Escuela Técnica Superior de Ingeniería Informática

Asignatura:
Lógica Informática

Autor:
Fernando José Mateos Gómez

Ultima Modificacion: **28 de Enero del 2022**

Indice

1. Tema 1: Sintaxis y semántica de la lógica proposicional Internet

1.1. Introducción

Cualquier sentencia la podemos descomponer en un lenguaje formal del cual podemos analizar su veracidad y su sentido, sin error a no comprenderlo. Para lograr esto debemos seleccionar las ideas y dividirlas en proposiciones, para así poder juntarlas.

1.2. Sintaxis básica

Usaremos 5 operadores para replicar el sentido de las sentencias en proposiciones:

- $\neg$: Operador de negación.
- $\wedge$: Operador de conjunción.
- $\vee$: Operador de disyunción.
- $\rightarrow$: Operador "implica que...".
- $\leftrightarrow$: Operador "si y solo si".

Podemos eliminar los paréntesis de una expresión, pero para mantener el sentido original, debemos de respetar una regla, además de no quitar los parentesis que violen parte del sentido original, y es colocar o poner los paréntesis en función del orden establecido arriba en la lista.

1.2.1. Fórmulas Proposicionales

$$(\neg p \rightarrow (\neg(q \wedge t) \vee p)) \leftrightarrow \neg q$$

Esta expresión es una fórmula proposicional, hemos añadido los paréntesis para mejorar su legibilidad.

1.3. Árboles de análisis y Subfórmulas

1.3.1. Árboles de análisis

Dada la expresión de arriba podemos dividirla en segmentos para despedazarla en subfórmulas:

$$\begin{array}{cc}
\neg p \rightarrow (\neg(q \wedge t) \vee p) & \neg q \\
\neg p \quad \neg(q \wedge t) \vee p & q \\
p \quad \neg(q \wedge t) \;\; p & \\
q \wedge t & \\
q \quad t &
\end{array}$$

Hemos dividido la expresión en partes.

1.3.2. Subfórmulas

Una subfórmula no es más que una proposición compleja, o no, que forma parte de la fórmula. Por ejemplo, q es una expresión atómica y $\neg(q \wedge t$ es una subfórmula compleja.

1.4. Tablas de Verdad

$I_n(T)$	F	G	$\neg F$	$\neg G$	$F \wedge G$	$F \vee G$	$F \rightarrow G$	$F \leftrightarrow G$
I_0	1	1	0	0	1	1	1	1
I_1	0	0	1	1	0	0	1	1
I_2	1	0	0	1	0	1	0	0
I_3	0	1	1	0	0	1	1	0

Mediante esta tabla seremos capaces de extraer todas las conclusiones que querramos. Siendo G y F sentencias complejas lógicas.

1.5. Interpretaciones

Como su nombre indica no son más que las combinaciones que tiene una sentencia al variar los valores de sus variables, a cada interpretación verdadera la denominamos como "*modelo*". Los modelos se expresan de esta forma:

$$\boxed{I \models F}$$

Las podemos dividir en distintos tipos:

- **Satisfacibles**: Fórmulas que tienen algún modelo que verifica que es cierta la proposición. Se le oponen las interpretaciones **Insatisfacibles**).
- **Tautologias**: Todas las interpretaciones son un modelo, son ciertas. Se le denominan *válidas*. Se escriben como:

$$\models F$$

- Se le oponen las **Contigentes**, que verifican que no es una tautología pero tampoco todas son instatisfacibles.

1.6. Conjuntos

Varias fórmulas que podemos agrupar, dan lugar a un conjunto:

$$\boxed{\mathrm{S} = \{p_1, p_2, ..., p_n\} \quad \leftrightarrow \quad I \models \{\mathrm{S}\}}$$

Al igual que las fórmulas, podemos distinguir distintos tipos de conjuntos:

- **Consistente**: Se le llama a aquel conjunto que posee algún modelo, se le opone los *inconsistentes*
- **Consecuencia Lógica**: Todos los modelos de S son modelos de la fórmula o cuando hay inconsistencias en el conjunto. Lo representamos como:

$$\boxed{\mathrm{S} \models F}$$

  Siendo F la fórmula.

2. Tema 2: Deducción natural proposicional

2.1. Reglas

2.1.1. Reglas de la Conjunción

Introducción:

$$\boxed{\dfrac{F \qquad G}{F \land G} \land i}$$

Eliminación:

$$\boxed{\dfrac{F \land G}{F} \land e_1 \qquad \dfrac{F \land G}{G} \land e_2}$$

2.1.2. Reglas de la Doble Negación

Introducción:

$$\boxed{\dfrac{F}{\neg\neg F} \neg\neg i}$$

Eliminación:

$$\boxed{\dfrac{\neg\neg F}{F} \neg\neg e}$$

2.1.3. Reglas del Condicional

Introducción:

$$\boxed{\dfrac{\begin{array}{|c|}\hline F \\ \dots \\ G \\ \hline\end{array}}{F \to G} \to i}$$

Eliminación:

$$\boxed{\dfrac{F \quad F \to G}{G} \to e}$$

2.1.4. Reglas de la Disyunción

Introducción:

$$\boxed{\dfrac{F}{F \lor G} \lor i_1 \qquad \dfrac{G}{F \lor G} \lor i_2}$$

Eliminación:

$$\boxed{\dfrac{F \lor G \quad \begin{array}{|c|}\hline F \\ \dots \\ H \\ \hline\end{array} \quad \begin{array}{|c|}\hline G \\ \dots \\ H \\ \hline\end{array}}{H} \lor e}$$

2.1.5. Reglas de la Negación

Introducción, siendo $\perp$ una contradicción:

$$\boxed{\dfrac{\begin{array}{|c|}\hline F \\ \dots \\ \perp \\ \hline\end{array}}{\neg F} \neg i}$$

Eliminación:

$$\boxed{\frac{\perp}{F}\perp e} \qquad \boxed{\frac{F \quad \neg F}{\perp}\neg e}$$

2.1.6. Reglas del Bicondicional

Introducción:

$$\boxed{\frac{F \to G \quad G \to F}{F \leftrightarrow G} \leftrightarrow i}$$

Eliminación:

$$\boxed{\frac{F \leftrightarrow G}{F \to G} \leftrightarrow e_1 \qquad \frac{F \leftrightarrow G}{G \to F} \leftrightarrow e_2}$$

2.2. Reglas Derivadas

2.2.1. Regla derivada Modus Tollens

Expresión:

$$\boxed{\frac{F \to G \quad \neg G}{\neg F} \quad MT}$$

2.2.2. Regla de Reducción al Absurdo

Expresión:

$$\boxed{\frac{\boxed{\begin{array}{c} F \\ \cdots \\ \perp \end{array}}}{F} \quad RAA}$$

3. Tema 3: Tableros semánticos proposicionales

3.1. Notación Uniforme

3.1.1. Fórmulas Alpha

F	F_1	F_2
$A_1 \wedge A_2$	A_1	A_2
$\neg(A_1 \rightarrow A_2)$	A_1	$\neg A_2$
$\neg(A_1 \vee A_2)$	$\neg A_1$	$\neg A_2$
$A_1 \leftrightarrow A_2$	$A_1 \rightarrow A_2$	$A_2 \rightarrow A_1$

Estas fórmulas no generarán nuevos conjuntos.
El conjunto de formulas atómicas resultantes genera una expresión del tipo: $F \equiv F_1 \wedge F_2$

3.1.2. Fórmulas Beta

F	F_1	F_2
$B_1 \vee B_2$	B_1	B_2
$B_1 \rightarrow B_2$	$\neg B_1$	B_2
$\neg(B_1 \wedge B_2)$	$\neg B_1$	$\neg B_2$
$\neg(B_1 \leftrightarrow B_2)$	$\neg(B_1 \rightarrow B_2)$	$\neg(B_2 \rightarrow B_1)$

Estas fórmulas generarán nuevos conjuntos.
El conjunto de formulas atómicas resultantes genera una expresión del tipo: $F \equiv F_1 \vee F_2$

3.1.3. Fórmulas interesantes

- $F \equiv \neg\neg F$
- $\neg(A \wedge B) \equiv \neg A \vee \neg B$
-
- $\neg(A \vee B) \equiv \neg A \wedge \neg B$

3.2. Ejemplo

$$\boxed{\{\neg(\neg p \vee \neg q \rightarrow \neg(p \wedge r))\ \}}$$

$$\boxed{\{\neg p \vee \neg q, \neg\neg(p \wedge r)\ \}}$$

$$\boxed{\{\neg p \vee \neg q, p, r\ \}}$$

$$\boxed{\{\neg p, p, r\ \}} \qquad \boxed{\{\neg q, p, r\ \}}$$

$$\boxed{\perp} \qquad \boxed{\text{Tendrá algún modelo}}$$

4. Tema 4: Formas Normales

4.1. Introducción

Usando transformaciones buscamos crear expresiones tales que obtenemos funciones FNC, *formas normales conjuntivas*, o FND, *formas normales disyuntivas*.

$$\text{FNC} \equiv (F_1^1 \vee F_2^1 ...) \wedge ...(F_1^n \vee F_2^n ...)$$

$$\text{FND} \equiv (F_1^1 \wedge F_2^1 ...) \vee ...(F_1^n \wedge F_2^n ...)$$

4.1.1. FNC

Podemos afirmar que las expresiones de este tipo pueden demostrar si existe una tautología, solo en caso en el que haya un conjunto de disyunciones tales que den como resultado una incongruencia.
Si queremos saber si es una tautología, comprobamos que esta FND sea satisfacible:

$$\boxed{G = \text{FNC}(F) \Rightarrow \text{FND}(\neg F)}$$

4.1.2. FND

Podemos afirmar que las expresiones de este tipo pueden demostrar si una expresión es satisfacible, solo y solo si podemos encontrar algún modelo, es decir haya alguna conjunción tal que sea contraria a otra.
Si queremos comprobar si es un insatisfacible, comprobamos que esta FNC sea una tautología

$$\boxed{G = \text{FND}(F) \Rightarrow \text{FNC}(\neg F)}$$

4.2. Expresiones a convertir

- $A \to B \equiv \neg A \vee B$
- $A \leftrightarrow B \equiv (A \to B) \wedge (B \to A)$

5. Tema 5: Resolución proposicional

5.1. Sintaxis Básica

Recordemos que las proposiciones son *variable proposicionales*: $p, r, ...$
Una variable proposicional puede combinarse con otras de forma que crea *literales*: $L, L_1, L_2, ...$
Combinar varios literales crean *cláusulas*: $C, C_1, C_2, ...$ Si estuviera vacío, lo identificaremos con el símbolo $\square$.
Si agrupamos cláusulas creamos *conjuntos*: $S, S_1, S_2, ...$ Si se encuentra vacío, se representa como $\emptyset$.

5.2. Semántica

Lógicamente sabemos cuando una variable o un literal valen verdadero o falso, pero las cláusulas y los conjuntos son diferentes:

5.2.1. Cláusulas

Será falsa una cláusula cuando para todos los literales que forman esa cláusula, haya uno tal que $I(l) = 0$ *O cuando la cláusula esté vacía* $I(\square) = 0$
Será verdadera una cláusula cuando para todos los literales $I(l) = 1$.

5.2.2. Conjuntos

Será verdadero un conjunto cuando para todas los cláusulas que forman ese conjunto, haya uno tal que $I(C) = 1$ *O cuando el conjunto esté vacío* $I(\emptyset) = 1$
Será falso un conjunto cuando para todas las clausulas $I(C) = 0$.

5.3. Sintaxis

Como hemos visto, un conjunto de cláusulas es una expresión como la siguiente:

$$\boxed{\{\{p, \neg q\}, \{\neg r, s, p\}\}}$$

Está formado por variables atómicas y las podemos montar como una FND:

$$\boxed{(p \wedge \neg q) \vee (\neg r \wedge s \wedge p)}$$

Considerando que un conjunto de cláusulas lo representamos como S y un literal con la letra L podemos decir que L es consecuencia directa de S solo si:

$$\boxed{S \models L \quad \leftrightarrow \quad S \cup \{\neg L\} \text{ Es inconsistente}}$$

6. Tema 6: Algoritmos para SAT. Aplicaciones

6.1. Equivalencias

Podemos decir que un conjunto es equivalente a otro si ambos son consistentes:

$$\boxed{\{\{p\}\} \approx \{\{p\},\{q\}\}}$$

$$\boxed{\{\{p\}\} \not\approx \{\{p\},\{\neg p\}\}}$$

6.1.1. Complementario

$$\boxed{I(L) = 1 \quad I(L^c) = \neg I(L) = 0}$$

6.2. Técnicas de Eliminación

6.2.1. Eliminación de Tautologías

Un conjunto de cláusulas puede reducirse solo a aquellas cláusulas que generan una tautología, eliminando las demás:

$$\boxed{\{\{p,q\},\{p,q,\neg p\}\} \approx \{\{p,q\}\}}$$

6.2.2. Eliminación Unitaria

Si existe una cláusula con un solo literal podemos eliminar todos aquellos literales que sean su complementario, y eliminar aquellas cláusulas con el mismo literal:

$$\boxed{\{\{p,q,r\},\{p,\neg q\},\{\neg p\},\{r,u\}\} \approx \{\{q,\neg r\},\{\neg q\},\{r,u\}\} \approx \{\{\neg r\},\{r,u\}\} \approx \{\{u\}\}}$$

6.2.3. Eliminación de Literales Puros

Si existe en el conjunto, un literal y no existe su complementario, podemos eliminar todos las cláusulas que contengan este literal:

$$\boxed{\{\{p,q\},\{p,\neg q\},\{r,q\}\ \{r,\neg q\}\} \approx \{\{r,q\},\{r,\neg q\}\} \approx \{\{\}\}}$$

6.2.4. Regla de División

En caso de que no podamos usar ninguna de estas reglas anteriores aplicaremos esta regla, mediante la cual indicaremos si un conjunto es consistente si añadiendo una cláusula con un solo literal, que pertenezca a este conjunto, podemos aplicar *eliminación Unitaria* siempre y cuando probemos con este literal y su complementario.

7. Tema 7: Sintaxis y semántica de la lógica de primer orden

7.1. Sintaxis

Aprovecharemos toda la sintaxis dada en la *Lógica Proposicional* más nuevas formas de representar el conocimiento, siendo más parecidas al lenguaje natural.

7.1.1. Nuevos Elementos

Representaremos nuestros entornos con 4 términos posibles:

- **Constantes**: Objetos de nuestro entorno, cuyo valor es fijo. Ej: $(a, b, c, ...$
- **Variables**: Objetos de nuestro entorno los cuales pueden representar cualquier objeto de nuestro dominio del problema. Ej: $x, y, z, ...$
- **Predicados**: Un tipo de expresión, que recibirá de 0 a N parámetros objeto, indicado por su *aridad*, y el cual devolverá o verdadero o falso. Se representan con la letra mayúscula
- **Funciones**: Expresiones las cuales reciben objetos y devuelven otro objeto. Se representan con la letra minúscula.

7.1.2. Nuevos Símbolos

- $\forall$: Es un cuantificador universal, que indica que verifica la condición para cualquier objeto del dominio del problema.
- $\exists$: Es un cuantificador existencial, que indica que la condición se verifica para al menos 1 elemento de nuestro dominio del problema.
- $=$: Es un operador que compara dos objetos.

7.2. Variables Libres o Ligadas

Denominamos a estas variables a aquellas tales que dada una expresión, serán **libres** si aparece fuera de un operador cuantificador una vez, y **ligadas** si aparece dentro de uno.

$$\forall_x (P(x) \to R(x, y))$$

x es ligada mientras que y es libre.

7.3. Estructura del Lenguaje

Una estructura del lenguaje es un par **Universo**, **Interpretación**, lo representamos así:

$$\mathbb{I} = (U, \mathrm{I})$$

- Denominamos **Universo** al conjunto de objetos que conforman nuestro dominio del problema.
- Denominamos **Interpretación** al conjunto de constantes, funciones y predicados que usaremos sobre el Universo.

$$\mathbb{I} = (\{u, v, w\}, \{(b, c), (P/1, Q/2), (f/2)\})$$

Con estas dos herramientas, Universo e Interpretación, podremos obtener modelos de una expresión aplicando las interpretaciones a cada operador de la expresión, usando como objetos del problema los objetos del universo.
Diremos que una **estructura** e **interpretación** es una realización de F tal que la expresión tiene modelo con esa interpretación.
La **estructura** es un modelo de F cuando para toda interpretación de la estructura, F es un modelo.

8. Tema 8: Deducción Natural en lógica de primer orden

8.1. Sustituciones

Definimos una *sustitución* σ a la aplicación de un objeto t a un término de nuestro Universo.
Los podemos definir de la siguiente forma:

$$\boxed{F\,[x_1/t_1, ..., x_n/t_n]}$$

Siendo F el término a evaluar, y x_n/t_n la sustitución del objeto x_n por el t_n.
Un ejemplo es el siguiente:

$$\sigma = [x/f(y), y/b]$$

$$A \equiv \forall_x (Q(x) \to R(x,y)) = \forall_x (Q(x) \to R(x,b))$$

$$B \equiv Q(x) \to \forall_x R(x,y) = Q(f(y)) \to R(x,b)$$

Son fórmulas resultantes de una sustitución de *ocurrencias libres*.

8.1.1. Sustituciones Libres

Esta clase de sustituciones no introducen ninguna ocurrencia nueva, de ninguna variable.
Un ejemplo de una que no lo es:

$$\exists_x (x < y) =_{[y/x]} \exists_x (x < x)$$

8.2. Reglas de Deducción Natural de Cuantificadores

Vamos a aprovechar todas las reglas anteriores para esto, más unas pocas más:

8.2.1. Regla del Cuantificador Universal Introducción

$$\boxed{\dfrac{\boxed{\begin{array}{c} x_o \;\; \text{supuesto} \\ \vdots \\ F\,[x/x] \end{array}}}{\forall_x F}\ \forall i}$$

8.2.2. Reglas del Cuantificador Existencial Introducción

$$\boxed{\dfrac{F\,[x/t]}{\exists_x F}\ \exists i}$$

8.2.3. Regla del Cuantificador Universal Eliminación

$$\boxed{\dfrac{\forall_x F}{F\,[x/t]}\ \forall e}$$

8.2.4. Reglas del Cuantificador Existencial Eliminación

$$\dfrac{\exists_x F \quad \boxed{\begin{array}{c} x_o \text{ supuesto} \\ \vdots \\ G \end{array}}}{G} \; \exists e$$

8.2.5. Ejemplos

$$\{\forall_x (P(x) \to Q(x))\} \models \forall_x P(x) \to \forall_x Q(x)$$

1) $\forall_x (P(x) \to Q(x))$	Premisa	
2) $\forall_x P(x)$	Supuesto	
3)x_0	Supuesto	
4)$P(x_0)$	$\forall e$	2
5)$P(x_0) \to Q(x_0)$	$\forall e$	1
6)$Q(x_0)$	$\to e$	$4, 5$
7)$\forall_x Q(x)$	$\forall i$	$3, 6$
8)$\forall_x P(x) \to \forall_x Q(x)$	$\to i$	$2, 7$

8.3. Reglas de Igualdad

8.3.1. Eliminación de la Igualdad

$$\dfrac{t_1 = t_2 \quad F[x/t_1]}{F[x/t_2]} \; =_e$$

8.3.2. Introducción a la Igualdad

$$\dfrac{}{t = t} \; =_i$$

9. Tema 9: Tableros semánticos en lógica de primer orden

9.1. Nuevas fórmulas

A parte de las fórmulas **alfas** y **betas** podremos usar dos nuevos tipos de fórmulas, que se usarán para eliminar los operadores.

9.1.1. Fórmulas Gamma

Los **términos básicos** son elementos ya existentes en el universo, por lo que usaremos esas fórmulas hasta que se nos acaben los elementos del universo.

F	F_1	Explicacion
$\forall G$	$G[x/t]$	Con t siendo un término básico
$\neg\exists G$	$\neg G[x/t]$	Con t siendo un término básico

9.1.2. Fórmulas Delta

Las **constantes** son nuevos elementos en el universo, por lo que las formulas resultantes no se repetirán.

F	F_1	Explicacion
$\exists D$	$D[x/a]$	Con a siendo una nueva constante
$\neg\forall D$	$\neg D[x/a]$	Con a siendo una nueva constante

10. Tema 10: Formas normales de Skolem y cláusulas

10.1. Forma Normal Prenexa

Considerando que la FNP, **forma normal prenexa**, es la formula por la cual se obtiene la forma clausal, se quiere obtener una fórmula tal que los operadores <u>universales</u> y <u>existenciales</u> se eliminen o queden lo más externalizados posibles. Nosotros trabajamos con fórmulas cerradas, por lo que realizaremos el siguiente algoritmo:

1) Rectificar la fórmula, es decir, si existe alguna variable que se esté repitiendo con otro operador, cambiar el nombre de la variable.

2) Eliminar los **bicondicionales**.

3) Eliminar los **condicionales**.

4) Interiorizar las negaciones usando Morgan.

5) Exteriorizar los operadores de LPO, de forma que siempre y cuando sean no libres.

10.1.1. Ejemplo

$$\boxed{\neg\exists_x \left[P(x) \to \forall_x P(x)\right] \approx \neg\exists_x \left[\neg P(x) \lor \forall_y P(y)\right] \approx \forall_x \exists_y \left[P(x) \land \neg P(y)\right]}$$

10.2. Skolem

Ahora que tenemos la FNP, podemos aplicar Skolem, eliminando los cuantificadores, de forma que sustituiremos las variable correspondientes a los cuantificadores **Existenciales** por funciones, con tantos argumentos como operadores universales tengan delante, si no poseen ninguno se sustituye por una constante.

$$\boxed{\mathrm{Skol}\left(\forall_x \exists_y \left[P(x) \land \neg P(y)\right]\right) \approx \forall_x \left[P(x) \land \neg P(f(x))\right]}$$

11. Tema 11: Resolución en lógica de primer orden

11.1. Forma Clausal

Considerando que partimos de una FNP, transformada con Skolem, creamos conjuntos de cláusulas por tantas conjunciones de disyunciones haya:

$$\boxed{\forall_x \left[P(x) \land \neg P(f(x)) \right]}$$

Los algoritmos para saturación y resolución son los mismos, salvo por los unificadores, mediante los cuales se pueden unir fórmulas transformando sus variables y constantes y que concuerden.