Lógica no Monotónica Un nuevo Modelo Inferencial para Aplicaciones Informáticas

Manuel Ramiro Flores Rojas
Postgrado en Informática
Universidad Mayor de San Andrés - UMSA
La Paz, Bolivia
fr_ramiro@yahoo.com

***Resumen*— Lógica es la ciencia que le proporciona el fundamento necesario a las aplicaciones informáticas y a la informática en su conjunto. Lógica es considerada como un lenguaje formal, un lenguaje simbólico; dispone de un sistema axiomático y un sistema inferencial, que permite derivar conclusiones a partir de un conjunto de premisas consideradas como verdaderas.**

En las aplicaciones informáticas se emplea lógica clásica: lógica de proposiciones y lógica de predicados. Sin embargo, una característica de la lógica clásica es su carácter monotónico, por lo cual presenta limitaciones en razonamientos en los que no se cuenta con información completa o que se halla comprometida con el entorno. En la realidad, el conocimiento es *no monotónico*, puesto que cada nuevo conocimiento que se adquiere en un razonamiento modifica e incluso puede invalidar los conocimientos previos que se tenía; el conocimiento cambia, evoluciona permanentemente.

En el presente trabajo se desarrollará el sistema axiomático e inferencial de una nueva lógica, una lógica de sentido común, una lógica que razone con el entorno: *una lógica no monotónica*.

Palabras clave— Lógica, monotónica, lenguaje formal, sistema axiomático, sistema inferencial, aplicaciones informáticas.

I. INTRODUCCIÓN

Informática es la ciencia que estudia el tratamiento automático y racional de la información. La diversidad de aplicaciones informáticas se desarrollan siguiendo criterios racionales, lógicos; y es precisamente la lógica la ciencia que les proporciona el fundamento necesario. El caso más objetivo son las aplicaciones de la Inteligencia Artificial; sin embargo, la informática en su conjunto encuentra en la lógica su fundamento.

Aristóteles, considerado como el precursor de la lógica, definió a la misma como el *estudio de la argumentación correcta y verdadera*. Otras definiciones más actuales y que tengan relación al ámbito informático, indican que "*lógica es una disciplina que estudia los métodos de formalización del conocimiento humano", "lógica es la ciencia de los principios de la inferencia formalmente válida"* (Alfredo Deaño).

Se identifica a la lógica como el estudio de las relaciones *inferenciales* emergentes de premisas a *conclusiones*, proceso denominado como *deducción*. La lógica deductiva es el área más desarrollada de esta ciencia.

En el desarrollo de aplicaciones informáticas generalmente se emplea lógica clásica: *lógica de proposiciones y lógica de predicados.* La gran limitante de la lógica clásica, es su carácter *monotónico*, según esta característica, si a consecuencia de cambios en el entorno agregamos un nuevo conocimiento o sentencia a un razonamiento, esta incorporación solo refuerza pero no modifica su conjunto inicial de sentencias; es decir, no existe afectación por los cambios que puedan producirse en el entorno.

Un razonamiento es *no monotónico* cuando es *retractable* o puede modificarse por los cambios que se producen en su entorno en tanto exista un aumento de información. En la realidad el conocimiento es *no monotónico*, cada nuevo conocimiento que se adquiere modifica e incluso puede invalidar los conocimientos previos que se tenía; el conocimiento cambia, evoluciona.

En el presente trabajo se desarrollará el sistema axiomático e inferencial para una lógica no monotónica, una lógica de sentido común, cuyas conclusiones y razonamientos sean retractables, revocables por la incorporación de nueva información; de tal manera que pueda ser empleado en

aplicaciones informáticas que tengan la capacidad de adecuarse a los cambios que puedan producirse en su contexto.

II. PROBLEMA

PLANTEAMIENTO DEL PROBLEMA

El sistema inferencial de la lógica clásica no permite que las conclusiones de un argumento o razonamiento puedan retractarse o modificarse como consecuencia de la incorporación de nueva información; esta lógica es insatisfactoria y no aplicable a intuiciones acerca del razonamiento natural, y la informática ha tenido que ajustarse a esta limitante. En muchas aplicaciones informáticas se evidencia la necesidad de interactuar con el entorno, y es necesario desarrollar una nueva lógica, una lógica no monotónica, que posibilite un cambio en el modelo inferencial, de tal manera que permita a dichas aplicaciones la retractación en sus conclusiones y razonamientos por cambios en su entorno.

FORMULACIÓN DEL PROBLEMA

¿De qué manera las conclusiones e inferencias de los argumentos emergentes de las aplicaciones informáticas pueden retractarse o modificarse como consecuencia de la incorporación de nueva información o por cambios en su contexto?

III. PLANTEAMIENTO DEL OBJETIVO

OBJETIVO GENERAL

Posibilitar que las conclusiones de los argumentos emergentes de las aplicaciones informáticas puedan retractarse o modificarse como consecuencia de la incorporación de nueva información o por cambios en su contexto, mediante el desarrollo de un sistema axiomático y un sistema inferencial de una lógica no monotónica.

OBJETIVOS ESPECÍFICOS

- Establecer las limitaciones de la lógica clásica en la formalización de los razonamientos que se presentan en la realidad.
- Definir las ventajas del empleo de un sistema inferencial basado en una lógica no monotónica en los razonamientos.
- Establecer un sistema axiomático para una lógica no monotónica.

IV. HIPÓTESIS

La presente investigación no es experimental y no trabaja con datos cuantitativos, lo que pretende es desarrollar un nuevo sistema axiomático e inferencial, en el que la investigación parte de una pregunta de trabajo y no de una hipótesis.

V. MARCO TEÓRICO

LÓGICA

Aristóteles (384 adC - 322 adC) considerado como el padre fundador de la lógica, utilizó el término lógica para referirse al estudio de los argumentos dentro del lenguaje natural. Aristóteles define la lógica como "*el arte de la argumentación correcta y verdadera*", y constituye la primera investigación sistemática acerca de los principios del razonamiento válido.

Otras definiciones de mayor interés para el desarrollo del presente trabajo indican que la lógica es la disciplina que estudia los métodos de formalización del conocimiento humano (Sperschneider & Antoniou, 1991). Lógica es la ciencia de los principios de la inferencia formalmente válida (Alfredo Deaño).

LÓGICA COMO UN SISTEMA FORMAL

La lógica formal es considerada como un lenguaje formal "*el mejor hecho de los lenguajes*", (Ferrater Mora). Un lenguaje es un sistema de símbolos y de convenios que se utiliza para la comunicación, sea ésta entre personas, entre personas y máquinas o entre máquinas. Un lenguaje formal es un sistema simbólico que se construye a partir de un alfabeto, y un conjunto de reglas, reglas para la formación de fórmulas bien formadas, las que nos permite construir las palabras o cadenas del lenguaje, que luego deben ser interpretadas.

LENGUAJE FORMALES: DEFINICIONES Y CONCEPTOS BÁSICOS

Las palabras de los lenguajes inglés, español, o de programación, y en general de cualquier lenguaje artificial formal, se desarrollan a partir de los siguientes conceptos:

ALFABETO, PALABRA (CADENA) Y LENGUAJE.

Alfabeto.

El alfabeto denotado por el símbolo $\sum$, se define como el conjunto finito no vacío de símbolos indivisibles u objetos atómicos.
Ejemplo:

$\sum_1$ = {A, B, C, ……Z, a, b, c, …….z}
$\sum_2$ = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}
$\sum_3$ = {0, 1}
$\sum_4$ = {Conjunto de palabras reservadas y símbolos legales de un lenguaje de Programación}

Palabra.

Una palabra, también denominada cadena, sobre el alfabeto $\sum$, es una secuencia finita, ordenada, con o sin repetición, de símbolos del alfabeto en cuestión.
Ejemplo:

Palabras sobre $\sum_1$: Universidad, Bolivia,
Palabras sobre $\sum_2$: 2011, 17475693
Palabras sobre $\sum_3$: 1010, 110110110

Las cadenas se representan con letras minúsculas ***x, y, z, u, v, w***.

Palabra o cadena vacía.

La cadena vacía, denota por el símbolo λ, se define como la *cadena que no tiene símbolos*. Esta cadena tiene propiedades al momento de definir operaciones sobre cadenas.

OPERACIONES SOBRE CADENAS.

*i) Longitud***.** La longitud de la cadena *u*, denotado como $|u|$, se define como la cantidad o número de símbolos que tiene la cadena.
Ejemplo. Consideremos las cadenas *u, x*:

u = Universidad $|u| = 11$
$v = 2011$ $|v| = 4$
$w = \lambda$ $|w| = 0$

ii) Igualdad. Las cadenas *u, v* son iguales si tienen la misma longitud, los mismos símbolos y en idénticas posiciones, y se denota como $u = v$.

iii) Concatenación. La concatenación de la cadena *u* con la cadena *v*, denotado por $u \cdot v$ (o simplemente *uv*), es la cadena que resulta de escribir los símbolos de *v* al lado derecho de los símbolos de *u*.
Ejemplo. Sean *u, v* cadenas:
u = Inteligencia
v = Artificial
Su concatenación es: $u \cdot v$ = InteligenciaArtificial

Propiedades de la concatenación: Sean *u, v, w* cadenas definidas sobre un alfabeto.

- Asociatividad $(u \cdot v) \cdot w = u \cdot (v \cdot w) =$
- Existe elemento neutro $u \cdot \lambda = \lambda \cdot u = u$
- En general, no es conmutativa $u \cdot v \neq u \cdot v$
- Ninguna cadena (excepto λ) tiene inverso, es decir, dada una cadena *x*, no existe ninguna cadena *y* tal que $x \cdot y = \lambda$

Dadas estas propiedades, se concluye que la operación de concatenación definida sobre un alfabeto cualquiera, define una *estructura algebraica de tipo monoide*, a la que se denomina monoide libre generado por el alfabeto.

iv) Inversa. La inversa o transpuesta de la cadena *u*, denotada por u^I, es la imagen refleja de *u*

Si u = abcd , u^I = dcba

v) Potencia. Sea *u* una cadena, y sea **n** un numero natural. La potencia **n**-enésima de *u*, denotada como u^n se define como:

$u^n = \lambda$, si $n = 0$
$u^n = u \cdot u^{n-1}$, si $n \geq 1$

Es decir, la concatenación de la cadena *u* consigo misma **n** veces.

vi) Subcadena. La cadena *w* es una subcadena de la cadena *z* si existen la cadena *x* y *y* talque:

$z = x \cdot w \cdot y$

Ejemplo. Sea z = abcde
w = cd x = ab y = e

$z = x \cdot w \cdot y$
z = ab·cd·e

OPERACIONES SOBRE ALFABETOS

Sea $\sum$ un alfabeto, y sea **n** un número natural. $\sum^n$ es el conjunto de todas las cadenas de longitud **n** que pueden formarse con los símbolos de $\sum$.

Ejemplo. Sea $\sum$ = {a, b}
$\sum^0$ = {λ} Cadenas de longitud 0
$\sum^1$ = {a, b} Cadenas de longitud 1
$\sum^2$ = {aa, ab, ba, bb} Cadenas de longitud 2
$\sum^3$ = {aaa, aab, aba, abb, baa, bab, bba, bbb}

Lenguaje universal. El lenguaje universal sobre el alfabeto $\sum$, denotado por $\sum^*$ es el conjunto de todas las cadenas, incluida λ, que pueden formarse a partir de $\sum$. Formalmente, es la unión infinita siguiente:

$\sum^* = U_{i=1}^{\infty} = \sum^0 \cup \sum^1 \cup \sum^2 \cup \ldots\ldots$

Ejemplo. Sea $\sum$ = {a, b}

$\sum^*$ = { λ, a, b, aa, ab, ba, bb, aaa, aab, aba,
abb, baa, bab, bba, bbb, aaaa, ….}

LENGUAJE.

Un lenguaje, denotado como **L**, es todo subconjunto del lenguaje Universal, puede ser vacío o infinito.

$L \subseteq \sum^*$

Ejemplo. Sea. $\sum$ = {a, b}

L = {aa, aba, abba, abbba, …..} = {$ab^n a$ / $n \geq 0$}
L = {aba, abba, abbba, ……} = {$ab^n a$ / $n > 0$}
L = {λ, ab, aab, aabbbb, aaaabbbb, …} = {$a^m b^n$ / m, $n \geq 0$}
L = {} = L_0 es el lenguaje vacío, no tiene palabras
L = { λ } Solo tiene la cadena vacía

Para mayores referencias consultar los textos [3], [4] y [5].

La descripción de un lenguaje artificial suele hacerse mediante unas reglas que permiten generar cadenas

(sentencias, frases o expresiones válidas) pertenecientes al lenguaje. Son las reglas de formación, que en otros contextos formales también se los denomina reglas de escritura o reglas gramaticales.

En un sistema axiomático formal, la formulación de los axiomas debe satisfacer lo siguiente:

- El sistema resultante debe ser ***completo***, es decir, todas las sentencias verdaderas que el sistema pretende formalizar tienen que poder demostrarse a partir de los axiomas definidos.

- El sistema debe ser ***consistente***, es decir, que no se pueden demostrar sentencias no verdaderas.

Finalmente, los axiomas deben ser independientes, es decir, ninguno de ellos puede demostrarse a partir de los restantes. Además, los teoremas del sistema axiomático constituyen la base de las reglas que permiten realizar inferencias.

LÓGICA DE PROPOSICIONES

Una proposición es un enunciado declarativo que puede ser o verdadero o falso, es decir, una frase expresada en el modo gramatical indicativo. Las proposiciones se representan a través de variables proposicionales, y para denotarlas emplearemos las letras ***p, q, r, s, t,….***

Una sentencia representa a un enunciado compuesto por enunciados elementales y vinculadas por las conectivas de conjunción (**∧**), disyunción (**∨**), condicional ($\rightarrow$) y bicondicional ($\leftrightarrow$).
Combinando adecuadamente variables proposicionales y conectivas se forman sentencias o cadenas válidas del lenguaje de la lógica.

Interpretación binaria de variables proposicionales y de sentencias.

Una interpretación binaria consiste, en asignar a cada una de las variables proposicionales uno de entre dos valores: "verdadero" o "falso", representado por los símbolos "1" y "0" respectivamente. Para poder dar una interpretación a la sentencia empleamos la siguiente tabla en la que se establece un significado a las conectivas:

	p	q	$\neg p$	$p \wedge q$	$p \vee q$	$p \rightarrow q$	$p \leftrightarrow q$
i_0	0	0	1	0	0	1	1
i_1	0	1	1	0	1	1	0
i_2	1	0	0	0	1	0	0
i_3	1	1	0	1	1	1	1

SINTAXIS

Alfabeto. Está formado por los siguientes símbolos:

- *Variables proposicionales: p. q. r, s, t...*
- *Conectivas:*'¬, ∧, v, $\rightarrow$, $\leftrightarrow$',
- *Símbolos de puntuación:* (,), [,], {, }.
- *Meta símbolos.* No son, propiamente, símbolos del lenguaje. Se emplean para abreviar ciertas expresiones, y serán los siguientes:
  - *A, B, C,…* para representar cualquier sentencia del lenguaje.
  - *k* para representar cualquier conectiva.
  - *l* para representar un *literal* (variable proposicional sola o negada).

Expresiones y Sentencias.

El concepto de sentencia se define recursivamente a través de las siguientes tres reglas de formación:

RF1: Una variable proposicional es una sentencia.
RF2: Si ***A*** es una sentencia, ¬ ***A*** también lo es.
RF3: Si ***A*** y **B** son sentencias, ***A k B*** también lo es.

En consecuencia, $(\neg p \vee q) \rightarrow \neg r$ es una sentencia.

Axiomas, teoremas.

La lógica se emancipa de la filosofía y se constituye en una ciencia completamente formalizada con los aportes de los investigadores Frege y Giussepe Peano, a los cuales se complementa Russell y Whitehead quienes desarrollan la obra clásica titulada "Principia Mathematica" (PM), cuyas tesis esenciales le exigen a la lógica asegurar los fundamentos matemáticos necesarios. Desde entonces se denomina lógica clásica a todo sistema lógico equivalente al formulado en esta obra. El PM emplea cuatro axiomas.

A1. $(p \vee p) \rightarrow p$
A2. $q \rightarrow (p \vee q)$
A3. $(p \vee q) \rightarrow (q \vee p)$
A4. $(p \rightarrow q) \rightarrow [(r \vee p) \rightarrow (r \vee q)]$

Con estos axiomas y unas reglas de transformación se demuestran teoremas como los siguientes:

Teorema 1. $(p \rightarrow q) \rightarrow [(r \rightarrow p) \rightarrow (r \rightarrow q)]$
Teorema 2. $p \rightarrow (p \vee p)$

Teorema 3. $p \rightarrow p$
Teorema 4. $\neg p \vee p$

SEMÁNTICA

Interpretación binaria.

Formalmente, es una interpretación $\langle i, V \rangle$ o su extensión $\langle I, V \rangle$, en la que V tiene dos elementos, que denominaremos "0" y "1" y, cinco operaciones definidas en este conjunto:

$\neg (0) = 1; \quad \neg (1) = 0$
$0 \wedge 0 = 0; \quad 0 \wedge 1 = 0; \quad 1 \wedge 0 = 0; \quad 1 \wedge 1 = 1$
$0 \vee 0 = 0; \quad 0 \vee 1 = 1; \quad 1 \vee 0 = 1; \quad 1 \vee 1 = 1$
$0 \rightarrow 0 = 1; \quad 0 \rightarrow 1 = 1; \quad 1 \rightarrow 0 = 0; \quad 1 \rightarrow 1 = 1$
$0 \leftrightarrow 0 = 1; \quad 0 \leftrightarrow 1 = 0; \quad 1 \leftrightarrow 0 = 0; \quad 1 \leftrightarrow 1 = 1$

Tautologías y contradicciones.

Una sentencia A es una *tautología* si $(\forall i_j)\ (I_j(A) = 1)$. Se emplea la notación "$\models A$" para indicar que "A es una tautología".

Una sentencia A es una *contradicción* si $(\forall i_j)\ (I_j(A) = 0)$.

Completitud y consistencia del sistema axiomático.

Un sistema axiomático es *completo* si toda sentencia A que sea una tautología es también una tesis. Es decir, para toda A, si $\models A$, entonces $\vdash A$.

Un sistema axiomático es *consistente* si toda sentencia A que sea una tesis es también una tautología. Es decir, si $\vdash A$, entonces $\models A$.

De esta manera, el sistema PM, con la interpretación binaria que hemos definido, es completo y consistente, puesto que, sus axiomas o teoremas son tautologías.

Inferencia.

Se denomina *inferencia* a los procesos mediante los cuales obtenemos una conclusión a partir de unas premisas de modo tal que el razonamiento es válido. Una *regla de inferencia* será la declaración de las condiciones bajo las cuales puede hacerse una inferencia.

Para el logro del propósito, elegimos una tesis que tenga la forma $A_1 \wedge A_2 \wedge \ldots \rightarrow B$, de tal manera que el antecedente se ajuste exactamente a una premisa o a la conjunción de dos o más de ellas; aplicándolo, se obtiene como conclusión el consecuente de la tesis, que se añade al conjunto de premisas, y repetir el proceso hasta que ya no puedan obtenerse más conclusiones.

En este caso, ¿qué conclusión obtenemos a partir de las siguientes premisas?:

P_1: $p \rightarrow \neg q$

P_2: $\neg q \rightarrow r$

Se observa que esta sentencia es un teorema en forma de condicional que tiene como antecedente la conjunción de P_1 y P_2, es decir, $(p \rightarrow \neg q) \wedge (\neg q \rightarrow r)$ podemos afirmar como conclusión el consecuente:

$$C: \ p \rightarrow \boldsymbol{r}$$

Leyes y reglas de inferencia.

A toda tesis (o ley) del cálculo proposicional que tenga la forma $P_1 \wedge P_2 \wedge \ldots \rightarrow C$ puede hacérsele corresponder una regla de inferencia.

Por ejemplo, consideremos la Ley del modus ponens para las sentencias A y B,

$$[A \wedge (A \rightarrow B)] \rightarrow B$$

Su correspondiente regla de inferencia se expresa de la siguiente manera:

"De A y de $A \rightarrow B$ puede inferirse B"

En la sintaxis definida, "$\rightarrow$" no es un símbolo que denote inferencia, por lo cual, en la simbolización de la regla de inferencia, cada condición se escribe en una línea, y la conclusión en una línea final, bajo una línea horizontal.

$$\begin{array}{r} A \\ A \rightarrow B \\ \hline B \end{array}$$

De este modo, no existen dificultades para escribir en esta forma otros teoremas o leyes de la lógica.

Una interpretación i_j *satisface* a una sentencia A si (y sólo si) $I_j(A) = 1$.

Definición. Una sentencia C (conclusión) *se deduce de* un conjunto de sentencias $P_1, P_2, \ldots, P_n$ (premisas) si (y sólo si) toda interpretación que satisface $P_1 \wedge P_2, \wedge \ldots \wedge Pn$ también satisface C.

Completitud y consistencia del sistema inferencial.

Un sistema inferencial es *completo* si, para cualquier conjunto de premisas, el sistema infiere toda conclusión que pueda deducirse de las premisas.

Un sistema inferencial es *consistente* si, para cualquier conjunto de premisas, toda conclusión que infiera el sistema también se deduce de las premisas.

Teorema. Todo sistema inferencial cuyas reglas de inferencia se puedan formalizar como tesis de un sistema axiomático consistente es un sistema inferencial consistente.

Como puede observarse, el sistema inferencial de la lógica de proposiciones es *completo y consistente*.

La lógica de predicados es una generalización de la lógica de proposiciones. Cuenta con una sintaxis, una semántica y un sistema inferencial, que son completos y consistentes.

Para mayores referencias consultar los textos [1] y [2].

VI. Solución Propuesta

A partir de este estudio y su correcta comprensión iniciaremos la explicación científica basado en lógica no monotónica, y la formalización de un nuevo sistema axiomático que posibilite un cambio en el modelo inferencial, una nueva mirada de racionalidad y una ampliación de la deductibilidad clásica.

Las lógicas no monotónicas son aquellas que partiendo de los criterios establecidos en la lógica clásica pretenden dar una explicación a los argumentos que se encuentran comprometidos denodadamente con el contexto, evidenciando que a pesar de que una lógica sea completa y consistente de manera formal, no implica que cuente con la capacidad para representar y trabajar con argumentos retractables, derrotables o incompletos.

No obstante de que la lógica tiene un historial de más de dos siglos, el estudio de la lógica no monotónica es reciente, y es emprendido por los científicos e investigadores de la Inteligencia Artificial que comenzaron a gestar la idea de simular el razonamiento humano en autómatas y estudiar los procesos del razonamiento humano a través de la psicología cognitiva. En estas investigaciones evidencian la necesidad de un estudio capaz de permitir la realización de tales operaciones de modo que sean computables, por lo que se busca en la lógica una respuesta, habida cuenta de que la lógica abarca el estudio de la reproducción formal del razonamiento deductivo.

Gladys Palau expone en *Lógica Natural e Inteligencia Artificial* una presentación completa a cerca del desarrollo de las lógicas no monotónicas en relación a la Inteligencia Artificial.

Dov M. Gabbay estableció por primera vez las propiedades generales del razonamiento de sentido común el año 1985 en su trabajo *Theoretical foundations for non Monotonic reasoning in expert systems*.

S. Krauss D. Lehmann y M. Magidor, el año 1990 presentan la obra titulada *Non Monotonic Reasoning, Preferential Models and Cumulative logics*, en el cual se brinda nociones de consecuencia involucrada en los argumentos.

La lógica no monotónica se desarrollará a partir de la lógica clásica a la cual se añade una ***relación de consecuencia no monótona.***

David Makinson, el año 1984 presentan la versión más completa de los trabajos sobre el tema, en su trabajo titulado *General Patterns in Nomonotonic Reasoning*, y en una obra más reciente del año 2005, *Bridges from Classical Non monotonic logic*, formula las diversas formas de consecuencia no monótona de los formalismos tradicionales, estableciendo las relaciones entre ellas y analizando las conexiones entre no monotonía y la probabilidad lógica.

Los sistemas de lógica no monotónica se construyen a partir de la lógica clásica añadiendo un signo para la relación de consecuencia no monótona $|\approx$ y el siguiente conjunto de propiedades básicas o elementales:

i) Reflexibilidad (R).
$A |\approx A$

ii) Corte o transitividad cumulativa
$A |\approx B,\ A \wedge B |\approx C$, entonces $A |\approx C$

iii) Monotonía cautelosa o cumulativa
$A |\approx B, A |\approx C$, entonces $A \wedge B |\approx C$

Es posible agregar otras reglas, que en su conjunto formarán distintas nociones de consecuencia lógica no monotónica.

De acuerdo a estos criterios, se desarrollará un sistema axiomático que sea completo y consistente, además de un sistema inferencial que cumpla estas características

Referencias

[1] W. Grassmann y J.P. Tremblay, Matemática discreta y lógica, Prentice Hall, 2001, Madrid.
[2] Gregorio Fernández, Fernando Sáez Vaca, Fundamentos de Informática, Alianza Editorial, Madrid, 1987.
[3] Introducción a la teoría de autómatas, lenguajes y computación, John E. Hopcroft, Rajeev Motwani, Jeffrey Ullman, Adelison Wesley, Madrid, 2002.
[4] Dean Kelley, Teoría de autómatas y lenguajes formales, Prentice Hall, España.
[5] Nonmonotonic Reasoning: Logical Foundations of Commonsense, Brewka, Gerhard, Cambridge University Press.
[6] A Logical Framework for Default Reasoning. Artificial Intelligence, Poole David.
[7] Computational Intelligence. A Logical Approach, Poole David, Alan Mackworth, Randy Goebel, Nueva York-Oxford, Oxford University Press.

[8] An Approach to Default Reasoning Based on a First Order Conditional Logic: Revised Report. En Artificial Intelligence, Delgrande, 1988.
[9] Gladys Palau, *Lógica Natural e Inteligencia Artificial*.
[10] Dov M. Gabbay, *Theoretical foundations for non Monotonic reasoning in expert systems*.
[11] S. Krauss D. Lehmann y M. Magidor, *Non Monotonic Reasoning, Preferential Models and Cumulative logics*.