Un mecanismo semántico aplicado a la documentación automática

Por A. Cristóbal Lorente

DESCRIPCION DEL TRABAJO

1. El Mecanismo Semántico que presentamos es un esquema matemático que unifica diferentes aspectos del Reconocimiento de Formas, Lenguajes Formales y Diagnóstico Automático. Sin embargo, intencionadamente, este trabajo presenta sólo el Mecanismo proyectado en su aplicación al problema de la Clasificación e Interpretación Automática de Documentos. Esta particularización la hemos escogido en base a lo siguiente:

1) Es altamente heurística y, por tanto, un importante *test* para nuestro mecanismo.

2) El Tratamiento de la Información, en el aspecto anteriormente citado, ha adquirido una gran importancia en los últimos años, pese a lo cual creemos que nuestra exposición es auténticamente nueva y completa.

2. En el punto 1 presentamos un modelo para un posterior desarrollo. Merece atención especial, dado que introduce la noción de «proyectividad» de una manera natural en el concepto de diccionario, fundamental en la lingüística matemática.

3. El teorema 2.10 tiene quizá interés en sí mismo, como notará el lector interesado en las investigaciones que en la actualidad se llevan a cabo sobre «algoritmos paralelos». En este mismo punto 2 se define un modelo aplicable a diversos modelos analíticos lingüísticos (4).

4. En el punto 3 presentamos un modelo que contiene, como caso particular, el algoritmo de Ferrari.

5. En el punto 4 se aplica el mecanismo a la clasificación de los «documentos», previa interpretación de los «conceptos».

1. NOCION DE ASOCIACION SEMANTICA EN UN DICCIONARIO

A continuación damos un desarrollo del concepto de diccionario, el cual viene considerado en este trabajo como un Sistema General. Además, nuestro planteamiento constituye un modelo que consigue, de una forma simple, la idea de proyectividad, esencial en la estructuración formal de los lenguajes naturales, y, pensamos, en los sistemas jerarquizados coordinables. También nuestro sistema puede tratarse fácilmente mediante la Teoría de Modelos (1).

1.1 Esencialmente, en un diccionario cada palabra, x, se describe mediante un conjunto finito, $C(x)$, *de palabras*.

Sea PY el conjunto de todos los subconjuntos finitos de un conjunto dado Y, no vacío, y sea X un conjunto, no vacío, que eventualmente puede ser el mismo Y.

A un sistema $S \subset X \times PY$ lo llamaremos en este apartado diccionario.

El diccionario debe dotarse con alguna regla de decisión sobre problemas extremales en la cantidad de sinónimos.

Para cada $x \in X$, $C(x) \neq \emptyset$, sea $\mathbf{P}$ un conjunto de símbolos de predicados, tal que cada $P \in \mathbf{P}$ (el grado $P=n$) transforma el conjunto producto

$$\underbrace{C(x) \times C(x) \ldots \times C(x)}_{n \text{ veces}}$$

en el conjunto $\{0,1\}$.

Nota. El conjunto $[C(x), \mathbf{P}]$ constituye un modelo de la palabra x.

1.2 Sea primero el espacio de asociación de un $x \in X$

$$S(x) = \{(x, y_1, \ldots, y_n) \,/\, (\forall_i)\, y_i \in C(x) : \\ : (y_1, \ldots, y_n) \in \mathbf{P}(x)\}$$

donde

$\mathbf{P}(x)$ significa el conjunto de símbolos predicados definido en $C(x)$.

$(y_1, \ldots, y_n) \in \mathbf{P}(x)$ significa que existe un $P \subset \mathbf{P}(x)$ tal que $P(y_1, \ldots, y_n) = 1$.

Un conjunto $A \subset C(x)$ es una asociación de x si para todo $y_1, \ldots, y_n$ tal que $P \subset \mathbf{P}(x)$, y $P(y_1, \ldots, y_n) = 1$, existe un i tal que $y_i \in A$.

Si A es una asociación de x, lo mismo sucede para todo $BcC(x)$ que contenga A.

Sea el espacio de asociación del diccionario

$$S=\{(x, y_1, \dots, y_n) / x \in X\ (\forall_i)\ y_i \in C(x) : \\ : (y_1, \dots, y_n) \in \mathbf{P}(x)\}$$

Dos espacios de asociación, S_1, S_2, sobre (X, C),

$$C=U\{C(x) / x \in X\}$$

son equivalentes, si para todo $x \in X$ el conjunto AcC es una asociación x para ambos espacios.

1.3 Podemos enunciar el siguiente teorema:

Sea

$$S=\{(x, y_1, \dots, y_n) / x \in X\ (\forall_i)\ y_i \in C(x) : \\ : (y_1, \dots, y_n) \in \mathbf{P}(x)\}$$

es un espacio de asociación. Consideremos el conjunto de símbolos predicados:

$\mathbf{P}'(x)$ definido sobre C, de tal manera que su dominio sea exactamente $(y_1, \dots, y_n)$, donde existe un i tal que

$$y_i \in C-C(x),\ y(\forall_j \neq i)\ y_j \in C(x)$$

Entonces, el espacio

$$S_1=\{(x, y_1, \dots, y_n) / x \in X\ (\forall_i)\ y_i \in C : \\ : (y_i, \dots, y_n) \in \mathbf{P}(x) \cup \mathbf{P}'(x)\}$$

es equivalente a S en el sentido de que AcC es una asociación de x en S_1 si y sólo si $A \cap C(x)$ lo es en S.

La demostración se deja al lector.

1.4 Consideremos ahora para cada $x \in X$ una pareja, $C(x)$ y $C'(x)$, de conjuntos relacionados con x.

Sea,

$$\Phi=\{(x_i x'_j) / x_i \in C(x),\ x'_j \in C'(x)\} \cup \\ \cup \{(x_i\ \Lambda) / x_j \in C(x)\} \cup \{(\Lambda\ x'_i) / x'_i \in C'(x)\}$$

Nota: Λ es la palabra vacía. En la práctica, a cada x se le puede asociar diversas parejas $C_i(x)$, $C'_i(x)$.

El par (x, Φ) significa verbalmente que el «punto» x está situado o contenido en Φ.

Sea Φ una familia de elementos definidos como antes. A cada x situado en Φ le asociamos un «entorno» $xE\,\Phi$ tal que:

1) $x\,\Phi\ xE\,\Phi$

2) $xE\,\Phi\ c\ \Phi$

3) Si x está situado en Φ_1 y en Φ_2 y $xE\,\Phi_1 \underline{c}\ \Phi_2\ c\ \Phi_1$, entonces $xE\,\Phi_1=xE\,\Phi_2$

1.5 *Ejemplo:* Sea x tal que

$$C(x)=\{x_1, \dots, x_n, \dots\} \\ C'(x)=\{x'_1, \dots, x'_n, \dots\}$$

Un entorno de orden 1, para x, lo definimos como el conjunto formado por

$$[x_1, \Lambda] \quad ; \quad [\Lambda, x'_1] \in \Phi$$

Un entorno de orden k se define como:

a) Todos los elementos de orden $k-1$.

b) Los elementos

$$[x_k, \Lambda] \quad ; \quad [\Lambda, x_k] \in \Phi$$

c) Todos los elementos

$$[x_k, x_j] \in \Phi \\ j=1, \dots, k-1 \\ [x_j, x_k] \in \Phi \\ j=1, \dots, k-1$$

Por tanto existe un sistema de entorno para cada x; es decir,

$$_xE^k\,\Phi\ \underline{c}\ _xE^{k+1}_{\Phi}$$

2. APLICACION A MODELOS ANALITICOS Y ALGORITMO CUASIPARALELO

La noción de sistema diccionario introducida en este apartado puede aplicarse a los modelos analíticos (4). Los resultados de (6) son válidos aquí.

2.1 *Definición.*—Llamaremos «diccionario» a una cuádrupla (X, Y, f, g),

donde:

a) X, Y, son conjuntos no vacíos.

b) $f : X \longrightarrow 2^Y$, tal que a $x \in X$ le hace corresponder un subconjunto finito $Y_1 \in 2^Y$.

c) $g : X \longrightarrow X$, de modo que existe un k tal $g^k=g^{k-1}$.

d) fg está siempre definido, dado que f, g en la práctica no necesitan más que estar parcialmente definidas.

2.2 *Definición.*—Sean PcY finito. Decimos que una variable x es efectiva respecto a P, si se verifica que $f(x) \cap P \neq \Phi$.

Sea $D=(X, Y, f)$ una proyección de un diccionario.

Sean D_1 y D_2 dos proyecciones, de manera que

$$D_1=(X, Y_1, f_1) \quad D_2=(X, Y_2, f_2)$$

2.3 *Definición.*—Decimos que $D_1 > D_2$ si y sólo si para todo $x_1, x_2 \in X$, sucede lo siguiente:

Si $f_1(x_1)\ c\ f_1(x_2)$, entonces $f_2(x_1)\ c\ f_2(x_2)$.

Escribimos:

$$D_1 \sim D_2 \quad \text{si} \quad D_1 > D_2 \quad \text{y} \quad D_1 < D_2$$

2.4 Sean

$$D=(X, Y, f) \quad \text{y} \quad D'=(X, Y', f')$$

tal que

1) $Y'=Y$
2) $f'(x)=P \cap f(x)$ para $x \in X$.

Teorema: $D' \sim D \quad (D')'=D'$.

Demostración:

1) $f(x_1)\ c\ f(x_2)$ si y sólo si $P \cap f(x_1)\ c\ P \cap \cap f(x_2)$.

2) *Sea* $x \in X$:

$$f''(x)=P \cap f'(x)=f'(x)$$

2.5 Sea ahora h un mapa de X en 2^X, de forma que

$$h(x)=\{x' \in X / |P \cap [f(x)-f(x')]| \geqq \Theta\}$$

es decir, el conjunto de $x' \in X$, tal que el cardinal de $P \cap [f(x)-f(x')] \geqq \Theta$.

Teorema: $h=h'$.

Demostración: Es decir,

$$|P \cap f'(x)-f'(x')| \geqq \Theta \\ \text{si} \quad |P \cap f(x)-f(x')| \geqq \Theta$$

En efecto,

$$P \cap [f'(x)-f'(x')]=P \cap [P \cap f(x)]-[P \cap f(x')]= \\ =P \cap [f(x)-f(x')]$$

2.6 Sea ahora $\bar{D}$ definido como sigue:

1.° $\bar{Y}=Y$.

2.° $\bar{f}(x)=P \cap \underset{x \in h(x)}{U} f(x)$ para $x \in X$

Teorema:

1.° $(\bar{D}')=\bar{D}$.

2.° $(\bar{D})'=D$.

Demostración: Es similar a las anteriores.

2.7 *Definición.*—Sea $\{A_i\}\ i \in I$ una familia finita de conjuntos finitos. Un documento es un elemento d, de la forma

$$d\ c\ X\ A_i$$

Nota: A efectos prácticos, se supone que cada documento es equivalente a sí mismo respecto cualquier relación.

Todos los resultados de (2) interesan aquí.

2.8 A continuación daremos un teorema que afectaría a la recogida automática de documentos.

Sean ahora X, Y conjuntos finitos; denotamos por π permutaciones sobre elementos de PX, Π permutaciones sobre elementos de PY.

Sean las parejas

$$A=(I,\ A_i) \\ B=(J,\ B_j)$$

donde I, JcX, para todo i, jA_i, B_jcY con α simbolizamos Π_{Π}; entonces

$$\alpha\,(A) \equiv \Pi\ A_{\Pi\,(i)}$$

Si

$$\beta \equiv \Pi'_{\Pi'} \quad \text{y} \quad \alpha=\Pi_{\Pi}$$

entonces

$$\beta\alpha \equiv \Pi'\Pi_{\Pi'\Pi}$$

2.9 *Definición.*—$A \sim B$ si existen dos permutaciones Π, π, tales que

$$\Pi\, A_{\Pi\,(i)} = B_j$$

lo representamos también por $\alpha\,(A) = B$.

Es fácil demostrar el siguiente teorema: $\sim$ es una relación de equivalencia.

2.10. Sea ahora la relación

$$[A, B] = \{\alpha \mid \alpha\,(A) = B\}$$

Entonces

$$\beta\,[A, B] = \{\beta\,\alpha \mid \alpha \in [A, B]\}$$

Definición:

$$[A, B] \sim [A', B'] \quad \text{si} \quad E\beta$$

tal que

$$\beta\,[A, B] = [A', B']$$

Teorema: Sean A, B, C tres parejas, como antes. Se verifica:

$$A \sim B \rightarrow [C, A] \sim [C, B]$$

Demostración: Sea

$$\alpha \equiv \Pi_{\Pi} \in [C, A]$$

entonces

$$\Pi_{\Pi}\,(C) = \Pi\, C_{\Pi(i)} = A_j$$

Sea

$$-\,\Pi_{1_{\Pi_1}} \quad \text{tal que} \quad \Pi_{1_{\Pi_1}} A = \Pi_1\, A_{\Pi_1\,(i)} = B_k$$

Entonces

$$(\Pi_{1_{\Pi_1}} \Pi_{\Pi}) \quad \text{pertenece a} \quad [C, B]$$

En efecto:

$$\Pi_{\Pi_1} \Pi_{\Pi}(C) = \Pi_{1_{\Pi_1}}(\Pi\, C_{\Pi\,(i)}) = \Pi_{1_{\Pi_1}}(A_j) = B_k$$

Si γ está en $[C, B]$, entonces será de la forma

$$\Pi_{\Pi}\,(C) = B_k$$

Entonces

$$(\Pi_{1_{\Pi_1}} \Pi^{-1}_{1_{\Pi_1}} \Pi_{\Pi}) \quad \text{pertenece a} \quad \Pi_{1_{\Pi_1}}[C, A]$$

En efecto, basta comprobar

$$\Pi^{-1}_{1_{\Pi_1 - 1}}[\Pi_{\Pi}(C)] = A_j$$

3. MODELO DE ESTRATEGIA DE BUSQUEDA EN BLOQUES DE GRAN MASA DE INFORMACION

La búsqueda y recuperación de información en un conjunto no estructurado de elementos necesita una previa actividad, normalmente basada en criterios no bien definidos, para la descripción de los elementos. Esa descripción debe permitir el agrupamiento de los datos mecánicamente y, posiblemente, tal procedimiento de clasificación automática admite asignar nuevos documentos a las categorías existentes.

La búsqueda de información consiste en un proceso de decisión sobre parte o toda la masa de documentos, escogiendo primero el grupo o grupos de elementos que cubren o «responden» a la pregunta realizada sobre el sistema, debiendo llegar, normalmente, a la individualización de los documentos pertenecientes a los grupos elegidos.

Este párrafo está relacionado con un método óptimo y versátil para descomponer un bloque de gran masa de información hasta individualizar la totalidad de sus elementos.

Cada elemento lleva asociado un conjunto de atributos. Aplicando un procedimiento sobre un subconjunto de los atributos se explicita el grafo de la descomposición.

La filosofía general del método que proponemos es la siguiente:

a) La masa de información es condensada en un conjunto de parámetros. Cada grupo de documentos relacionados es procesado independientemente. Los parámetros de información (PDI) asociados con cada bloque no son ni «palabras-clave», ni «resúmenes», sino características o propiedades de la información respecto algún criterio dado.

b) La recuperación de la información es un proceso que se realiza utilizando en memoria los PDI y no la masa de información misma.

Nota.—De esta manera es posible obtener gran aprovechamiento de la memoria, así como tiempos de respuesta rápidos. Por ejemplo, una masa de información de 4.000 documentos tiene un árbol explícito asociado de 11 niveles de promedio para clasificar todos los documentos. Este trabajo no contiene los programas escritos en ALGOL 60 que se utilizaron sobre la máquina IBM-7090 del CCUM.

3.1 *Sistema de Información.* — Llamamos Sistema de Información a la cuádrupla

$$(X, Y, M, T)$$

donde

X, Y, son conjuntos abstractos.
X, conjunto de documentos.
Y, conjunto de atributos.
M, es un mapa: $M : X \rightarrow Y$.
T, es una constante: «tolerancia» del sistema.

A cada elemento x, X le asociaremos un conjunto de elementos «próximos»:

$$B_x = \{x' \mid |M(x) \cap M(x')| \geq T\}$$

donde $|M(x)|$ representa el cardinal asociado con $M(x)$.

Tenemos así el conjunto de documentos clasificado en bloques redundantes, cuyo número puede ser controlado fácilmente por la tolerancia del sistema.

La definición anterior nos encuadra el problema objeto de este párrafo; es decir, clasificar los documentos pertenecientes a bloques de gran masa de información descritos mediante un número grande de atributos y controlados por T.

El proceso descompone el bloque de información B, en los subbloques B_p, tal que $\cup B_p = B$, $\cap B_p = 0$. El proceso se repite continuamente hasta obtener individualizados cada uno de los documentos.

3.2 *Resolución general aproximada.*—Sean X, Y, Z, conjuntos arbitrarios.

Sea v una función de $X \times Y$ en el conjunto Z ; Z parcialmente ordenado; W, una aplicación, tal que para cada elemento $y \in Y$, transforme $\cup\, v\,(x, y)$ en Z; es decir:

$$\begin{gathered} x \in X \\ v\colon\ X \times Y \rightarrow Z \\ w\colon\ \cup\, v\,(x, y) \rightarrow Z \\ x \in X \end{gathered}$$

La desigualdad

$$v\,(x, y) \leq w\,[\cup\, v\,(x, y)] \qquad [1]$$
$$x \in X$$

define para cada y el subconjunto $X_1 \subset X$, que constituye la información del bloque B_1; $(X_1)^c$ constituye el bloque B_2, por ejemplo.

La cuádrupla (X, Y, v, w) especifica la resolución del problema, y cualquier elemento y que satisfaga la desigualdad [1] para algún subconjunto X_1 constituye una «solución».

La pareja (y, Θ), donde

$$\Theta = w\,[\cup\, v\,(x, y)]$$
$$x \in X$$

se denomina «parámetro de información» perteneciente al bloque B.

Aún podemos controlar la clasificación mediante la quíntupla

$$(X, Y, v, w, P)$$

donde los elementos X, Y, v, w son definidos como antes. P es un predicado de grado $|X|$, donde $|X|$ es el cardinal de X.

Por último, puede utilizarse la desigualdad

$$|v\,(x, y) - \Theta| \leq \eta$$

en lugar de [1]. Donde η es un «punto de corte» definido de antemano según las características de cada elemento de Y.

3.3 *Documentos y atributos afectados.*—Una situación frecuente en los Sistemas de Información es la de contar con una información extra sobre los conjuntos X e Y.

La quíntupla

$$(X, Y, v, w, P)$$

junto con un «peso» que caracterice la importancia del atributo respecto a los documentos contenidos en el bloque, permite la clasificación hasta en cuatro subbloques en el mismo nivel.

Procedimiento similar hemos empleado en el caso de conocer información extra sobre X. En este caso definimos una función

$$F : X \rightarrow \{0, 1\}$$

tal que

$$F(x) = 1 \quad \text{si} \quad i \sum m_i y_i \leq \eta$$

$$i = 1, \ldots, l \text{ y } l$$

donde m_i representa los «pesos» de cada elemento de Y respecto de X.

Nota: Los procedimientos anteriores solucionan de una manera eficiente la clasificación automática de bloques de gran masa de información contenida en el computador en forma secuencial.

La pregunta p —broleana o secuencial— realizada al sistema se procesa con un programa general junto con los PDI, obtenidos en un tratamiento anterior, sobre la información del bloque.

4. DESCRIPCION DEL MECANISMO

4.1 Usamos la palabra «algoritmo» en el sentido de Markov. Entonces, un algoritmo A es *aplicable* a una palabra P si el proceso de aplicar A termina finalmente en alguna palabra Q. Se dice que A transforma P en Q, lo cual se expresa así: $A(P) = Q$. Un algoritmo es recursivo si es posible determinar para cualquier palabra R, si ésta es generada por A desde P (4).

Un conjunto se dice que es recursivo si existe un algoritmo recursivo definido en él.

El mecanismo produce la sucesión

$$(T_i, T_i^*)$$

$$0 \leq i \leq n$$

para todo i, $T_i \subset X$, $T_i^* \subset Y$.

La clasificación de documentos se realiza por los elementos T_i.

4.2 Sea I un subconjunto finito de X, y P un subconjunto finito de Y.

Sea

$$T_o(P) = \{x \,/\, x \in I, f(x) \cap P \neq \Phi\}$$

Es inmediato probar que:

Teorema:

a) $P < Q \rightarrow T_o(P) \leq T_o(Q)$

b) $T_o(P \cup Q) = T_o(P) \cup T_o(Q)$

c) $T_o(P \cap Q) \leq T_o(P)\, T_o(Q)$

4.3 Aparte, puede interesar la utilización de

$$T'_o(P, Q) = \{x \,/\, x \in I : f(x) \cap P \neq \Phi \text{ y } f(x) \cap Q \neq \Phi\}$$

Entonces se verifica el siguiente teorema:

$$T'_o(P, Q) \geq T_o(P, Q)$$

4.4 Definimos la sucesión

$$T_o, T_1, T_2, \ldots, T_m, \ldots$$

tal que

$$T_m = T_{m-1} \cup \{j \,/\, \exists\, i \in T_{m-1} : j = g(i), f(j) \cap P \neq \Phi\}$$

Teorema: El conjunto X es recursivo, es decir, para cada $x \in X$ podemos decir si es o no efectivo a P.

Demostración: Para todo m, $T_{m-1} \subseteq T_m$ por construcción.

Todo elemento $x \in X$ efectivo a P debe pertenecer a algún T_i. Unicamente basta mostrar que el algoritmo calcula hasta algún k, tal que

$$T_k = T_{k-1}$$

Entonces, si x no está en T_k, no estará en ningún $T_i\, i > k$; por tanto, x no es efectivo a P.

4.5 *Definición.*—Una sucesión cualquiera entrada-salida,

$$(T_i, T_i^*)$$

$$0 \leq i \leq n$$

está dirigida por un diccionario

$$D = (X, Y, f, g)$$

si

$$T_i = \bar{g}^i(T_o)$$

$$T_i^* = \bar{f}(T_i)$$

donde $\bar{f}, \bar{g}$ son las extensiones a conjuntos de f, g.

Teorema: Si la sucesión (T_i, T_i^*) está dirigida, entonces

$$T_i = T_k \quad \text{implica} \quad T_i = T_k \quad \text{y} \quad T_{i+1} = T_{k+1}$$

Demostración: Sea

$$(T_i, T_i^*)$$

dirigida por

$$D = (X, Y, f, g)$$

si

$$T_i = T_k, T_{i+1} = \bar{g}^{i+1}(T_o) = \bar{g}(T_i) = \bar{g}(T_k)$$

pero

$$\bar{g}(T_k) = \bar{g}^{k+1}(T_o) = T_{k+1}$$

$$T_i^* = \bar{f}(T_i) = \bar{f}(T_k) = T_k^*$$

4.6 *Definición.*—Sea G la familia $\{T_k\}$, decimos que

$$d_1 \sim d_2 \quad \text{si} \quad \{x \in X \,/\, d_1(x) = d_2(x)\} \in G$$

Es fácil demostrar el siguiente teorema:

$\sim$ es una relación de equivalencia.

REFERENCIAS

(1) BELL-SLOMSON: *Models and Ultraproducts.* North-Holland. 1969.

(2) DE LOBEL: «Aspects theóriques sur la structure de l'Information dans une base de données. *Revue française d'Informatique et de Recherche Operationnelle,* núm. B-3, septiembre 1971.

(3) FERRARI: *A Versatile System for the Automatic Reading of Texts Typographically Printed.* Mechanized Information Storage, Retrieval and Dissemination. North-Holland, 1968.

(4) MARKOV: *Theory of Algorithms.* IRST. Jerusalén, 1962.

(5) MARCUS: *Algebraic Linguistics, Analytical Models Academic Press,* 1967.

(6) SUNKOFS: *Dictionary Structure and Probability Measures. Information and Control.* Vol. 19, núm. 2, septiembre 1971.