http://dx.doi.org/10.12795/PH.1997.v11.i02.06

ESPECIFICACIONES LINGÜÍSTICAS PARA GRAMÁTICAS EN ESTRUCTURAS DE RASGOS TIPIFICADAS

Toni Badia

1. Introducción

Este artículo pretende presentar parte de los resultados de un proyecto cooperativo entre siete grupos de investigación cuya finalidad ha sido elaborar especificaciones lingüísticas para gramáticas en estructuras de rasgos tipificadas.[1] El proyecto se inscribe en un conjunto de proyectos y en un entorno de investigación que se describe en la sección primera del artículo. La sección segunda, central en el artículo, está dedicada a la definición de las características formales de las especificaciones. En ella se discuten algunos de los aspectos centrales para la formulación de gramáticas en estructuras de rasgos tipificadas que resulten aptas para aplicaciones reales de procesamiento del lenguaje natural; entre los aspectos tratados sobresale, por su importancia intrínseca y por las consecuencias que tiene para el resto de especificaciones, la definición del sistema de tipos. Finalmete, en una breve tercera sección se muestran algunas de las consecuencias que tienen las especificaciones elaboradas para uno de los fenómenos centrales en cualquier gramática: la estructura de predicado y argumentos.

2. Entorno del proyecto

Desde finales de los años 80 se han producido cambios significativos en las técnicas de representación del conocimiento lingüístico que no han repercutido todavía en los sistemas de procesamiento del lenguaje natural. Durante las décadas de los 70 y los 80 los sistemas de procesamiento del lenguaje natural han venido usando técnicas derivadas de la teoría lingüística generativa clásica, por un lado, y de la evolución de los analizadores a partir de las ATN, por el otro. Aún cuando estos sistemas fueron adoptando el principio de la separación entre el conocimiento lingüístico y la técnica de análisis (favoreciendo de este modo la declaratividad en la formulación del primero), sus puntos de partida imponían limitaciones graves a la representación del conocimiento lingüístico: la distribución en distintos niveles de representación, las estructuras arbóreas, la multiplicidad de reglas... daban como resultado gramáticas muy complejas y difíciles de manejar. Un caso paradigmático en este sentido es el de las gramáticas del sistema de traducción automática

[1] Se trata del proyecto "Investigation of linguistic specificactions for future industrial standards: The Eurotra Reference Manual" (MLAP 93-15). Los equipos de investigación que han intervenido han sido: IAI (Saarbrücken) (centro coordinador del proyecto), Universidad de Essex (Colchester), Universitat Pompeu Fabra (Barcelona), Gruppo DIMA (Torino), Université de Nancy, Katholieke Universiteit Leuven y UMIST (Manchester).

Eurotra (Steiner, 1991). Aunque básicamente el formalismo para especificar gramáticas permitía una formulación declarativa de las mismas, en la práctica el sistema en su conjunto no resultaba declarativo ni monótono: la distribución de la representación lingüística en tres o cuatro niveles que manejaban árboles y la necesidad de efectuar transformaciones entre los niveles introducía aspectos procedurales (p.ej., el orden de aplicación de los niveles era relevante) y no monótonos (p.ej., entre un nivel y el siguiente se podían modificar, ampliar, reducir... los árboles de representación). Por otra parte, la descripción lingüística resultante era muy poco manejable (es decir, legible, inspeccionable, ampliable...) : la misma distribución de la información en niveles independientes junto con la ausencia absoluta de una teoría sobre el léxico hacían que el tratamiento de los distintos fenómenos lingüísticos estuviera diseminado en una multiplicidad de reglas (con influencias mutuas varias) que resulta muy difícil de comprender. En cambio, la aplicación de técnicas de representación del conocimiento desarrolladas en el marco de la inteligencia artificial a la representación del conocimiento lingüístico ha comportado el nacimiento de una familia de formalismos (y teorías) para la representación lingüística. Se trata de los originalmente conocidos como formalismos (y teorías) basados en la unificación (Shieber, 1986) y que actualmente se suelen describir como basados en restricciones (siendo la unificación uno de los modos de expresar las restricciones), Aunque se han desarrollado algunos pequeños sistemas de procesamiento del lenguaje natural basados en estos formalismos y teorías, no ha habido ningún proyecto de desarrollo de gramáticas de este tipo orientadas a aplicaciones reales, hasta el conjunto de proyectos en torno a la plataforma de desarrollo gramatical Alep. En su conjunto estos proyectos presentan un formalismo gramatical (conocido también como Alep),[2] una gramáticas de tamaño medio en varias lenguas,[3] y unas especificaciones lingüísticas para desarrollar gramáticas de este tipo.[4] El proyecto MLAP 93-15, pues, tenía como objetivo principal elaborax unas especificaciones lingüísticas para gramáticas escritas en formalismos con estructuras de rasgos tipificadas. Fundamentalmente, el proyecto tenía dos puntos de partida complementarios: por un lado, el formalismo Alep y, por el otro, las especificaciones lingüísticas elaboradas en Eurotra. El formalismo Alep representa una implementación de las ideas desarrolladas alrededor de los formalismos (y teorías) basados en restricciones. Está, pues, plenamente en línea con los desarrollos más recientes en teoría gramatical y su computación: la información lingüística está codificada en estructuras de rasgos, es un formalismo monótono y declarativo, tiene un esqueleto libre de contexto, resulta suficientemente expresivo y es independiente de las teorías lingüísticas. No obstante, como este no es el único formalismo existente a este nivel, las especificaciones elaboradas en el proyecto han partido de un estudio formal previo (ver sección 3ª) para que

[2] El formalismo Alep fue diseñado en el proyecto ET6/1. El resultado del estudio es Alshawi et al. (1991) y su implementación ha corrido a cargo, primero, de BIM y, después, de Cray Systems.

[3] Se trata de las gramáticas desarrolladas en los proyectos LSGRAM "Large-scale grammars for EC languages" (LRE - 61029), que han elaborado gramáticas de tamaño medio para textos reales escogidos de entre un sublenguaje para las siguientes lenguas: alemán, danés, español, francés, griego, inglés, italiano, neerlandés y portugués.

[4] Estas especificaciones se han desarrollado en proyectos financiados dentro del programa MLAP. En conjunto había un proyecto de ámbito general (que es en el que se han obtenido los resultados expuestos en este artículo) y una serie de proyectos particulares para distintas lenguas (en concreto, se trataron el alemán, el danés, el francés, el griego, el italiano, el neerlandés y el portugués).

resulten adecuadas para formalismos distintos de Alep, aunque dentro de la misma familia. Por otra parte, las especificaciones lingüísticas elaboradas en Eurotra fueron juzgadas por los evaluadores finales del proyecto como útiles para futuros proyectos y desarrollos industriales (Oakley, 1992). Esto era consecuencia de que tenían una amplia cobertura, para varias lenguas, ofreciendo un todo coherente (entre las distintas lenguas y entre fenómenos lingüísticos distintos). No obstante, por las razones expuestas anteriormente, estas especificaciones no resultaban adecuadas desde el punto de vista descriptivo. Así pues, el planteamiento en el proyecto fue el siguiente: adaptar, en la medida de lo posible, las especificaciones de Eurotra al nuevo formalismo; para algunos casos, rediseñar el tratamiento ofrecido en función de los distintos planteamientos permitidos por el formalismo (p,ej., el lexicalismo); y, finalmente, mejorar y completar el tratamiento de fenómenos en que el formalismo de Eurotra impidió obtener especificaciones satisfactorias. Y todo ello, organizando el resultado de manera que garantizara un cierto nivel de completitud y un alto nivel de coherencia e integración.

3. Marco formal del proyecto[5]

Desde un punto de vista teórico, el proyecto se basa en HPSG.[6] Esta teoría ofrece en estos momentos los siguientes aspectos ventajosos: está motivada, entre otras razones, por consideraciones computacionales; existen estudios precisos sobre la lógica subyacente a la misma; en la práctica, se ha convertido en un estándar en lingüística computacional; permite una clara formulación lexicalista de los fenómenos lingüísticos; por lo tanto, la teoría gramatical resulta consistente con una teoría del léxico; permite establecer generalizaciones adecuadas mediante los principios de la teoría...

No obstante, en relación con la implementación de HPSG hay que tener en cuenta que coexisten dos tendencias claramente diferenciadas. Por un lado existen concreciones (y/o implementaciones) de la teoría (Zajac, 1993; Krieger y Nerbonne, 1991; Riehemann, 1993) que presuponen un aparato expresivo enormemente rico, sea en la definición de los tipos (Zajac, 1993; Riehemann, 1993), sea en el uso de las dependencias relacionales (Krieger y Nerbonne, 1991). Estos planteamientos no pueden ser usados en una descripción lingüística que intenta tener una amplia cobertura para ser usada en sistemas reales de procesamiento del lenguaje natural, a causa de su poca eficiencia computacional. En cambio, existen implementaciones de HPSG poco costosas expresivamente y que, por lo tanto, consiguen una mayor eficiencia en sistemas de procesamiento del lenguaje natural. Entre ellas encontramos los formalismos ALE (creado por Carpenter, a partir de sus trabajos sobre la lógica de los tipos) y Alep. Especialmente importantes en esta línea son las investigaciones de B. Carpenter (1992) que han llevado a especificar las bases formales de HPSG de una forma que mantienen un claro compromiso entre expresividad y eficiencia computacional.[7]

---

[5] En el desarrollo de esta sección seguimos bastante de cerca la exposición de P. Schmidt sobre el tema para el informe final del proyecto.

[6] "Head-driven Phrase Structure Grammar", teoría lingüística desarrollada a partir de los estudios sobre las gramáticas sintagmáticas ampliadas o generalizadas (GPSG, "Generalized Phrase Structure Grammar"). Para GPSG, ver Gazdar et al. (1985). Para HPSG, ver Pollard y Sag (1987; 1994), Nerbonne et al. (1994) y Balari y Dini (en prensa).

[7] De hecho, la concreción formal que se realiza en Pollard y Sag (1994) se basa casi exclusivamente en Carpenter (1992).

Asimismo, en algunos casos (especialmente, en los aspectos más estructurales de la descripción lingüística, es decir, morfología y estructura sintagmática), se ha propuesto un sistema aún menos costoso expresivamente que las versiones mas débiles de HPSG, En esto, el proyecto sigue los pasos iniciados por Pulman (1994) de tratar de reducir tratamientos tradicionalmente complejos a operaciones de bajo coste computacional. Por ejemplo, el tratamiento de la indeterminación en las dependencias de larga distancia mediante la técnica del "gap threading" (Pereira y Shieber, 1987), o la posibilidad de usar términos Prolog para codificar parcialmente la negación y la alternancia de valores (restringida a valores booleanos). En esta línea las características básicas del formalismo propuesto en el proyecto se discuten en los apartados siguientes.

3.1 El sistema de tipos

Respecto al sistema de tipos se plantean diversas cuestiones, en relación con las cuales los distintos autores no adoptan una posición común, En general en la caracterización del sistema de tipos para gramáticas HPSG se nota la división (comentada antes -ver sección la) entre los que permiten toda la expresividad posible y los que la restringen en aras a la eficiencia computacional. En primer lugar, la misma definición de los tipos se puede plantear en términos de condiciones de propiedad o mediante estructuras de rasgos. Una declaración de tipos en base a condiciones de propiedad consiste en especificar para cada tipo sus subtipos, los atributos que lo definen y los valores que estos atributos pueden tener. Así, la siguiente sería una correcta definición del tipo signo:

(1) tipo(signo) :
FON: lista(átomos)
SINSEM: tipo(sinsem)
subtipos: sintagmático, léxico

Según (1) el tipo signo permite/exige dos atributos: "fon" y "sinsem" (cuyos valores son del tipo lista de átomos y sinsem, respectivamente) y tiene dos subtipos (sintagmático y léxico). Naturalmente estos subtipos tienen que estar definidos a su vez en la declaración de tipos.

Ante una especificación de tipos como esta, se plantea la cuestión de cómo se determina la adecuación de las estructuras de rasgos. En principio podemos hablar de tres niveles distintos:

a) una estructura de rasgos es legítima cuando está bien tipificada, es decir, cuando sólo contiene atributos que son apropiados para su tipo,

b) una estructura de rasgos es legítima cuando está totalmente bien tipificada, es decir, cuando además contiene todos los atributos apropiados para su tipo,

c) una estructura de rasgos es legítima cuando está totalmente bien tipificada y además todos los tipos que contiene son maximales (o sea, lo más específicos posible).

Por otra parte, los tipos pueden ser definidos mediante estructuras de rasgos. En esta aproximación, la idea es que no hay restricción en cuanto a las especificaciones formales de

los tipos, puesto que las estructuras de rasgos usadas para definirlos pueden contener caminos indefinidamente largos y todo tipo de compartición de estructuras.

Una especificación de tipos mediante estructuras de rasgos permite la introducción de los principios (universales o particulares de la lengua tratada) como tipos en la jerarquía, Por ejemplo, el "HFP" ('Head Feature Principle', o principio de los rasgos de núcleo) podría ser representado en una estructura de rasgos y, consiguientemente, podría ser integrado en la jerarquía de tipos.

Claramente la adopción de uno u otro sistema de especificación de los tipos tiene consecuencias, tanto para el lingüista que implementa la gramática de una lengua particular, como para el sistema en su conjunto. Por un lado, según el sistema de definición de tipos que se adopte, el lingüista tendrá más o menos elementos expresivos a su disposición. Así, la definición de los tipos mediante condiciones de propiedad limita considerablemente los hechos expresables en el sistema de tipos. Por otro lado, la eficiencia (y, en definitiva, la computabilidad) de las gramáticas resultantes se ve claramente afectada por la opción de usar estructuras de rasgos para definir los tipos, En consecuencia, todos los sistemas pensados para ser usados realmente (notablemente, ALE y Alep) permiten solamente la definición de los tipos mediante condiciones de propiedad.[8]

Teniendo en cuenta que el objetivo principal del proyecto era la formulación de especificaciones lingüísticas para gramáticas reales (posiblemente implementadas en Alep) estaba claro que debíamos optar por la definición de tipos mediante condiciones de propiedad. A su vez, otro factor venía a reforzar la opción menos expresiva: en unos estudios llevados a cabo independientemente sobre la reutilización de gramáticas de un formalismo a otro (Markantonatou y Sadler, 1994), se llegó a la conclusión que los casos más problemáticos para la reutilización de los recursos entre formalismos basados en estructuras de rasgos se daban cuando se pretende traspasar información de un formalismo bastante expresivo a uno que lo es menos. Aplicando este principio a nuestro caso, es relativamente fácil trasladar gramáticas basadas en un sistema de tipos simple a otro formalismo que permite definir los tipos de forma más compleja, pero la operación en sentido inverso resultaría habitualmente muy costosa.

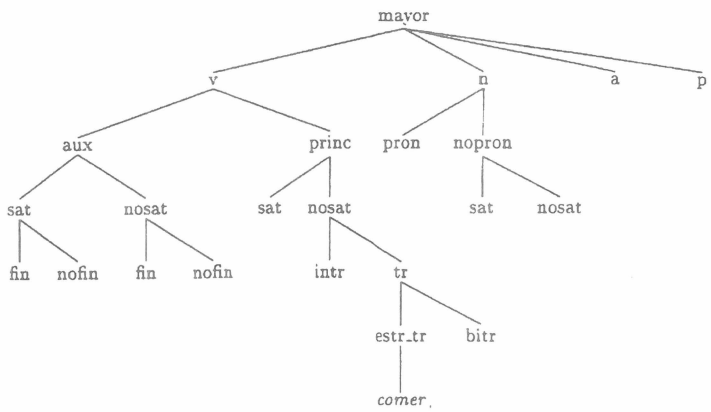
Otro aspecto importante respecto al sistema de tipos es la clase de herencia permitida. La organización de la información en redes de herencia ha sido estudiada ampliamente en el marco de la inteligencia artificial. En estos estudios se han propuesto muchas alternativas distintas a la simple organización taxonómica. Los sistemas de herencia usados en las aplicaciones lingüísticas pueden ser agrupados en las siguientes clases según el tipo de herencia usada:

a) con herencia simple y monótona
b) con herencia múltiple y monótona
c) con herencia simple por defecto (no monótona)
d) con herencia múltiple por defecto

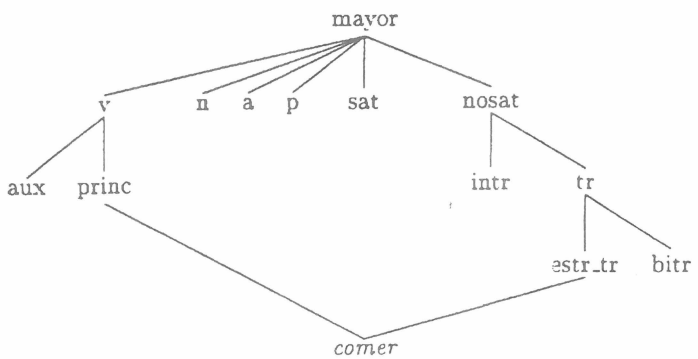
---

[8] En este sentido, hay que tener en cuenta que en la misma teoría de HPSG (en su segunda versión -ejemplificada notablemente por los ocho primeros capítulos de Pollard y Sag, 1994) se adopta la definición de tipos mediante condiciones de propiedad, siguiendo los trabajos de Carpenter (1992).

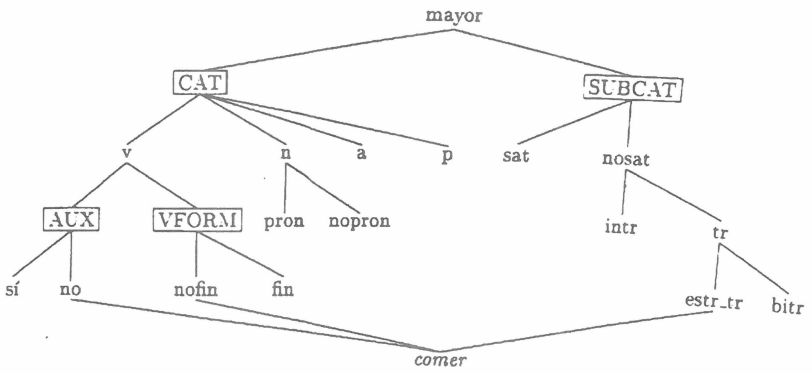
Los sistemas más simples pueden ser caracterizados como simples taxonomías: cada nodo hereda la información por un único camino; además cada nodo hereda toda la información que tiene su antecesor en el camino. El uso de herencias simples y monótonas para la descripción lingüística es claramente inadecuada, puesto que los elementos lingüísticos se clasifican según varios criterios que se entrecruzan mutuamente. En estos casos mantener la condición de que cada nodo tiene un único camino por el que hereda toda la información conlleva unos grados muy elevados de redundancia en la red. Véase, por ejemplo, el fragmento de una red pensada para dar cuenta de algunos aspectos esenciales de las categorías mayores:



Obviamente la solución a este tipo de redundancia reside en la herencia múltiple; es decir, en permitir que cada nodo herede información pox más de un camino. La siguiente figura muestra como podría representarse mediante herencia múltiple parte de la información contenida en el sistema de herencia anterior:



En una jerarquía como esta, el problema aparece cuando puede llegar información conflictiva en un mismo nodo. Para evitar este tipo de problemas se puede recurrir a determinar qué caminos son prioritarios (de forma que la información que llega por ellos prevalece sobre la de los demás), o a establecer particiones, o sea, conjuntos de nodos que contienen información incompatible. Al establecer particiones, los conflictos de información quedan resueltos ya que sólo se permite que cada nodo herede información de un único camino en cada partición (o sea, dentro de cada partición la herencia es simple). Un ejemplo de jerarquía de herencia con particiones es el siguiente:



Para muchos investigadores, la herencia múltiple con particiones no es suficiente para tratar varios aspectos del lenguaje natural, especialmente para representar la estructura del léxico. Existen varios fenómenos léxicos (especialmente en las relaciones morfológicas y en semántica léxica) que difícilmente pueden ser tratados si no se dispone de herencia por defecto. Es decir, si no se pueden tratar las excepciones en la relación de herencia a base de sobreescribir, en el nodo pertinente, la información heredada.

Los estudios generales sobre jerarquías de herencia (Touretzky et al., 1989; Russell, 1992; Daelemans et al., 1992) muestran que la interacción entre herencia múltiple y herencia por defecto presenta verdaderas dificultades, especialmente cuando se construye un sistema de tipos y una jerarquía relativamente grandes. Además, hay que tener en cuenta que la opción de definir los tipos mediante condiciones de propiedad obliga a usar tipos poco estructurados (o complejos). Las dos consideraciones nos llevan a adoptar una posición minimalista en cuanto al tipo de herencia permitida en nuestras especificaciones: herencia múltiple (basada en particiones) y monótona (o sea, sin el mecanismo de la herencia por defecto).[9] Es importante notar que la solución adoptada en relación con la herencia no tiene consecuencias especiales para las especificaciones; en particular, no hace más difícil el tratamiento del léxico. La verdadera dificultad deriva del hecho de que permitamos sólo la definición de tipos mediante condiciones de propiedad (impidiendo, por

[9] Este tipo de herencia se conoce a veces bajo el nombre de *herencia ortogonal.*

lo tanto, que existan tipos complejos o estructurados, que tendrían que ser definidos mediante estructuras de rasgos).[10]

3.2 Operaciones lógicas sobre tipos

En la línea del tratamiento poco costoso dado al sistema de tipos, consideramos que las operaciones sobre tipos debían estar restringidas para no introducir en este aspecto los problemas de computabilidad que habíamos tratado de resolver antes. El problema en este caso recaería en el hecho de que la introducción generalizada de la negación (o la disyunción) de tipos provocaría problemas graves de resolución *on line* de la unificación de tipos. Es decir, la negación de un tipo (si no está restringida) es algo absolutamente indeterminado. Por lo tanto, cualquier operación sobre este tipo resulta enormemente costosa.

Por otra parte, Pulman (1994) ha mostrado que hay varios tipos de aspectos (la concordancia, por ejemplo) que pueden ser tratados satisfactoriamente con tipos que tienen una traducción inmediata a ténninos Prolog. Se trata de tipos atómicos, para los cuales es posible definir relaciones de carácter booleano (con la negación y la disyunción incluidas, por supuesto). Así, por ejemplo, podemos definir el tipo booleano "concordancia" como una conjunción de valores atómicos (o su negación, o su disyunción) para representar el género, el número y la persona. Así, pues, el siguiente sería un rasgo de concordancia bien fonnado para un nombre como "análisis", que es masculino, puede ser singular o plural, y es de tercera persona.

(2) concordancia: masc & (sing $\vee$ plu) & $\neg$ ($1^a \vee 2^a$)

3.3 Clases de datos adicionales

En general, los formalismos basados en restricciones tienden a usar listas y conjuntos, como elementos adicionales a los tipos y estructuras de rasgos. Las listas no presentan ningún problema y pueden ser incorporadas en cualquier formalismo, por muy débil que deba ser.

Por otra parte, los conjuntos presentan algún problema adicional, puesto que las operaciones con ellos pueden introducir indeterminismo. No obstante, hay muchos aspectos de la descripción lingüística que no pueden ser tratados adecuadamente sin los conjuntos: son casi imprescindibles para la representación semántica (especialmente de los modificadores y de los aspectos contextuales), las técnicas más estándar de tratamiento de las dependencias de larga distancia también los usan, la relación de subcategorización para las lenguas de orden libre puede ser representada mucho más fácilmente con conjuntos que con listas, etc. Así pues, decidimos aceptar los conjuntos en nuestras especificaciones.

3.4 Reglas

Las teorías lingüísticas modernas tienen maneras distintas de expresar los principios de estructuración y combinación de la información lingüística. Aunque tradicionalmente se

[10] En este punto, conviene recordar que el objetivo principal del proyecto era la construcción de especificaciones gramaticales. Si hubiéramos estado pensando en especificaciones léxicas, sin duda las conclusiones habrían sido distintas.

han usado mucho las reglas de reescritura, en los planteamientos de los formalismos basados en restricciones se ha tendido a usar otros mecanismos expresivos. Desde GPSG (Gazdar et al., 1985), se tiende a distinguir entre dos aspectos distintos que en las reglas de reescritura no era posible distinguir: la relación de dominancia y la relación de precedencia. Una regla de reescritura expresa una relación de dominancia entre la madre de la regla y sus hijas y, al mismo tiempo, indica en qué orden aparecen las hijas.[11] Asimismo, en HPSG se ha introducido la idea de que estas relaciones se pueden expresar dentro de las estructuras de rasgos correspondientes a las categorías sintagmáticas, mediante el atributo DTRS (de *daughters).*

Aunque el poder expresivo de los tres mecanismos no es totalmente idéntico, en muchos aspectos se pueden equiparar (siempre, claro está, que las categorías que se manejan en las reglas de reescritura estén representadas por estructuras de rasgos). Las diferencias básicas radican en el nivel de fijación de la relaciones lineales: las reglas de reescritura imponen un orden fijo, el formato ID-LP permite variaciones limitadas de orden, y el uso del atributo DTRS permite expresar cualquier combinación lineal. En nuestras especificaciones lingüísticas hemos usado los tres tipos de mecanismos, dependiendo fundamentalmente, de las necesidades expresivas del fenómeno que se estaba describiendo.

Otro tipo importante de reglas son las léxicas. Desde los inicios de las teorías orientadas léxicamente,[12] el uso de las reglas léxicas se ha ido difundiendo cada vez más. Se usan para establecer relaciones morfológicas (tanto de inflexión, como de derivación), para declarar los vínculos entre entradas léxicas (por ejemplo, para relacionar variaciones en la estructura argumental de los verbos), y para tratar fenómenos típicamente considerados como sintácticos (por ejemplo, la pasiva o la extracción de elementos *qu-* en oraciones de relativo e interrogativas).

A pesar de esta proliferación, no existe una caracterización formal rigurosa de las reglas léxicas en ninguno de los formalismos en que se usan. Por el contrario, a menudo las reglas léxicas se usan sin tener especificadas claramente su sintaxis y su semántica. Por ello, se va generalizando el convencimiento que no se debería abusar de ellas, y que un sistema de tipos con herencia, debería aprovechar el poder relacionar de la 'erarquía para establecer las relaciones entre elementos léxicos. No obstante, las limitaciones impuestas al sistema de tipos y a la jerarquía de herencia (especialmente la ausencia de tipos estructurados y complejos) no facilita (en general, impide) establecer generalizaciones léxicas, ya que la única técnica disponible es la subespecificación (y, aún, limitada a los tipos simples que permitimos). Por lo tanto, resulta inadecuado rechazar totalmente el uso de las reglas léxicas. En cualquier caso, a menudo las reglas léxicas pueden ser modeladas mediante reglas de reescritura con una única hija; de esta manera, no se suponen características especiales para este tipo de reglas.

---

[11] Este es el llamado formato ID-LP (de immediate dominance y linearprecedence).

[12] Desde el punto de vista computacional se puede marcar este inicio con la aparición de LFG (Bresnan, 1982). Para la evolución de la teoría lingüística en este sentido de lexicalización, puede verse Badia (1994).

4. Ejemplificación: la estructura de predicado y argumentos

Uno de los,aspectos centrales en un sistema de procesamiento del lenguaje natural es la llamada estructura argumental, que pretende describir la relación que se establece entre un predicado y sus complementos. Se trata de un nivel de descripción lingüística más abstracto que el de las relaciones puramente sintácticas, en el que se expresan de manera común las relaciones del objeto con el verbo en una oración transitiva activa, las del sujeto con el verbo en una oración pasiva, las de algún complemento con *de* con un nombre deverbal...[13]

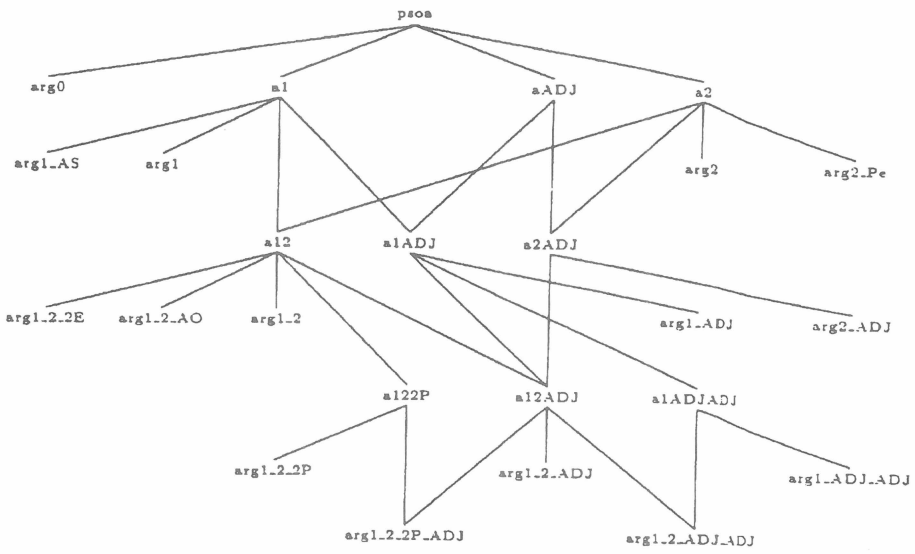
Un examen detallado de las propuestas de HPSG en este aspecto pone de relieve algunas limitaciones importantes para sistemas pensados para tratar textos reales. La codificación de los distintos psoa ("parametrized states of affairs"), con que se representan las estructuras argumentases de los predicados, presupone un sistema de tipos enorme (con todos los predicados léxicos introducidos como tipos distintos, y relacionándose entre sí a través de la jerarquía de tipos), que en la práctica no resulta manejable para los sistemas actuales. La ausencia de una propuesta uniforme y de amplia cobertura para tratar los modificadores de los predicados (similar a la existente para los modificadores de signos nominales) limita en gran medida las posibilidades reales de estas gramáticas, puesto que en la gran mayoría de oraciones reales aparece por lo menos un modificador oracional. En las propuestas de HPSG falta también un tratamiento para los signos nominales predicativos (como, por ejemplo, en *la destrucción de la ciudad).* Para presentar brevemente los resultados obtenidos en el proyecto nos vamos a centrar en estos aspectos, en que HPSG presenta más dificultades.

En primer lugar, para la definición de los tipos predicativos hemos partido del trabajo realizado en el proyecto Eurotra. Allí, en una investigación conducida por Lee Humphreys, se llegó a la caracterización de las relaciones argumentases en base a un sistema mixto entre la codificación puramente sintáctica y la propiamente semántica. Así, los argumentos primero y segundo de una predicación se codifican de una manera no especificada semánticamente (simplemente, como *arg1 y arg2);* mientras que el resto de argumentos se codifica mediante etiquetas relativamente transparentes desde el punto de vista semántica: *arg_2P* para el segundo participante animado, *arg_2E* para el segundo participante inanimado, *arg_PLACE* para los argumentos locativos, etc. Por otra parte cada predicado introduce un tipo que restringe los argumentos que puede tener: así, un verbo estrictamente transitivo introducirá el tipo *arg12,* cuyos únicos complementos van a ser el *arg1* y el *arg2.* Todos los tipos necesarios para tratar las relaciones argumentases pueden representarse en una jerarquía de tipos que es fiel a los principios mencionados en la segunda sección. En la figura siguiente ofrecemos una representación simplificada de la misma.[14]

---

[13] Se trata de un nivel de descripción intermedio entre el sintáctico y el puramente semántica. Fluede verse Badia (1993), para una descripción del mismo y su utilidad en el procesamiento del lenguaje natural.

[14] Las abreviaturas usadas en la figura y no explicadas en el texto son las siguientes: *argAS,* argumento atributo de sujeto; *argAO,* argumento atributo de objeto; *argPe,* argumento que expresa el perceptor (válido, por ejemplo, para el dativo del verbo *parecer); argADJ,* o argumento adjunto, para agrupar esquemáticamente el grupo de argumentos semánticamente llenos y con valor básicamente locativo (lugar, destino, origen ...) .



Nótese que el esquema de la figura debería ser completado con los atributos introducidos en cada nodo; de este modo resultaría más evidente que estos atributos se heredan de arriba a abajo por los distintos nodos de la jerarquía.

En segundo lugar, es importante señalar que el sistema de tipos adoptado presenta un limitación importante, que tiene consecuencias para la labor lexicográfica en este tipo de gramáticas. Un formalismo como el de HPSG, con estructuras de rasgos complejas, permite declarar fácilmente las relaciones que se establecen entre distintos aspectos del signo lingüístico. En este sentido, una de las posibilidades que antes vienen a la mente del lexicógrafo es la de formular explícitamente la relación entre la subcategorización sintáctica y la semántica, es decir, entre la lista de complementos necesarios sintácticamente y la lista de argumentos del predicado. Esta correlación, que puede ser expresada sin ninguna dificultad en las entradas léxicas, no se puede declarar en un sistema de tipos como el propuesto. La razón está en la no admisibilidad de la definición de tipos mediante estructuras de rasgos, puesto que en una entrada léxica el punto común entre la subcategorización sintáctica y la estructura argumental está a una cierta distancia de cada aspecto en particular. Así pues, en un sistema como el propuesto el sistema de tipos debe dar cuenta de cada uno de los dos aspectos en particular y las generalizaciones a nivel léxico se deben formular de otra manera.

En tercer lugar, las dos limitaciones más importantes de HPSG en relación con la estructura argumental (falta de tratamiento de los modificadores de predicados y de los nombres predicativos) pueden ser resueltas de manera paralela, puesto que de hecho las dos resultan de una visión reducida de la relación de predicación. Una pequeña modificación a la semántica de los signos predicativos permite plantear el problema de manera satisfactoria.

La idea, básicamente, consiste en tratar los signos predicativos de acuerdo con las propuestas de la semántica eventiva (Davidson, 1967; Parsons, 1990). El aspecto fundamental de la propuesta consiste en la incorporación de un índice (de valor similar al de una variable cuantificada existencialmente) en la estructura de rasgos que describe la semántica de los signos predicativos, en un claro paralelo con la representación de la semántica de los signos nominales.[15] Así, en la figura siguiente aparece un esquema de la representación de los signos predicativos, que muestra como se integra el índice en el conjunto de la semántica.

$$\left[\begin{array}{ll} \text{ÍNDICE} & \boxed{0}\ \textit{eventualidad} \\ \text{NÚCLEO} & \left[\begin{array}{ll} \text{REL} & \textit{relación} \\ \text{ARG\_EVENTO} & \boxed{0} \\ \ldots & \end{array}\right]_{psoa} \\ \text{RESTRICCIONES} & \left\{ {}_{psoa}[\ \ ] \,\&\, {}_{psoa}[\ \ ] \ldots \right\} \end{array}\right]$$

La incorporación de un índice permite representar fácilmente los complementos de los nombres predicativos (tanto si tienen una interpretación dinámica, como si denotan a un participante en la acción). De hecho, el índice propio de los nombres predicativos hereda información del índice nominal y, también, del índice predicativo (es decir, tiene propiedades de cada uno de ellos: del primero, hereda el género y el número, mientras que hereda del segundo las propiedades de temporalidad y aspectualidad). Por otra parte, al incorporar un índice los modificadores de los signos predicativos pueden ser tratados de una forma totalmente paralela a la de los signos nominales: la ligazón entre la denotación del predicado y la del modificador se establece a través del índice común. De esta manera, casi todos los signos semánticamente llenos introducen un índice y se consigue una representación mucho más uniforme y coherente de la semántica.

5. Conclusiones

Como muestran los pequeños ejemplos de la sección anterior, las decisiones en el diseño fonnal de las especificaciones tienen consecuencias para el tratamiento de los fenómenos lingüísticos. En algunos casos, claramente imponen limitaciones (como en la caracterización de los tipos de relaciones); en otros, por el contrario, permiten toda la expresividad necesaria (como en los últimos aspectos expuestos). En definitiva, se trata de un compromiso entre expresividad, por un lado, y eficiencia o computabilidad, por el otro, que pretende sacar todo el partido posible de los conocimientos actuales sobre computación de gramáticas en estructuras de rasgos tipificadas.

---

[15] Para una discusión detallada de la propuesta, ver Badia y Colominas (1995), donde se presentan los elementos básicos de la propuesta y se muestra la operatividad de la propuesta en varios ejemplos significativos (que incluyen los signos para los nombres predicativos y la incorporación de los modificadores a los signos predicativos).

BIBLIOGRAFÍA

Alshawi et al. (1991), *EUROTRA ET6/1: Rule Formalism and Virtual Machine Design Study,* Comission of the European Union. Luxembourg.

Badia, T. (1993), 'Dependency and Machine Translation', en F. van Eynde (ed.) *Linguistic Issues in Machine Translation.* Pinter. London.

Badia, T. (1994), Lexicografia i models lingüístics: les teories lingüístiques i el léxic, *en Caplletra,* 17; pp. 15

Badia, T. y C. Colominas (1995), 'Propuesta para una representación semántica de la estructura de predicado y argumentos', en *Procesamiento del Lenguaje Natural,* 17, setiembre 1995.

Balari, S. y L. Dini (eds.) (en prensa), *Romance in HPSG.* CSLI. Stanford (Calif.)

Bresnan, J. (ed.) (1982), *The Mental Representation of Grammatical Relations.* MIT Press. Cambridge (Mass.) / London.

Carpenter, B. (1992), *The Logic of Typed Feature Structures.* MIT Press. Cambridge (Mass.) / London.

Daelemans, W.; K. De Smedt y G. Gazdar (1992), 'Inheritance in natural language processing', en *Computational Linguistics, Special Issue on Inherintace, II.*

Davidson, D. (1967). 'The logical form of action sentences', en N. Rescher (ed.) *The Logic of decision and Action* University of Pittsburg Press. Pittsburg.

Gazdar, G.; E. Klein; G. Pullum; y I. Sag (1985), *Generalized Phrase Structure Grammar.* B. Blackwell. Oxford.

Krieger, H.-U. y J. Nerbonne (1991), 'Feature-based Inheritance Networks for Computational Lexicons'. Research Report 31, Deutsches Forschungszentrum für Künstliches Intelligenz. Saarbrücken.

Markantonatou, S. y L. Sadler (eds.) (1994), *Grammatical Formalisms: Isses in Migration.* Office for Official Publications of the European Communities. Luxembourg.

Nerbonne, J.; K. Netter; y C. Pollard (eds.) (1994), *German in Head-driven Phrase Structure Grammar.* CSLI Lecture Notes, 46. CSLI. Stanford (Calif.).

Oakley, B. (1992), 'Eurotra final review panel report. Review Report. Logica. London.

Parsons, T. (1990), *Events in the semantics of English: A Study in Subatomic Semantics.* MIT Press. Cambridge (Mass.)

Pereira, F. C. N. y S. M. Shieber (1 987), *Prolog and Natural Language Analysis.* CSLI Lecture Notes, 10. CSLI. Stanford (Calif.).

Pollard, C. y I. Sag (1987), *Information-based Syntax and Semantics. CSLI* Lecture Notes, 13. CSLI. Stanford (Calif.).

Pollard, C. y I. Sag (1994), *Head-driven Phrase Structure Grammar. CSLI* / University of Chicago Press. Stanford / Chicago.

Pulman, S. (1994) 'Expressivity of lean formalisms', en Markantonatou y Sadler (1994).

Riehemann, S. (1993), 'Word formation in lexical type hierarchy'. SfS Report 02, University of Tübingen. Tübingen.

Russell, G. et al. (1992), 'A practical approach to multiple default inheritance for unification-based lexicons', en *Computational Linguistics, Special Issue on Inheritance, II.*

Shieber, S.M. (1986), *An Introduction to Unifiaction-based Approaches to Grammar.* CSLI Lecture Notes, 4. CSLI. Stanford (Calif.).

Steiner, E. (ed.) (1991), *Machine Translation. Special Issue on Eurotra I-II.*

Touretzky, D.; J. Horty; y R. Thomason (1989), 'A clash of intuitions: The current state of nonmonotonic multiple inheritance systems', en *Proceedings of IJCAI '89.* Commission of the European Union. Luxembourg.

Zajac, R. (1993), 'Issues in the design of a language for representing linguistic information based on inheritance and feature systems', en Briscoe et al. *(eds.) Inheritance, Defaults and the Lexicon.* Cambridge University Press. Cambridge.