos: priorización de reglas; reglas ordenadas por uso más reciente (reglas disparadas apenas una vez para los mismos datos/tripletas); reglas que usan datos/tripletas más recientes en la memoria de trabajo.

2.3.3.2.2 MÉTODOS DE INFERENCIA

El proceso de inferir nuevos hechos a partir de informaciones ya disponibles es conocido por *inferencia* o *encadenamiento*. El término *inferencia* se usa generalmente para sistema expertos, para el pensamiento humano se utiliza *razonamiento*. Las reglas de producción permiten que la inferencia sea desarrollada de dos formas:

- **encadenamiento hacia adelante**, o dirigido por datos, o progresivo, donde el sistema dispara todas las reglas aplicables a partir del conjunto inicial de hechos conocidos;
- **encadenamiento hacia atrás**, o dirigido por hipótesis, o regresivo, donde el objetivo a ser alcanzado es conocido y el sistema intenta disparar solamente las reglas que pueden alcanzar el objetivo.

El motor de inferencia compara cada regla almacenada en la base de conocimientos con los hechos contenidos en la base de datos. Cuando el SI (condición) de la regla coincide con un hecho, la regla se dispara y la parte ENTONCES (acción) se ejecuta. La regla disparada puede cambiar el conjunto de hechos mediante la adición de un hecho nuevo. Las letras en la base de datos y la base de conocimientos se utilizan para representar situaciones o conceptos. El emparejamiento de la parte SI de la regla con los hechos produce *cadena de inferencia*. La cadena de inferencia indica cómo un sistema experto aplica las reglas para llegar a una conclusión.

2.3.3.2.2.1 ENCADENAMIENTO HACIA ADELANTE

El *encadenamiento hacia adelante* es el razonamiento basado en datos. El razonamiento parte de los hechos informados por el usuario y procede hacia las conclusiones. Cada vez se ejecuta sólo la regla de nivel superior. Cuando se dispara, la regla añade un nuevo hecho a la base de datos. Cualquier regla puede ser ejecutada solo una vez. El ciclo de coincidencia - disparo se detiene cuando no hay más reglas que puedan ser disparadas [NEG2005].

Para ilustrar el encadenamiento hacia adelante, veamos un ejemplo sencillo. Supongamos que la base de datos incluye, inicialmente, los hechos A, B, C, D y E, y la base de conocimientos contiene las siguientes reglas:

- Regla 1: $Y \& D \rightarrow Z$
- Regla 2: $X \& B \& E \rightarrow Y$
- Regla 3: $A \rightarrow X$
- Regla 4: $C \rightarrow L$
- Regla 5: $L \& M \rightarrow N$

La Figura 2.8 muestra cómo funciona el encadenamiento hacia adelante para este conjunto simple de reglas.

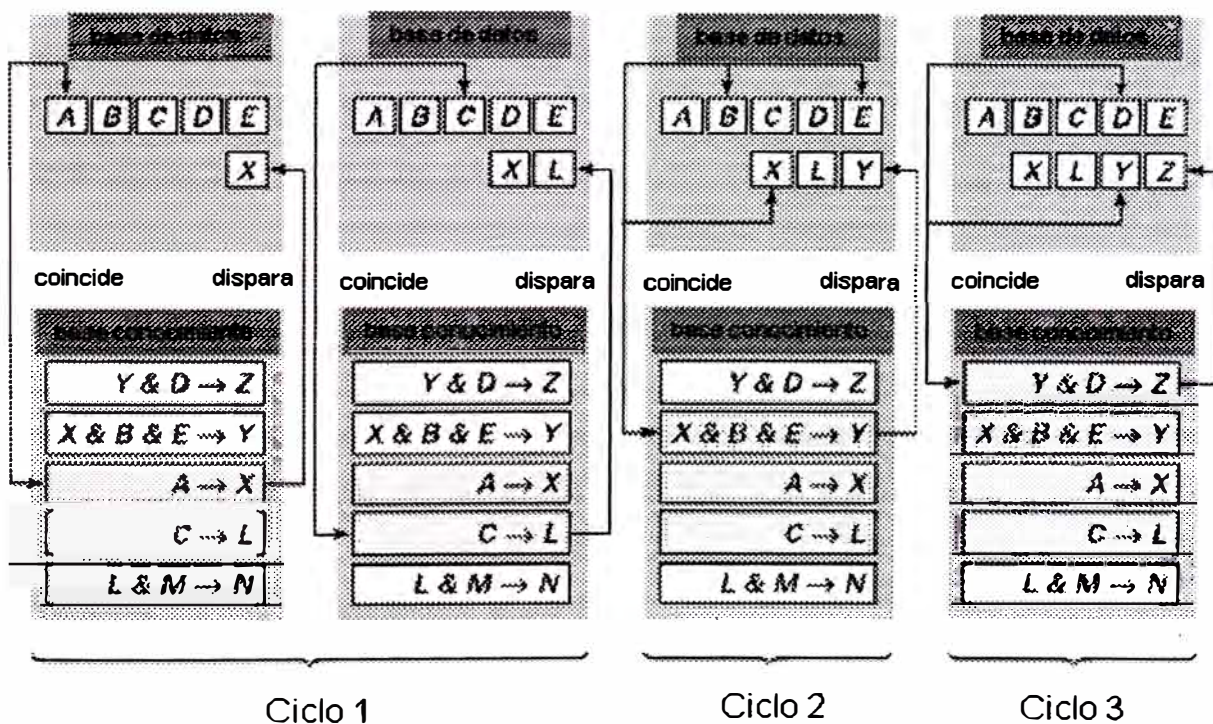


Figura 2.9: Encadenamiento hacia adelante.

Fuente: [Negnevitsky, 2005]

En el primer ciclo, sólo dos reglas, la Regla 3: $A \rightarrow X$ y la Regla 4: $C \rightarrow L$, coinciden con los hechos en la base de datos.

La Regla 3: $A \rightarrow X$ es disparada primero en el nivel superior. La parte SI de esta regla coincide con un hecho de la base de datos, su parte ENTONCES se ejecuta y el hecho nuevo X se agrega a la base de datos. A continuación, la Regla 4: $C \rightarrow L$ se dispara y el hecho L también se agrega a la base de datos.

En el segundo ciclo, la Regla 2: $X \& B \& E \rightarrow Y$ se dispara porque los hechos B , E y X ya están en la base de datos, y como consecuencia el hecho Y se infiere y se coloca en la base de datos. Esto a su vez hace que la Regla 1: $Y \& D \rightarrow Z$ se ejecute, agregando el hecho Z a la base de datos (ciclo 3).

Ahora el ciclo de coincidencia disparo se detiene porque la parte SI de la Regla 5: $L \& M \rightarrow N$ no corresponde con todos los hechos en la base de datos y por lo tanto la Regla 5 no puede ser disparada.

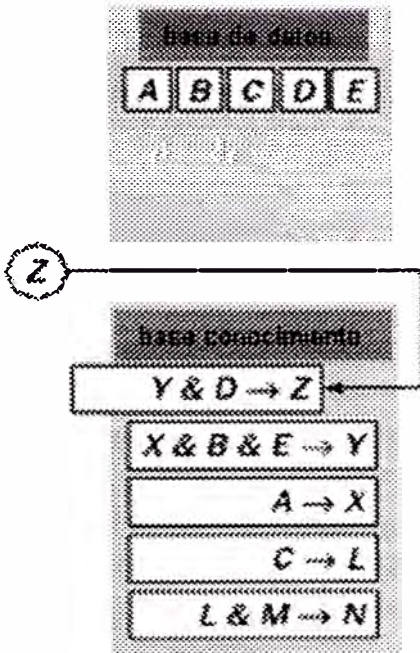
El *encadenamiento hacia adelante* es una técnica para recopilar información y luego deducir de ella lo que se pueda deducir. Sin embargo, en el *encadenamiento hacia adelante*, se pueden ejecutar muchas reglas que no tienen nada que ver con la meta establecida. En nuestro ejemplo, el objetivo fue determinar el hecho Z . Hemos tenido sólo cinco reglas en la base de conocimientos y cuatro de ellas fueron disparadas. Sin embargo, la Regla 4: $C \rightarrow L$, que no está relacionada al hecho Z , también fue disparada, entre otras. Un verdadero sistema experto basado en reglas puede tener cientos de reglas, muchas de las cuales podrían ser disparadas para derivar nuevos hechos que son válidos, pero desafortunadamente no guardan relación con la meta. Por lo tanto, si nuestro objetivo es inferir un solo hecho concreto, la técnica de inferencia de *encadenamiento hacia adelante* no

sería eficiente. En tal situación, el encadenamiento hacia atrás es el más apropiado.

2.3.3.2.2 ENCADENAMIENTO HACIA ATRÁS

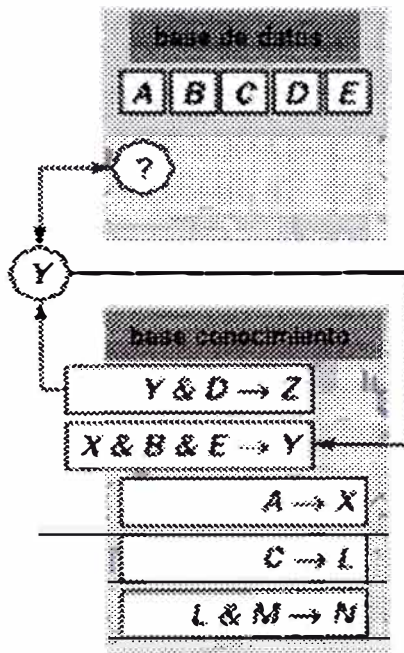
El *encadenamiento hacia atrás* es el razonamiento orientado a objetivos. En el *encadenamiento hacia atrás*, un sistema experto tiene el objetivo (una solución hipotética) y el motor de inferencia trata de encontrar la evidencia para probarlo. En primer lugar, se busca en la base de conocimiento para encontrar las reglas que podrían tener la solución deseada. Tales reglas deben tener el objetivo en sus partes ENTONCES (acción). Si tal regla se encuentra y su parte SI (condición) coincide con los datos en la base de datos, entonces la regla se activa y el objetivo queda demostrado. Sin embargo, esto sucede raramente. Así, el motor de inferencia deja de lado la regla con la que está trabajando (la regla se dice que se **apila**) y establece una nueva meta, un objetivo secundario, probar la parte SI de esta regla. A continuación, se busca de nuevo en la base de conocimientos las reglas que pueden probar el objetivo secundario. El motor de inferencia repite el proceso de apilar las reglas hasta que ninguna regla se encuentre en la base de conocimientos para probar el actual sub-objetivo [NEG2005].

Paso 1



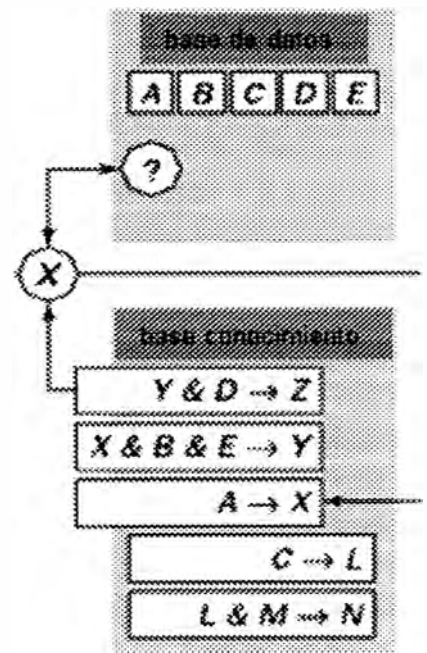
Objetivo: Z

Paso 2



Sub-objetivo: Y

Paso 3



Sub-objetivo: X

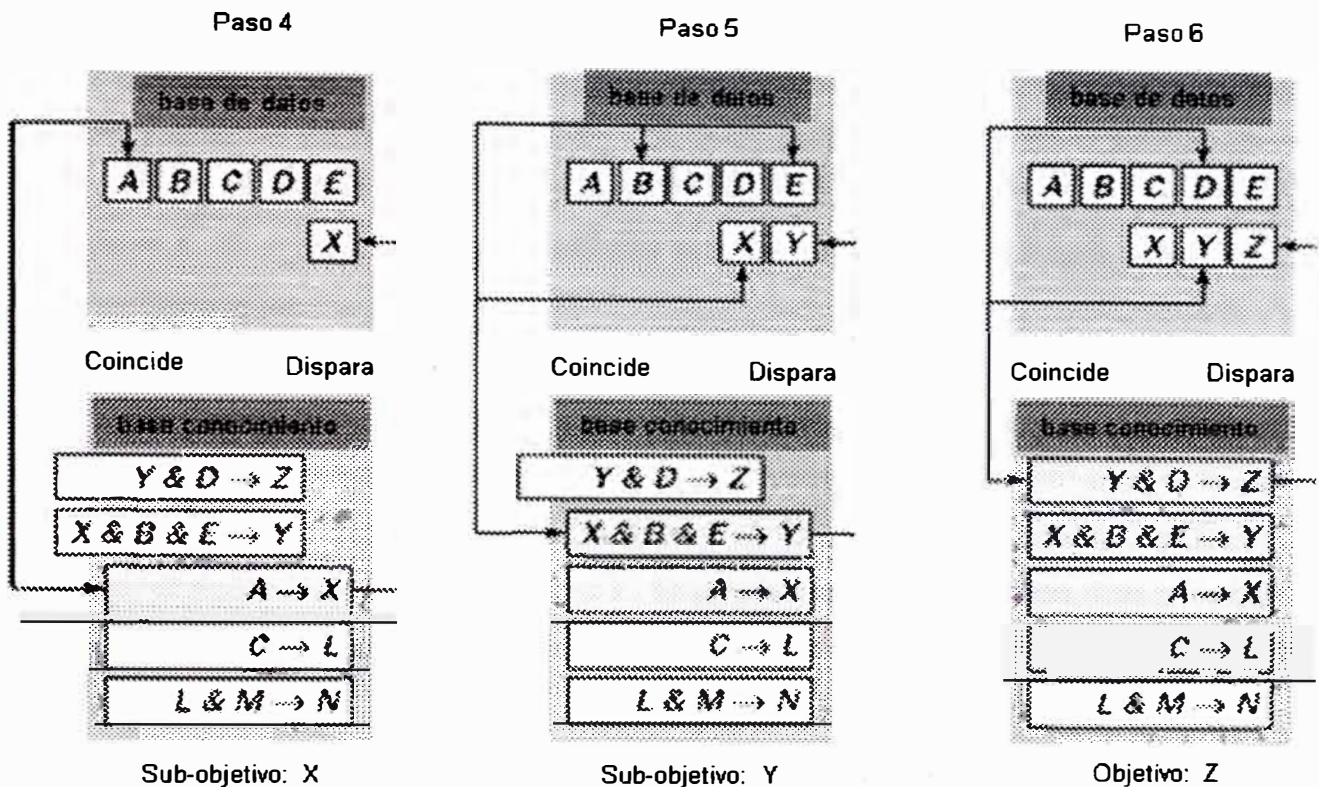


Figura 2.10. Encadenamiento hacia atrás.

Fuente: [Negnevitsky, 2005]

La Figura 2.9 muestra cómo trabaja el encadenamiento hacia atrás, usando las reglas para el ejemplo de encadenamiento hacia adelante. En el Paso 1, el motor de inferencia trata de inferir el hecho Z. Busca en la base de conocimientos para encontrar la regla que tiene como objetivo, en nuestro caso el hecho Z en su parte ENTONCES. El motor de inferencia lo halla y coloca en la pila la Regla 1: $Y \& D \rightarrow Z$. La parte SI de la Regla 1, incluye los hechos Y y D, y por lo tanto estos hechos deben ser establecidos.

En el Paso 2, el motor de inferencia establece como sub-meta, el hecho Y, y trata de determinarlo. En primer lugar, verifica la base de datos, pero el hecho Y no está. A continuación, se busca de nuevo la base de conocimientos la regla con el hecho Y en su parte ENTONCES.

El motor de inferencia localiza y apila la Regla 2: $X \& B \& E \rightarrow Y$. La parte Si de la regla 2 consiste de los hechos X, B y E, y estos hechos también tienen que ser establecidos. En el Paso 3, el motor de inferencia establece un nuevo sub-objetivo, el hecho X. Comprueba en la base de datos el hecho X, y como falla, busca la regla que infiera X. El motor de inferencia halla y coloca en la pila la Regla 3: $A \rightarrow X$. Ahora se debe determinar el hecho A.

En el Paso 4, el motor de inferencia encuentra el hecho A en la base de datos, la Regla 3: $A \rightarrow X$ se activa y se infiere el nuevo hecho X. En el Paso 5, el motor de inferencia retorna al sub-objetivo, el hecho Y, y una vez más intenta ejecutar la Regla 2: $X \& B \& E \rightarrow Y$. Los hechos X, B y E están en la base de datos y por lo tanto la Regla 2 se dispara y un hecho nuevo, el hecho Y, se añade a la base de datos.

En el paso 6, el sistema vuelve a la Regla 1: $Y \& D \rightarrow Z$ tratando de determinar la meta original, el hecho Z. La parte Si de la Regla 1, coincide con todos los hechos en la base de datos así que se ejecuta la Regla 1, y por lo tanto la meta original se establece finalmente.

Si comparamos la Figura 2.8 con la Figura 2.9 podemos ver que cuando fue utilizado el encadenamiento hacia adelante, se activaron cuatro reglas, pero sólo tres reglas cuando se aplicó el encadenamiento hacia atrás. Este ejemplo simple muestra que la técnica de inferencia de encadenamiento hacia atrás es más eficaz cuando tenemos que inferir un hecho particular, en nuestro caso el hecho Z.

La mayor parte de los sistemas expertos basados en reglas ya implementados utilizan la estructura de control por encadenamiento hacia atrás, que es la más adecuada para clasificación. Ese mecanismo provoca una reducción en el espacio de búsqueda del sistema, optimizando las consultas y alcanzando resultados en problemas que no serían resolubles a través de razonamiento hacia adelante.

También es posible combinar ambos métodos, razonando por frente a través de las reglas de un conjunto inicial de evidencias observadas para generar las hipótesis de tentativa. Se trabaja entonces para atrás, a través de otras reglas, para obtener evidencias que comprueben las hipótesis levantadas.

Encadenamiento hacia Adelante	Encadenamiento hacia Atrás
Utilizado en problemas de planeamiento, monitoreo y control	Utilizado en problemas de diagnóstico
Dirigido a datos (datos hipótesis)	Dirigido a metas (hipótesis datos)
Presente al futuro	Presente al pasado
Del antecedente de una regla al consecuente	Del consecuente de una regla al antecedente
Trabaja hacia delante para encontrar soluciones que derivan de los hechos	Trabaja hacia atrás para encontrar los hechos que soporten las hipótesis levantadas
Los antecedentes de las reglas determinan la búsqueda	Los consecuentes de las reglas determinan la búsqueda
Algoritmo del tipo búsqueda a lo ancho	Algoritmo del tipo búsqueda en profundidad

Tabla 2.9: Comparación entre Encadenamiento Progresivo y Regresivo

Fuente: [Giarratano y Riley, 1998]

2.3.3.2.3 EVALUACIÓN DEL USO DE REGLAS DE PRODUCCIÓN COMO FORMA DE REPRESENTACIÓN DE CONOCIMIENTO

Ventajas de los Sistemas de Producción

Los *sistemas de producción* son la forma de representación del conocimiento más usada en IA. La causa de esto reside en el hecho de que es natural al humano usar el par "condición – acción" para razonar y decidir. Entre las ventajas que tienen los sistemas de producción destacan [RAB1995]:

a) Modularidad

Las reglas de los *sistemas de producción* pueden ser consideradas para efectos de manipulación, como piezas independientes. Como los programas de IA casi siempre están incompletos pueden ser incorporadas nuevas reglas al conjunto ya existente sin mayores preocupaciones. Esta es una característica extremadamente importante, pues el volumen del conocimiento en general es grande y la verificación de su consistencia, difícil. Para sistemas muy grandes la modularidad es un elemento que aumenta indirectamente la complejidad.

b) Naturalidad

Las reglas de producción se consideran una forma natural de pensar la solución de problemas, por eso, lo que debe hacerse en determinada circunstancia, es directamente traducido en reglas de producción.

c) Uniformidad

En un *sistema de producción* se puede notar que todas las reglas están escritas siguiendo el mismo patrón. Esta forma rígida de representación permite que las personas no familiarizadas con el sistema puedan también analizar su conocimiento.

Desventajas de los Sistemas de Producción

a) Opacidad

Es resultante de la modularidad y de la uniformidad, lo que hace que sea difícil verificar la completitud de estos sistemas, así como verificar los posibles flujos de procesamiento. Podríamos conseguir aliviar este problema intentando separar las reglas en conjuntos, para hacerlas más entendibles.

b) Ineficiencia

Resulta, particularmente, del número de reglas a combinar y también del esfuerzo de "matching" necesario al soporte de ejecución de las reglas. Entenderemos por

“Matching” la verificación de las reglas que se aplican al estado del problema, así como la verificación de cuales reglas anteceden o suceden otra regla. Una forma de disminuir la ineficiencia es proponer una ordenación apropiada de las reglas y formas de seleccionarlas [RAB1995].

2.3.3.3 REDES SEMÁNTICAS

Una **red semántica** o **red** se utiliza para representar información relativa a las proposiciones; por eso se le llama también **red de proposiciones**. En términos matemáticos, una *red* es una gráfica rotulada y con dirección [GIA1998]. Las *redes semánticas* originalmente se usaron para dar soporte al lenguaje natural. El inicio de su uso se remonta a 1968, cuando Ross Quillian las usó para representar la memoria y la comprensión del lenguaje del ser humano [RAB1995]. Una *red semántica* está conformada por los *nodos* y los *arcos* que lo conectan. A los nodos suelen denominárseles como objetos y a los arcos como vínculos o bordes.

Los nodos se utilizan para representar objetos físicos, situaciones o conceptos. Los vínculos se utilizan para expresar relaciones entre estos elementos [GIA1998].

La semántica más común es la *semántica descriptiva*, que intenta mostrar a través del lenguaje natural en cuanto la descripción se aproxima de la realidad representada. Tenemos también la *semántica procedural*, que acude a programas para expresar su significado, y la *semántica de equivalencia*, que intenta de alguna forma establecer equivalencias entre la representación efectuada y una u otra forma de representar conocimiento. La *semántica descriptiva* es más atractiva que las otras, porque sus recursos de representación son más naturales. La *semántica procedural* exige, por ejemplo, el entendimiento de un lenguaje de programación [RAB1995].

Podemos elaborar un ejemplo de red semántica a partir del siguiente texto: “El computador es de María. Ella lo usa para escribir un libro importante. El libro es parte de los trabajos de investigación que María realiza en la Universidad”.

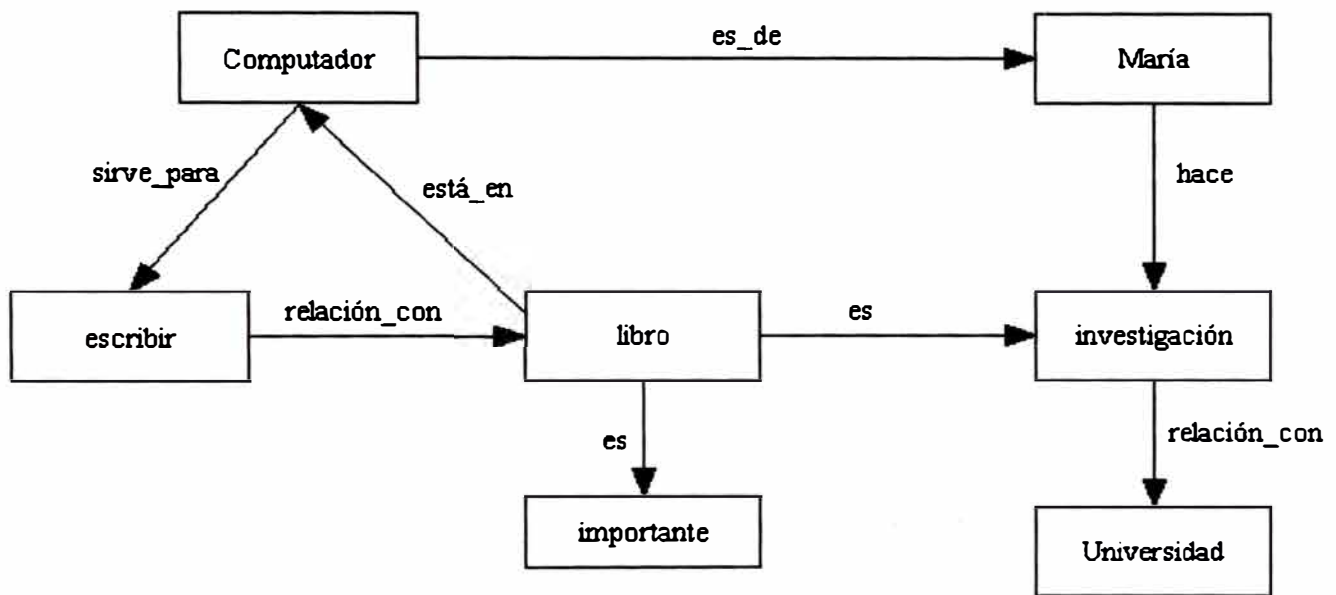


Figura 2.11: Red Semántica construida a partir de un texto.

Fuente: modificado de [Rabuske, 1995. p. 60]

Tipos de Vínculos en Redes Semánticas

Para facilitar la comprensión de la red es habitual usar tipos estándar de vínculos. Los dos tipos de vínculo más comunes son: “es_parte_de” y “es_un” o “es_una”. Los vínculos se pueden dividir en cuatro tipos básicos, a saber: propiedad, subparte, subclase y relación.

El vínculo de tipo “*propiedad*” relaciona un nodo a otro, dando a entender que se trata de una propiedad. En la figura, la relación entre “libro” e “importante” pertenece a este tipo. El vínculo del tipo “*subparte*” en general expresado por “es_parte_de” indica que un nodo es subparte o componente de otro. Por ejemplo

la hoja se puede entender como parte de un libro. El vínculo del tipo “*subclase*” expresa esta característica entre los nodos involucrados. Las tesis pueden ser consideradas “subclases” en relación al nodo “investigación”. El vínculo del tipo “*relación*” informa que los nodos implicados se deben entender relacionados de algún modo. En la figura son nodos de relación “libro” y “escribir”; “universidad” e “investigación” [RAB1995].

A la duplicación de las características de un nodo por parte de un descendiente se le llama *herencia*. Una importante característica de las redes semánticas es su propiedad de jerarquía. Los nodos de niveles jerárquicos más bajos denotan individuos o instancias, y son conectados por arcos *es_un*, mientras que los nodos de niveles jerárquicos más altos representan clases o categorías de individuos.

2.3.3.3.1 EVALUACIÓN DEL USO DE REDES SEMÁNTICAS COMO FORMA DE REPRESENTACIÓN DEL CONOCIMIENTO

Presentan las siguientes ventajas:

- Permiten representar asociaciones similares al pensamiento humano.
- Hacen factible la descripción de objetos del dominio y sus relaciones.
- Permiten la definición de nuevos subtipos de objetos que heredan todas las propiedades de la clase original.

Si bien es cierto que las *redes semánticas* pueden ser muy útiles y una manera elegante de representar el conocimiento, ellas poseen las siguientes limitaciones:

- Falta de relaciones estandarizadas, dificultando la comprensión del objetivo de la red, así como su consistencia, pudiendo generar ambigüedad.
- La simplificación del modelo puede generar inferencias incorrectas.

2.3.3.4 TRIPLETA OBJETO – ATRIBUTO - VALOR

Los elementos objeto, atributo y valor ocurren tan frecuentemente que es posible construir una red semántica simplificada utilizando únicamente estos elementos.

Así por ejemplo en la proposición “el color de la máquina es verde” la podemos representar en una estructura OAV como sigue:

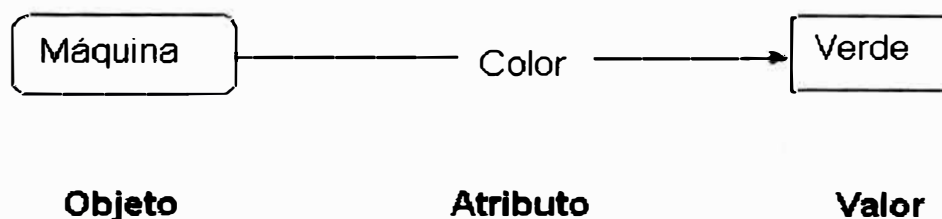


Figura 2.12: Tripleta OAV

Fuente: [Elaboración Propia, 2011]

Una **tripleta objeto- atributo – valor** (OAV) puede utilizarse para caracterizar todo el conocimiento de una red semántica y se usó en el SE MYCIN para diagnosticar enfermedades infecciosas. La representación del trío OAV es conveniente para ordenar el conocimiento en forma de tabla y luego traducirla al código de la computadora mediante inducción de reglas. Los tríos OAV son especialmente útiles para representar los hechos y sus patrones para llevarlos al antecedente de una regla. La red semántica para este tipo de sistema consta de nodos para objetos,

atributos y valores conectados por vínculos TIENE_UN y ES_UN. A continuación se presenta un ejemplo [GIA1998].

Objeto	Atributo	Valor
Manzana	Color	Roja
Manzana	Tipo	Delicia
Manzana	cantidad	100
Uvas	Color	Verde
Uvas	Tipo	Italia
Uvas	cantidad	500

Tabla 2.10: Tabla OAV

Fuente: Modificado de [Giarratano y Riley, 1998. p. 67]

2.3.3.5 MARCOS

La representación de objetos o eventos típicos ha sido la preocupación de una forma de representación del conocimiento denominada marcos (frames) propuesta por *Marvin Minsky* en 1975. El marco describe típicamente clases de objetos. Un marco representa objetos o conceptos a través del conjunto de sus atributos y operadores de manipulación [RAB1995].

Un marco tiene como característica principal que representa conocimiento relacionado con un tema concreto y cuenta con mucho conocimiento predeterminado. Las redes semánticas son básicamente representaciones bidimensionales del conocimiento; los marcos agregan una tercera dimensión para permitir que los nodos tengan estructuras que pueden ser valores simples u otros marcos [GIA1998].

El marco está constituido por un grupo de atributos llamados **ranuras** (slots o campos) y un conjunto de **métodos** (fillers) para su utilización. Las *ranuras* hacen referencia a los valores de los datos, son huecos para rellenar con valores y los

rellenos son procedimientos (demonios) que modifican los valores de esos datos [PAJ2006].

Ejemplo

Ranuras	Dato
Fabricante	General Motors
Modelo	Chevrolet Caprice
Año	1979
Transmisión	Automática
Motor	A gasolina
Ruedas	4
color	Azul

Tabla 2.11: Marco que describe a un vehículo
Fuente: [Giarratano y Riley, 1998. p 75]

Desde el punto de vista del OAV, el automóvil es un objeto, el nombre de la ranura es el atributo y el dato es el valor [GIA1998].

Los marcos suelen estar organizados jerárquicamente, en la cual los nodos superiores representan conceptos generales y los nodos inferiores representan instancias más específicas de los conceptos / objetos. Se acostumbra construir sistemas de marcos conectados entre sí. Esto permite establecer un mecanismo de herencia para activar y actualizar los datos de los campos, que es lo que constituye su sistema de inferencia.

Los marcos pueden ser:

1. **Clases**, que representan conceptos o entidades generales.
2. **Instancias**, concreciones o ejemplos particulares de las clases.

Si una secuencia de acciones se aplica a un marco, algunos de sus atributos cambiarán [PAJ2006].

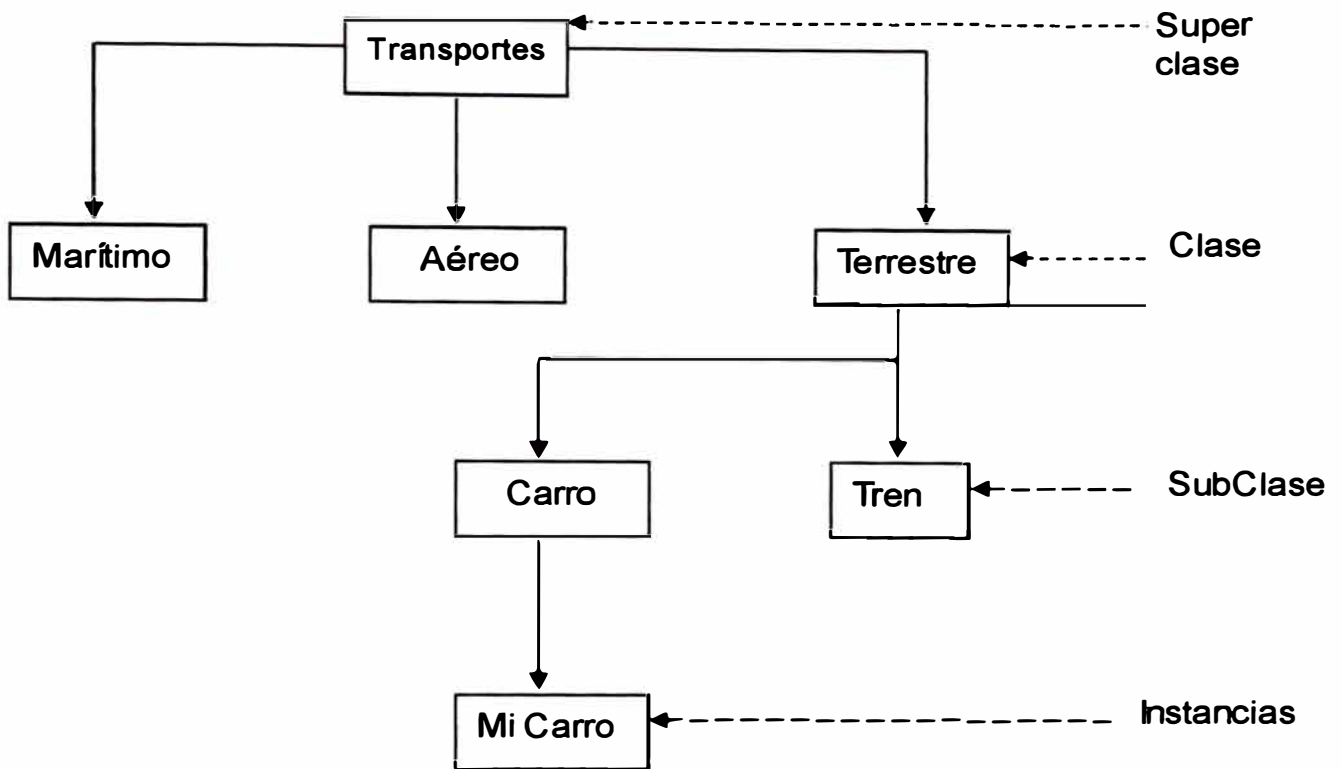


Figura 2.13: Organización Jerárquica del Conocimiento usando Marcos.

Fuente: [Pajares, 2006]

Un marco es muy similar a un objeto en el sentido en el que se definen estos dentro del paradigma de la programación orientada a objetos. Se puede establecer un paralelismo entre ellos. En esto se fundamentan algunos lenguajes de programación (Java, C++, etc.) como mecanismo basado en marcos para la representación del conocimiento. Por ejemplo se puede representar mediante marcos un subconjunto del dominio de conocimiento de los deportes. Todos los deportes tienen en común que los realizan personas. Algunos deportes exigen balón y otros no, y algunos de ellos requieren solamente una persona y otros varias. Como se puede observar en la figura algunos elementos heredan atributos.

Cada conexión como elemento único es una instancia.

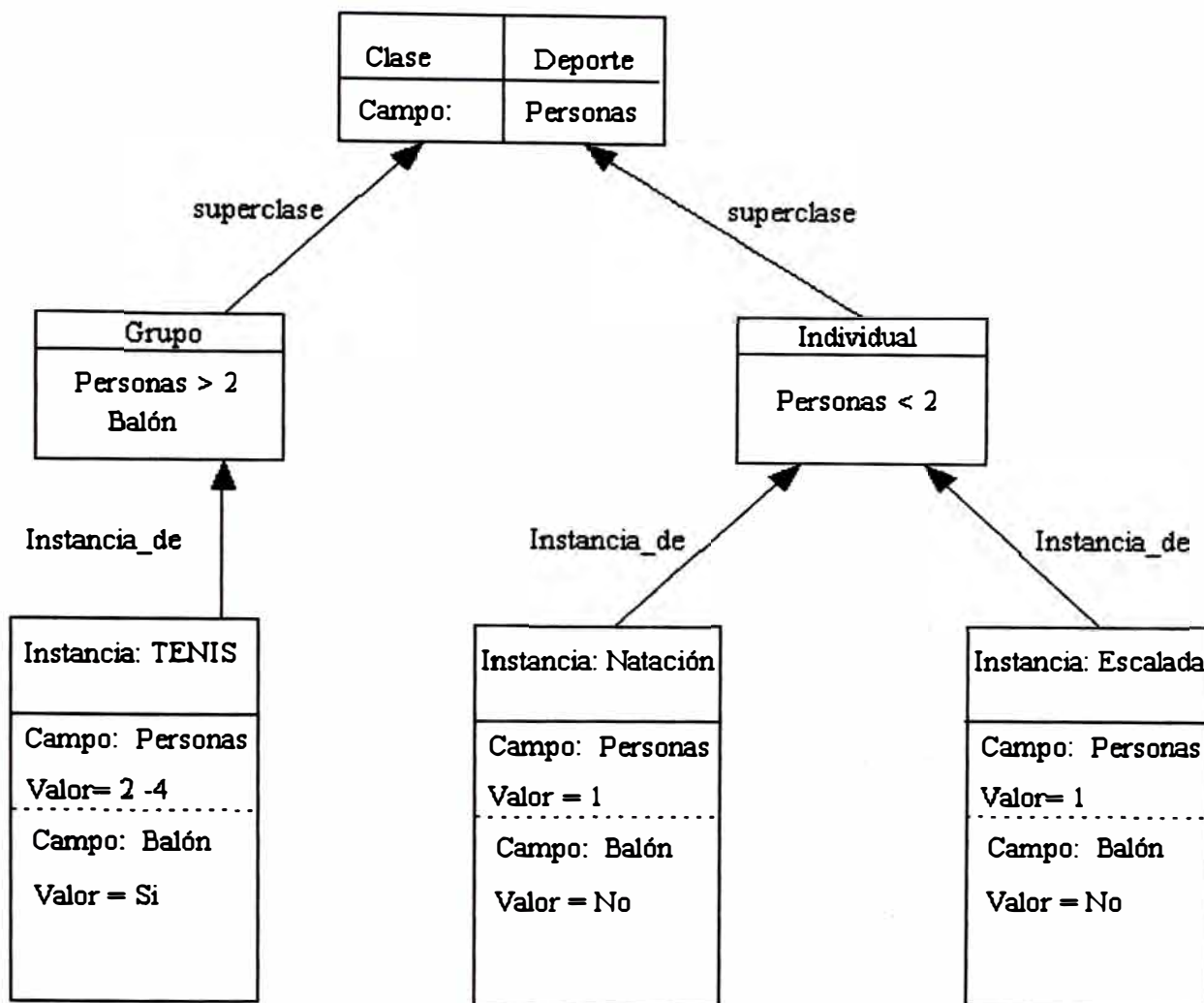


Figura 2.14: Representación mediante marcos.

Fuente: [Pajares, 2006.]

Cada marco tiene un conjunto de ranuras y los marcos están enlazados entre sí para mostrar las relaciones que los unen. La información fluye con un mecanismo denominado de activación por el que los marcos de nivel inferior se activan con los marcos de niveles superiores que han rellenado sus datos, y así sucesivamente con una estrategia descendente [PAJ2006]. Los SE basados en marcos son muy útiles para representar al conocimiento causal, porque su información está organizada en causas y efectos. Por el contrario, los SE basados en reglas

dependen generalmente del conocimiento no organizado, que no es causal[GIA1998].

2.3.3.5.1 EVALUACION DEL USO DE MARCOS COMO FORMA DE REPRESENTACION DE CONOCIMIENTO

Entre las ventajas tenemos:

- Los marcos permiten un rápido modelado del dominio, ya que las estructuras de representación son grandes y flexibles;
- Los marcos permiten obtener un modelo descriptivo del dominio de uso genérico, sin que esté estrechamente asociado a la forma como será utilizado;
- Los marcos son la forma de representación de conocimiento más utilizada en las herramientas de última generación para construcción de sistemas expertos, como el KEE (Intellicorp Co), ART Enterprise (Inference Co) y GOLDWORKS (GoldHill Co).

Las desventajas incluyen:

- Los marcos requieren gran capacidad de memoria en la computadora para almacenamiento.
- Los marcos son representados de forma jerárquica, luego la búsqueda aún es lenta.
- El mecanismo de inferencia por herencia de atributos, típico de marcos, es insuficiente para modelar la mayoría de los dominios de aplicación, exigiendo la combinación de estos con reglas de producción, aumentando la complejidad de la solución;

- Debido a la complejidad del recurso, los sistemas de marcos son mayores, más caros y demandan mayor tiempo de aprendizaje para ser utilizados que las otras formas de representación de conocimiento.

2.3.3.6 ÁRBOLES

Un **árbol** es una estructura jerárquica de datos conformada por **nodos** que almacenan información o conocimiento y por **ramas** que conectan a los nodos. A veces a las ramas se les denomina **vínculos** o **bordes** y a los nodos **vértices**. En la figura se muestra un árbol binario general y tiene cero, una o dos ramas por nodo. En un **árbol orientado**, el **nodo raíz** ocupa el lugar más alto de la jerarquía y las **hojas**, el más bajo. Un árbol puede considerarse como una clase especial de red semántica en la que cualquier nodo, excepto la raíz tiene exactamente un nodo **padre** y cero o más nodos **hijo**. Para el tipo usual de árbol binario hay un máximo de dos hijos por nodo y los nodos hijo de la derecha son diferentes a los de la izquierda [GIA1998].

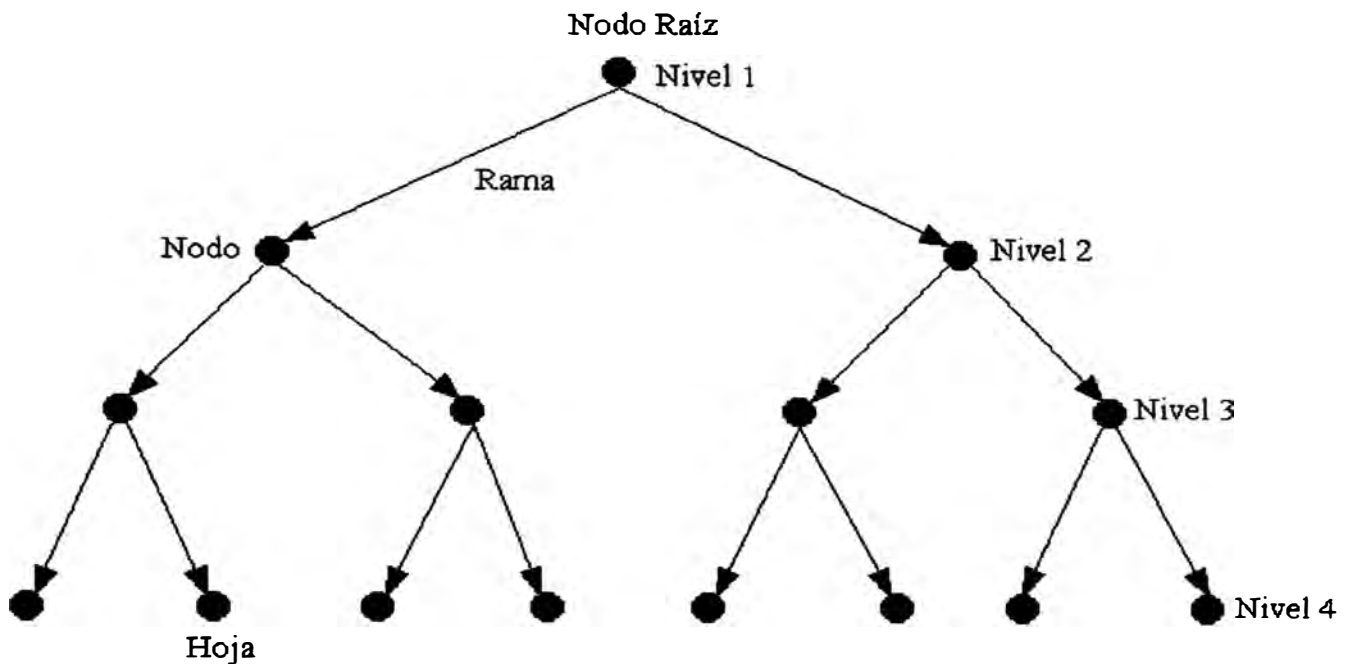


Figura 2.15: Árbol binario

Fuente: [Giarratano y Riley, 1998. p.97]

Si un nodo tiene más de un padre se encuentra en una red. Los árboles son un caso especial de grafo. El término *red* se usa como sinónimo de grafo cuando se describe un ejemplo particular de grafo como una red telefónica. Un grafo puede tener cero o más vínculos entre los nodos y ninguna distinción entre padres e hijos. Los vínculos pueden tener flechas o direcciones asociadas con ellos y un peso para diferenciarse. Si el grafo es la ruta de una línea aérea, los pesos pueden ser los kilómetros entre las ciudades, el costo del vuelo, el consumo de combustible, etc.

Los árboles y rejillas son útiles para clasificar objetos a causa de su naturaleza jerárquica. Una aplicación de los árboles y rejillas es la toma de decisiones, en estos casos se les llama **árboles** o **rejillas de decisión**. En lo sucesivo usaremos

el término **estructura** para referimos a ambos. Una *estructura de decisión* es tanto un esquema de representación del conocimiento como un método de razonamiento sobre su conocimiento. En la figura se muestra un *árbol de decisión* para clasificar animales. En un árbol de decisión los conjuntos más grandes de alternativas se examinan primero y después el proceso de decisión comienza a reducir las posibilidades a grupos más pequeños.

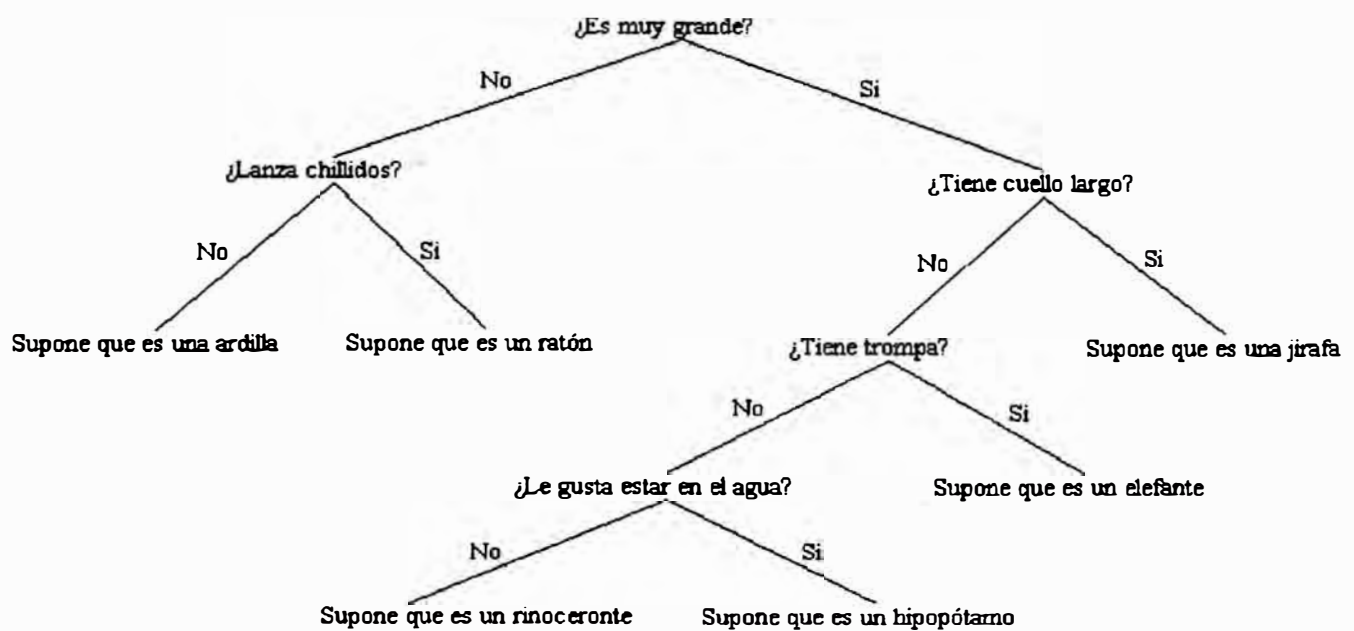


Figura 2.16: Árbol de decisión mostrando conocimiento acerca de animales

Fuente: [Giarratano y Riley, 1998. p.99]

Si las decisiones son binarias, un árbol de decisión binario es a la vez fácil de construir y muy eficiente. Otra característica útil de los árboles de decisión es que pueden realizar un **autoaprendizaje**. Si la suposición es errónea, puede llamarse a un procedimiento para pedir al usuario una nueva pregunta de clasificación correcta y las respuestas a las opciones "si" y "no". Deben crearse dinámicamente nuevos nodos, ramas y hojas y añadirse al árbol.

2.3.3.7 SISTEMAS PROBABILÍSTICOS

2.3.3.7.1 PROBABILIDAD CLÁSICA

La probabilidad es una herramienta cuantitativa para manejar la incertidumbre que se originó en el siglo XVII cuando apostadores franceses, debido a la popularidad alcanzada por los juegos de azar, buscaban métodos que les ayudaran a calcular sus posibilidades de ganar.

Pascal y Fermat propusieron por primera vez la teoría de la probabilidad clásica en 1654. Posteriormente se han desarrollado varias ramas nuevas y las aplicaciones de la probabilidad han servido a la ciencia, ingeniería y casi a todos los campos.

A la probabilidad clásica se le llamó también **probabilidad a priori**. El término a priori significa “antes”, es decir, sin tomar en cuenta a la realidad.

La fórmula fundamental de la probabilidad clásica se define como

$$P = \frac{W}{N}$$

donde W es el número de triunfos y N es el número de eventos igualmente posibles que son los posibles resultados de un experimento o prueba. La probabilidad de perder, Q , es

$$Q = \frac{N - W}{N} = 1 - P$$

Cuando una prueba se repite varias veces y arroja exactamente el mismo resultado, el sistema es **determinístico**, en caso contrario, es **no determinístico** o aleatorio. En algunas clases de sistemas expertos se utilizan shells no determinísticos para evitar desviación en el disparo de las reglas.

Espacio Muestral

El resultado de una prueba es un punto de muestra y el conjunto de todos los puntos de muestra posibles definen el espacio muestral. Un evento es un subconjunto del espacio muestral. Un evento simple sólo tiene un elemento y uno compuesto tiene más de uno.

Una manera gráfica de determinar el espacio de la muestra consiste en construir un árbol de eventos. Supongamos que hay dos computadoras que funcionan, F, o que no funcionan, N. El espacio de la muestra es $\{FF, FN, NF, NN\}$

Dada una población conocida, la deducción nos permite hacer inferencias sobre la muestra desconocida; de otro lado dada una muestra conocida, la inducción nos permite hacer inferencias sobre la población desconocida.

2.3.3.7.2 TEORÍA DE LA PROBABILIDAD

Se puede crear una teoría formal de la probabilidad mediante tres axiomas:

Axioma 1: $0 \leq P(E) \leq 1$

A un evento seguro se le asigna la probabilidad 1 y a un evento imposible se le asigna la probabilidad 0.

Axioma 2: $\sum_i P(E_i) = 1$

La suma de todos los eventos mutuamente excluyentes es 1. Como corolario de este axioma se tiene que: $P(E) + P(E') = 1$

Axioma 3: $P(E_1 \cup E_2) = P(E_1) + P(E_2)$

Donde E_1 y E_2 son eventos mutuamente excluyentes.

A la teoría axiomática también se le llama **teoría objetiva de la probabilidad**.

Estos axiomas en particular fueron ideados por Kolmogorov.

Probabilidades Experimentales y Subjetivas

La probabilidad experimental define la probabilidad de un evento $P(E)$ como el límite de una distribución de frecuencia

$$P(E) = \lim_{N \rightarrow \infty} \frac{f(E)}{N}$$

Donde $f(E)$ es la frecuencia de los resultados de un evento para N resultados observados.

A este tipo de probabilidad se le llama **probabilidad a posteriori**, que significa, después del evento. La idea de ésta es medir la frecuencia con la que ocurre un evento, durante un gran número de pruebas, y, a partir de ésta, inducir la probabilidad experimental.

Una probabilidad subjetiva en realidad es una creencia u opinión expresada como una probabilidad. Las creencias y opiniones de un especialista desempeñan importantes papeles en los sistemas expertos.

Probabilidades Compuestas

A los eventos que no se afectan entre sí de ninguna manera se les denomina eventos independientes. Para dos eventos independientes su probabilidad es:

$$P(A \cap B) = P(A) \cdot P(B)$$

A dos eventos se les llama estocásticamente independientes si y sólo si la fórmula previa es verdadera. “**Estocástico**” significa suposición, suele utilizarse como sinónimo de probabilidad, azar u oportunidad.

La fórmula para la independencia mutua de N eventos requiere que se satisfagan 2^N ecuaciones:

$$P(A_1 \cap A_2 \cap \dots \cap A_N) = P(A_1) \cdot P(A_2) \cdot \dots \cdot P(A_N)$$

Donde los asteriscos significan que deben satisfacerse todas las combinaciones de un evento y su complemento.

Para el caso de unión de dos eventos se tiene la Ley Aditiva:

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

La Ley Aditiva para tres eventos es:

$$P(A \cup B \cup C) = P(A) + P(B) + P(C) - P(A \cap B) - P(A \cap C) - P(B \cap C) + P(A \cap B \cap C)$$

2.3.3.7.3 PROBABILIDADES CONDICIONALES

Los eventos que no son mutuamente excluyentes se influyen entre sí. Saber que un evento ha ocurrido puede llevarnos a revisar la probabilidad de que otro evento ocurrirá.

Ley Multiplicativa

A la probabilidad de un evento A, dado que ocurrió el evento B, se le llama **probabilidad condicional** y se define como:

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \text{ para } P(B) \neq 0$$

$P(B)$ es la probabilidad a priori o previa. Se le llama probabilidad no condicional o probabilidad absoluta.

La ley multiplicativa de la probabilidad para dos eventos se define como:

$$P(A \cap B) = P(A|B) \cdot P(B)$$

Que es equivalente a: $P(A \cap B) = P(B|A) \cdot P(A)$

La ley multiplicativa para tres eventos es la siguiente:

$$P(A \cap B \cap C) = P(A|B \cap C) \cdot P(B|C) \cdot P(C)$$

Y la ley multiplicativa generalizada es:

$$P(A_1 \cap A_2 \cap \dots \cap A_n) = P(A_1|A_2 \cap \dots \cap A_n) \cdot P(A_2|A_3 \cap \dots \cap A_n) \cdot \dots \cdot P(A_{n-1}|A_n) \cdot P(A_n)$$

Si cualquiera de las siguientes ecuaciones es verdad, entonces los eventos A y B son independientes:

$$P(A|B) = P(A)$$

$$P(B|A) = P(B)$$

$$P(A \cap B) = P(A) \cdot P(B)$$

2.3.3.7.4 TEOREMA DE BAYES

La probabilidad inversa establece la probabilidad anterior dado que ha ocurrido uno posterior. Este tipo de probabilidad sucede a menudo en el diagnóstico médico, donde los síntomas aparecen y el problema consiste en encontrar la causa más

probable. La solución a este problema es el Teorema de Bayes, también llamado fórmula de Bayes, regla de Bayes o ley de Bayes, en honor al clérigo y matemático del siglo XVIII, Thomas Bayes.

La forma general del teorema de Bayes puede escribirse mediante eventos, E, e hipótesis(suposiciones), H, en las siguientes formas alternas:

$$\begin{aligned} P(H_i | E) &= \frac{P(E \cap H_i)}{\sum_j P(E \cap H_j)} \\ &= \frac{P(E|H_i)P(H_i)}{\sum_j P(E|H_j)P(H_j)} \\ &= \frac{P(E|H_i)P(H_i)}{P(E)} \end{aligned}$$

2.4 PERSONAS INVOLUCRADAS EN LA CONSTRUCCIÓN DE SISTEMAS EXPERTOS

Revisaremos ahora que profesionales se necesita para desarrollar un sistema experto y qué habilidades se requieren. En general, hay cinco miembros en el equipo de desarrollo de sistemas expertos: el experto del dominio, el ingeniero del conocimiento, el programador, el director del proyecto y el usuario final. El éxito en la elaboración del sistema experto depende íntegramente de lo bien que los miembros trabajen juntos. Las relaciones básicas en el equipo de desarrollo se resumen en la Figura [NEG2005].

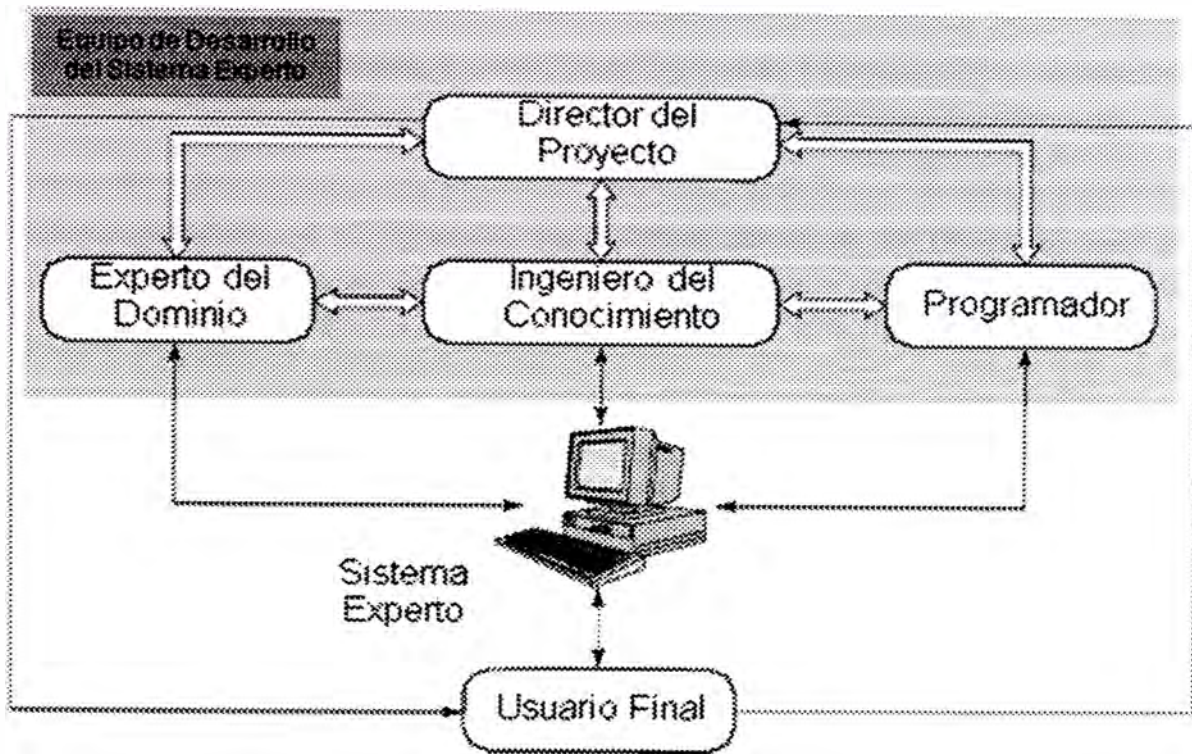


Figura 2.17: Actores principales del equipo de desarrollo del SE

Fuente: [Negnevitsky, 2005. p.29]

El **experto del dominio** es una persona con conocimientos y habilidades que lo hacen capaz de resolver problemas en un área específica o **dominio**. Esta persona tiene la mayor experiencia en un dominio dado. Esta experiencia será capturada en el sistema experto. Por lo tanto, el experto debe ser capaz de comunicar sus conocimientos, estar dispuesto a participar en el desarrollo del sistema experto y comprometer una cantidad considerable de tiempo al proyecto. El *experto del dominio* es el actor más importante en el equipo de desarrollo del sistema experto.

El **ingeniero del conocimiento** es aquel que es capaz de diseñar, construir y probar un sistema experto. Esta persona es responsable de seleccionar una tarea apropiada para el sistema experto. Él entrevista al *experto del dominio* para averiguar cómo resuelve un determinado problema. A través de la interacción con

el *experto*, el *ingeniero del conocimiento* establece qué métodos de razonamiento utiliza el *experto* para manejar los hechos y las reglas y decide cómo representarlos en el sistema experto. El *ingeniero del conocimiento* a continuación, elige algún software de desarrollo o una Shell de sistema experto, o mira los lenguajes de programación para la codificación del conocimiento (y, a veces lo codifica el mismo). Y por último, el *ingeniero del conocimiento* es responsable de las pruebas, revisión e integración del sistema experto en el lugar de trabajo. Por lo tanto, el *ingeniero del conocimiento* se compromete con el proyecto desde el diseño inicial hasta la entrega final del sistema experto, e incluso después de finalizado el proyecto, él también puede estar implicado en el mantenimiento del sistema.

El **programador** es la persona encargada de la programación, describe el conocimiento del dominio de modo que pueda ser codificado en una computadora. El *programador* necesita tener conocimientos de programación simbólica en lenguajes de IA como LISP, Prolog y OPS5 y también un poco de experiencia en la aplicación de diferentes tipos de Shells de sistemas expertos. Además, el *programador* debe conocer los lenguajes de programación convencionales como C, Delphi, FORTRAN y Basic.

Si se piensa utilizar un Shell de sistema experto, el *ingeniero del conocimiento* puede codificar el conocimiento en el sistema experto y eliminar así la participación del *programador*. Sin embargo, si no puede ser utilizada una Shell, el *programador* debe desarrollar las estructuras de representación de datos y el conocimiento (base de conocimientos y base de datos), la estructura de control (motor de inferencia) y la estructura de diálogo (interfaz de usuario). El *programador* también puede estar involucrado en la prueba del sistema experto.

El **director del proyecto** es el líder del equipo de desarrollo del sistema experto, es el responsable de mantener el proyecto en marcha. Se asegura de que todos los

entregables e hitos se cumplan, interactúa con el *experto*, *ingeniero del conocimiento*, *programador* y el *usuario final*.

El **usuario final**, a menudo llamado simplemente el **usuario**, es la persona encargada de utilizar el sistema experto. El *usuario* podría ser un químico analítico para determinar la estructura molecular del suelo de Marte (Feigenbaum, 1971), un joven médico para diagnosticar una enfermedad infecciosa de la sangre (Shortliffe, 1976), o un geólogo de exploración, tratando de descubrir un yacimiento nuevo (Duda, 1979). Cada uno de estos *usuarios* de sistemas expertos tiene necesidades diferentes, que el sistema debe cumplir; la aceptación definitiva del sistema dependerá de la satisfacción del usuario. El *usuario* no sólo debe tener confianza en el rendimiento del sistema experto, sino también debe sentirse cómodo usándolo. Por lo tanto, el diseño de la interfaz de usuario del sistema experto es también vital para el éxito del proyecto, la contribución del *usuario final* aquí puede ser crucial.

El desarrollo de un sistema experto puede empezar cuando se integren al equipo los cinco actores. Sin embargo, ahora se desarrollan muchos sistemas expertos en computadoras personales, con Shells de sistemas expertos. Esto puede eliminar la necesidad del *programador* y también puede reducir el papel del *ingeniero del conocimiento*. Para sistemas expertos pequeños, el *director del proyecto*, el *ingeniero del conocimiento*, el *programador* e incluso el *experto* podrían ser la misma persona. Pero cuando se desarrollan los grandes sistemas expertos es obligatoria la participación de todo el equipo de desarrollo [NEG2005].

2.5 VENTAJAS Y DESVENTAJAS DE LOS SE

Entre las ventajas de los Sistemas Expertos tenemos [NEG2005]:

a) Representación del conocimiento natural

Un experto generalmente explica el procedimiento de solución del problema con expresiones como esta: 'En tal y tal situación, debo hacer esto y lo otro'. Estas expresiones se pueden representar con toda naturalidad como reglas de producción *si-entonces*.

b) Estructura uniforme

Las reglas de producción tienen la estructura uniforme *si-entonces*. Cada regla es una pieza independiente de conocimiento. La sintaxis de las reglas de producción les permite ser auto-documentadas.

c) Separa el conocimiento de su procesamiento

La estructura de un sistema experto basado en reglas proporciona una separación efectiva entre la base de conocimientos y el motor de inferencia. Esto hace posible desarrollar diferentes aplicaciones que utilicen la misma caparazón del sistema experto. También permite una expansión elegante y fácil del sistema experto.

Para hacer al sistema más inteligente, un ingeniero del conocimiento se limita a añadir algunas reglas a la base de conocimientos sin intervenir en la estructura de control.

d) Trata con conocimiento incompleto e incertidumbre

La mayoría de los sistemas expertos basados en reglas son capaces de representar y razonar con conocimiento incompleto e incierto. Por ejemplo, la regla

Si la temporada es otoño

Y el cielo está "nublado"

Y el viento es escaso,

ENTONCES

pronóstico es claro (FC 0,1);

pronóstico es llovizna (FC 1,0);

pronóstico es de lluvia (FC 0,9)

podría ser utilizada para expresar la incertidumbre de la siguiente declaración:

"Si la temporada es el otoño y parece que llovizna, entonces probablemente será otro día lluvioso".

La regla representa la incertidumbre a través de números llamados factores de certidumbre.

El sistema experto utiliza los factores de certidumbre para establecer el grado de confianza o el nivel de creencia de que la conclusión de la regla es verdadera.

e) Disponibilidad

En contraste con el humano, un sistema experto tiene toda la experiencia en su interior, nunca se cansa o se muere, excepto cuando la máquina que almacena el programa o esas reglas se corrompe o se daña.

f) Coherencia

Una vez que un sistema experto está programado para preguntar y usar datos inciertos, este no es propenso al olvido. Si una línea de razonamiento es aceptable, lo seguirá siendo en diferentes consultas.

g) Exhaustivo

En algunos dominios, un sistema experto puede encapsular el conocimiento de más de un experto y por lo tanto ofrecer varias opciones.

Asimismo los SE tienen tres desventajas principales [NEG2005]:

a) Relaciones opacas entre las reglas

A pesar que las reglas de producción individuales tienden a ser relativamente simples y auto-documentadas, sus interacciones lógicas dentro del amplio conjunto de reglas puede ser opaco. Los sistemas basados en reglas hacen que sea difícil de observar cómo las reglas individuales sirven a la estrategia global. Este problema está relacionado con la falta de representación del conocimiento jerárquico en los sistemas expertos basados en reglas.

b) Estrategia de búsqueda ineficaz

El motor de inferencia aplica una búsqueda exhaustiva a través de todas las reglas de producción durante cada ciclo. Los sistemas expertos con un amplio conjunto de

reglas (más de 100 reglas) pueden ser lentos, y por lo tanto los grandes sistemas basados en reglas pueden no ser adecuados para aplicaciones en tiempo real.

c) Incapacidad para aprender

En general, los sistemas expertos basados en reglas no tienen la capacidad de aprender de la experiencia. A diferencia de un experto humano, quién sabe cuándo "romper las reglas", un sistema experto no puede automáticamente modificar su base de conocimientos, o adaptar las reglas existentes o agregar otras nuevas. El ingeniero del conocimiento sigue siendo responsable de la revisión y el mantenimiento del sistema.

En consecuencia, debido a las desventajas encontradas en los sistemas expertos basados en reglas se optó por desarrollar un sistema experto basado en razonamiento probabilístico.

CAPÍTULO III

DOMINIO DEL PROBLEMA

En este capítulo se describirá el dominio del Problema referente al síndrome de flujo vaginal y las patologías que la conforman. Como primer paso se explicará las entidades que comprenden el proceso de diagnóstico médico.

3.1 ELEMENTOS DEL DIAGNÓSTICO MÉDICO

3.1.1 SIGNO

Un signo es la manifestación objetivable consecuyente a una enfermedad o alteración de la salud, y que se hace evidente en la biología del enfermo (eMedicine, 2010).

Un signo es la evidencia objetiva de una enfermedad, en especial cuando ésta es observada e interpretada más por el médico que por el paciente. Un signo físico es una indicación de la condición corporal que puede ser percibida directamente (por ejemplo, a través de la auscultación) cuando el médico examina al paciente. El signo es parte principal del proceso de diagnóstico médico, ya que su principal virtud es el ser un dato objetivo.

Como ejemplo de signos se pueden citar los siguientes asociados al diagnóstico de la enfermedad Vaginitis por candida.

Diagnóstico	Vaginitis por candida
Signos	Flujo vaginal escaso Flujo vaginal blanco-amarillo Flujo vaginal grumoso Flujo vaginal indiferente

Tabla 3.1: Signos asociados a una enfermedad

Fuente: [Elaboración Propia]

Todos ellos se obtienen durante el proceso de exploración física y por lo tanto sólo son detectados por el médico que realiza la exploración.

3.1.2 SÍNTOMA

Un síntoma es la referencia subjetiva que da un enfermo por la percepción o cambio que reconoce como anómalo, o causado por un estado patológico o enfermedad. (Liddell & Scott, 2010)

El síntoma es cualquier prueba subjetiva de enfermedad o del estado de un paciente. Se refiere a los datos proporcionados por el paciente, relacionados con alguna enfermedad. Son subjetivos, matizados por el padecimiento y las características psicológicas, sociológicas y biológicas del paciente.

Durante el proceso de diagnóstico médico, los síntomas ayudan a localizar al sistema anatómico que pudiera estar comprendido o asociado a la enfermedad del paciente. También existen síntomas totalmente inespecíficos y que pudieran estar, por lo tanto, asociados a muchas enfermedades y aunque no orientan directamente

al sistema anatómico afectado, si refieren sobre la intensidad o evolución del padecimiento.

Como ejemplo de síntoma se pueden citar el siguiente asociado al diagnóstico de la enfermedad Vaginitis por Candida:

Diagnóstico	Vaginitis por Candida
Síntoma	Prurito

Tabla 3.2: Síntomas asociados a una enfermedad

Fuente: [Elaboración Propia]

3.1.2.1 PATOGNOMÓNICO

El término patognomónico se utiliza para denominar aquellos signos (manifestaciones visibles), síntomas (manifestaciones no visibles, subjetivas) o pruebas diagnósticas que, si están presentes, confirman que el sujeto padece un determinado trastorno. Así por ejemplo, el nivel de pH entre 5.0 a 7.0 es patognomónico para el diagnóstico de la tricomoniasis.

3.1.3 PRUEBAS DE LABORATORIO

Las pruebas de laboratorio, conocidas también como análisis clínico hacen referencia comúnmente a "la exploración complementaria solicitada a un laboratorio clínico por un médico para confirmar o descartar un posible diagnóstico" [JAC1990]. Este elemento en determinadas situaciones es clave para confirmar o desmentir un determinado diagnóstico.

En las pruebas de laboratorio, la principal característica que se considera en estas entidades es el hecho de que esta prueba esté confirmando o no un determinado hecho, es decir, que su valor sea positivo o negativo. Esta es una simplificación

demasiado general, pero necesaria para la conceptualización del dominio a diseñar.

Las pruebas de laboratorio pueden ofrecer valores más amplios que el simple “positivo” o “negativo” (valores numéricos fundamentalmente). Sin embargo, en ciertos casos se puede asumir la posibilidad de asociar un determinado valor numérico de una prueba de laboratorio a una conclusión positiva o negativa, que en este caso el experto ha de tener presente. Las pruebas de laboratorio forman parte de los signos que presenta el paciente.

Como ejemplo de prueba de laboratorio, se puede citar el siguiente asociado al diagnóstico de la enfermedad Vaginosis bacteriana:

Diagnóstico	Vaginosis bacteriana
Prueba de Laboratorio	Test de Aminas

Tabla 3.3: Prueba para descartar una enfermedad

Fuente: [Elaboración Propia]

3.1.4 SÍNDROME

Un síndrome es el grupo de signos y síntomas que se manifiestan simultáneamente y que, considerados como un todo, son característicos de cierta enfermedad. Existen procesos mórbidos, o grupo de enfermedades que se caracterizan siempre por la presencia de ciertos signos y síntomas, y cuando éstos no están presentes en su totalidad, no es posible arribar a ese diagnóstico sindromático. En el proceso diagnóstico, el síndrome puede ocupar el diagnóstico final o el paso previo a este diagnóstico cuando no se puede encontrar la causa definitiva y única de ese síndrome. Como ejemplo de síndrome podemos citar el síndrome de flujo vaginal, el cual se caracteriza por la presencia de descenso vaginal anormal. Este síndrome es percibido por la paciente(síntoma).

Como ejemplo de síndrome podemos citar el siguiente, asociado al diagnóstico de la enfermedad vaginosis bacteriana:

Diagnóstico	Vaginitis bacteriana
Síndrome	Flujo Vaginal

Tabla 3.4: Ejemplo de Síndrome

Fuente: [Elaboración Propia]

3.1.5 ENFERMEDAD

Relacionado a la nosología (del griego nosos: enfermedad y logos: palabra, razón, tratado), la enfermedad es un dato físico y funcional, manifestado por síndromes y comprobado por análisis de laboratorio.

Como ejemplo de las enfermedades en este dominio podemos citar a la Vaginosis Bacteriana, que es una enfermedad infecciosa, la cual es de las más frecuentes entre las mujeres.

3.1.6 DIAGNOSTICO DIFERENCIAL

En Medicina, el diagnóstico diferencial, se define como “el procedimiento por el cual se identifica una determinada enfermedad, entidad nosológica, síndrome, o cualquier condición de salud-enfermedad mediante la exclusión de otras posibles causas que presenten un cuadro clínico semejante al que el paciente padece” [Marecos, 2003].

El diagnóstico diferencial además se caracteriza por proponer, por lo general, una serie de posibles diagnósticos (el conocido como diagnóstico, a secas, se consideraría como la conclusión final a la que el experto médico ha llegado tras

desestimar otras posibles opciones mediante la información de la que se disponga), con el objetivo de llegar al mencionado diagnóstico final.

3.2 INFECCIONES DE TRANSMISIÓN SEXUAL

Las Infecciones de Transmisión Sexual (ITS) están entre las condiciones descritas más antiguas en la historia médica. Referencias a la gonorrea pueden hallarse en la Biblia. El número de protozoarios, bacterias, virus e infecciones ectoparasitarias que han sido identificadas como transmisoras sexuales han continuado incrementándose desde tiempos muy remotos.

El control de las ITS es en la actualidad un asunto principal de salud pública, especialmente porque hay una creciente apreciación que individuos infectados con otras ITS son más susceptibles a la infección con HIV.

En octubre de 2010 el Instituto Nacional de Estadística e Informática (INEI), el Programa Conjunto de las Naciones Unidas sobre el VIH/SIDA (ONUSIDA) y el Fondo de Población de las Naciones Unidas (UNFPA), publicaron una investigación acerca del conocimiento de los peruanos sobre el VIH e ITS titulada: *“Perú: Conocimiento, Actitudes y Autopercepción de los varones de 15 a 59 años sobre el VIH e ITS, 2008”*.

La evidencia basada en tratamientos y cura es efectiva para la mayoría de ITS. La prevención primaria de ITS se enfoca en consejería en individuos en alto riesgo en un intento de controlar este problema de salud principal. Aunque la tasa de incidencia de muchas ITS es alta entre adolescentes, es esencial que la prevención, consejería, exámenes y estrategias de tratamiento incluyan las necesidades especiales de éste grupo etéreo. El Ministerio de Salud(MINSA) promueve periódicamente estrategias orientadas a tal fin. Campañas tales como “Tú PreVIHenes Infórmate” implementada en el año 2008 por El Ministerio de

Salud, la Coordinadora Nacional Multisectorial en Salud (CONAMUSA) y la Estrategia Sanitaria Nacional de Prevención y Control de ITS, VIH y Sida, se vienen trabajando, con el objetivo de fortalecer los conocimientos sobre las infecciones de transmisión sexual y el VIH.

3.3 DEFINICIÓN DE INFECCIONES DE TRANSMISIÓN SEXUAL

Las infecciones de transmisión sexual⁴ (ITS) (también conocidas como enfermedades de transmisión sexual (ETS), antes enfermedades venéreas) son un conjunto de afecciones clínicas infectocontagiosas que se transmiten de persona a persona por medio de contacto sexual que se produce, casi exclusivamente, durante las relaciones sexuales, incluido el sexo vaginal, el sexo anal y el sexo oral, especialmente si no se utiliza preservativo o condón; también por uso de jeringuillas contaminadas o por contacto con la sangre, y algunas de ellas pueden transmitirse durante el embarazo, es decir, de la madre al hijo.

La mayor parte de las enfermedades de transmisión sexual son causadas por dos tipos de gérmenes: bacterias y virus, pero algunas también son causadas por hongos y protozoarios.

En 1996, la OMS estimaba que más de 1 millón de personas se infectaban diariamente. Cerca del 60 por ciento de estas infecciones ocurren entre menores de 25 años, y el 30 por ciento de éstos tienen menos de 20 años. Entre los 14 y los 19 años de edad, las ITS ocurren con más frecuencia en muchachas que muchachos en una proporción casi de 2:1; esto se iguala en ambos sexos hacia los 20 años.

⁴ La Organización Mundial de la Salud prefiere el término *infecciones de transmisión sexual*, pues en muchos casos las personas pueden estar infectadas sin manifestar ninguno de los síntomas.

Existen más de 20 tipos de enfermedades venéreas entre las cuales la más mortal es el SIDA producido por el virus de inmunodeficiencia humano o VIH . Otras infecciones transmitidas sexualmente incluyen:

- Clamidia
- Gonorrea
- Herpes genital
- VIH/SIDA
- VPH
- Sífilis
- Tricomoniasis

3.4 CAUSAS DE LAS INFECCIONES DE TRANSMISIÓN SEXUAL

Una enfermedad infecciosa es aquella que puede ser transmitida de una persona a otra. Las ITS forman una categoría de enfermedades transmitidas mediante actividades sexuales, tales como penetración vaginal, sexo oral o anal. [X-PLAIN, 2012].

Las enfermedades venéreas son causadas por uno de los 3 siguientes tipos de organismos:

1. Parásitos: Los parásitos son animales muy pequeños que se alimentan de la persona a la cual infectan. La persona infectada se llama portador. Pulgas púbicas también llamadas ladillas, son un ejemplo de parásitos transmitidos sexualmente.

2. Bacterias: Las bacterias son organismos hechos de una célula. Permanecen juntos para alimentarse del cuerpo de su portador. Gonorrea, sífilis, clamidia y chancro blando son ejemplos de bacterias transmitidas sexualmente.

3. Virus: Los virus son moléculas complejas que pueden multiplicarse cuando invaden las células de su portador. El VIH, el herpes y la hepatitis son ejemplos de virus transmitidos sexualmente.

En su gran parte las infecciones causadas por un virus no pueden ser curadas, pero en algunos casos pueden ser controladas.

Las infecciones parasitarias pueden ser tratadas con medicamentos recetados por el médico o de venta libre.

Las infecciones causadas por bacterias pueden ser tratadas con antibióticos.

El objetivo central del manejo de casos de ITS es el de hacer un diagnóstico adecuado que permita dar un tratamiento oportuno en la primera visita. El manejo sintomático permite cumplir con este objetivo.

Los síndromes más frecuentes de ITS son:

- Síndrome de Flujo Vaginal.
- Síndrome de Descarga Uretral.
- Síndrome de Ulcera Genital.
- Síndrome de Dolor Abdominal Bajo.
- Síndrome de Bubón Inguinal.

3.5 SÍNDROME DE FLUJO VAGINAL

3.5.1 Flujo vaginal normal

Primero es importante definir que es el flujo vaginal normal o humedad normal de la vagina, y luego explicar cuando se convierte en flujo vaginal o descenso vaginal y como se reconoce. [MINSA2006]

El flujo vaginal normal o humedad de la vagina deriva de la trasudación de fluidos de los vasos capilares de la vagina mezclados con secreciones de las glándulas de Bartholino, Skene, endometrio, de las trompas de Falopio y del cérvix.

El flujo vaginal está constituido por agua, electrolitos y glucosa, y mantiene un pH menor de 4.5 que favorece el crecimiento de organismos de medio ácido (los lactobacilos) inhibiendo el crecimiento de otros. El conjunto de los organismos que normalmente viven en la vagina (flora vaginal) está compuesto principalmente por lactobacilos y otros organismos como *Staphilococcus epidermidis*, *Corynebacterias*, *Gardnerella vaginalis*, anaerobios y otros.

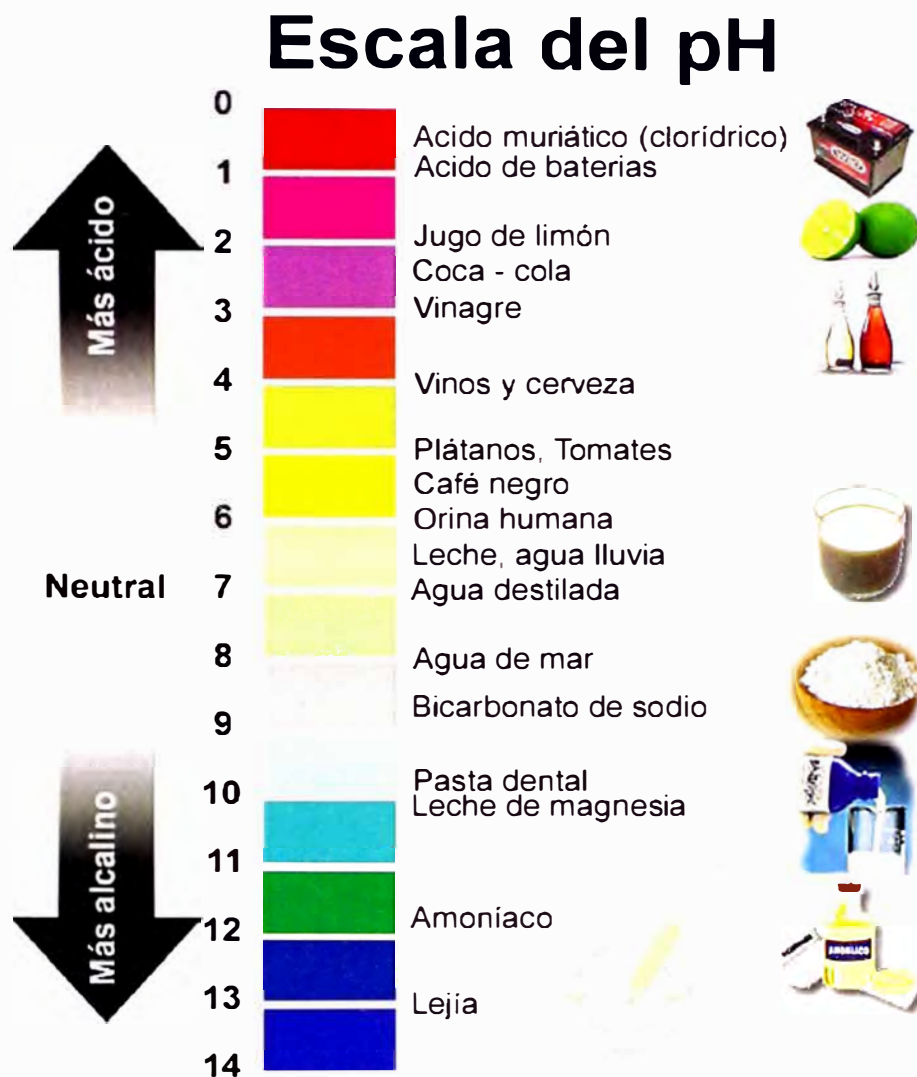


Figura 3.1: Escala del pH.

Fuente: [<http://amazings.es/2012/05/09/ph-para-adultos/>]

La flora vaginal mantiene a la vagina sana creando un ambiente ácido adverso para bacterias patógenas. La flora vaginal y su entorno se conoce como ecosistema vaginal y tiene varios reguladores importantes:

- Estrógenos: que afectan directamente la trasudación, a mayor nivel de estrógeno mayor flujo.
- Lactobacilos: que metabolizan los azúcares a ácido láctico, manteniendo el pH ácido, lo que inhibe el crecimiento de otras bacterias. Además producen H_2O_2 que inhibe el crecimiento de bacterias anaeróbicas.

El flujo vaginal normal o humedad normal de la vagina no es constante en cantidad y hay variaciones en consistencia. Aumenta cuando la persona se excita, durante la actividad sexual, antes y durante la ovulación y durante la gestación. Disminuye durante la lactancia y en el periodo menopáusico. Normalmente el olor es característico para cada mujer y el color transparente o discretamente blanquecino.

El equilibrio del ecosistema vaginal puede ser alterado por las duchas vaginales⁵, espermicidas, uso de antisépticos vaginales, etc. Igualmente, la menstruación puede condicionar algunos cambios, ya que eleva el pH vaginal. Debe tenerse en cuenta que un desbalance en este ecosistema puede ocasionar un aumento del flujo vaginal, o cambios en sus características (olor, color, consistencia) Los métodos anticonceptivos hormonales y los dispositivos intrauterinos también pueden ocasionar un aumento en el flujo de algunas mujeres.

3.5.2 Definición de Síndrome de Flujo Vaginal

Se considera síndrome de flujo vaginal al aumento de la cantidad del flujo vaginal, con cambio de color, cambio de olor (mal olor) y cambio de consistencia asociado a síntomas de prurito o irritación vulvar; en algunos casos con disuria, dispareunia, o dolor en el hemiabdomen inferior. En esta última situación, se deberá enfrentar el problema como un síndrome de dolor abdominal bajo y ser manejado como tal. [MINSA2006]

Este flujo vaginal o “descenso” como comúnmente se conoce, puede ser debido a un proceso inflamatorio localizado en la propia vagina ocasionado por trastornos del ecosistema vaginal o por la introducción de algún patógeno externo sexualmente transmitido como las infecciones por tricomonas.

⁵ Revisar Higiene Vaginal: <http://www.youtube.com/watch?v=r_7ShtozZAU&feature=player_embedded>

Si bien es cierto que el flujo vaginal es la afección ginecológica más común de las mujeres sexualmente activas, no todos los casos de flujo vaginal son ITS.

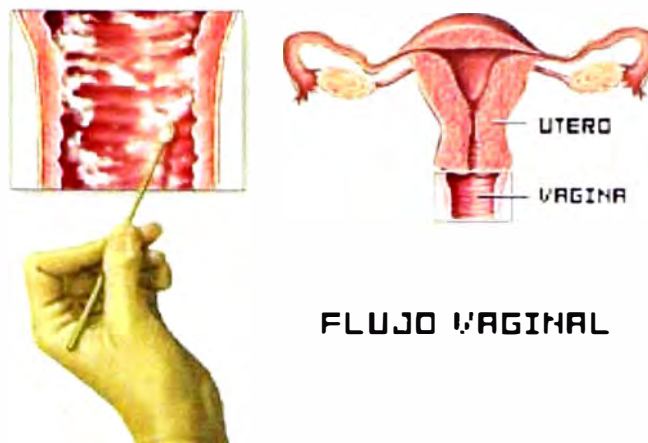


Figura 3.2: Flujo Vaginal.

Fuente: [<http://www.drgdiaz.com/eco/salud/flujos.shtml>]

3.5.3 Etiología

El descenso vaginal tiene diversas etiologías y localizaciones. Las principales causas de flujo vaginal anormal son las infecciones vaginales o vaginitis, entre ellas:

- Vaginosis bacteriana.
- *Trichomona vaginalis* (que si es sexualmente transmitida)
- Infección por *Candida* (especialmente *C. Albicans*)

El flujo vaginal anormal puede ser también causado por infecciones originadas en el cervix (cervicitis por clamidia y gonorrea, verrugas cervicales, ulceraciones del cervix, ejm. Herpes genital, chancroide, carcinoma) y el útero (Enfermedad inflamatoria pélvica, endometritis). Sin embargo, en el Perú, de lejos las causas más comunes de flujo vaginal anormal son las infecciones vaginales o vaginitis.

3.5.4 VAGINOSIS BACTERIANA

Es actualmente la infección vaginal más común. Anteriormente la vaginitis bacteriana se conocía como vaginitis no específica y vaginitis por Gardnerella. MOR2005]

3.5.4.1 Etiología

a. La vaginosis bacteriana es causada por un sobrecrecimiento de una variedad de especies de bacterias, particularmente anaerobios, que frecuentemente se encuentran en forma normal en la vagina. Los organismos más frecuentemente involucrados incluyen bacteroides, peptostreptococcus, Gardnerella vaginalis y hominis mycoplasma.

b. La bacteria anaeróbica produce enzimas que degradan los péptidos a aminoácidos y aminas resultando en compuestos asociados con la descarga y olor característico de esta infección.

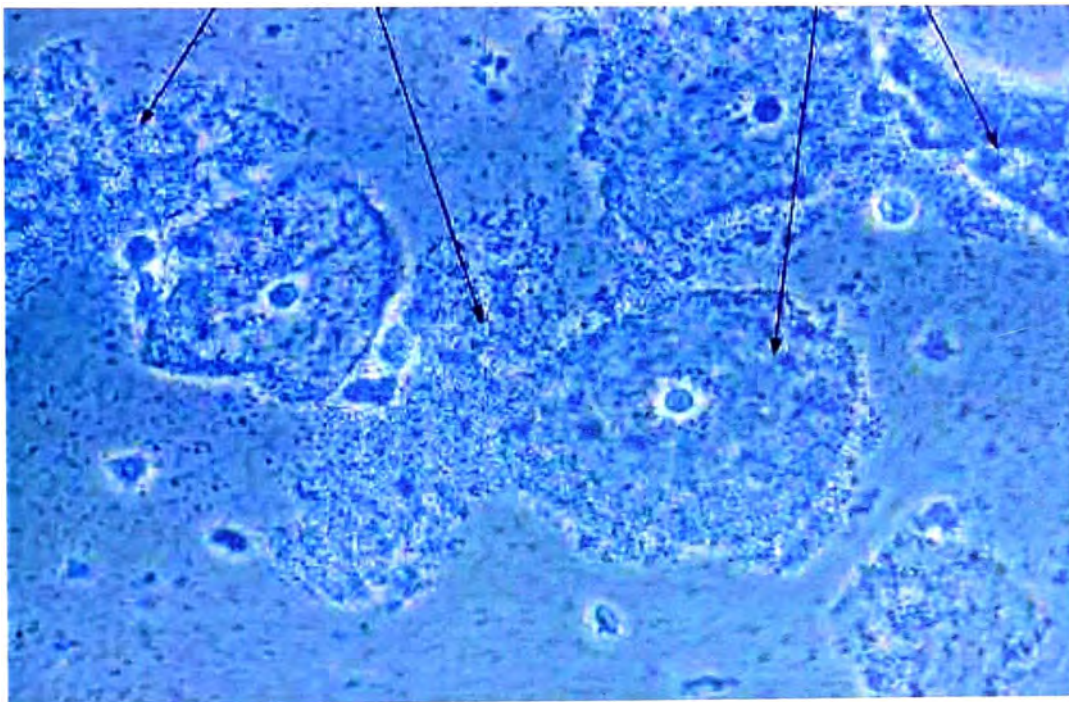


Figura 3.3: Células escamosas del cérvix cubiertas con la bacteria *Gardnerella vaginalis* (flechas).

Fuente: [http://es.wikipedia.org/wiki/Vaginosis_bacteriana]

3.5.4.2 Presentación clínica

Cincuenta por ciento de mujeres con vaginosis bacteriana son asintomáticas. En pacientes sintomáticos, la presentación más común es una descarga gris de olor desagradable.

3.5.4.3 Diagnóstico

Tres de los cuatro siguientes criterios deben estar presentes:

- a. El pH vaginal está generalmente entre 5.0 y 5.5
- b. Preparaciones salinas húmedas con cloruro de sodio revelan un fondo "limpio" con mínimo o ningún leucocito, abundantes bacterias, y las características células clave. Las células clave son células escamosas en la cual la bacteria cocobacilaria ha oscurecido los bordes y el citoplasma.
- c. Aplicación del 10% de KOH a la muestra en sal húmeda produce un olor a pescado, indicando un test positivo "WHIFF".
- d. Está presente una descarga gris, homogénea y de olor desagradable.

3.5.4.4 Tratamiento

La terapia está basada en el uso de agentes con actividad anaeróbica e involucra ambos agentes sistémicos. La combinación parece ser 90% efectiva.

a. Preparación vaginal

(1) Clindamicina al 2 % intravaginal en crema antes de acostarse por 7 días.

(2) Aplicar metrodinazol intravaginal dos veces al día por 5 días.

b. Regímenes Orales

(1) Metronidazol podría ser administrado de tres maneras: 50 mg dos veces diarias por 7 días, 250 mg tres veces diarias por 7 días o una dosis simple 2-g

(2) Clindamicina, 300 mg tres veces diarias por 7 días (podría estar asociado con diarreas especialmente por *c.difficile*)

c. La pareja sexual debería ser tratada en caso de episodios repetidos de vaginosis bacteriana. La rutina de tratamiento de parejas no ha sido probada para mejorar el porcentaje de cura o disminución del porcentaje de reinfección.

d. El tratamiento durante el embarazo es crítico; los datos sugieren una asociación de complicaciones del embarazo con la vaginosis bacteriana.

(1) La clindamicina podría ser usada durante el embarazo.

(2) Metronidazol podría ser usada después del primer trimestre.

e. Los pacientes con recurrencia deberían ser estudiados para descartar otras ITS.

3.5.5 CANDIDIASIS

Es la segunda infección vaginal más común [MOR2005].

3.5.5.1 Etiología

a. El agente etiológico es un organismo fúngico, usualmente *candida albicans*. El organismo es un habitante común del intestino y región perianal. Treinta por ciento de mujeres podría tener colonización vaginal y no tener síntomas de infección.

b. Varios factores podrían iniciar la infección sintomática en lugar de colonización.

(1) Prácticas contraceptivas (ejm. píldoras anticonceptivas orales y espermicidas los cuales influyen el pH vaginal)

(2) Uso de esteroides sistémicos, los cuales influyen el sistema inmune.

(3) Uso de antibióticos, los cuales alteran la microbiología de la vagina; 25 a 70 % de mujeres reportan infecciones por hongos después de uso de antibióticos. Cualquier antibiótico, en particular un agente de amplio espectro, podría jugar un rol causal.

(4) Ropas ajustadas, panty hose y duchas vaginales (los hongos se reproducen en entornos húmedos, oscuros y calientes)

(5) Diabetes mellitus no diagnosticada o no controlada.

(6) La fructosa causa candidiasis.

355 mililitros de	Coca Cola	Jugo de Naranja	Jugo de Manzana	Jugo de Cereza	Jugo de Uvas
Carbohidratos Totales	40 g	39 g	42 g	49.5 g	60 g
Carbohidratos de Azúcar	40 g	33 g	39 g	37.5 g	58.5 g
Azúcar (cucharaditas)	10	8	10	9	15
Calorías	160	165	156	210	240

Tabla 3.5: Comparación de calorías entre jugos y comida chatarra

Fuente: [<http://www.niunadietamas.com/blog/?p=7400>]

c. Otra razón para una infección por *Candida* refractaria podría ser el estado inmune comprometido, en una vulvovaginitis por *Candida* recurrente es necesaria una prueba para descartar HIV.

d. Ha habido un reciente incremento en el número de infecciones causadas por especies no albicans. Encima del 20% de infecciones podría ser causada por organismos tales como *Candida tropicalis* y *Torulopsis glabrata*. Estos organismos podrían ser resistentes a regímenes de tratamiento estándar.

3.5.5.2 Presentación Clínica

Los pacientes con vaginitis monilial se quejan característicamente de una densa descarga blanca y prurito vulvar extremo. La vulva podría estar roja e inflamada.

a. Los síntomas podrían recurrir y ser más prominentes sólo antes de la menstruación o en asociación con relaciones sexuales.

b. Las infecciones por hongos podrían ocurrir más frecuentemente durante el embarazo.

c. Los pacientes con infecciones causadas por *C. Tropicalis* y *T. Glabrata* podrían tener una presentación atípica. La irritación podría ser principalmente, con pequeña descarga o prurito.



Figura 3.4: Cultivo en una placa de agar de *Candida albicans*.

Fuente: [<http://es.wikipedia.org/wiki/Candidiasis>]

3.5.5.3 Diagnóstico

El diagnóstico es realizado por historia, examen físico y examen microscópico de la descarga vaginal en solución salina y 10% KOH.

- a. En el examen, podrían evidenciarse excoriaciones de la vulva; la vulva y la vagina podrían estar eritematosas, con parches de descarga como requesón adherente.
- b. La infección con *C.tropicalis* y *T.glabrata* podría no estar asociada con la descarga clásica; la descarga podría ser blanca – gris y fina.
- c. El pH vaginal podría ser normal o ligeramente más bajo que el normal(4.0 a 4.7)
- d. El examen microscópico de una muestra húmeda revela hifas o pseudohifas con gemación en 50 a 70% de mujeres con infección por hongos.

e. Los cultivos no son necesarios para realizar el diagnóstico excepto en algunos casos de infecciones recurrentes.

3.5.5.4 Tratamiento

Muchos agentes están disponibles para el tratamiento de candidiasis vulvovaginal. Estos incluyen agentes tópicos, los cuales deberían estar disponibles para ser comprados sin receta médica o por prescripción, y agentes orales, los cuales están disponibles sólo por prescripción.

a. Agentes intravaginales antifúngicos son administrados como supositorios o cremas. Estas drogas están disponibles en tres regímenes: una dosis simple por 3 días o 7 días. Los agentes incluyen butoconazol, clotrimazol, miconazol, tioconazol. Regímenes sin receta médica deberían ser usados solo por mujeres que han sido diagnosticadas con una infección por hongos en el pasado y presentan síntomas de experiencia idéntica.

b. Los agentes orales incluyen fluconazol y ketoconazol.

(1) El fluconazol está disponible como un tratamiento de dosis simple(150 mg) para candidiasis vaginal no complicada.

(2) El ketoconazol es usado efectivamente para el tratamiento de candidiasis crónica y recurrente; una incidencia del 5 % de hepatotoxicidad limita su uso frecuente.

El esquema de dosificación es 200 mg. dos veces al día durante 5 días, luego 100 a 200 mg diarios durante 6 meses.

c. Podrían ser efectivas las cápsulas intravaginales de ácido bórico, 600 mg durante 14 días.

5. Infecciones crónicas recurrentes por hongos (5% de mujeres)

En la mayoría de casos, ningún factor exacerbante puede ser hallado; de allí deberían ser consideradas las siguientes posibilidades:

- a. Falla para completar el tratamiento completo.
- b. Infección por HIV. Candidiasis recalcitrante podría ser un síntoma presente en mujeres con infección por HIV. Pruebas para descartar el HIV deberían ser consideradas y ofrecerlas al paciente.
- c. Terapia antibiótica crónica.
- d. Infección con un organismo resistente tal como *C. tropicalis* o *T. glabrata*.
- e. Transmisión sexual de la pareja masculina.
- f. Reacción alérgica al semen de la pareja o un espermicida vaginal.

3.5.6 TRICOMONIASIS

Es la tercera vaginitis más común, conformando el 25% de los casos.

3.5.6.1 Etiología

El protozoo móvil *T. vaginalis* es el agente etiológico.

La tricomona puede ser recuperada de 70 a 80% de las parejas masculinas del paciente infectado; de allí la vaginitis por tricomonas es una ITS.

Trichomoniasis (*Trichomonas vaginalis*)

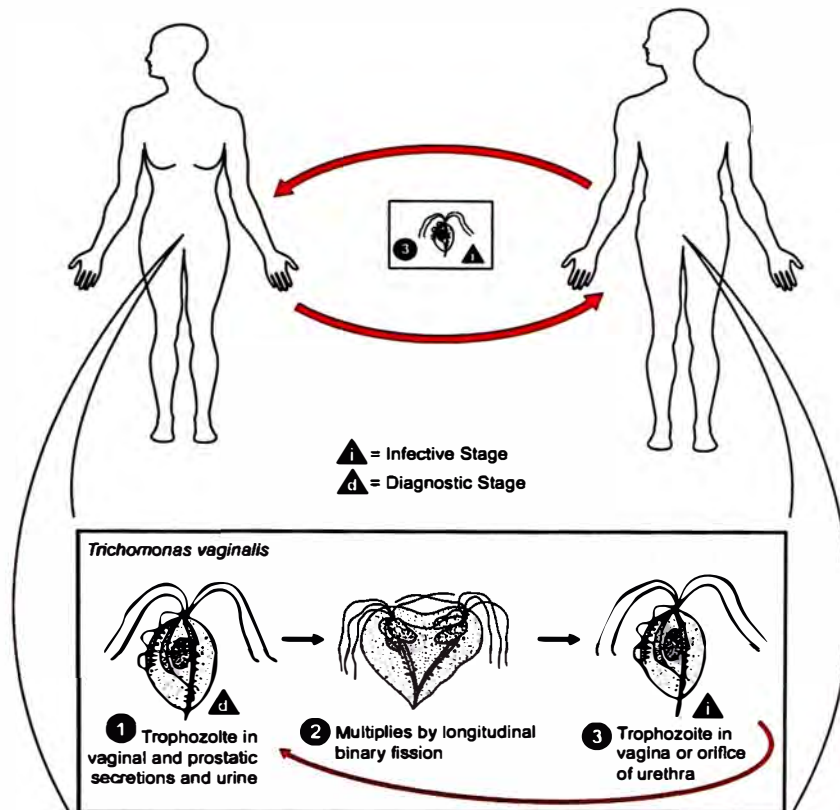


Figura 3.5: Epidemiología de Trichomoniasis.

Fuente: [<http://es.wikipedia.org/wiki/Tricomoniasis>]

3.5.6.2 Presentación Clínica

La vaginitis por tricomonas es una infección multifocal involucrando el epitelio vaginal, glándulas de Skene, glándulas de Bartholino y la uretra.

a. A menos que se pregunte directamente el 25 a 50% de mujeres podrían no reportar síntomas.

b. La mayoría de mujeres reportan una descarga que la describen como copiosa, verde y espumosa. La descarga podría estar asociada con un olor desagradable e irritación vulvar o prurito.

3.5.6.3 Diagnóstico

a. Examen Físico

Podría observarse evidencia clásica de tricomoniasis.

(1) Podría evidenciarse la característica descarga verde.

(2) Lesiones puntiformes, descrita clásicamente como el *cervix frambuesa* es evidente en sólo el 25% de pacientes.

b. Pruebas de Laboratorio

(1) El pH vaginal está usualmente entre 5.0 y 7.0.

(2) Una muestra húmeda salina de la descarga vaginal revela numerosos leucocitos y tricomonas altamente móviles y flageladas (alrededor del 75% de casos)

(3) Los cultivos no son necesarios usualmente para emitir el diagnóstico. Estos deberían obtenerse cuando el diagnóstico se sospecha pero no puede ser confirmado por examen de la muestra húmeda.

(4) PAP extendido podría ser positivo en el 65% de los casos. El PAP extendido positivo debería ser confirmado por examen de muestra húmeda debido al alto porcentaje falso-positivo.

4. Tratamiento

Debido a que la tricomonas es transmitida sexualmente, la pareja requiere terapia; 25 % de mujeres podrían ser reinfectadas si su pareja no recibe tratamiento.

a. La terapia vaginal sola es poco efectiva debido a los múltiples lugares de infección y son necesarios agentes sistémicos.

b. Si ambas parejas son tratadas simultáneamente, el porcentaje de cura es alcanzado en un 90% con tratamiento con metronidazol. Los pacientes deberían ser advertidos que una reacción como *disulfiram* podría ocurrir y que ellos deberían abstenerse de ingerir alcohol durante el tratamiento.

(1) El régimen preferido es 2g en una dosis debido a la fácil aceptación. Cerca al 10% de pacientes podrían experimentar vómitos.

(2) Un régimen alternativo es 500 mg, 2 veces diarias durante 7 días o 250 mg 3 veces diarias durante 7 días.

d. Casos resistentes podrían requerir tratamiento con metronidazol intravenoso. Debido a que la resistencia es rara, otras causas tales como rechazo del paciente o pareja deberían ser consideradas.

e. Metronidazol está contraindicado para usarlo durante el primer trimestre de embarazo. Pasado este tiempo, puede ser usado para tratar infecciones de tricomonas.

f. Los pacientes infectados deberían ser estudiados para descartar otras ITS.

La siguiente tabla nos puede orientar en el diagnóstico del flujo vaginal:

Patología	Cantidad	Color	Consistencia	Olor
Candidiasis	Escasa- moderada	Blanco- amarillento	Grumosa	Indiferente
Tricomonas	Aumentada	Amarillo- Verdoso	Espumosa	Maloliente
Vaginosis	Moderada	Blanco- Grisáceo	Homogéneo- Adherente	Maloliente

Tabla 3.6: Características del flujo vaginal según la causa

Fuente: [<http://www.fisterra.com/guias-clinicas/vulvovaginitis/>]

CAPÍTULO IV

MODELO PARA EL DIAGNÓSTICO DIFERENCIAL MEDIANTE EL CUADRO CLÍNICO

4.1 INTRODUCIENDO LOS SISTEMAS EXPERTOS BASADOS EN PROBABILIDAD

En los sistemas expertos probabilísticos las relaciones entre las variables se describen mediante su función de probabilidad conjunta. La función de probabilidad conjunta forma parte de lo que se llama conocimiento. El diagnóstico médico es una de las áreas en las que los sistemas expertos han encontrado mayor número de aplicaciones y algunos modelos de sistemas expertos probabilísticos fueron desarrollados para resolver problemas con la estructura "síntomas-enfermedad".

La Base de Conocimiento

La base de conocimiento de un sistema experto probabilístico consiste en un conjunto de variables, $\{X_1, \dots, X_n\}$, y una función de probabilidad conjunta definida sobre ellas, $p(X_1, \dots, X_n)$.

El modelo más general posible se basa en especificar directamente la función de probabilidad conjunta; es decir un valor numérico (parámetro) a cada una de las posibles combinaciones de valores de las variables. Sin embargo, la especificación directa de la función de probabilidad conjunta implica un gran número de parámetros. Así, con n variables binarias, la función de probabilidad conjunta más general tiene 2^n parámetros (las probabilidades $p_i(x_1, \dots, x_n)$, para toda posible realización $\{x_1, \dots, x_n\}$ de las variables), un número tan grande que no hay ordenador capaz de almacenarlo incluso para un valor de n tan pequeño como 50. Esta fue una de las primeras críticas al uso de la probabilidad en los sistemas expertos. Sin embargo, en la práctica, muchos subconjuntos de variables pueden ser independientes o condicionalmente independientes. Luego, podemos obtener simplificaciones del modelo más general teniendo en cuenta la estructura de independencia de las variables. Esto suele dar lugar a una reducción importante del número de parámetros. Revisaremos los siguientes modelos:

1. El Modelo de Síntomas Dependientes (MSD)
2. El Modelo de Síntomas Independientes (MSI)
3. El Modelo de Síntomas Relevantes Independientes (MSRI)
4. El Modelo de Síntomas Relevantes Dependientes (MSRD)

Estos cuatro modelos son modelos *ad hoc* que se aplican principalmente en el campo médico. Existen modelos probabilísticos más generales y potentes como los modelos de redes de Markov, modelos de redes Bayesianas y modelos especificados condicionalmente que pueden utilizarse en el campo médico y también en otros campos.

Para estos modelos se considera el problema del diagnóstico médico en la que se tenían n síntomas S_1, \dots, S_n y una variable aleatoria E , que podía tomar uno de m valores posibles e_1, \dots, e_m . En este problema se desea diagnosticar la presencia de

una enfermedad dado un conjunto de síntomas s_1, \dots, s_n . Para ello se tiene la función de probabilidad conjunta de la enfermedad y los síntomas $p(e, s_1, \dots, s_n)$. Para reducir el número de parámetros, en los modelos presentados se pueden imponer algunas hipótesis (restricciones) entre los parámetros. En todos ellos, se supone que las enfermedades son independientes de los síntomas.

4.1.1 EL MODELO DE SÍNTOMAS DEPENDIENTES

En este modelo se supone que los síntomas son dependientes pero que las enfermedades son independientes entre sí, dados los síntomas. En el MSD todo síntoma se conecta con los demás síntomas y con todo valor posible de E (indicando dependencia).

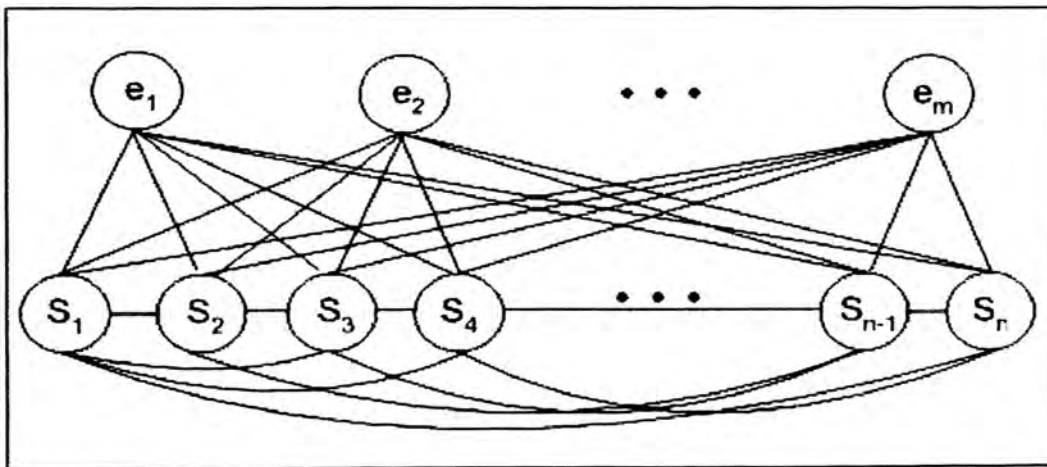


Figura 4.1: Una ilustración gráfica del modelo de síntomas dependientes

Fuente: [Castillo, 1997. p.94]

Entonces la función de probabilidad conjunta para el MSD puede escribirse como:

$$p(e, s_1, \dots, s_n) = p(s_1, \dots, s_n) p(e_i | s_1, \dots, s_n)$$

luego,

$$\begin{aligned}
 p(e_i | s_1, \dots, s_n) &= \frac{p(e_i, s_1, \dots, s_n)}{p(s_1, \dots, s_n)} \\
 &= \frac{p(e_i) p(s_1, \dots, s_n | e_i)}{p(s_1, \dots, s_n)} \\
 &\propto p(e_i) p(s_1, \dots, s_n | e_i)
 \end{aligned}$$

La proporcionalidad se sigue de que $p(s_1, \dots, s_n)$ es una constante de normalización.

Esta ecuación sólo incluye probabilidades “a priori” y verosimilitudes (probabilidades condicionales de los síntomas para cada una de las enfermedades) cuyos valores pueden estimarse a partir de la información objetiva dada por las frecuencias de enfermedades y síntomas en la población. Los parámetros necesarios para la base de datos del MSD son:

- Las probabilidades marginales $p(e_i)$, para todos los valores posibles de E .
- Las verosimilitudes $p(s_1, \dots, s_n | e_i)$, para todas las combinaciones posibles de síntomas y enfermedades.

Así, para m enfermedades y n síntomas binarios, la función de probabilidad marginal de E , $p(e_i)$, depende de $m - 1$ parámetros. Por ello, se necesita especificar $m - 1$ parámetros para la función de probabilidad marginal de E . Con respecto a las verosimilitudes $p(s_1, \dots, s_n | e_i)$, se necesita especificar $(2^n - 1)$ parámetros⁶ para cada valor posible de E , un total de $m(2^n - 1)$ parámetros.

Por ello, el MSD requiere un total de $m - 1 + m(2^n - 1) = m2^n - 1$ parámetros.

El principal problema del MSD es que requiere un número muy alto de parámetros.

⁶ Observe que para n síntomas binarios hay 2^n parámetros (un parámetro para cada combinación posible de síntomas). Sin embargo, estos parámetros deben sumar uno; en consecuencia, se tienen sólo $2^n - 1$ parámetros libres para cada valor posible de E .

Especificar las frecuencias para todas esas combinaciones es muy difícil y se hace imposible al crecer los números de las enfermedades y los síntomas. Con 100 enfermedades y 200 síntomas el número de frecuencias(parámetros) necesarios es mayor que 10^{42} .

Las dificultades se incrementan notablemente en el MSD en casos en los que se tengan síntomas con múltiples (más de dos) opciones o niveles, tales como fiebre alta, fiebre media, fiebre baja y no fiebre.

4.1.2 EL MODELO DE SÍNTOMAS INDEPENDIENTES

Una simplificación del modelo anterior consiste en suponer que, para una enfermedad dada, los síntomas son condicionalmente independientes entre sí. El modelo resultante se denomina *modelo de síntomas independientes (MSI)*.

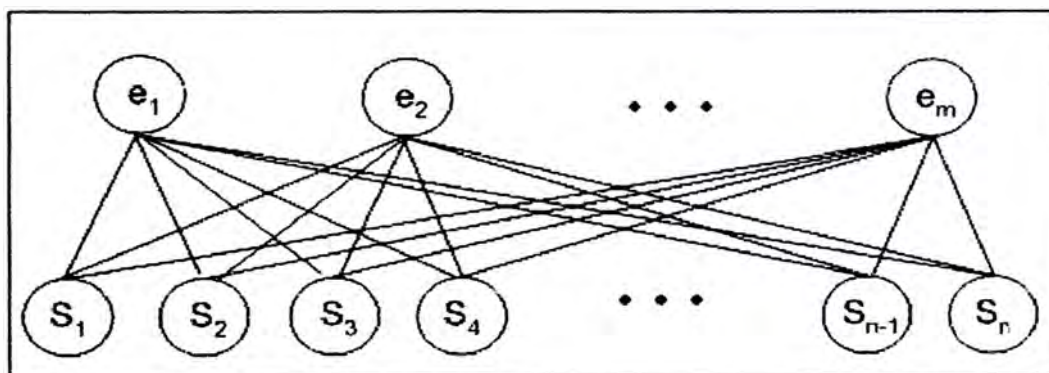


Figura 4.2: Una ilustración gráfica del modelo de síntomas independientes

Fuente: [Castillo, 1997. p.97]

Puesto que los síntomas se suponen condicionalmente independientes dada la enfermedad, se tiene

$$p(s_1, \dots, s_n | e_i) = \prod_{j=1}^n p(s_j | e_i)$$

Luego, se puede escribir la función de probabilidad conjunta de la enfermedad E dados los síntomas s_1, \dots, s_n como

$$\begin{aligned} p(e_i | s_1, \dots, s_n) &= \frac{p(e_i) p(s_1, \dots, s_n | e_i)}{p(s_1, \dots, s_n)} \\ &= \frac{p(e_i) \prod_{j=1}^n p(s_j | e_i)}{p(s_1, \dots, s_n)} \\ &\propto p(e_i) \prod_{j=1}^n p(s_j | e_i) \end{aligned}$$

La ecuación muestra cómo la hipótesis de independencia modifica las probabilidades de todas las enfermedades cuando se conocen nuevos síntomas. La probabilidad inicial de la enfermedad e_i es $p(e_i)$, pero tras conocer los síntomas s_j , para $j = 1, \dots, k$, resulta proporcional a $p(s_j | e_i)$. Cada nuevo síntoma conduce a un nuevo factor.

Los parámetros necesarios para la base de conocimiento del MSI son

- Las probabilidades marginales $p(e_i)$, para todos los valores posibles de la enfermedad E .
- Las probabilidades condicionales $p(s_j | e_i)$, para todos los valores posibles del síntoma s_j y la enfermedad E .

En el MSI, para m enfermedades posibles y n síntomas binarios, el número total de parámetros es $m(n + 1) - 1$. Por ejemplo, con $m = 100$ enfermedades y $n = 200$ síntomas se tienen 20099 parámetros en el MSI en vez de más de 10^{62} parámetros para el MSD.

El correcto comportamiento de un sistema experto probabilístico se basa en la especificación correcta de la función de probabilidad conjunta.

Aunque la hipótesis de independencia da lugar a una gran reducción del número de parámetros, el número de parámetros en el MSI es todavía muy alto para ser práctico.

4.1.3 EL MODELO DE SÍNTOMAS RELEVANTES INDEPENDIENTES

Para reducir aún más el número de parámetros se puede suponer que cada enfermedad tiene un número reducido de síntomas relevantes. Para cada valor e_i de la enfermedad E se seleccionan algunos síntomas relevantes s_1, \dots, s_r (relativamente pocos frente al total de síntomas) y los restantes síntomas se suponen independientes para ese valor de E .

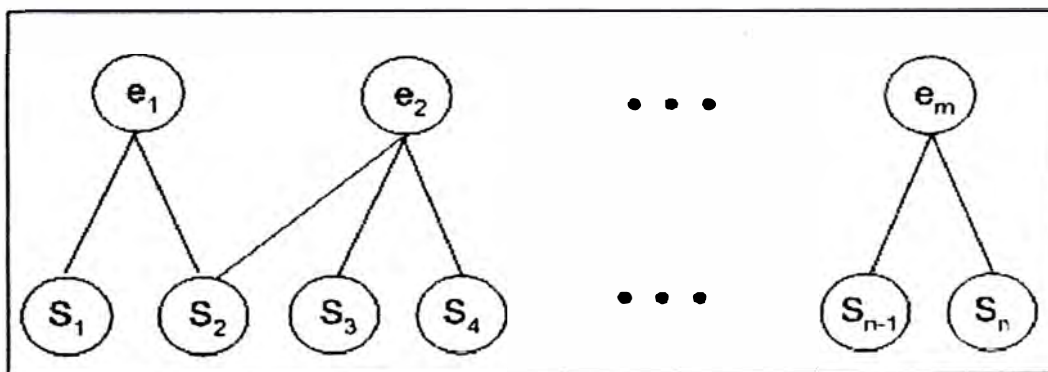


Figura 4.3: Una ilustración gráfica del modelo de síntomas relevantes independientes.

Fuente: [Castillo, 1997. p.100]

Supóngase que s_1, \dots, s_r son relevantes para la enfermedad e_i y que los restantes síntomas s_{r+1}, \dots, s_n son irrelevantes. Según el MSRI, $p(s_j | e_i)$ se supone idéntica para todos los síntomas que son irrelevantes para la enfermedad e_i . Entonces su función de probabilidad conjunta estará dada por

$$\begin{aligned}
p(e_i | s_1, \dots, s_n) &= \frac{p(e_i) p(s_1, \dots, s_n | e_i)}{p(s_1, \dots, s_n)} \\
&= \frac{p(e_i) \prod_{j=1}^{r_i} p(s_j | e_i) \prod_{j=r_i+1}^n p(s_j | e_i)}{p(s_1, \dots, s_n)} \\
&= \frac{p(e_i) \prod_{j=1}^{r_i} p(s_j | e_i) \prod_{j=r_i+1}^n p_j}{p(s_1, \dots, s_n)} \\
&\propto p(e_i) \prod_{j=1}^{r_i} p(s_j | e_i) \prod_{j=r_i+1}^n p_j
\end{aligned}$$

Donde $p_j = p(s_j | e_i)$, que es la misma para todas las enfermedades para la que s_j es irrelevante. Luego, es necesario almacenar las probabilidades siguientes en la base de conocimiento del MSRI:

- Las probabilidades marginales $p(e_i)$, para todos los valores posibles de la enfermedad E.
- Las probabilidades condicionales $p(s_j | e_i)$, para cada valor posible de E y cada uno de sus correspondientes síntomas relevantes
- Las probabilidades p_j , para cada valor posible de E que tiene al menos un síntoma irrelevante.

Si se tienen m posibles enfermedades y n síntomas binarios, el número de parámetros en el MSRI es

$$m - 1 - n - a - \sum_{i=1}^m r_i$$

donde r_i es el número de síntomas relevantes para la enfermedad e_i y a es el número de síntomas que son relevantes para todas las enfermedades.

El número de parámetros se reduce significativamente cuando r_i es mucho menor que n . Para 100 enfermedades y 200 síntomas, si $r_i = 10$ para todas las enfermedades, el número de parámetros se reduce de 20099 para el MSI a 1299 para el MSRI.

4.1.4 EL MODELO DE SÍNTOMAS RELEVANTES DEPENDIENTES

Si bien el MSRI reduce considerablemente el número de parámetros, es poco realista, ya que los síntomas asociados a ciertas enfermedades suelen producirse en grupos o síndromes. Por ello, puede ser poco razonable suponer que los síntomas relevantes son independientes. El *modelo de síntomas relevantes dependientes* (MSRD) es el mismo que el MSRI pero sin obligar a los síntomas relevantes a ser independientes, dada la correspondiente enfermedad. Se supone que sólo los síntomas irrelevantes son independientes pero los síntomas relevantes pueden ser dependientes.

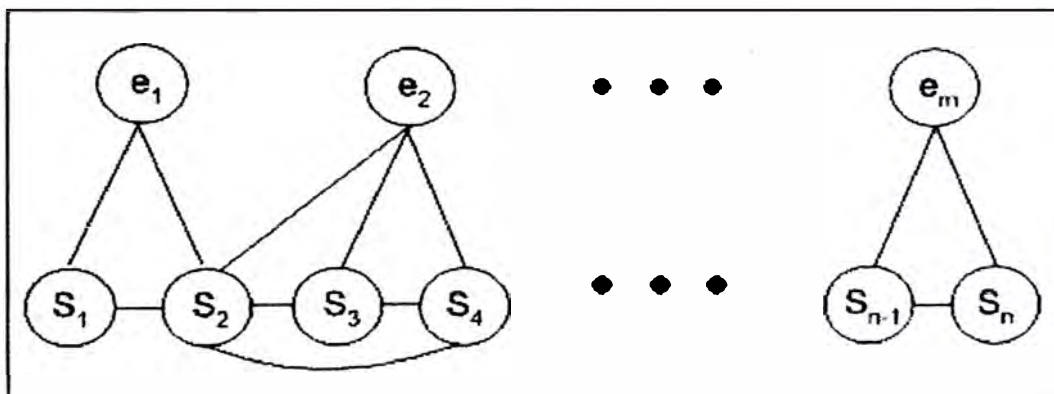


Figura 4.4: Una ilustración gráfica del modelo de síntomas relevantes dependientes

Fuente: [Castillo, 1997. p.102]

Supóngase que s_1, \dots, s_{r_i} son relevantes para la enfermedad e_i y que los restantes síntomas s_{r_i+1}, \dots, s_n son irrelevantes. Según el MSRD, la función de probabilidad conjunta puede escribirse como

$$\begin{aligned}
p(e_i | s_1, \dots, s_n) &= \frac{p(e_i) p(s_1, \dots, s_{r_i} | e_i) \prod_{j=r_i+1}^n p(s_j | e_i)}{p(s_1, \dots, s_n)} \\
&= \frac{p(e_i) p(s_1, \dots, s_{r_i} | e_i) \prod_{j=r_i+1}^n p_j}{p(s_1, \dots, s_n)} \\
&\propto p(e_i) p(s_1, \dots, s_{r_i} | e_i) \prod_{j=r_i+1}^n p_j
\end{aligned}$$

donde $p_j = p(s_j | e_i)$, que es la misma para todas las enfermedades para las que s_j es irrelevante. Para este modelo, es necesario almacenar las probabilidades siguientes en la base de conocimiento:

- Las probabilidades marginales $p(e_i)$, para todos los valores posibles de la enfermedad E.
- Las probabilidades condicionales $p(s_1, \dots, s_{r_i} | e_i)$, para cada valor posible de E y sus síntomas relevantes s_1, \dots, s_{r_i} .
- Las probabilidades p_j , para cada valor posible de E que tenga al menos un síntoma irrelevante.

Luego, si se tienen m posibles enfermedades y n síntomas binarios, el número de parámetros en el MSRD es

$$m - 1 - n - a - \sum_{i=1}^m (2^{r_i} - 1) = n - 1 - a - \sum_{i=1}^m 2^{r_i}$$

Nótese que cuando $r_i = r$ para todos los valores e_i , entonces la ecuación resulta $m2^r - n - 1 - a$.

Para el desarrollo del presente prototipo utilizaremos el modelo de síntomas relevantes dependientes.

CAPÍTULO V

METODOLOGÍAS PARA DESARROLLAR UN SE

5.1 METODOLOGÍAS PARA DESARROLLAR UN SE

Entendemos por metodologías a aquellas herramientas utilizadas por el ingeniero de conocimiento, que le dan pautas sobre cómo desarrollar un Sistema Experto, guían la construcción del SE permitiendo una correcta documentación y ayudan a la detección de problemas durante el desarrollo para poder corregirlos a tiempo, evitando así posibles errores.

Lamentablemente antes de los inicios de la década de los 80, desde el punto de vista metodológico poco o nada se desarrolló. Recién en 1983 Hayes-Roth, Waterman realizan **Building Expert Systems**, que es la primera publicación importante que recoge la dimensión metodológica del desarrollo de SE. Posteriormente siguieron Weiss y Kulikowski en 1984 con **A practical guide to designing expert systems**, Domingo Carrillo y Juan Pazos en 1987 con la tesis doctoral **Metodología para el desarrollo de sistemas expertos** y muchos otros.

Sistema Experto

Experto Humano

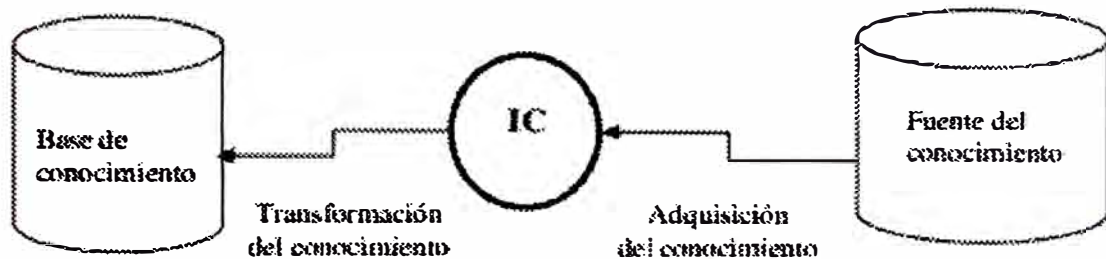


Figura 5.1: Proceso aplicado por la Ingeniería del Conocimiento.

Fuente: [Romero y Rodríguez, 2012. p.7]

Todos estos investigadores afrontan el desarrollo de los SE mediante modelos iterativos, en los que se parte de un prototipo sencillo que incrementalmente se irá expandiendo y mejorando. También algunos autores afrontan la adquisición del conocimiento como parte de un proceso estructurado. Para tal fin se tomaron técnicas de adquisición del conocimiento del campo de la psicología y se adaptaron a las características de estos tipos de sistemas, potenciando de esta forma la transferencia del conocimiento desde el experto humano a la computadora.

5.1.1 METODOLOGÍA DE WEISS Y KULIKOWSKI

Weiss y Kulikowski sugieren las siguientes etapas para el diseño e implementación de un sistema experto:

- 1. Planteamiento del problema.** La primera etapa en cualquier proyecto es normalmente la definición del problema a resolver. Puesto que el objetivo principal de un sistema experto es responder a preguntas y resolver problemas, esta etapa es quizás la más importante en el desarrollo de un sistema experto.

Si el sistema está mal definido, se espera que el sistema suministre respuestas erróneas.

- 2. Encontrar expertos humanos que puedan resolver el problema.** En algunos casos, sin embargo, las bases de datos pueden jugar el papel del experto humano.
- 3. Diseño de un sistema experto.** Esta etapa incluye el diseño de estructuras para almacenar el conocimiento, el motor de inferencia, el subsistema de explicación, la interfaz de usuario, etc.
- 4. Elección de la herramienta de desarrollo, concha, o lenguaje de programación.** Debe decidirse si realizar un sistema experto a medida, o utilizar una Shell, una herramienta, o un lenguaje de programación. Si existiera una concha satisfaciendo todos los requerimientos del diseño, ésta debería ser la elección, no sólo por razones de tipo financiero sino también por razones de fiabilidad. Las Shells y herramientas comerciales están sujetas a controles de calidad, a los que otros programas no lo están.
- 5. Desarrollo y prueba de un prototipo.** Si el prototipo no pasa las pruebas requeridas, las etapas anteriores (con las modificaciones apropiadas) deben ser repetidas hasta que se obtenga un prototipo satisfactorio.
- 6. Refinamiento y generalización.** En esta etapa se corrigen los fallos y se incluyen nuevas posibilidades no incorporadas en el diseño inicial.

7. Mantenimiento y puesta al día. En esta etapa el usuario plantea problemas o defectos del prototipo, corrige errores, actualiza el producto con nuevos avances, etc.

Todas estas etapas influyen en la calidad del sistema experto resultante, que siempre debe ser evaluado en función de los aportes que den los usuarios.

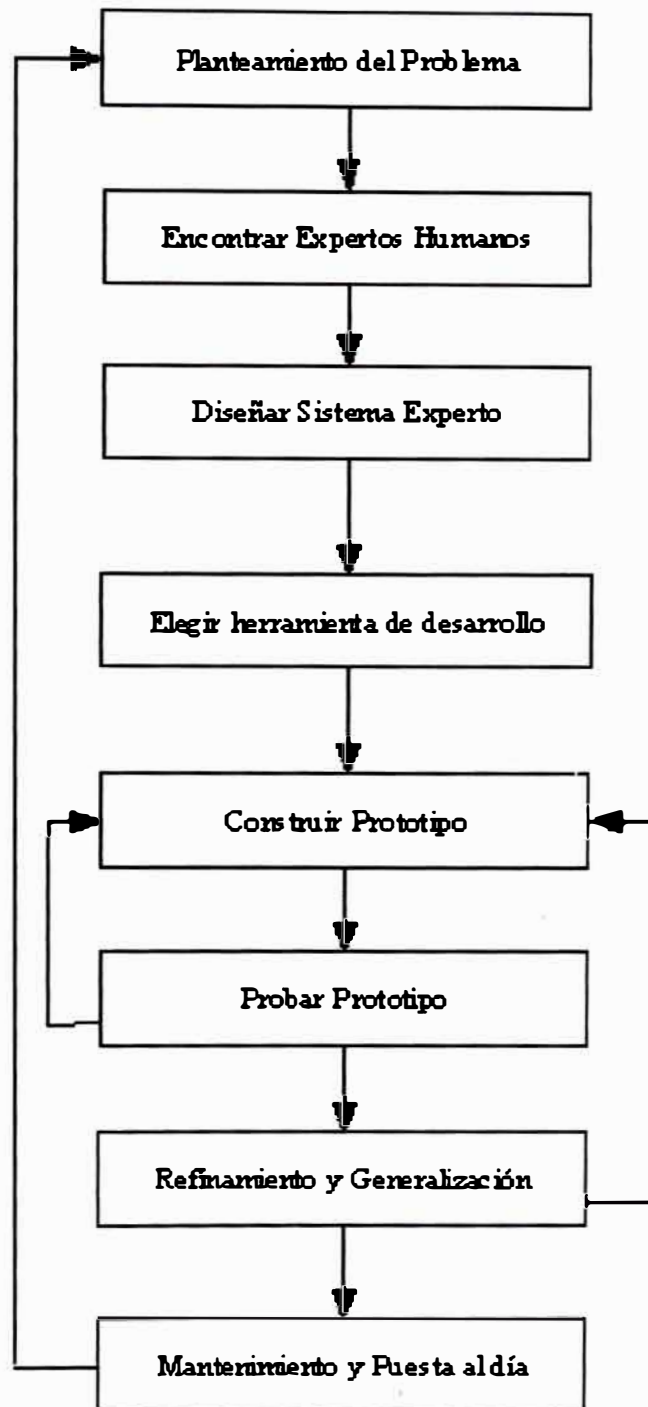


Figura 5.2 Etapas en el desarrollo de un Sistema Experto.

Fuente: [Castillo, 1997. p.15]

5.1.2 METODOLOGÍA IDEAL

La Metodología IDEAL elaborada por Juan Pazos y Domingo Carrillo [CAR1987] propone un ciclo de vida en espiral en tres dimensiones, y se ajusta a la tendencia del software actual, esto es:

- Ser Reutilizable.
- Ser Integrable.
- Poseer Requisitos Abiertos.
- Diversidad de Modelos Computacionales.

Los requisitos están sometidos a constantes cambios y por ende el sistema también, por lo que como resultado se obtiene un sistema en constante evolución de manera que puede considerarse como un prototipo en constante perfeccionamiento, mediante el agregado de nuevos marcos compuestos, mediante nuevas técnicas de descomposición del problema, mediante nuevas formas de documentación o estándares a los que debe ajustarse.

El siguiente cuadro presenta las fases y etapas que componen la metodología I.D.E.A.L. y que guían el desarrollo de un sistema experto:

FASES Y ETAPAS DE LA METODOLOGIA I.D.E.A.L.	
FASE I: Requerimientos, viabilidad, especificación técnica.	
Etapa I.1. Plan de requisitos y adquisición de conocimientos.	
Etapa I.2. Evaluación y selección de la tarea.	
Etapa I.3 Definición de las características del sistema.	
FASE II: Desarrollo de los prototipos de demostración, investigación, campo y operacional	
Etapa II.1. Concepción de la solución.	
Etapa II.2. Adquisición y Conceptualización de los conocimientos.	
Etapa II.3. Formalización de los conocimientos y definición de la arquitectura	
Etapa II.4. Selección de la herramienta e implementación.	
Etapa II.5. Validación y evaluación del prototipo.	
Etapa II.6. Definición de nuevos requisitos, especificaciones y diseño.	
<i>Nota: Las etapas II.1 a II.6 se repiten por cada iteración del prototipo.</i>	
FASE III: Ejecución de la construcción del sistema integrado	
Etapa III.1 Requisitos y diseño de la integración	
Etapa III.2 Implementación y evaluación del sistema integrado	
Etapa III.3 Aceptación del sistema por el cliente	
FASE IV: Actuación para conseguir el mantenimiento perfecto	
Etapa IV.1. Definir el mantenimiento del sistema global.	
Etapa IV.2. Definir el mantenimiento de las bases de conocimientos.	
Etapa IV.3. Adquisición de nuevos conocimientos.	
FASE V: Lograr una adecuada transferencia tecnológica	
Etapa V.1. Organizar la transferencia tecnológica.	
Etapa V.2. Completar la documentación del SBC construido.	

Tabla 5.1: Fases y Etapas de la metodología IDEAL

Fuente: [Rizzi, 2001. p.30]

A continuación se describe brevemente las diferentes etapas del ciclo de vida de un SE:

Estudio de viabilidad:

Al intentar resolver un problema con la tecnología de SE, debemos evaluar previamente si la tarea se puede abordar en el campo de la Ingeniería del

Conocimiento. Debe definirse si el desarrollo es plausible, justificable y si se puede garantizar su éxito.

Adquisición de conocimiento:

Los problemas abordados con la tecnología de la Ingeniería del Conocimiento intentan imitar a través de software la pericia de un experto humano al desempeñar una tarea específica. Una de las actividades que requiere mayor esfuerzo por su complejidad es la extracción de conocimientos, en la cual se busca descubrir el dominio de la aplicación, el problema y el proceso de solución al problema.

Conceptualización:

En esta fase se describe el proceso de organización de los conocimientos adquiridos. Esta actividad está conformada por dos tareas fundamentales: una de análisis basada en la detección de conocimientos estratégicos, tácticos y la actividad de síntesis en la cual dichos conocimientos se expresan en forma estructurada.

Formalización:

En esta etapa se busca encontrar la representación de conocimiento más adecuada y garantizar su correcta manipulación. Es el primer acercamiento a la máquina para su posterior implementación.

Implementación:

Es la fase en la que se transforman los conocimientos representados en el modelo formal al modelo computacional.

Evaluación:

Establece el grado de experiencia alcanzado por el sistema. En esta etapa los expertos que han participado o no en el desarrollo del proyecto se comprometen a evaluar el desempeño del sistema tratando de determinar la calidad de asistencia

que brinda el Sistema Experto ante diferentes casos a resolver por el software representativo.

5.2 DESARROLLO DE LA METODOLOGÍA IDEAL

5.2.1 FASE I: REQUERIMIENTOS, VIABILIDAD Y ESPECIFICACIÓN TÉCNICA

5.2.1.1 ESTUDIO DE VIABILIDAD

El estudio de viabilidad nos permite asegurar que el problema planteado puede ser resuelto mediante un Sistema Experto. Detallaremos la siguiente etapa del test de viabilidad:

5.2.1.1.1 DEFINICIÓN DE LAS CARACTERÍSTICAS

Se consideran cuatro dimensiones, que se describen a continuación:

5.2.1.1.1.1 PLAUSIBILIDAD:

Se debe garantizar que existan verdaderos expertos en el área del problema. Estos expertos deberían estar totalmente comprometidos para trabajar en el proyecto y se debe buscar que el experto sea cooperativo y capaz de articular sus conocimientos y modos de razonamiento. Asimismo sería deseable disponer de un conjunto de casos de prueba que permita entender como los expertos resuelven el problema.

Para el desarrollo del proyecto se contó con el apoyo del siguiente staff de médicos:

- Dr. Ronald Cabrera, con CMP 22565, médico especialista en Ginecología y Obstetricia, que tiene amplia experiencia en Hospitales y Clínicas.
- Dra. Inés Sovero, médico especialista en Ginecología.
- Dr. Eddy Angles, Jefe de Infectología.

5.2.1.1.1.2 JUSTIFICACIÓN:

En esta característica se busca justificar el desarrollo del sistema, desde el punto de vista de la Ingeniería del Conocimiento, por ejemplo cuando la tarea del experto debe realizarse en entornos hostiles o peligrosos, por lo que no se desea mantener un experto humano en el lugar, o bien cuando los expertos humanos escasean y una organización necesita expertos en distintas ubicaciones a la vez.

En el caso de este proyecto, la organización que es el MINSA no se abastece para destinar a todos los especialistas que quisiera a lo largo de todo el territorio nacional y se da el caso que en los pueblos, los centros de salud o postas médicas están a cargo de un técnico en enfermería que debe atender adicionalmente a un promedio de cuatro o cinco anexos que pertenecen a la periferia del pueblo al que ha sido designado y necesita estar rotando entre todos esos centros poblados, por lo que en muchas ocasiones es difícil ubicarlo, ya que puede estar de gira.

5.2.1.1.1.3 ÉXITO:

Esta característica nos ayuda a determinar las probabilidades a priori de éxito que tiene el sistema que se pretende desarrollar. Es de suma importancia que el sistema sea bien acogido por los trabajadores pues un rechazo del sistema, ya sea por motivos de inseguridad a la hora de conservar el puesto o la falta de confianza en la veracidad de las soluciones puede provocar el fracaso del proyecto y si esto ocurriera se produciría una gran pérdida de recursos y tiempo.

En el caso del proyecto los especialistas colaboradores están suficientemente entrenados, pues se encuentran en capacitación continua y en constante práctica y además están comprometidos en lograr la solución. El prototipo de sistema experto se ubicará en una posición estratégica para que los usuarios, que pueden ser enfermeras, técnicas o médicos residentes comprueben que es una herramienta que mejorará su calidad laboral.

5.2.1.2 ADQUISICIÓN DE CONOCIMIENTOS

Como se sabe, la adquisición de conocimientos se refiere a la obtención de conocimiento de diversas fuentes: libros, documentos y de expertos humanos.

Para la adquisición de conocimientos en el proyecto se realizó una serie de entrevistas a diversos especialistas en ginecología e infectología. Ver el Anexo A: *Extracto de una entrevista a un especialista*. En las primeras reuniones con los distintos médicos es común que exista cierto grado de desconfianza al no tener muy claro cuál es el horizonte. El término que les resulta más familiar al profesional de salud para referirse a un software que realice el diagnóstico a partir del cuadro clínico es el diagnosticador.

Aquí fue muy importante seleccionar la enfermedad de estudio, pues una de las observaciones de los médicos se refería al hecho que hay muchas patologías en las que los pacientes no presentan sintomatología, son asintomáticos, de modo que éstas tenían que ser descartadas pues no se ajustaban a nuestra hipótesis de trabajo. Al respecto, un libro que tiene este enfoque que fue proporcionado gentilmente por los médicos es “El diagnóstico a través de la historia clínica” [KRA1986] y que puede ser de utilidad para elegir un dominio del problema.

La conclusión de tales reuniones de presentación fue el definir como dominio de problema el síndrome de flujo vaginal que consta de un número limitado de enfermedades que se ajustaban a lo que se requería, en algunos casos los mismos médicos proporcionaron libros especializados o en todo caso recomendaban artículos para poder conocer en detalle el dominio del problema y su proceso de solución.

En reuniones posteriores, las entrevistas eran más focalizadas y se concentraban en describir la etiología, diagnóstico y los hallazgos que caracterizaban a cada una de las enfermedades seleccionadas.

5.2.2 FASE II: DESARROLLO DE LOS PROTOTIPOS DE DEMOSTRACIÓN, INVESTIGACIÓN, CAMPO Y OPERACIONAL

5.2.2.1 ETAPA 2.1 CONCEPCIÓN DE LA SOLUCIÓN

En esta etapa en mi calidad de ingeniero de conocimiento y con colaboración con los médicos delineamos las especificaciones del prototipo, cuyo diseño general debería contar mínimamente con:

- Un módulo para el manejo de enfermedades.
- Un módulo para el manejo de síntomas.
- Un módulos para las asociaciones de síntomas a enfermedades.

Estos representan las entradas al sistema y adicionalmente el módulo central para el diagnóstico de enfermedades donde actúa el motor de inferencia.

5.2.2.2 ETAPA 2.2 CONCEPTUALIZACIÓN DE CONOCIMIENTOS

Luego de la etapa de entrevistas e inmersión en la literatura especializada así como normas técnicas, se pasó al proceso de organización de los conocimientos adquiridos. Aquí se tuvo que hacer uso de nuestra capacidad de síntesis para expresar todos los conocimientos en una forma estructurada.

La clasificación de los componentes para cada una de las siguientes entidades se realizó bajo supervisión y sugerencia de los médicos especialistas, en base a su experiencia profesional. A continuación describo la educación de los conocimientos privados de los médicos expertos.

ENFERMEDADES

El siguiente cuadro presenta las enfermedades que son de mayor incidencia en el síndrome del flujo vaginal, las mismas que se tomaron en cuenta para el desarrollo del prototipo:

Número	Enfermedades	Abreviaturas
1	Candidiasis	cand
2	Tricomoniasis	tric
3	Vaginosis Bacteriana	vbac

Tabla 5.2: Relación de patologías para el flujo vaginal

Fuente: [Elaboración Propia]

SIGNOS

La siguiente tabla, presenta los signos que han sido estandarizados luego de las entrevistas sostenidas con los médicos y de la revisión de literatura recomendada.

Número	Signos
1	FLUJO VAGINAL ESCASO
2	FLUJO VAGINAL AUMENTADO
3	FLUJO VAGINAL MODERADO
4	FLUJO VAGINAL BLANCO-AMARILLENTO
5	FLUJO VAGINAL AMARILLO - VERDOSO
6	FLUJO VAGINAL BLANCO - GRISACEO
7	FLUJO VAGINAL GRUMOSO
8	FLUJO VAGINAL ESPUMOSO
9	FLUJO VAGINAL HOMOGENEEO ADHERENTE
10	FLUJO VAGINAL INDIFERENTE
11	FLUJO VAGINAL MALOLIENTE
12	CERVIX DE FRESA

Tabla 5.3: Relación de signos para el flujo vaginal

Fuente: [Elaboración Propia]

SINTOMAS

La siguiente tabla, presenta los síntomas que han sido estandarizados luego de las entrevistas sostenidas con los médicos y de la revisión de literatura recomendada.

Número	Síntomas
1	PRURITO

Tabla 5.4: Relación de síntomas para el flujo vaginal

Fuente: [Elaboración Propia]

PRUEBAS DE LABORATORIO

La siguiente tabla, presenta las pruebas de laboratorio que han sido estandarizadas luego de las entrevistas sostenidas con los médicos y de la revisión de literatura recomendada.

Número	Prueba de Laboratorio
1	NIVEL DE PH: 4.0 - 4.5
2	NIVEL DE PH: 4.5 - 5.5
3	NIVEL DE PH: 5.0 - 7.0
4	TEST DE AMINAS

Tabla 5.5: Relación de pruebas de laboratorio para el flujo vaginal

Fuente: [Elaboración Propia]

PROBABILIDADES A PRIORI PARA CADA ENFERMEDAD

Según estudios de revistas especializadas se tiene la siguiente proporción de casos para las patologías del flujo vaginal.

Número	Enfermedades	Probabilidad a priori
1	Candidiasis	0.25
2	Tricomoniasis	0.10
3	Vaginosis Bacteriana	0.55

Tabla 5.6: Enfermedades junto con sus probabilidades a priori

Fuente: [Elaboración Propia]

VEROSIMILITUDES PARA DIAGNOSTICAR CANDIDIASIS

En la siguiente tabla se presentan los síntomas, signos y pruebas de laboratorio que se asocian a esta enfermedad, así como la verosimilitud que le asignó el experto humano a cada uno de ellos.

Enfermedad	Candidiasis	
Número	Signos, Síntomas y Pruebas	Verosimilitud
1	FLUJO VAGINAL MODERADO	0.75
2	FLUJO VAGINAL BLANCO-AMARILLENTO	0.75
3	FLUJO VAGINAL GRUMOSO	0.75
4	FLUJO VAGINAL INDIFERENTE	0.75
5	NIVEL DE PH: 4.0 - 4.5	1.00
6	PRURITO	0.50

Tabla 5.7: Verosimilitud de signos, síntomas y pruebas para la Candidiasis

Fuente: [Elaboración Propia]

VEROSIMILITUDES PARA DIAGNOSTICAR TRICOMONIASIS

En la siguiente tabla se presentan los síntomas, signos y pruebas de laboratorio que se asocian a esta enfermedad, así como la verosimilitud que le asignó el experto humano a cada uno de ellos.

Enfermedad	Tricomoniasis	
Número	Signos, Síntomas y Pruebas	Verosimilitud
1	FLUJO VAGINAL AUMENTADO	0.75
2	FLUJO VAGINAL AMARILLO – VERDOSO	0.75
3	FLUJO VAGINAL ESPUMOSO	0.75
4	FLUJO VAGINAL MALOLIENTE	0.75
5	CERVIX DE FRESA	0.75
6	NIVEL DE PH: 5.0 - 7.0	1.00
7	PRURITO	0.75

Tabla 5.8: Verosimilitud de signos, síntomas y pruebas para la Tricomoniasis

Fuente: [Elaboración Propia]

VEROSIMILITUDES PARA DIAGNOSTICAR VAGINOSIS BACTERIANA

En la siguiente tabla se presentan los síntomas, signos y pruebas de laboratorio que se asocian a esta enfermedad, así como la verosimilitud que le asignó el experto humano a cada uno de ellos.

Enfermedad	Vaginosis Bacteriana	
Número	Signos, Síntomas y Pruebas	Verosimilitud
1	FLUJO VAGINAL ESCASO	0.75
2	FLUJO VAGINAL BLANCO - GRISACEO	0.75
3	FLUJO VAGINAL HOMOGENEO ADHERENTE	0.75
4	FLUJO VAGINAL MALOLIENTE	0.75
5	NIVEL DE PH: 4.5 - 5.5	1.00
6	TEST DE AMINAS	0.90

Tabla 5.9: Verosimilitud de signos, síntomas y pruebas para la Vaginosis Bacteriana

Fuente: [Elaboración Propia]

5.2.2.3 ETAPA 2.3 FORMALIZACIÓN DE LOS CONOCIMIENTOS

Se concluyó que la representación del conocimiento más adecuada era la de un sistema probabilístico basado en estadísticas y el teorema de Bayes. Se manejó como alternativa realizar un sistema de producción pero cada institución de salud vive una realidad diferente y lo cierto es que por déficit de presupuesto en las instituciones públicas es difícil que se realicen todos los exámenes auxiliares que requiere el médico. Es por esto que el software debería permitir al usuario configurar la base de conocimientos de modo que se ajuste a la realidad de la institución donde se va poner en producción y esto se vuelve complejo en un sistema basado en reglas de producción.

Las aplicaciones de diagnóstico diferencial basadas en el teorema de Bayes usan como enfoque el construir una matriz que registra las probabilidades de que un signo o síntoma dado sea asociado a una patología, que puede ser resultado de un estudio estadístico en una muestra de pacientes en un área del diagnóstico médico.

5.2.2.4 ETAPA 2.4 SELECCIÓN DE LA HERRAMIENTA

Para seleccionar una herramienta, es necesario que el ingeniero de conocimiento (o el programador, si hubiera) tome conciencia de que tan profundas son sus habilidades de programación Si su experiencia en este campo no es muy amplia recomiendo que utilice una Shell, pues hay muchas en el mercado y varias de ellas son gratuitas, por ejemplo CLIPS, que es un producto que fue liberado hace varios años por la NASA y que tiene gran cantidad de documentación. Si ha participado de algún curso de Inteligencia Artificial puede optar por algún lenguaje que haya aprendido en esa materia, como es el caso del Prolog y todas sus variantes entre las que destacan SWI-Prolog, Visual Prolog, etc. Por último, si tiene experiencia en lenguajes de programación puede optar por el desarrollo propio donde deberá definir sus estructuras para representar el conocimiento y el modelo para su motor de inferencia, lo cual significa dedicarle muchas horas a la programación, es decir es muy costoso en tiempo. Aquí utilice la herramienta que más domine como puede ser C, Visual FoxPro, Delphi, Java, etc. Antes de realizar la exploración, es necesario establecer las diferencias entre Lenguajes, Herramientas y Shells.

5.2.2.4.1 LENGUAJES, HERRAMIENTAS Y SHELLS DE SE

En el mercado informático existe una gran variedad de software para IA. Algunos vendedores aluden a sus productos como "herramientas", mientras que otros

hablan de “Shells” y algunos otros de “ambientes integrados”, de modo que precisamos establecer las siguientes definiciones:

Lenguaje: Es un traductor de comandos escrito con una sintaxis específica. Un lenguaje para SE también proporcionará un mecanismo de inferencia que ejecute las instrucciones del lenguaje. El mecanismo de inferencia puede proporcionar encadenamiento hacia atrás, hacia adelante o ambos. Bajo esta definición de lenguaje, LISP no es un lenguaje para SE pero PROLOG sí [GIA1998].

Herramienta: Es un lenguaje adicionalmente asociado con programas de utilerías para facilitar el desarrollo, la depuración y el uso de los programas de aplicación. Los programas de utilerías pueden incluir editores de texto e imágenes, depuradores, administradores de archivos e incluso generadores de código. También pueden proporcionar ensambladores de plataforma cruzada para portar el código de desarrollo a un hardware diferente [GIA1998].

Shell: Las primeras Shell se crearon por la circunstancia que tras la creación de un SE, se extrajo toda la información referente al caso de aplicación en particular para el que fue escrito. Este resto recibe el nombre de Shell, es decir una caparazón para un SE [NEB1988]. Las Shell se hicieron disponibles comercialmente cerca a 1980, destacando entre las primeras AL/X y EMYCIN, ambas orientadas a reglas. El Shell EMYCIN surgió al eliminar la base del conocimiento médico del SE MYCIN [GIA1998]. Desde 1980 se desarrollan cada vez más sistemas que son concebidos

como Shells puros. Un Shell ayuda en el desarrollo de los SE pues ofrece al Ingeniero del Conocimiento métodos auxiliares como por ejemplo estructuras para la representación del conocimiento, mecanismos de inferencia, apoyo para un componente explicativo. A menudo se dan también estructuras de control que permiten influir en el procesamiento del conocimiento. Los mecanismos de inferencia y las estructuras de control no siempre pueden ser claramente separados [NEB1988].

No todos estos componentes han podido ser introducidos en los Shells disponibles actualmente. Los componentes más deseados en un Shell son:

- Formalismo para la representación del conocimiento.
- Medios de estructuración para la Base de Conocimientos.
- Mecanismo de inferencia.
- Interfaz de usuario, adaptado a los requisitos en la confección de un SE, en la estructuración y ampliación de la base de conocimientos y en el uso final.
- Apoyo en la creación de un componente explicativo.
- Mecanismos para la comprobación de consistencia, búsqueda de errores y ampliación de la base de conocimientos.
- Ayudas en la adquisición de conocimiento.

Un Shell consta por lo tanto, de partes de un SE, independientes del conocimiento. Con el uso de un Shell poderoso, el encargado del desarrollo puede concentrarse

plenamente en la confección de la Base de Conocimientos. Es por ello que la utilización de Shells reduce el esfuerzo de desarrollo de SE. Una ventaja adicional de la utilización de Shells radica en que no se requiere un profundo dominio de un lenguaje de programación [NEB1988].

5.2.2.4.1.1 Revisión de Lenguajes, Herramientas y Shells de SE

A continuación se describirá algunas Shells de SE:

CLIPS

CLIPS(C Language Integrated Production System o Lenguaje para sistemas de producción integrada) es una herramienta para desarrollar sistemas expertos producida por Software Technology Branch para la NASA/ Lyndon B. Jhonson Space Center. [San Juan A. Sistema CLIPS].

Características

- CLIPS es un entorno completo para la construcción de SE basados en reglas y/o objetos.
- Es un sistema de producción con encadenamiento hacia adelante escrito en C estándar (ANSI C).
- Tiene gran portabilidad, es posible compilarlo en varias plataformas como Unix, Linux y Windows.
- Permite integración completa con otros lenguajes como C, Java o Fortran.
- Cuenta con extensiones para Lógica Difusa (Fuzzy CLIPS).

El shell provee los elementos básicos de un SE:

1. Memoria global de datos.
2. Base de conocimiento.
3. Motor de inferencia

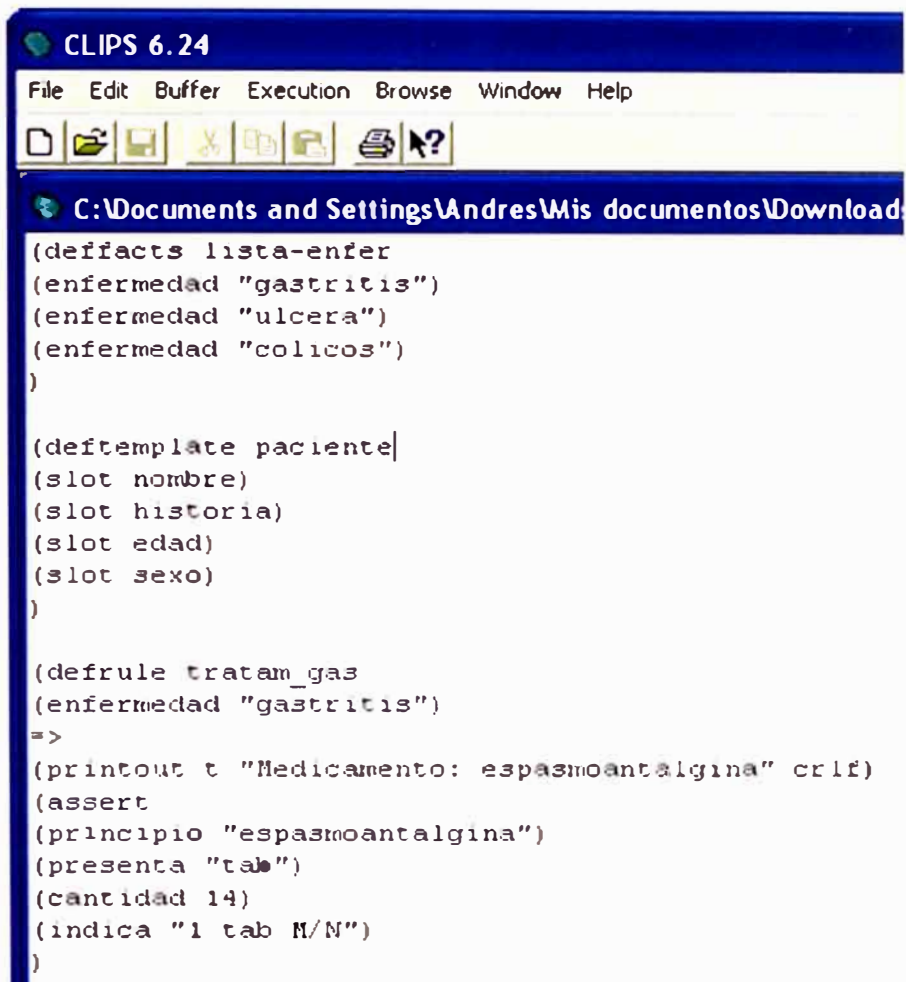
Un programa escrito en CLIPS puede consistir en reglas, hechos y objetos.

Ejemplo de definición de hechos

```
(deffacts personas_empadronadas
  (persona (nombre "Hermes") (edad 36))
  (persona (nombre "Moises") (edad 17))
  (persona (nombre "Marcela") (edad 25))
  (persona (nombre "Fernando") (edad 38))
)
```

Ejemplo de definición de reglas

```
(defrule censar
  (persona (nombre ?n) (edad ?e))
  (test (> ?e 17)) =>
  (assert (elector (nombre ?n)))
```



```
(defacts lista-enfer
(enfermedad "gastritis")
(enfermedad "ulcera")
(enfermedad "colicos")
)

(deftemplate paciente
(slot nombre)
(slot historia)
(slot edad)
(slot sexo)
)

(defrule tratam_gas
(enfermedad "gastritis")
=>
(printout t "Medicamento: espasmoantalgina" crlf)
(assert
(principio "espasmoantalgina")
(presenta "tab")
(cantidad 14)
(indica "1 tab M/N")
)
```

Figura 5.3: Implementación de hechos, plantillas y reglas en CLIPS

Fuente: [Elaboración Propia]

Para mayor información podemos acceder al siguiente enlace:

<http://clipsrules.sourceforge.net/>

Jess (Java Expert System Shell)

Fue desarrollado en Sandia National Laboratories, es un entorno de desarrollo de sistemas expertos que goza de gran popularidad y es de uso libre para el mundo

académico. Jess es fácil de usar y brinda gran robustez para el desarrollo de sistemas expertos. Su mayor ventaja radica en su capacidad de integración con programas externos desarrollados en Java, mediante su bien definida API, la cual, permite controlar el razonamiento basado en reglas desde programas de Java. La máquina de inferencia de Jess está basada en el encadenamiento hacia adelante del algoritmo Rete, sin embargo, también soporta el encadenamiento hacia atrás. Otra importante ventaja de Jess es la interoperabilidad dado que hace posible el almacenamiento de las bases de conocimiento en formato XML [NEG2005] .

Para mayor información podemos acceder al siguiente enlace:

<http://herzberg.ca.sandia.gov/jess/>

XPertRule KBS Shell

Ejecuta la inducción de reglas basada en árboles de decisión e incorpora lógica difusa, la interfaz al Web se hace a través de ASP (Active Server Pages) [NEG2005] .

Para mayor información podemos acceder al siguiente enlace:

<http://www.xpertrule.com/>

ExSys CORVID

Incorpora razonamiento basado en reglas con lógica difusa. Está basado en programas CGI. Permite el desarrollo de aplicaciones locales o basadas en Web donde la interacción con el usuario emula una conversación con el experto humano

para obtener las respuestas, aún en áreas donde existe una lógica probabilística compleja y muchos niveles de razonamiento [NEG2005] .

Para mayor información podemos acceder al siguiente enlace:

<http://www.exsys.com/>

Blaze Advisor (Fair Isaac Corporation)

Está basado en programas CGI ("Common Gateway Interface"). Es un administrador de reglas, diseñado para la automatización de las decisiones operativas de las empresas, brindándoles agilidad, consistencia y precisión en las interacciones con sus clientes [NEG2005] .

Para mayor información podemos acceder al siguiente enlace:

<http://www.fairisaac.com/Fairisaac/Solutions/Product+Index/Blaze+Advisor/>

eXpertise2Go

Provee el "applet" de Java denominado e2gLite que carga la base de conocimiento del servidor y se ejecuta completamente en el navegador. Se distribuye gratuitamente para fines educativos. [NEG2005]

Para mayor información podemos acceder al siguiente enlace:

<http://www.expertise2go.com/>

A continuación se describirá una herramienta que fue útil para realizar un modelamiento inicial de la base de datos.

MS VISUAL FOXPRO

MS Visual FoxPro es un entorno de desarrollo orientado a objetos para la construcción de bases de datos relacionales y desarrollo de aplicaciones. Permite crear rápidamente formularios, consultas e informes mediante las herramientas visuales de diseño. [VAQ1998]

Características

- Interoperabilidad e Internet.
- Creación de soluciones cliente / servidor
- Archivos de ayuda gráfica y ayuda estilo DBF.
- Aplicaciones distribuidas.
- Acceso a las API, bibliotecas externas y la API de VFP.

Menú

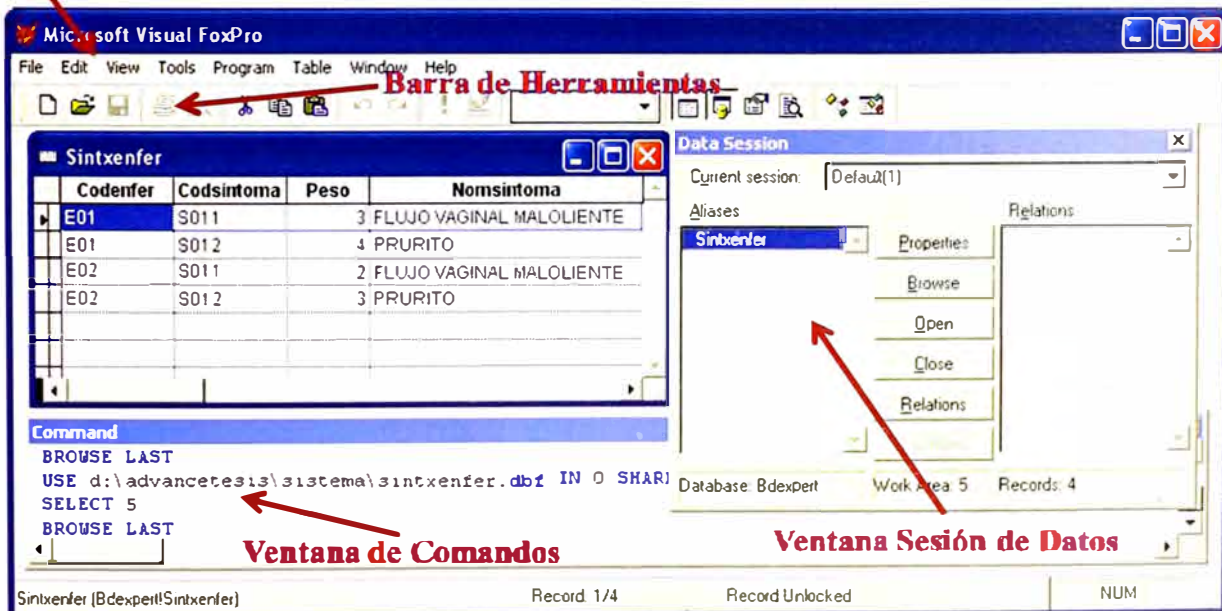


Figura 5.4: Entorno Integrado de Desarrollo de Visual FoxPro.

Fuente: [Elaboración Propia]

Este software de cuarta generación es una herramienta RAD que en el desarrollo de una aplicación proporciona:

1. Diseñador de Bases de Datos.
2. Diseñador de Formularios.
3. Diseñador de Informes, entre otros.

Luego de las primeras entrevistas con el médico especialista se pudieron identificar las siguientes entidades, que fueron almacenadas en un modelo de base de datos preliminar denominado *BdExpert*.

Paciente:

En la cual se registran los datos generales del paciente como código, nombres, apellido paterno, apellido materno, sexo, fecha de nacimiento, domicilio, etc.

Atención:

En la cual se registra el número de atención asignado a un paciente en una fecha determinada. Se consideran los campos código de atención, código de paciente, fecha de atención.

Enfermedad:

En la cual se registra el código, la descripción de la enfermedad y su probabilidad

Síntomas:

En la cual se registra el código, el nombre del síntoma o signo y el detalle que es la descripción del síntoma o signo.

SintxEnfer:

En la cual se registra los síntomas asociados a cada enfermedad de estudio, especificando en cada uno de estos el peso, que es un valor numérico entre 0 y 4. Sus campos son: código de la enfermedad, código del síntoma, descripción del síntoma y peso.

SintxPac:

En la cual se registra los síntomas identificados en una atención del paciente. Sus campos son código de atención, código de paciente y código del síntoma.

Resultado:

En el cual se registra las posibles enfermedades en una atención. Se consideran los campos código de enfermedad, resultado obtenido y probabilidad.

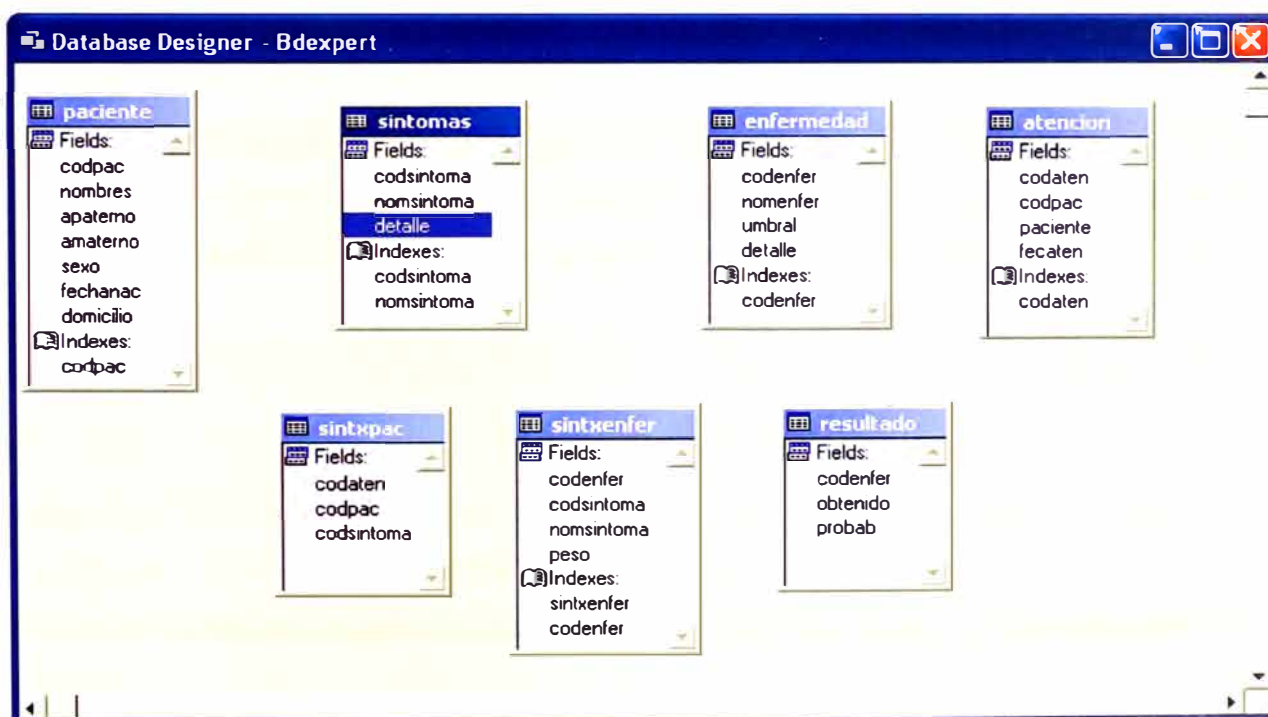


Figura 5.5: Modelo inicial que describe las entidades en el diagnóstico médico

Fuente: [Elaboración Propia]

En los inicios de la investigación se hizo un primer desarrollo usando esta herramienta, que es muy útil porque integra a la vez el entorno de programación con el diseñador de base de datos nativo y además su instalación portátil ocupa poco espacio en disco.

DELPHI

En relación con otros ambientes de programación, se puede afirmar que Delphi tiene el poder de C++ y la facilidad de Visual Basic. La principal ventaja de Delphi radica en el lenguaje usado, el *Object Pascal*, que es una evolución del Pascal estándar.

Pascal es un lenguaje de programación desarrollado en 1969 por Niklaus Wirth, catedrático de una universidad de Suiza. Se trata de un lenguaje de propósito general, capaz de tratar con los más variados tipos de datos. Una de las principales motivaciones para el desarrollo de Pascal fue la enseñanza de programación. Aún hoy en día Pascal es considerado como uno de los mejores lenguajes para el aprendizaje de la programación.

Pascal es un lenguaje normalizado, siendo el estándar designado por "Pascal ISO". En 1984, la compañía Borland lanzó el Turbo Pascal, que se convirtió en el mejor compilador de Pascal del mercado, y a partir de entonces, pasó a incluir nuevos recursos en este lenguaje, como unidades y objetos, hasta la aparición de Windows, cuando fue lanzado el Turbo Pascal para Windows y, después, el Borland Pascal, cuyo lenguaje es considerado como la primera versión del Object Pascal. En la actual versión usada por Delphi, el Object Pascal es un lenguaje poderoso, sólido y respetado, sin perder su peculiar facilidad. Es un lenguaje de alto nivel, compilado y fuertemente tipificado que soporta la implementación orientada a componentes, usando frameworks y el ambiente RAD (*Rapid Application Development*).

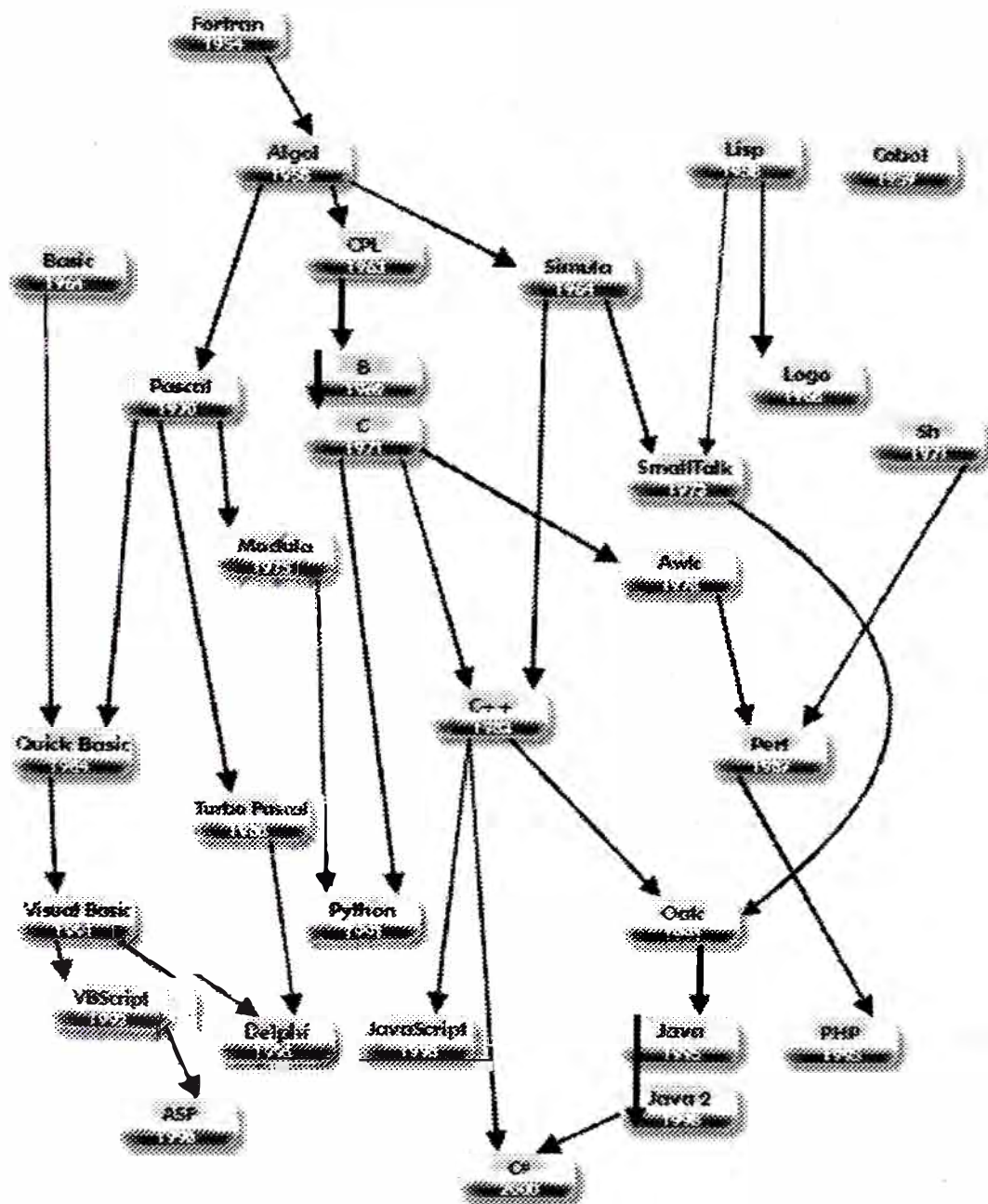


Figura 5.6: Evolución de algunos lenguajes de programación

Fuente: [Sánchez. Manual de Java2.2004. p.2]

Delphi es un paquete de herramientas de programación concebido para la programación en Windows. Un programa en Delphi no controla el flujo del

procesamiento sino responde y trata eventos que ocurren en el sistema. Existen muchos eventos que pueden ocurrir, siendo los principales aquellos generados por el usuario a través del mouse y del teclado. Esto sucede a grandes rasgos del siguiente modo: el usuario hace un clic con el mouse y Windows verifica que aplicación esta debajo del mouse en el momento que se hizo clic. A continuación manda un mensaje para la aplicación informando que ocurre un clic y las coordenadas del cursor del mouse en la ventana en el momento del clic. La aplicación luego responde al mensaje ejecutando una función de acuerdo con la posición del mouse en la ventana. Delphi se encarga del trabajo más pesado y facilita muchas cosas al programador. Detalles como las coordenadas de la ventana en la que ocurre el clic, aunque estén disponibles, difícilmente son necesarios en los programas.

Los objetos son diseñados en la ventana de forma visual, con ayuda del mouse y no por medio de programación. La programación en sí está orientada a eventos. Cuando un evento sucede tal como presionar una tecla o un clic del mouse, se envía un mensaje a la fila de mensajes de Windows. El mensaje estará disponible para todos los aplicativos en ejecución, pero sólo aquel interesado en el evento responderá al mensaje. Todo lo que el usuario necesita hacer es detectar el evento y mandar que un segmento de código sea ejecutado cuando esto suceda.

Un objeto, en Delphi, es un conjunto de códigos de programación (sub-rutinas) y propiedades. Cada elemento que vemos en un programa Windows es un objeto individual, dotado de propiedades que pueden ser alteradas.

Ejemplo: Objeto avión:

Propiedades: velocidad, altitud, inclinación.

Métodos: despegar, aterrizar.

Cambiando las propiedades de un objeto, se puede cambiar la forma como este interactúa con el ambiente. En la programación básica en Delphi no es necesario

tener un conocimiento muy grande de la programación orientada a objetos, como si lo es en otros lenguajes. Los objetos están listos para ser usados y Delphi cuida de casi todo.

La creación de aplicaciones comienza con el montaje de componentes en ventanas, como si fuese un programa gráfico, ofreciendo al programador la facilidad de manipulación de ventanas, cajas, botones, barras de desplazamiento, listas de opciones, cuadros de entradas de texto y otros comunes al ambiente. El usuario puede utilizar componentes desarrollados por terceros o crear sus propios componentes. Hay también las herramientas necesarias para el manejo de archivos y la creación de bases de datos, permitiendo la creación de aplicativos con bases de datos sin la necesidad de adquirir otro programa.

5.2.2.4.2 ELECCION DE LA HERRAMIENTA

El prototipo de sistema experto probabilístico fue desarrollado en el ambiente de programación Delphi versión 7.0. Las razones para la elección de esta herramienta son la experiencia que el investigador tiene en el manejo del software y la versatilidad del lenguaje para el uso de estructuras de datos, sobre todo cadenas, así como arrays numéricos y de cadenas, listas enlazadas, archivos; la manipulación de objetos y la fácil creación de ejecutables para las diversas revisiones del prototipo.

He creído conveniente incluir la etapa que corresponde a la Implementación del prototipo y los resultados en el siguiente capítulo.

CAPÍTULO VI

DESARROLLO E IMPLEMENTACIÓN DEL PROTOTIPO

6.1 DESCRIPCIÓN DEL SISTEMA DESARROLLADO

6.1.1 OBJETIVOS

Esta sección tiene como objetivo presentar el desarrollo del prototipo de sistema experto al cual se le ha denominado ExpertITS (Sistema Experto para la detección de Infecciones de Transmisión Sexual), que se utiliza para el diagnóstico de enfermedades del síndrome de flujo vaginal usando el enfoque probabilístico. Este trabajo está basado en conocimientos de especialistas en el área, así como de informaciones contenidas en libros de Infectología recomendados por ellos, así como libros de programación del ambiente de desarrollo Delphi.

6.1.2 Público Objetivo del Sistema Desarrollado

Este software está dirigido a profesionales que están inmersos en la investigación y aprendizaje en el área de Ginecología. Sirve también de soporte al médico especialista ginecólogo durante la realización de la consulta.

6.1.3 Descripción General

Este software se idealizó como una manera de contribuir con el trabajo de profesionales y estudiantes del área, así como para servir de apoyo en los estudios y también para actuar de manera conjunta con el infectólogo en el consultorio.

6.1.4 Productos del Sistema

Este sistema crea archivos con datos de las enfermedades y sus verosimilitudes, los síntomas así como los síntomas asociados a cada enfermedad, acompañado de sus respectivas probabilidades. Luego en base a las enfermedades cargadas en su base de hechos y de acuerdo a la consulta al paciente genera el diagnóstico diferencial.

6.1.5 Características del usuario final

Todos los usuarios del sistema deben tener nociones básicas acerca del manejo de una PC con sistema operativo Windows.

6.1.6 Control al acceso externo

El sistema no posee un módulo específico para controlar la seguridad en el acceso de los usuarios.

6.1.7 Requisitos de desempeño

Un equipo básico con procesador medio y una cantidad razonable de memoria RAM será suficiente para la ejecución de las funcionalidades del sistema. El prototipo no requiere la instalación de software adicional para su funcionamiento.

6.1.8 Pantallas del Sistema

6.1.8.1 Pantalla Principal

A continuación se presenta la pantalla principal del sistema. Esta ventana contiene un menú con opciones para el manejo de enfermedades, síntomas y generación del diagnóstico.

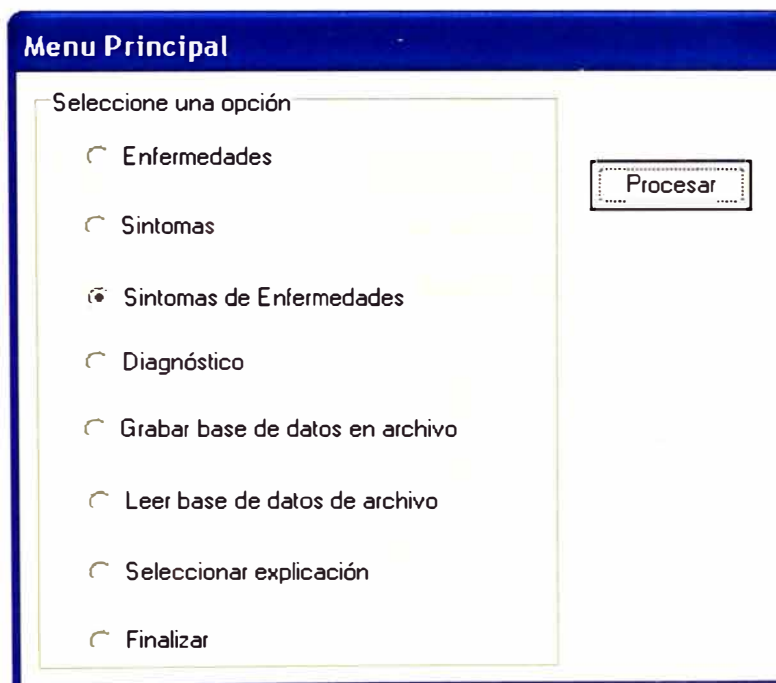


Figura 6.1: Pantalla principal del prototipo de Sistema Experto.

Fuente: [Elaboración Propia]

6.1.8.2 Descripción de las opciones

6.1.8.2.1 Síntomas

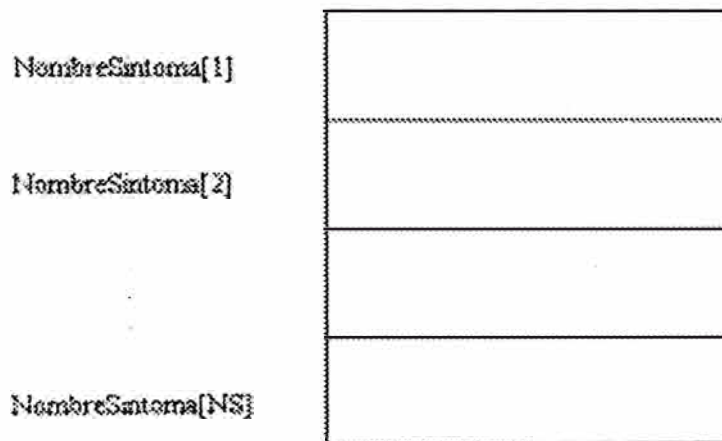
Nos permitirá realizar las siguientes operaciones sobre la entidad Síntomas:

AÑADIR: Incorpora un síntoma a la lista de síntomas.

BORRAR: Elimina un síntoma de la lista de síntomas.

LISTAR: Presenta en pantalla la relación de síntomas.

Los síntomas que consideremos se almacenarán en el siguiente array de cadenas:



donde
NS: número de síntomas

Figura 6.2: Almacenamiento interno de los Síntomas.

Fuente: [Elaboración Propia]

6.1.8.2.1.1 Síntomas - Añadir

Incorpora un síntoma a la lista de síntomas.

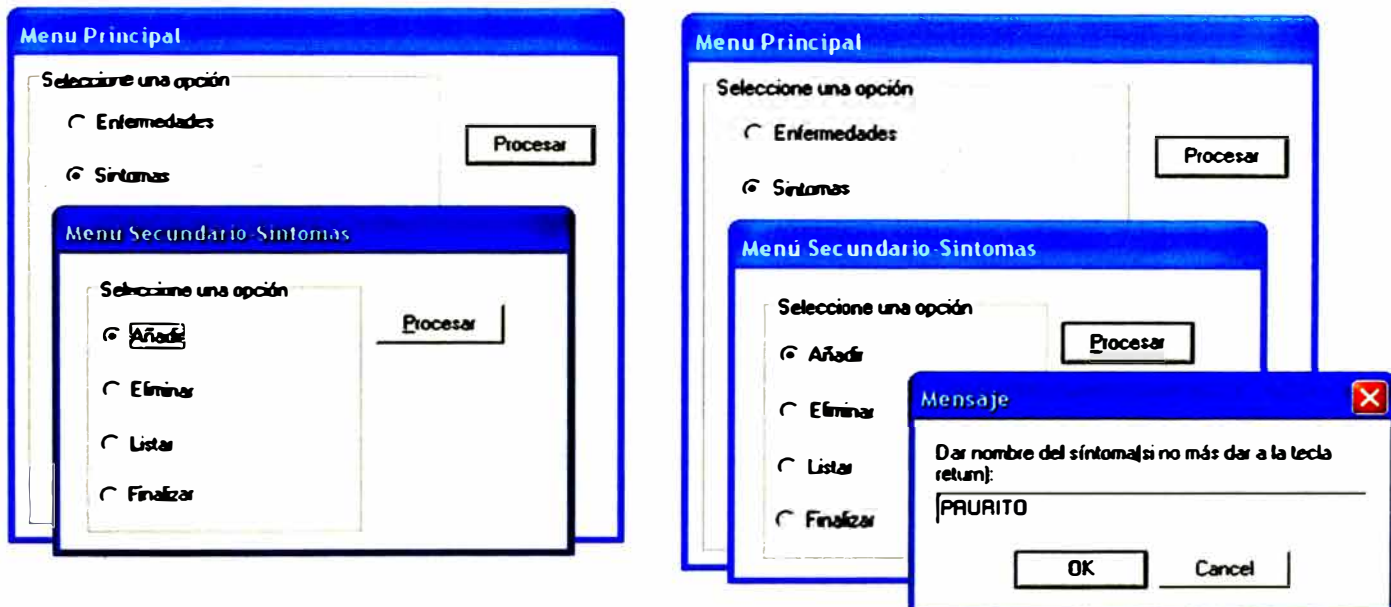


Figura 6.3: Ingreso de los Síntomas.

Fuente: [Elaboración Propia]

6.1.8.2.1.2 Síntomas - Listar

Presenta en pantalla la relación de síntomas registrados en la base de conocimientos.



Figura 6.4: Listado de los Síntomas.

6.1.8.2.1.3 Síntomas - Eliminar

Elimina un síntoma registrado en la lista de síntomas.

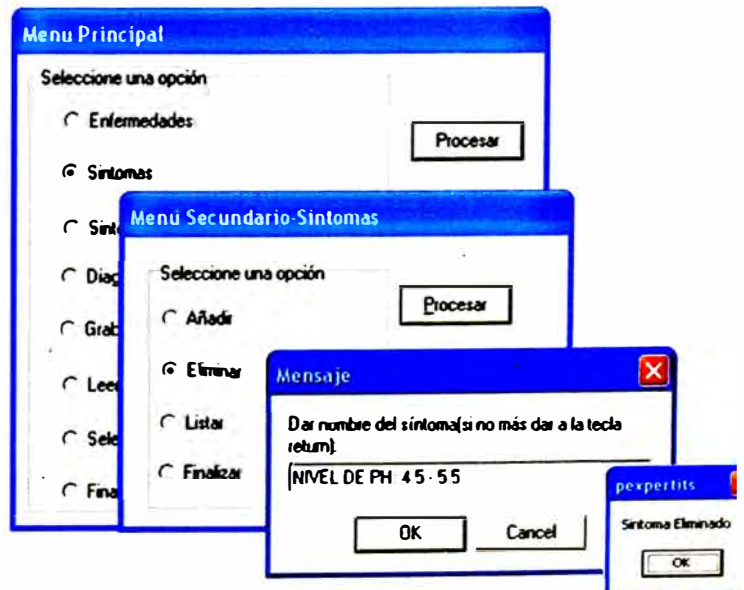




Figura 6.5: Eliminación de Síntomas.

Fuente: [Elaboración Propia]

6.1.8.2.2 Enfermedades

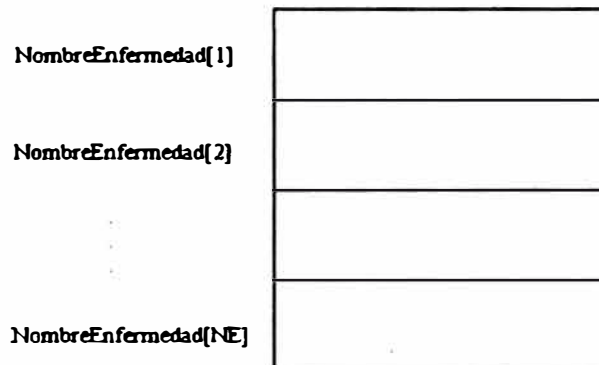
Nos permitirá realizar las siguientes operaciones sobre la entidad Enfermedades:

AÑADIR: Incorpora una enfermedad a la lista de enfermedades.

BORRAR: Elimina una enfermedad de la lista de enfermedades.

LISTAR: Presenta en pantalla la relación de enfermedades.

Las Enfermedades de estudio se almacenarán en el siguiente array de cadenas:



donde
NE: número de enfermedades

Figura 6.6: Almacenamiento interno de las Enfermedades.

Fuente: [Elaboración Propia]

6.1.8.2.2.1 Enfermedades - Añadir

Incorpora una enfermedad y su probabilidad a priori a la lista de enfermedades.

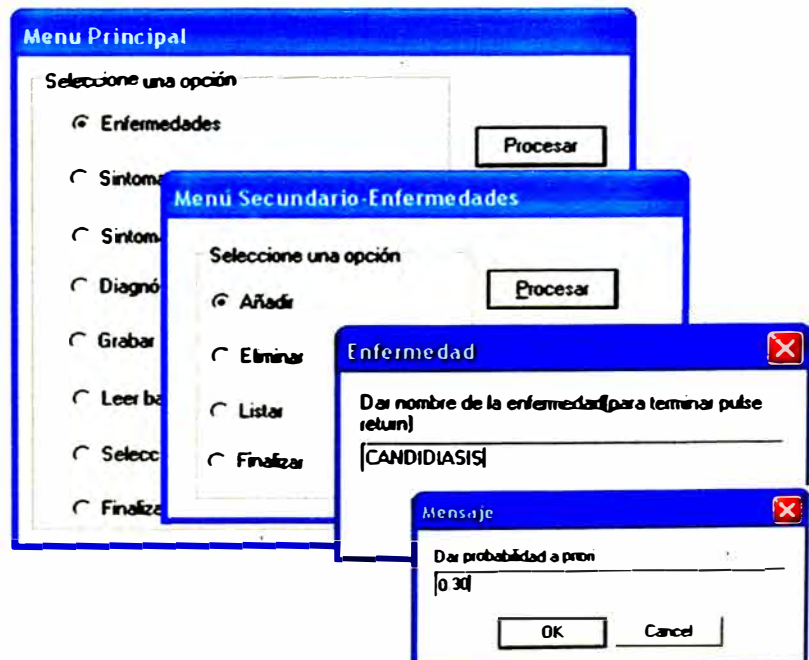
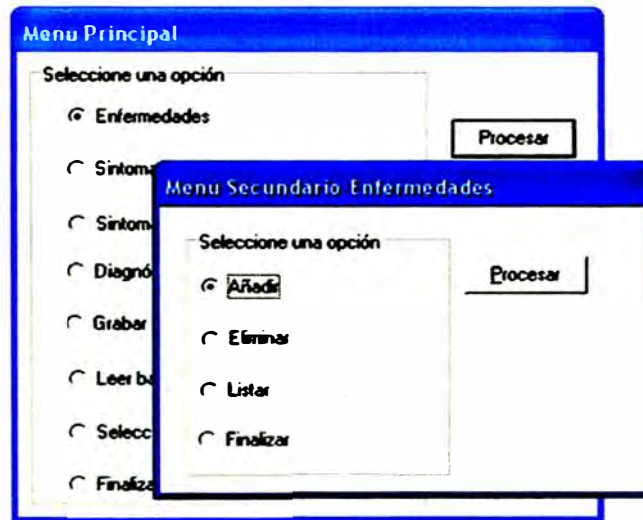


Figura 6.7: Ingreso de las Enfermedades.

Fuente: [Elaboración Propia]

6.1.8.2.2 Enfermedades - Listar

Presenta en pantalla la relación de enfermedades registradas en la base de conocimientos.

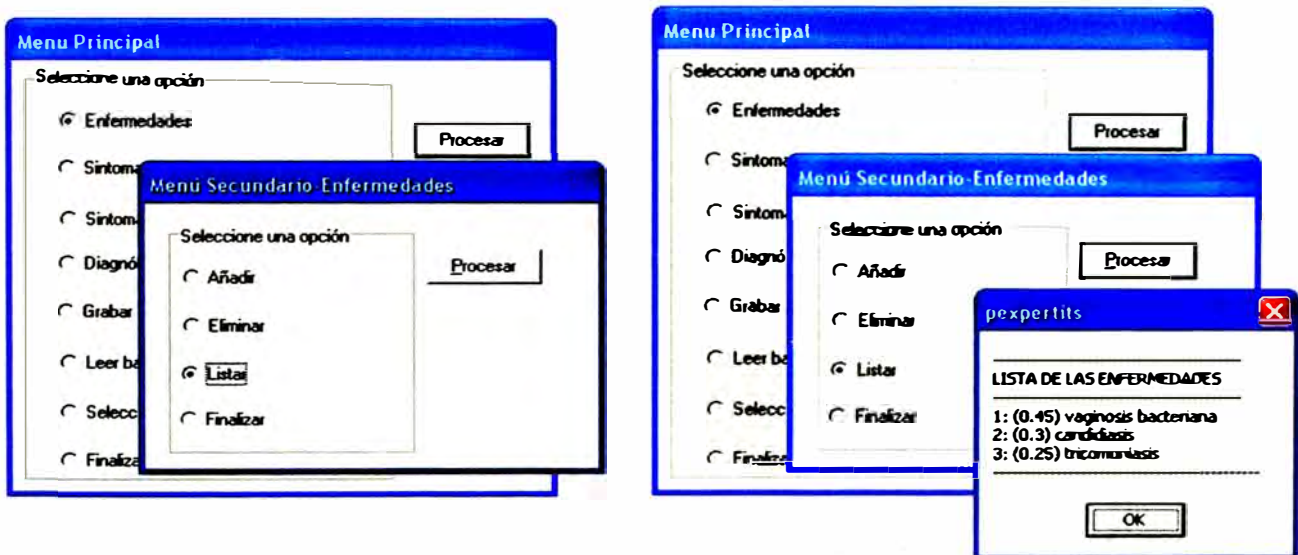


Figura 6.8: Listado de las Enfermedades.

Fuente: [Elaboración Propia]

6.1.8.2.3 Enfermedades - Eliminar

Elimina una enfermedad de la lista de enfermedades.

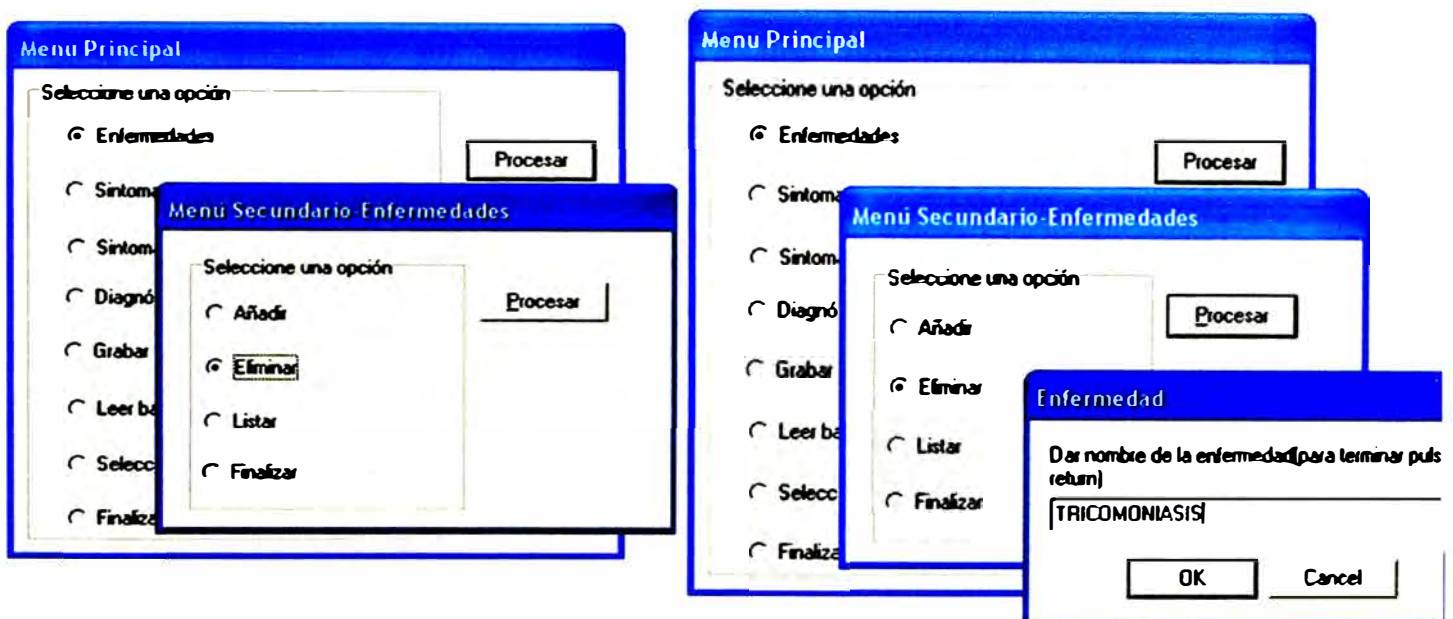


Figura 6.9: Eliminación de las Enfermedades.

Fuente: [Elaboración Propia]

6.1.8.2.3 Síntomas de Enfermedades

Nos permitirá realizar las siguientes operaciones sobre la entidad SíntomasEnfermedades:

AÑADIR: Asocia un síntoma a una enfermedad en la lista de SíntomasEnfermedades

BORRAR: Elimina un síntoma asociado a una enfermedad.

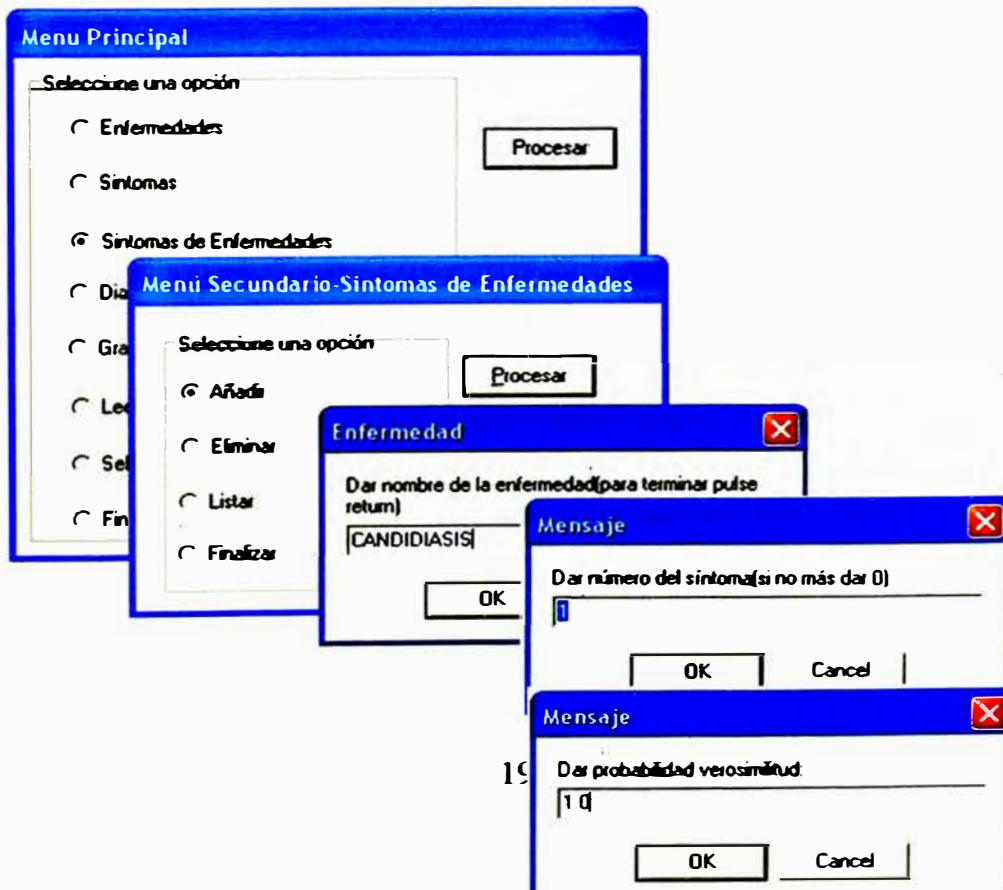
LISTAR: Presenta en pantalla la relación de síntomas asociados a cada enfermedad.

6.1.8.2.3.1 Síntomas de Enfermedades - Añadir

Asocia un síntoma (y su verosimilitud) a una enfermedad en la lista de SíntomasEnfermedades.



Figura 6.10: Asociación de Síntomas a las Enfermedades.



6.1.8.2.3.2 Síntomas de Enfermedades - Listar

Presenta en pantalla la relación de síntomas asociados a cada enfermedad.

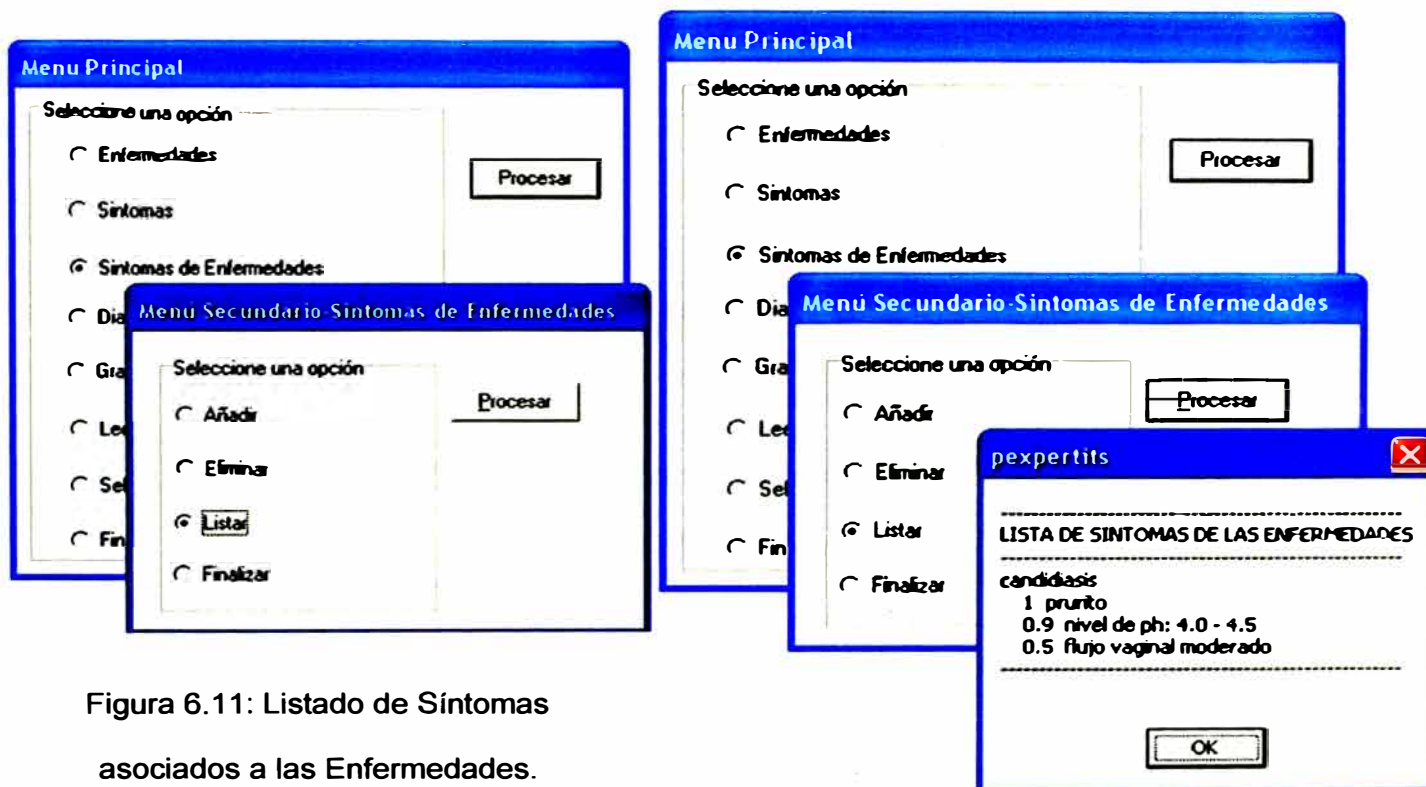


Figura 6.11: Listado de Síntomas asociados a las Enfermedades.

Fuente: [Elaboración Propia]

6.1.8.2.4 Grabar base de datos en archivo

Transfiere los datos almacenados en memoria a un archivo en disco.

Se ingresa la Ruta y nombre del archivo de texto a crear.



Figura 6.12: Grabar base de datos en archivo.

Fuente: [Elaboración Propia]

6.1.8.2.5 Leer base de datos de archivo

Recupera información de un archivo de datos externo, la misma que es usada para inicializar los diversos arrays y variables definidas.

Se ingresa la Ruta y nombre del archivo de texto a leer.

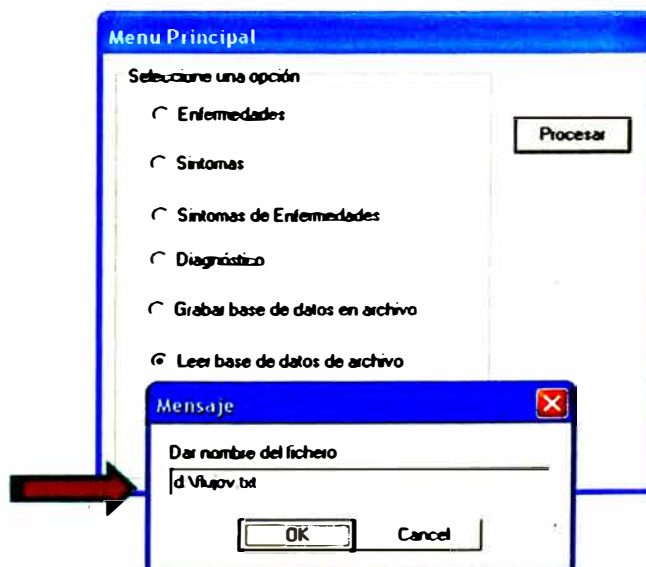


Figura 6.13: Leer base de datos de archivo.

Fuente: [Elaboración Propia]

Luego de esto, pasamos a revisar la carga de datos.

Revisamos la relación de enfermedades cargadas a la base de conocimientos.

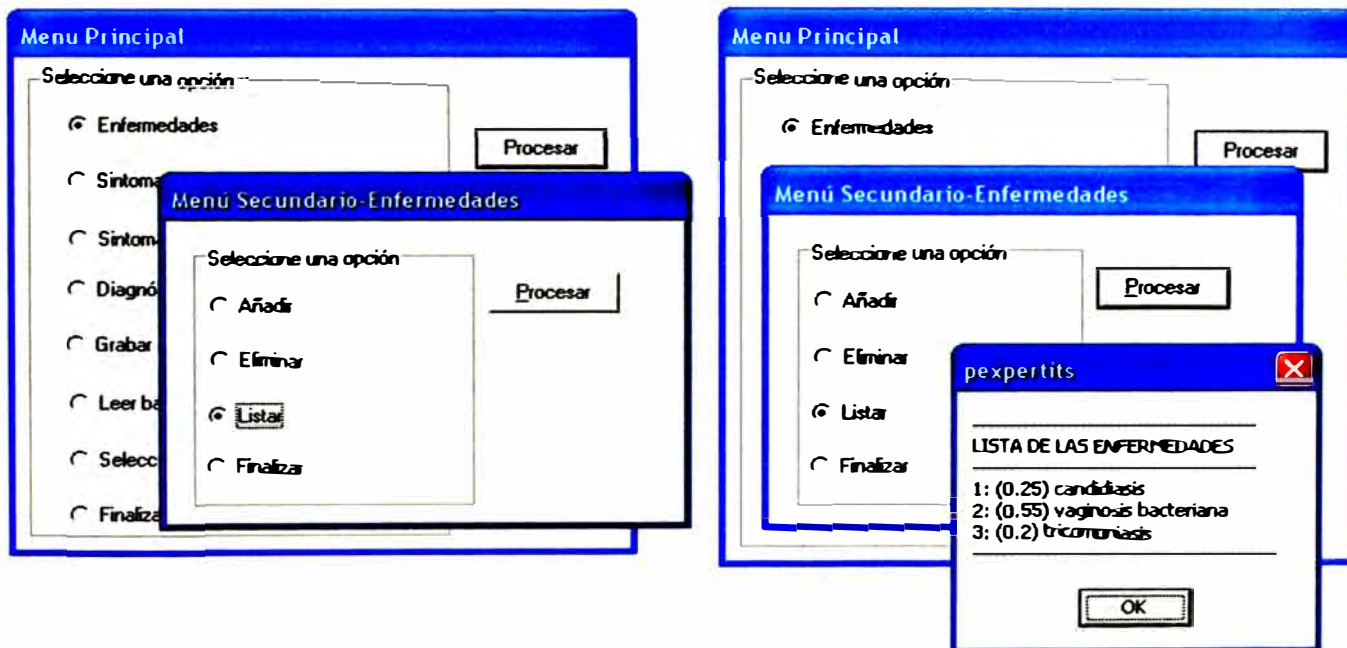


Figura 6.14: Carga de datos de enfermedades.

Fuente: [Elaboración Propia]

Revisamos la relación de síntomas cargados a la base de conocimientos.

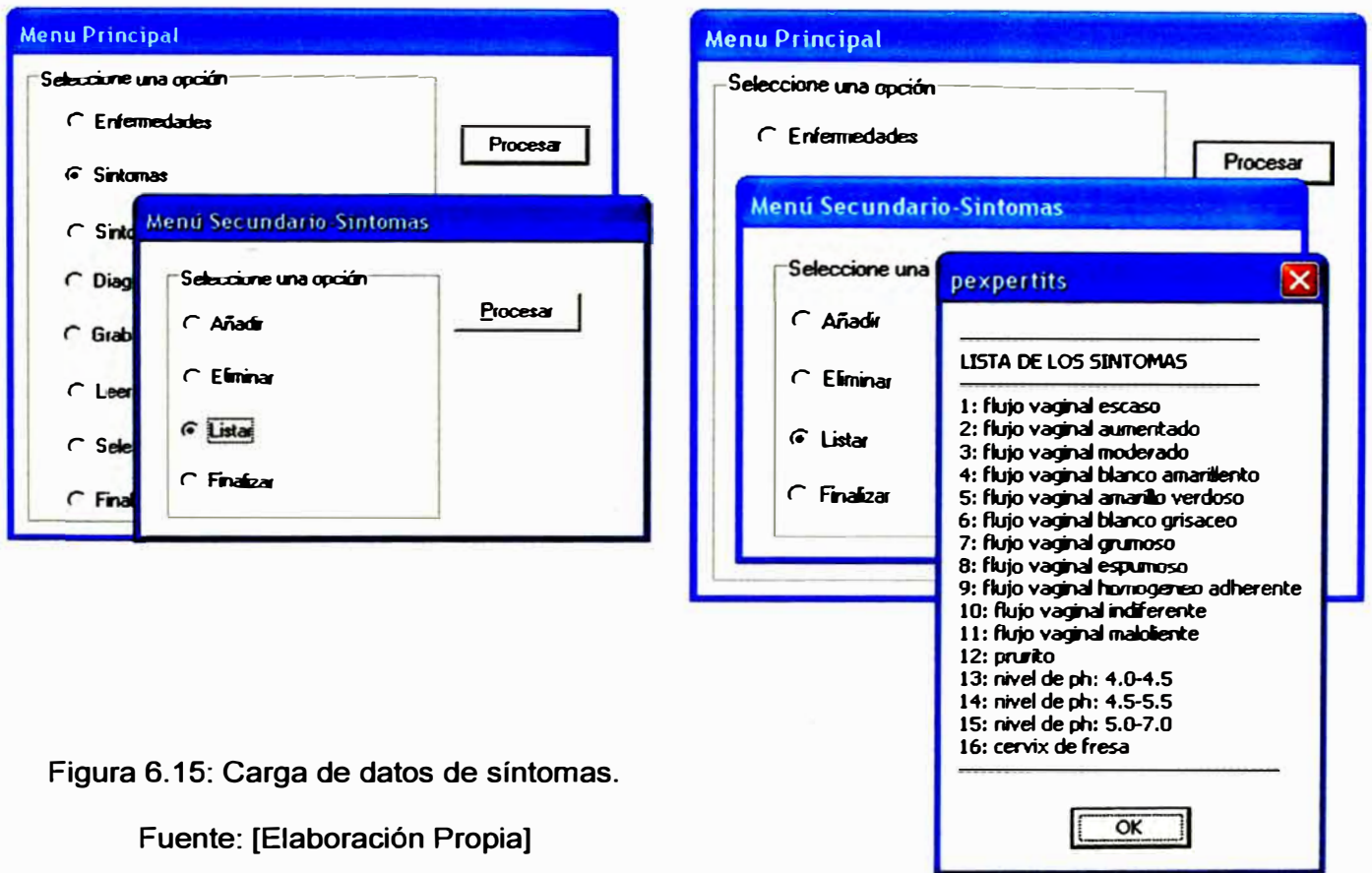


Figura 6.15: Carga de datos de síntomas.

Fuente: [Elaboración Propia]

Revisamos la relación de síntomas asociados a cada enfermedad.

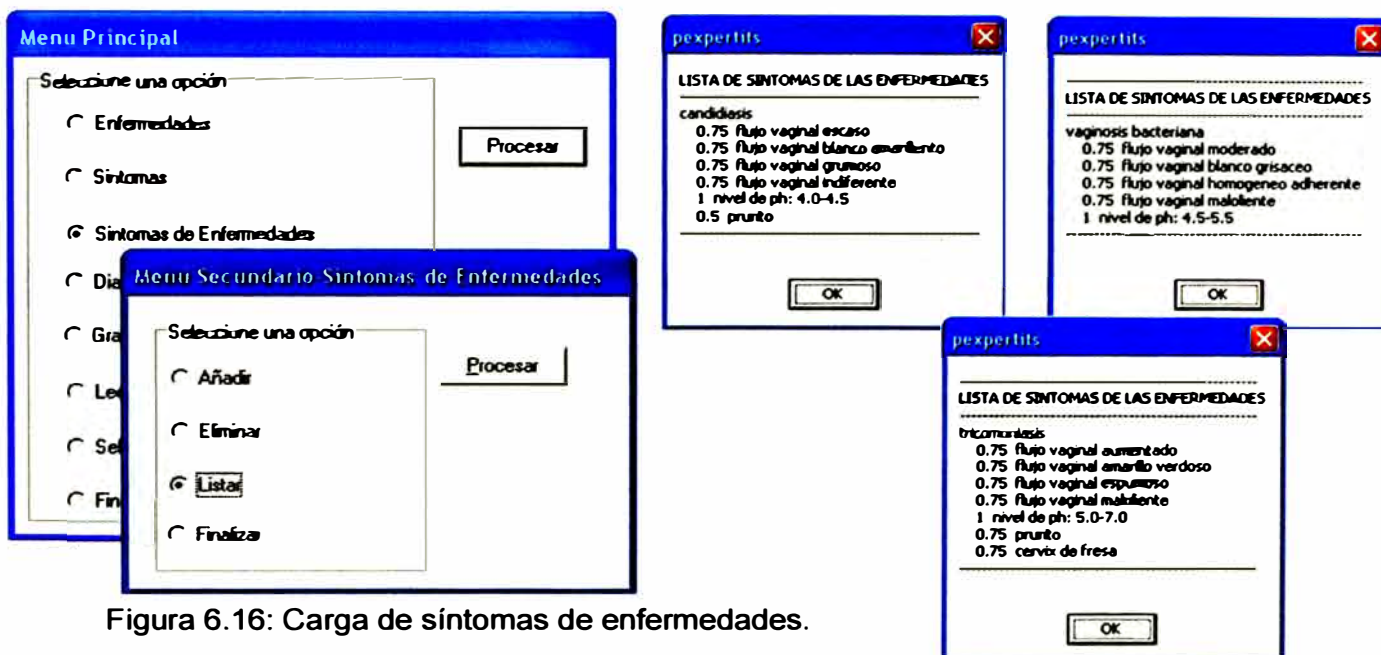


Figura 6.16: Carga de síntomas de enfermedades.

Fuente: [Elaboración Propia]

6.1.8.2.6 Diagnóstico

Solicita el ingreso de los síntomas que decida el paciente. Usa una variable bandera que indica el término del primer ingreso de síntomas (bandera = 9999).

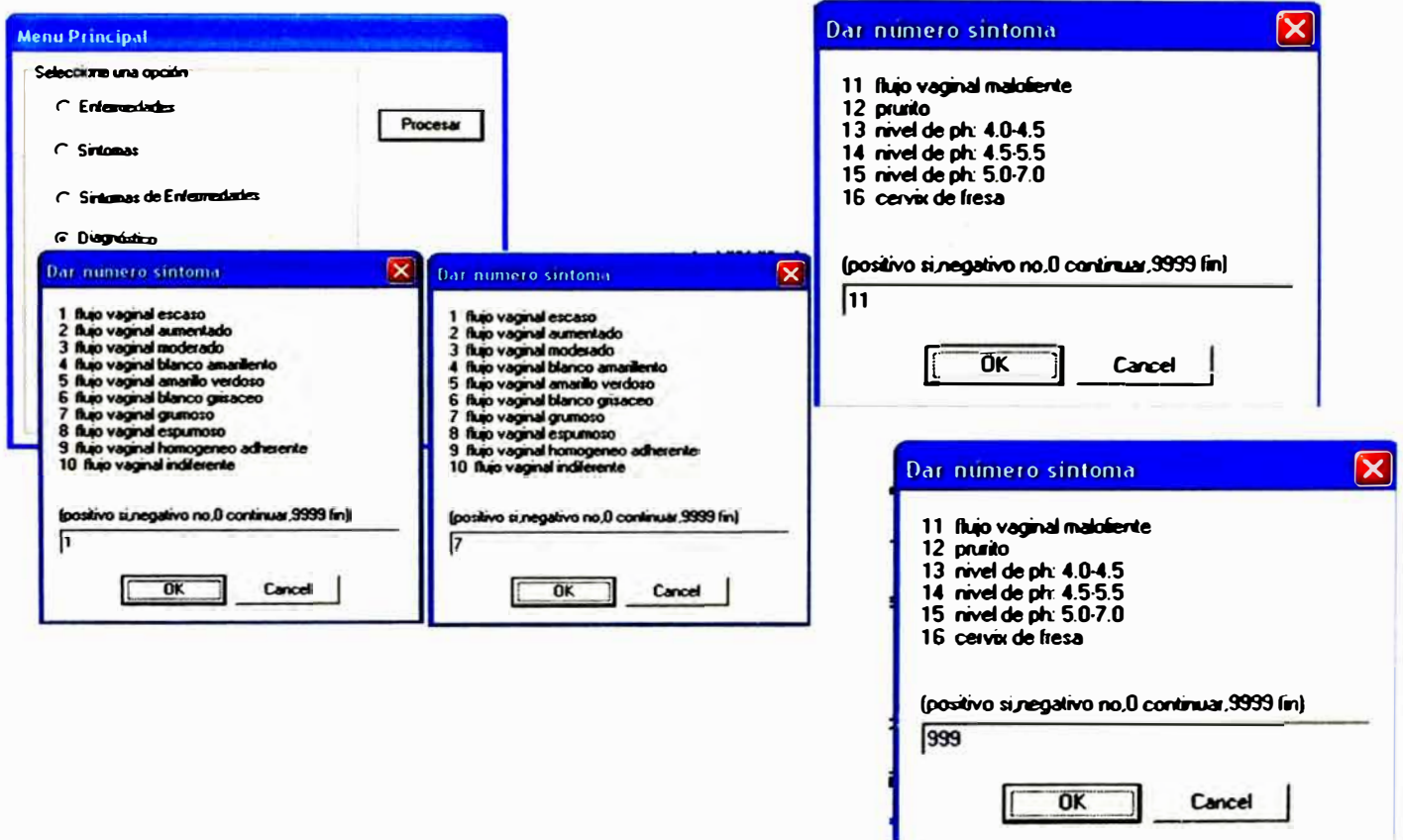


Figura 6.17: Ingreso inicial de síntomas en la consulta.

Fuente: [Elaboración Propia]

A continuación presenta un diagnóstico presuntivo con las posibles enfermedades ordenadas de acuerdo a de probabilidad obtenida.



Figura 6.18: Diagnóstico diferencial inicial.

Fuente: [Elaboración Propia]

Una vez conocida la lista se manejan 3 opciones:

Automático: el prototipo se encarga de preguntar.

No: El usuario pregunta libremente.

Si: Termina definitivamente el ingreso de síntomas.



Figura 6.19: Opciones de cuestionario

Fuente: [Elaboración Propia]

Luego el prototipo pregunta acerca de un síntoma y el usuario tiene 3 opciones: si, no, no sabe.

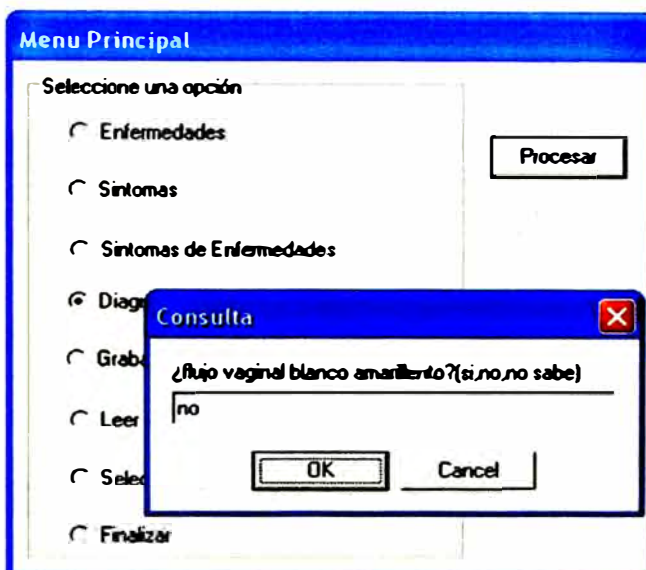


Figura 6.20: Preguntas del cuestionario automático

Una vez conocida la respuesta se presenta una segunda aproximación de diagnóstico, también ordenada descendentemente.



Figura 6.21: Segunda aproximación de Diagnóstico.

En seguida el prototipo pregunta acerca de otro síntoma y el usuario tiene 3 opciones: si, no, no sabe. Después de conocida la respuesta se presenta una tercera aproximación de diagnóstico, ordenada descendentemente.



Figura 6.22: Tercera aproximación de Diagnóstico.

Fuente: [Elaboración Propia]

A continuación el prototipo pregunta acerca de otro síntoma y el usuario tiene 3 opciones: si, no, no sabe. Después de conocida la respuesta se presenta una cuarta aproximación de diagnóstico, ordenada descendentemente.

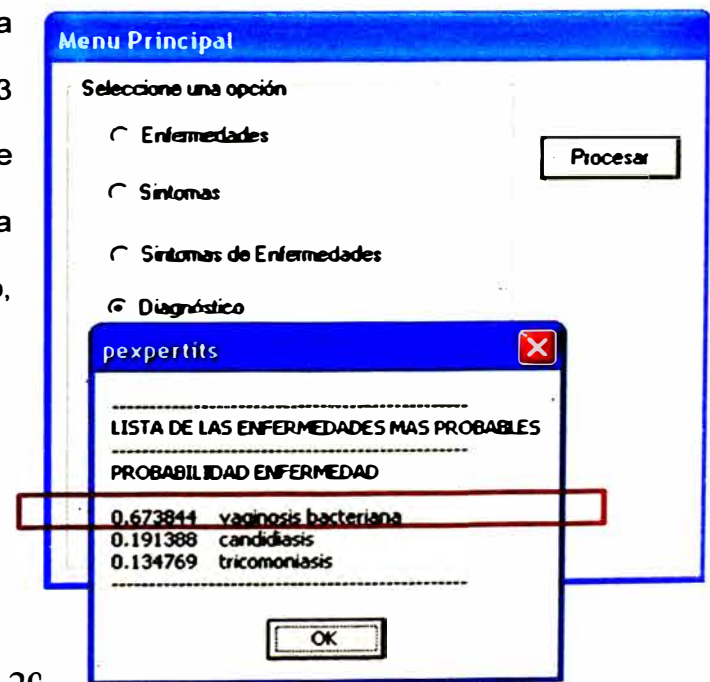




Figura 6.23: Cuarta aproximación de Diagnóstico.

A continuación el prototipo pregunta acerca de otro síntoma y el usuario tiene 3 opciones: si, no, no sabe. Después de conocida la respuesta se presenta la quinta aproximación de diagnóstico, ordenada descendientemente.

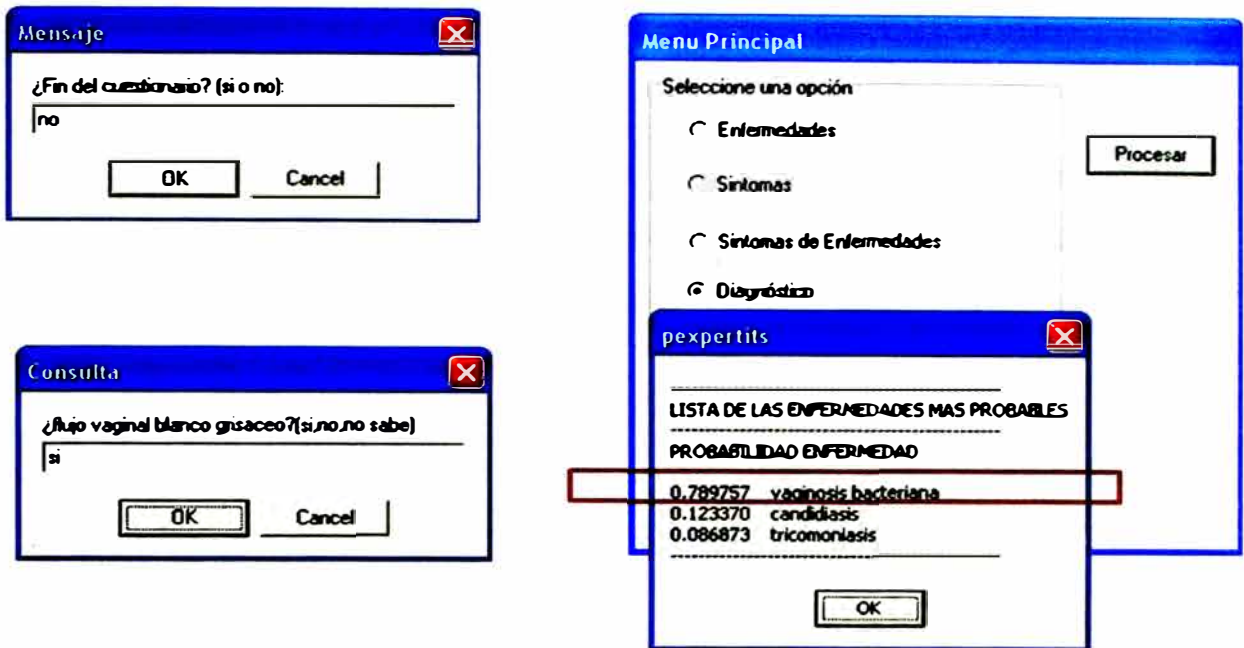


Figura 6.24: Quinta aproximación de Diagnóstico.
Fuente: [Elaboración Propia]

A continuación el prototipo pregunta acerca de otro síntoma y el usuario tiene 3 opciones: si, no, no sabe. Después de conocida la respuesta se presenta la sexta aproximación de diagnóstico, ordenada descendentemente.

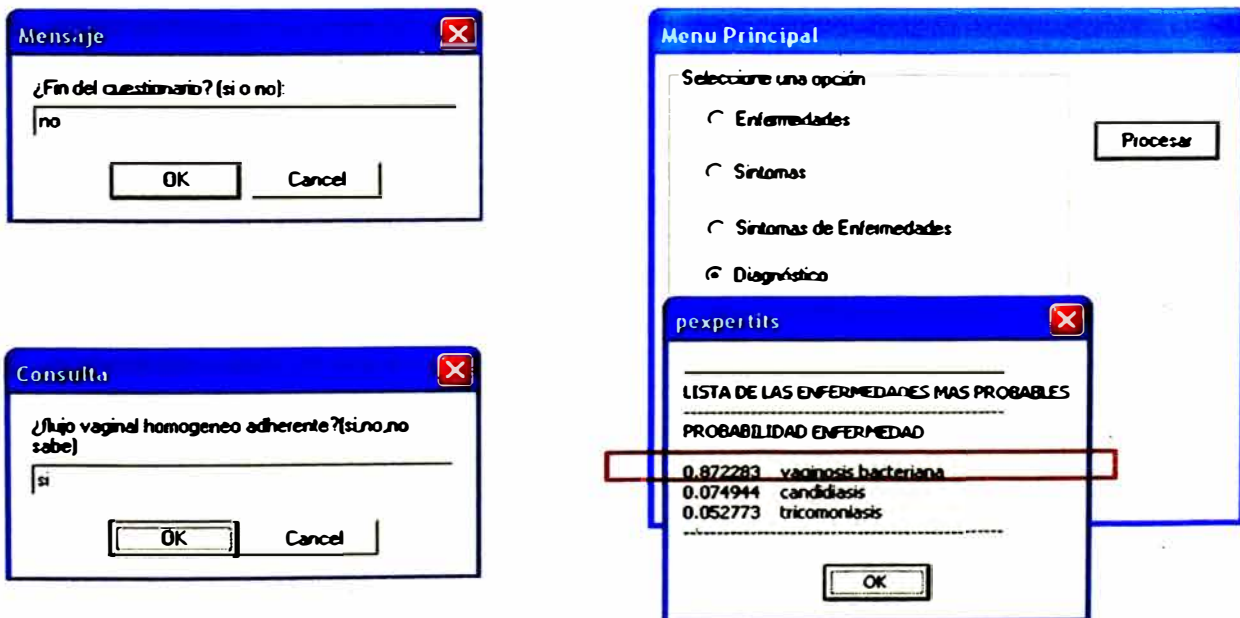


Figura 6.25: Sexta aproximación de Diagnóstico.
Fuente: [Elaboración Propia]

A continuación el prototipo pregunta acerca de otro síntoma y el usuario tiene 3 opciones: si, no, no sabe. Después de conocida la respuesta se presenta la séptima aproximación de diagnóstico, ordenada descendientemente.

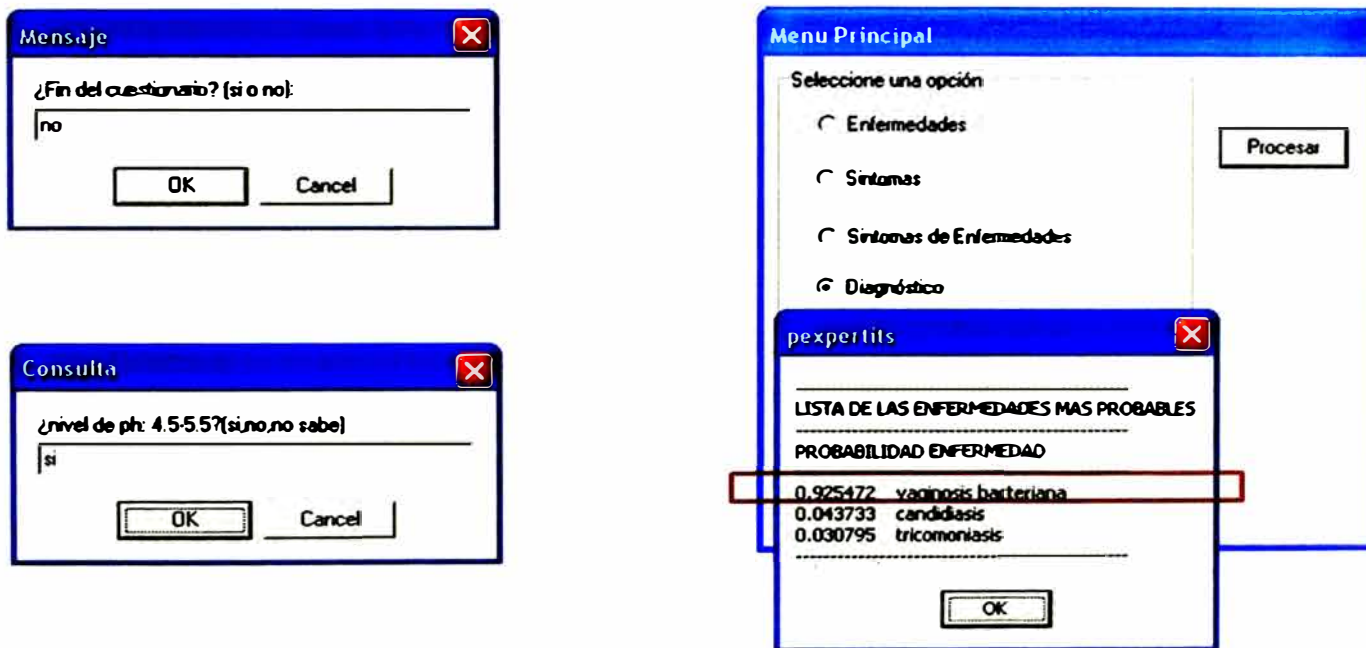


Figura 6.26: Séptima aproximación de Diagnóstico.

Fuente: [Elaboración Propia]

A continuación el prototipo pregunta acerca de otro síntoma y el usuario tiene 3 opciones: si, no, no sabe. Después de conocida la respuesta se presenta la octava aproximación de diagnóstico, ordenada descendientemente.

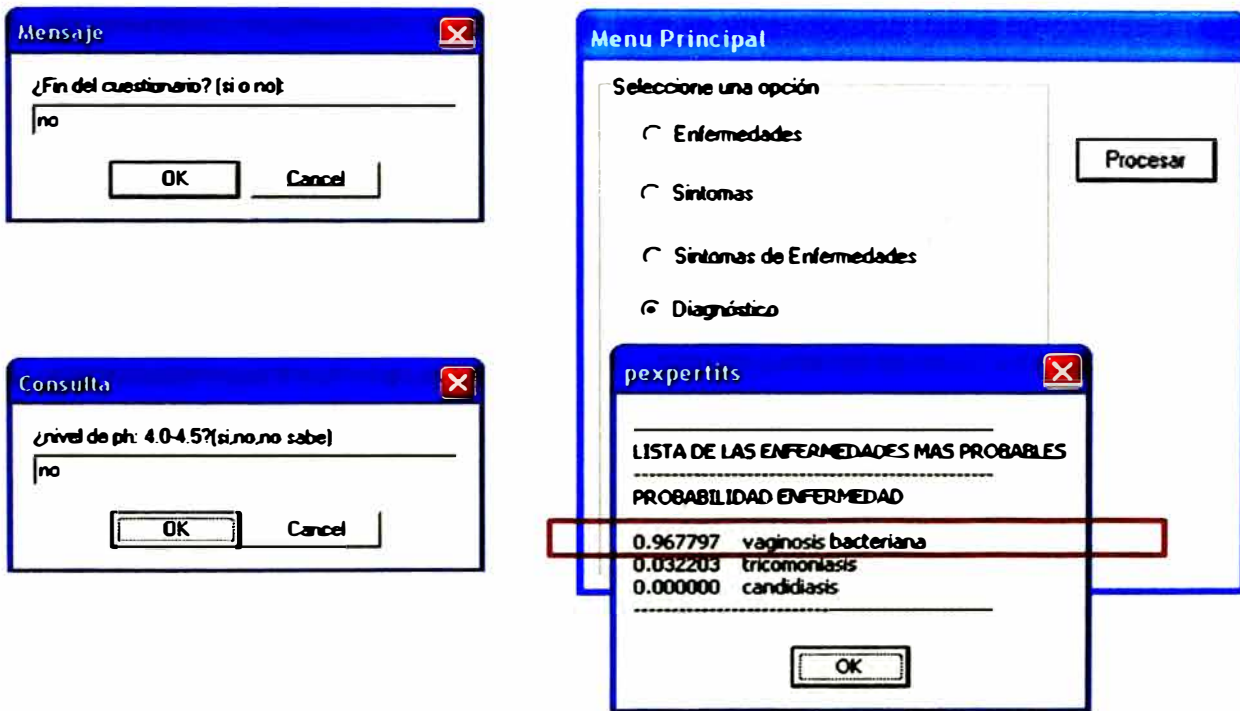


Figura 6.27: Octava aproximación de Diagnóstico.

Fuente: [Elaboración Propia]

A continuación el prototipo pregunta acerca de otro síntoma y el usuario tiene 3 opciones: si, no, no sabe. Una vez conocida la respuesta se presenta una última aproximación en la que el diagnóstico es evidente, por lo que acaba.

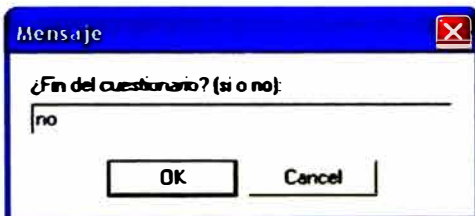
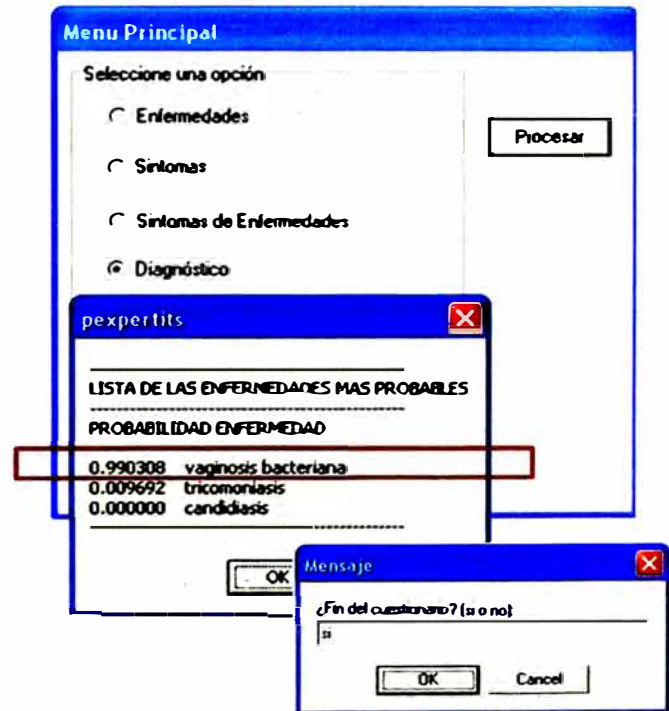


Figura 6.28: Diagnóstico Final.



CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES:

1. Se presentó un estudio sobre el estado del arte de los SE en el que se describió la historia y su importancia para los avances en la IA. Se concluye que el desarrollo de SE partió casi desde los inicios de la IA y está fuertemente ligado a orígenes académicos donde siempre fue de interés emular las habilidades de un especialista.
2. Se desarrolló de manera extensa, incluyendo ejemplos ilustrativos, las principales técnicas que existen para representar el conocimiento, las mismas que pueden ser tomadas como referencia en futuras investigaciones para abordar el estudio de algún dominio en cualquier otra especialidad, no necesariamente la médica.

3. En esta investigación se elaboró un prototipo de sistema experto para el diagnóstico del Flujo Vaginal utilizándose el enfoque probabilístico, y se constató que es una metodología muy eficiente para lograr el objetivo fundamental que es el obtener resultados muy parecidos a los alcanzados por los especialistas humanos frente a un determinado problema.
4. Se logró desarrollar un prototipo que separa la base de conocimientos del motor de inferencia, lo cual permitirá abordar el estudio de otra patología de cualquier especialidad aplicando el mismo enfoque probabilístico.
5. Se escogió el diagnóstico de infecciones de transmisión sexual pues en la actualidad representa uno de los problemas más comunes de salud pública a nivel mundial agudizándose su situación en países con problemas de extrema pobreza como el Perú.
6. La asociación de los signos y síntomas a las enfermedades de estudio, así como la asignación de probabilidades a priori de las enfermedades y las verosimilitudes de los síntomas de enfermedades se basaron en la experiencia del médico en su ejercicio profesional.
7. Se desarrolló un software que sigue los lineamientos para el tratamiento sindrómico definidos en la norma técnica de salud para el manejo de ITS publicada por el MINSA cuyo ámbito de aplicación es a nivel nacional, es decir el prototipo se idealizó para que pueda ser utilizado como una herramienta que ayude a dar solución a una problemática de nuestra sociedad.

8. Esta investigación traza un camino a seguir en el área de la inteligencia artificial médica, el enfoque probabilístico, que puede traer muchos beneficios tanto para los médicos como para los pacientes, contribuyendo de este modo a profundizar el conocimiento en el proceso de diagnóstico médico y tratamiento de enfermedades así como para el control de los procesos médicos.

RECOMENDACIONES:

1. Luego de la implementación de una nueva funcionalidad del prototipo se sugiere desarrollar las pruebas en presencia del médico especialista de modo que apruebe y/o haga las observaciones que contribuyan a mejorar la performance del software.
2. Se recomienda asegurar un alto nivel de compromiso de cada uno de los actores del equipo de desarrollo del SE y que interioricen que el éxito del proyecto estará sujeto a cuan bien se desempeñen en la labor que se les haya encomendado para poder garantizar la confiabilidad en los resultados inferidos por el SE orientado al diagnóstico.
3. Es recomendable la puesta en producción de un prototipo de SE médico en una institución de salud de cualquier envergadura pues esto obligaría a que el personal asistencial ponga mayor cuidado en el registro de los hallazgos obtenidos del paciente en cada atención debido a que con esta información se podría ampliar el número de casos evaluados por el sistema experto, y esto contribuiría a mejorar su base de conocimientos y por ende podría inferir un diagnóstico más certero.
4. Se propone fomentar acuerdos con los Jefes de Servicios Médicos a través de la suscripción de protocolos, que son estudios interdisciplinarios destinados a la investigación que buscan mejorar la rapidez en el diagnóstico, están sujetos a la autorización de la Alta Dirección y su utilidad es la de mejorar el proceso de atención.

5. Se recomienda el uso del ambiente de programación Delphi 7.0 para diseñar sistemas expertos basados en la propagación de probabilidades ya que es una herramienta eficiente para la representación del conocimiento y además es flexible en la aplicación de los diversos paradigmas de la programación, a lo que le podemos sumar la facilidad para generar ejecutables lo cual ayuda en el modelo del prototipado.

TRABAJOS FUTUROS:

El área de la medicina que se escogió para la aplicación del sistema experto constituye un campo amplio para la investigación. Se proponen las siguientes mejoras cuyo objeto es el de expandir las funcionalidades del sistema ayudando de manera práctica al profesional en salud:

- Módulo de análisis estadístico de los datos clínicos, que vendría dotado del mecanismo de seguridad para proteger la confidencialidad de los pacientes, para la realización de los estudios sobre el comportamiento de la enfermedad.
- Desarrollo de una interfaz web de modo que el sistema permita el acceso simultáneo de varios usuarios y así puedan trabajar en forma paralela y desde cualquier lugar. Esto permitiría la actualización constante de los datos clínicos y el crecimiento continuo de la base de datos hará factible la realización de estudios con una vasta y actualizada base de datos lo que ayudará a obtener información sobre el comportamiento de la enfermedad en los casos más diversos; así también podríamos incluir un módulo de tratamiento, luego de la identificación de la enfermedad.

Como trabajo futuro en el área de sistemas probabilísticos sería interesante revisar otras patologías de estudio de infecciones de transmisión sexual.

GLOSARIO DE TÉRMINOS DE INTELIGENCIA ARTIFICIAL

AGENTE: Es algo que razona (agente viene del latín agere, hacer). En los agentes se espera que estén dotados de controles autónomos, que perciban su entorno, que persistan durante un periodo de tiempo prolongado, que se adapten a los cambios y que sean capaces de alcanzar objetivos diferentes.

ALGORITMO: Se remonta a la época de al-Khowarizmi, un matemático persa del siglo IX, con cuyos escritos también se introdujeron números arábigos. Se piensa que el primer algoritmo no trivial es el algoritmo euclídeo para el cálculo del máximo común divisor.

CIENCIAS DE LA COMPUTACIÓN: Disciplina que se ocupa del estudio de sistemas de cómputo incluyendo procesos algorítmicos y principios que involucran el diseño de software y hardware.

Los profesionales en ciencias de la computación se encargan del diseño de algoritmos, lenguajes, herramientas y sistemas de software. Diseñan y construyen software, creando soluciones eficientes a problemas del mundo real en campos como la medicina, el comercio, la biología y los negocios. Entre sus campos están los algoritmos y las estructuras de datos, lenguajes de programación y compiladores, bases de datos, Inteligencia Artificial y Robótica, computación científica, gráficos por computador, Procesamiento Paralelo y Sistemas Distribuidos, etc.

DENDRAL: Primer sistema de conocimiento intenso cuya base de conocimiento estaba formada por grandes cantidades de reglas de propósito particular.

DOMINIO DE CONOCIMIENTO: Es el conocimiento que tiene el especialista para resolver problemas específicos. En un SE médico para diagnosticar enfermedades infecciosas, el dominio del conocimiento es la medicina y consta del conocimiento de las enfermedades, sus síntomas y tratamientos.

DOMINIO DE MODELO: Es el conjunto de objetos que contiene. A estos objetos a veces se les denomina elementos del dominio.

EMULAR: Significa que el sistema experto tiene el objetivo de actuar en todos los aspectos como un especialista humano.

ENCADENAMIENTO: Conjunto de varias inferencias que conectan el problema con la solución.

HEURÍSTICA: Proviene del término griego *heuriskein* que significa hallar, inventar. Es una técnica que busca soluciones buenas aunque no hay pruebas de que la solución no pueda ser arbitrariamente errónea en algunos casos; o se ejecuta razonablemente rápido, aunque no existe tampoco prueba de que siempre será así.

INFERENCIA: Capacidad de derivar nuevas sentencias a partir de las antiguas.

INGENIERO DEL CONOCIMIENTO: Es alguien que investiga un dominio concreto, aprende que conceptos son los importantes en ese dominio y crea una representación formal de los objetos y relaciones del dominio.

MARCOS: Propuestos por Minsky(1974). Permiten recopilar información sobre objetos concretos y tipos de eventos, organizando estos tipos en grandes jerarquías taxonómicas, similares a las biológicas.

MYCIN: Sistema experto diseñado por Feigenbaum, Buchanan y el doctor Edward Shortliffe para el diagnóstico de infecciones sanguíneas. Con 450 reglas aproximadamente, MYCIN era capaz de hacer diagnósticos tan buenos como los de un experto. Realizaba el cálculo de la incertidumbre con un elemento denominado factores de certeza.

PROBABILIDAD A PRIORI: O incondicional asociada a una proposición a , es el grado de creencia que se le otorga en ausencia de cualquier otra información y se le escribe como $P(a)$.

PROLOG: Es el lenguaje de programación lógica más usado. Los programas en Prolog son conjuntos de cláusulas positivas escritos en una notación algo diferente a la estándar de la lógica de primer orden.

PROTOTIPO: Es una versión preliminar de un sistema con fines de demostración o evaluación de ciertos requisitos. Es el objeto que tiene todas las características comunes. Literalmente significa el primer tipo.

PRUEBA DE TURING: Propuesta por Alan Turing(1950), se diseñó para proporcionar una definición operacional y satisfactoria de inteligencia. El computador supera la prueba si un evaluador humano no es capaz de distinguir si las respuestas a una serie de preguntas son de una persona o no.

RED BAYESIANA: Es una estructura de datos para representar las dependencias entre las variables y para mostrar una descripción escueta de cualquier distribución de probabilidad conjunta completa. Una red bayesiana es un grafo dirigido en el que cada nodo está comentado con información probabilista cuantitativa.

SISTEMA: Es una unión de partes o componentes, conectados en una forma organizada. Las partes se afectan por estar en el sistema y se cambian si lo dejan.

GLOSARIO DE TÉRMINOS MÉDICOS

COLPITIS: Inflamación, infección de la vagina.

DISPAREUMIA: Relación sexual dolorosa o coitalgia. Dolor en el área pélvica durante el acto sexual. Esto puede ocurrir tanto en hombres como mujeres. Se define como dolor o molestia antes, después o durante la relación sexual.

DISURIA: Del griego dus, mal y ouron, orina. Dificultad que se experimenta al orinar.

ENFERMEDAD: Proceso y fase que atraviesan los seres vivos cuando padecen una afección que atenta contra su bienestar caracterizado por una alteración de su estado de salud.

ERITEMA: Del griego, eruthema, rubicundez. Inflamación superficial de la piel. Nombre genérico de una serie de afecciones, caracterizada por enrojecimiento difuso o manchas de la piel causada por congestión de los capilares, que desaparece momentáneamente por la presión.

EXCORIACIÓN: Rasguño. Lesión cutánea superficial.

PAPULA: Tumor eruptivo en la piel. Pequeña eminencia eruptiva en la piel, que desaparece sin dejar cicatriz.

PATOGNOMÓNICO: Término que se utiliza para denominar aquellos signos (manifestaciones visibles) o síntomas (manifestaciones no visibles, subjetivas) que, si están presentes, aseguran que el sujeto padece un determinado trastorno.

PH: Es una medida de acidez o alcalinidad de una disolución. El pH indica la concentración de iones hidronio [H₃O⁺] presentes en determinadas sustancias. La sigla significa "potencial de hidrógeno" (pondus Hydrogenii o potentia Hydrogenii; del latín pondus, n. = peso; potentia, f. = potencia; hydrogenium, n. = hidrógeno). Este término fue acuñado por el químico danés Sorensen, quien lo definió como el logaritmo negativo en base 10 de la actividad de los iones hidrógeno.

PÚSTULA: Llega o absceso. Acumulación de pus en los tejidos orgánicos. Vesícula cutánea purulenta.

PROTOCOLO MÉDICO: Documento que describe la secuencia del proceso de atención de un paciente en relación a una enfermedad o estado de salud. Es el producto de una validación técnica que puede realizarse por consenso o por juicio de expertos.

PRURITO: Es un hormigueo o irritación incómoda de la piel que provoca el deseo de rascarse en el área afectada. Comezón, picazón.

SIGNOS: Son las manifestaciones objetivas, clínicamente fiables, palpables por el médico examinador.

SÍNDROME: Conjunto de síntomas que se producen de forma simultánea en una enfermedad, alteración o trastorno. Como ejemplo, podemos citar al síndrome de Down, el síndrome de Asperger, el síndrome de Estocolmo, etc.

SÍNTOMA: Referencia subjetiva que da un enfermo por la percepción o cambio que reconoce como anómalo, o causado por un estado patológico o enfermedad.

BIBLIOGRAFÍA

- [ALT1984] ALTY J.L. *Sistemas Expertos, conceptos y ejemplos*. Ed Diaz de Santos. Madrid Barcelona España.1984.
- [CAB1990] CABALLERO, Alejandro. *Metodología de la Investigación Científica*. Editorial Técnico Científica S.A. Segunda Edición. Lima 1990.
- [CAR1987] CARRILLO, José. *Metodología para el Desarrollo de Sistemas Expertos*. Tesis Doctoral. Facultad de Informática. Universidad Politécnica de Madrid. 1987.
- [CAS1997] CASTILLO, Enrique, GUTIERREZ José Manuel y HADI Ali S. *Expert Systems and Probabilistic Network Models*. Springer-Verlag. New York. 1997.
- [CAN2003] CANTU Marco. *La biblia de Delphi 7*. Editorial Anaya Multimedia – España.2003
- [CHA2000] CHARTE Francisco. *Programación con Delphi 5*. Editorial Anaya Multimedia – España.2000.
- [CHA2003] CHARTE Francisco. *Programación con Delphi 7 y Killix*. Editorial Anaya Multimedia – España.2003
- [ERT2011] ERTEL Wolfgang. *Introduction to Artificial Intelligence*. Springer-Verlag London. 2011.
- [FON2007] FONSECA F. Cleuzio, *Historia da Computacao: O Caminho do Pensamento e da Tecnologia*. EdiPUCRS. Porto Alegre. 2007.
- [FRO1989] FROST Richard, *Bases de Datos y Sistemas Expertos.Ingeniería del Conocimiento*. Ediciones Díaz de Santos S.A. Madrid. 1989
- [DAV1998] DAVIS William, YEN David. *The Information System Consultant's Handbook: Systems Analysis and Design*. CRC Press. USA. 1998.

- [GIA1998] GIARRATANO, Joseph and RILEY Gary. *Sistemas Expertos: Principios y Programación*. 3ra edición, México. International Thomson Editores. 1998.
- [HAR1988] HARMON Paul, KING David. *Sistemas Expertos. Aplicaciones de la Inteligencia Artificial en la actividad empresarial*. Ediciones Diaz de Santos S.A. – España. 1988.
- [JAC1990] JACOBS D, DEMOTT W. & OXLEY Dwight. *Laboratory Test Handbook*. Lexi-Comp Inc. 5th Edition. 1990.
- [KRA1986] KRAYTMAN, Maurice. *El diagnóstico a través de la historia clínica*. Internacional de Ediciones y Publicaciones S.A., 1986
- [KER1971] KERLINGER, Fred. *Investigación del comportamiento*. Editorial Interamericana. México, 1971.
- [KUL2000] KULIKOWSKI, Casimir. *Artificial Intelligence in Medical Decision Making: History, Evolution, and Prospects*. The Biomedical Engineering Handbook: Second Edition. Ed. Joseph D. Bronzino Boca Raton: CRC Press LLC, 2000
- [LAR1992] LARA R. Felipe. *Los Sistemas Expertos en la Tecnología*. En: Revista de la Coordinación de Estudios de Posgrado UNAM. Año 8. Número 25 Diciembre de 1992. [acceso 13 de octubre 2011] disponible en:
<http://www.posgrado.unam.mx/publicaciones/ant_omnia/25/03.pdf>
- [LEI2010] LEIVA, Jorge. *Sistema Experto KALM para el mantenimiento preventivo correctivo en la red telefónica peruana*. Tesis de Maestría en Ingeniería de Sistemas. UNI - Perú. 2010.

- [NIL1995] NILSSON Nils J. *Inteligencia Artificial, una Nueva Sintesis*. España. Editorial Mc Graw Hill Interamericana, 1995.
- [NEB1988] NEBENTHAL Dieter. *Sistemas Expertos. Introducción a la Técnica y Aplicación*. Marcombo Bolxareau editores. Barcelona, 1988.
- [NOR1995] NORAN Ovidiu S. *Expert Systems Design and Development. The Evolution of expert Systems*. Griffith University, 2000. [acceso 19 de noviembre 2011] disponible en:
<<http://www.ict.griffith.edu.au/noran/Docs/ES-Evolution.pdf>>
- [OBL1987] OBLITAS, Luis. *Metodología de la Investigación*. Biblioteca Peruana de Psicología. Lima, 1987.
- [PAJ2006] PAJARES M Gonzalo, SANTOS P Matilde. *Inteligencia Artificial e Ingeniería del Conocimiento*. Editorial AlfaOmega, Tercera Edición. México, 2006.
- [PON2010] PONCE C Pedro. *Inteligencia Artificial con Aplicaciones a la Ingeniería*. México. Editorial AlfaOmega, 2010
- [RAB1995] RABUSKE, Renato Antonio. *Inteligencia Artificial*. Brasil, Florianópolis. Editora de la UFSC, 1995
- [RIC1993] RICH Eliane; KNIGHT Kevin. *Inteligencia Artificial*. 2da Edición. Editorial Sao Paulo: Makron Books, 1993.
- [RUS1996] RUSSELL, Stuart y NORVIG Peter. *Inteligencia Artificial: Un enfoque moderno*. México. Prentice Hall Hispanoamericana S.A. 1996.

- [SAJ2010] SAJJA Priti Srinivas, AKERKAR Rajendra. *Knowledge - Based Systems for Development*. En: *Advanced Knowledge Based Systems: Model, Applications and Research*. (Eds. Sajja & Akerkar), Vol. 1, pp 1 – 11, 2010. [acceso 16 de noviembre 2011] disponible en:
<<http://www.tmrfindia.org/eseries/ebookv1-c1.pdf>>
- [TIE2001] TIERNEY, Lawrence; McPHEE Stephen y PAPADAKIS Maxine. *Current Medical Diagnosis & Treatment*. 41th Edition. Mc Graw Hill Companies, 2001
- [TUR1950] TURING Allan. *Computing Machinery and Intelligence* [archivo de internet] *Mind* 49: 433 - 460. [acceso 13 de octubre 2011] disponible en:
<<http://www.csee.umbc.edu/courses/471/papers/turing.pdf>>
- [VAQ1998] VAQUERO Antonio, QUIROZ Gerardo. *Microsoft Visual FoxPro 6.0: Manual del Programador*. 2da Edición. Mc Graw Hill / Interamericana. España. 1998.
- [VIL2003] VILCHEZ Ricardo. *Aplicación de los Sistemas Expertos en el sector pesquero*. Tesis de Maestría en Ingeniería de Sistemas. UNI, Perú. 2003.
- [WEI1984] WEISS, Sholom M. KULIKOWSKI, Casimir A. *A Practical Guide to Designing Expert Systems*. New Jersey: Rowman & Allanheld Publishers, 1984.
- [WIN1994] WINSTON Patrick H. *Inteligencia Artificial*. Editorial Addison Wesley Iberoamericana. Tercera Edición. EUA. 1994

BIBLIOGRAFÍA COMPLEMENTARIA

BARCELO Miquel. *Inteligencia Artificial*. 2004. [libro virtual] [Acceso Noviembre 2011] disponible en web:

<http://books.google.com.pe/books?id=GCFi-TeQmZ4C&pg=PA20&lpg=PA20&dq=representacion+del+conocimiento+logica+redes+sem+anticas&source=bl&ots=UgH_HXiLe3&sig=YvcW5JfJvO-HZSSX9_7kL4gzqg&hl=es&sa=X&ei=mo9NT_KSOYzrggeZ1pWbAg&ved=0CFAQ6AEwCDgo#v=onepage&q&f=false>

CYWARLIE Carolina, SAN ROMÁN María y UCCELLO Mónica. *Inteligencia Artificial en la Medicina*. 2004. [archivo de internet] [Acceso Octubre 2011] disponible en web:

<<http://galeon.com/iaenlamedicina/menu.html>>

MARECOS Edgardo. *Revista de Posgrado de la VI Cátedra de Medicina*. N° 128 - Junio/2003. Página: 01 - 05. [archivo de internet] [Acceso Octubre 2011] Disponible en web:

<http://med.unne.edu.ar/revista/revista128/diag_dif.htm>

PEARL Judea. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. [Libro virtual] [Acceso febrero 2012] disponible en web:

<http://books.google.com.pe/books?id=AvNID7LyMusC&printsec=frontcover&hl=es&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false>

RIZZI, Francisco. Sistema Experto Asistente de Requerimientos. Instituto Tecnológico Buenos Aires - ITBA. Buenos Aires. Argentina. 2001. [archivo de internet] [Acceso Agosto 2011] disponible en web:

<<http://iidia.com.ar/rgm/tesistas/rizzi-tesisdemagister.pdf>>

ROMERO Mariluz, RODRÍGUEZ Jorge. *Sistemas Basados en el Conocimiento*. [archivo de internet] [Acceso Abril 2012]. Revista Vínculos. 1ra Edición. Universidad Distrital Francisco José de Caldas. Colombia. Disponible en web:

<<http://gemini.udistrital.edu.co/comunidad/dependencias/revistavinculos/vinculos/revista/1edicion/12004104.pdf>>

SAN JUAN Aitor. *Sistema CLIPS*. [archivo de internet] [Acceso Mayo 2011]. Nro Págs 100. Disponible en web:

<http://www.fdi.ucm.es/profesor/belend/ISBC/clips_archivos/clips.pdf>

THE COMPUTER CHRONICLES. Artificial Intelligence. [Video de internet] [Acceso Enero 2012]. Disponible en web:

http://archive.org/details/CC1024_artificial_intelligence

WORDPRESS. Expert Systems. Unit 09. [archivo de internet] [Acceso setiembre 2011] disponible en web:

<<http://train-srv.manipalu.com/wpress/?p=119855>>

X-PLAIN. *Enfermedades de Transmisión Sexual*. Programa de Educación al Paciente. The Patient Education Institute. Medline Plus. 1995-2012. [archivo de internet] [Acceso abril 2012] disponible en web:

<http://www.nlm.nih.gov/medlineplus/spanish/tutorials/sexuallytransmitteddiseasesspanish/htm/_no_50_no_0.htm>

ANEXO A

EXTRACTO DE UNA ENTREVISTA A UN ESPECIALISTA

Dr. Eddie A. Angles Y.

Médico Infectólogo/Tropicalista

Hospital Nacional Arzobispo Loayza (HNAL)

Fecha: Agosto del 2011

Doctor, ¿Hay una diferencia entre ETS e ITS?

Antes se llamaban enfermedades de transmisión sexual y ahora ya cambió de término ahora se dicen infecciones de transmisión sexual, ITS, el MINSA hace las publicaciones usando ese término, pero en algunas referencias se mantienen aun como ETS. Las ITS son causadas por bacterias.

¿Qué diferencia hay entre síntoma y signo clínico?

Síntoma es lo que el paciente te refiere, como el dolor de cabeza, yo no lo puedo ver, es subjetivo. Muchas veces lo que dice el paciente no se puede ver. Por ejemplo en el prurito, el paciente manifiesta que tiene picazón pero no tengo forma de constatarlo.

Signo es lo que yo veo, es algo objetivo ...si me dicen: doctor me duele la pierna, a ver muestre y si efectivamente tiene una lesión y yo describo eso. Un ejemplo de signo en el flujo vaginal es la descarga espumosa, de color amarillenta.

¿Qué es el diagnóstico médico?

Es lo que en base a signos, síntomas yo concluyo en algo. A mí me parece que es tal enfermedad... El tratamiento viene después.

¿Qué es la vulvovaginitis?

Vulvovaginitis es una inflamación. Tiene parásitos, tiene bacterias, tiene virus, etc.

¿Es lo mismo hablar de vulvovaginitis que de síndrome de flujo vaginal?

No es lo mismo. El síndrome de flujo vaginal es un enfoque más específico. Son determinados gérmenes, sólo cubre bacterias y hongos. En el síndrome de flujo vaginal el tratamiento que se le da al paciente es de nivel de atención 1.

¿Qué características las patologías que comprenden el síndrome de flujo vaginal?

Que todas presentan como característica común el descenso o flujo vaginal, las característica es que todas hagan flujo.

¿En cuanto a las probabilidades existe ya un estándar para asociar síntomas a enfermedades?

Si hay ... digamos en el caso de los síndromes, por ejemplo en la descarga uretral hay varios trabajos. ¿Qué síntomas tiene?. Que tenga secreción por la uretra es un síntoma. ¿Qué porcentaje se presenta en todos los pacientes que se ha estudiado?: 98%. Tiene dolor, ...¿en que se presentan?: 95%... Arde, ardor se produce menos ,90, ... hace fiebre, ... será menos, 50, esos datos si hay, en porcentaje, pero es particular, hay que buscar uno por uno. En las normas técnicas probablemente haya algo de eso, habría que revisar.

No te detengas! ... si te caes, levántate pero no te distraigas hasta llegar a la meta. Cumple tus sueños.