ONOMÁZEIN 26 (2012/2): 13-48

En defensa del procesamiento del lenguaje natural fundamentado en la lingüística teórica

In defence of a linguistic-aware approach to natural language processing

Carlos Periñán Pascual
Universidad Politécnica de Valencia
España

Resumen

A pesar de que podríamos ubicar el procesamiento del lenguaje natural entre la lingüística aplicada y la inteligencia artificial, el papel que ha desempeñado la lingüística teórica a lo largo de la historia de esta disciplina ha sido generalmente poco notorio. Uno de los objetivos de este artículo es desgranar las causas de esta malograda simbiosis entre las investigaciones de lingüistas e informáticos, donde los enfoques probabilísticos han ido gradualmente relegando los modelos lingüísticos a un segundo plano, en el mejor de los casos. A pesar de este desalentador panorama, FunGramKB, una base de conocimiento particularmente útil para sistemas que requieran la comprensión del lenguaje, sirve para ilustrar cómo actualmente la lingüística teórica y la ciencia cognitiva pueden contribuir al desarrollo de un proyecto de ingeniería del conocimiento.

Palabras clave: ingeniería lingüística; procesamiento del lenguaje natural; lingüística computacional; FunGramKB.

Abstract

Although natural language processing can be deemed as a discipline between applied linguistics and artificial intelligence, theoretical linguistics has played a remarkably minor role in this field of research. One of the goals of this paper is to portray the reasons of the failed symbiosis between linguists' research and that of computer scientists, where probabilistic approaches haven been steadily overshadowing linguistic models. In spite of this discouraging scenario, FunGramKB, a knowledge base particularly designed for natural language understanding systems, serves to illustrate

Afiliación: Carlos Periñán Pascual: Departamento de Lingüística Aplicada, Universidad Politécnica de Valencia.
Correo electrónico: jopepas3@upv.es
Dirección postal: Paranimf, 1-46730 Gandia, Valencia, España.

Fecha de recepción: enero de 2012
Fecha de aceptación: octubre de 2012

how a language-aware and cognitively-plausible approach to human-like processing can contribute to the development of enhanced knowledge-engineering projects.

Keywords: *language engineering; natural language processing; computational linguistics; FunGramKB.*

Anytime a linguist leaves the group the recognition rate goes up.
Frederick Jelinek (cf. Jurafsky y Martin, 2009: 83)

1. Introducción

El procesamiento del lenguaje natural (PLN) es un campo del conocimiento al que han contribuido a su desarrollo disciplinas como la lingüística, la informática, la ciencia cognitiva y la ingeniería electrónica, esta última más estrechamente relacionada con las tecnologías del habla. A lo largo de la historia del PLN los dos enfoques principales de investigación adoptados han sido los paradigmas simbólico y estadístico[1]:

(i) El enfoque simbólico se caracteriza por la construcción de sistemas que almacenan explícitamente los hechos lingüísticos (p.ej. fonológicos/fonéticos, morfológicos, sintácticos, semánticos, pragmáticos o discursivos) a través de esquemas de representación del conocimiento, desarrollados principalmente de forma manual.

(ii) El enfoque estadístico se caracteriza por la construcción de sistemas que no almacenan explícitamente el conocimiento lingüístico o del mundo, sino que aplican técnicas matemáticas sobre extensos córpora informatizados con el fin de inferir dicho conocimiento.

Las tecnologías del lenguaje se apoyan en modelos formales del conocimiento de la lengua (p.ej. máquinas de estado[2], sistemas de reglas, lógica, o modelos probabilísticos, entre los más importantes), pero el tiempo ha demostrado que no siempre un

[1] Otro enfoque de investigación en PLN es el conexionista, en forma de redes neuronales. Léase Christiansen y Chater (1999) para una exposición detallada de la influencia de este enfoque sobre el tratamiento computacional del lenguaje.

[2] Éste es el caso de los autómatas y transductores de estados finitos, utilizados principalmente para los análisis morfológico y sintáctico y el procesamiento del habla.

sistema del PLN conlleva la adopción de una teoría lingüística, ni siquiera en los sistemas de procesamiento simbólico. De hecho, existen programas del PLN que funcionan perfectamente sin necesidad de estar basados en alguna teoría lingüística, pero se trata en realidad de programas *engañosamente inteligentes*. Por el contrario, las aplicaciones informáticas robustas requieren una base teórica que pueda servir de fundamento al comportamiento deseado (Halvorsen, 1988). Tras presentar las diversas etiquetas terminológicas que surgen con la convergencia de la informática y la lingüística, el apartado 3 realiza un breve recorrido por la historia del PLN, donde nos centraremos en el papel que ha desempeñado la lingüística teórica en esta disciplina.

2. Un caos terminológico

El tratamiento computacional del lenguaje, escrito u oral, ha experimentado tal evolución a lo largo de sus 70 años de historia que ha dado lugar a etiquetas como lingüística computacional, PLN, tecnologías lingüísticas, ingeniería lingüística, industrias de la lengua o lingüística informática. En este apartado explicamos por qué la lingüística informática y la ingeniería lingüística son campos de investigación propios de la lingüística y la informática respectivamente, mientras que términos como lingüística computacional, PLN y tecnologías lingüísticas suelen referirse a una misma área de conocimiento aunque enfatizando aspectos diferentes dependiendo del punto de vista de la disciplina que la estudie.

En un sentido estricto, cualquier actividad que implique un análisis o generación de la lengua utilizando el ordenador puede considerarse como lingüística computacional. Debido a la finalidad práctica de las investigaciones, los lingüistas prefieren hablar de la lingüística computacional como un área de conocimiento dentro de la lingüística aplicada. En cambio, debido a la posibilidad de desarrollar sistemas de computación que simulen algún aspecto de la capacidad lingüística del ser humano, los informáticos consideran la lingüística computacional como una rama de la inteligencia artificial, al igual que los sistemas expertos o la robótica, en cuyo caso prefieren hablar de PLN. Por tanto, mientras la lingüística computacional se centra más en la modelización del conocimiento lingüístico para posibilitar la construcción de sistemas computacionales que analicen y/o

generen textos en lenguaje natural, el PLN hace un mayor énfasis en la búsqueda de soluciones a los problemas que plantea la lingüística computacional, pero en el marco de aplicaciones concretas: p.ej. recuperación y extracción de información, resúmenes automáticos, traducción mecánica, etc. (Martí Antonín, 2003). Finalmente, se prefiere el término tecnología del lenguaje cuando describimos cómo esas aplicaciones del PLN mejoran la comunicación en la sociedad de la información por encima de las barreras que impone la distancia, el uso de lenguas distintas o el modo en que tiene lugar la comunicación (Martí Antonín y Llisterri, 2001).

Un tipo de investigación marcadamente diferente se encuentra en la *lingüística informática*, la cual está orientada hacia el desarrollo de programas de apoyo en los estudios realizados en los diversos campos de la filología (Martí Antonín, 2003). La principal finalidad de este tipo de programas es la extracción de datos estadísticos, concordancias, colocaciones, etc., a partir de los córpora textuales. Por ejemplo, obtener información estadística sobre la aparición de determinadas unidades lingüísticas resulta útil tanto en la descripción de la lengua como en la selección del vocabulario y las construcciones más usuales para la elaboración de programas de enseñanza de lenguas (Moure y Llisterri, 1996).

Un área muy diferente donde convergen lingüística e informática la encontramos en la *ingeniería lingüística*, también conocida como *industrias de la lengua*. Estos términos referencian un campo de investigación todavía en desarrollo que sirve para describir aquellos productos comerciales en los que se aplican técnicas propias del PLN. Por tanto, este tipo de disciplina se caracteriza por estar estrechamente vinculado al mundo empresarial, el cual desarrolla y comercializa una serie de productos dirigidos a unos usuarios finales no especializados que poseen unas necesidades específicas (Moure y Llisterri, 1996).

En este artículo nos centramos en los aspectos teórico-prácticos de las investigaciones sobre el tratamiento computacional del lenguaje, por lo cual hemos optado por aglutinar el PLN y la lingüística computacional bajo la etiqueta del primero, ya que ambos representan en realidad los dos lados de una misma moneda[3].

[3] No obstante, las tecnologías del habla no son consideradas en este estudio.

3. La lingüística teórica y el PLN

3.1. Los años 40 y 50

La investigación en PLN se remonta a los años 40, siendo la traducción automática una de sus primeras aplicaciones. En 1949, el interés por la traducción automática despertó gracias a un famoso memorándum del matemático Warren Weaver, en el cual se propone la aplicación de las técnicas del descifra-miento criptográfico, los métodos estadísticos y la teoría de la información (Shannon, 1948) para la traducción automática de textos con el fin de "solucionar los problemas mundiales de traducción". En realidad, esta idea no era tan original como parecía, ya que una de las tareas habituales de un ordenador ha sido siempre la *traducción* del código escrito en un lenguaje informático de alto nivel al lenguaje máquina, el cual se limita a una secuencia de ceros y unos.

A principios de los años 50 existieron numerosos grupos de investigación sobre traducción automática, pero todos los trabajos se caracterizaban por una gran ingenuidad en la manera de abordar el tema. Concebían la lengua como un código y pensaban que lo único que tenían que hacer era descifrar ese código de una lengua fuente a una lengua destino, dando como resultado la construcción de los sistemas de traducción directa. En otras palabras, se partía de la idea de que las diferencias entre lenguas se basaban en sus vocabularios y en el orden de las palabras dentro de la oración. Evidentemente, estos sistemas de traducción no se fundamentaban en ninguna teoría lingüística, sino dependían más bien de diccionarios muy bien desarrollados además de un analizador morfológico que permitía presentar traducciones gramaticalmente aceptables. Las traducciones resultantes estaban tan plagadas de errores, muchos de ellos provocados por la ambigüedad léxica, que requerían una profunda postedición del texto de salida.

3.2. Los años 60

A partir de esta década, empieza a consolidarse el enfoque simbólico en las investigaciones del PLN, gracias a la contribución de dos fenómenos centrales, como apuntan Jurafsky y Martin (2009): el nacimiento de la Inteligencia Artificial y la Gramática Generativo-Transformacional.

3.2.1. El nacimiento de la Inteligencia Artificial

Uno de los hitos en las investigaciones en PLN en la década de los 60 fue el nacimiento de la inteligencia artificial, la cual tuvo lugar en un seminario de dos meses organizado por John McCarthy y celebrado en el verano de 1956 en el Dartmouth College en Hanover, New Hampshire. Las discusiones intelectuales que se mantuvieron en este seminario, en el que participaron los investigadores pioneros en inteligencia automática –tales como Marvin Minsky, Nathaniel Rochester y Claude Shannon, entre otros muchos–, sirvieron para colocar los pilares de la nueva disciplina de la *Inteligencia Artificial*. Uno de los aspectos del problema de la inteligencia artificial que se trató fue la posibilidad de que un ordenador pudiera ser programado para utilizar una lengua, en concreto el inglés. Especulaban con la idea de que la mayor parte del pensamiento humano consistía en la manipulación de palabras según un conjunto de reglas de razonamiento. Por tanto, y debido a la marcada formación matemática de los organizadores, el centro de interés de estas actividades radicó principalmente en el desarrollo de sistemas basados en el razonamiento lógico.

Con respecto a la comprensión automática del lenguaje, uno de los sistemas de diálogo más representativos en la década de los 60 fue ELIZA (Weizenbaum, 1966), el cual simulaba ser un psicoterapeuta que mantenía una conversación con el usuario. El algoritmo que simulaba la inteligencia de ELIZA consistía básicamente en leer una oración de entrada, buscar la presencia de un patrón a modo de plantilla predefinida (i.e. constantes y variables), el cual se activaba a través de una palabra clave, y finalmente transformar la entrada en una respuesta. Por tanto, la base de conocimiento consistía en un conjunto de reglas de transformación[4], cada una de las cuales describía (a) una serie de posibles patrones y (b) un conjunto de posibles respuestas asociadas. Con el fin de que las respuestas de ELIZA parecieran lo más naturales posibles, el sistema elegía una respuesta al azar de las respuestas asociadas al patrón, incorporando igualmente expresiones utilizadas en el propio texto de entrada. En esta década, los sistemas comprendían las preguntas sólo en el caso de que tuviera lugar una coincidencia de patrones y las respuestas

[4] Como reconoce el propio Weizenbaum (1966), el término *transformación* se utiliza en un sentido genérico, desprovisto de toda connotación chomskiana.

estuvieran explícitamente almacenadas de antemano en la base de conocimiento. Por tanto, estos sistemas no se fundamentaban en ningún modelo semántico. Por ejemplo, en el caso de ELIZA, si un usuario introducía la oración "I am BLAH", ésta podía ser transformada a la respuesta "How long have you been BLAH", independientemente del significado de BLAH. Sólo se trataba de crear la ilusión de que la máquina podía interactuar con el hombre. Al igual que otros muchos programas de diálogo de esta década, ELIZA no realizaba prácticamente ningún análisis de las estructuras lingüísticas presentes en las entradas textuales, ya que la complejidad del lenguaje natural se trataba por medio del reconocimiento de patrones. En ese tipo de aplicaciones los avances investigadores se centraban en mejorar la inferencia de información en lugar de proporcionar un tratamiento adecuado al procesamiento lingüístico (Ramsay, 2004).

3.2.2. La Gramática Generativo-Transformacional

De forma paralela a los trabajos en Inteligencia Artificial, los investigadores en PLN intentaron desarrollar gramáticas oracionales y analizadores con el fin de resolver los problemas de ambigüedad sintáctica y semántica que presentaban los procesamientos de la década anterior. En esta búsqueda de una caracterización explícita del lenguaje pensaron que las teorías lingüísticas del momento podían aportarles las respuestas que necesitaban, principalmente en torno a la representación sintáctica. Fue, por ello, que la Gramática Generativo-Transformacional entró en escena.

En 1957, Chomsky publicó *Syntactic Structures*, presentando a la comunidad lingüística un modelo generativo del lenguaje al que posteriormente el propio autor realizó diversas modificaciones con la publicación de *Aspects of the Theory of Syntax* (1965), dando lugar a lo que se conoció como la *teoría estándar* de la Gramática Generativo-Transformacional. En este modelo gramatical, donde la semántica y la fonología se relegan a meros componentes interpretativos, la parte central es la sintaxis, la cual permite describir los aspectos regulares del lenguaje. En realidad, la Gramática Generativo-Transformacional no se ideó pensando en el PLN. De hecho, los lingüistas generativistas nunca concibieron el PLN como un escenario donde probar su teoría lingüística, ya que pensaban que las realizaciones lingüísticas estaban íntimamente conectadas con las intuiciones

de los hablantes en lugar de con los procesos computacionales (Wilks, 2005). No obstante, provocó el interés inicial de muchos investigadores en PLN porque era un modelo formal del lenguaje que teóricamente facilitaba su implementación computacional.

3.3. Los años 70

3.3.1. La reacción al paradigma generativo estándar

La influencia del paradigma lingüístico dominante de los años 60 disminuyó notablemente en la década de los 70, ya que se demostró que la Gramática Generativo-Transformacional era inadecuada para el PLN: la teoría mostró tanto interés en el procesamiento sintáctico que no prestaba atención alguna al tratamiento semántico. La estructura profunda era una estructura sintáctica que, sólo después de ser generada, recibía una interpretación semántica. Los modelos generativistas de esta década intentaron integrar la semántica en la teoría sintáctica, lo cual terminó germinando con las teorías lexicistas en los años 80. Chomsky (1970) y Jackendoff (1972) propusieron una "semántica interpretativa" dentro del seno de lo que se conoció como *Teoría Estándar Ampliada*, mientras Gruber y Fillmore llevaron a cabo las primeras iniciativas de una teoría de la *semántica generativa*.

Por otra parte, y como reacción a las teorías generativistas, surgieron diversos modelos de orientación funcionalista[5]. Según este paradigma, la lengua se concibe como un objeto funcional, i.e. como un instrumento de comunicación. Una de las primeras teorías lingüísticas funcionales que se expandió al campo del PLN fue la Gramática Sistémica Funcional, la cual, a pesar de ser formulada inicialmente en la década de los 60 (Halliday, 1961, 1967), se empezó a desarrollar significativamente sólo en los años 70[6].

---

[5] Léase Butler (2003a, 2003b) y Gonzálvez-García y Butler (2006) para una descripción detallada de los fundamentos metodológicos de los modelos asociados al paradigma funcional.

[6] La teoría de Halliday no es sólo un modelo funcional del lenguaje, el cual permite explicar por qué elegimos determinados rasgos lingüísticos cuando utilizamos la lengua, sino también una teoría social del lenguaje, donde el contexto situacional establece una estrecha relación entre la lengua y el mundo extralingüístico.

3.3.2. La comprensión del lenguaje en Inteligencia Artificial

La Inteligencia Artificial de los años 70 se orientó principalmente hacia el desarrollo de sistemas de comprensión del lenguaje natural. La Escuela de Yale, i.e. Roger Schank y sus colaboradores (Schank, 1972, 1975; Schank y Abelson, 1977), lideró las investigaciones en este campo, incorporando la teoría del guión como base para un modelo dinámico de la memoria[7]. Este modelo de comprensión del lenguaje, el cual resultó ser una importante influencia en la semántica y la representación del conocimiento, se basó en el formalismo de la Dependencia Conceptual (Schank, 1972), i.e. grafos que permiten representar conceptualmente un texto de entrada a partir de la descomposición semántica de los verbos –usando la Gramática de Casos de Fillmore (1968) centrada en torno a una serie de acciones primitivas– y la inferencia de la información basada en estos primitivos. La teoría de la Dependencia Conceptual puede concebirse como una combinación entre las redes semánticas y los marcos. En esta década se desarrollaron diversos programas informáticos con el propósito de demostrar que esta teoría de la memoria dinámica podía replicar el proceso de comprensión en el ser humano, p.ej. MARGIE (Schank y otros, 1973), SAM (Cullingford, 1978) o PAM (Wilensky, 1978).

Los sistemas resultantes de las investigaciones de la Escuela de Yale no hicieron uso de un nivel sintáctico intermedio en el procesamiento lingüístico[8]. En cambio, SHRDLU (Winograd, 1972) –un sistema de comprensión automática del inglés que podía manipular bloques de juguete sobre una mesa a partir de unas órdenes, además de poder ser interrogado sobre el escenario resultante– incorporó la teoría funcional de Halliday. Este sistema utilizaba información semántica y del contexto para comprender el discurso, ya que se basaba en la idea de que no es posible construir un sistema informático razonablemente inteligente a menos que pueda *comprender* el tema sobre

[7] No obstante, sus teorías sobre la comprensión del lenguaje natural se desarrollaron más profundamente en la siguiente década (Schank y Riesbeck, 1981; Schank, 1982a, 1982b, 1986).

[8] En el modelo de la Dependencia Conceptual, el objetivo del procesamiento es obtener una representación semántica del aducto, como demuestra el *English Language Interpreter* (Riesbeck, 1975; Riesbeck y Schank, 1978), utilizando la sintaxis sólo cuando el procesamiento semántico lo requiere.

el cual está trabajando, lo cual implicaba proporcionarle un modelo detallado del conocimiento que requería[9]. No obstante, otras interfaces de diálogo persona-máquina en esta década no requirieron un complejo formalismo de representación del significado con el fin de simular la interacción comunicativa. Éste es el caso de PARRY (Colby, 1973), un sistema que permitía simular las respuestas de un paciente paranoico que sufría la psicosis de que lo estaba persiguiendo la mafia. Aunque inspirado en el trabajo de ELIZA, PARRY no repetía las palabras de sus entrevistadores, sino que contribuía a la conversación de forma fluida además de reaccionar tal y como lo haría un paranoico.

A pesar de que SHRDLU y PARRY coincidían en ser sistemas basados en el diálogo, las diferentes concepciones de sus autores sirven para representar dos enfoques muy diferentes de entender el PLN. El primero empleaba un análisis lingüístico basado en un modelo teórico gramatical incorporando igualmente conocimiento del mundo sobre el cual se aplicaba un razonamiento basado en la lógica. En cambio, el segundo apostaba por el simple reconocimiento de patrones en la superficie de la entrada textual y un módulo de interpretación-acción cuyas reglas de producción permitían recrear el modelo de paranoia.

3.4. Los años 80

El debate entre la semántica interpretativa y la semántica generativa durante la década de los 70 contribuyó a reconsiderar el papel del lexicón en el procesamiento del lenguaje, convirtiéndose este lexicón en foco de interés de la lingüística de esta década. Por ejemplo, la Teoría de la Rección y el Ligamiento (Chomsky, 1981) se distancia principalmente de la teoría estándar de la Gramática Generativo-Transformacional con respecto a la autonomía que se le concedía al componente sintáctico. La sintaxis empezó a ponerse en relación con el léxico, de tal forma que las propiedades léxicas podían contribuir a determinar la forma sintáctica de una oración. Además, el componente transformacional

[9] Dentro del marco de la lingüística computacional, la Gramática Sistémica Funcional ha tenido un especial impacto desde los años 80 en los sistemas de generación lingüística, p.ej. PENMAN (Mann, 1983), y especialmente en los años 90, p.ej. Genesys (Fawcett y Tucker, 1990), TECHDOC (Rösner y Stede, 1994), Gist (Not y Stock, 1994), WAG-KRL (O'Donnell, 1994), DRAFTER (Paris y Vander Linden, 1996) y KPML (Bateman, 1997).

se redujo al principio de traslado de constituyentes de un lugar a otro de la oración con el fin de reordenarlos.

El paradigma simbólico se caracterizó durante esta década por la aparición de *teorías lexicistas*, p.ej. la Gramática Léxico-Funcional (Kaplan y Bresnan, 1982), la Gramática de Estructura Sintagmática Generalizada (Gazdar y otros, 1985) o la Gramática de Estructura Sintagmática Nuclear (Pollard y Sag, 1987), las cuales reducían el papel de la gramática en aras de dotar de mayor importancia al léxico. La mayoría de estos modelos lexicistas se fundamentaba en el supuesto de que la estructura argumental de un verbo está directamente determinada por sus propiedades léxicas. En otras palabras, la entrada léxica de un verbo determina su comportamiento sintáctico[10]. Estas teorías lexicistas, aunque todavía pertenecían al paradigma generativo, rechazaban todo tipo de transformaciones, por lo cual en el proceso de análisis no se distinguía entre estructura profunda y estructura superficial. Evidentemente, la ausencia de reglas transformacionales se compensó con un modelo léxico más rico, i.e. estructuras léxicas más complejas. Desde el punto de vista formal, a todas estas teorías lexicistas, junto a otras como la Gramática de Unificación Funcional (Kay, 1985)[11] o la Gramática Categorial de Unificación (Uszkoreit, 1986; Karttunen, 1989), también se les conocieron como *gramáticas basadas en la restricción* o *gramáticas de unificación*, donde la estructura sintáctica de las lenguas se representa mediante gramáticas sintagmáticas independientes del contexto aumentadas con el uso de rasgos sobre los que se aplican las operaciones de subsunción y unificación[12]. En realidad, las gramáticas de unificación están

[10] De hecho, la tendencia actual en la lingüística es identificar los diversos argumentos que configuran la estructura argumental del verbo a partir de la semántica del evento (cf. Levin y Rappaport Hovav, 2005). En este sentido, Vendler (1967) fue uno de los primeros que propuso una clasificación de los eventos, basándose en una serie de propiedades aspectuales, o *Aktionsart*. La tipología de Vendler ha servido como fundamento sobre el cual otros investigadores han desarrollado sus teorías lingüísticas, como es el caso de la Gramática del Papel y la Referencia (Van Valin y LaPolla, 1997; Van Valin, 2005).

[11] La Gramática de Unificación Funcional se inspiró en la Gramática Sistémica Funcional de Halliday. A pesar de que sus formalismos rotacionales son bastante diferentes, ambos comparten muchos supuestos sobre el lenguaje y la gramática (cf. Kasper, 1987).

[12] Estos rasgos suelen representarse a través de matrices atributo-valor, aunque una representación alternativa son los grafos acíclicos dirigidos, i.e. máquinas de estados finitos.

muy extendidas actualmente en aquellos proyectos de ingeniería lingüística cuya información se almacena en forma de rasgos, ya que combinan "la flexibilidad de las redes semánticas con el poder expresivo y la capacidad de inferencia de la programación lógica" (Moreno Ortiz, 2000).

Finalmente, esta década también fue testigo del auge de los modelos probabilísticos, principalmente en las tecnologías del habla, el etiquetado gramatical, el análisis sintáctico y la semántica. Mientras los enfoques simbólicos fueron utilizados para tratar los problemas más significativos del PLN, los enfoques estadísticos servían como complemento a los enfoques simbólicos.

3.5. Los años 90

En esta década tuvo lugar un fuerte resurgimiento de las tendencias empiristas, no sólo con respecto al análisis de datos lingüísticos sino principalmente en la aplicación de métodos estadísticos al PLN. El paradigma estadístico fue convirtiéndose progresivamente en el estándar de numerosos campos del PLN. Por ejemplo, Brown y otros (1990) fueron los primeros que aplicaron a la traducción automática métodos estadísticos utilizados en el reconocimiento automático del habla. El problema que empezó a atribuirse al paradigma simbólico fue su incapacidad de proporcionar de forma flexible un tratamiento adecuado a (i) un *input* defectuoso (p.ej. una oración elíptica o agramatical) o (ii) una realización lingüística nueva. Los sistemas basados en técnicas estadísticas pueden ser más robustos en ambas situaciones, siempre y cuando se les entrene con un repositorio de datos suficientemente pertinente. Por ejemplo, el procesamiento sintáctico puede ser realizado a través de técnicas de aprendizaje automático que tomen un corpus anotado para el entrenamiento del sistema y que a partir de dicho corpus puedan ser inferidas las construcciones lingüísticas sin necesidad de escribir muchas reglas gramaticales. El racionalismo de los métodos basados en la codificación manual de reglas dejó paso a métodos probabilísticos y de aprendizaje automático. Pero ¿qué ocurrió en esta década que provocó que numerosas líneas de investigación sobre el PLN empezaran a basar sus trabajos en los modelos estocásticos? Las causas de este cambio en la tendencia investigadora suelen atribuirse a los siguientes tres fenómenos (Liddy, 2001):

(i) la disponibilidad de extensos córpora textuales que pueden ser procesados por el ordenador,

(ii) los avances en *hardware*, con ordenadores dotados de más memoria, mayor velocidad de procesamiento y mayor capacidad de almacenamiento, y

(iii) la llegada de Internet, lo cual favorece no sólo la diseminación del conocimiento especializado sino también la accesibilidad de los recursos lingüísticos.

Estos tres factores cambiaron drásticamente el panorama del PLN. Hasta los años 90, se crearon principalmente prototipos de laboratorio más o menos sofisticados, donde los sistemas se basaban en complejos modelos formales teóricos. A partir de los años 90, en cambio, los trabajos en PLN se enfocaron hacia la ingeniería lingüística. Por ejemplo, la idea de producir un análisis sintáctico completo y profundo de un texto de entrada fue perdiendo interés, debido a la necesidad inmediata de obtener soluciones realistas. En el caso de la clasificación documental, por ejemplo, la identificación de palabras clave sólo precisaba el análisis de grandes cantidades de texto que permitieran al ordenador aprender conocimiento de forma automática mediante técnicas de inferencia. Por tanto, un análisis sintáctico más superficial era suficiente, el cual delimitara las oraciones en sintagmas carentes de estructura interna.

No obstante, el predominio de los enfoques probabilísticos en esta década no implicó un abandono completo de los sistemas simbólicos, sino más bien un cambio en las prioridades investigadoras. La influencia del enfoque lexicista de los años 80 fue incrementándose durante esta década, dejando en el componente sintáctico unas pocas reglas generales. De acuerdo con Hanks (2003), trabajos como FrameNet (Fillmore y Atkins, 1992, 1994) y el Lexicón Generativo (Pustejovsky, 1991, 1995) representaron las investigaciones más destacadas en lexicografía computacional.

Además, la década de los 90 fue testigo de la aparición de un nuevo tipo de recurso lingüístico en el ámbito del PLN: las ontologías. De hecho, la ontología es actualmente uno de los componentes centrales en una base de conocimiento para el PLN. En el campo de la informática, la ontología se define como un inventario del tipo de cosas que existen en un dominio desde la perspectiva de una persona que habla sobre ese dominio (Sowa, 2000). Por tanto, las ontologías tienen como objetivo presentar el conocimiento compartido por una comunidad acerca de un

dominio. Con este fin, diseñar una ontología implica determinar el conjunto de categorías semánticas que refleje adecuadamente la organización conceptual del dominio sobre el que el sistema debe trabajar, optimizando la cantidad y calidad de la información almacenada (Lenci, 2001). Las principales razones para utilizar una ontología en un sistema del PLN siguen siendo las siguientes (Bateman, 1991; Nirenburg y otros, 1996):

(i) almacenar el conocimiento del mundo y permitir que los lexicones de diferentes lenguas compartan ese mismo conocimiento,

(ii) realizar inferencias sobre el conocimiento del mundo a partir de los significados de las unidades léxicas, y

(iii) proporcionar una base para la construcción de una interlingua, la cual se utilice para la representación del significado de un texto de entrada o salida

Desde principios de esta década, los *sistemas basados en el conocimiento* cobraron cada vez mayor fuerza, especialmente en el campo de la traducción automática (Farwell y Wilks, 1991; Mitamura y otros, 1991; Nirenburg y otros, 1992; Onyshkevych y Nirenburg, 1995; Palmer y Wu, 1995). En estos casos, como explicaron Onyshkevych y Nirenburg (1992), la comprensión se modela por medio de la representación del análisis del texto de entrada a través de un lenguaje formal, cuyos átomos se interpretan en términos de una ontología. Por tanto, las unidades léxicas y sintácticas del texto de entrada se hacen corresponder con los elementos del lenguaje formal de representación. El lexicón no sólo contiene información sobre las propiedades morfológicas y sintácticas de las palabras, sino también contiene proyecciones sobre la ontología con el fin de describir el significado léxico. Por tanto, el panorama de la semántica léxica estuvo marcado por dos tendencias investigadoras dentro de la lingüística computacional (Nirenburg y Levin, 1992):

(i) la semántica léxica orientada a la sintaxis, la cual buscaba describir las propiedades semánticas de las palabras a partir de las cuales podía predecirse su comportamiento sintáctico, como ha ocurrido en la mayoría de las teorías lexicistas, y

(ii) la semántica léxica orientada a la ontología, donde el significado del texto se infería a partir de un modelo del mundo, u ontología, que se construía independientemente de la lengua, pero que se ponía en correspondencia con el lexicón.

Actualmente, los sistemas del PLN basados en el conocimiento suelen adoptar uno solo de estos dos enfoques, aunque

estos autores recomendaron que ambos enfoques coexistieran en un mismo sistema computacional, ya que sólo uno no sería suficientemente efectivo. Más concretamente, sugirieron que el modelo de semántica léxica orientada a la ontología sirviera para vincular el modelo de semántica léxica orientada a la sintaxis de una lengua determinada con una representación del significado textual independiente de la lengua.

3.6. El PLN en el siglo XXI

Desde los años 90 hasta la actualidad, los enfoques estadísticos han dominado las investigaciones en PLN, todo ello propiciado por la disponibilidad de numerosos recursos computacionales que permiten tratar los fenómenos lingüísticos en un contexto real. Los córpora son ahora la fuente canónica de datos lingüísticos más importante con la que los sistemas del PLN pueden trabajar, pero su explotación es tal que "se pretende conseguir la mayor efectividad posible incluso a expensas de una clara fundamentación lingüística teórica" (Moure y Llisterri, 1996). De hecho, la mayoría de las ponencias en congresos internacionales sobre PLN tratan sobre soluciones de ingeniería a problemas prácticos, pero casi nadie se enfrenta a temas fundamentales en lingüística desde el marco del PLN, p.ej. la representación semántica del significado construccional. En definitiva, las investigaciones actuales en ingeniería lingüística no se fundamentan en la lingüística, sino en la estadística y la teoría de las probabilidades[13], incluso a costa de obtener soluciones *sucias* (Ferrari, 2004), i.e. soluciones menos plausibles desde el punto de vista teórico. Como advierte Wintner (2009), esta situación no deja de resultar bastante paradójica, especialmente si lo comparamos con lo que ocurre en otras áreas de la ingeniería aplicada: p.ej. la ingeniería química exige conocimientos en química y los ingenieros biomédicos deben estudiar biología y medicina. ¿Cuáles son las razones por las cuales la ingeniería lingüística parece no necesitar a la lingüística teórica? ¿Por qué los ingenieros informáticos no recurren a las investigaciones de los lingüistas a la hora de diseñar un sistema del PLN? Wintner (2009) presentó muy brevemente los posibles tres factores que

[13] Por ejemplo, los actuales traductores de Google no se fundamentan en ninguna teoría lingüística, sino se basan en el análisis estadístico de los textos, por lo cual carecen de componentes gramaticales y lexicones.

han propiciado esta situación, los cuales ampliamos con más detalle en los próximos párrafos.

En primer lugar, se suele argumentar que los sistemas fundamentados en teorías lingüísticas no satisfacen las necesidades del mundo real. Ya desde la década de los 90, una de las objeciones de la ingeniería lingüística hacia la lingüística computacional es que se había producido mucho trabajo teórico, pero ninguna aplicación práctica. Según Ferrari (2004), existen dos razones principales por las cuales la lingüística teórica poco tiene que ver con la realidad lingüística:

(i) Los modelos lingüísticos se centran en el estudio de la competencia lingüística, mientras que los ingenieros lingüísticos tratan los fenómenos de la actuación.

(ii) Los modelos lingüísticos no pueden modelar todo el conocimiento lingüístico, de ahí que cubran un número reducido de fenómenos.

Sin embargo, a estas dos posibles causas de la supuesta inutilidad de la lingüística teórica podemos esgrimir diversos contraargumentos. Con respecto a la cuestión (i), no es cierto que todas las teorías gramaticales centren su interés en la competencia lingüística. Éste es el caso de los modelos funcionales del lenguaje, los cuales conciben la lengua como un objeto funcional, i.e. como un instrumento de comunicación, siendo de especial relevancia aquellas que centran su interés en las funciones pragmáticas del lenguaje (cf. Halliday, 1973; Dik, 1989; Van Valin y LaPolla, 1997; Van Valin, 2005; Hengeveld y Mackenzie, 2008). Con respecto a la cuestión (ii), la ausencia de una determinada palabra en el lexicón o de una determinada construcción gramatical en la gramática, puede seguir permitiendo un procesamiento adecuado si utilizamos métodos estadísticos o técnicas de aprendizaje automático sobre córpora con el fin de adquirir automáticamente el conocimiento lingüístico del que carece nuestro sistema. Por tanto, los enfoques simbólico y estadístico pueden coexistir perfectamente con el fin de construir un sistema más robusto, i.e. suplir las carencias de un modelo con las ventajas del otro.

En segundo lugar, el PLN es un campo de investigación de naturaleza aplicada, por lo cual sus objetivos se orientan en definitiva hacia la construcción de aplicaciones informáticas. Por ello, las instituciones, los organismos y las empresas que financian los proyectos quieren obtener resultados a muy corto plazo, generalmente no superior a dos años, lo cual no deja

lugar a la investigación básica a largo plazo, como suele ser el caso de la lingüística teórica.

En tercer lugar, las teorías lingüísticas se han vuelto tan "oscuras, barrocas y egocéntricas" (Wintner, 2009) que resultan muy poco atractivas para los informáticos. Por una parte, oscuras y barrocas porque los lingüistas no terminan de formalizarlas, y sólo lo formalizable puede ser programable. De hecho, existe muy poco interés por parte de la lingüística teórica de producir gramáticas que puedan ser procesadas, quizás por la propia naturaleza de la lingüística teórica, ya que "los investigadores dedicados a empresas teóricas sospechan de la excesiva solicitud al rendimiento práctico […] de las tecnologías; mientras que quienes se afanan en esta cara de la realidad critican toda la aproximación teórica por su desvinculación de los problemas inmediatos" (Moure y Llisterri, 1996). En general, los lingüistas no se preocupan mucho, o incluso se muestran reticentes, de que sus consistentes modelos teóricos puedan tener aplicabilidad alguna en los sistemas del PLN. Como consecuencia de todo ello, los informáticos han quedado tan frustrados con la lingüística teórica que han terminado por abandonarla, o incluso por despreciarla[14].

¿Qué debemos hacer cuando la tendencia investigadora no facilita la incorporación de nuestras propias investigaciones, a sabiendas que nuestras aportaciones podrían mejorar la calidad de los sistemas? En esta situación, la postura del lingüista debe ser la de ayudar a transferir sus investigaciones en lingüística teórica, o en cualquier otra disciplina lingüística, al entorno de un modelo computacional, considerando igualmente que los métodos probabilísticos pueden ayudar incluso a mejorar las propias investigaciones lingüísticas. Evidentemente, no todas las teorías lingüísticas facilitan su implementación computacional, pero la causa de este divorcio entre el PLN y la lingüística no debemos buscarla tanto en el hecho de que no todos los resultados conseguidos por la lingüística teórica puedan resultar útiles en el PLN, sino más bien en la falta de formalización que caracteriza a muchos estudios lingüísticos. A pesar de que no siempre los objetivos de la lingüística teórica puedan ser compatibles con los del PLN, nos oponemos diametralmente a la postura propuesta por Gazdar (1987) de que la lingüística computacional debe

[14] Véase la cita inicial en este artículo.

apoyarse en una *lingüística computacional teórica* en lugar de en la lingüística teórica. Nuestra postura, en cambio, defiende la reutilización del conocimiento lingüístico teórico, hasta donde sea posible, al igual que los recursos léxicos (p.ej. diccionarios o córpora) han sido reutilizados en numerosas ocasiones para el PLN. Sin duda alguna, esta estrategia requiere la implicación activa de los propios lingüistas en la adaptación de sus investigaciones al nuevo escenario de la ingeniería lingüística, donde su papel de implicación puede ser pleno o periférico, tal y como describimos en el siguiente apartado.

4. El papel del lingüista en el procesamiento del lenguaje natural

Los investigadores en PLN pueden pertenecer a comunidades científicas con *culturas* muy diferentes: la lingüística y la informática. La integración de informáticos y lingüistas en un equipo del PLN implica un cierto "cambio de perspectiva" (Llisterri, 2003). No se trata de que el lingüista se haga informático o el informático se vuelva un lingüista, sino más bien que sean capaces de compartir sus conocimientos expertos en un entorno multidisciplinar. Por tanto, el lingüista tiene que saber presentar sus problemas y soluciones de modo que el informático las entienda y pueda así darles un tratamiento adecuado, lo cual requerirá dotar al lingüista de la información esencial sobre los sistemas informáticos y proveer al informático de conocimiento básico sobre lingüística descriptiva. Este escenario implicaría, por tanto, la integración plena de lingüistas e informáticos en un proyecto del PLN. No obstante, estas dos comunidades de investigadores no sólo difieren en su conocimiento especializado, sino también en la forma en que perciben a la otra comunidad, lo cual suele malograr finalmente la colaboración investigadora. En particular, como apunta Sparck Jones (1996), mientras muchos lingüistas no confían en la tecnología, los informáticos suelen interesarse muy poco por el trabajo de los lingüistas. La investigación en un proyecto del PLN requiere adquirir conocimientos de ambas culturas, pero si el cambio de perspectiva no se produce, p.ej. porque los lingüistas carezcan de suficiente formación técnica, entonces suelen ser los lingüistas los que terminen integrándose de manera periférica en el proyecto.

4.1. La integración periférica del lingüista

Calificamos de integración *periférica* en el PLN cuando tiene lugar una realidad investigadora en la que el lingüista no se implica ni en el diseño ni en el desarrollo del proyecto, con la única posibilidad de acabar siendo "proveedores de datos o revisores de la información obtenida por procedimientos automáticos" (Llisterri, 2003). Generalmente, por falta de formación técnica, el trabajo del lingüista se limita a la construcción de los recursos lingüísticos necesarios para la ingeniería lingüística moderna: gramáticas, lexicones, córpora y ontologías. En estos casos, los lingüistas suelen tener áreas de trabajo bien delimitadas dentro del proyecto del PLN:

(i) La búsqueda de regularidades gramaticales que puedan ser expresadas por medio de reglas conlleva la descripción formalizada de la estructura morfológica, sintáctica y/o semántica de una lengua, cuyo conocimiento pueda ser utilizado tanto en el análisis como en la generación de textos.

(ii) La construcción de lexicones computacionales –monolingües, bilingües o multilingües– implica la formalización y la estructuración del conocimiento morfológico, sintáctico y/o semántico de las unidades léxicas.

(iii) La anotación de córpora informatizados desempeña un papel fundamental en las investigaciones del PLN (p.ej. la resolución de ambigüedades, la extracción terminológica, etc), ya que tanto los sistemas probabilísticos como el aprendizaje automático de conocimiento lingüístico requieren grandes cantidades de datos[15].

(iv) En la construcción de ontologías, los lingüistas pueden explotar sus destrezas de organización terminológica y representación semántica dentro de un modelo de conocimiento más abstracto.

La construcción manual de estos recursos lingüísticos, los cuales requieren a veces tanto profundidad como cobertura, constituye en ocasiones un cuello de botella para los sistemas del PLN. Por esta razón, siempre ha resultado una idea muy atractiva, al menos para los informáticos, poder automatizar las tareas (i)-(iv) y así prescindir del laborioso trabajo del lingüista con el fin de acelerar la evolución de la ingeniería lingüística. No obstante, los ingenieros lingüistas no pueden prescindir de los

[15] Un tipo de corpus de especial importancia para el análisis sintáctico son los *treebanks*, en los cuales cada oración está anotada sintácticamente con un análisis arbóreo.

lingüistas, aunque sea sólo para validar los resultados obtenidos por un procedimiento automático de adquisición de conocimiento, en el caso de que se quieran desarrollar sistemas provistos de fuentes de conocimiento de mayor calidad con el propósito de ir más allá de la resolución de pequeños problemas *ad hoc*.

4.2. La integración plena del lingüista

La participación periférica de los lingüistas en un proyecto del PLN implica que éstos posean no sólo capacidad de abstracción, razonamiento lógico y organización y estructuración de los datos, sino además capacidad para el trabajo en equipo (Martí Antonín y Llisterri, 2001). En cambio, una integración *plena* en un equipo del PLN exige, por parte del lingüista, una formación técnica más intensiva que durará varios años[16]. En otras palabras, aquel lingüista que desee desempeñar un papel central en el diseño del modelo de un sistema del PLN, convendría que adquiriera conocimientos informáticos básicos (p.ej. lenguajes de programación, gestión de bases de datos e ingeniería del *software*), además de una formación más específica en métodos estadísticos utilizados en el PLN. Aunque este tipo de participación implique que sea conveniente que los lingüistas adquieran un cierto conocimiento especializado sobre ingeniería del *software*, no estamos sugiriendo en ningún momento que se dediquen a programar el propio sistema, ya que podrían correr el riesgo de *reinventar la rueda*: una cuestión es conocer los fundamentos básicos de la programación informática (p.ej. variables, operadores, estructuras de control, etc.) y otra cuestión muy diferente es ser un experto en algoritmia. Por tanto, los lingüistas deben limitarse a la producción de investigación avanzada y no a la producción de aplicaciones informáticas. Como ilustra Ferrari (2004), aunque los puentes y edificios permanecen de pie gracias a algunos principios físicos, no son realmente los físicos quienes intervienen finalmente en los proyectos de ingeniería.

Igualmente, los informáticos que participen en proyectos del PLN deben poseer una sólida formación en lingüística descriptiva (Moore, 2009), e incluso tener la posibilidad de estar directamente implicados en proyectos de investigación lingüística, ya que su

[16] En España, estos conocimientos especializados sólo se adquieren en cursos de postgrado.

formación computacional puede aportar una visión diferente (Wintner, 2009). En esta ocasión, sugerimos, por tanto, que los informáticos sean capaces de comprender los principales modelos en lingüística teórica, ya que existen actualmente algunas líneas de investigación dentro de la lingüística teórica de las cuales la ingeniería lingüística podría beneficiarse. Por ejemplo, Bender (2009) sugiere que las investigaciones sobre tipología lingüística pueden representar una fuente rica de conocimiento para ser incorporado en sistemas del PLN. Desgraciadamente, como confiesa Moore (2009), muchos informáticos se especializan en lingüística computacional sin poseer suficientes conocimientos sobre la estructuración interna de las lenguas.

En conclusión, la colaboración entre lingüistas e informáticos es necesaria si deseamos construir un sistema robusto del PLN, donde los lingüistas pueden limitarse a proporcionar el conocimiento lingüístico necesario (i.e. integración periférica), o bien involucrarse en el lado más *creativo* del proyecto (i.e. integración plena). Aunque atraídos por las diversas aplicaciones de las tecnologías del lenguaje, muchos lingüistas se muestran con frecuencia reacios a adquirir conocimiento técnico, alegando incluso que no se sienten preparados para ello. Ésta es una de las razones por las cuales los propios lingüistas rehúyen a integrarse de forma plena en un proyecto del PLN. En el siguiente apartado presentamos un proyecto de ingeniería del conocimiento que ha facilitado dicha integración, donde las teorías funcionales del lenguaje pueden ser incorporadas a los sistemas del PLN con el fin de desarrollar aplicaciones más inteligentes.

5. FunGramKB

FunGramKB[17] es una base de conocimiento léxico-conceptual multipropósito diseñada principalmente para su uso en sistemas del PLN, y más concretamente, para aplicaciones que requieran la comprensión del lenguaje. Por una parte, esta base de conocimiento es *multipropósito* en el sentido de que es tanto multifuncional como multilingüe. En otras palabras, FunGramKB ha sido diseñada con el fin de ser potencialmente reutilizada en diversas tareas del PLN (p.ej. recuperación y

[17] www.fungramkb.com

extracción de información, traducción automática, sistemas basados en el diálogo, etc) y con diversas lenguas[18]. Por otra parte, FunGramKB comprende tres niveles principales de conocimiento, cada uno de los cuales está constituido por diversos módulos independientes aunque claramente interrelacionados[19]:

Nivel léxico:

(i) El Lexicón almacena la información morfosintáctica de las unidades léxicas.

(ii) El Morficón asiste al analizador y al generador en el tratamiento de los casos de morfología flexiva.

Nivel gramatical:

(iii) El Gramaticón almacena los esquemas construccionales y su representación semántica.

Nivel conceptual:

(iv) La Ontología se presenta como una jerarquía IS-A de unidades conceptuales, las cuales contienen el conocimiento del sentido común en forma de postulados de significado.

(v) El Cognicón almacena el conocimiento procedimental por medio de guiones, los cuales permiten describir, por ejemplo, cómo se hace una tortilla o cómo se realiza una compra *online*.

(vi) El Onomasticón almacena el conocimiento enciclopédico sobre instancias de entidades y eventos, tales como Cervantes o el 11-M.

Como explicamos en el resto de apartados de esta sección, esta base de conocimiento propicia la construcción de sistemas del PLN fundamentados tanto en la ciencia cognitiva como en la lingüística teórica.

5.1. FunGramKB y la ciencia cognitiva

El modelo de *esquema* originado en la psicología cognitiva, e implementado posteriormente en inteligencia artificial, es fundamental para la representación del conocimiento conceptual

---

[18] Actualmente, FunGramKB ha sido modelada para poder trabajar con siete lenguas: alemán, búlgaro, catalán, español, francés, inglés e italiano.

[19] Para una información más detallada sobre el conocimiento almacenado en FunGramKB, léanse Periñán Pascual y Arcas Túnez (2004, 2007, 2008, 2010a, 2010b), Periñán Pascual y Mairal Usón (2010), y Jiménez Briones y Luzondo Oyón (2011) acerca del nivel conceptual, y Mairal Usón y Periñán Pascual (2009) con respecto al nivel léxico. El nivel gramatical está actualmente en desarrollo.

en FunGramKB. Según este enfoque, un esquema es una representación mental de una entidad o un evento, la cual consiste generalmente en un conjunto de expectativas que se van desarrollando a medida que los recuerdos se nutren de experiencias similares. Típicamente los esquemas contienen conocimiento generalizado a partir de las experiencias pasadas, facilitando así la inferencia de información a partir de nuestra percepción del mundo. Las experiencias futuras se interpretan de acuerdo con los patrones construidos a partir de las experiencias pasadas, lo cual alivia la sobrecarga cognitiva.

Los esquemas conceptuales de FunGramKB desempeñan un papel primordial en la inferencia de conocimiento durante el proceso de comprensión del lenguaje. En nuestra base de conocimiento, los esquemas conceptuales se clasifican atendiendo a dos parámetros: (i) prototipicidad y (ii) temporalidad. De un lado, los esquemas conceptuales almacenan conocimiento prototípico (i.e. protoestructuras), o bien pueden servir para describir una instancia de una entidad o un evento (i.e. bioestructuras). Por ejemplo, la descripción del significado de la unidad léxica "película" implica describir la protoestructura del concepto al que va asignada; en cambio, si deseamos proporcionar información sobre la película "Todo sobre mi madre" necesitamos hacerlo a través de una bioestructura. Igualmente, podemos presentar el conocimiento atemporalmente (i.e. microestructuras), o inserto en un paradigma temporal (i.e. macroestructuras). Por ejemplo, la descripción de la biografía de Pedro Almodóvar requiere una macroestructura, mientras que una microestructura es suficiente para describir la profesión de director cinematográfico. Cuando combinamos estos dos parámetros, obtenemos nuestro inventario de esquemas conceptuales: proto-microestructuras (o postulados de significado), proto-macroestructuras (o guiones), bio-microestructuras (o retratos) y bio-macroestructuras (o historias).

Con respecto a la dimensión de la prototipicidad, y gracias a estos esquemas conceptuales, FunGramKB permite describir diversos tipos de conocimiento, los cuales ilustramos a continuación:

- conocimiento del sentido común, el cual almacena las creencias precientíficas sobre las cuales se construyen las actividades cognitivas cotidianas, p.ej.:

(1) When you forbid someone to do something, you tell them that they are not allowed to do it.

+(e1: +SAY_00 (x1: +HUMAN_00)Theme (x4: (e2: obl n +DO_00 (x3: +HUMAN_00)Theme (x2)Referent))Referent (x3)Goal)

- conocimiento especializado, el cual contiene las creencias aceptadas por una comunidad experta sobre dominios académicos, científicos o técnicos, p.ej.:

(2) A circumfix is an affix made up of two parts which surround a stem.

+(e1: +BE_00 (x1: $CIRCUMFIX_00)Theme (x2: +AFFIX_00)Referent)

+((e2: +COMPRISE_00 (x1)Theme (x3: 2 +PART_00)Referent)(e3: +SURROUND_00 (x3)Theme (x4: +STEM_00)Location))

- conocimiento cultural, el cual consiste en información factual sobre nuestro modelo del mundo en el pasado, presente o futuro, p.ej.:

(3) Paris is the capital of France.

*(e1: +BE_00 (x1: %PARIS_00)Theme (x2: $CAPITAL_00)Referent)

+(e2: +BE_02 (x1)Theme (x3: %FRANCE_00)Location)

- conocimiento personal, cuyas creencias sólo son completamente verdaderas desde el punto de vista de la persona que las tiene, p.ej.:

(4) My wife Julia can prepare avocado and shrimp cocktails.

*(e1: +BE_00 (x1: %JULIA_00)Theme (x2: +WIFE_00)Referent)

*((e2: pos +CREATE_00 (x1)Theme (x3: $COCKTAIL_00)Referent)(e3: +COMPRISE_00 (x3)Theme (x4: $AVOCADO_00 & $SHRIMP_00) Referent))

Observamos que en FunGramKB los diferentes tipos de conocimiento están formalizados a través del mismo lenguaje de interfaz, i.e. COREL (COnceptual Representation Language)[20].

Por otra parte, con respecto a la dimensión de la temporalidad, FunGramKB organiza su conocimiento de acuerdo con la distinción de Barsalou (1985, 1991) entre categorías taxonómicas y categorías derivadas de objetivos. Mientras las categorías taxonómicas van provistas de representaciones independientes del contexto organizadas jerárquicamente a través de un modelo ontológico, las categorías derivadas de objetivos se conceptualizan a través de representaciones que tienen en cuenta la situación de fondo. En este sentido, FunGramKB implementa el enfoque de la *conceptualización situada* (Barsalou, 2002) a través de las

[20] Léase Periñán Pascual y Mairal Usón (2010) para una descripción detallada del lenguaje de notación de COREL.

categorías derivadas de objetivos, las cuales adoptan la forma de guiones e historias. A continuación presentamos cuatro macroestructuras que también sirven para ilustrar los tipos de conocimiento descritos anteriormente:

(5) After washing your face, spread the foam over your beard and shave it off.

*(e1: +WASH_00 (x1)Theme (x2: +FACE_00)Referent)

*(e2: +COVER_00 (x1)Agent (x3: $FOAM_00)Theme (x4)Origin (x5: +BEARD_00)Goal)

*(e3: $SHAVE_00 (x1)Theme (x5)Referent)

(6) Once you have selected the hard disk you want to defragment, click the "Analyze" button and a progress bar will appear.

*(e1: +CHOOSE_00 (x1)Theme (x2: $HARD_DISK_00)Referent (f1: (e2: $DEFRAGMENT_00 (x1)Theme (x2)Referent))Purpose)

*(e3: +PUSH_00 (x1)Agent (x3: %ANALYZE_BUTTON_00)Theme (x4)Location (x5)Origin (x6)Goal)

*(e4: $APPEAR_00 (x7: $PROGRESS_BAR_00)Theme)

(7) The German army invaded Poland on September 1, 1939.

+(e1: past $INVADE_00 (x1)Agent (x2: %WEHRMACHT_00)Theme (x3)Location (x4: %GERMANY_00)Origin (x5: %POLAND_00)Goal (f1: 1 +DAY & $SEPTEMBER_00 & 1939 +YEAR_00)Time)

(8) John and Linda worked in Spain in 1994.

+(e1: past +WORK_01 (x1: %JOHN_00 & %LINDA_00)Theme (f1: %SPAIN_00)Location (f2: 1994 +YEAR_00)Time)

En definitiva, la conceptualización en FunGramKB está en consonancia con la teoría de Lakoff (1987) sobre los Modelos Cognitivos Idealizados (MCI) proposicionales, i.e. aquellas configuraciones conceptuales que no están basadas en los *mecanismos imaginativos* tales como la metáfora y la metonimia. De hecho, las diversas estructuras en las que los MCI pueden tener lugar están presentes en FunGramKB, donde nuestra base de conocimiento marca la diferencia en su capacidad de integrar plenamente los diversos tipos de esquemas.

5.2. FunGramKB y la lingüística teórica

Uno de los objetivos de FunGramKB consiste en que el PLN vuelva a beneficiarse de las investigaciones en lingüística

teórica[21], más concretamente, de la teoría funcional de la Gramática del Papel y la Referencia (RRG) (Van Valin y LaPolla, 1997; Van Valin, 2005) y el Modelo Léxico Construccional (MLC) (Ruiz de Mendoza y Mairal Usón, 2008; Mairal Usón y Ruiz de Mendoza, 2009).

La RRG, una de las teorías funcionales más relevantes del panorama lingüístico actual, es un modelo gramatical de carácter monoestrático, donde los componentes sintáctico y semántico se vinculan directamente en virtud de un algoritmo de enlace bidireccional, adoptando además un enfoque comunicativo-cognitivo del lenguaje. La RRG ha repercutido notablemente en el diseño del modelo léxico de FunGramKB, ya que, como apuntamos a continuación, determinadas características de esta teoría gramatical resultan bastante atractivas para el PLN:

(i) Se trata de un modelo funcional del lenguaje, donde las estructuras morfosintácticas y las reglas gramaticales deben ser explicadas en relación con sus funciones semánticas y comunicativas.

(ii) El algoritmo de enlace es bidireccional, lo cual implica que sirve tanto para el análisis como para la producción de expresiones lingüísticas.

(iii) Su adecuación tipológica permite introducir distinciones universales como parte del aparato lingüístico.

(iv) El principal componente para la descripción lingüística es el lexicón, donde los predicados se almacenan como descomposiciones semánticas en forma de *estructuras lógicas*.

En realidad, los rasgos (i-iii) son esenciales para cualquier modelo computacional del lenguaje. En primer lugar, un enfoque funcional del lenguaje nos permite capturar las generalizaciones sintáctico-semánticas que son fundamentales para explicar la motivación semántica de los fenómenos gramaticales. En segundo lugar, la adecuación psicológica es particularmente relevante para aquellos modelos cuya finalidad es la comprensión del lenguaje. En tercer lugar, la adecuación tipológica debe ser un requisito

[21] En realidad, como apunta Grishman (1986), la lingüística computacional se presenta además como un terreno idóneo para la verificación de las teorías lingüísticas. Así, por ejemplo, Periñán Pascual y Mairal Usón (2012) han podido extender el modelo teórico de la RRG con el fin de proporcionar, dentro del marco ontológico, una solución más elegante a la representación de la semántica léxica en un entorno multilingüe.

en los modelos multilingües. Por otra parte, la característica (iv) refleja la idea comúnmente aceptada de que la entrada léxica debe contener una rica descomposición semántica que vaya más allá de aquellos aspectos gramaticalmente relevantes; no obstante, mientras la RRG es un ejemplo de teoría lexicista, FunGramKB adopta un enfoque conceptualista.

Como consecuencia de este giro conceptualista, la estructura lógica de la RRG ha sido mejorada, dando como resultado un nuevo formalismo denominado *estructura lógica conceptual* (Periñán Pascual y Mairal Usón, 2009). Por ejemplo, a partir de la oración (9), la estructura lógica (10) es reemplazada por la estructura lógica conceptual (11), donde la diferencia radica en la sustitución de los predicados por unidades ontológicas acompañadas de los papeles temáticos que permiten el vínculo con los postulados de significado.

(9) John ate the bread.

(10) $<_{IF}$ DEC $<_{TNS}$ PAST <**do'** (John, [**eat'** (John, bread)]) & INGR **consumed'** (bread)>>>

(11) $<_{IF}$ DEC $<_{TNS}$ PAST <do (%JOHN_00-Agent [+EAT_00 (%JOHN_00-Agent, +BREAD_00-Theme)]) & INGR +EAT_00 (+BREAD_00-Theme)>>>

La principal ventaja de la estructura lógica conceptual reside en el hecho de que se trata de una representación independiente de la lengua, ya que no está configurada por unidades léxicas sino por conceptos ontológicos. Gracias a este nuevo enfoque interlingüístico, somos capaces de minimizar la redundancia de información al mismo tiempo que maximizamos la informatividad.

En realidad, con el propósito de facilitar el procesamiento computacional, una estructura lógica conceptual de una realización activa como (11) se simplificaría a (12), cuyo formalismo es más tratable por la máquina[22]:

(12) $<_{IF}$ DEC $<_{TNS}$ PAST $<_{AKT}$ ACA [+EAT_00 (%JOHN_00-Agent, +BREAD_00-Theme)]>>>

Desde nuestro enfoque, la utilidad de una estructura lógica convencional implica conocer el Aktionsart subyacente, y no su esqueleto pseudosemántico, el cual se ve ahora enriquecido por los diversos tipos de esquemas conceptuales en COREL.

[22] En realidad, ACA representaría todo el armazón formal de una realización activa (en inglés, *active accomplishment*).

Por ejemplo, la estructura lógica conceptual (12) corresponde al esquema conceptual (13), el cual puede expandirse al esquema (14) gracias al postulado de significado del evento +EAT_00:

(13) +(e1: past +EAT_00 (x1: %JOHN_00)Agent (x2: +BREAD_00)Theme (x3)Location (x4)Origin (x5)Goal)

(14) *(e1: past +INGEST_00 (x1: %JOHN_00)Agent (x2: +BREAD_00) Theme (x3)Location (x4)Origin (x5: +STOMACH_00)Goal (f1: +MOUTH_00)Means (f2)Instrument (f3: (e2: +CHEW_00 (x1)Theme (x2)Referent))Manner)

En otras palabras, a partir del aducto (9) podemos inferir que "John ingirió el pan, masticándolo con la boca y con ayuda de algún instrumento, terminando el pan finalmente en el estómago".

Otra de las teorías lingüísticas de las cuales FunGramKB se ha beneficiado es el MLC, el cual proporciona a la Gramática del Papel y la Referencia un tratamiento más adecuado sobre la construcción del significado, yendo más allá de la gramática nuclear e incorporando dimensiones del significado de larga tradición en la pragmática y el análisis del discurso. Más concretamente, el MLC reconoce cuatro niveles construccionales (i.e. argumental, implicativo, ilocutivo y discursivo) que dan forma a los cuatro Constructicones del Gramaticón. Por ejemplo, la oración (15) necesita la información sobre la construcción resultativa almacenada en el Constructicón de nivel 1, o argumental, con el fin de obtener la estructura lógica conceptual (16) y el esquema conceptual en COREL (17).

(15) The water froze solid.

(16) $<_{IF}$ DEC $<_{TNS}$ PAST $<_{L1C}$ RESU $<_{AKT}$ ACC [+FREEZE_00 (+WATER_00-Referent) (+SOLID_00-Attribute)]>>>

(17) +(e1: past +FREEZE_00 (x1)Theme (x2: +WATER_00)Referent (f1: (e2: +BECOME_00 (x2)Theme (x3: +SOLID_00)Attribute))Result)

En definitiva, las dos interlinguas de FunGramKB se complementan, pero desempeñando papeles diferentes en los marcos de la lingüística y de la inteligencia artificial: las estructuras lógicas conceptuales se construyen en los niveles léxico y gramatical durante la fase de procesamiento del aducto, mientras que los esquemas conceptuales en COREL se construyen en el nivel conceptual durante el proceso de comprensión automática del lenguaje. Es decir, la utilidad de la estructura lógica conceptual se limita al tratamiento de los fenómenos gramaticales

que tienen lugar durante los procesos de análisis o generación lingüísticos. En el caso de que el sistema deba aplicar alguna tarea de razonamiento sobre el texto de entrada, es preciso que la estructura lógica conceptual se *traduzca* automáticamente a un esquema conceptual en COREL. De esta forma, la estructura lógica conceptual puede ser enriquecida con el conocimiento de los diversos tipos de esquemas conceptuales almacenados en FunGramKB. Por tanto, FunGramKB integra perfectamente un modelo de lingüística teórica con un enfoque simbólico propio de la Inteligencia Artificial.

6. Conclusiones

A lo largo de su historia, el PLN se ha ido construyendo a partir de la combinación de diversos componentes: modelos lingüísticos, representación del conocimiento y razonamiento lógico, métodos estadísticos y recursos lingüísticos. No obstante, como indica Wilks (2005), los vínculos entre el PLN y la lingüística no han sido ni tan numerosos ni tan productivos como podríamos imaginar. De hecho, sólo el trabajo de unos pocos lingüistas, p.ej. Chomsky, Halliday y Fillmore, ha influido notablemente en el desarrollo de este campo de investigación, el cual favorece actualmente la adopción de enfoques estadísticos en detrimento de cualquier teoría lingüística. En este panorama, FunGramKB irrumpe como una base de conocimiento léxico, construccional y conceptual desarrollada a partir de un modelo plausible tanto lingüística como cognitivamente, cuyo propósito es contribuir a la implementación de sistemas computacionales que simulen el razonamiento humano. Uno de los axiomas metodológicos más importantes en esta base de conocimiento es la nítida separación entre los niveles lingüístico y cognitivo, una distinción que motiva la presencia de dos metalenguajes que sirven como pilares básicos de un puente que conecta la lingüística y la inteligencia artificial: la estructura lógica conceptual y el esquema conceptual en COREL. Gracias a este nuevo respaldo a la lingüística teórica, los lingüistas pueden seguir desempeñando el papel incuestionable que les corresponde en un proyecto del PLN.

7. Agradecimientos

Este trabajo forma parte de dos proyectos de investigación financiados por el Ministerio de Ciencia y Tecnología de España, códigos FFI2011-29798-C02-01 y FFI2010-15983. También quiero expresar mi agradecimiento a Francisco Cortés Rodríguez, Carlos González Vergara y Ricardo Mairal Usón por sus comentarios sobre el primer borrador de este artículo.

8. Bibliografía citada

Barsalou, Lawrence W., 1985: "Ideals, central tendency, and frequency of instantiation as determinants of graded structure in categories", *Journal of Experimental Psychology: Learning, Memory, and Cognition* 11, 629-654.

—, 1991: "Deriving categories to achieve goals" en Gordon H. Bower (ed.): *The psychology of learning and motivation: advances in research and theory,* vol. 27, San Diego: Academic Press, 1-64.

—, 2002: "Being there conceptually: simulating categories in preparation for situated action" en Nancy L. Stein, Patricia J. Bauer y Mitchell Rabinowitz (eds.): *Representation, memory and development: essays in honor of Jean Mandler*, Mahwah: Lawrence Erlbaum, 1-15.

Bateman, John A., 1991: "The theoretical status of ontologies in natural language processing" en Susanne Preuss y Birte Schmitz (eds.): *Text representation and domain modelling: ideas from linguistics and AI*. Informe técnico, Technische Universitaet Berlin, 50-99.

—, 1997: *KPML Development Environment: multilingual linguistic resource development and sentence generation*. Informe técnico, German National Center for Information Technology, Institute for Integrated Publication and Information Systems, Darmstadt.

Bender, Emily M., 2009: "Linguistically naïve != language independent: why NLP needs linguistic typology" en *Proceedings of the European Chapter of the ACL 2009 Workshop on the Interaction between Linguistics and Computational Linguistics*, Association for Computational Linguistics, 26-32.

Brown, Peter F. y otros, 1990: "A statistical approach to machine translation", *Computational Linguistics* 16 (2), 79-85.

Butler, Christopher S., 2003a: *Structure and function: A guide to three major structural-functional theories. Part 1: Approaches to the simplex clause,* Ámsterdam-Filadelfia: John Benjamins.

—, 2003b: *Structure and Function: A guide to three major structural-functional theories. Part 2: From clause to discourse and beyond*, Ámsterdam-Filadelfia: John Benjamins.

Chomsky, Noam, 1957: *Syntactic structures*, La Haya: Mouton.

—, 1965: *Aspects of the theory of syntax*, Cambridge (Mass.): MIT Press.
—, 1970: "Remarks on nominalization" en Roderick A. Jacobs y Peter S. Rosenbaum (eds.): *Readings in English Transformational Grammar,* Cambridge: Waltham (Mass.): Ginn and Co, 184-221.
—, 1981: *Lectures on government and binding*, Dordrecht: Foris.
Christiansen, Morten H. y Nick Chater, 1999: "Connectionist natural language processing: the state of the art", *Cognitive Science* 23, 417-437.
Colby, Kenneth, 1973: "Simulation of belief systems" en Roger C. Schank y Kenneth Colby (eds.): *Computer models of thought and language*, San Francisco: Freeman, 251-286.
Cullingford, Richard Edward, 1978: *Script application: computer understanding of newspaper stories*. Informe técnico, Yale University.
Dik, Simon C., 1989: *The theory of Functional Grammar*, Dordrecht: Foris.
Farwell, David y Yorick Wilks, 1991: "ULTRA: a multilingual machine translator" en *Proceedings of the Machine Translation Summit III*, Washington, DC, 19-24.
Fawcett, Robin P. y Gordon H. Tucker, 1990: "Demonstration of GENESYS: a very large, semantically based systemic functional grammar" en *Proceedings of the 13th International Conference on Computational Linguistics,* Helsinki, 47-49.
Ferrari, Giacomo, 2004: "State of the art in computational linguistics" en Piet van Sterkenburg (ed.): *Linguistics today: facing a greater challenge*, Ámsterdam-Filadelfia: John Benjamins, 163-186.
Fillmore, Charles J., 1968: "The case for case" en Emmon W. Bach y Robert T. Harms (ed.): *Universals in linguistic theory*, Nueva York: Holt, Rinehart and Winston, 1-88.
Fillmore, Charles J. y Beryl T. Atkins, 1992: "Toward a frame-based lexicon: the semantics of RISK and its neighbors" en Adrienne Lehrer y Eva Feder Kittay (ed.): *Frames, fields, and contrasts*, Hillsdale: Lawrence, 75-102.
—, 1994: "Starting where the dictionaries stop: the challenge of corpus lexicography" en B.T.S. Atkins y Antonio Zampolli (eds.): *Computational approaches to the lexicon*, Oxford: Oxford University Press, 349-393.
Gazdar, Gerald, 1987: "Linguistic applications of default inheritance mechanisms" en Peter Whitelock y otros (eds.): *Linguistic theory and computer applications*, Londres: Academic Press, 37-67.
Gazdar, Gerald y otros, 1985: *Generalised Phrase Structure Grammar*, Oxford: Basil Blackwell.
Gonzálvez-García, Francisco y Christopher S. Butler, 2006: "Mapping functional-cognitive space", *Annual Review of Cognitive Linguistics* 4, 39-96.
Grishman, Ralph, 1986: *Computational linguistics: an introduction*, Cambridge, Cambridge University Press.
Halliday, Michael, 1961: "Categories of the theory of grammar", *Word* 17, 241-92.

—, 1967: "Notes on transitivity and theme in English", *Journal of Linguistics* 3, 199-244.

—, 1973: *Explorations in the functions of language*, Londres: Edward Arnold.

Halvorsen, Per-Kristian, 1988: "Computer applications of linguistic theory" en Frederick Newmeyer (ed.): *Linguistics: the Cambridge survey II. Linguistic theory: extensions and implications*, Cambridge: Cambridge University Press, 198-219.

Hanks, Patrick, 2003: "Lexicography" en Ruslan Mitkov (ed.): *The Oxford handbook of computational linguistics*, Oxford: Oxford University Press, 48-69.

Hengeveld, Kees y J. Lachlan Mackenzie, 2008: *Functional Discourse Grammar: a typologically-based theory of language structure*, Oxford: Oxford University Press.

Jackendoff, Ray, 1972: *Semantic interpretation in Generative Grammar*, Cambridge (Mass.): MIT Press.

Jiménez Briones, Rocío y Alba Luzondo Oyón, 2011: "Building ontological meaning in a lexico-conceptual knowledge base", *Onomázein* 23, 11-40.

Jurafsky, Daniel y James H. Martin, 2009: *Speech and language processing: an introduction to natural language processing, speech recognition, and computational linguistics*, New Jersey: Prentice Hall.

Kaplan, Ronald M. y Joan Bresnan, 1982: "Lexical-Functional Grammar: a formal system for grammatical representation" en Joan Bresnan (ed.): *The mental representation of grammatical relations*, Cambridge (Mass.): MIT Press, 173-280.

Karttunen, Lauri, 1989: "Radical lexicalism" en Mark R. Baltin y Anthony S. Kroch (eds.): *Alternative conceptions of phrase structure*, Chicago: University of Chicago Press, 43-65.

Kasper, Robert, 1987: "Systemic grammar and functional unification grammar" en James D. Benson y Williams S. Greaves (eds.): *Systemic perspectives on discourse*, vol. 1, Norwood: Ablex, 176-199.

Kay, Martin, 1985: "Parsing in Functional Unification Grammar" en David R. Dowty, Lauri Karttunen y Arnold M. Zwicky (eds.): *Natural language parsing,* Cambridge: Cambridge University Press, 251-278.

Lakoff, George, 1987: *Women, fire, and dangerous things*, Chicago: University of Chicago Press.

Lenci, Alessandro, 2001: "Building an ontology for the lexicon: semantic types and word meaning" en Per Anker Jensen y Peter Rossen Skadhauge (eds.): *Ontology-Based Interpretation of Noun Phrases*, Kolding: University of Southern Denmark, 103-120.

Levin, Beth y Malka Rappaport Hovav, 2005: *Argument realization*, Cambridge: Cambridge University Press.

Liddy, Elizabeth D., 2001: "Natural language processing" en *Encyclopedia of library and information science*, segunda edición, Nueva York: Marcel Decker.

Llisterri Boix, Joaquim, 2003: "Lingüística y tecnologías del lenguaje", *Lynx. Panorámica de Estudios Lingüísticos* 2, 9-71.

Mairal Usón, Ricardo y Carlos Periñán Pascual, 2009: "The anatomy of the lexicon within the framework of an NLP knowledge base", *Revista Española de Lingüística Aplicada* 22, 217-244.

Mairal Usón, Ricardo y Francisco Ruiz de Mendoza, 2009: "Levels of description and explanation in meaning construction" en Christopher Butler y Javier Martín Arista (eds.): *Deconstructing constructions*, Ámsterdam-Filadelfia: John Benjamins, 153-198.

Mann, William C., 1983: *An overview of the PENMAN text generation system*. Informe técnico ISI/RR-83-114, University of Southern California.

Martí Antonín, María Antonia (ed.), 2003: *Tecnologías del lenguaje*, Barcelona: Universitat Oberta de Catalunya.

Martí Antonín, Maria Antònia y Joaquim Llisterri, 2001: "La ingeniería lingüística en la sociedad de la información", *Digithum, Revista Digital d'Humanitats* 3 [http://www.uoc.edu/humfil/articles/esp/llisterri-marti/llisterri-marti_imp.html, fecha de consulta: 12 de diciembre de 2011].

Mitamura, Teruko, Eric Nyberg y Jaime Carbonell, 1991: "An efficient interlingua translation system for multilingual document production" en *Proceedings of the Machine Translation Summit III*, Washington, DC, 55-61.

Moreno Ortiz, Antonio, 2000: *Diseño e implementación de un lexicón computacional para lexicografía y traducción automática*. Tesis doctoral [http://elies.rediris.es/elies9/, fecha de consulta: 12 de diciembre de 2011]

Moore, Robert C., 2009: "What do computational linguists need to know about linguistics?" en *Proceedings of the European Chapter of the ACL 2009 Workshop on the Interaction between Linguistics and Computational Linguistics*, Association for Computational Linguistics, 41-42.

Moure, Teresa y Joaquim Llisterri, 1996: "Lenguaje y nuevas tecnologías: el campo de la lingüística computacional" en Milagros Fernández Pérez (ed.): *Avances en lingüística aplicada*, Santiago de Compostela: Universidad de Santiago de Compostela, 147-227.

Nirenburg, Sergei y Lori Levin, 1992: "Syntax-driven and ontology-driven lexical semantics" en James Pustejovsky y Sabine Bergler (eds.): *Lexical semantics and knowledge representation*, Berlín-Heidelberg: Springer, 5-20.

Nirenburg, Sergei y otros, 1992: *Machine translation: a knowledge-based approach*, San Mateo: Morgan Kaufmann.

—, 1996: "Lexicons in the MikroKosmos project" en *Proceedings of the AISB'96 Workshop on Multilinguality in the Lexicon*, Brighton, 26-33.

Not, Elena y Oliviero Stock, 1994: "Automatic generation of instructions for citizens in a multilingual community" en *Proceedings of the European Language Engineering Convention*, París.

O'Donnell, Michael, 1994: *Sentence analysis and generation: a systemic perspective*. Tesis doctoral, University of Sydney.

Onyshkevych, Boyan A. y Sergei Nirenburg, 1992: "Lexicon, ontology, and text meaning" en James Pustejovsky y Sabine Bergler (eds.): *Lexical semantics and knowledge representation*, Berlín-Heidelberg: Springer, 289-303.

—, 1995: "A lexicon for knowledge-based MT", *Machine Translation* 10 (1-2), 5-57.

Palmer, Martha y Zhibiao Wu, 1995: "Verb semantics for English-Chinese translation", *Machine Translation* 10 (1-2), 59-92.

Paris, Cécile L. y Keith Vander Linden, 1996: "DRAFTER: an interactive support tool for writing multilingual instructions", *IEEE Computer* 29 (7), 49-56.

Periñán Pascual, Carlos y Francisco Arcas Túnez, 2004: "Meaning postulates in a lexico-conceptual knowledge base" en *Proceedings of the 15th International Workshop on Databases and Expert Systems Applications*, Los Alamitos: IEEE, 38-42.

—, 2007: "Cognitive modules of an NLP knowledge base for language understanding", *Procesamiento del Lenguaje Natural* 39, 197-204.

—, 2008: "A cognitive approach to qualities for NLP", *Procesamiento del Lenguaje Natural* 41, 137-144.

—, 2010a: "Ontological commitments in FunGramKB", *Procesamiento del Lenguaje Natural* 44, 27-34.

—, 2010b: "The architecture of FunGramKB" en *Proceedings of the Seventh International Conference on Language Resources and Evaluation*, Malta: ELRA, 2667-2674.

Periñán Pascual, Carlos y Ricardo Mairal Usón, 2009: "Bringing Role and Reference Grammar to natural language understanding", *Procesamiento del Lenguaje Natural* 43, 265-273.

—, 2010: "La gramática de COREL: un lenguaje de representación conceptual", *Onomázein* 21, 11-45.

—, 2012: "La dimensión computacional de la GPR: la estructura lógica conceptual y su aplicación en el procesamiento del lenguaje natural" en Ricardo Mairal Usón, Lilián Guerrero y Carlos González Vergara (eds.) *La Gramática del Papel y la Referencia: introducción, avances y aplicaciones*, Akal: Madrid, 333-348.

Pollard, Carl J. e Ivan I. Sag, 1987: *Information-based syntax and semantics*, Stanford: CSLI.

Pustejovsky, James, 1991: "The Generative Lexicon", *Computational Linguistics* 17 (4), 409-441.

—, 1995: *The Generative Lexicon*, Cambridge (Mass.): MIT Press.

Ramsay, Allan., 2004: "Artificial intelligence" en Kirsten Malmkjaer (ed.): *The linguistics encyclopedia*, Londres y Nueva York: Routledge, 34-46.

Riesbeck, Christopher K., 1975: "Conceptual analysis" en Roger C. Schank (ed.): *Conceptual information processing*, Ámsterdam: North-Holland, 83-156.

Riesbeck, Christopher K. y Roger C. Schank, 1978: "Comprehension by computer: expectation-based analysis of sentences in context" en Willem J. M. Levelt y Giovanni B. Flores d'Arcais (eds.): *Studies in the perception of language*, Nueva York: Wiley, 247-294.

Rösner, Dietmar y Manfred Stede, 1994: "Generating multilingual documents from a knowledge base: the TECHDOC project" en *Proceedings of the 15th International Conference on Computational Linguistics*, 339-346.

Ruiz de Mendoza, Francisco y Ricardo Mairal Usón, 2008: "Levels of description and constraining factors in meaning construction: an introduction to the Lexical Constructional Model", *Folia Linguistica* 42 (2), 355-400.

Schank, Roger C., 1972: "Conceptual Dependency: a theory of natural language understanding", *Cognitive Psychology* 3 (4), 532-631.

—, 1975: *Conceptual information processing*, Ámsterdam: North-Holland.

—, 1982a: *Dynamic Memory: a theory of reminding and learning in computers and people*, Londres: Cambridge University Press.

—, 1982b: *Reading and understanding*, Hillsdale: Lawrence Erlbaum.

—, 1986: *Explanation patterns: understanding mechanically and creatively*, Hillsdale: Lawrence Erlbaum.

Schank, Roger C. y Robert P. Abelson, 1977: *Scripts, plans, goals and understanding: an inquiry into human knowledge structures*, Hillsdale: Lawrence Erlbaum.

Schank, Roger C. y Christopher K. Riesbeck (eds.), 1981: *Inside computer understanding: five programs plus miniatures*, Hillsdale: Lawrence Erlbaum.

Schank, Roger C. y otros, 1973: "Margie: memory, analysis, response generation, and inference on English" en *Proceedings of the Third International Joint Conference on Artificial Intelligence.* Stanford, 255-261.

Shannon, Claude E., 1948: "A mathematical theory of communication", *Bell System Technical Journal* 27, 379-423.

Sowa, John F., 2000: "Ontology, metadata, and semiotics" en Bernhard Ganter y Guy Mineau (eds.): *Conceptual structures: logical, linguistics, and computational issues*, Berlín: Springer, 55-81.

Sparck Jones, Karen, 1996: "How much has information technology contributed to linguistics?" en *British Academy Symposium on Information Technology and Scholarly Disciplines*.

Uszkoreit, Hans, 1986: "Categorial unification grammars" en *Proceedings of the 11th International Conference on Computational Linguistics*, Association of Computational Linguistics.

Van Valin, Robert D. Jr., 2005: *Exploring the syntax-semantics interface*, Cambridge: Cambridge University Press.

Van Valin, Robert D. Jr. y Randy J. LaPolla, 1997: *Syntax, structure, meaning and function*, Cambridge: Cambridge University Press.

Vendler, Zeno, 1967: *Linguistics in philosophy*, Ithaca: Cornell University Press.

Weizenbaum, Joseph, 1966: "ELIZA: a computer program for the study of natural language communication between man and machine", *Communications of the ACM* 9: 36-45.

Wilensky, Robert, 1978: *Understanding goal-based stories*. Tesis doctoral, Yale University.

Wilks, Yorick, 2005: "Computational linguistics: history" en *Encyclopedia of Language and Linguistics*, segunda edición, Oxford: Elsevier, 761-769.

Winograd, Terry, 1972: *Understanding natural language*, San Diego: Academic Press.

Wintner, Shuly, 2009: "What science underlies natural language engineering?", *Computational Linguistics* 35 (4), 641-644.