Etiquetado Estadístico de Roles Semánticos

Fermín L. Cruz Mata, 31719167M

fcruz@us.es

Supervisado por Prof. Dr. José Antonio Troyano Jiménez

Departamento de
Lenguajes y Sistemas Informáticos
Universidad de Sevilla

Memoria del Periodo de Investigación
en el Departamento de Lenguajes y Sistemas Informáticos
de la Universidad de Sevilla.

(Periodo de Investigación)

Tabla de Contenido

Índice de Figuras

Índice de Tablas

Agradecimientos

A mi familia, amigos y compañeros de grupo y departamento, especialmente a los que se incluyen en más de una categoría.

Abstract

En el presente trabajo se expone el estado del arte en la tarea del Etiquetado de Roles Semánticos, que se enmarca dentro de la disciplina del Procesamiento del Lenguaje Natural. El Etiquetado de Roles Semánticos permite la formalización semántica del lenguaje natural, permitiendo la implementación de procesos que trabajen con dicha información semántica, tales como sistemas de diálogo en lenguaje natural, traductores automáticos del habla o recuperadores semánticos de información. Se exponen además un contexto investigador y un conjunto de líneas de trabajo futuro en el área.

Prefacio

En la sociedad que vivimos, la ingente cantidad de información disponible hace necesario el empleo de técnicas de procesamiento automático de la misma, para permitir llevar a cabo búsquedas en dicha información, extraer estadísticas, y en general dotarla de utilidad para la comunidad y el individuo. Tradicionalmente, con la aparición de los ordenadores y la computación, el hombre ha dedicado un enorme esfuerzo a la codificación y formalización de la información para expresarla en términos manejables por máquinas algorítmicas. A lo largo de la historia de la informática, sin embargo, se ha ido dando importancia cada vez más a acercar el ámbito de trabajo de los ordenadores al de los humanos, en vez de ser éstos quienes se adapten a las formas de trabajar de las máquinas. Ejemplo de esto sería la evolución de los lenguajes de programación o de las interfaces hombre-máquina. En este camino de humanización de los ordenadores un paso fundamental es la manipulación directa y la comprensión de la información en el lenguaje utilizado por los seres humanos, el lenguaje natural. Llevar a buen puerto esta misión acabaría con los esfuerzos dedicados a formalizar la información para su consumo computacional, y permitiría a las aplicaciones acceder directamente a la enorme de cantidad de información disponible en lenguaje natural en libros, periódicos, páginas web,....

Como se expondrá en este trabajo, este objetivo último es tremendamente ambicioso. Multitud de problemas aparecen al tratar de alcanzarlo, siendo imprescindible la división en subproblemas de más fácil solución. Algunos de estos problemas consisten en detectar la estructura subyacente existente en los textos en lenguaje natural a distintos niveles (léxico, sintáctico, semántico,...). La tarea que nos ocupa en este trabajo consiste en inferir y explicitar automáticamente la estructura semántica subyacente en las oraciones, utilizando para ello un acercamiento basado en predicados y argumentos semánticos conocido como *frame semantics* [15]. Los sistemas que abordan esta tarea etiquetan los constituyentes de la oración con información acerca del rol que desempeñan desde un punto de vista semántico, y son conocidos como *etiquetadores de roles semánticos*. En inglés, el problema es conocido como *Semantic Role Labeling* (SRL).

Estructura del documento

En el capítulo 1 se planteará una introducción al Procesamiento del Lenguaje Natural (PLN), haciendo hincapié en la vertiente estadística de dicha disciplina. Se lleva a cabo una introducción a las técnicas de aprendizaje automático y a algunas de las aplicaciones clásicas del PLN.

Los capítulos 2,3 y 4 se centran en describir todo lo relacionado con el etiquetado de roles semánticos. En primer lugar, el capítulo 2 describe en detalle en qué consiste la tarea, y de qué manera su resolución puede ayudar a resolver o mejorar otras tareas del Procesamiento del Lenguaje Natural. En el capítulo 3 se describen los principales recursos semánticos disponibles, cuya aparición ha permitido abordar la implementación de etiquetadores de roles semánticos estadísticos. Por último, en el capítulo 4 se lleva a cabo una descripción de la arquitectura-tipo de los actuales etiquetadores de roles semánticos estadísticos, describiendo cada una de las fases de dicha arquitectura y las características utilizadas en los algoritmos de aprendizaje automático de estos sistemas.

El capítulo 5 se centra en mi proyecto investigador. En primer lugar se resume mi experiencia anterior en investigación. Después se expone el escenario actual de trabajo en etiquetado de roles semánticos: grupos de investigación dedicados al tema y revistas y congresos que se encuentran dentro del ámbito. Finalmente se exponen las líneas de trabajo futuro y una planificación temporal de las subtareas necesarias para desempeñarlas.

Capítulo 1

Introducción al Procesamiento del Lenguaje Natural

1.1 Introducción

El Procesamiento del Lenguaje Natural es una disciplina que estudia la implementación de procedimientos y algoritmos que permitan a los ordenadores analizar, generar y sobretodo *comprender* el lenguaje natural utilizado por los humanos para comunicarse. En realidad ésta sería una definición simplificada de todo lo que se puede considerar englobado por el Procesamiento del Lenguaje Natural. En general, toda tarea computacional relacionada con el lenguaje natural se puede enmarcar dentro de la disciplina en cuestión: traductores automáticos, interfaces hombre-máquina basados en lenguaje natural, buscadores de información textual, ... Pero el objetivo último que motiva todas estas aplicaciones y que las resolvería de golpe si fuese alcanzado es sin duda el de la comprensión del lenguaje natural.

Como puede imaginarse fácilmente, este objetivo dista mucho de ser trivial y hoy día es incluso considerado por algunos como utópico. El lenguaje utilizado por los humanos está muy alejado del lenguaje formal procesable directamente por un autómata: es ambiguo, en ocasiones incluso ofuscado, y el mensaje transmitido en el mismo es fuertemente dependiente de los conocimientos previos sobre la realidad objetiva y subjetiva de los participantes en la comunicación. Considérese este ejemplo extraído de [28]:

At last, a computer that understands you like your mother

Esta frase se utilizó como eslogan publicitario de una compañía que desarrollaba sistemas hombre-máquina en la década de los ochenta. La misma frase sirve para ilustrar el problema de la ambigüedad del lenguaje natural, principal quebradero de cabeza de los investigadores en el área del Procesamiento

del Lenguaje Natural. Es muy posible que precisamente los sistemas de entendimiento del lenguaje natural desarrollados por la compañía en cuestión no fuesen capaces de interpretar adecuadamente el mensaje trasmitido por la frase. Si nos paramos a pensarla detenidamente, veremos que la frase puede tener tres interpretaciones distintas:

- El ordenador te entiende tan bien como lo haría tu madre.
- El ordenador entiende que te gusta tu madre.
- El ordenador te entiende tan bien como entendería a tu madre.

Sin embargo, nuestra mente funciona de tal manera que a simple vista ni siquiera parece percibir ninguna ambigüedad, descartando las opciones segunda y tercera y quedándose con la primera. Para hacer esto, se basa en el conocimiento previo que posee del mundo: qué tipo de relación se espera entre una madre y un hijo, qué intención suelen tener las frases publicitarias, ...

Aplicando a la problemática propuesta por el Procesamiento del Lenguaje Natural una visión ingenieril, se ha tratado a lo largo de los años de descomponer los problemas finales que se pretenden resolver en una serie de subproblemas menores, algunos de los cuáles han sido ya resueltos en menor o mayor medida. Estos subproblemas se pueden enmarcar en ocasiones en la denominada Lingüistica Computacional, que trata de determinar las estructuras lingüisticas implícitas en los textos. Han sido así implementados etiquetadores morfosintácticos, analizadores sintácticos, reconocedores de entidades, *chunkers*, ... Todos ellos son herramientas que dotan de una estructura lo más formal posible a lo que en un principio es un texto totalmente crudo, posibilitando la utilización de dicha estructura para la consecución de tareas de más alto nivel más cercanas al objetivo original del procesamiento del lenguaje natural (la *comprensión* del lenguaje). Este objetivo último, que puede ser subdividido también en subobjetivos menores o enfocado desde distintas perspectivas (sistemas de diálogo, recuperación de información, traducción automática...) tiene connotaciones propias de la Inteligencia Artificial.

Por tanto, y según la tarea en concreto en la que se centre, el Procesamiento del Lenguaje Natural puede considerarse una disciplina a caballos entre la Lingüística Computacional y la Inteligencia Artificial. Son por tanto informáticos y lingüistas principalmente quienes se ocupan del estudio de la misma, aunque también participan ocasionalmente matemáticos, psicólogos e incluso filósofos. Se trata entonces de un área de trabajo fuertemente multidisciplinar, con objetivos finales aún muy lejanos pero motivadores de otros subobjetivos abordables a más corto plazo e igualmente útiles en una sociedad de la información donde el tratamiento algorítmico de la información contenida en el lenguaje natural se hace imprescindible. Un buen libro para completar la información contenida en el presente capítulo es [20].

1.2 Niveles de análisis del lenguaje natural

En la necesaria tarea de análisis y formalización del lenguaje, partiendo del texto en crudo y tratando de llegar al entendimiento computacional del lenguaje natural, se suelen distinguir cuatro niveles, cada uno de los cuales se apoya en el anterior, y con una dificultad creciente de resolución. Estos niveles son:

Análisis morfológico. El análisis de las palabras para extraer raíces, rasgos flexivos, unidades léxicas compuestas y otros fenómenos. Por ejemplo, la palabra *útiles* tendrá como lexema *útil*. Una de las principales ventajes de contar con un análisis morfológico consistente es su aplicación a corpus de textos, lo que posibilita un estudio estadístico que consiga un mayor nivel de generalización al tener en cuenta la lexicalización de las palabras. Conocer algunas propiedades morfológicas de las palabras, como pueden ser el género y el número, también será de gran ayuda en la resolución de problemas como la correferencia, o incluso el etiquetado de roles semánticos que nos ocupa.

Análisis sintáctico. El análisis de la estructura sintáctica de la frase mediante una gramática de la lengua en cuestión. Se trata de ver como los distintos constituyentes de la frase se combinan entre sí para dar lugar a las oraciones: las palabras forman sintagmas, los sintagmas forman cláusulas y proposiciones, y éstas forman oraciones. El análisis sintáctico ha sido tradicionalmente llevado a cabo mediante la utilización de gramáticas y sistemas basados en reglas, pero actualmente se utilizan analizadores estadísticos que consiguen muy buenos resultados, aunque aún no exentos de fallos. En la tarea que nos ocupa en el presente informe, la información extraída de los analizadores sintácticos será fundamental para decidir la función semántica desempeñada por los constituyentes. Por tanto, se dedicará una sección del capítulo actual a describir brevemente los analizadores sintácticos utilizados actualmente.

Análisis semántico. La extracción del significado de la frase, y la resolución de ambigüedades léxicas y estructurales. El nivel semántico se asienta en el sintáctico y en el morfológico, puesto que parte del contenido del mensaje viene implícito en las estructuras y relaciones entre las palabras. Es en este nivel donde empezamos a preocuparnos por el modelo conceptual o mental existente detrás del texto, y donde surgen las mayores dificultades, puesto que se entra en terrenos propios de la Inteligencia Artificial. El etiquetado de roles semánticos se enmarcaría dentro de este nivel de análisis del lenguaje, y como se verá más adelante en este trabajo proporcionará una vez resuelto adecuadamente un nivel de formalización semántica útil para la implementación de cualquier tarea del Procesamiento del Lenguaje Natural relacionada con la semántica.

Análisis pragmático. El análisis del texto más allá de los límites de la frase, por ejemplo, para determinar los antecedentes referenciales de los pronombres. En el análisis pragmático, entran en juego fenómenos relacionados

con el conocimiento previo de la realidad del emisor y del receptor del mensaje codificado en el texto. Es el nivel más próximo a la disciplina de la Inteligencia Artificial, y también el menos desarrollado actualmente. Ejemplos de fenómenos habituales en el lenguaje y que dan una idea de la dificultad del análisis a este nivel serían la ironía y el humor, los valores morales o los estados anímicos de los participantes de una conversación.

1.3 Tareas de etiquetado

Para llevar a cabo el análisis del lenguaje natural a cada uno de los niveles expuestos anteriormente, se implementan algoritmos que deben deducir a partir del texto la información y las estructuras subyacentes a nivel léxico, sintáctico, semántico y pragmático. La mayoría de estas tareas se pueden abordar entendiéndolas como problemas de etiquetado estadístico. Esto es, dada una secuencia de palabras de entrada (o genéricamente unidades, que pueden ser por ejemplo nodos de un árbol obtenido en un análisis anterior) se trata de asignarles a cada cual una o varias etiquetas, elegidas de entre un conjunto de etiquetas posibles. Dichas etiquetas añadirán información formal o estructural al nivel que estemos trabajando.

Por ejemplo, consideremos el problema clásico del etiquetado morfosintáctico de palabras, que fue uno de los primeros en ser atacado estadísticamente y que hoy en día consigue unos resultados muy precisos. En esta tarea, dada una secuencia de palabras que conforman una oración, trataremos de asignarle a cada palabra una etiqueta correspondiente a la función morfosintáctica que le corresponde, esto es, nombre, *verbo, adjetivo, adverbio*, ... Veamos un ejemplo:

Secuencia de entrada :

Tu desconfianza me inquieta y tu silencio me ofende.

Secuencia de salida :

Tu[ADJ] desconfianza[NOM] me[PRON] inquieta[VERB] y[CONJ] tu[ADJ] silencio[NOM] me[PRON] ofende[VERB].

Si consultamos un diccionario léxico, la mayoría de las palabras poseen una única categoría morfosintáctica posible. Sin embargo, la palabra *inquieta* puede funcionar como verbo o como adjetivo, dependiendo del contexto en el que aparezca. Son este tipo de ambigüedades las que debe resolver el sistema.

Las tareas de etiquetado, que en principio pueden abordarse mediante sistemas arriba-abajo, esto es, mediante pasos o reglas secuenciales que resuelvan las ambigüedades según algún algoritmo concreto (enfoque racionalista, como se verá en la siguiente sección), se adaptan muy bien a ser atacadas mediante clasificadores estadísticos o de aprendizaje automático. Tal como se verá más adelante en este capítulo, estas técnicas consisten en la construcción de un modelo estadístico que, a partir de un aprendizaje o entrenamiento, es capaz de

aprender a llevar a cabo una clasificación de nuevos casos similar a la observada en el proceso de entrenamiento. En el caso de las tareas de etiquetado, podemos construir nuestro modelo estadístico y entrenarlo a partir de corpus anotados, utilizando posteriormente dichos modelos para el etiquetado automático de nuevas oraciones.

Los clasificadores estadísticos llamados *generativos* como los basados en modelos ocultos de Markov se adaptan bien a problemas de la naturaleza del etiquetado morfosintáctico, en los que podemos hablar de un etiquetado secuencial puro. Esto es, dada una palabra, a partir de la misma y del contexto de ésta, hemos de decidir la etiqueta que aplicamos a la misma. En este tipo de tareas las decisiones que hay que tomar son relativamente locales. Por ello, el sistema puede funcionar de manera secuencial. Palabra a palabra, el etiquetador decide en función del contexto o de determinadas características cuál es la etiqueta más probable para la palabra, y después se pasa a la siguiente.

Existen otras tareas sin embargo que por su naturaleza no se adaptan bien a ser abordadas de manera puramente secuencial. Es el caso de los analizadores sintácticos, o de la tarea de etiquetado semántico que nos ocupa en este trabajo. El problema es que ahora las estructuras de salida tienen un componente jerárquico, de manera que existen constituyentes que hay que etiquetar y que están contenidos a su vez en otros constituyentes. El etiquetado resultante es más complejo que asignar una etiqueta a cada palabra, y la implementación del etiquetador exige la utilización de otro tipo de algoritmos de aprendizaje, conocidos como discriminativos, junto a técnicas de inferencia o decodificación, generalmente realizadas mediante programación dinámica, que se encargan de combinar los resultados parciales propuestos por los algoritmos discriminativos para asegurar que la estructura final de salida cumpla una serie de requisitos globales.

Los algoritmos de aprendizaje discriminativo más empleados serán expuestos en una sección siguiente de este mismo capítulo. Así mismo, el capítulo que habla sobre la arquitectura de los etiquetadores de roles semánticos estadísticos, sirve como ejemplo de los problemas encontrados al abordar una tarea de etiquetado jerárquico, y de posibles maneras de resolverlos.

Para evaluar los resultados obtenidos por un etiquetador, se utilizan dos medidas llamadas precisión (*precision*) y cobertura (*recall*).

Precision: es la proporción de palabras o constituyentes etiquetados correctamente, del total de palabras o constituyentes etiquetados en el corpus de test.

Recall: es la proporción de etiquetas asignadas correctamente por el etiquetador del total de etiquetas correctas del corpus de test.

Ambas medidas son en cierto modo contrarias: es mas sencillo conseguir una cobertura alta a costa a reducir la precisión, y viceversa. Es por ello que

se utiliza una medida que combina las dos anteriores, con objeto de facilitar la optimización y la comparación de distintos etiquetadores. Esta medida se calcula de la siguiente forma:

$$F_{\beta=1} = \frac{2 * Precision * Recall}{Precision + Recall}$$

1.4 Racionalismo y empirismo en el Procesamiento del Lenguaje Natural

Los investigadores en el área del Procesamiento del Lenguaje Natural se han visto influenciados a lo largo de la corta historia de la disciplina por dos visiones filosóficas contrarias sobre la mente y el conocimiento humano. Estas dos aproximaciones son la racionalista y la empirista, y es interesante observar los periodos de tiempo en los que cada una de estas tendencias eran las predominantes en el pensamiento científico y su influencia en la forma de afrontar la resolución de los problemas del Procesamiento del Lenguaje Natural.

La aproximación racionalista se caracteriza por la creencia de que una parte significativa del conocimiento y las capacidades de manipulación del mismo por parte de la mente humana son innatos. Si nos centramos en lo relativo a la lingüística, los racionalistas defenderían que el cerebro está genéticamente dotado de una serie de mecanismos y facultades para el manejo de la información lingüística. Quienes sostienen estos postulados, argumentan que si no fuese así sería difícil de entender que los niños puedan aprender algo tan complejo como el lenguaje natural a partir de los limitados estímulos que reciben durante su primera infancia. Esta argumentación es conocida como la *pobreza de los estímulos* y fue planteada por Noam Chomsky, principal defensor de la visión racionalista en la lingüística. El lenguaje humano sería por tanto la proyección o el resultado necesario de las estructuras mentales innatas del ser humano.

Aquellos investigadores influidos por el racionalismo que abordaban la construcción de sistemas que pretendían manipular y comprender algorítmicamente el lenguaje humano, intentan implementar dichos mecanismos mentales innatos a mano, incorporando un conjunto inicial de conocimientos y reglas de razonamiento para manipular dicho conocimiento. Se emprende así la tarea de averiguar mediante análisis cuáles son esos mecanismo e implementarlos algorítmicamente. Este enfoque se mantiene aproximadamente desde 1960 hasta 1985.

La aproximación empirista aporta una visión distinta de la mente humana y de su capacidad para manejar el lenguaje. Ciertamente, los empiristas parten de una base común a los racionalistas: también para ellos el cerebro posee una serie de habilidades cognitivas innatas necesarias para el pensamiento abstracto.

Pero estas capacidades iniciales son absolutamente generales y carentes de una finalidad instrínseca a priori. No existen mecanismos y procedimientos innatos específicos en el cerebro para manejar el lenguaje, o para otras tareas abstractas, sino que la mente de un bebé implementa simplemente operaciones generales que permiten a partir de las experiencias sensoriales generalizar, reconor patrones, asociar y en definitiva aprender. Estas capacidades iniciales otorgan a la mente una extraordinaria capacidad plástica, de manera que a partir de las numerosas entradas sensoriales a la misma propician la construcción mediante aprendizaje de los circuitos neuronales necesarios para resolver problemas como entender la estructura y contenido del lenguaje natural. Si en la visión racionalista el lenguaje natural es el resultado necesario o la proyección única de las estructuras mentales del hombre, en la concepción empirista es la mente la que se adapta y especializa al lenguaje que culturalmente le haya tocado en gracia (teniendo en cuenta que no dejan de existir unos mecanismos mentales básicos que determinan los límites del conocimiento, y que el lenguaje y cualquier otra habilidad abstracta del hombre debe estar dentro de dichos márgenes).

La implicación práctica de aplicar la visión empirista al Procesamiento del Lenguaje Natural es un cambio de base en la construcción de los sistemas dirigidos al manejo del lenguaje. Si en el enfoque racionalista los investigadores trataban de analizar los mecanismos mentales que se encargan de manipular y entender el lenguaje e implementarlos mediante algoritmos, ahora el acercamiento al problema se lleva a cabo desde un punto de vista principalmente estadístico: se parte de un conjunto de datos lo suficientemente amplio y , según los casos, convenientemente enriquecido por expertos, y a partir de dichos datos se tratan de construir modelos probabilísticos que *aprendan* o *generalicen* las estructuras y fenómenos complejos que se producen en dichos textos y sean capaces de utilizar el conocimiento extraído para llevar a cabo las tareas de las que se ocuparon los expertos en los datos iniciales.

La visión empirista fue la predominante en los primeros trabajos teóricos relacionados con el lenguaje natural, entre los años 1920 y 1960. Estos primeros trabajos no podían ser implementados en un primer momento pues la tecnología aún no lo permitía. Los primeros experimentos prácticos sin embargo resultaron desalentadores, lo que condujo a un cambio de enfoque a partir de la década de los 60, interesándose los investigadores de la época por el enfoque racionalista. Sin embargo, a partir de 1985, algunos resultados prometedores utilizando una aproximación estadística, primeramente en el reconocimiento automático del habla y posteriormente en otras áreas como la traducción automática (ambas impulsadas por investigadores de la empresa IBM), han hecho reconsiderar el planteamiento empirista. Actualmente, con máquinas con capacidad de cálculo y almacenamiento considerablemente superiores a las de 1960, y con un creciente número de recursos lingüisticos y corpus de textos, las técnicas estadísticas ocupan un lugar central en la mayoría de las tareas del Procesamiento del Lenguaje Natural.

1.5 Enfoque estadístico y aprendizaje automático en el Procesamiento del Lenguaje Natural

En el presente trabajo se estudia el estado del arte en la construcción de sistemas de etiquetado de roles semánticos estadísticos, estos es, basados en técnicas de aprendizaje automático. Se hará en la presente sección una pequeña introducción a los principales algoritmos de clasificación utilizados en la disciplina del Procesamiento del Lenguaje Natural y en concreto en los etiquetadores de roles semánticos que serán descritos más adelante en el presente informe. Un buen libro sobre Procesamiento del Lenguaje Natural Estadístico es [32].

Primeramente, hemos de distinguir entre aprendizaje automático supervisado y no supervisado. La modalidad supervisada parte de una serie de ejemplos ya clasificados (o de forma general, ya asociados a una determinada estructura de salida), a partir de los cuáles se construyen modelos que intentan capturar la manera en que se conectan las entradas y las salidas. Este proceso es conocido como entrenamiento. De esta forma ante la llegada de nuevos ejemplos, se utiliza el modelo construido para deducir la clase o estructura de salida más probable. En el caso del aprendizaje no supervisado, la fase de entrenamiento se lleva a cabo sin disponer de ejemplos previamente clasificados o asociados a estructura alguna, tratándose de extraer el conocimiento directamente de un conjunto de datos al desnudo. Este segundo tipo de aprendizaje es aplicable a algunos problemas dentro del Procesamiento del Lenguaje Natural, especialmente a aquellos en los que hay que resolver determinadas ambigüedades, pudiéndose encontrar en los mismos datos de entrada ejemplos de como resolver las mismas. Un ejemplo de esto sería el etiquetador morfosintáctico de Brill, basado en el algoritmo *transformation-based learning (TBL)* [4]. Pero para otros muchos problemas, como el que nos ocupa en este trabajo, es necesario aplicar aprendizaje automático supervisado, ya que la entrada en si no contiene suficiente información como para inducir modelos predictivos directamente a partir de ella.

El aprendizaje automático supervisado se puede plantear en términos generales de la siguiente forma. Se parte de un conjunto de datos de entrenamiento y de una función de error definidos de la siguiente manera:

Conjunto de datos de entrenamiento: está formado por ejemplos (x, y) donde:

- $x \in X$ son datos de entrada, por ejemplo, frases o palabras.
- $y \in Y$ son las clases o estructuras a las que corresponden los datos de entrada, por ejemplo, estructuras lingüisticas.
- El conjunto se supone que se ha generado siguiendo cierta distribución D desconocida sobre $X \times Y$.

Función error o pérdida: que se define como:

- $error(y, \hat{y})$ = coste de proponer $\hat{y}$ cuando el valor correcto de salida era y.

	Clases contenidas en Y	$\|Y\|$	**Enumeración de Y**	**Error**
Clasificadores binarios	$\{x, y\}$	2	No es necesaria	0 - 1
Clasificadores de multiclases	A,B,C,...	m	Exhaustiva	0 - 1
Aprendizaje de estructuras	todas las estructuras posibles	exponencial	No es enumerable	*precision* y *recall* en los nodos

Tabla 1.1: Tipos de aprendizaje automático supervisado

Con estas premisas, el objetivo buscado es calcular una función *hipótesis*, $h : X \rightarrow Y$, que minimice el error en la distribución D.

En la mayoría de ocasiones, no se trabaja con los datos de entrada directamente, sino con una versión enriquecida de los mismos, obtenida mediante cierta transformación que proporciona a partir de cada dato de entrada un vector conocido como vector de características. Estas características reflejan propiedades de la entrada que los diseñadores del sistema consideran útiles para decidir cuál es la salida correspondiente. Por ejemplo, si estamos trabajando en la resolución de ambigüedades morfosintácticas, una característica útil que debería formar parte del vector de características sería la categoría morfosintáctica de la palabra anterior. Incluyendo dicha característica, posibilitamos que el algoritmo de aprendizaje encuentre las posibles correlaciones entre ambigüedades a resolver y determinadas categorías morfosintácticas de la palabra anterior. Por ejemplo, existirá previsiblemente una fuerte tendencia a etiquetar como *nombre* una palabra con ambigüedad entre *nombre* y *verbo* que venga precedida por una palabra que funcione como *determinante*.

Según la cardinalidad y el tipo de los elementos que conformen el conjunto de salida, los algoritmos de aprendizaje supervisado se dividen en clasificadores binarios, clasificadores de multiclases y aprendizaje de estructuras. Algunas características de los mismos pueden verse en la tabla 1.1. Mientras que en los clasificadores, ya sean binarios o de multiclases, se dispone a priori del conjunto de clases de salida, en el aprendizaje de estructuras la salida puede ser por ejemplo una estructura en forma de árbol, con virtualmente cualquier configuración y forma. Un ejemplo de aprendizaje de estructuras serían los analizadores sintácticos estadísticos que serán comentados en la siguiente sección. Los etiquetadores de roles semánticos también son un ejemplo de aplicación del aprendizaje de estructuras.

Además de la división en supervisados y no supervisados, los algoritmos de aprendizaje pueden clasificarse en generativos o discriminativos. En los algoritmos generativos, se estiman probabilidades de la entrada en función de la salida (en el caso de los modelos ocultos de Markov, basado en este enfoque, dichas probabilidades son conocidas como probabilidades de emisión). Para llevar a cabo este enfoque es necesaria la construcción de algún tipo de representación gráfica del mecanismo de generación, como un autómata o una gramática, en la que se hagan patente las dependencias de las entradas y las salidas (todo esto quedará más claro cuando se expongan los modelos ocultos de Markov seguidamente).

Los modelos generativos permiten una gran eficiencia en el entrenamiento y aún más en el etiquetado, y funcionan muy bien en tareas de desambiguación locales, como el etiquetado morfosintáctico, pero son restrictivos a la hora de poder definir características de manera flexible, y exigen realizar suposiciones de independencia que no siempre se pueden justificar. Por el contrario, los algoritmos discriminativos (salvo los modelos ocultos de Markov, el resto de los que se exponen en los siguientes apartados) tratan de modelar directamente la probabilidad de las salidas condicionada a las entradas. Se requieren algoritmos más complejos para esto, que requieren de mas recursos para el entrenamiento y para el etiquetado, pero que permiten mayor flexibilidad en la definición de características y no imponen tantas restricciones de independencia como los modelos generativos.

1.5.1 Modelos ocultos de Markov

Los modelos ocultos de Markov ([38]) consisten en un autómata probabilístico, en el cual los estados son los posibles valores de salida del clasificador y las transiciones entre los estados *emiten* los valores de entrada. Se parte de una planteamiento probabilístico del problema que se quiere resolver, que viene a ser la siguiente optimización:

$$argmax_{y_1,\ldots,y_n} P(y_1,\ldots,y_n \mid x_1,\ldots,x_n)$$

Aplicando la regla de Bayes, podemos reescribir la optimización anterior de la siguiente manera:

$$argmax_{y_1,\ldots,y_n} \frac{P(x_1,\ldots,x_n \mid y_1,\ldots,y_n) \cdot P(y_1,\ldots,y_n)}{P(x_1,\ldots,x_n)}$$

La suposición necesaria para poder emplear los modelos ocultos de Markov es que cada entrada depende únicamente de la salida para dicha entrada y de un número concreto de salidas anteriores. Se define así el orden de Markov como el número de salidas de las que depende la entrada. Por ejemplo, para un modelo de Markov de orden 2, y teniendo en cuenta que el denominador es constante, la optimización anterior quedaría de la siguiente forma:

$$argmax_{y_1,\ldots,y_n} \prod_{k=1}^{n} P(y_k \mid y_{k-2}, y_{k-1}) \cdot P(x_k \mid y_k)$$

Por lo tanto, será necesario calcular las siguientes probabilidades:

- **Probabilidades de emision:** $P(x_k \mid y_k)$.
- **Probabilidades de transición:** $P(y_k \mid y_{k-2}, y_{k-1})$ (para un modelo de Markov de segundo orden).

- **Probabilidad del estado inicial:** $P(y_1)$

Para calcular las probabilidades se usan estimaciones de máxima verosimilitud, que son aproximaciones calculadas mediante simples conteos de los datos del corpus de entrenamiento, de la siguiente manera:

Probabilidades de emision: se cuenta el número de veces que la palabra x_k aparece etiquetada con la etiqueta y_k, y se divide entre el total de apariciones de la palabra.

Probabilidades de transición: para un modelo de segundo orden, se cuenta el número de apariciones de la secuencia de etiquetas y_{k-2}, y_{k-1}, y_k (en terminología lingüística, se habla de *trigramas*),y se divide entre el total de apariciones de la etiqueta y_k.

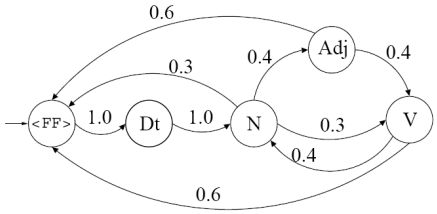
Probabilidad del estado inicial: se cuenta el número de apariciones de la etiqueta y_1 y se divide entre el total de etiquetas posibles.

El principal problema de la aplicación de las estimaciones de máxima verosimilitud es que en ocasiones se estiman ciertas probabilidades como nulas, lo cual anula el cálculo de la optimización, ya que un factor igual a 0 implica que el productorio de probabilidades sea nulo. Para evitar esto se utilizan técnicas conocidas como *suavizado* en las que no se entrará en el presente trabajo.

Los nombres de estas probabilidades hacen referencia a la representación en forma de autómata probabilístico de la que hablamos antes. Se puede evitar el calculo de la probabilidad del estado inicial creando uno con una etiqueta especial de salida, y asignándole probabilidad igual a 1. En la figura 1.1 se muestra un ejemplo de representación en forma de autómata probabilístico y matriz de probabilidades de emisión para un etiquetador morfosintáctico basado en modelos ocultos de Markov de primer orden, en el que las entradas son las palabras y las salidas son categorías morfosintácticas.

Una vez construido el modelo, el proceso de etiquetado se lleva a cabo buscando la secuencia de valores de salida que maximizan la probabilidad globalmente. Para ello, se utilizan técnicas de programación dinámica, generalmente el algoritmo de Viterbi, que nos asegura una complejidad lineal con respecto al número de datos de entrada y una solución óptima.

Los modelos ocultos de Markov son una opción interesante dada su facilidad de implementación, y sobretodo por su velocidad tanto en entrenamiento como en su aplicación posterior. Pero la simplificación hecha al imponer que un dato de entrada sólo pueda depender de un subconjunto de los datos de salida hace poco recomendable su aplicación en problemas de etiquetado jerárquico como el que nos ocupa en este trabajo. Para este tipo de problemas se suelen utilizar modelos de aprendizaje discriminativo como los que serán descritos a continuación.



Emission probabilities	.	el	la	gato	niña	come	corre	pescado	fresco	pequeña	grande
<FF>	1.0										
Dt		0.6	0.4								
N				0.6	0.1			0.3			
V						0.7	0.3				
Adj									0.3	0.3	0.4

Figura 1.1: Ejemplo de autómata y matriz de emisión de un modelo de Markov de segundo orden aplicado al problema del etiquetado morfosintáctico.

1.5.2 Conditional Random Fields

Los clasificadores basados en *conditional random fields* ([26]) están construidos, al igual que los modelos ocultos de Markov, sobre una representación gráfica de las dependencias entre las variables aleatorias de entrada y salida X e Y. Como se ha visto anteriormente, en los modelos ocultos de Markov se pretende modelar la probabilidad conjunta $P(X, Y)$, descomponiéndola en probabilidad de transición y probabilidad de emisión ($P(X|Y) \cdot P(Y)$). Para poder hacer esto, que se conoce como enfoque generativo, es necesario imponer condiciones de independencia entre las variables aleatorias de la secuencia. Sin embargo, en un clasificador del tipo que nos ocupa se modela la probabilidad $P(Y \mid x)$. El grafo que representa el modelo se compone de un único nodo representando la secuencia de entrada completa, del que dependen una serie de nodos que representan cada una de las etiquetas de salida (figura 1.2).

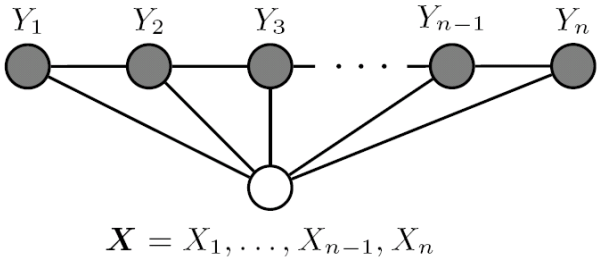


Figura 1.2: Estructura gráfica de un clasificador basado en Conditional Random Fields

La probabilidad de una secuencia de etiquetas y dada una observación concreta de entrada x se define como un producto normalizado de funciones de potencia, cada una de la siguiente forma:

$$exp(\sum_j \lambda_j t_j(y_{i-1}, y_i, x, i) + \sum_k \mu_k s_k(y_i, x, i))$$

, siendo $t_j(y_{i-1}, y_i, x, i)$ las funciones de transición que modelan la dependencia entre cada dos etiquetas de salida y la secuencia completa de entrada, a partir de las características seleccionadas por los diseñadores del sistema como relevantes para el problema que se esté modelando; y $s_k(y_i, x, i)$ son las funciones de estado que actúan de forma similar pero considerando la etiqueta concreta de manera individual. Las parámetros λ y μ son los valores que han de ser estimados mediante un determinado algoritmo en el proceso de entrenamiento del modelo.

Ambas familias de funciones se conocen bajo el nombre común de funciones de características, y para definirlas primero hay que definir las características del etiquetador mediante expresiones binarias como la siguiente:

$$b(x, i) = \begin{cases} 1 \text{ si la observación } x_i \text{ es la palabra } \textit{casa} \\ 0, \text{ en otro caso} \end{cases}$$

A partir de estas características, se definen las funciones de transición y de estado para cada uno de los posibles valores de salida de la variable de salida actual (y de la anterior en el caso de la función de transición). Por ejemplo, una de las funciones de transición podría ser la siguiente:

$$t_j(y_{i-1}, y_i, x, i) = \begin{cases} b(x, i), \text{ si } y_{i-i} = DET \text{ y } y_i = VB \\ 0, \text{ en otro caso} \end{cases}$$

En ocasiones, se suele simplificar la expresión general de la probabilidad de una secuencia de salida dada una secuencia de entrada entendiendo que las funciones de estado se pueden escribir también como funciones de las etiquetas actual y la anterior, aunque en la práctica la etiqueta anterior será constante en dichas funciones. Con esta consideración, la expresión queda de la siguiente forma general:

$$p(y \mid x, \lambda) = \frac{1}{Z(x)} exp(\sum \lambda_j F_j(y, x))$$

, donde $Z(x)$ es un factor de normalización para asegurar que la suma de todas las probabilidades de salida para una entrada determinada es 1.

El cálculo de los parámetros necesarios para construir el modelo se lleva a cabo buscando aquellos valores que maximicen la entropía para los datos no observados durante el entrenamiento. Este principio se conoce como máxima entropía, y bajo esta denominación encontramos toda una familia de clasificadores que utilizan una distribución de probabilidad como la explicada. El algoritmo recién explicado es el miembro de la familia de clasificadores de máxima entropía más utilizado actualmente en procesamiento del lenguaje natural.

1.5.3 Árboles de decisión

Los árboles de decisión ([35]) son una manera bastante sencilla de construir clasificadores. A partir de los datos de entrenamiento, y una vez aplicada la función de características para obtener los vectores necesarios, se construye un árbol binario (ver figura 1.3), esto es, cada nodo con dos hijos, de la siguiente manera:

- Se dispone de un conjunto de preguntas relativas a las características, de tipo lógico, generalmente relacionales. Por ejemplo, ¿es la tercera características mayor de 3.5? (suponiendo que el vector de características se define sobre los números reales). Este conjunto de preguntas puede ser definido por los diseñadores del sistema o calculado automáticamente por el algoritmo a partir de ciertas reglas.

- Se divide la población total de los datos de entrada según cumplan o no las condiciones de cada una de las preguntas, y se estima mediante cierta medida cuál de las preguntas es la que separa a la población en dos conjuntos de la manera más discriminativa posible con respecto a la salida. En las variaciones más utilizadas de algoritmos de árboles de decisión, se suele emplear la medida de *ganancia de información*, que se define a partir del concepto de entropía.

- Se escoge esa pregunta, y se almacena en un nodo, creándose dos nodos hijos, uno para los datos que hayan cumplido la condición y otro para los que no.

- Para cada uno de los nodos, se repite el proceso de encontrar la pregunta más discriminativa. Se va construyendo así el árbol, parando el proceso en aquellos nodos con una población de datos lo suficientemente homogénea según alguna medida, generalmente un determinado nivel de entropía (en el caso óptimo, cuando todos los datos contenidos en el nodo pertenezcan a la misma clase de salida).

A la hora de utilizar el clasificador, ante un dato de entrada, el algoritmo se coloca en el nodo raíz y realiza la pregunta almacenada en el mismo, optando por el camino correspondiente a la respuesta obtenida. Así se recorre el árbol hasta alcanzar un nodo hoja, decidiéndose así la clase a asignar al dato de entrada.

Existe un caso particular de árbol de decisión en el que la salida del sistema no es una clase, sino un valor real. En este tipo de árboles, llamados de regresión, se construyen polinomios de regresión con los datos caídos en los nodos hoja en el entrenamiento, de manera que ante un nuevo dato que caiga en una hoja se pueda calcular un valor real de salida.

Las ventajas del uso de árboles de decisión son fundamentalmente la velocidad del etiquetado, que se lleva a cabo en un tiempo constante dependiente del número de niveles del árbol formado, y la posibilidad de interpretar fácilmente

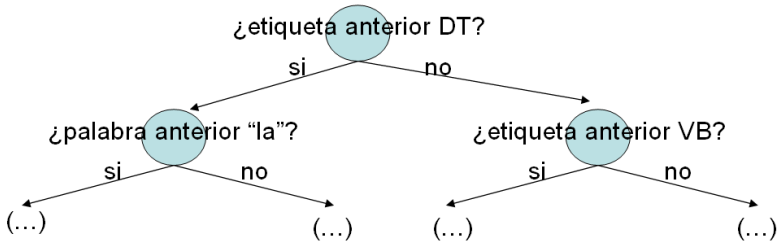


Figura 1.3: Porción de un árbol de decisión construido para etiquetar categorías morfosintácticas

el modelo. Por ejemplo, las preguntas aparecidas en los primeros nodos informarán sobre qué características de las introducidas en el sistema poseen mayor capacidad discriminativa. Por otro lado, el problema más habitual cuando se utilizan árboles de decisión como clasificadores es el fenómeno conocido como *overfitting*, que se produce cuando el árbol se adapta demasiado a los datos de entrenamiento, perdiendo por tanto capacidad de generalización, lo que repercute en malos resultados al intentar clasificar nuevas muestras no vistas durante el entrenamiento. Esto obliga a llevar a cabo un proceso de calibración de los parámetros que controlan la creación del árbol que puede ser bastante laborioso.

1.5.4 Redes neuronales artificiales

Las redes neuronales artificiales ([49]) constituyen un paradigma de aprendizaje bioinspirado que trata de imitar el funcionamiento de las redes neuronales del cerebro de los animales. De forma muy simplista, las redes neuronales del cerebro están constituidas por neuronas que se interconectan unas con otras formando complejos grafos. Cada neurona tiene una serie de prolongaciones (dendritas y axones) mediante las cuales se propagan señales químicas y eléctricas. La propiedad fundamental de las neuronas es su capacidad de reaccionar de determinada manera cuando recibe una señal , emitiendo otra señal a partir de la entrada. La forma en que se realiza esta transferencia es motivo de complejos estudios aún hoy día, pero es patente que existen mecanismos por los cuáles las neuronas son capaces de reforzar o inhibir cada una de las conexiones que poseen, de manera que la red aprende a generar determinadas salidas a partir de ciertas entradas.

Se intenta modelar este comportamiento mediante neuronas artificiales, que serán nodos de un grafo con n entradas y m salidas, y varias funciones matemáticas que determinan las salidas a partir de las entradas. Se utilizan generalmente tres tipos de funciones para describir el comportamiento de cada

neurona:

Función de propagación o de excitación : suele consistir en una combinación lineal de las entradas . La salida de la función será la suma de cada entrada a la neurona multiplicada por el peso de la conexión. Estos pesos pueden ser positivos (conexión excitatoria) o negativos (conexión inhibitoria).

Función de transferencia : esta función toma como entrada el valor devuelto por la función de propagación y lo adapta a las características exigidas para la salida por el problema en cuestión. Por ejemplo, si nuestra neurona debe devolver un valor lógico cierto o falso, la función de transferencia debe realizar una aplicación entre el valor real generado por la función de propagación y dos valores discretos 0 o 1.

Función de activación : se utiliza sólo en algunos algoritmos de redes neuronales artificiales. Añade un comportamiento no lineal a la función de propagación.

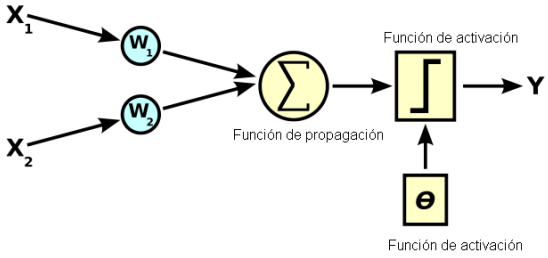


Figura 1.4: Perceptrón con dos entradas

Con estas unidades, se pueden formar topologías muy simples como el perceptrón 1.4, que consta de una sola neurona, o todo lo complejas que se pueda imaginar. La topología que se utilice determina la potencia y los problemas a los que se puede aplicar el método de aprendizaje. En el caso del etiquetado, la mayoría de los trabajos utilizan la topología más simple del perceptrón.

El perceptrón consta de n entradas a partir de las cuales genera una única salida, generalmente binaria. Para llevar a cabo el proceso de entrenamiento, se eligen unos pesos previos para cada una de las conexiones de la red y se empiezan a introducir datos de entrada. Estos datos generaran salidas aleatorias, que ocasionalmente serán correctas. Se aplica un algoritmo de aprendizaje que consiste en incrementar los pesos de las conexiones cuando se consiguen salidas correctas o en decrementarlos cuando se consiguen resultados erróneos. De esta manera, la red es capaz de aprender a identificar la correlación entre las entradas

y las salidas, y conseguirá generalizar dicha relación para predecir cuál debe ser la salida ante una entrada no observada previamente.

El perceptrón es una técnica de muy fácil implementación y muy eficiente en tiempo de ejecución, pero tiene la deventaja principal de que sólo es capaz de clasificar datos que sean linealmente separables. Cuando no se de esta característica, un algoritmo apropiado de aprendizaje serían las *support vector machines*.

1.5.5 Support Vector Machines

Las máquinas de soporte vectorial o *support vector machines* ([10]) son clasificadores lineales, esto es, dados una serie de vectores de entrada, el clasificador trata de dividir el espacio vectorial linealmente creando una serie de regiones, a cada una de las cuales se le asocia una clase de salida. Los vectores de entrada se forman a partir de las características definidas por los diseñadores del clasificador, y tendrán por tanto generalmente un número alto de dimensiones. El problema principal de los clasificadores lineales es que en ocasiones no es posible realizar una partición lineal del espacio vectorial que sea capaz de aislar todos los datos de entrada que corresponden a cada clase. Para solucionar esto, en las *support vector machines* se proyectan los vectores de entrada en un espacio de mayor dimensión. En este nuevo espacio, se construyen una serie de hiperplanos que dividen a los vectores de entrada de tal manera que se maximiza la distancia geométrica de todas las muestras con respecto a los hiperplanos. De esta manera se supone que el modelo es capaz de generalizar ante la llegada de nuevos datos de entrada no observados con el mínimo error posible.

La formulación matemática necesaria para llevar a cabo este proceso es compleja y no será objeto de estudio en el presente trabajo. Explicado de manera intuitiva, si se están intentando separar vectores de entrada pertenecientes a dos clases distintas, se construyen un par de hiperplanos de manera que cada uno de ellos engloba en una de sus caras a todos los datos de entrada de cada uno de las clases de salida, con la condición de que dichos planos contienen en el mismo el mayor número de datos de entrada posible. Es decir, los hiperplanos en cuestión serán los más próximos posibles a las muestras. Posteriormente, se calcula el hiperplano que se encuentra equidistante a los dos anteriores, y será este el que se utilice como hiperplano separador (ver figura 1.5). Una vez construido el sistema, la manera de llevar a cabo la clasificación será mediante cálculos vectoriales para decidir en cuál de las regiones delimitadas por los hiperplanos se encuentra el vector a clasificar, y se le asignará la clase correspondiente a dicha región.

El algoritmo de clasificación basado en *support vector machines* es el que utilizan los etiquetadores de roles semánticos que actualmente han conseguido los mejores resultados. El principal problema es que el tiempo empleado para el entrenamiento es muy grande, llegando a estar varios días llevándose a cabo el proceso según se relata en los artículos en los que se describen estos etiquetadores.

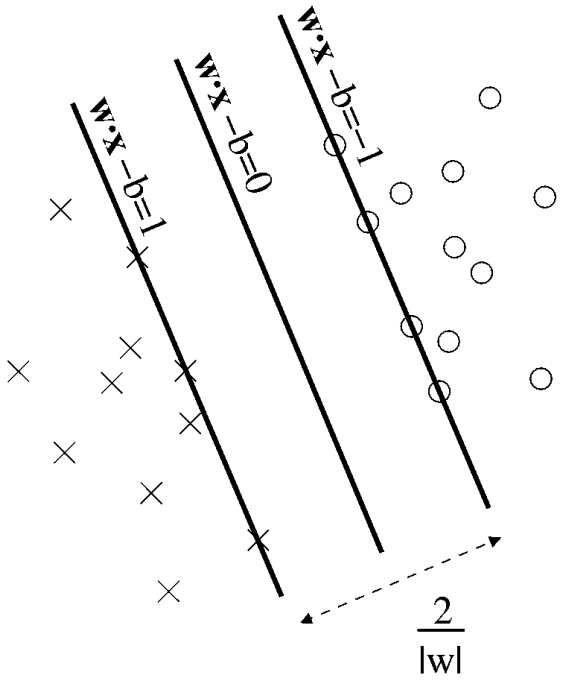


Figura 1.5: Cálculo del hiperplano que maximiza el margen geométrico en un clasificador basado en *Support Vector Machines*

1.5.6 Transformation-based learning

El algoritmo de aprendizaje basado en transformaciones fue introducido por Eric Brill en 1992, aunque la referencia más influyente sobre el mismo suele ser su artículo de 1995 [4], donde se presenta un sistema de etiquetado morfosintáctico haciendo uso del algoritmo en cuestión.

En los sistemas para etiquetado construidos a base de reglas, un conjunto de expertos se encargan de definir reglas de transformación que, aplicadas en un orden concreto, son capaces de eliminar las posibles ambigüedades existentes. El algoritmo de Brill trata de generar dicho conjunto de transformaciones automáticamente. Para ello, se parte de un corpus de entrenamiento, sobre el que se aplican un conjunto de transformaciones definidas mediante plantillas por los diseñadores del etiquetador. El sistema entonces escoge aquella transformación que supone un mejor resultado. Este proceso se repite iterativamente hasta que la transformación que se escoge en un paso determinado no supone un cambio suficiente en los datos, o bien cuando ninguna transformación supone una mejora (ver figura 1.6).

Para entender el funcionamiento del algoritmo, se explicará su aplicación al problema del etiquetado morfosintáctico. El corpus de entrenamiento estará formado de un conjunto de frases con las palabras etiquetadas correctamente con su categoría morfosintáctica correspondiente. Sea la siguiente una de las frases del corpus de entrenamiento:

La[DET] historia[NOM] es[VB] una[DET] rama[NOM] de[PREP] la[DET] literatura[NOM].

Primero se contabilizan las posibles etiquetas para cada palabra, y se estima cuál es la más probable. Entonces, el algoritmo parte de las palabras del corpus e ignora las etiquetas correctas, emulando que se está tratando de llevar a cabo el proceso de etiquetado automáticamente. Para ello, se asigna a cada una de las palabras la etiqueta más probable según se ha computado anteriormente. Supongase que el resultado para la frase anterior es el siguiente:

La[PRON] historia[NOM] es[VB] una[DET] rama[NOM] de[PREP] la[PRON] literatura[NOM].

El algoritmo ahora aplicará, una a una, todas las posibles transformaciones definidas, y en cada una de las aplicaciones se estimará cuál es el índice de error, comparando para ello el resultado obtenido tras aplicar cada transformación con las etiquetas correctas. Estos son algunos ejemplos de reglas de transformación permitidas:

1. Cambia la etiqueta actual NOM por VB si la etiqueta anterior es PRON
2. Cambia la etiqueta actual PRON por DET si la etiqueta siguiente es NOM y la palabra actual es *la*

3. Cambia la etiqueta actual VB por NOM si la etiqueta anterior es DET y la etiqueta posterior es ADV

De las tres transformaciones propuestas, la que consigue rebajar más el error es la segunda, ya que con ella se obtiene un resultado correcto para la frase de ejemplo (se sustituirá la etiqueta PRON por DET para los dos artículos *la* que aparecen en la frase). Por tanto, el algoritmo seleccionará dicha transformación y la almacenará como la primera del modelo. El proceso se repetirá una y otra vez, seleccionando cada vez una de las transformaciones que queden disponibles, hasta que se acaben las reglas o en alguno de los pasos no se consiga una mejora apreciable en el error del etiquetado.

Una vez generado el conjunto de reglas que conforman el modelo, el etiquetado consistirá en asignar a cada palabra primeramente la etiqueta más probable observada en el entrenamiento, para posteriormente ir aplicando las transformaciones una a una según están listadas en el modelo.

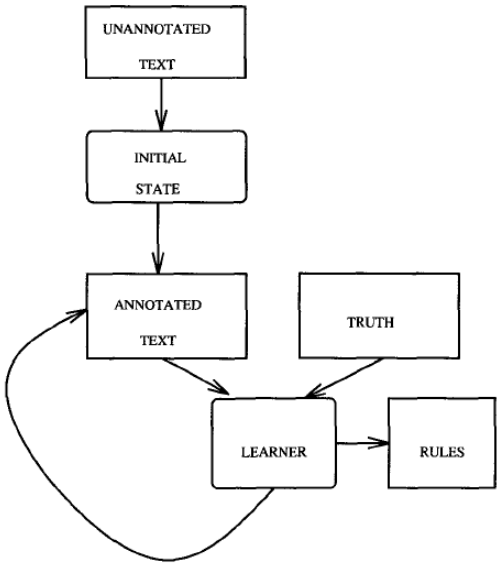


Figura 1.6: Transformation-based learning

En aquellas tareas en las que las ambigüedades se pueden solucionar con una visión local del problema, como el etiquetado morfosintáctico, el algoritmo basado en transformaciones consigue muy buenos resultados en un tiempo muy corto de ejecución. Incluso existen versiones del algoritmo que no necesitan de un corpus etiquetado previamente, funcionando de manera no supervisada. Pero en tareas de etiquetado jerárquico como la que nos ocupa, la cantidad de reglas de transformación a tener en cuenta en cada paso hacen poco práctica la utilización de este algoritmo.

Existe una implementación rápida del etiquetado basado en transformaciones de utilización libre llamada fnTBL, que puede descargarse desde *http://nlp.cs.jhu.edu/ rflorian/fntbl/*.

1.6 Tareas abordadas por el Procesamiento del Lenguaje Natural

De modo muy breve, se proporciona a continuación un listado de las aplicaciones y tareas más habituales que se enmarcan dentro del Procesamiento del Lenguaje Natural, junto a una pequeña descripción de cada una de ellas:

Síntesis de voz: La síntesis de voz consiste en la generación de una señal acústica similar a la voz humana a partir de un texto. En realidad, aunque puede enmarcarse dentro de las tareas del Procesamiento del Lenguaje Natural, lo cierto es que tiene más relación con disciplinas de teoría de señales más propias de ingenierías electrónicas o de telecomunicaciones. La calidad de un sistema de síntesis de voz se mide (siempre de manera cualitativa, puesto que no es posible estimar un valor objetivo de calidad) en función de la inteligibilidad y la naturalidad conseguidas. La inteligibilidad es la propiedad de la señal acústica por la cual resulta fácil a un oyente humano entender el texto dictado en dicha señal. La naturalidad se define en términos de lo humana que resulta la voz sintética producida, esto es, que posea características humanas de calidez, coherencia de la prosodia a lo largo de las frases y en general que no de sensación robótica.

Actualmente, los sistemas de síntesis de voz existentes han resuelto completamente el problema de la inteligibilidad y es en la naturalidad donde aún se intentan conseguir mejoras, aún cuando los avances han sido espectaculares en los últimos tiempos. Es por esto que para muchos la síntesis de voz es un problema resuelto, al contrario que el reconocimiento del habla, que es el proceso inverso.

Reconocimiento del habla: El reconocimiento del habla es el proceso por el cual se genera automáticamente una transcripción en texto a partir de una señal acústica que codifique una voz humana. A partir de este texto y su posterior tratamiento se pueden implementar interfaces controladas mediante la voz, o realizar aplicaciones de dictado, entre otras. Es necesario llevar a cabo complejos procesos de preprocesamiento de la señal acústica, por lo que en general se suele enmarcar el reconocimiento del habla dentro de las disciplinas del tratamiento de señales, al igual que la síntesis de voz.

Se puede distinguir entre reconocedores de palabras aisaladas, cuyos resultados actuales son casi perfectos, y los reconocedores de habla continua, aquellos orientados a ser capaces de transcribir habla natural. Estos últimos aún deben mejorar para llegar a ser perfectos, sobretodo en aquellas aplicaciones en las que no se conoce previamente al locutor (los sistemas con entrenamiento específico al locutor consiguen mejores resultados

de reconocimiento). También se puede distinguir entre el reconocimiento abierto y el basado en gramáticas. En los primeros, se debe poder reconocer cualquier secuencia de palabras que el locutor diga, de entre todas las oraciones que se puedan formar en una lengua concreta. En el segundo caso, el reconocimiento basado en gramáticas sólo reconoce producciones de una gramática determinada, lo que limita mucho el espacio de posibles soluciones y hace que el reconocimiento alcance resultados mucho mejores.

Los reconocedores del habla de hoy día siguen el modelo propuesto por IBM en la década de los 80, y que fue uno de los causantes de la vuelta al enfoque empirista por parte de la comunidad científica que trabajaba en el Procesamiento del Lenguaje Natural. Por un lado, se utiliza un modelo acústico, generalmente implementado en base a una serie de modelos ocultos de Markov, que son entrenados a partir de una conjunto de frases grabadas por locutores humanos y cuyas transcripciones en texto están disponibles. También se dispone de modelos de lenguaje, en los que existen estimaciones bayesianas de la probabilidad de aparición de palabras formando unigramas, bigramas y trigramas. De esta forma, para las n mejores transcripciones proporcionadas por el modelo acústico, el modelo del lenguaje ayuda a decidir cuál de ellas es más probable que sea una frase correcta en el lenguaje en que se esté trabajando.

Generación de lenguaje natural: Consiste en la generación de texto correctamente expresado en lenguaje natural, que exprese determinado mensaje extraído generalmente de una base de conocimiento. La idea es que a partir de un conjunto de unidades de información que una aplicación desea comunicar a un usuario, el generador de lenguaje natural debe ser capaz de generar sentencias correctas y aparentemente humanas para comunicar dicha información. Además del problema enmarcable en la inteligencia artificial de codificación de la base de conocimientos y manejo de la misma, se requieren una serie de sintetizadores o generadores que sean capaces de generar palabras usando correactamente las reglas léxicas del lenguaje, conectar las mismas siguiendo reglas sintácticas y gramaticales correctas, utilicen correctamente la morfología del lenguaje, ... La generación de lenguaje natural es un componente necesario en los sistemas de traducción automática basados en interlingua, como se explicará más adelante.

Traducción automática: la traducción automática trata de reemplazar el trabajo realizado por los intérpretes. Esto es, dado un texto en una lengua, ser capaz de traducirlo automáticamente a otra lengua, conteniendo el mensaje traducido la misma información que el mensaje original y estando correctamente construido según las reglas de la lengua destino. Se realiza una descripción un poco más profunda de la tarea de traducción automática y se discuten sus conexiones con los etiquetadores de roles semánticos en la sección *Aplicaciones del Etiquetado de Roles Semánticos* del capítulo *Etiquetado de Roles Semánticos*.

Respuesta a preguntas: conocido en inglés como *question answering*, el pro-

blema consiste en ser capaz de encontrar un documento de entre un conjunto amplio de documentos en el que se encuentre información que responda a una pregunta concreta efectuada al sistema mediante lenguaje natural. También se puede incluir en la tarea la construcción de la respuesta concreta a la pregunta a partir de la información de dicho documento. Esta tarea, que es una de las más duras del Procesamiento del Lenguaje Natural, ya que requiere de una comprensión casi total del lenguaje, es descrita en más profundidad y relacionada con los etiquetadores de roles semánticos en la sección *Aplicaciones del Etiquetado de Roles Semánticos* del capítulo *Etiquetado de Roles Semánticos*, al igual que las cuatro siguientes, todas las cuales conforman la disciplina de recuperación de información.

Recuperación de documentos: a partir de una gran cantidad de documentos, se trata de encontrar aquellos en los que aparecen determinados términos o que están relacionados con algún tema, aportando además determinados mecanismos para proporcionar una lista de resultados ordenada en función de cierta estimación de la calidad de los documentos o algún otro criterio. El ejemplo más íntuitivo de sistema de recuperación de documentos es el buscador de páginas web, con ejemplos tan famosos como Google o Altavista.

Extracción de información: se trata de encontrar cierta información a partir de la información contenida en documentos de texto (o en un caso concreto, en páginas web), pero a diferencia de la recuperación de documentos, la salida del sistema no será una lista con los documentos relacionados con la información buscada, sino que se generará una base de conocimiento estructurada en base a lo encontrado en los documentos textuales. No se trata por tanto de una tarea puramente sintáctica como puede ser la recuperación de documentos, sino que habrá que llevar a cabo análisis a nivel semántico y en ocasiones pragmático. Un ejemplo de extracción de información sería encontrar automáticamente a partir de la web los nombres y datos personales de los investigadores en el campo del Procesamiento del Lenguaje Natural, generándose una base de datos con la información obtenida.

Clasificación de documentos: los clasificadores de documentos deben escoger a qué categoría pertenecen una serie de documentos, de entre un conjunto de categorías determinadas. Por ejemplo, un clasificador de documentos podría especializarse en determinar la especialidad médica a la que se refieren un conjunto de documentos hospitalarios. Otro ejemplo, en este caso experimentable, es el portal de noticias de la empresa Google (*http://news.google.es*), que además de realizar labores de recuperación de documentos para encontrar automáticamente las noticias del día, las clasifica posteriormente temáticamente para mostrar una configuración automática de las noticias del día de manera similar a como lo hace un periódico convencional (política, sociedad, economía...).

Resumen automático: como cabe esperar por el nombre de la tarea, se trata de resumir automáticamente documentos de texto. Esto se puede abordar de diversas maneras. La más habitual consiste en seleccionar un número de frases del texto intentando que sean lo más representativas posibles del contenido del documento. Otras iniciativas más ambiciosas tratan de escoger trozos de oraciones y concatenarlas en una narración coherente. El resumen de textos automático, si se resuelve de manera adecuada, es un paso previo muy útil para otras tareas de recuperación de información cuando se parte de un volumen muy grande documentos, como es el caso de los sistemas basados en la web.

Analizadores sintácticos: un analizador sintáctico es un programa que toma a la entrada un conjunto de palabras que conforman una oración y devuelve información acerca de las relaciones sintácticas que se establecen entre las palabras. Dicha información será en el mejor de los casos un árbol de derivación que explica las transformaciones gramaticales para llegar desde un símbolo inicial o axioma hasta la oración que se está analizando (ver figura 1.7).

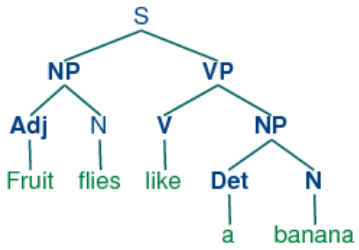


Figura 1.7: Ejemplo del resultado obtenido con un analizador sintáctico completo.

Un analizador sintáctico que genera a la salida un árbol de derivación con todos los constituyentes, gramaticales y sintácticos, y terminando en nodos individuales para cada palabra con la categoría morfosintáctica de las mismas, se denomina analizador completo. Este tipo de analizadores son los que generan una información sintáctica más rica y completa, pero en ocasiones el índice de errores que cometen no es lo suficientemente bajo como para poder utilizarlos. Actualmente, los analizadores sintácticos completos más utilizados alcanzan una tasa de acierto entre el 80% y el 90% para F_1. En determinadas ocasiones, no es necesario conocer todo el árbol de dependencias sintácticas, sino que es suficiente con conocer qué grupos de palabras se agrupan bajo qué sintagmas. Esta información es la proporcionada por los analizadores sintácticos superficiales (*shallow parsers* o *chunkers*). Estos analizadores consiguen una tasa de acierto superior al 90% en todos los casos. Los analizadores sintácticos son una herramienta necesaria en todo sistema de etiquetado de roles semánticos,

ya sea en su versión completa o superficial.

Los dos analizadores sintácticos completos más utilizados actualmente son los de *Collins* [9] y *Charniak* [7]. Ambos analizadores son estadísticos, esto es, para llevar a cabo su tarea se basan en modelos probabilísticos que han sido calculados a partir de corpus de entrenamiento.

Capítulo 2

Etiquetado de Roles Semánticos

2.1 Introducción

En los últimos años, las tareas relacionadas con la comprensión del lenguaje natural han experimentado un avance increíble. Igual que ocurrió en su momento con otras tareas del Procesamiento del Lenguaje Natural, tales como el reconocimiento y la síntesis del habla, el etiquetado morfosintáctico, etc., las tareas relacionadas de un modo u otro con el contenido semántico del lenguaje han empezado a despegar y a propiciar la aparición de aplicaciones como sistemas de diálogo hombre-máquina en lenguaje natural, sistemas de recuperación de información, respuesta a preguntas, resumen y categorización de textos, ... Aunque muchas de estas aplicaciones se encuentran aún en fases muy rudimentarias, las expectativas de futuro para el campo son muy prometedoras, y las implicaciones económicas de fondo lo suficientemente suculentas como para esperar una inversión considerable en este área por parte de empresas privadas y administraciones públicas. Es de destacar en este caso la tarea de la traducción automática, cuya resolución ahorraría a ciertas organizaciones internacionales como la Unión Europea o las Naciones Unidades cantidades inmensas de dinero, razón por la cuál todas ellas invierten actualmente en investigación de la tarea.

Los últimos avances en el análisis semántico del lenguaje conseguidos se deben al igual que en el avance en otras muchas tareas del Procesamiento del Lenguaje Natural al paso de metodologías basadas en la construcción artesanal de sistemas basados en reglas a metodologías conducidas por los datos. Este último enfoque se basa en la aplicación de herramientas estadísticas a grandes bases de datos de conocimiento para la obtención de modelos matemáticos capaces de deducir el comportamiento observado en dichos corpus de conocimiento e inducirlo a nuevas entidades no observadas anteriormente (ver sección *Enfoque estadístico y aprendizaje automático en el Procesamiento del Lenguaje Natural* del capítulo

Introducción al Procesamiento del Lenguaje Natural). Estas herramientas y modelos son similares a los utilizados en disciplinas como la minería de datos y el aprendizaje automático.

La clave para que se pueda llevar a cabo un acercamiento empírico basado en aprendizaje automático a cualquier tarea es la disponibilidad de grandes corpus de textos en lenguaje natural enriquecidos a mano con la información necesaria para abordar la tarea en cuestión. En el caso de las tareas semánticas, el punto de inflexión se ha producido por la aparición de recursos que aportan un conjunto significativo de oraciones en lenguaje natural anotadas con información semántica (primeramente FrameNet [16], y más recientemente PropBank [34]. Ambos serán estudiados en el capítulo *Recursos semánticos*).

Un ejemplo reciente del uso de técnicas estadísticas son los *parsers* sintácticos, que se vinieron abordando durante años mediante sistemas basados en reglas y gramáticas de complejidades cada vez mayores. En los últimos años, la disponibilidad de grandes corpus de textos anotados a mano con información sintáctica ha permitido la construcción de *parsers* sintácticos mucho más robustos, los cuales han tenido gran impacto en el área del procesamiento del lenguaje en los últimos años. Sin embargo, el análisis sintáctico generado por estos sistemas está lejos de ser realmente útil por sí solo en el análisis completo del significado de una frase. Por ejemplo, en dos oraciones tan simples como:

1. Andrés rompió la silla.

2. La silla se rompió.

el análisis sintáctico nos indica que *la silla* es el complemento directo del verbo en la primera oración, y el sujeto en la segunda, pero en ningún momento nos informa de que en ambos casos el sintagma en cuestión está desempeñando el mismo *rol semántico*. Esto es, desde el punto de vista del significado, en ambas oraciones *la silla* es el *el objeto que se rompe*. Esto es sólo un ejemplo simple de la cantidad de casuística distinta que nos podemos encontrar a la hora de tratar de *entender* el significado de una oración basándonos exclusivamente en la información proporcionada por los analizadores sintácticos. En general, un mismo contenido semántico, puede ser expresado sintácticamente con una amplia variedad de oraciones. Mediante el etiquetado de roles semánticos, trataremos de identificar los argumentos de un verbo desde la óptica del papel que representa cada uno, generalizando así las distintas realizaciones sintácticas del mismo contenido semántico y posibilitando el procesamiento del significado de los textos en pos de la construcción de aplicaciones que necesiten procesar semánticamente el lenguaje.

A lo largo de las últimas décadas, los lingüistas se han ocupado de estudiar las relaciones existentes entre la sintaxis del lenguaje y la semántica. En general, es aceptado por todos la existencia de tales relaciones, de manera que para un determinado contenido semántico existen una serie de posibles realizaciones sintácticas, dependiendo del verbo o predicado que utilicemos para expresar el

mensaje. Estos trabajos, que se suelen incluir en un área conocida como *linking theory* (teoría del nexo), son la base de inspiración de los investigadores del Procesamiento del Lenguaje Natural para postular que, a la inversa, analizando la estructura sintáctica de una oración se puede llevar a cabo un etiquetado semántico de la misma. El hecho de que las relaciones existentes entre sintaxis y semántica sean difíciles de concretar hace especialmente apropiado el acercamiento estadístico a la resolución del problema de etiquetado semántico.

El enfoque basado en los roles semánticos es el que actualmente se encuentra en la base de todos los trabajos que intentan construir modelos para la formalización semántica de textos. La razón es la aparición de un recurso llamado FrameNet[16], que basándose en los trabajos de Charles J. Fillmore [15], proporciona un conjunto de oraciones en inglés enmarcadas en distintas clases semánticas, las cuáles representan acciones o relaciones semánticas abstractas entre distintos participantes. Cada uno de los participantes de cada clase semántica desempeña un rol semántico concreto. Por tanto, en el recurso FrameNet, cada una de las oraciones incluidas se encuentra clasificada dentro de alguna clase semántica, y las palabras que conforman la frase se encuentran etiquetadas según el rol semántico que desempeñan. De esta manera, el etiquetado de roles semánticos proporciona un nivel de formalización semántica superficial, que posibilita distintos tratamientos semánticos del texto. FrameNet será descrito con más detalle en el capítulo *Recursos semánticos*. La aparición de este recurso, además de otros que surgen más tarde como PropBank, es aprovechada por diversos investigadores (inicialmente, Gildea y Jurafsky, en [17]) para construir los primeros sistemas de etiquetado de roles semánticos basados en aprendizaje automático, y marca el punto de inflexión clave de los actuales avances en todas las tareas relacionadas con la semántica y el entendimiento del lenguaje natural.

En toda tarea a abordar mediante clasificadores estadísticos, es necesario determinar una serie de características o métricas a extraer del corpus a partir de las cuáles se construye el modelo. Dichas características deben estar relacionadas de alguna forma con la tarea que se pretende abordar, de manera que a partir de la extracción de dichas características de una nueva entidad, el modelo sea capaz de llevar a cabo el etiquetado, la clasificación o la tarea que se intenta resolver. En el caso del etiquetado de roles semánticos, las características utilizadas son de carácter léxico y sintáctico. Los investigadores que iniciaron el camino del etiquetado estadístico de roles semánticos, se basaron en los trabajos lingüísticos del área de la teoría del nexo (linking theory) comentada anteriormente, que es una parte de la gramática que estudia las relaciones entre los roles semánticos y sus realizaciones sintácticas.

Anteriormente a la aparición de los primeros trabajos sobre etiquetado de roles semánticos, se han desarrollado en el campo del los sistemas de *natural language understanding* aplicaciones basadas en clases y roles semánticos pero orientados a un dominio específico, que con la intención de implementar sistemas de diálogo inteligentes etiquetaban los textos semánticamente. Por ejemplo, en

[43] se definen acciones relativas a transacciones de reserva de billetes de avión, en los que los *slots* de información a rellenar son del estilo de ORIG_CITY, DEST_CITY, o DEPART_TIME para implementar un sistema de diálogo. De forma similar, en [19] se utilizan roles como PRODUCTS, RELATIONSHIP, JOINT_VENTURE_COMPANY o AMOUNT en un sistema de extracción de información orientado al estudio de fusiones y adquisiciones empresariales. En este tipo de sistemas, en los que el dominio es conocido, es viable la construcción de sistemas que localicen estos *slots* con un enfoque basado en reglas. En el caso del sistema de compra-venta de billetes de avión, por ejemplo, una serie de expertos lingüístas se encargan de escribir gramáticas capaces de detectar la mayoría de las preguntas y respuestas posibles que un usuario puede introducir en el diálogo, y mediante esas gramáticas extraer la información que el sistema necesita del diálogo. Pero si nos movemos de las tareas específicas para un dominio a la tarea que nos ocupa en el presente trabajo, mediante la que pretendemos realizar un etiquetado semántico similar a los realizados en los trabajos anteriormente citados pero aplicado a cualquier texto en lenguaje natural de cualquier procedencia, se hace virtualmente imposible construir un sistema basado en reglas capaz de llevar a cabo la tarea. El único enfoque posible será el estadístico, como se verá en el capítulo siguiente cuando se describa la arquitectura de los etiquetadores de roles semánticos actuales.

2.2 Descripción de la tarea

El etiquetado de roles semánticos se puede describir como la realización de los siguientes pasos a llevar a cabo para cada una de las proposiciones y oraciones a etiquetar:

- Identificar cuál es la clase semántica a la que pertenece la oración. El verbo suele ser el que informa sobre esto, aunque en ocasiones puede ser un predicado de otro tipo, como un nombre o un adjetivo. La correspondencia entre la clase semántica y el predicado no tiene por qué ser de directa, ya que un mismo predicado puede evocar distintas clases semánticas. Por tanto, esta parte del etiquetado de roles semánticos puede verse como un problema de desambiguación de significados.

- Una vez decidida la clase semántica en la que nos encontramos, hay que detectar los roles semánticos participantes en la misma, de entre los constituyentes de la oración, y etiquetarlos adecuadamente. Hay que tener en cuenta que para una misma clase semántica pueden aparecer en distintas oraciones un número distintos de roles semánticos.

Dependiendo del recurso en el que basemos nuestro etiquetador de roles semánticos, ciertas características de la tarea varían, haciendo variar con ello la dificultad de la misma. Básicamente, existen trabajos basados en la filosofía propuesta por FrameNet y otros basados en PropBank (ver capítulo *Recursos semánticos*). En FrameNet se dispone de una taxonomía bien jerarquizada de

clases semánticas, y los roles semánticos de cada una de estas clases son específicos para cada una de ellas y poseen nombres que hacen referencia a la acción o situación expresada por la clase. En PropBank sin embargo se prescinde de realizar ninguna taxonomía de clases semánticas, conformándose con las distintas acepciones de los verbos propuestas en VerbNet. Además los roles semánticos son independientes de la clase semántica. En general, y como será más ampliamente discutido en el capítulo dedicado a los recursos semánticos, la propuesta de PropBank es más fácil de llevar a cabo que la de FrameNet, aunque también proporciona un análisis semántico más pobre.

Los distintos investigadores con trabajos en etiquetado de roles semánticos divergen también en la amplitud con la que consideran la tarea. Algunos de ellos, entienden que el problema de la identificación de la clase semántica es un problema de desambiguación de significados que debe ser entendido como un paso previo y no incluido en el problema, mientras que otros si lo incluyen.

Los argumentos o roles semánticos cumplen en todo momento dos propiedades que es necesario tener en cuenta a la hora de plantear los sistemas. En primer lugar, para una clase semántica dada, los roles no se solapan unos con otros, apareciendo secuencialmente, y sin tener que cubrir todos los componentes de la proposición. En segundo lugar, un rol puede aparecer divido en una serie de fragmentos no contiguos. Estas propiedades caracterizan a la tarea de etiquetado de roles semánticos como una tarea de etiquetado jerárquico, y no secuencial, según lo explicado en la sección *Tareas de etiquetado* del primer capítulo del presente trabajo, con las dificultades que ello conlleva expuestas en dicho capítulo.

El etiquetado de roles semánticos tiene algunas peculiaridades que la hacen ser más difícil que otras tareas de etiquetado, como el morfosintáctico o el reconocimiento de entidades. A continuación se citan algunas, aquellas que son entendibles en este punto de la exposición (en capítulos posteriores se plantearán más dificultades a medida que profundicemos en la arquitectura de los etiquetadores de roles semánticos) :

1. No siempre es deducible a partir de las estructuras sintácticas las relaciones semánticas de los constituyentes, ya que en ocasiones los participantes humanos en un diálogo se apoyan en su conocimiento previo del mundo para interpretar correctamente el mensaje. Por tanto, es de esperar la utilización de recursos semánticos y bases de conocimiento en futuros sistemas "inteligentes" de etiquetado semántico, aunque hoy por hoy los sistemas existentes ignoran estos detalles y se centran en las relaciones entre sintaxis y semánticas.

2. Hay palabras que participan en una gran cantidad de roles distintos, lo que supone una gran ambigüedad. Algunos roles genéricos pueden estar instanciados por cualquier palabra. Este tipo de fenómenos tienden a

inutilizar cualquier acercamiento léxico al etiquetado de roles semánticos, y reafirman la necesidad de partir de un análisis sintáctico.

3. La estructura interna de un sintagma no siempre es un buen estimador del rol semántico que desempeña. Por ejemplo, "in the hole" puede funcionar como rol LOCATION (según nomemclatura FrameNet) en la frase *she sat in the hole* o puede funcionar como GOAL en una clase semántica MOVEMENT en la frase *She jumped in the hole*.

4. Los analizadores sintácticos necesarios para la construcción de los sistemas de etiquetado semántico cometen fallos habitualmente, lo que viene a dificultar la ya de por si compleja tarea de descubrir las relaciones entre sintaxis y semántica.

2.3 Aplicaciones del Etiquetado de Roles Semánticos

2.3.1 Traducción automática

La traducción automática de texto o de habla es una de las disciplinas más clásicas dentro de la lingüística informática, y uno de los problemas que más resistencia están ofreciendo a ser resueltos. Se puede definir como la implementación de sistemas que sean capaces de traducir de forma automática textos o habla de una lengua a otra lengua cualquiera. Entiéndase dicha traducción de manera que el texto o habla de salida sea completamente correcto en términos de forma según la lengua destino y recoja lo más fielmente posible el contenido semántico del mensaje original.

Como puede intuirse, la dificultad de la tarea es altamente elevada. No basta con realizar una traducción literal entre palabras o conjuntos de palabras de un lenguaje a otro. Las dependencias entre los constituyentes de una frase, las distintas construcciones gramaticales y sus relaciones con la semántica, fenómenos de pragmática, las diferencias morfológicas,estructurales y gramaticales entre lenguas de orígenes lejanos, como el castellano y el ruso, ..., son sólo algunos de los motivos que hacen de esta tarea un problema tremendamente complejo. Hoy por hoy, los resultados obtenidos por los sistemas de traducción automática no son directamente utilizables, aunque sirven de apoyo en la tarea de traducción a profesionales humanos. Al mismo tiempo, existe un gran interés en la resolución del problema por parte de diversos sectores de la sociedad, dado que ello eliminaría un cuello de botella fundamental para una sociedad cada vez más globalizada y dependiente de las comunicaciones entre las personas, ya sea en política, cultura, economía, ...

Existen distintos enfoques hacia la tarea. En primer lugar, se puede distinguir entre sistemas basados en reglas y sistemas estocásticos. Estos últimos, introducidos por IBM en la década de los setenta, marcaron un punto de inflexión en la calidad de los resultados obtenidos, y en la disminución del coste

de fabricación de sistemas para nuevos pares de lenguas. Por otro lado, existen sistemas basados en interlingua o en transferencia (ver figura 2.1, extraído de [48]). En los sistemas basados en interlingua se parte de la premisa teórica de que para traducir un texto hay que *comprenderlo*. A partir del mensaje original, se lleva a cabo un trabajo de análisis del contenido semántico, para generar una representación del mismo expresado en un lenguaje conceptual intermedio conocido como *interlingua*. Para cada lengua de origen que se quiera considerar, se generará un analizador distinto, generando todos ellos salidas en interlingua. Posteriormente, para cada lenguaje de destino a considerar, se llevan a cabo *generadores* que a partir de un mensaje expresado en interlingua construyen las estructuras sintácticas y gramaticales y las unidades léxicas necesarias para expresar en la lengua de destino el mensaje en cuestión.

Los sistemas basados en transferencia, por su parte, realizan un análisis más superficial, pudiendo quedarse a nivel de información léxica, sintáctica o semántica. A partir de ese nivel de análisis, se lleva a cabo una transferencia, generalmente estocástica. Por ejemplo, si el análisis se llevase a cabo a nivel sintáctico, el árbol sintáctico obtenido debería ser transformado en un árbol sintáctico de la lengua destino. Posteriormente también será necesario un proceso de generación para llegar al mensaje de salida. Si la transferencia se realiza a nivel semántico, el sistema se encuentra muy próximo realmente al enfoque basado en interlingua.

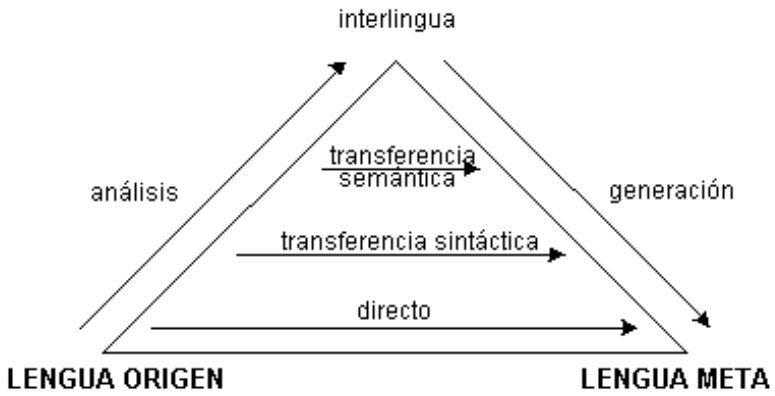


Figura 2.1: Gráfico piramidal de transferencia e interlingua

Es en estos sistemas de transferencia a nivel semántico donde la utilización de etiquetadores de roles semánticos resultan prometedores. La estructura en forma de predicados semánticos instanciados por argumentos o roles semánticos se perfila como una excelente representación semántica del contenido de un mensaje, y hasta cierto punto independiente de la lengua (siempre que se garanticen ciertas características básicas como que se trate de lenguas predicativas, lo cuál ocurre para la gran mayoría de lenguas modernas). Esta representación a nivel semántico independiente de los distintos fenómenos sintácticos, gramaticales y

léxicos que vienen a dificultar la tarea de traducción sería un punto excelente para llevar a cabo la transferencia. En la mayoría de los casos, tal como se ha dicho, siempre que las lenguas compartan un mínimo de características básicas, dicha transferencia tendrá que realizar mínimas transformaciones en la representación basada en roles semánticos. Los generadores de una representación sintáctica a partir de la representación semántica tampoco representan a priori un gran problema. Por todo esto, es de esperar un notable avance en el estado del arte de la traducción automática una vez que se consigan etiquetadores de roles semánticos con un bajo índice de fallos para las distintas lenguas.

2.3.2 Desambiguación de significados

La desambiguación de significados es el proceso por el cual dado un texto cualquiera se decide para cada palabra polisémica del mismo cuál es el sentido correcto de ésta, de entre los posibles sentidos recogidos en un diccionario semántico, o más frecuentemente, en un diccionario léxico (p.e. WordNet [33]). Para abordar la tarea, es común utilizar el contexto de la palabra para decidir probabilísticamente cuál es el sentido de la misma.

Desambiguar las palabras de un texto es un paso previo necesario para muchas tareas del Procesamiento del Lenguaje Natural, como la traducción automática, la recuperación de información, los sistemas de preguntas y respuestas, e incluso los propios etiquetadores de roles semánticos. Tal como se verá en el apartado de arquitectura del presente informe, el primer paso a llevar a cabo por un etiquetador de roles semánticos será decidir para cada predicado de la frase a etiquetar cuál es el sentido o acepción con que está funcionando el verbo (permítase la licencia de considerar sólo el verbo como posible núcleo semántico, aunque como se verá es también posible en determinados recursos semánticos la aparición de nombres y adjetivos desempeñando esta función). Por tanto, la desambiguación de significados ocupa una parte importante en la arquitectura de todo etiquetador de roles semánticos.

Pero además, al mismo tiempo, la utilización de etiquetadores semánticos para determinar los roles que ocupan las distintas palabras de un texto es considerada por diversos autores como una ayuda inestimable en la resolución del problema de la desambiguación de significados. Es de esperar que el análisis estadístico de un corpus de texto de entrenamiento(con todas las palabras desambiguadas, y en el caso que nos ocupa etiquetado semánticamente) descubriera determinadas correlaciones entre ciertas acepciones de una palabra y la aparición de la misma como algún rol concreto o roles de un marco o marcos semánticos determinados.

Esta doble vertiente de la desambiguación de significados, como paso previo al etiquetado de roles semánticos, y al mismo tiempo como tarea que se beneficia de la utilización de un etiquetador semántico, justificaría el estudio de técnicas de *bootstrapping* entre ambos sistemas que ocasionalmente mejoren la precisión alcanzada por ambos.

2.3.3 Recuperación de información

Estando disponibles en la actualidad grandes cantidades de información plasmada en texto en lenguaje natural, en formato electrónico, un problema habitual consiste en localizar la información que nos interesa. Para abordar esta tarea, se desarrollan distintas disciplinas dentro del Procesamiento del Lenguaje Natural, entre las que se enmarca la recuperación de documentos, la extracción de información, los sistemas de preguntas y respuestas, la clasificación de textos o el resumen automático de textos.

Mediante todas estas técnicas se pretende posibilitar la utilización automática de la inmensa cantidad de información contenida en documentos de texto en lenguaje natural. Se trata de llegar a la estructuración y comprensión algorítmica de dicha información desestructurada y ambigua, mediante la aplicación de técnicas de Procesamiento del Lenguaje Natural. Otro enfoque distinto para posibilitar el consumo computacional de la información de la web consiste en dotar de una estructura lógica a los nuevos documentos que vayan apareciendo, mediante la construcción de ontologías que faciliten la tarea a los programas que traten de acceder a la información contenida en los mismos. Pero la responsabilidad de construir estas estructuras recaería en aquellas personas que crean nuevos documentos en la web. En el enfoque de la recuperación de información, se libera de esa responsabilidad a los participantes humanos, y se trata de adaptar, formalizar, estructurar, ..., *automáticamente* toda la información contenida en los millones de documentos con contenido en lenguaje natural disponibles actualmente.

La construcción de sistemas que etiqueten semánticamente textos en lenguaje natural está íntimamente relacionada con las probabilidades de éxito en las tareas involucradas en la recuperación de información.

Recuperación de documentos

Dado un conjunto grande de documentos con textos en lenguaje natural, la recuperación de documentos trata de localizar aquellos documentos cuyo contenido está relacionado con una temática o un conjunto de términos en concreto. Un ejemplo de sistema de recuperación de documentos aplicado en el entorno de la web son los buscadores web, tales como Google o Altavista. Estos son ejemplos de sistemas de recuperación de documentos que podemos denominar como léxicos o sintácticos, ya que buscan ocurrencias de los términos de la búsqueda en los documentos de sus bases de datos, y devuelven listados de dichos documentos ordenados según distintos criterios de calidad o relevancia de los mismos. Pero no todos los sistemas de recuperación de documentos deben quedarse en el nivel sintáctico. Un sistema de recuperación de documentos puede acceder a la información semántica de los documentos, para de esta forma devolver un conjunto mayor de documentos relacionados semánticamente con los términos de la búsqueda. Por ejemplo, ante una búsqueda con el término *altercados*, un sistema de este tipo podría devolver un documento en el que se hablara de que *la policía detuvo a los manifestantes que habían comenzado a lanzar pie-*

dras contra el dispositivo.... De todas formas, el análisis semántico de los textos para encontrar los documentos se suele englobar en los sistemas de preguntas y respuestas más que en los de recuperación de documentos.

El uso de los etiquetadores de roles semánticos puede ayudar a implementar sistemas de recuperación de documentos que tengan en cuenta la semántica de los textos sobre los que realizan la búsqueda.

Extracción de información

La extracción de información intenta encontrar unidades básicas estructuradas de información relativa a algo en concreto, a partir de la información desestructurada contenida en documentos de texto como pueden ser las páginas web. Por ejemplo, una posible tarea de extracción de información sería localizar nombres de congresos científicos en la web, junto a las fechas de celebración.

Generalmente, se define una base de datos que después hay que poblar de manera automática a partid de los documentos de texto. Una subtarea que podría considerarse paso previo a la extracción de información sería el reconocimiento de entidades, que consiste en detectar conjuntos de palabras que hacen referencia a nombres de personas, organizaciones, lugares, ... La tarea de extracción de información es más complicada que esto, ya que trata de descubrir ciertas relaciones entre entidades, que implican un análisis semántico de las oraciones. Es de esta necesidad de análisis semántico de las frases de donde se deriva la utilidad de los etiquetadores de roles semánticos para la mejora de los sistemas actuales de extracción de información. Otro problema a resolver para llevar a cabo la extracción de información es la resolución de correferencias, que consiste en detectar cuando varios conjuntos de palabras se refieren a un mismo objeto. En este terreno también representan una ayuda considerable los etiquetadores de roles semánticos, ya que el rol que desempeñen distintos constituyentes de varias oraciones puede apuntar a que se traten de un mismo objeto.

Las posibilidades de estos sistemas son inmensas y sus implicaciones comerciales también, por lo que existen muchos recursos dedicados a la investigación en este área.

Clasificación de documentos

La clasificación de documentos consiste en decidir a partir de un documento de texto a qué categoría temática pertenece de entre un conjunto de categorías posibles. Por ejemplo, un sistema clasificador de textos periodísticos podría asignar automáticamente a un artículo dado la sección del periódico en la que debería aparecer (política, economía, sociedad, deportes,...).

En los primeros sistemas, una serie de expertos construían reglas manualmente que identificaran patrones propios de cada una de las categorías a considerar. Posteriormente, y de forma análoga a como ha venido pasando en otras tareas de la minería de textos, se ha pasado a sistemas estocásticos que utilizan aprendizaje automático para, a partir de un conjunto inicial de textos

previamente clasificados, construir modelos probabilísticos capaces de predecir la categoría en la que se enmarca un nuevo texto dado. Estos sistemas consiguen un rendimiento similar a los construídos manualmente, con un consumo sensiblemente menor de recursos para la construcción de nuevos sistemas.

Los sistemas de clasificación de textos actuales extraen estadísticas de ocurrencias de palabras, o en algunos casos de entidades (basándose para ello en el trabajo previo de un reconocedor de entidades). La utilización de etiquetadores de roles semánticos sería una ayuda considerable en la mejora de los sistemas de reconocimiento de entidades e indirectamente en los clasificadores de documentos.

Resumen automático de textos

Dado un documento o conjunto de documentos de texto, se trata de realizar un resumen de una determinada extensión, tratando de capturar en el mismo la información esencial contenida en los documentos originales.

Por supuesto, dado un texto original no existe un único resumen posible, ya que entre otras cosas decidir cuál es la información "esencial" es una labor ambigua y cargada de subjetividad. Esto hace difícil la evaluación de los sistemas. Existen enfoques simples que tratan de seleccionar las frases que condensan mejor el contenido del texto. Otros acercamientos más complejos tratan de extraer las líneas principales de información y los datos independientemente de las frases. En general, todos los sistemas actuales hacen uso de técnicas estadísticas y de minería de textos para llevar a cabo la labor, salvo algunas excepciones, como una propuesta asombrosamente simple basada en el algoritmo de *TextRank* descrita en [39]. Algunos sistemas además utilizan WordNet como recurso de apoyo. Los sistemas más simples utilizan sólo información a nivel léxico y sintáctico. Los más avanzados, tratan de utilizar de alguna forma la información semántica, justificándose en el hecho de que un operador humano que resume un texto lo hace abstrayendo el contenido semántico del mensaje, seleccionando del mismo las partes fundamentales y plasmándolo posteriormente en un texto.

En este nivel semántico, al igual que ocurría en la traducción automática, la representación basada en roles semánticos puede ser de gran utilidad.

Respuesta a Preguntas

Así como los sistemas de recuperación de documentos servían para realizar búsquedas de información en un conjunto de documentos a partir de una serie de términos, y generalmente se llevaba a cabo la tarea buscando ocurrencias de dichos términos, los sistemas de respuestas a preguntas van un paso más allá y tratan de encontrar directamente la respuesta a una pregunta formulada en lenguaje natural por el usuario. La respuesta puede consistir en un documento del conjunto de búsqueda que contenga información referida a la pregunta introducida en el sistema, o incluso un trozo de texto, literal o modificado, de alguno de los documentos, donde se responda explícitamente a la pregunta planteada.

Así mismo, las preguntas pueden ser escogidas de entre un conjunto cerrado de combinaciones, o bien ofrecer al usuario la libertad de plantear preguntas en lenguaje natural de la manera que estime oportuno.

Siendo un problema totalmente inmerso en la semántica del lenguaje, y para algunos tan relacionado con la Inteligencia Artificial como con el Procesamiento del Lenguaje Natural, los sistemas de respuestas a preguntas están todavía en un estado primitivo de desarrollo. Aún así, se tienen depositadas grandes esperanzas en ellos como sustitutos de los buscadores sintácticos actuales.

Se trata del problema más complejo y ambicioso de los que se engloban en la temática de recuperación de información, y como tal, hace uso del resto de disciplinas que conforman a la misma. Por ejemplo, la utilización previa de sistemas de extracción de información que pueblen una base de datos con cierto tipo de información facilitará la resolución de preguntas relacionadas con dicha información; o la clasificación de textos, que reduce el espacio de búsqueda en el que encontrar las respuestas, una vez se identifica la temática de una pregunta. El uso de todas estas herramientas, que a su vez se beneficiarían de la existencia de etiquetadores de roles semánticos lo suficientemente eficaces, convierte a los sistemas de preguntas y respuestas en otra de las posibles aplicaciones prácticas de estos etiquetadores.

Además, existe una relación directa y evidente entre las partículas interrogativas utilizadas en el planteamiento de las preguntas, y los roles semánticos esperados de los constituyentes que pueden responder a dichas preguntas. Por ejemplo, antes la pregunta *¿Quién envió la carta?*, si tenemos un documento donde aparece la frase *La carta del banco fue mandada por el director del mismo a Luis*, la partícula interrogativa *quien* me está informando de que el constituyente que buscamos es aquel que desempeña el rol *agente* en un marco semántico *enviar*. Por tanto, poder etiquetar semánticamente los documentos que constituyen la base de conocimiento haría más fácil la implementación de sistemas de preguntas y respuestas, existiendo por supuesto a pesar de ello muchos otros problemas a solucionar (por ejemplo, *el director del mismo* contiene una correferencia que habría que resolver para poder contestar correctamente que fue el director del banco quien envió la carta).

2.3.4 Modelos del lenguaje enriquecidos semánticamente

Los modelos del lenguaje son modelos estadísticos que nos informan de la probabilidad de la ocurrencia de una sucesión de palabras en un lenguaje determinado. Esto se hace considerando las oraciones de un lenguaje simplemente como una sucesión de elementos sin estructura sintáctica o semántica alguna.

Para calcular los modelos, se parte de un corpus de textos en el lenguaje que nos interese. Simplemente se llevará a cabo un conteo de las apariciones de las palabras, así como de las apariciones de bigramas y trigramas, esto es, cuántas veces aparece cada palabra precedida por determinada palabra, o por determinado par de palabras. A partir de estos conteos se estima la probabilidad de aparición de cada palabra, bigrama y trigrama. Una vez hecho esto, para calcular la probabilidad de ocurrencia de una oración en concreto, se calcula la

probabilidad combinada de la aparición de cada una de las palabras, bigramas y trigramas para la oración. El problema principal es la imposibilidad de encontrar suficientes ejemplos de todas las posibles palabras en todas las posibles combinaciones en que pueden aparecer. Para solucionar esto, en ocasiones se realizan los modelos del lenguaje basándose en las categorías morfosintácticas en lugar de en las palabras directamente. De todas formas, los modelos del lenguaje basados en palabras se pueden generar a partir de la inmensa cantidad de textos disponibles en internet. Por ejemplo, Google ha puesto a disposición de la comunidad científica un modelo de lenguaje calculado sobre un total de 1,024,908,267,229 palabras.

Los modelos del lenguaje se utilizan en aquellas tareas del Procesamiento del Lenguaje Natural donde se necesita realizar una estimación de la probabilidad de que determinada secuencia de palabras constituyan una oración adecuada en determinado lenguaje. Dos ejemplos clásicos son el reconocimiento del habla y la traducción automática. En el primer caso, tal como se explicó en una sección anterior, un modelo acústico genera una serie de posibles oraciones reconocidas. Para cada una de ellas, se utiliza un modelo del lenguaje para seleccionar aquella oración más probable en el lenguaje utilizado. Algo similar se hace en la traducción automática con las posibles oraciones traducidas.

En todas estos casos, sería útil la construcción de modelos del lenguaje que podríamos etiquetar como enriquecidos semánticamente. La idea es llevar a cabo la construcción del modelo de apariciones de palabras, bigramas y trigramas, pero en lugar de realizar los conteos para palabras, llevarlo a cabo con los roles semánticos que componen la oración. De esta forma, el modelo del lenguaje nos daría la probabilidad de que determinados roles semánticos aparezcan en determinada secuencia formando una oración en cierto lenguaje. Esta probabilidad podría ser utilizada entonces de manera similar a la obtenida con los modelos del lenguaje basados en palabras, posiblemente en combinación con esta misma, para mejorar aquellas tareas en las que de forma clásica se vienen usando modelos del lenguaje.

2.3.5 Sistemas de diálogo

Los sistemas de diálogo en lenguaje natural son actualmente una de las utilidades relacionadas con el Procesamiento del Lenguaje Natural en cuya realización más recursos se están invirtiendo actualmente. La intención es permitir a los usuarios acceder a sistemas de información o de cualquier otro tipo a través de una interfaz conversacional. Un ejemplo tipo sería una aplicación para realizar reservas de vuelos a través de teléfono, de manera que el usuario lleve a cabo la tarea conversando con la aplicación.

Los sistemas de diálogo se suelen componer de cinco módulos, estos son:

- Módulo de Reconocimiento Automático del Habla
- Módulo de Comprensión

- Módulo de Gestión del Diálogo
- Módulo de Generación de Respuestas
- Módulo de Síntesis de Voz

Una descripción en profundidad de cada uno de los módulos puede ser encontrada en [51], [27] y [41].

La utilización de etiquetadores de roles semánticos facilita al menos los dos primero módulos. En el módulo de Reconocimiento del Habla, a través de los modelos del lenguaje enriquecidos semánticamente. En el caso de la comprensión, es patente que disponer de la representación basada en clases y roles semánticos de las oraciones reconocidas es de gran ayuda. La experimentación de esta integración entre los etiquetadores de roles semánticos y los sistemas de diálogo es una de las posibles líneas de trabajo futuro propuestas al final del presente informe.

Capítulo 3

Recursos Semánticos

3.1 Introducción

Este capítulo se centrará principalmente en realizar una descripción de dos recursos lingüísticos muy relacionados con la tarea del etiquetado de roles semánticos. El primero de ellos, FrameNet [16], sirvió de base para el primer gran trabajo sobre etiquetado automático de roles semánticos [17], precursor de los posteriores sistemas y del actual interés de la comunidad del Procesamiento del Lenguaje Natural en esta problemática. El segundo de ellos, PropBank [34], es responsable de la sensible mejora en los resultados de los sistemas actuales, principalmente debido a su clara vocación de corpus enfocado a la construcción de sistemas de etiquetado de roles semánticos.

Las descripciones que se realizarán de estos recursos no pretenden ser exhaustivas, sino meramente introductorias. Se resaltarán aquellos detalles más relevantes con respecto al foco de este trabajo, y se buscarán puntos de coincidencia y divergencia entre ambos recursos. Además, también serán introducidos otros recursos semánticos que, si bien no están enfocados directamente a su utilización en el marco del etiquetado de roles semánticos, sí pueden ser útiles en la concepción de posibles mejoras a los sistemas actuales.

3.2 FrameNet

FrameNet [16] es un proyecto que pretende identificar y describir los aspectos lexicográficos de las palabras de un gran corpus de texto en inglés (esencialmente extraído del British National Corpus [5]), tratando de reflejar con ello la relación entre las propiedades sintácticas y semánticas existentes en el idioma. El proyecto FrameNet contiene, entre otras cosas, un conjunto de oraciones que intentan abarcar exhaustivamente toda la casuística que se da en el inglés en relación a las realizaciones sintácticas de todos los posibles contenidos semánticos, proporcionando para dichas frases un etiquetado parecido al que hemos descrito como la tarea de etiquetado de roles semánticos. De hecho, la teoría de los roles

semánticos, cuyo desarrollo se remonta a la segunda mitad de la década de los 60, es precursora de las ideas que sirven de base al proyecto FrameNet, que se basa en una evolución de esta teoría conocida como *semántica basada en marcos* [8].

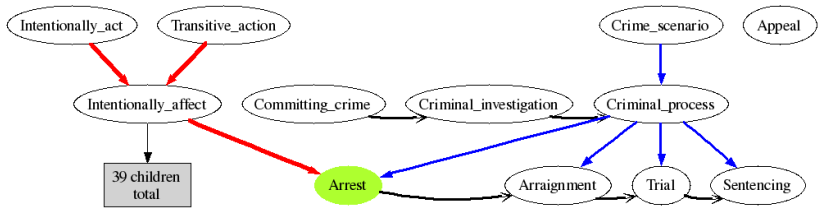


Figura 3.1: Un ejemplo de las relaciones entre marcos semánticos en FrameNet

El nombre de FrameNet refleja precisamente la relación con esta teoría, así como con el hecho de que se establecen relaciones de herencia y composición entre estos marcos semánticos, formándose las *redes de significado* en las que participan las palabras (figura 3.1). Aunque esta idea puede parecer a priori parecida a la implementada en WordNet (ver sección 3.5.1), la idea central de la semántica de marcos es que los significados de una palabra tienen que ser descritos en términos de su relación con los marcos semánticos, que son representaciones esquematizadas de las estructuras conceptuales y patrones sintácticos en los que se manifiestan los conocimientos, prácticas, instituciones, acciones, imágenes y en general todos los distintos contenidos semánticos que se pueden expresar en el idioma. Como se verá, estos marcos semánticos vienen a ser lo que en la descripción de la teoría de roles semánticos se conoce como clases semánticas, y los elementos que conforman el marco semántico serán en cierto modo equivalentes a los roles semánticos . En realidad, los conceptos de marco semántico y elementos de un marco considerados en la teoría de marcos semánticos y por ende en FrameNet son más ambiciosos y tienen mayor complejidad conceptual que los conceptos correspondientes de la teoría de roles semánticos. En FrameNet se identifican y describen los posibles marcos semánticos existentes en el inglés, y se analizan los significados de las palabras directamente refiriéndose al marco semántico en el que aparecen, estudiando las propiedades sintácticas de las palabras y cómo las propiedades semánticas se plasman en una realización sintáctica concreta.

Un conjunto determinado de palabras, que pueden constituir una proposición o ser simplemente un sintagma de algún tipo, estarán enmarcadas en términos semánticos en un marco o *frame* concreto. El significado particular de alguna de las palabras participantes es el que determina cuál es el marco semántico correcto. Este par formado por una palabra y un significado concreto para la misma se conoce en FrameNet como unidad léxica (*lexical unit*), y se dice entonces que una unidad léxica *evoca* un marco semántico. Así es como toma forma la idea base de la semántica basada en marcos, según la cuál el significado

de las palabras debe ser explicado en términos del marco semántico en el que se enmarcan.

Además de los conceptos de marco semántico y unidad léxica, es necesario definir también el concepto de elemento de un marco o *frame element* antes de citar algunos ejemplos que resultaran muy clarificadores. Los elementos de un marco o *frame elements* son los distintos tipos de entidades que participan en un marco semántico determinado. En los términos empleados en el contexto del etiquetado de roles semánticos, un *frame element* viene a ser un rol semántico para una clase semántica determinada. Los elementos que participan en un marco semántico concreto son específicos de dicho marco, por lo que existen multitud de elementos distintos, siendo ésta una diferencia fundamental con la visión más generalista utilizada en la mayoría de los etiquetadores de los roles semánticos y en otros recursos linguísticos como PropBank, en la que se utilizan un conjunto de roles más reducido y compartido por las distintas clases semánticas. Puede hacerse una analogía entre los elementos de un marco y los argumentos de un predicado de lógica de primer orden, o simplemente con los argumentos de algún tipo de función. Así pues, dada una secuencia de palabras que evocan un marco semántico determinado, hay que decidir cuáles de esas palabras instancian cada uno de los elementos requeridos por el marco semántico.

Se presenta a continuación un ejemplo de marco semántico y los elementos que participan en el mismo:

Frame : *Transfer*

Frame Elements : *DONOR, THEME, RECIPIENT*

Descripción : Alguien (*DONOR*) está en posesión de algo (*THEME*) y entonces hace que alguien más (*RECIPIENT*) esté en posesión del *THEME*, quizás ocasionando que el *THEME* se mueva al *RECIPIENT*.

Los nombres que se utilizan para identificar los elementos del marco no deben entenderse literalmente. Por ejemplo, *DONOR* no significa necesariamente "donante", como indica la definición de la palabra, sino que debe entenderse en los términos expuestos en la descripción del marco semántico. Los nombres utilizados cumplen simplemente un objetivo mnemónico. Veamos ahora dos realizaciones sintácticas del marco semántico anterior:

1. The teacher *gave* the student a book.

2. The teacher *gave* a book to the student.

Según la filosofía de FrameNet, el significado de los constituyentes de la oración debe ser entendido en términos de los roles semánticos y gramaticales que desempeñan con respecto al verbo *give*. El verbo es en este caso la palabra

give	FEs:	Donor	Theme	Recipient
	PTs:	NP	NP	NP
	GFs:	Ext	Comp	Obj
give	FEs:	Donor	Theme	Recipient
	PTs:	NP	NP	PP-to
	GFs:	Ext	Obj	Comp

Tabla 3.1: Patrones de valencia para el verbo *give* en FrameNet. Para cada combinación de *frame elements*, se expresan las funciones sintácticas y gramaticales de las posibles realizaciones sintácticas de cada *frame element*.

que evoca el marco semántico *Transfer*, siendo la palabra que juega ese papel conocida como *target* en la terminología usada por FrameNet (se puede traducir por objetivo o, utilizando la terminología utilizada generalmente en los etiquetadores de roles semánticos y en recursos como PropBank, predicado). Los roles semánticos que participan en la oración serán los elementos del marco.

En FrameNet, al requisito por el cual una palabra debe combinarse con tipos particulares de sintagmas en una oración se le conoce como la valencia de una palabra (*valence*), por analogía con el término utilizado en química para referirse a las posibilidades de combinación de los átomos. La valencia puede entenderse en términos sintácticos y semánticos.

La valencia semántica vendría de una palabra especificada por los elementos del marco que evoca la palabra. Por ejemplo, la palabra *give* en el marco semántico *Transfer* debe ir acompañada de los elementos *DONOR, THEME,* y *RECIPIENT*. Para describir la valencia semántica de la palabra en toda su extensión será necesario por tanto especificar todos los marcos semánticos que puede evocar, y qué conjuntos de elementos deben acompañarla en cada caso.

Por otro lado, la valencia sintáctica de una palabra debe expresarse en términos de cuáles son las funciones sintácticas y gramaticales de los elementos semánticos de cada uno de los marcos semánticos asociados a la palabra. En el ejemplo anterior, las dos oraciones que aparecen reflejan parte de las propiedades de valencia sintáctica de la palabra *give* en el marco semántico *Transfer*. En ambos casos, el rol *DONOR* está formado por un sintagma nominal (*the teacher*), y el rol *THEME* está expresado por otro sintagma nominal (*a book*). El tercer elemento semántico en discordia, *RECIPIENT* está constituido en la primera oración por un sintagma nominal (*the student*) y en la segunda por un sintagma preposicional (*to the student*). Además, gramaticalmente, *the teacher* es el sujeto de *gave* en ambas oraciones (en FrameNet, el sujeto es denominado argumento externo, o de forma abreviada *Ext*). En la primera oración, *the student* es el copmlemento directo (*Obj*) del verbo, y *a book* funciona como complemento indirecto (*Comp*). En la segunda, sin embargo, *a book* funciona como complemento directo y *to the student* como indirecto. Toda esta información que caracteriza las propiedades de valencia de un predicado se encuentran anotadas en FrameNet mediante "patrones de valencia" (ver tabla 3.1).

El proyecto FrameNet incluye la creación de una serie de herramientas que facilitan las tareas de búsqueda de ejemplos y etiquetado (ver figura 3.2).

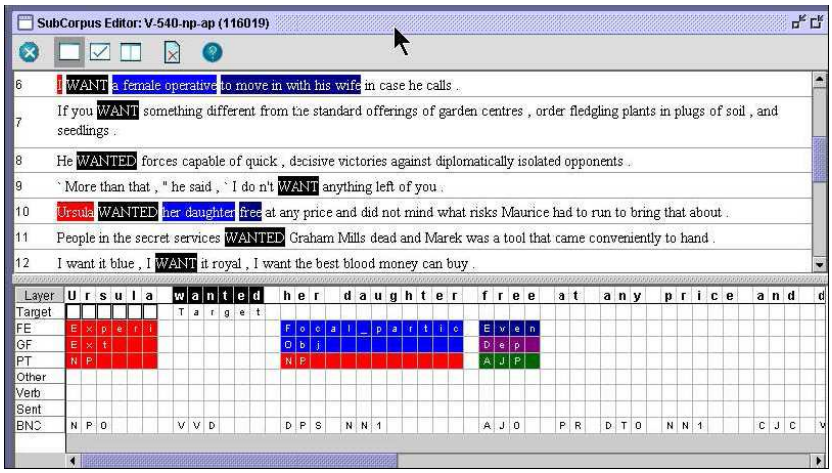


Figura 3.2: Aplicación para el etiquetado de ejemplos en FrameNet

Los marcos semánticos constituyen un método para caracterizar las relaciones semánticas entre palabras. Consideremos por ejemplo los verbos *give* y *receive*. Ambas palabras evocan el marco anterior *Transfer*. Es evidente para un lector que es al menos *parecido* decir que el profesor le dio un libro al alumno, o que el alumno recibió un libro del profesor. En FrameNet se considera que ambos verbos aportan una perspectiva distinta del mismo marco semántico. Por supuesto, las realizaciones sintácticas son distintas, y las funciones gramaticales realizadas por cada elemento del marco semántico también son distintas. Pero a nivel semántico, ambas oraciones quedan relacionadas a través del marco semántico. Esto contrasta con la visión clásica usada por los linguistas para describir las estructuras de argumentos en las teorías del nexo, en la que se utilizan un conjunto más general de roles temáticos. Los roles temáticos tratan de capturar las regularidades existentes en las relaciones entre la semántica y la función gramatical de los constituyentes. Un análisis basado en roles temáticos asignará distintos roles a los participantes de una oración con el verbo *give* y a una oración con el verbo receive.

give	Agent	Theme	Recipient
receive	Recipient	Theme	Source

Con la vista puesta en la utilización de los datos para construir un etiquetador automático de roles semánticos mediante modelos estadísticos, la utilización de

frame elements en lugar de roles temáticos tendrá consecuencias que habrá que considerar a la hora de utilizar FrameNet como corpus de entrenamiento. Por un lado, se gana en generalidad entre distintas palabras que evocan un mismo marco semántico, al tener todas ellas el mismo conjunto de roles temáticos sin importar la perspectiva impuesta por cada palabra. Pero por otro lado, perdemos las generalizaciones relacionadas con las teorías del nexo, que vienen mejor expresadas en términos de roles temáticos, como por ejemplo que el rol *Agent* suele funcionar gramaticalmente como sujeto.

Los distintos marcos semánticos están además relacionados entre si en FrameNet. Existen básicamente dos tipos de relaciones: de herencia y de composición. En las relaciones de herencia, un marco semántico se dice que hereda de otro marco si posee todas las propiedades del marco padre y añade algunos detalles específicos. Por ejemplo:

a. The teacher *gave* the student a message.

b. The teacher *mail* the student a message.

Si enviamos un correo electrónico, estamos realizando una transferencia en el sentido descrito en el marco *Transfer*, sólo que ahora, por ejemplo, el *donante* pasa a ser *emisor*. La descripción del marco semántico será distinta, más especializada. También los nombres de los elementos del marco serán distintos, aunque el número de estos y las propiedades sintácticas de los verbos que evocan el nuevo marco semántico serán idénticos a los del padre.

Otras veces, la relación es de composición. Por ejemplo , en el marco semántico *Commercial Transaction*, evocado por verbos como *sell* o *buy*, podemos considerar que aparecen dos eventos, cada uno de los cuáles vendría a corresponderse con un marco semántico *Transfer*: un comprador da al vendedor dinero, y el vendedor le da algo a cambio. De esta manera, las propiedades sintácticas de estos verbos son las mismas que las de los verbos que evocaban el marco semántico *Transfer*, aunque existen ahora más roles semánticos.

Una vez se tienen claras las palabras que pueden evocar un marco semántico determinado, y se estudian todas las propiedades de valencia de cada una, el proyecto FrameNet busca frases que sirvan de ejemplo de todo esto, y las etiqueta con la información semántica y sintáctica anterior. De esta manera, decimos que el corpus de frases anotadas que nos proporciona FrameNet busca ante todo la exhaustividad, es decir, al menos un ejemplo de todas las combinaciones posibles de cada marco semántico, como si de un diccionario de marcos semánticos se tratara. Esto puede tener sus ventajas para otras aplicaciones, pero veremos que dicha exhaustividad hace difícil de utilizar el corpus en tareas de etiquetado automático de corpus. Recordemos que para que un clasificador estadístico pueda extraer el conocimiento necesario de un corpus, se requiere que haya una masa considerable de individuos de las distintas clases a considerar.

3.3 PropBank

Este recurso, cuyo nombre completo es realmente *Proposition Bank*, consiste en una versión enriquecida del corpus *Penn Treebank II* [30] y [31], que básicamente incluía información de las estructuras sintácticas. A diferencia del enfoque utilizado en el recurso *FrameNet*, en *PropBank* se lleva a cabo un acercamiento eminentemente práctico al problema del etiquetado semántico, de forma que los integrantes del grupo de trabajo del proyecto no estaban interesados en realizar un estudio tan pormenorizado como en *FrameNet* de todas las clases semánticas existentes y de las relaciones de herencia y composición entre ellas, ni tampoco en representar en el corpus fenómenos semánticos globales complejos tales como la correferencia, la cuantificación o la resolución de anáforas. En vez de esto, lo que se busca en *PropBank* es realizar un análisis superficial de la estructura semántica de cada oración, identificando para cada una de las oraciones del corpus *TreeBank* los argumentos o roles semánticos que participan en cada una de las proposiciones. Se pretende con ello disponer de un corpus lo suficientemente amplio como para ser relevante desde el punto de vista estadístico, posibilitando su posterior uso en tareas como la que nos ocupa del etiquetado de roles semántico automático.

PropBank es un recurso más reciente, posterior a los primeros trabajos publicados sobre el etiquetado de roles semánticos automático, y esto queda patente en el enfoque práctico escogido. Mientras en *FrameNet* se intentan analizar *todas* las posibles realizaciones sintácticas de *todas* las clases semánticas existentes en el inglés y aportar un ejemplo para cada una de ellas, constituyendo así una especie de diccionario semántico del idioma, *PropBank* tiene vocación de corpus anotado con roles semánticos útil para la construcción de modelos de aprendizaje automático. De hecho, así como *FrameNet* inspiró el primer trabajo importante sobre etiquetado automático de roles semánticos, la aparición de *PropBank* ha propiciado la explosión de trabajos en este área y la mejora en el rendimiento de los sistemas actuales.

Para cada uno de los verbos que aparecen en el corpus original, se han definido un conjunto de posibles roles semánticos, para posteriormente anotar cada ocurrencia de los mismos en el texto. PropBank se centra exclusivamente en los verbos, estudiando los roles semánticos como argumentos de los verbos. En ningún momento se contempla la posibilidad de que un nombre, adjetivo o adverbio funcionen como núcleos o predicados para un conjunto de roles, tal como ocurría en FrameNet, habiéndose dejado esta tarea para futuras revisiones.

Dada la dificultad de definir un conjunto general de roles semánticos común a todos los predicados posibles, lo cuál sería muy interesante desde el punto de vista de la generalización entre verbos que aportaría, en PropBank se han definido los roles para cada uno de los verbos por separado, pero este proceso se ha realizado tratando de permitir algún grado de generalización entre los distintos verbos, aunque no de manera totalmente estricta. Para cada verbo,

los argumentos o roles semánticos son numerados empezando en 0. Por ejemplo, para un verbo en particular, el rol *Arg0* será habitualmente aquel argumento del verbo que cumple las funciones de *Agente*, mientras que el rol etiquetado como *Arg1* se reservará siempre que sea posible al argumento que funciona como *Paciente*. Para los argumentos siguientes no es posible realizar generalizaciones tan claras, aunque en la medida de lo posible se han intentado seguir unos criterios comunes (en concreto, se utiliza la organización de roles que aparece en el recurso *VerbNet*). Además de los roles numerados específicos de cada verbo, también se definen algunos roles genéricos que pueden ser aplicados a cualquier verbo.

Para cada acepción considerada de un verbo, se definen un conjunto de roles que participan en el predicado en cuestión, recibiendo este conjunto el nombre de *roleset*. Además, cada *roleset* se puede asociar con las posibles realizaciones sintácticas del predicado, indicando las funciones sintácticas en las que pueden aparecer cada uno de los roles anteriores. La unión entre un conjunto de roles y las posibles realizaciones sintácticas es conocida en PropBank como *frameset*. Un verbo polisémico podrá tener de este modo varios *framesets*, siempre que las diferencias en el significado sean lo suficientemente profundas como para requerir participantes o roles semánticos distintos.

Todos los *framesets* utilizados en PropBank son definidos en un fichero (*Frame File*), en el que para cada *frameset* se incluyen:

- El verbo en cuestión junto a un número que indica la acepción que se está considerando.
- El conjunto de roles numerados, junto a un descriptor para cada uno que indica al menos superfluamente cuál es el papel que juega cada argumento en la acepción actual. Este descriptor debe entenderse sólo como un mnemónico informativo para los anotadores que participan en un proyecto, y no tiene ninguna intención teórica.
- Por último, una serie de oraciones de ejemplo extraídas del corpus etiquetadas convenientemente con los roles anteriores , que tratan de reflejar las distintas realizaciones sintácticas en las que puede presentarse el verbo que se está considerando en su acepción actual.

Aquí se muestran un par de ejemplos de *framesets*:

1. Frameset **accept.01** "take willingly"

   Arg0: Acceptor

   Arg1: Thing accepted

   Arg2: Accepted-from

   Arg3: Attribute

   Ex:[Arg0 He] [ArgM-MOD would][ArgM-NEG n't] accept [Arg1 anything of value] [Arg2 from those he was writing about].

2. Frameset **kick.01** "drive or impel with the foot"

   Arg0: Kicker

   Arg1: Thing kicked

   Arg2: Instrument (defaults to foot)

   Ex1: [ArgM-DIS But] [Arg0 two big New York banksi] seem [Arg0 *trace*i] to have kicked [Arg1 those chances] [ArgM-DIR away], [ArgM-TMP for the moment], [Arg2 with the embarrassing failure of Citicorp and Chase Manhattan Corp. to deliver $7.2 billion in bank financing for a leveraged buy-out of United Airlines parent UAL Corp].

   Ex2: [Arg0 Johni] tried [Arg0 *trace*i] to kick [Arg1 the football], but Mary pulled it away at the last moment.

Generalmente, como puede verse en los ejemplos, cada *frameset* consta de dos, tres o hasta cuatro argumentos numerados, aunque existen casos en los que puede haber hasta seis argumentos numerados, especialmente en algunos verbos relacionados con el movimiento como el siguiente:

1. Frameset **edge.01** "move slightly"

   Arg0: causer of motion

   Arg1: thing in motion

   Arg2: distance moved

   Arg3: start point

   Arg4: end point

   Arg5: direction

   Ex: [Arg0 Revenue] edged [Arg5 up] [Arg2-EXT 3.4%] [Arg4 to $904 million] [Arg3 from $874 million] [ArgM-TMP in last year's third quarter].

Además de los argumentos numerados, existe uno especial etiquetado como *ArgA* que se utiliza en situaciones en las que existe más de un argumento funcionando en cierto modo como agente. Por ejemplo, en la frase *Mary hustled John off to school promptly at 7:30 pm*, es *John* quien lleva a cabo la acción de escaparse de clases antes de tiempo, pero aún así *Mary* también está ejerciendo de agente de alguna manera incitando a *John* a llevar a cabo la acción. Es en estos casos en los que se etiqueta a este segundo participante, *Mary* en nuestro ejemplo, con la etiqueta *ArgA*.

Por último, también se utilizan etiquetas para roles que son independientes de los verbos, y que en general podemos asociar con el concepto gramatical de adjuntos (aunque esto no es absolutamente preciso en todos los casos). Estos argumentos se conocen en PropBank como *Funcional tags*. Los argumentos independientes de este tipo que aparecen en PropBank son los siguientes:

Funcional tag	Descripción
ArgM-TMP	Modificador temporal.
ArgM-LOC	Modificador de lugar.
ArgM-DIR	Modificador de dirección.
ArgM-MNR	Modificador de manera o modo.
ArgM-CAU	Indica la causa de algo.
ArgM-ADV	Se utiliza para adverbios de nivel de oración y otros agentes que no queden recogidos en ninguna otra categoría.
ArgM-DIS	Etiquetan a partículas conectivas del discurso.
ArgM-NEG	Partículas de negación.
ArgM-PNC	Indican la motivación (no la causa) de una acción.
ArgM-REC	Indican acciones reflexivas o recíprocas.

Hay dos casos particulares de argumentos funcionales que no son independientes de los verbos, sino que aparecen en los *framesets* como parte de los argumentos participantes en una acepción de un verbo. Son los argumentos *EXT*, que indica un constituyente numérico o de cantidad, y *PRD*, que marca una relación predicativa entre dos argumentos. Este último es un poco más difícil de entender. Sea la siguiente oración:

1. Mary called John a doctor

Existe ambigüedad en el significado de la frase, ya que por un lado podemos entender que *Mary* llamó doctor a John (es decir, dijo que *John era* un doctor), o bien *Mary* llamó a un doctor para que viera a John. En el primer caso, se establece una relación predicativa entre *John* y *doctor*, y por tanto en PropBank vendrá etiquetado el argumento *a doctor* con la etiqueta funcional *PRD*

Mary called John a doctor	Mary called John a doctor
Arg0: Mary	Arg0: Mary
Arg1: John (objeto que es calificado)	Arg1: John (beneficiario)
Arg2-PRD: a doctor (atributo)	Arg2: a doctor (objeto solicitado)

3.4 Comparación entre FrameNet y PropBank

Una vez descrito brevemente en qué consisten ambos recursos anteriores, repasando las claves principales en que se asientan, se hace evidente que, si bien ambos recursos comparten la vocación de aportar conocimiento relativo a la realización sintáctica de los argumentos de los predicados existentes en el inglés, mediante la anotación de los roles semánticos que aparecen en un corpus, las metodologías aplicadas y las motivaciones de cada recurso son distintas. Estas diferencias en la forma del etiquetado resultante y en los métodos utilizados para llevarla a cabo repercuten en los resultados que se obtienen al utilizar un

recurso u otro como base para la construcción de sistemas de etiquetado de roles semánticos. Se llevará a cabo a continuación un repaso por las claves de cada recurso, orientado a resaltar las diferencias esenciales en cuanto a nuestro trabajo se refiere: aquellas que influirán en los sistemas de etiquetado que construyamos.

FrameNet [16] se centra en los marcos semánticos, definidos éstos como una representación esquemática de situaciones que involucran a varios participantes (roles semánticos). La metodología utilizada trata de realizar un recorrido exhaustivo marco a marco. Es decir, se escoge un marco semántico concreto y se definen cuáles son los elementos que participan (*frame elements* según la notación FrameNet, roles semánticos de manera genérica), así como qué palabras evocan dicho marco (predicados). Una vez hecho esto, se buscan frases de ejemplo para cada uno de dichos predicados, y cada una de las realizaciones sintácticas distintas existentes. Se trata por tanto de un trabajo de *documentación* de las realizaciones concretas que se pueden encontrar en un texto en inglés de los distintos marcos semánticos y sus roles participantes, algo así como la construcción de una gran guía de referencia de las relaciones entre semántica y sintáctica enunciadas por las teorías del nexo. En los ejemplos extraídos del corpus (se utiliza el *British National Corpus*), se busca esencialmente la simplicidad antes que la complejidad en las estructuras sintácticas que aparezcan, ya que la finalidad de los ejemplos es resultar explicativos en cuanto a realización sintáctica concreta de un marco semántico, un predicado y unos roles semánticos concretos. Esta vocación de guía de ejemplos es claramente contraproducente de cara a la utilización de dichas frases anotadas como punto de partida para la construcción de modelos estadísticos . Sin embargo,un punto interesante de FrameNet es que la definición de los marcos semánticos, los cuáles pueden ser evocados por múltiples palabras (verbos, nombres y adjetivos), proporciona indirectamente una generalización entre dichos predicados. Éstos compartirán en muchas ocasiones construcciones sintácticas con respecto a sus argumentos muy parecidas, siendo éste un punto positivo de cara a la construcción de modelos estocásticos para el etiquetado automático de roles semánticos.

PropBank [34], por el otro lado, está más interesado en la construcción de un corpus anotado semánticamente para ser utilizado en la construcción de sistemas estadísticos, no como guía o diccionario de relaciones entre sintáxis y semántica como en FrameNet. La metodología ahora no consiste en ir clase a clase buscando un ejemplo simple para cada realización, sino en recorrer frase a frase cada una de las que aparecen en el corpus Penn Treebank, anotando en ellas la aparición de los distintos roles semánticos. Ahora sin embargo no existe la generalización proporcionada por el uso de marcos semánticos que son evocados por distintos predicados como en FrameNet, ya que para cada acepción de cada verbo se utiliza un conjunto determinado de roles semánticos sin preocuparse de agrupar distintos predicados bajo un mismo nombre y un mismo conjunto de roles semánticos. A pesar de ello, sí se intenta mantener la coherencia entre

predicados relacionados o cercanos semánticamente, basándose principalmente en las clases que se establecen en VerbNet a la hora de determinar la cercanía entre distintos predicados. Para estos predicados similares se intenta utilizar un orden coherente y lo más parecido posible en las etiquetas semánticas para los argumentos numerados (*Arg0*, *Arg1*, etc...). Esto sin embargo no ocurre en todos los casos. Por ejemplo, en el caso de los verbos *buy* y *sell*, la persona que se desprende de la mercancía es en el primer caso etiquetada con el rol *Arg1* y en el segundo con el rol *Arg0*. En este caso, lo que se intenta en PropBank es preservar la función gramatical de agente a la etiqueta *Arg0*, algo que en FrameNet no se indica. Otra diferencia fundamental con FrameNet es que en PropBank sólo se identifican como predicados a los verbos, mientras que en FrameNet también los nombres y los adjetivos pueden evocar un marco semántico. Esto tampoco es tan relevante puesto que en la mayoría de los casos los predicados semánticos están basados en verbos, y de hecho los sistemas de etiquetado automático de roles semánticos desarrollados hoy por hoy se centran sólo en encontrar los roles semánticos para los verbos. Una diferencia de PropBank con respecto a FrameNet que sí que es a priori bastante relevante es que los encargados de etiquetar el corpus PennTreebank disponían de un árbol sintáctico para cada frase, a priori, y además debían colocar las etiquetas correspondientes a los roles semánticos del predicado del que se tratase sobre alguno de los nodos de dicho árbol sintáctico. Esto, que es una restricción en términos teóricos importantes (habrá veces que será más correcto seleccionar como rol semántico un conjunto de palabras que pertenezcan a distintos sintagmas), facilita la tarea del etiquetador automático (al menos, a aquellos que utilicen un *parsing* sintáctico completo como entrada al sistema), puesto que el conjunto de posibles candidatos a ser etiquetados con los roles semánticos convenientes queda reducido con respecto al conjunto de todos los posibles grupos de palabras de la frase. En FrameNet por el contrario, los participantes en el proyecto no parten de ningún análisis sintáctico previo, teniendo libertad absoluta para seleccionar los conjuntos de palabras que consideren apropiados para desempeñar cada rol semántico; una vez localizados, asignan etiquetas con funciones gramaticales a estos grupos de palabras. Esto por supuesto establece una dificultad más a la hora de decidirse por FrameNet en lugar de PropBank como recurso base de nuestro sistema de etiquetado.

3.5 Otros recursos de apoyo

3.5.1 WordNet

WordNet [33] es una gran base de datos léxico-semántica creada en 1985 por George A. Miller. El objetivo del proyecto es representar la información semántica de las palabras del inglés con la vista puesta en el procesamiento computacional de dicha información. Así como en un diccionario se expresa el significado de las palabras mediante definiciones, en WordNet se establecen grupos de palabras sinónimas o synsets, de forma que una palabra se define por equiparación con otras que significan lo mismo o, al menos, algo muy parecido. Además también

se establecen relaciones semánticas entre estos synsets, formando una gran red de palabras que da nombre al recurso.

Si una palabra tiene varios significados, aparece en distintos synsets. Para cada uno de los synsets, se incluye una pequeña definición y unos cuántos ejemplos de su uso dentro del lenguaje. Dentro de un synset podemos encontrar palabras individuales o secuencias de palabras que juntas expresan un significado concreto (*collations*), como por ejemplo *máquina de coser*. Para cada palabra se almacena el numero de significados en que aparece (en cuántos synsets está incluida), y para cada acepción, existe una estimación de la frecuencia con la que se da. Actualmente en el proyecto existen 150.000 palabras agrupadas en 115000 synsets (versión 1.5 de WordNet).

Las relaciones semánticas dependen del tipo de palabra sobre la que se definan. Entre los nombres se establecen relaciones de hiperonimia e hiponimia (que en términos informáticos definen una relación de generalización y especialización respectivamente), así como de holonimia, meronimia (relaciones de composición) y términos coordinados (hermanos en la jerarquía de herencia, siguiendo con la metáfora informática). Para los verbos se definen la hiperonimia y la troponimia (esta última es similar a la hiponimia en los nombres), así como la implicación y términos coordinados. Para los adjetivos y adverbios se definen relaciones para indicar si están relacionados con algún nombre,verbo o adjetivo. Todo esto queda resumido en la siguiente enumeración:

- Nombres

  hiperonimias : Y es una hiperonimia(generalización) de X si todo X es un (o algún tipo de) Y

  hiponimias : Y es una hiponimia(especialización) de X si todo Y es un (o algún tipo de) X

  términos coordinados : Y is a término coordinado con X si X e Y comparten una hiperonimia (es una relación conmutativa)

  holonimia : Y es una holonimia de X si X es una parte de Y

  meronimia : Y es una meronimia de X si Y es una parte de X

- Verbos

  hiperonimia : el verbo Y es una hiperonimia del verbo X is la actividad X es un (o algún tipo de) Y

  troponimia : el verbo Y es una troponimia del verbo X si la actividad Y implica hacer X de alguna manera

  implicación : el verbo Y está implicado por X si haciendo X debes estar haciendo Y

  términos coordinados : dos verbos que comparten una hiperonimia

- Adjectivos

  nombres relacionados

participio de verbo

- Adverbios

  adjetivo raíz

También existen relaciones entre palabras directamente, básicamente relaciones entre antónimos y derivados. La relación más importante y en la que más hincapié se hace en Wordnet para los nombres y verbos es la de hypernym (IS A). Según estas relaciones, todos los nombres y verbos están organizados en jerarquías hasta llegar a un conjunto base de categorías generales o primitivas, 25 para los nombres, 15 para los verbos. En la siguiente tabla muestro las categorías base de las que parten todos los nombres por hiperonimia:

act,action,activity	animal,fauna	artifact
attribute,property	body,corpus	cognition,knowledge
communication	event,happening	feeling,emotion
food	groups,collection	location,place
motive	natural object	natural phenomenon
person,human being	plant,flora	possession
process	quantity,amount	relation
shape	state,condition	substance
time		

Los adjetivos están organizados principalmente mediante relaciones de antonimia. Los adverbios se organizan en base a los adjetivos de los que se derivan.

WordNet constituye un precursor de FrameNet, y de ahí por tanto su importancia en el etiquetado de roles semánticos. Además, la información contenida en WordNet podría servir para mejorar los sistemas de etiquetado de roles semánticos, por ejemplo enriqueciendo recursos, o utilizando el synset al que pertenece una palabra para ayudar a decidir sobre el rol semántico que desempeña.

3.5.2 VerbNet

VerbNet [24] es un diccionario léxico de verbos organizado jerárquicamente, que aporta información sintáctica y semántica para los verbos del inglés. Los verbos se agrupan formando clases según criterios semánticos, pudiendo un mismo verbo estar en varias clases si posee distintas acepciones. Cada una de estas clases es descrita en términos sintácticos y semánticos, detallando los argumentos o roles semánticos que participan en el predicado en cuestión, y las distintas alternancias y transformaciones sintácticas que pueden darse. Es claro el parecido con los *rolesets* de PropBank, siendo la diferencia fundamental que el conjunto de roles utilizado son temáticos. Además, las clases de verbos que se definen son

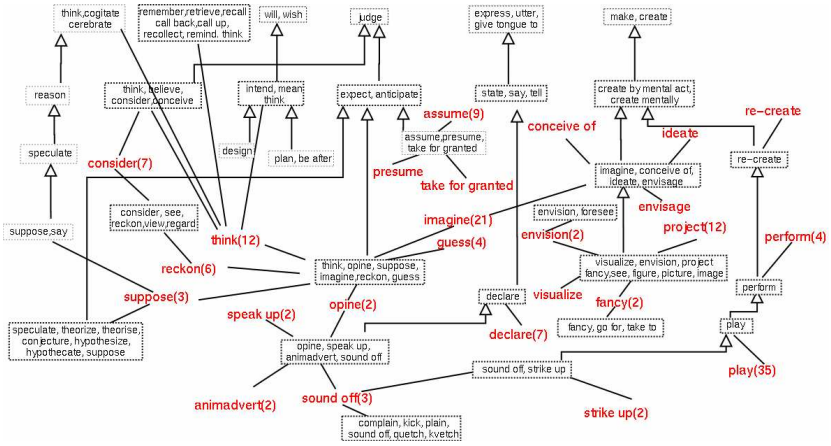


Figura 3.3: Representación gráfica de algunos synsets (cuadrados punteados), palabras que pertenecen a los mismos (en rojo) y relaciones de hiperonimia entre los synsets

las mínimas posibles en términos de que representen un mismo conjunto de roles involucrados y las mismas realizaciones sintácticas, a diferencia de PropBank donde para cada verbo existe un *roleset*, y no se hacen generalizaciones entre los distintos predicados.

Actor	Agent	Theme	Patient
Asset	Attribute	Beneficiary	Cause
Destination	Experiencer	Instrument	Location
Material	Pacient	Product	Recipient
Source	Stimulus	Time	Topic

Tabla 3.2: Roles temáticos de VerbNet

concrete	abstract	location
organization	currency	communication
phys_obj	human	animate
body_part	int_control	...

Tabla 3.3: Ejemplo de restricciones a los roles temáticos de VerbNet

El primer nivel de la jerarquía descrita en VerbNet está formado por las clases verbales de Levin [29], y posteriormente cada clase es refinada destacando las diferencias sintácticas y semánticas dentro de la misma. Cada nodo de la red está caracterizado por el conjunto de verbos que forman la clase, y por una lista de argumentos semánticos para dichos verbos, así como información semántica y

sintáctica sobre los verbos. La lista de argumentos consiste en una serie de argumentos temáticos extraídos de un total de 20 posibles (ver tabla 3.2), y también en ocasiones una serie de restricciones sobre estos argumentos mediante el uso de predicados binarios (ver tabla 3.3). La información sintáctica se expresa asignando a los argumentos semánticos anteriores argumentos sintácticos profundos. Por último, la información semántica que se ofrece de los verbos consiste en un conjunto de predicados semánticos, como *motion*, *contact* o *transfer_info*. Estos predicados toman como argumentos los propios argumentos semánticos del verbo y también algunos eventos temporales y existenciales. Por tanto, la definición semántica de las clases se hace siempre en base a una serie de eventos y predicados lógicos. Se puede ver un ejemplo de todo esto en 3.4, correspondiente a los datos incluidos en VerbNet para la clase *hit-18.1*. Aunque no se incluyen los verbos que forman la clase, puede verse en los ejemplos que no sólo hablamos del verbo *hit*, sino de una serie de verbos que comparten comportamiento sintáctico y semántico. Por tanto, el nombre de la clase debe entenderse sólo como un mnemónico.

Según se describe en [22], VerbNet 1.0 contiene descripciones para 4100 verbos, distribuidos en 191 clases de primer nivel y 74 de segundo nivel en la jerarquía (especializaciones de las anteriores). Existen 21 roles temáticos, 36 restricciones de selección, 314 estructuras sintácticas y 64 predicados semánticos.

Existe un mapeo entre los verbos de PropBank y las clases de VerbNet [23], mediante el cuál es posible interpretar los roles de PropBank como roles temáticos, además de realizar generalizaciones entre los verbos que PropBank por sí solo no permite dado que cada acepción de cada verbo constituye un único elemento de estudio. De esta forma, actualmente en la base de datos de PropBank consta la clase de VerbNet a la que pertenecen cada una de las acepciones que se contemplan.

Class	hit-18.1			
Parent	—			
Themroles	Agent Patient Instrument			
Selfrestr	Agent[+int_control] Patient [+concrete] Instrument[+concrete]			
Frames	Name	Example	Syntax	Semantics
	Basic Transitive	Paul hit the ball	Agent V Patient	cause(Agent, E) manner(during(E),directedmotion,Agent) !contact(during(E),Agent,Patient) manner(end(E),forceful,Agent) contact(end(E),Agent,Patient)
	Resultative	Paul kick the door open	Agent V Patient Adj	cause(Agent, E) manner(during(E),directedmotion,Agent) !contact(during(E),Agent,Patient) manner(end(E),forceful,Agent) contact(end(E),Agent,Patient) Pred(result(E),Patient)
	Resultative	Paul hit the window to pieces	Agent V Patient Prep[to/into] Oblique [+state]	cause(Agent, E) manner(during(E),directedmotion,Agent) !contact(during(E),Agent,Patient) manner(end(E),forceful,Agent) contact(end(E),Agent,Patient) Pred(result(E),Patient)
	Conative	Paul hit at the window	Agent V at Patient	cause(Agent, E) manner(during(E),directedmotion,Agent) !contact(during(E),Agent,Patient)

Tabla 3.4: Entrada simplificada de VerbNet para la clase *hit-18.1*

3.5.3 ConceptNet

ConceptNet es una base de datos de conocimiento que trata de capturar las relaciones entre los conceptos que se utilizan en un lenguaje, relaciones que podríamos entender que conforman lo que llamamos *sentido común*.La idea es tener representada toda esta información mediante una gran red de conceptos, de forma que las aplicaciones de inteligencia artificial puedan hacer uso fácilmente de esta información. El punto de vista de ConceptNet es siempre puramente práctico, y se aleja de las rigurosidades lingüísticas formales que caracterizan a recursos como WordNet. Lo importante es representar las relaciones semánticas que se dan entre los diversos conceptos que maneja el ser humano, entendiendo estas relaciones semánticas de una manera mucho más relajada que en WordNet. Podemos ver un ejemplo de la red en que consiste ConceptNet en la figura 3.4.

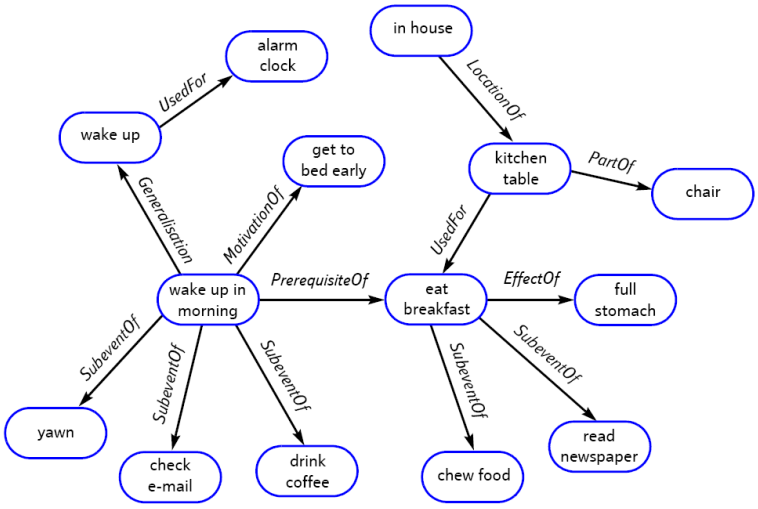


Figura 3.4: Representación gráfica de un extracto de conceptos y relaciones de ConceptNet

La base de datos de ConceptNet es enorme, contando actualmente con más de 300.000 conceptos (nodos de la red) y más de 1.6 millones de relaciones. Toda esta enorme información se ha generado automáticamente a partir de las más de 700.000 frases introducidas por usuarios mediante una página web denominada Open Mind Common Sense. Esta iniciativa proponía diversos mini-juegos a los usuarios, pidiéndoles que relacionasen diversos conceptos, o que indicarán las consecuencias de algún evento o situación típica, etc., de manera que se iban reconociendo los nuevos conceptos introducidos por los usuarios y se utilizaban éstos para generar nuevas situaciones que plantear a los usuarios. Con esto se obtuvo una gran cantidad de frases que condensaban el sentido común de los participantes (o al menos, parte de éste). Es a partir de estas frases y mediante

procedimientos automáticos (ver [18] para una descripción de dichos procedimientos) como se generó la base de datos de ConceptNet, empleando gracias a la participación de usuarios anónimos muchos menos recursos en tiempo y dinero que otros recursos como WordNet o PropBank. Por supuesto, este método basado en la participación desinteresada de colaboradores anónimos es válido en ConceptNet por la naturaleza práctica y poco rigurosa escogida; usar este mismo enfoque en la construcción de corpus que se basan en la exactitud y la corrección lingüística sería mucho menos trivial y desde luego nunca totalmente automático.

La red de conceptos de la que se compone ConceptNet es parecida estructuralmente a la de WordNet: tenemos nodos que representan conceptos y aristas que unen dichos nodos mediante ciertas relaciones semánticas. Pero dónde WordNet se decanta por la corrección y la rigurosidad, ConceptNet se basa en la flexibilidad y adaptabilidad de criterios, aún redundando en que la información plasmada tenga menos entidad teórica (recordemos que el enfoque de ConceptNet es siempre práctico, pues es ésta la única forma de abarcar el conocido como *sentido común*, conocimiento universal que de ningún modo puede ser expresado de forma rígida e inapelable, sino mediante aproximaciones inexactas). Las diferencias de base entre WordNet y ConceptNet son tres.

- Los nodos en WordNet son siempre items léxicos, generalmente palabras, y en algunos casos conjuntos de palabras que funcionan como una única entidad léxica (coche de carreras, máquina de afeitar...). Sin embargo en ConceptNet se permiten construcciones más complejas que permiten representar conceptos compuestos de más alto nivel, pudiendose para ello utilizar un verbo y varios argumentos del mismo para representar un solo concepto que vendrá a ser un nodo de nuestra red. Por ejemplo, son conceptos válidos en ConceptNet "comprar comida", "ser despedido del trabajo" o "tener un hijo".

- Las relaciones semánticas que se permitían en WordNet eran básicamente las de sinonimia/antonimia, la especialización/generalización y la composición. En ConceptNet se permite un conjunto mucho más amplio de relaciones semánticas, que permiten que la red resultante esté mucho más conectada: conceptos que a priori, desde el punto de vista semántico, no parecen estar relacionados, pueden estarlo a traves de estas relaciones pragmáticas. Por ejemplo, "ser despedido" puede ser *efecto de* "tener problemas personales" (la relación en cuestión en ConceptNet sería *EffectOf*). El conjunto completo de relaciones semánticas (o pragmático-semánticas, para ser más exactos) se encuentra en la tabla 3.5.

- El conocimiento que contiene ConceptNet es mucho más informal, constituye una visión práctica, en contraposición con el enfoque riguroso y rígido de WordNet. Un ejemplo de esto es la existencia de multitud de relaciones que no son irreprochables o universales, como sería condición en WordNet, sino que pueden ocurrir en determinadas ocasiones (o incluso puede que

K-LINES	ConceptuallyRelatedTo ThematicKLine SuperThematicKLine
THINGS	IsA PropertyOf PartOf MadeOf DefinedAs
AGENTS	CapableOf
EVENTS	PrerequisiteEventOf FirstSubEventOf SubEventOf LastSubeventOf
SPACIAL	LocationOf
CAUSAL	EffectOf DesirousEffectOf
FUNCTIONAL	UsedFor CapableOfReceivingAction
AFFECTIVE	MotivationOf DesireOf

Tabla 3.5: Relaciones disponibles en ConceptNet

ocurran sólo muy de vez en cuando). Por ejemplo, la relación de *efecto* entre "caerse de la bicicleta" y "hacerse daño" no es para nada obligatoria. Pero esto no quita que dicha relación forme parte del sentido común de cualquier persona, el cuál nos lleva a preguntarle a alguien que se ha caído de una bicicleta *si se ha hecho daño*. Es esta clase de flexibilidad la que otorga potencia a ConceptNet para su uso en diversas aplicaciones de inteligencia artificial, y también por ende la que le resta utilidad para tareas formales como la que nos ocupa en el presente trabajo.

A pesar de que, a priori, este recurso no está enfocado al etiquetado de roles semánticos, y de hecho no he encontrado ningún trabajo en el que se haga o se proponga el uso de ConceptNet como apoyo a la tarea en cuestión, he incluido este apartado porque una de las vías que quiero explorar en mi investigación futura es precisamente la forma de utilizar este conocimiento sobre el sentido común en un sistema de etiquetado de roles semánticos. Me parece intuitivamente claro a priori que dicho conocimiento *tiene* que ser útil en cualquier tarea relacionada con la semántica, y por tanto también en la tarea concreta del etiquetado de roles semánticos. Quedándonos únicamente en la tarea preliminar necesaria de identificar él marco o clase semántica de una frase (para lo que hay que desambiguar el sentido del predicado), el sentido común podría indicarnos cuál debe ser la acepción correcta, o al menos descartar definitivamente alguna de ellas. Por ejemplo, si leemos que "Juan suspendió ocho asignaturas", nuestro

sentido común relaciona el concepto *asignatura* con la acción de *suspender* en su acepción de *no sacar la nota mínima necesaria para aprobar*. Sin embargo, si leemos que "El ayuntamiento suspendió los actos previstos para hoy", el mismo sentido común anterior descarta definitivamente el significado anterior de *suspender*. Todo estos es por supuesto una visión intuitiva de la posible utilidad de ConceptNet que todavía tengo que explicitar en métodos concretos y evaluables.

Capítulo 4

Arquitectura de un Etiquetador de Roles Semánticos Estadístico

4.1 Arquitectura del sistema

El primer sistema de etiquetado automático de roles semánticos que fue presentado a la comunidad científica fue el construido por Jurafsky y Gildea, en 2002, en su ya clásico artículo [17]. En este artículo se sentaron las bases de todos los sistemas de etiquetado de roles semánticos aparecidos hasta la fecha, al menos en los siguientes aspectos:

- Se emplean métodos estadísticos para construir modelos a partir de corpus etiquetados semánticamente (FrameNet en el artículo de Gildea, PropBank en la mayoría de los sistemas actuales). Por tanto, una de las variables fundamentales para determinar la eficacia de los sistemas será la calidad y extensión de los recursos utilizados en estas técnicas de aprendizaje automático, además de las características particulares de cada recurso (ver capítulo 3 sobre recursos semánticos).

- Basándose en las teorías del nexo, que recordemos enunciaban las conexiones existentes entre el contenido semántico de una sentencia y sus posibles realizaciones sintácticas, todos los sistemas hacen uso de *parsers* sintácticos, ya sean completos o superficiales, aplicados a los textos de entrada y las frases a etiquetar. Esta información sintáctica es utilizada para la extracción de características que serán introducidas en los algoritmos estadísticos para generar los modelos de etiquetado automático. En otros trabajos, además de esta información, se emplean etiquetadores morfosintácticos, reconocedores de entidades y otras herramientas y recursos lingüísticos como apoyo a la construcción del vector de características.

- El conjunto de características a extraer de las oraciones que se van a etiquetar para formar el vector de entrada al clasificador, propuesto por Gildea y Jurafsky en su artículo, se ha mantenido prácticamente invariable en la mayoría de los sistemas actuales. Estas características están basadas en su mayor parte en la información proporcionada por el analiador sintáctico, y serán descritas más adelante. Desde luego, muchas otras nuevas características a introducir en los clasificadores han sido sugeridas por diversos trabajos, e incluso algunos han llevado a cabo un análisis crítico de las características utilizadas y de la posible inutilidad o solapamiento de algunas de ellas [50].

- Por último, los autores del artículo realizaron una división en dos subtareas que se ha mantenido en la mayoría de los sistemas actuales, aunque existen enfoques distintos que después serán comentados. Dada una frase de entrada, lo primero que se lleva a cabo es identificar qué conjuntos de palabras son candidatos a ser considerados roles semánticos del predicado en cuestión, mediante un clasificador binario. Después, cada uno de los candidatos pasa por un clasificador de multiclases que decide qué rol semántico en concreto se le asigna o si no le corresponde ninguno.

Además de las dos fases citadas, denominadas *argument identification* y *argument classification* respectivamente, aparecen a lo largo de los distintos sistemas de etiquetado de roles semánticos otras etapas previas (y en algún caso posteriores, como se verá). A pesar de la gran cantidad de variaciones que se pueden encontrar en los distintos artículos, se intentará mostrar una arquitectura prototipo que refleje el mínimo común denominador de todos los sistemas estudiados así como las características más asentadas en la literatura actualmente. La arquitectura en cuestión puede verse representada en la figura 4.2. La notación utilizada en la misma para referirse a los roles semánticos está extraída de la terminología utilizada en PropBank, aunque un sistema basado en FrameNet tendría una arquitectura equivalente.

El problema del etiquetado de roles semánticos no es trivial y puede ser enfocado desde distintas perspectivas. La más simple de ellas sería entender el problema como una tarea de etiquetado secuencial, de tal forma que a cada palabra de la frase de entrada, se le debe asignar una etiqueta de entre el conjunto de roles del predicado en cuestión. Este proceso se irá haciendo palabra a palabra. Esta primera aproximación ofrece, sin embargo, muchas dificultades. En primer lugar, etiquetando de esta manera se puede llegar a resultados del todo incoherentes, ya que existen ciertas condiciones a respetar tales como el no solapamiento de roles semánticos. Además, la tarea tiene un claro componente de decisión global que no se contempla con un acercamiento tan local. No se deben olvidar tampoco los enunciados de las teorías del nexo, según los cuales existen relaciones complejas entre las estructuras sintácticas y los contenidos semánticos del lenguaje; dichas relaciones no están siendo explotadas en un acercamiento tan básico como el mostrado. Por todas estas razones, y algunas

Figura 4.1: Enfoque secuencial vs. enfoque jerárquico en un sistema de etiquetado de roles semánticos. En el enfoque secuencial, el sistema decide la etiqueta a asignar a cada palabra, una tras otra, generando etiquetados potencialmente incoherentes. En el enfoque jerárquico, el sistema decide la etiqueta a asignar a cada nodo del árbol sintáctico de la oración, facilitando la obtención de etiquetados coherentes.

más que aún no son visibles en este punto de la explicación de la arquitectura, la aproximación al etiquetado de roles semánticos como una tarea de etiquetado secuencial, palabra a palabra, no es la mejor elección. A pesar de ello, existen sistemas construidos siguiendo esta filosofía, con resultados mejores de los que cabría esperar en un primer momento [25].

Otro enfoque distinto sería entender el etiquetado de roles semánticos como una tarea de etiquetado jerárquico. En este caso, la entrada al etiquetador va a ser una estructura en forma de árbol, en nuestro caso el árbol sintáctico de la oración a etiquetar. Las etiquetas serán asignadas a los nodos de dicho árbol. Esta forma de proceder parece más lógica, en primer lugar porque es evidente en la mayoría de los casos que los roles semánticos son desempeñados por unidades sintácticas completas. Etiquetando los nodos del árbol sintáctico minimizamos el espacio de búsqueda, que en el enfoque secuencial puro estaría formado por todas las posibles combinaciones de palabras de la oración de entrada al sistema. Otra ventaja de esta elección es que nos permite realizar una poda previa del árbol sintáctico o *pruning*, aplicando para ello un conjunto de reglas muy simples que serán descritas más adelante. Se consigue así reducir aún más el conjunto de candidatos a ser identificados como roles semánticos. De todas formas, no todo son ventajas, ya que la utilización necesaria de *parsers* sintácticos introduce errores en el sistema que pueden repercutir en los resultados finales del etiquetador de roles. A pesar de ello, este enfoque es sin lugar a dudas el preferido por los distintos equipos que trabajan en la implementación de etiquetadores de roles semánticos.

Un enfoque a medio camino que ha demostrado buenos resultados consiste en la utilización de *parsers* sintácticos superficiales, también conocidos como

chunkers. Esto algoritmos consiguen dividir las frases en los distintos sintagmas que la conforman, pero no generan el árbol sintáctico completo. Por tanto, la tarea vuelve a ser entendida como un etiquetado secuencial, pero el conjunto de candidatos para la fase de identificación de los roles semánticos también se ha visto reducido. La ventaja fundamental es la menor probabilidad de error de los *chunkers* frente a los *parsers* sintácticos completos, a costa de ofrecer menos información e imposibilitar la extracción de algunas características que se basan en la misma. Un ejemplo de sistema construido utilizando un *chunker* es [21]. En el artículo [47] por su parte, se plantea la utilización de ambas entradas combinadas, dependiendo de la fase en la que nos encontremos.

Ya se emplee un analizador sintáctico completo o superficial, la arquitectura genérica de un sistema actual de etiquetado de roles semánticos consta de las fases que se muestran en la figura 4.2, que pasarán a ser descritas y comentadas en los siguientes apartados. Además de las fases que conforman la arquitectura en cadena (*frame identification, pruning, argument identification, argument classification, inference*), se llevan a cabo de manera paralela otras tareas tomando como entrada el texto a etiquetar. Dicho texto es introducido en una serie de herramientas linguísticas, tales como etiquetadores morfosintácticos, reconocedores de entidades, analizadores sintácticos alternativos, ..., además de ser enriquecido en algunos casos mediante el uso de recursos lingüísticos de apoyo como WordNet o VerbNet. La finalidad de estos procesos es obtener un cierto nivel de formalización y estructuración linguística del texto, sobretodo a nivel léxico y sintáctico, que será posteriormente usada en un proceso de extracción de características, las cuales conformarán el vector de entrada a los algoritmos de clasificación de las fases de identificación y clasificación de argumentos. La explicación de las distintas características encontradas en la bibliografía revisada se encuentra en secciones siguientes de este mismo capítulo.

4.1.1 Frame Identification

Para cada una de las proposiciones que conforman nuestra oración de entrada, la primera tarea a llevar a cabo es decidir dentro de qué marco semántico se encuentra la misma. Existirá una palabra (*predicate* en PropBank,*target word* en FrameNet) que ejerce de núcleo semántico en la proposición, generalmente un verbo, aunque ésta puede ser también un adjetivo o un nombre (al menos en el recurso FrameNet; en PropBank, a día de hoy, sólo se tienen en cuenta a los verbos como núcleos semánticos). Simplificaremos la explicación centrándonos en que dicha palabra sólo pueda ser un verbo. Este verbo puede tener distintas acepciones, entendidas como distintos significados expresados en un diccionario. Por ejemplo, ciñéndonos a la nomenclatura usada en PropBank, veamos dos posibles acepciones para el verbo *act*:

1. **act.01** *"play a role"*: Hoare Govett is acting as the consortium 's investment bankers

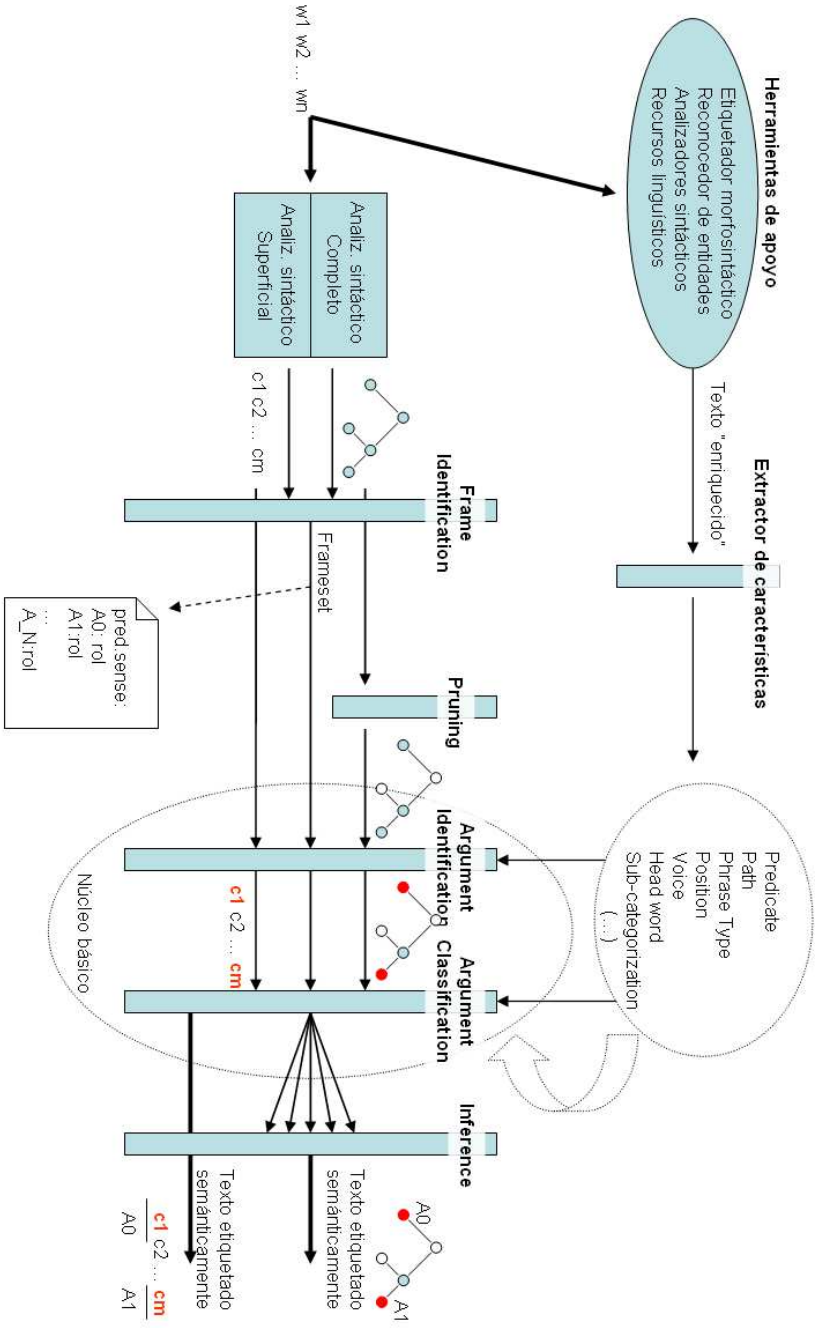


Figura 4.2: Arquitectura genérica de los sistemas actuales de etiquetado de roles semánticos

2. **act.02** *"do something"*: Why did n't the Bank Board act sooner ?

De forma que, una vez encontrado el verbo, y lematizado convenientemente para obtener el infinitivo, la primera dificultad que se encuentra quien aborda la implementación de un sistema de etiquetado de roles semánticos es decidir qué acepción del verbo se está utilizando de entre las posibles. Esta problemática es ampliamente conocida para todos los investigadores en el campo del Procesamiento del Lenguaje Natural bajo el nombre de *desambiguación de significados*. La mayoría de los autores no incluyen por tanto esta fase como parte del etiquetado de roles semánticos, considerando que es una tarea previa totalmente independiente que no deben abordar como parte del sistema. No se va a profundizar por tanto en esta fase, pero ha sido incluida puesto que en arquitecturas que no sean exclusivamente en cadena podría ser interesante algún tipo de retroalimentación entre las salidas de fases posteriores y ésta que nos ocupa. Tengamos en cuenta que la propia tarea de desambiguación de significados es una de las aplicaciones inmediatas de los sistemas de etiquetado de roles semánticos; por tanto, podrían llevarse a cabo la aplicación de técnicas de bootstrapping que mejoraran ocasionalmente los resultados.

Podríamos incluir en esta fase también la tarea consistente en, dada una acepción, elegir qué conjunto de roles semánticos se van a instanciar. Por ejemplo, para el ejemplo anterior, PropBank nos sugiere varios escenarios posibles para la primera acepción del verbo *act*, pudiendo aparecer sólo un rol *ARG0*, o bien éste junto con *ARG1*. Todos los artículos leídos no tienen en cuenta esta información, usando de manera invariable el conjunto de etiquetas posibles ARG0-ARG5, más las funcionales. A mi parecer, decidir en una fase previa qué argumentos o roles debemos instanciar puede otorgar cierta ventaja. Para ello, se podrían construir clasificadores independientes para cada uno de los verbos, entrenados de modo que pudiesen decidir antes una frase de entrada qué acepción y conjunto de roles deben utilizarse en fases posteriores. El principal problema para llevar a cabo esto es la escasez de ejemplos en los recursos actuales, que muy posiblemente no otorguen la suficiente relevancia estadística para construir los clasificadores independientes propuestos. Aún así será esta una de las líneas de investigación a tener en cuenta en un futuro.

4.1.2 Pruning

Aquellos sistemas que optan por la utilización de un analizador sintáctico completo suelen introducir una fase de poda del árbol o *pruning*. La fase en cuestión consiste en eliminar aquellos nodos que claramente no pueden ser roles semánticos, o al menos aquellos que muy probablemente no lo sean. Para ello, la mayoría de los sistemas utilizan un algoritmo bastante simple que consiste en lo siguiente:

1. Nos colocamos en el nodo correspondiente al predicado cuyos roles estamos tratando de instanciar.

2. Añadimos a nuestra lista de nodos candidatos todos los nodos hermanos del nodo actual, a no ser que estos nodos hermanos estén coordinados con el nodo actual. Si uno de los nodos hermanos es un sintagma preposicional, también se añaden los nodos hijos de la generación inmediatamente posterior del mismo.

3. Nos desplazamos ahora al nodo padre y repetimos el paso 2, hasta que hayamos llegado al nodo raíz.

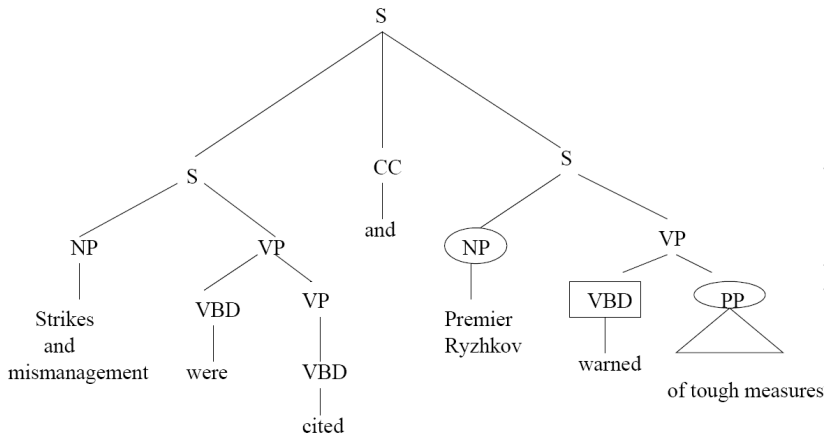


Figura 4.3: Ejemplo de la aplicación del algoritmo de pruning de Xue y Palmer. Situados en el segundo predicado (VBD: warned), primero se añadirán los nodos hermanos, en este caso un sintagma preposicional. Añadimos los hijos de dicho sintagma por ser preposicional. Subimos ahora al nodo padre (VP), añadimos el sintagma nominal hermano. Volvemos a subir de nodo (S). El único hermano que tiene (otra proposición) está coordinado con el actual, por lo que no se añade a la lista de candidatos. Por último, volvemos a subir al nodo padre, llegando a la raíz del árbol y acabando el algoritmo.

Esta estrategia de poda fue propuesta en [50]. Existen otros acercamientos al problema, como el propuesto en [40]. En este sistema, se utiliza un clasificador binario para decidir si un nodo es o no candidato a pasar a la siguiente fase de la arquitectura. El clasificador es entrenado con todas las frases del corpus de entrenamiento. La ventaja fundamental de este método es que permite distintos grados de filtrado, según establezcamos un valor u otro como umbral de probabilidad para admitir el nodo como candidato.

Si el sistema en cuestión ha optado por la utilización de un analizador sintáctico parcial, la fase de *pruning* tal como ha sido descrita no tiene sentido. En la mayoría de los casos, se opta por eliminarla, de forma que todos los *chunks* o sintagmas detectados por el analizador, así como cualquier conjunto de sintagmas sucesivos, son candidatos a ser identificados como roles semánticos.

4.1.3 Argument Identification

Esta fase y la siguiente forman el núcleo básico de todo sistema de etiquetado de roles semánticos. Cabría plantearse la tarea en una sola fase en lugar de dos, esto es: para cada posible candidato a ser etiquetado como rol semántico, utilizar un clasificador que lo etiquete como alguno de los posibles roles (por ejemplo, si utilizamos PropBank, A0-A5 o alguno de los roles funcionales), o con una etiqueta especial que indique que no es ningún rol semántico. Sin embargo, si se utiliza este enfoque, se presenta un problema de desbalanceo considerable de las clases del clasificador, ya que la etiqueta utilizada para señalar aquellos constituyentes que no forman parte de ningún rol semántico aparecerá de forma mucho más frecuente que el resto de las etiquetas. Todos los investigadores con experiencia en aplicaciones de minería de datos para la construcción de clasificadores de multiclases saben que para solucionar este tipo de desbalanceos es preciso utilizar en primer lugar un clasificador binario, que sea capaz de decidir si una muestra pertenece o no a la clase que produce el desbalanceo. Posteriormente, para aquellas muestras que no pertenezcan a dicha clase, se empleará un clasificador de multiclases que decida la etiqueta a aplicar a la muestra de entre el resto de clases que queden. Es justamente ésta la estrategia seguida desde el primer artículo aparecido sobre etiquetado de roles semánticos, y que se ha mantenido invariable en el resto de los sistemas revisados.

En el caso de contar con parsing sintáctico completo, en la fase de identificación de argumentos se utilizarán clasificadores binarios entrenados y ejecutados sobre el conjunto de los candidatos obtenidos en la fase de *pruning*. Fijémonos en que *sólo* aquellos candidatos seleccionados en dicha fase, que a su vez han sido seleccionados de entre los nodos generados por el analizador sintáctico completo, pueden optar a ser etiquetados como roles semánticos. Es decir, cualquier error por omisión en la fase de pruning, o anteriormente en el analizador sintáctico, producirá inevitablemente un fallo por omisión en esta fase de identificación de argumentos y consecuentemente en el resultado final. Es ésta la principal limitación de utilizar una arquitectura serie, en la que el resultado de cada etapa funcione como entrada de la siguiente etapa. Una tarea importante en la construcción de un sistema como el que nos ocupa será la calibración adecuada de las fases anteriores a la de clasificación de argumentos, para asegurarnos *al menos* un valor alto de *recall*, aún a pesar de bajar un poco en precisión, de manera que limitemos en lo posible la pérdida de candidatos válidos en el trascurso de las distintas etapas.

Si no se cuenta con un analizador sintáctico completo, cualquier subconjunto de palabras contiguas o *chunks* contiguos puede ser entendido como un posible candidato. En la mayoría de los artículos revisados se entrena un clasificador binario que determina si una palabra o *chunk* es comienzo de un argumento, y otro que determina si una palabra o *chunk* es final de un argumento. Después, en la ejecución, se contrastan las predicciones de ambos clasificadores mediante algoritmos de programación dinámica, maximizándose la probabilidad de la com-

binación elegida y asegurándose de que se cumplen las condiciones necesarias para que se formen roles semánticos válidos. Recordemos que estas condiciones eran las siguientes:

1. Los argumentos no pueden incluir al predicado (al verbo)
2. Un argumento no puede estar "a caballo" entre dos cláusulas.
3. Los argumentos deben estar contenidos en la misma cláusula que el predicado al que pertenecen.

A pesar de que lo explicado es la estrategia más repetida en los distintos sistemas analizados, existen trabajos que utilizan otros acercamientos. Los hay que fusionan ambas fases de identificación y clasificación en una sola, haciendo uso de una etiqueta adicional para señalar los constituyentes que no son roles semánticos, ignorando el problema del desbalanceo de dicha clase [46]. En otros casos, se utiliza un clasificador con tres clases en lugar de uno binario, distinguiéndose a la salida entre candidatos descartados, aceptados y "probables" [42].

A la hora de implementar el clasificador (o los clasificadores, si no se dispone de analizador sintáctico completo), existen multitud de opciones por las que decantarnos: algoritmos de aprendizaje basados en modelos de Markov, Support Vector Machines, Conditional Random Fields, ... Los principales algoritmos utilizados actualmente en la construcción de etiquetadores de roles semánticos han sido descritos brevemente en el primer capítulo de este trabajo. Todos estos algoritmos de aprendizaje basan sus decisiones en los datos ofrecidos por el vector de características, que recoge información extraída de los datos de entrada. Dicha información ha sido determinada por los diseñadores del sistema como relacionada y útil con la tarea que debe afrontar el sistema de aprendizaje. En el caso que nos atañe, será en su mayoría información relacionada con la estructura sintáctica de la oración de entrada, siguiendo así las indicaciones de las teorías del nexo. Las características más consensuadas por todos los grupos de trabajo, así como algunas menos comunes, serán descritas y comentadas en una próxima sección de este capítulo. En principio, la mayoría de los sistemas utilizan el mismo conjunto de características para las fases de identificación y clasificación de argumentos. Sin embargo, algunos trabajos como [50] han demostrado que algunas de las características consideradas ya genéricas son útiles para una de las fases pero totalmente carentes de información para la otra fase. Se hará hincapié sobre esto en la sección de descripción de las características.

4.1.4 Argument Classification

En esta fase se parte de un conjunto de candidatos a ser roles semánticos, que pueden ser nodos del árbol de sintaxis que han pasado positivamente las fases de *pruning* y *argument identification*, o conjuntos de palabras o *chunks* que han sido identificados en la fase anterior como roles semánticos (por ejemplo,

mediante el uso de un par de clasificadores binarios para detectar comienzo y final de roles y la posterior aplicación de algoritmos de programación dinámica para obtener resultados coherentes). Generalmente se utilizará un clasificador de multiclases para asignarle a cada candidato la etiqueta adecuada de entre los posibles roles semánticos.

Recordemos que la mayoría de los sistemas mantienen constante el conjunto de clases del clasificador, conteniendo todas las posibles etiquetas semánticas del recurso que estemos usando. Por ejemplo, si usamos PropBank, las etiquetas o clases del clasificador serán A0-A5, las funcionales AM-xxx, y algunas otras como las relacionales (ver capítulo 3). Además, cada posible candidato se etiqueta de manera independiente, esto es, sin tener en cuenta qué etiquetas han sido ya asignadas, con lo que puede ocurrir:

- Que aparezcan roles semánticos que no sean correctos para la clase semántica en la que nos encontramos.
- Que aparezcan roles semánticos duplicados.

El primero de los problemas puede evitarse introduciendo como característica de entrada al clasificador la clase o *frameset* en nomenclatura PropBank a la que pertenece la oración actual. De esta manera, el clasificador debería ser capaz de utilizar dicha información para decidir qué roles semánticos no tiene sentido instanciar. Esto se aplica en todos los sistemas revisados, que siempre incluyen como característica de entrada a los clasificadores el predicado (verbo) sobre el que se trabaja. Por supuesto, con esto no evitamos que ocasionalmente el clasificador se decida por roles semánticos improcedentes, puesto que la capacidad de discriminación de los datos de entrada que aporta la clase semántica es pequeña en relación a otras características. Otra solución sería generar distintos clasificadores, con distintos conjuntos de clases, para cada uno de los verbos o *framesets* en general con distinto número de roles semánticos; esto es, aquellos verbos que siempre van acompañados de un solo rol semántico, constituirían un corpus de entrenamiento para un clasificador particular. Aquellos que como mucho necesitan de un par de roles semánticos, otro clasificador distinto, y así sucesivamente. Esta estrategia no ha sido utilizada por ninguno de los sistemas revisados, o al menos no ha sido detallada como tal en los artículos que los describen, por lo que sería necesario un desarrollo experimental para constatar su utilidad. Y además aún nos queda por resolver el segundo de los problemas, la duplicidad de roles.

En general, ambos problemas pueden ser vistos como consecuencias de un tratamiento local de un problema con ciertas restricciones globales. Al igual que en la fase de identificación de argumentos para sistemas basados en analizadores sintácticos superficiales se utilizan algoritmos de programación dinámica para garantizar la coherencia de los etiquetados de los clasificadores de detección de comienzos y de finales de roles semánticos, también ahora se emplean técnicas similares. Para ello, se puede optar por que el clasificador de argumentos no

genere una salida única, esto es, una etiqueta para cada uno de los candidatos, sino una serie de posibles etiquetas. Este conjunto de etiquetas posibles puede ir acompañado además por otra información, tal como la probabilidad estimada por el clasificador para cada una de las etiquetas que forman el conjunto. Posteriormente, utilizando técnicas de programación dinámica que tengan en cuenta las restricciones en cuanto al *frameset* y la no duplicidad de roles expuesta anteriormente, se genera un etiquetado con coherencia global.

Esta fase, y en menor medida la anterior, son las más críticas del sistema. Todo el trabajo previo es importante, pero es en el diseño de estas fases, los algoritmos de aprendizaje escogidos, la elección de las características de entrada a los mismos, en donde los distintos grupos de trabajo consiguen arrancar unas décimas de rendimiento de unos sistemas a otros. Y entran en juego aquí diseños cada vez más enrevesados desde el punto de vista de la minería de datos. A lo largo del tiempo, se observa la utilización de algoritmos de aprendizaje cada vez más complejos. Si inicialmente se utilizaban modelos probabilísticos basados en estimadores de máxima verosimilitud, actualmente se imponen en cuando a resultados obtenidos los sistemas basados en *Support Vector Machines*, a costa de una complejidad conceptual y temporal sensiblemente superior. Así mismo, muchos grupos de trabajo optan por la construcción de n modelos distintos, usando quizás distintos algoritmos de aprendizaje, que trabajan en paralelo generando n etiquetados distintos. O de forma análoga, n analizadores sintácticos distintos (o las n mejores producciones de un analizador sintáctico en concreto) para generar distintos vectores de características con los que estimar de nuevo n etiquetados distintos. Todas estas tendencias a atacar el problema desde distintos ángulos para posteriormente decidir entre los distintos resultados implican una fase final de elección entre los distintos etiquetados propuestos, que será descrita a continuación.

4.1.5 Inference

En algunos casos, sobretodo en los sistemas más recientes, la salida de la fase de clasificación de argumentos consiste en un conjunto de posibles etiquetados distintos, en lugar de un único etiquetado final. En un intento de conseguir mejorar los sistemas llevando al máximo los artificios relacionados con el aprendizaje automático, la tendencia ha sido la construcción de distintos subsistemas que producen distintas salidas, las cuales son combinadas en una última fase que podríamos llamar de inferencia. Se toma en dicha fase una decisión para obtener así una salida final. Los distintos subsistemas pueden ser replicas de uno mismo utilizando distintos analizadores sintácticos a la entrada, o bien los n árboles mejores generados por un analizador concreto. También se pueden formar distintos subsistemas aplicando varios algoritmos de aprendizaje a una misma entrada. Con todo ello, se pretende conseguir sistemas más robustos frente a los eventuales fallos de los analizadores sintácticos, y explotar las cualidades de los distintos algoritmos de aprendizaje.

Una vez generadas las distintas tentativas, la elección entre una u otra se lleva a cabo de diversas formas según los sistemas en que nos fijemos: algoritmos voraces, programación lineal entera, programación por restricciones, o incluso esquemas basados de nuevo en clasificadores automáticos (*stacking*). Todos ellos tratan de tener en consideración características globales de la oración a considerar, ya sea mediante la codificación directa de reglas linguísticas que se pretenden hacer prevalecer, o mediante la utilización de dichas características globales en la construcción del vector de entrada al clasificador en caso de haber optado por la técnica de *stacking*.

Los sistemas que utilizan varios subsistemas y una fase final de inferencia son los que actualmente consiguen mejores resultados, pero también es cierto que la mejora que consiguen es muy poco considerable con respecto a sistemas basados en un único modelo. Una de las principales críticas que se leen en artículos recientes se refiere precisamente a la creciente complejidad de los sistemas aparecidos frente a las mínimas mejoras, lo que plantea la necesidad de encontrar cambios de enfoque más radicales. Uno de los principales lastres de los sistemas actuales es el uso de una arquitectura serie o *pipeline*, en la que la salida de cada fase sirve de entrada a la siguiente. El uso de este tipo de arquitectura es muy común en la mayoría de las tareas abordadas en la disciplina del Procesamiento del Lenguaje Natural, ya que permiten descomponer problemas complejos en subproblemas más fácilmente abordables. Sin embargo, la desventaja es que los errores cometidos en la resolución de cada uno de los subproblemas se van arrastrando a lo largo de toda la cadena. En el caso que nos ocupa, los errores cometidos por los analizadores sintácticos, por ejemplo, son introducidos en la cadena y amplificados a lo largo de las distintas fases. Se necesitan por tanto arquitecturas radicalmente distintas en la que la división en subproblemas no sea tan rígida y en la que las distintas fases se retroalimenten unas a las otras. Será éste otro de los puntos planteados como líneas de investigación futura en este proyecto investigador.

4.2 Descripción de las características

Según se enuncia en la teoría del nexo o *linking theory*, la realización sintáctica de los argumentos de un predicado es predecible a partir de la semántica. Si esto es así, es razonable pensar que es posible aprender a reconocer las relaciones semánticas entre los constituyentes de una oración a partir de información sintáctica y léxica. Este fue el razonamiento seguido por Gildea y Jurafsky en su artículo [17] para definir una serie de características extraídas a partir de información sintáctica y léxica del texto a etiquetar semánticamente.

Como se verá cuando sean descritas en el siguiente apartado, dichas características son sorprendentemente simples. Resulta bastante increíble que a partir de tales *features*, con una arquitectura del sistema relativamente simple y a partir de un corpus con poca extensión y por tanto con poca entidad probabilística, el sistema en cuestión llegase a identificar y clasificar correctamente

con una precisión y un *recall* en torno al 60%.

Han sido dos las vías principales por las cuales los grupos de trabajo han tratado de mejorar esos resultados iniciales. La primera ha sido construyendo arquitecturas del sistema cada vez más avanzadas y complejas, con nuevos algoritmos de aprendizaje con más capacidad de generalización, varias fases anteriores y posteriores a las de identificación y clasificación de argumentos, varias etapas paralelas que generan distintos resultados que posteriormente son combinados en una última fase de inferencia, ... Y por el otro lado, se han ido perfeccionando las características básicas propuestas en el artículo original, y añadiendo otras que tratan de capturar más y mejor información léxicosintáctica , y en algunos casos con una visión más global. Teniendo en cuenta el nivel de complejidad al que están llegando las arquitecturas de los últimos sistemas, y la poca mejora que se consigue a pesar de ello, la investigación de nuevas y mejoradas características a medir se revela como uno de los caminos por explorar en un futuro cercano; existen ya artículos que se centran en realizar un análisis crítico a las características utilizadas actualmente por la comunidad científica y en proponer algunas nuevas, como por ejemplo [50].

En el siguiente apartado se describen las características originales propuestas por Gildea y Jurafsky, que se han mantenido como básicas hasta ahora. Posteriormente, se describirán algunas nuevas características aportadas por artículos más actuales y que han demostrado ser una aportación al estado del arte de los etiquetadores de roles semánticos.

4.2.1 Características básicas

Palabra del predicado (*Predicate Word*)

La palabra que evoca la clase semántica, a partir de la cual se decide en qué marco semántico nos encontramos, y cuáles son los roles semánticos a identificar, conocida como *predicate* en PropBank y como *target word* en FrameNet, y que generalmente será un verbo, es la primera de las características a ser utilizadas por el clasificador. Los primeros sistemas utilizaban simplemente el infinitivo de dicho verbo. Posteriormente, otros sistemas han utilizado también la forma lexicalizada tal como aparece en la oración considerada, o incluso la categoría morfosintáctica del predicado (en aquellos sistemas que tienen en cuenta la existencia de clases semánticas evocadas por nombres o adjetivos en lugar de por verbos).

Esta medida es constante para todos los constituyentes de una oración, y no es especialmente discriminativa en la fase de identificación de argumentos, tal como se desprende de los experimentos presentados en el artículo [50].

Tipo de sintagma (*Phrase Type*)

Esta característica consiste en indicar la categoría sintáctica del constituyente en cuestión, de entre las especificadas en PropBank: sintagma nominal (NP),

sintagma preposicional (PP), adverbios (ADVP), partículas (PRT), y cláusulas (SBAR y S). La idea es que distintos roles semánticos tienden a estar expresados por diferentes categorías sintácticas. Así que es evidente que esta característica le proporciona información útil al clasificador de argumentos. También es útil en la fase de identificación de argumentos, pues por ejemplo una partícula (PRT) es mucho menos probable que sea un rol semántico que un sintagma nominal (NP).

Categoría regente (*Governing Category*)

Esta característica sólo se aplica a aquellos constituyentes que sean sintagmas nominales (NP), indicando si se trata de un sujeto u objeto del verbo (S o VP). Es evidente la correlación existente entre determinados roles semánticos y esta característica. Por ejemplo, si usamos PropBank, y estando en una frase en activa, en la mayoría de los casos el rol A0 estará desempeñado por un sintagma nominal que funciona de sujeto del verbo.

Para calcular esta característica, se parte del nodo del árbol del constituyente que estemos considerando, y se va subiendo en la jerarquía hasta llegar a un nodo S o a uno VP. Existen casos en los que esta manera de proceder no funciona del todo bien. En ocasiones, algunos sintagmas nominales serán considerados objetos del verbo cuando no lo son. Por ejemplo, en la frase *I arrived yesterday*, el sintagma nominal *yesterday* no es realmente el objeto del verbo, que en este caso funciona como intransitivo. De todas formas, el clasificador se puede apoyar en otras características para detectar casos como el presentado.

Esta medida no aporta ningún poder de discriminación para la etapa de identificación de argumentos, como se señala en el artículo [50].

Camino en el árbol de análisis (*Parse Tree Path*)

Con esta medida se trata de capturar la relación sintáctica entre el predicado (*target word* en FrameNet) y el constituyente que estemos tratando de clasificar. Se trata de representar el camino en el árbol sintáctico desde el nodo del predicado (generalmente, el verbo) hasta el nodo que representa al constituyente en cuestión. Esta representación se realiza mediante una cadena de texto que comienza con la categoría morfosintáctica del predicado, y continúa con cada uno de los nodos del árbol hasta llegar al constituyente a clasificar por el camino más corto, separando cada dos nodos por un símbolo que indique si estamos subiendo o bajando en el árbol sintáctico (ver figura 4.4). Las etiquetas morfosintácticas utilizadas para verbos en sus distintas formas (gerundios, participios, presente, pasado, ...) son reducidas a una sola etiqueta VB, ya que así se minimiza el número de posibilidades y se demuestra experimentalmente que se consigue una mejor generalización.

La cadena así formada es tratada como si de un valor atómico se tratase, de manera similar a si tuviésemos un tipo de datos enumerado que contuviese todos los posibles caminos de nuestro árbol. En principio puede parecer que existen infinitas combinaciones, pero si se realiza un pequeño estudio obtendremos un

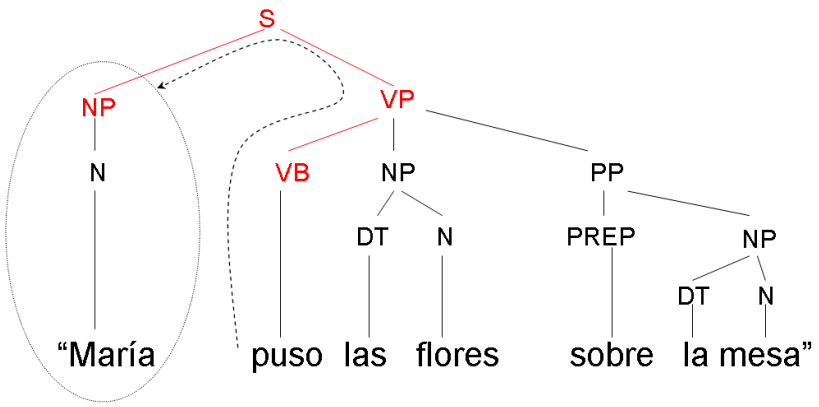


Figura 4.4: En este ejemplo, se refleja el cálculo de la característica *Parse Tree Path* para el constituyente sintagma nominal *María*. La cadena resultante sería VB↑VP↑S↓NP

conjunto perfectamente delimitado de posibles caminos, a cada uno de los cuales asignaremos un valor numérico determinado.

Esta característica ha demostrado ser efectiva tanto en la fase de identificación como en la de clasificación de argumentos. Para la fase de clasificación, la característica anterior (*Governing Category*) ha demostrado ser prácticamente igual de informativa que la que nos ocupa, siendo mucho más simple. Sin embargo, para la fase de identificación, esta medida sí que aporta información adicional. En cierto modo, la cadena que se forma recoge información relacionada con la función gramatical que desempeña el constituyente. Por ejemplo, el camino VB↑VP↑S↓NP indica que el constituyente es un sujeto, de forma similar a como se indicaba con la característica *governing category*, con la diferencia de que la que nos ocupa es mucho más expresiva.

Evidentemente, para la utilización de esta característica es necesario utilizar un analizador sintáctico completo, por lo que los sistemas que utilizan analizadores superficiales no pueden hacer uso de la misma.

Subcategorización (*Subcategorization*)

Esta característica está muy relacionada con la anterior. En esta ocasión, se intenta codificar la estructura sintáctica del sintagma verbal de la frase, independientemente de qué constituyente estemos tratando de etiquetar. La idea es que esta información ayudará a decidir en casos en los que distintos roles se pueden asignar a la misma posición sintáctica para un mismo verbo. Por ejemplo, en las oraciones *He opened the door* y *The door opened*, el mismo verbo es capaz de llevar como sujeto al agente que realiza la acción, y en la segunda al objeto paciente que recibe la acción.

Para codificar la estructura del sintagma verbal, se parte del nodo etiquetado como VP en el árbol sintáctico, y se escribe en forma de producción a qué hijos pa lugar el nodo (ver figura 4.5). En el ejemplo, el valor "VP → VB NP" indica que el verbo está funcionando posiblemente de manera transitiva, y en el segundo caso el valor "VP → VB" indica un funcionamiento intransitivo del mismo.

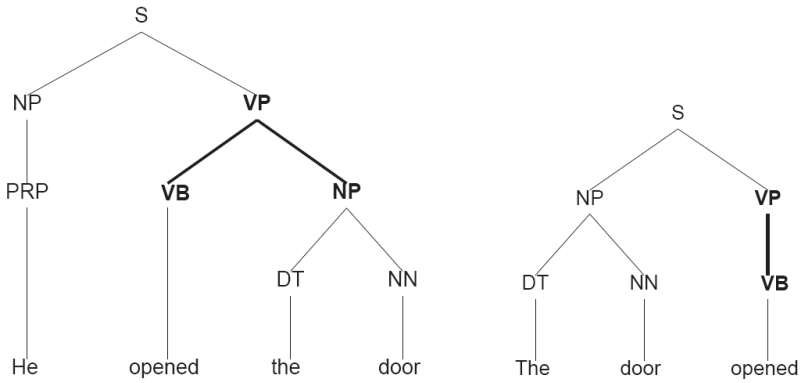


Figura 4.5: Dos ejemplos de cálculo de la característica de *subcategorization* para el predicado *open*.

Posición (*Position*)

Consiste simplemente en indicar si el constituyente que estamos tratando de clasificar aparece *antes* o *después* del predicado (generalmente, del verbo). A pesar de su sencillez, esta característica ofrece una clara correlación con la función gramatical del constituyente, sin basar su obtención en la información (potencialmente errónea) del analizador sintáctico. Es por esta independencia frente a los fallos del analizador por lo que esta medida se ha mantenido prácticamente en todos los sistemas actuales analizados para este trabajo.

De forma similar a la característica *governing category*, esta medida no es útil para la fase de identificación de argumentos, como se recoge en los experimentos de Xue&Palmer [50].

Voz (*Voice*)

Se trata de identificar si la oración actual está en forma activa o pasiva. Para ello se utilizan un conjunto de patrones a reconocer para cada una de las formas, que han demostrado ser fiables. La idea es que saber si una oración es activa o pasiva es el complemento necesario para que otras características relacionadas con las funciones gramaticales, como las de *governing category*, *parse tree path* o *position*, puedan ser correctamente interpretadas por el clasificador.

Esta medida se mantendrá por supuesto constante para todos los constituyentes de una oración. Señalar que, al igual que se ha comentado para otras características anteriores, esta medida no ofrece utilidad en la fase de identificación de argumentos, sólo en la de clasificación.

Palabra principal (*Head Word*)

Todas las características descritas anteriormente hacen uso de la información sintáctica y/o gramatical de la oración, a partir de los enunciados de las teorías del nexo. Pero se hace patente también la existencia de dependencias entre el léxico y el rol semántico que desempeña un constituyente. Por ejemplo, si la oración actual corresponde a un *frameset* evocado por el verbo *send*, un sintagma nominal cuyo núcleo sea la palabra *carta* casi con toda probabilidad estará ocupando el rol semántico que esté definido en el *frameset* para el *objeto que es enviado*. Esta dependencia léxica se trata de capturar mediante la característica *head word*, que consistirá en la palabra que funciona de núcleo del constituyente que estemos clasificando.

En principio, la palabra se utiliza tal cual como valor para la característica, esto es, tal como venga lexicalizada (con su género, su número, ...). Aún así, también se pueden añadir como variaciones de esta característica la categoría morfosintáctica de la palabra o la raíz de la misma.

4.2.2 Otras características

A continuación se describen otras características propuestas por artículos posteriores al de Jurafsky y Gildea [17].

Entidades en los constituyentes

En realidad son un conjunto de características binarias, una por cada tipo de entidad considerado en el sistema, de manera que cada característica indica si dentro del constituyente en cuestión se encuentra o no una entidad del tipo considerado. Tras ser utilizada esta característica en [44], ha pasado a ser considerada un estándar desde entonces. De hecho, en la competición organizada por el CoNLL-2005 se incluía una columna con el resultado de aplicar el reconocimiento de entidades al corpus de entrenamiento.

Categoría morfosintáctica de la palabra principal del constituyente

Esta característica también fue presentada por el mismo artículo que la anterior [44], y desde entonces también ha pasado a ser aceptada como un estándar.

Palabra principal para sintagmas preposicionales

En los sintagmas preposicionales, la palabra que actúa de núcleo es la preposición, la cual en la mayoría de los casos no aporta demasiada información al

clasificador. Por ejemplo, en los sintagmas *in the city* y *in a few minutes*, la característica *head word* sería *in* en ambos casos. La característica que nos ocupa sin embargo valdría *city* en el primer caso y *minutes* en el segundo, aportando mucha más información al clasificador.

Primera y última categoría morfosintáctica del constituyente

También esta información parece ayudar a los clasificadores en el etiquetado de roles semánticos, como complemento a la palabra principal del constituyente. Las categorías morfosintácticas de dichas palabras también son utilizadas en determinados sistemas.

Distancia con respecto al predicado

Se indica con un número la distancia en constituyentes y/o en palabras desde el constituyente actual al predicado. De esta manera, entre otras cosas, aquellos constituyentes muy alejados del predicado son más fácilmente descartados como roles semánticos del mismo.

Marco sintáctico

Esta característica fue planteada en [50]. La idea es complementar a las características que indican el camino en el árbol hasta el predicado y la subcategorización. Se definen los sintagmas nominales y el predicado como pivotes, de forma que para cada constituyente la característica que se describe consiste en la sucesión de sintagmas nominales alrededor del predicado, indicando en mayúsculas aquel sintagma nominal que contiene (o que es en si mismo) al consituyente actual. Por ejemplo, en el árbol que puede verse en la figura 4.6, la característica para el constituyente *state* valdría *np_v_NP_np*, mientras que para el constituyente *more leeway to restrict abortions* sería *np_v_np_NP*. Esta característica ha demostrado ser un buen aporte a los sistemas de etiquetado de roles semánticos y ha pasado a ser un estándar en la literatura.

4.3 Rendimiento actual de los etiquetadores de roles semánticos estadísticos

A continuación se muestran los resultados obtenidos por los 4 mejores sistemas presentados a la competición del CoNLL-2005 [6] (punyakanok [37], haghighi [1], marquez [25], pradhan [40]) . Los resultados de todos ellos se muestran en la primera tabla al aplicarlos sobre el corpus de test, que está extraído de noticias del *Wall Street Journal*, esto es, pertenece a la misma categoría de textos que el corpus de entrenamiento, que en esta competición consistió en el corpus PropBank. En la segunda tabla aparecen los resultados obtenidos al aplicar los etiquetadores a otro tipo de textos cuya temática no era conocida a priori por los participantes en la competición.

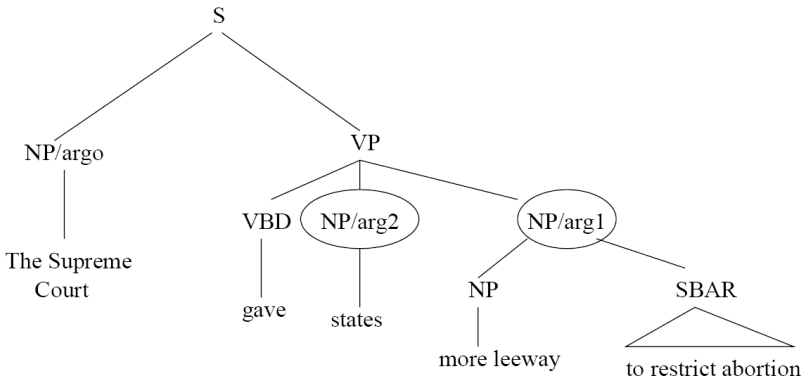


Figura 4.6: Característica *Marco Sintáctico*

	Precision	Recall	F1
punyakanok	82.28	76.78	79.44
haghighi	79.54	77.39	78.45
marquez	79.55	76.45	77.97
pradhan	81.97	73.27	77.37

Tabla 4.1: Mejores resultados en el CoNLL-2005 Shared Task sobre corpus WSJ

	Precision	Recall	F1
punyakanok	73.38	62.93	67.75
haghighi	70.24	65.37	67.71
marquez	70.79	64.35	67.42
pradhan	73.73	61.51	67.07

Tabla 4.2: Mejores resultados en el CoNLL-2005 Shared Task sobre corpus Brown

Como se puede ver en la tabla, los sistemas presentados en 2005, y que hasta ahora no han sido mejorados en rendimiento, se quedan a las puertas de un 80% de acierto (F_1). Todos los sistemas que aparecen en esta tabla son sistemas que hacen uso de varios clasificadores o varias versiones de los datos de entrada para construir varios modelos cuyas salidas son posteriormente combinadas en una etapa de *inference*. Por tanto, este tipo de estrategia se muestra como la más efectiva actualmente para abordar la construcción de etiquetadores de roles semánticos.

Aunque un 80% de acierto aún queda lejos de ser un resultado aceptable y que permita utilizar estos sistemas en aplicaciones como las comentadas en el presente informe, el dato más alarmante es el que nos brinda la tabla con los resultados al aplicar los etiquetadores a un corpus de otra naturaleza a la utilizada en el entrenamiento. En estos casos el rendimiento cae hasta el 67% aproximadamente. Esta espectacular caída fue en su momento causa de una gran desilusión en la comunidad que abordaba el etiquetado de roles semánticos. Se plantearon posibles causas para estos malos resultados; por un lado la necesidad de contar con más recursos etiquetados semánticamente, que recojan una cantidad mayor de tipos de textos; por otro lado, la arquitectura tipo *pipeline* utilizada, en la cuál la salida de cada etapa actúa como entrada a la siguiente, puede llevar pareja la acumulación de errores de las distintas fases. Esto es importante sobre todo en lo relacionado con el análisis sintáctico, que tal como se ha visto en este trabajo no está exenta de errores. Dichos errores podrían estar siendo encubiertos en cierto modo al mantenerse constantes para un determinado tipo de textos, y sin embargo constituir un elemento importante a tener en cuenta cuando se aplican los etiquetadores construídos a otra clase de textos. Para solucionar esto, los organizadores de la tarea compartida del CoNLL-2005 plantean la necesidad de estudiar arquitecturas alternativas en las que se rompa la dependencia hacia un solo lado del sistema, y que permitan la retroalimentación de las distintas etapas de la arquitectura entre sí.

Capítulo 5

Proyecto Investigador

5.1 Trabajos anteriores

5.1.1 Primeros contactos con el Procesamiento del Lenguaje Natural

Mi primer contacto con la investigación en el área del Procesamiento del lenguaje natural tuvo lugar durante la realización de una beca de estudiante en el Centro de Tecnologías del Lenguaje de IBM en Sevilla. Esta experiencia la desarrollé durante el transcurso de mi último año de estudios en Ingeniería Informática. En el centro en cuestión se realizaban tareas relacionadas principalmente con el reconocimiento automático del habla y con la síntesis de voz. En el transcurso de la beca tomé contacto con tareas como el etiquetado morfosintáctico, algoritmos de aprendizaje como los modelos ocultos de Markov y los árboles de decisión, los modelos del lenguaje, ... También realicé una implementación del algoritmo de Brill basado en transformaciones para el etiquetado morfosintáctico, en su versión no supervisada [4]. A pesar de no obtener ninguna publicación de la realización de la beca, me sirvió de punto de contacto con el área al que actualmente me dedico, y me inspiró para realizar el proyecto fin de carrera basado en un sistema de diálogo utilizando VoiceXML.

En este primer contacto con el Procesamiento del Lenguaje Natural descubrí las enormes dificultades que plantea trabajar con el lenguaje y la existencia de multitud de retos aún por resolver en el área. Posteriormente, cuando comencé mi periodo docente en el programa de doctorado del departamento de Lenguajes y Sistemas Informáticos de la Universidad de la Sevilla, asistí al curso sobre Procesamiento del Lenguaje Natural, donde tomé contacto con el grupo de investigación ITALICA del departamento en cuestión, que trabaja en este área. Al mismo tiempo, realicé otra beca por periodo de un año en el mismo centro de IBM, esta vez ocupándome de desarrollar un trabajo relacionado con la puntuación de textos basándome en información acústica, enmarcado dentro

de un proyecto sobre traducción automática subvencionado por la Unión Europea (TC-STAR). En este periodo seguí descubriendo nuevos campos de investigación, y realicé una implementación del algoritmo C.45 para la construcción de árboles de decisión para llevar a cabo el trabajo anteriormente comentado. Adquirí así experiencia en la manera de trabajar en problemas de etiquetado, la construcción de características, y en general el uso de clasificadores para afrontar tareas del Procesamiento del Lenguaje Natural desde una perspectiva estadística.

5.1.2 Técnica de *stacking* aplicada al reconocimiento de entidades

En el mismo año comienzo a trabajar con el grupo de investigación ITALICA, bajo la supervisión del Dr. D. Jose Antonio Troyano Jiménez, en las investigaciones que en ese momento estaba llevando a cabo relacionadas con la aplicación de técnicas de *stacking* al problema del reconocimiento de entidades. El *stacking* consiste en la aplicación de algoritmos de aprendizaje automático a las salidas proporcionadas por distintos modelos, de manera que el sistema aprende cuando un modelo acierta o se equivoca, y el resultado final es previsiblemente mejor que los resultados parciales proporcionados por cada uno de los modelos participantes. En el transcurso de estas investigaciones, que desembocan en la publicación de un artículo en Eurocast en 2005 [45], empleamos la herramienta WEKA, que implementa una serie de algoritmos de aprendizaje basados en árboles de decisión y en tablas de reglas. La tarea que se pretende abordar es el reconocimiento de entidades, entendido como un problema de etiqueta con notación IOB. En realidad, y puesto que el foco principal del trabajo es estudiar las ventajas del uso de *stacking* en tareas de etiquetado lingüístico, sólo se afronta la identificación de las entidades, y no su categorización. Por ejemplo, un trozo del corpus de entrenamiento utilizado se muestra en la tabla 5.1.

Word	Tag
La	O
Delegación	B
de	I
la	I
Agencia	I
EFE	I
en	O
Extremadura	B
transmitirá	O
hoy	O
...	...

Tabla 5.1: Ejemplo de notación IOB para el reconocimiento de entidades.

A este corpus inicial se le realizan distintas transformaciones, con cada una de las cuales se construye un etiquetador basado en modelos ocultos de Markov, en concreto se utilizó la herramienta TnT [3]. Las transformaciones consideradas son las siguientes:

Reducción de vocabulario : se sustituyen las palabras del texto que comienzan por mayúsculas por un token único que viene a señalar esta característica. Las palabras formadas en su totalidad por letras mayúsculas se cambian también por otro token especial.

Añadido de información morfosintáctica : se añade a cada una de las palabras que aparecen en el corpus de entrenamiento un trozo de texto que codifica la categoría morfosintáctica de la palabra.

Añadido de etiquetas : se etiquetan las palabras anteriores y posteriores a una entidad con etiquetas especiales. También se marcan con una etiqueta especial las palabras contenidas entre dos entidades.

Word	Tag
La	O
_starts_cap_	B
de	I
la	I
_starts_cap_	I
_all_cap_	I
en	O
_starts_cap_	B
transmitirá	O
lower	O
...	...

Reducción de vocabulario.

Word	Tag
La	O
Delegación	B
de	I
la	I
Agencia	I
EFE	E
en	O
Extremadura	BE
transmitirá	O
hoy	O
...	...

Añadido de información morfosintáctica.

Word	Tag
La_det_	O
_starts_cap__noun_	B
de_prep_	I
la_det_	I
_starts_cap__noun_	I
_all_cap__noun_	I
en_prep_	O
_starts_cap__noun_	B
transmitirá_verb_	O
_lower__adv_	O
...	...

Añadido de etiquetas.

Tabla 5.2: Las tres transformaciones generadas a partir del corpus inicial

Con el corpus original y los transformados se generan cuatro modelos, que son posteriormente combinados mediante *stacking*, utilizando árboles de decisión. El resultado final obtenido ($F_{\beta=1} = 84.43\%$) es comparable a los mejores resultados conseguidos para la tarea en cuestión en la competición realizada en el *CoNLL* del 2002.

El acercamiento al problema del reconocimiento de entidades, que es una tarea de etiquetado semántico superficial, fue el punto inicial que me llevaría mas adelante a la tarea que me ocupa actualmente. El mismo congreso que organizó en 2002 una competición sobre reconocimiento de entidades, organizaría en 2004

y 2005 otra sobre etiquetado de roles semánticos, y fue ésta precisamente la manera en que tomé contacto con esta tarea.

5.1.3 Grupo de investigación Julietta

Tras los cursos de doctorado y la finalización de la beca en IBM, trabajé durante un único mes para el grupo de investigación Julietta a cargo del doctor D. Jose Gabriel de Amores Carredano, del departamento de Filología Inglesa de la Universidad de Sevilla. Este grupo se dedica al igual que el grupo ITALICA al Procesamiento del Lenguaje Natural, con una visión más centrada en la lingüística computacional. La estancia en el grupo fue breve debido a que al poco tiempo entré a ocupar una plaza de profesor asimilado a ayudante en el departamento de Lenguajes y Sistemas Informáticos de la Universidad de Sevilla. A pesar de ello, tuve conocimiento de las investigaciones a las que se dedicaban, principalmente centradas en el desarrollo de sistemas de diálogo, y espero poder colaborar con ellos en un futuro ya que mi tema de tesis es muy aplicable a dichos sistemas de diálogo.

5.1.4 TextRank supervisado

Una vez ocupando puesto de profesor en el departamento e incluido en el grupo de investigación ITALICA, inicio una línea de investigación junto a Jose Antonio Troyano sobre la aplicación del algoritmo de PageRank [36] a tareas de etiquetado. El algoritmo de PageRank es el utilizado por el motor de búsqueda de Google para decidir la relevancia de las páginas web sobre las que se realizan las búsquedas. La idea principal subyacente en el mismo es que una página web que es enlazada desde otra página web que previamente se considera importante, recibe un voto para ser considerada asimismo como una página relevante.

La inspiración para esta investigación vino a partir de un artículo de Rada Mihalcea en el que aplica dicho algoritmo a problemas lingüísticos como el resumen automático de textos o la búsqueda de palabras clave [39]. En dicho artículo, se aplica una variación del algoritmo PageRank al que denomina *TextRank*. Para aplicarlo, primeramente se define un grafo que modela el problema a resolver, buscando qué elementos del problema serán los nodos y cómo se construyen las aristas. Posteriormente se le aplica el algoritmo de PageRank (modificado para permitir aristas con pesos) al grafo en cuestión, lo cuál asigna a cada nodo una puntuación que refleja la importancia del nodo en la red en función de la topología de la red. Se utiliza entonces dicha puntuación para resolver el problema, escogiendo generalmente los n nodos con mayor puntuación como solución.

Por ejemplo, para la resolución del resumen automático mediante TextRank, se modela un grafo en el que cada frase constituye un nodo, y existen aristas entre cada par de nodos. Cada arista se pondera con un peso que mide la similitud entre las frases: dos oraciones con las mismas palabras en el mismo orden tendrán una similitud de 1, mientras que dos frases con ninguna palabra común

obtendrán una puntuación nula de similitud (antes de computar la similitud, se eliminan palabras consideradas inútiles, como determinantes, preposiciones o adverbios). Se aplica posteriormente el algoritmo de TextRank sobre el grafo formado, para posteriormente escoger las frases que han obtenido mayor puntuación (el número de frases a escoger como resumen será un parámetro de entrada al sistema). Con este acercamiento sin supervisión y de sencilla implementación Rada Mihalcea consiguió resultados similares a otros sistemas de resumen automático mucho más sofisticados y que hacían uso de grandes corpus de entrenamiento.

El algoritmo PageRank y consecuentemente TextRank calcula la importancia de cada nodo de un grafo de manera que aquellos nodos con aristas entrantes procedentes de un nodo determinado reciben un aumento en su puntuación proporcional a la puntuación del nodo del que procede la arista, e inversamente proporcional al número de aristas que salen de dicho nodo. Esto queda reflejado en la siguiente expresión, versión TextRank, que tiene en cuenta además los pesos de las aristas, de manera que por ejemplo una arista con peso igual a 2 equivale a la existencia de dos aristas en la versión del algoritmo original (en la que no se tenían en cuenta la existencia de pesos):

$$P(V_i) = (1-d) + \ d \sum_{j \in E(V_i)} \frac{p_{ji}}{\sum_{k \in S(V_j)} p_{jk}} \ P(V_j)$$

El valor de la constante d suele ser 0.85. Se comienza asignando un valor cualquiera a los nodos del grafo a puntuar, y posteriormente se aplica la fórmula a cada uno de los nodos. Se repite el cálculo de la puntuación hasta que en uno de los pasos la variación entre las puntuaciones inicial y final de los nodos sea menor a un valor umbral introducido como parámetro de entrada al algoritmo.

Los resultados obtenidos en ciertas aplicaciones del Procesamiento del Lenguaje Natural por el algoritmo de TextRank, sin necesidad de llevar a cabo un proceso de entrenamiento supervisado, nos llevó a plantearnos la experimentación de estrategias para utilizar información extraída de un proceso de entrenamiento previo para la resolución de otras tareas de Procesamiento del Lenguaje Natural. La tarea escogida fue el etiquetado morfosintáctico, principalmente por la disponibilidad de corpus etiquetados gratuitos. El proceso de entrenamiento consistió en realizar a partir del corpus de entrenamiento los conteos de unigramas, bigramas y trigramas necesarios para realizar las estimaciones de máxima verosimilitud de las probabilidades de emisión y transición propias de los etiquetadores basados en modelos ocultos de Markov (ver apartado *Modelos ocultos de Markov* en el capítulo *Introducción al Procesamiento del Lenguaje Natural*). Una vez hecho esto, y dada una frase a etiquetar, se construye un grafo generando un nodo por cada palabra y posible etiqueta de la frase. Los nodos correspondientes a cada palabra se unen con los nodos de

las palabras contiguas, y dichas aristas se ponderan con las probabilidades de emisión y transición (ver figura 5.1). Después se lleva a cabo el algoritmo de TextRank y se selecciona para cada palabra la etiqueta correspondiente al nodo mejor puntuado para la palabra en cuestión.

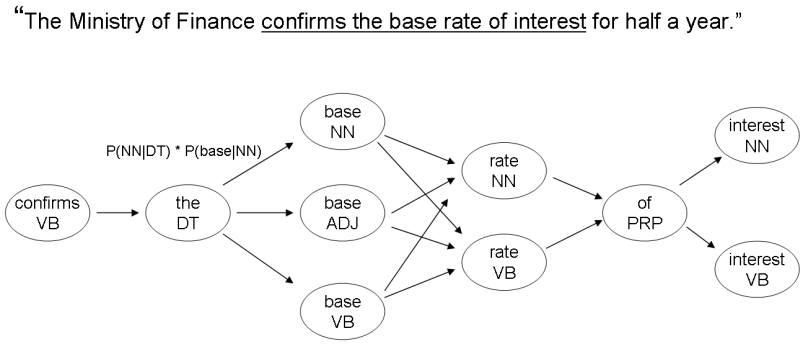


Figura 5.1: Ejemplo de construcción de grafo para etiquetado morfosintáctico al que se aplicará TextRank

Se experimentó también con algunas variaciones en la construcción del nodo que no serán aquí explicadas pero que pueden consultarse en los artículos [11] y [12]. Con las distintas construcciones del grafo, se generan varios modelos que son combinados mediante *stacking*, aplicando lo aprendido en el trabajo anterior sobre reconocimiento de entidades (ver figura 5.2). Tras todo esto, se consiguen resultados similares,y en algún caso superiores, a otros métodos de etiquetado morfosintáctico (ver tabla 5.3) . Se probó también la aplicación del método desarrollado a otras tareas de etiquetado como el reconocimiento de entidades y el *chunking* o analizador sintáctico superficial. En todas ellas se consiguieron resultado próximos a los mejores conseguidos por otros etiquetadores (ver tabla 5.4) . La intuición obtenida tras el desarrollo de este trabajo es que la aplicación de TextRank no está reñida con la utilización de información proveniente de una fase previa de entrenamiento, y que sería de interés estudiar su aplicación a otras tareas de Procesamiento del Lenguaje Natural. Para ello, se encuentra actualmente en fase de desarrollo una herramienta en JAVA que permitirá definir mediante un sencillo lenguaje la topología de los grafos a construir y aplicar el algoritmo de TextRank supervisado a los mismos. La idea es poner dicha herramienta a disposición de la comunidad para la utilización de las ideas aquí explicadas a cuantos más trabajos mejor.

5.1.5 Ampliación automática de corpus

En paralelo a la realización de este trabajo, colaboré también con Fernando Enríquez, compañero del grupo ITALICA, en sus investigaciones sobre la ampliación automática de corpus [14]. El auge actual del enfoque estadístico en la

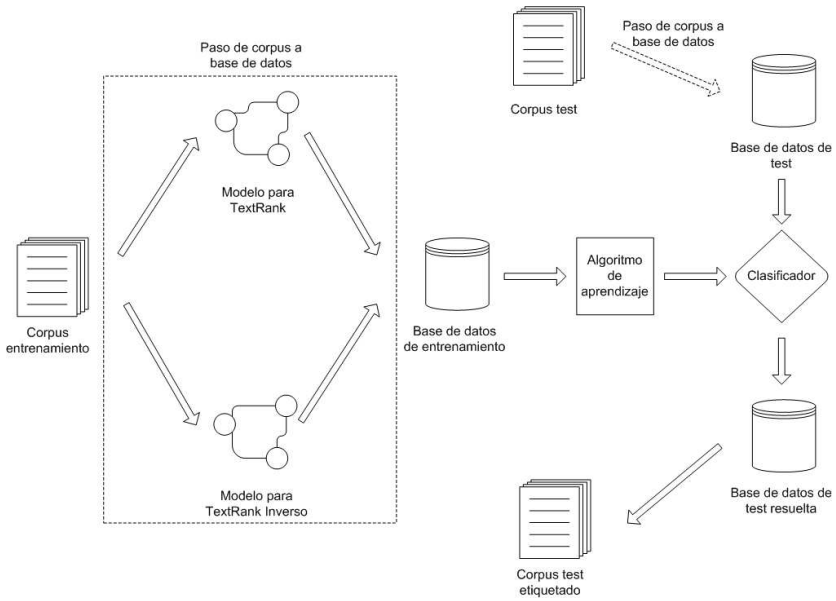


Figura 5.2: Combinación mediante *stacking* de distintas propuestas de construcción del grafo para TextRank

resolución de problemas de Procesamiento del Lenguaje Natural pone de manifiesto la importancia de la disponibilidad de corpus de entrenamiento para las diferentes tareas. En general, este material es muy costoso de producir, y para algunas tareas de reciente aparición existen pocos recursos de entrenamiento disponibles aún. Por todo esto, se hace interesante estudiar mecanismos para conseguir ampliar automáticamente corpus etiquetados, a partir de un pequeño corpus ya etiquetado denominado semilla. El acercamiento llevado a cabo en este trabajo consiste en aplicar en una primera fase la técnica de *co-training* [2], modificada mediante la aplicación de *stacking*. La técnica de *co-training* consiste en utilizar varios etiquetadores sobre el corpus semilla. Los modelos construidos son utilizados para etiquetar un conjunto nuevo de frases. Las frases etiquetadas por cada uno de los modelos son añadidas al corpus de entrenamiento del resto de etiquetadores, para posteriormente repetir el proceso de entrenamiento para todos los modelos. De esta manera, cada uno de los etiquetadores se ve enriquecido por la manera de etiquetar de los demás, consiguiéndose de esta forma ampliar el corpus de entrenamiento manteniendo una cierta calidad, al menos durante un número pequeño de iteraciones del método.

Nuestra propuesta añade una etapa de *stacking* que se encarga de decidir de entre el conjunto de frases etiquetadas por cada uno de los modelos cuáles se escogen para pasar a formar parte del corpus de entrenamiento (ver figura 5.3). Tras ser aplicado el método a la tarea de etiquetado morfosintáctico y al

	Susanne	Penn
Línea base	79.15%	80.01%
TnT	93.61%	95.48%
TreeTagger	85.91%	94.28%
fnTBL	93.01%	95.04%
MBT	91.16%	94.40%
MaxEnt	93.09%	95.47%
TextRank	90.32%	92.14%
TextRankI	89.84%	91.51%
TextRankC	91.51%	93.09%

Tabla 5.3: Resultados obtenidos en etiquetado morfosintáctico por algunos etiquetadores estándar y por nuestra propuesta TextRank, en sus dos versiones de construcción del grafo normal e invertida, y mediante *stacking* de las dos anteriores (TextRankC)

reconocimiento de sintagmas o *chunking*, los resultados son prometedores. En todos los casos, se consiguen resultados superiores a la utilización del *co-training* básico, especialmente en la tarea de etiquetado sintáctico superficial (ver tabla 5.5).

Estos resultados y algunos más en los que se experimenta con una simulación de una fase de participación de expertos en el sistema para la ampliación semiautomática de corpus están publicados en un artículo aceptado para el congreso Eurocast 2007 [13].

5.2 Escenario actual en la investigación sobre etiquetadores de roles semánticos

En este apartado se intentan condensar informaciones de interés para desarrollar mis expectativas de investigación y conseguir los contactos con otro grupos y publicaciones necesarias para avanzar hacia mi tesis doctoral.

5.2.1 Grupos de investigación

A continuación se enumeran algunos de los grupos de investigación cuyos investigadores son autores fundamentales de los artículos sobre etiquetado de roles semánticos. Forme esto parte de un trabajo de documentación previa que estamos llevando a cabo con idea de ponernos en contacto con algunos de ellos para futuras colaboraciones y , si es posible, llevar a cabo alguna estancia en sus universidades.

Cognitive Computation Group (*Universidad de Illinois*): dedicados a distintas áreas de la inteligencia natural, en lo relativo al Procesamiento del

	NER-E	NER-B	Chunk
Línea base	71.90%	72.64%	63.08%
TnT	94.78%	88.97%	89.62%
TreeTagger	90.58%	84.79%	84.40%
fnTBL	94.30%	90.49%	89.54%
MBT	94.38%	88.71%	90.61%
MaxEnt	95.03%	87.52%	92.83%
TextRank	92.72%	86.75%	87.34%
TextRankI	90.85%	87.78%	78.84%
TextRankC	92.93%	89.71%	89.24%

Tabla 5.4: Resultados para las tareas NER-E, NER-B y Chunk de algunos etiquetadores estándar y de nuestra propuesta TextRank.

	Semilla	*Co-training*	*Stacking*
TnT	76,5	70,18 (-6,32)	76,99 (+0,49)
TT	68,98	69,79 (+0,81)	72,71 (+3,73)
MBT	74,19	70,04 (-4,15)	76,14 (+1,95)

Tabla 5.5: Resultados tras la ampliación automática de recursos para el corpus CoNLL 2000 (*chunking*).

Lenguaje Natural es de destacar las distintas herramientas implementadas por miembros de este grupo de investigación, entre las que destaca SNoW, un paquete de aprendizaje automático que fue utilizado por Vasin Punyakanok y Dan Roth en sus artículos [37] y [47], ganando con ello el primer puesto en la competición organizada en el CoNLL 2005. Su sistema de etiquetado de roles semánticos puede ser probado *on-line* en la dirección *http://l2r.cs.uiuc.edu/ cogcomp/srl-demo.php*.

The Stanford Natural Language Processing Group (*Universidad de Stanford*): Uno de los grupos más importantes (si no el más) en todo lo relacionado con el Procesamiento del Lenguaje Natural. Sus dos componentes fundamentales, Chris Manning y Dan Jurafsky, son autores de sendos libros considerados como referencia obligada a todo aquel que desea iniciarse en la disciplina del Procesamiento del Lenguaje Natural (*Foundations of Statistical Natural Language Processing* [32] y *Speech and Language Processing* [20], respectivamente). En lo relativo al etiquetado de roles semánticos, Dan Jurafsky es uno de los autores del artículo que sirvió de pistoletazo de salida en el tema [17], además de haber seguido publicando artículos relacionados en los últimos años (quedó cuarto en la competición sobre etiquetado de roles semánticos organizada en el CoNLL 2005, como co autor de [40]). Jurafsky trabajó para el artículo [17] con

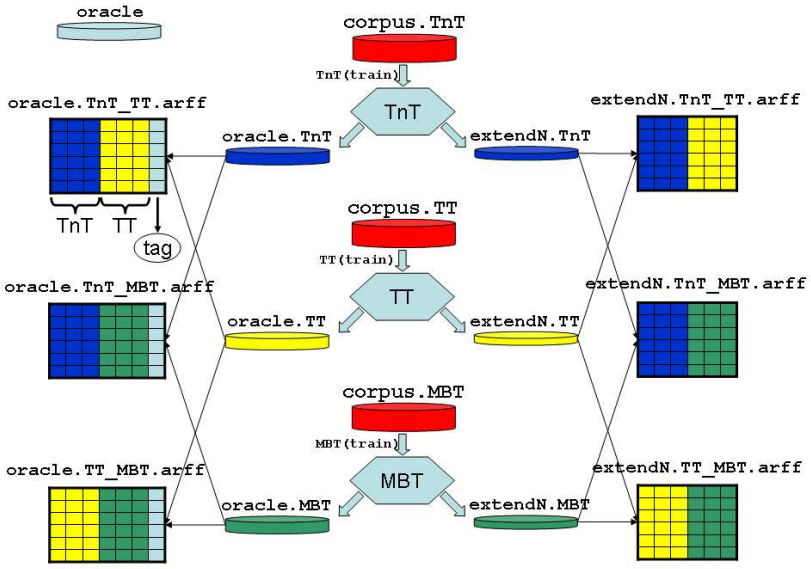


Figura 5.3: Arquitectura del método de ampliación de corpus basado en *co-training* y *stacking*.

Dan Gildea, que pertenece a la Universidad de Rochester. Ambos están además inmiscuidos en los proyectos FrameNet y PropBank, por lo que se puede considerar que están a la cabeza en todo lo relacionado con el etiquetado de roles semánticos.

NLP Research Group (*TALP Research Center, Universidad de Cataluña*): este grupo de investigación cuenta entre sus integrantes con los dos organizadores de las competiciones de 2004 y 2005 del CoNLL sobre etiquetado de roles semánticos: Lluis Márquez y Xavier Carreras. Lluis Márquez estuvo presente además como participante en la edición de 2005, en la que quedó tercero con su aproximación al etiquetado de roles semánticos como tarea de etiquetado secuencial [25], lo que puso en evidencia a algunos trabajos presentados cuya complejidad y grado de sofisticación no se correspondió con mejores resultados. Además, ambos forman parte del equipo invitado de editores que están preparando un número especial de Computational Linguistics sobre etiquetado de roles semánticos. Este número será publicado a finales del presente 2007, y es de esperar que aparezcan en él trabajos muy interesantes que traigan un poco de movimiento dentro de la investigación sobre el tema que nos ocupa, que en el último año parece haberse enfriado.

Computer Science Division (*Universidad Berkeley de California*): dentro

de este departamento, Dan Klein desempeña numerosos trabajos de Procesamiento del Lenguaje Natural con muy buenos resultados en inducción de gramáticas, traducción automática, extracción de información, ... El trabajo de uno de sus alumnos de doctorado, Aria D. Haghighi, quedó segundo en la competición del CoNLL 2005 [1].

Berkeley Linguistics (*Universidad Berkeley de California*): en este caso, con un enfoque lingüístico, en este grupo se integra Charles J. Fillmore, que se puede considerar padre teórico del enfoque utilizado en el etiquetado de roles semánticos, ya que contribuyó activamente al desarrollo de las teorías del nexo de las que hemos hablado en este trabajo, y además es el director del proyecto FrameNet.

The Center for Spoken Language Research (*Universidad de Colorado*): Destacamos este grupo de investigación por pertenecer al mismo Martha Palmer, fundadora de VerbNet y una de las integrantes del equipo que desarrolla el proyecto PropBank. En este grupo se encuentra también Nianwen Xue, investigador que está llevando a cabo una versión para el chino de PropBank, y que ha publicado algunos artículos junto a Martha Palmer realmente interesantes sobre etiquetado de roles semánticos, como [50], donde hace un repaso crítico sobre las características comúnmente utilizadas en la construcción de los modelos probabilísticos para etiquetado semántico.

5.2.2 Congresos

A continuación se enumeran algunos de los congresos sobre Procesamiento del Lenguaje Natural más destacados, nacionales e internacionales:

Congreso de la SEPLN : En este congreso se dan cita los distintos grupos de investigación del Procesamiento del Lenguaje Natural a nivel estatal. Está organizado por la Sociedad Española del Procesamiento del Lenguaje Natural, la cuál está formada principalmente por informáticos y lingüistas. Se organiza anualmente, y se publican dos volúmenes en papel al año con los trabajos aceptados. La asistencia a este congreso es obligada para nuestro grupo de investigación, pues nos sirve para mantener el contacto con otros investigadores de nuestra área de conocimiento y tomar el pulso al rumbo que toman las investigaciones sobre Procesamiento del Lenguaje Natural en nuestro país. La edición de este año se celebrará en Sevilla, en la ETS de Ingeniería Informática, y estará organizada por nuestro grupo de investigación.

TAL : son congresos organizados bianualmente, cada vez en un país distinto. El último celebrado fue el FinTAL (Finlandia, 2006), en el que publicamos un artículo sobre nuestra versión supervisada del algoritmo de TextRank [11]. El congreso se centra tanto en Procesamiento del Lenguaje Natural como en lingüística computacional.

ACL : las siglas corresponden a *Association for Computational Linguistics*, y bajo las mismas se engloban una serie de servicios y eventos que constituyen un punto de referencia obligada en el área del Procesamiento del Lenguaje Natural. Dentro de los eventos se incluyen entre otros dos que nos son de especial interés por ocuparse del enfoque estadístico, como son:

- EMNL (Empirical Methods in Natural Language Processing): se centra en trabajos que aborden tareas lingüísticas desde un punto de vista empírico, esto es, utilizando técnicas de aprendizaje automático y otras basadas en datos.
- CoNLL (Computational Natural Language Learning): dentro de este evento se celebra una competición consistente en abordar una tarea concreta. En las ediciones de 2004 y 2005, se centró en el etiquetado de roles semánticos, siendo este el punto inicial desde el que tomé contacto con esta tarea. Entre ambas ediciones se contabilizaron multitud de artículos con distintas propuestas de etiquetadores semánticos, que constituyeron un punto de inicio en el trabajo de revisión bibliográfica y conocimiento del estado del arte.

IBERAMIA : bajo esta denominación se reunen una serie de asociaciones iberoamericanas de investigación en Inteligencia Articial, incluyendo a grupos dedicados al Procesamiento del Lenguaje Natural. Bianualmente organiza un congreso que poco a poco se está convirtiendo en un referente para la comunidad científica internacional de Inteligencia Artificial y en especial para la latinoamericana. En 2006 se celebró la edición décima de este congreso, que tuvo lugar en Brasil.

CAEPIA : es el congreso español por excelencia en el área de la Inteligencia Artificial. Está organizado por la AEPIA (Asociación Española de Inteligencia Artificial), que forma parte a su vez de IBERAMIA. Este año se celebra en Salamanca, y trataremos de presentar un artículo de revisión bibliográfica basado en el presente trabajo.

5.2.3 Revistas

Las revistas son el medio de publicación de artículos con más interés para el curriculum de un investigador. Como medida de la calidad de las mismas, se utiliza un índice de impacto llamado Journal Citation Reports (JCR). Este índice consiste en un número real que, a más valor, indica un mayor impacto de los artículos publicados en la comunidad científica, y por tanto también una mejor valoración por parte de los organismos dedicados a la evaluación de la calidad de aquellos curriculums que contengan artículos publicados en dichas revistas. Es por tanto este un dato importante a tener en cuenta, siendo un objetivo fundamental en la carrera investigadora conseguir cuantas más publicaciones en revistas con buenos índices de impacto. Por supuesto, el nivel de competencia y la dificultad de conseguir que un artículo sea aceptado son directamente proporcionales al índice en cuestión.

En nuestro grupo de investigación hemos llevado a cabo un trabajo de búsqueda y documentación de revistas en las que sería adecuado publicar artículos. A continuación se presentan algunas de ellas con un resumen de algunos datos importantes:

Computational Linguistics

- Índice JCR: 0.8
- Número de artículos: 44 en el último año.
- Tiempo respuesta: No consta en la web. Alta online.
- Número de páginas/artículo: artículos cortos de hasta 15 páginas, largos de hasta 40 páginas. ¡
- ¿Admite propuestas para *special issues*?: Sí. Actualmente hay un número especial en preparación sobre etiquetado de roles semánticos.
- Periodicidad: 4 números al año

Languages Resources and Evaluation

- Índice JCR: -
- Número de artículos: 19 en el último año.
- Tiempo respuesta: No consta en la web. Alta online.
- Número de páginas/artículo: artículos cortos de hasta 6 páginas, largos de hasta 20 páginas.
- ¿Admite propuestas para *special issues*?: Sí.
- Periodicidad: 2 números al año más un especial.

Computer Speech and Language

- Índice JCR: 0.487
- Número de artículos/año: 29 en el último año.
- Cercanía al PLN estadístico: alta.
- Tiempo respuesta: 3 meses la primera contestación, pueden solicitar cambios. Alta online.
- Número de páginas/artículo: no parece haber límites claros, hay artículos de 5 o 10 páginas y otros de hasta 50 páginas.
- ¿Admite propuestas para *special issues*?: Sí, uno de cada 4 números del año es un número especial.
- Periodicidad: 4 números al año

Journal of Universal Computer Science

- Índice JCR: 0.337
- Número de artículos/año: 98 en el último año.
- Cercanía al PLN estadístico: baja.
- Tiempo respuesta: 8 meses aproximadamente. Alta online.
- Número de páginas/artículo: de 20 a 30 páginas aproximadamente.
- ¿Admite propuestas para *special issues*?: Sí.
- Periodicidad: mensual.

Journal of Artificial Intelligence Research

- Índice JCR: 2.247
- Número de artículos/año: de 15 a 20 artículos.
- Cercanía al PLN estadístico: media. En el último vólumen 2 de los 6 artículos versan sobre de procesamiento del lenguaje natural.
- Tiempo respuesta: de 7 a 9 semanas.
- Número de páginas/artículo: de 20 a 30 páginas aproximadamente.
- ¿Admite propuestas para *special issues*?: Sí.
- Periodicidad: anual.

5.3 Líneas de trabajo futuro

En los últimos meses, el trabajo realizado ha sido principalmente de revisión bibliográfica, primer paso a la hora de afrontar una tarea para la que nuestro grupo no poseía ninguna experiencia previa. Una vez hecho esto, creemos conocer lo suficientemente bien la naturaleza del problema y las posibles vías de contribución por parte de nuestro grupo.

La línea de trabajo que ahora mismo se muestra más prometedora es la aplicación de la técnica de *co-training* y *stacking* para la ampliación automática de los recursos semánticos utilizados en etiquetado de roles semánticos, FrameNet y PropBank. Tal como se explicó en la sección de trabajo previo, la técnica consiste en utilizar varios etiquetadores, que son entrenados sobre un corpus de entrenamiento inicial. Se aplican entonces dichos etiquetadores a nuevas frases de entrada no etiquetadas, generando cada uno una salida diferente. Se dispone entonces de una fase de *stacking*, consistente en la utilización de algún algoritmo de aprendizaje (en los trabajos que hemos publicado hemos utilizado árboles de decisión) que decide para cada frase qué candidato tiene más probabilidad de

ser el etiquetado más próximo al correcto. Las frases escogidas son entonces añadidas al corpus de entrenamiento inicial, volviéndose a entrenar los etiquetadores y a repetir el proceso un número determinado de veces, hasta alcanzar las proporciones buscadas o hasta que las salidas obtenidas sean de mala calidad como para ser añadidas al corpus inicial.

Para llevar a cabo experimentos en este sentido, y dada la complejidad de la arquitectura de un etiquetador de roles semánticos, que podría llevarnos muchos meses implementar, debemos hacer una búsqueda de implementaciones disponibles de otros grupos de investigación que hayan afrontado la construcción de dichos etiquetadores. Nos consta de la existencia de varios de ellos. Una vez hayamos escogido los etiquetadores a utilizar, es de esperar que la implementación de la idea no sea costosa puesto que podemos reutilizar todo el trabajo realizado en investigaciones anteriores.

Existen trabajos publicados que trabajan en la integración de los corpus FrameNet y PropBank, traduciendo las notaciones de un sistema al otro y viceversa. Esto nos lleva a plantearnos la posibilidad de llevar a cabo el *co-training* con etiquetadores semánticos entrenados sobre ambos recursos, una vez solucionado el problema de las distintas notaciones. Es de esperar que de esta manera, las visiones aportadas por los etiquetadores entrenados en distintos corpus sean más complementarias y nos permitan obtener resultados de más calidad en la ampliación de los recursos.

Una línea que también parece interesante es la selección de características para el etiquetado de roles. En los últimos artículos se han ido proponiendo nuevas características para los algoritmos de aprendizaje automático, y sería útil evaluar cuál es la aportación de cada una de ellas. Es sabido que un uso descontrolado de características en la construcción de modelos estadísticos puede traer consigo la introducción de demasiada dispersión en los datos de entrada y la caída en rendimiento del clasificador. Por ello sería interesante llevar a cabo esta evaluación y la selección de aquellas características que maximicen los resultados para un etiquetador de roles semánticos. Para poder abordar esta línea, sin embargo, es necesario que alguno de los etiquetadores de roles semánticos disponibles actualmente permita personalizar el conjunto de características de entrada utilizadas, o al menos disponer del código fuente de las implementaciones para evaluar la viabilidad de introducir nosotros los cambios pertinentes para permitirlo. Esta incertidumbre es la que nos hace decantarnos como prioridad por la línea de trabajo anterior.

También hemos considerado otras posibles líneas de trabajo que tras ser evaluadas han sido relegadas a un plano secundario, aunque las mantendremos siempre como posibles vías de trabajo en caso de fracasar en las dos propuestas anteriores. La primera sería la aplicación de etiquetadores de roles semánticos a sistemas de diálogo. La idea es establecer contactos con otros grupos de investigación que se dedican a la investigación en sistemas de diálogo, como el

grupo JULIETTA en el que estuve trabajando un corto periodo antes de ocupar mi plaza de profesor. Es previsible que la utilización de etiquetadores de roles semánticos en estos sistemas sea de una gran utilidad.

Por otro lado, también existe la posibilidad de experimentar la aplicación de técnicas de *bootstrapping* entre un reconocedor de entidades o un desambiguador de significados y un etiquetador de roles semánticos. Las técnicas de *bootstrapping* consisten en la combinación de dos o más sistemas complementarios en un proceso iterativo, de manera que la salida de uno de los sistemas es utilizada por un segundo sistema para entrenarse. A su vez este sistema re-entrenado genera una salida que es utilizada por el primero para perfeccionarse, iterándose este proceso un número de veces determinado. En el caso que nos ocupa, tanto la desambiguación de significados como el reconocimiento de entidades son tareas basadas en aprendizaje automático que participan en la arquitectura de los etiquetadores de roles semánticos, tal como se ha visto en el presente trabajo. Así mismo, según señalan diversos autores, tanto el reconocimiento de entidades como la desambiguación de significados son tareas con un fuerte componente semántico y que podrían beneficiarse de la utilización de etiquetadores de roles semánticos. Es esta doble dependencia la que nos hace plantearnos la aplicación de técnicas de *bootstrapping*.

5.4 Planificación temporal

En esta última sección de mi proyecto investigador voy a realizar una planificación temporal de las tareas que debo desempeñar para llevar a buen término mi tesis doctoral. Actualmente me encuentro en un estadio quizás demasiado temprano para desglosar con gran detalle cada una de las subtareas que me llevarán a completar dicho fin, por lo que en el diagrama siguiente (figura 5.4) sólo se encuentra desglosado al detalle el primer año de los tres planificados para la consecución de mi tesis.

La primera tarea que he llevado a cabo aproximadamente desde febrero de este año ha sido plantear las distintas opciones disponibles en cuanto a investigación relacionada con los etiquetadores de roles semánticos, una vez completada la fase de revisión de la literatura y toma de contacto con el problema. El resultado actual de dicha fase son las líneas de trabajo futuro expuestas en la sección anterior de este capítulo. A partir de este momento (abril 2007) y por un periodo aproximado de dos meses, llevaré a cabo una búsqueda, documentación y evaluación de herramientas de etiquetado de roles semánticos implementadas por los distintos grupos de investigación que trabajan en el tema y que estén disponibles para la comunidad científica. A pesar de decantarme actualmente por la línea de trabajo relacionada con la ampliación automática de los corpus semánticos, durante este periodo de dos meses seguiré estudiando en paralelo las otras líneas de trabajo propuestas, poniéndolas en concordancia con la información que vaya obteniendo sobre los etiquetadores de roles semánticos disponibles. Desde junio de 2007 hasta finales de año plantearé y llevaré a cabo los experimentos

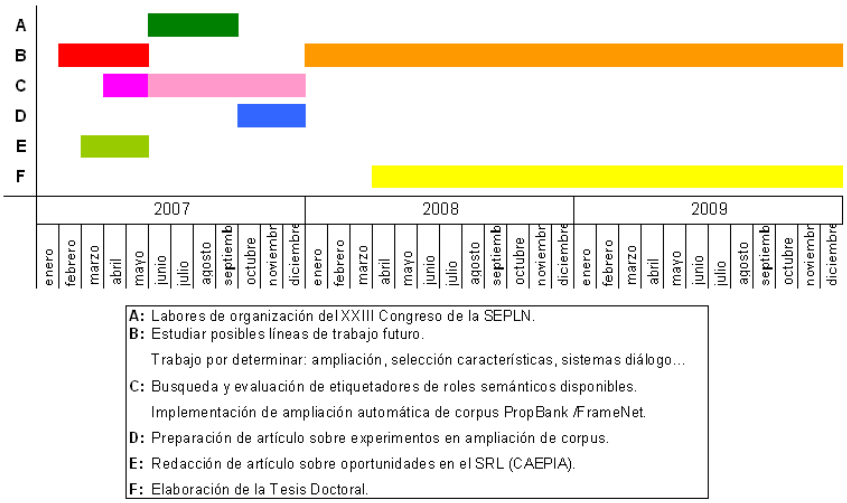


Figura 5.4: Diagrama de planificación temporal

relacionados con la ampliación automática de los corpus semánticos.

En este año intentaré escribir dos artículos. El primero de ellos cuya elaboración está en proceso y que debo concluir antes de finales de mayo de este año consistirá en una reflexión sobre las posibilidades y retos de investigación que se plantean en el campo del etiquetado de roles semánticos, aprovechando para ello el trabajo de revisión bibliográfica realizado para la consecución de este proyecto investigador. Presentaré este artículo al congreso de la CAEPIA que se celebra en noviembre de 2007 en Salamanca. El segundo artículo recogerá las experiencias y resultados, aún por ver si positivos o negativos, obtenidos del trabajo en ampliación de corpus semánticos que desarrollaré este año. Dicho artículo debería estar acabado este mismo año.

La otra tarea que realizaré este año está relacionada con los trabajos de organización del congreso de la Sociedad Española de Procesamiento del Lenguaje Natural que se celebrará en septiembre de 2007 en Sevilla y cuya realización corre este año a cuenta del grupo de investigación al que pertenezco (ITALICA).

A partir de 2008, la idea es continuar los trabajos desarrollados sobre ampliación automática de corpus. Es demasiado pronto para predecir qué derroteros seguirá mi trabajo. Si los resultados en ampliación automática son prometedores, seguiré ese camino. Si no, tendría que decantarme por alguna de las otras líneas de trabajo planteadas. Sea como fuere, mi primera estimación para tener lista mi tesis se sitúa a finales de 2009, con toda la prudencia de saber que aún me encuentro en una fase demasiado temprana como para saberlo con certeza.

Bibliografía

[1] C. Manning A. Haghighi, K. Toutanova. A joint model for semantic role labeling. *Proceedings of CoNLL-2005*, 2005.

[2] A. Blum and T. Mitchell. Combining labelled and unlabeled data with co-training. *11th Annual Conference on Computational Learning Theory*, pages 92–100, 1998.

[3] T. Brants. Tnt. a statistical part-of-speech tagger. *In Proceedings of the 6th Applied NLP Conference (ANLP00)*, pages 224–231, 2000.

[4] E. Brill. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics*, (21):543–565, 1995.

[5] Lou. Burnard. Users reference guide for the british national corpus. *Oxford University Computing Services*, 1995.

[6] X. Carreras and L. Màrquez. Introduction to the conll-2005 shared task: Semantic role labeling. *Proceedings of the 9th Conference on Computational Natural Language Learning*, 2005.

[7] Eugene Charniak. A maximum-entropy-inspired parser. In *Proceedings of the first conference on North American chapter of the Association for Computational Linguistics*, pages 132–139, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[8] Fillmore C.J. Frame semantics. *In Linguistics in the Morning Calm, pp. 111–137*, 1982.

[9] M. Collins. Head-driven statistical models for natural language parsing, 1999.

[10] Nello Cristianini and John Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, March 2000.

[11] F. Cruz, J. A. Troyano, and F. Enriquez. Supervised textrank. *Advances in Natural Language Processing - FINTAL'06 - LNAI*, 4139:632–639, 2006.

[12] F. Cruz, J. A. Troyano, F. Enriquez, and F.J. Ortega. Textrank como motor de aprendizaje en tareas de etiquetado. *Procesamiento del Lenguaje Natural*, 37:33–40, 2006.

[13] F. Enriquez, J. A. Troyano, F. Cruz, and F. J. Ortega. Bootstrapping applied to a corpus generation task. *Computer Aided Systems Theory (Eurocast 2007)*, pages 130–131, 2007.

[14] F. Enriquez, J. A. Troyano, F. Cruz, and F.J. Ortega. Ampliación automática de corpus mediante la colaboración de varios etiquetadores. *Procesamiento del Lenguaje Natural*, 37:11–18, 2006.

[15] Charles J. Fillmore. Frame semantics and the nature of language. *Annals of the New York Academy os Sciences: Conference on the Origin and Development of Language and Speech*, 1976.

[16] Johnson C.R. Fillmore, C.J. and M.R.L. Petruck. Background to framenet. *International Journal of Lexicography, Vol. 16.3: 235-250.*

[17] D. Gildea and D. Jurafsky. Automatic labeling of semantic roles. *Computational Linguistics*, 2002.

[18] P. Singh H. Liu. Conceptnet: A practical commonsense reasoning toolkit. *BT Technology Journal, Volume 22*, 2004.

[19] Douglas Appelt John Bear David Israel Megumi Kameyama Mark E. Stickel Mabry Tyson Hobbs, Jerry R. Fastus: A cascaded finite-state transducer for extracting information from natural-language text. *Finite-State Language Processing. MIT Press*, 1997.

[20] Daniel Jurafsky and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition.* Prentice Hall, January 2000.

[21] W. Ward J. H. Martin D. Jurafsky K. Hacioglu, S. Pradhan. Semantic role labeling by tagging syntactic chunks. *Proceedings of CoNLL-2004*, 2004.

[22] N. Ryant K. Kipper, A. Korhonen and M. Palmer. Extending verbnet with novel verb classes. *Fifth International Conference on Language Resources and Evaluation (LREC 2006)*, 2006.

[23] O. Rambow K. Kipper, M. Palmer. Extending propbank with verbnet semantic predicates. *Workshop on Applied Interlinguas*, 2002.

[24] Karen Kipper. *VerbNet: A broad-coverage, comprehensive verb lexicon.* PhD thesis, University of Pennsylvania, 2005.

[25] J.Giménez N.Català L. Màrquez, P. Comas. Semantic role labeling as sequential tagging. *Proceedings of CoNLL-2005*, 2005.

[26] John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. 18th International Conf. on Machine Learning*, pages 282–289. Morgan Kaufmann, San Francisco, CA, 2001.

[27] L. Lamel, S. Rosset, J. Gauvin, S. Bennacef, and G. Prouts. The limsi arise system, 1998.

[28] Lillian Lee. "i'm sorry dave, i'm afraid i can't do that": Linguistics, statistics, and natural language processing circa 2001. In Committee on the Fundamentals of Computer Science: Challenges, Computer Science Opportunities, and National Research Council Telecommunications Board, editors, *Computer Science: Reflections on the Field, Reflections from the Field*, pages 111–118. The National Academies Press, 2004.

[29] B. Levin. English verb classes and alternation, a preliminary investigation. *The University of Chicago Press*, 1993.

[30] M.A. Marcinkiewicz M. Markus, B. Santorini. Building a large annotated corpus of english: the penn treebank. *Computational Linguistics*, 1993.

[31] M.A. Marcinkiewicz et al. M. Markus, G. Kim. The penn treebank: Annotating predicate argument structure. *Proc of ARPA speech and Natural language workshop*, 1994.

[32] Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts, 1999.

[33] G. Miller. Wordnet: A lexical database. *Communication of the ACM,38(11)*, pages 39–41, 1995.

[34] D.Gildea M.Palmer, P.Kingsbury. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics, Volume 31*, 2005.

[35] Sreerama K. Murthy. Automatic construction of decision trees from data: A multi-disciplinary survey. *Data Mining and Knowledge Discovery*, 2(4):345–389, 1998.

[36] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[37] Koomen P. Roth D. Yih W. Punyakanok, V. Generalized inference with multiple semantic role labeling systems. *Proceedings of CoNLL-2005*, 2005.

[38] L. Rabiner and B. Juang. An introduction to hidden markov models. *ASSP Magazine, IEEE [see also IEEE Signal Processing Magazine]*, 3(1):4–16, 1986.

[39] Paul Tarau Rada Mihalcea. Textrank: Bringing order into texts. *Proceedings of EMNLP 2004*, 2004.

[40] V. Krugler W. Ward J. Martin D. Jurafsky S. Pradhan, K.Hacioglu. Support vector learning for semantic argument classification. *Machine Learning. Special issue on Speech and Natural Language Processing*, 2005.

[41] Stephanie Seneft. Dialogue management in the mercury flight leservation system.

[42] Prashanth Reddy Sriram Venkatapathy, Akshar Bharati. Inferring semantic roles using subcategorization frames and maximum entropy model. *Proceedings of CoNLL-2005*, 2005.

[43] David Stallard. Talk'n'travel: A conversational system for air travel planning. *Proceedings of the 6th Applied Natural Language Processing Conference (ANLP'00)*, 2000.

[44] M. Surdeanu, S. Harabagiu, J. Williams, and P. Aarseth. Using predicate-argument structures for information extraction, 2003.

[45] J.A. Troyano, Víctor J. Díaz, F. Enríquez, and Vicente Carrillo. Applying stacking and corpus transformation to a chuncking task. *Computer Aided Systems Theory (Eurocast 2005). LNCS*, 3643:150–158, 2005.

[46] Yu-Chun Lin Wen-Lian Hsu Tzong-Han Tsai, Chia-Wi Wu. Exploiting full parsing information to label semantic roles using an ensemble of me and svm via integer linear programming. *Proceedings of CoNLL-2005*, 2005.

[47] Wen-tau Yih V. Punyakanok, D. Roth. The necessity of syntactic parsing for semantic role labeling. *Proceedings of the International Joint Conference on Artificial Intelligence*, 2005.

[48] H. L. Somers W. J. Hutchins. An introduction to machine translation. *New York: Academic Press*, 1992.

[49] P. D. Wasserman. *Neural computing: theory and practice*. Van Nostrand Reinhold Co., New York, NY, USA, 1989.

[50] N. Xue and M. Palmer. Calibrating features for semantic role labeling. *Proceedings of EMNLP-2004*, 2004.

[51] V. Zue. Jupiter: A telephone-based conversational interface for weather information, 2000.