Universidad Nacional de Educación a Distancia
Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

Anotación Semántica no Supervisada

TESIS DOCTORAL

David José Fernández Amorós
Licenciado en Cc. Matemáticas
2004

Universidad Nacional de Educación a Distancia
Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

Anotación Semántica no Supervisada

Memoria que presenta para optar al grado de Doctor

David José Fernández Amorós
Licenciado en Cc. Matemáticas por la Universidad Complutense de Madrid

Director:

Julio Antonio Gonzalo Arroyo
Profesor Titular de Universidad del Departamento de Lenguajes y Sistemas Informáticos
de la Universidad Nacional de Educación a Distancia

Quiero dedicar esta tesis a mi madre y mi padre por su amor y por el apoyo me han demostrado incluso cuando no se lo proponían, o precisamente por eso.

Agradecimientos

Quisiera dar gracias a muchas personas e instituciones sin cuyo apoyo esta tesis no habría sido posible o deseable.

En primer lugar a Maribel Dierckx por su paciencia infinita conmigo mientras me encerraba a estudiar en mi habitación, pero también también a mi director Julio Gonzalo por sus consejos, su aguante y por su sentido del humor, a la directora del grupo de Procesamiento del Lenguaje Natural de la UNED, Felisa Verdejo sin cuyo conocimiento y apoyo logístico este libro nunca habría existido, ni muchas otras cosas tampoco, a mis compañeros del susodicho grupo; a Fernando López Ostenero, a Anselmo Peñas Padilla, a Irina Chugur, a Celina Santamaría y a los que vinieron después.

A Carlos Vicente Alvárez por aguantarme en casa y en el trabajo y también a los otros compañeros y compañeras de los cuatro pisos por los que he pasado durante esta odisea.

En especial a mi hermano Marcial Fernández Amorós, por todo.

A José Luis Pérez García, Daniel Abellán, a Rafael Tovar y a Amalia y a Jesús Sánchez y a Nuria.

También quiero agradecer su apoyo moral y de toda índole a todos los miembros del departamento de Lenguajes y Sistemas Informáticos no mencionados todavía, en especial a Covadonga Rodrigo por ser tan auténtica, para ti un abrazo infinito. A Nacho Mayorga, a Juan José Escribano, a José Félix Estívariz, a Juan Antonio Mascarell, a Rubén Heradio, a Timothy Read, por sus templados comentarios.

A las personas con las que he colaborado directa o indirectamente en todo este tiempo o simplemente conocido, en particular a la gente de los grupos de PLN de EHU, por

su compañerismo, en particular a Aitziber Salazar Atutxa, a David Martínez y a Gregorio Hernández Piris, pero también a los de la UPC, UAB y UA.

A Susana Nieva, por haberme puesto en contacto con el grupo de PLN de la UNED y por extensión a la Universidad Complutense de Madrid, *alma mater*, donde terminé la licenciatura y comencé mi doctorado.

A la Federación de Jóvenes Investigadores (www.precarios.org) por algunos buenos ratos que hemos pasado y por hacerme creer en un futuro (imperfecto) para la investigación.

En otro apartado quiero darle las gracias una vez más a Felisa Verdejo por haberme dejado colaborar en algunos de los proyectos que ha dirigido, a José Antonio Cerrada por confiar en mi para alguno de sus cursos, pero también a la Comunidad Autónoma de Madrid y al Fondo Social Europeo por haber financiado mi beca predoctoral, y a la Universidad Pontifica de Salamanca y a la UNED por haberme contratado. También al centro asociado de la UNED en Madrid por el mismo motivo. Por último a la oficina del INEM de Infanta Mercedes en Madrid. Por su apoyo financiero.

Al software libre, por su apoyo tecnológico. La tesis y todos los experimentos y gráficas han sido realizadas sobre GNU/Linux, salvo el experimento de los sintagmas alineados que ha sido realizado bajo Solaris.

Y a muchos otros...

Resumen

En esta tesis se trata el problema de la desambiguación del sentido de las palabras (i.e. dados un diccionario, una palabra y un contexto, decidir en qué sentido del diccionario se está usando la palabra en el contexto). Cuando se desambiguan todas las palabras de un texto se suele utilizar el término anotación semántica (*semantic tagging.*) Existen aproximaciones supervisadas que realizan aprendizaje automático sobre ejemplos resueltos manualmente y otras no supervisadas que se basan en otras fuentes de conocimiento.

En esta tesis se toma la segunda aproximación. Se realiza una combinación de diversas fuentes de conocimiento para lograr un sistema de anotación semántica no supervisado. Las diferentes fuentes de información utilizadas son :

1. La información de origen taxonómico basada en la relación *es-un*, por ejemplo, un águila es-un pájaro. Utilizando una base de datos léxica que provee esta información (WordNet) investigaremos su potencial, demostrando que es una fuente de información valiosa para la tarea. Nuestro punto de partida será el algoritmo original de densidad conceptual de Eneko Agirre y German Rigau, al que aplicaremos algunas modificaciones, principalmente parametrizando algunos aspectos. El algoritmo resultante, evaluado sobre la colección SemCor, produce una mejora del 25 % sobre el original, tanto en precisión como en recall, aunque sigue sin alcanzar los resultados de la heurística del sentido más frecuente.

2. La información de coocurrencias. Tomando como punto de partida un corpus de casi 300 millones de palabras provinientes de libros en formato electrónico (Proyecto Gutenberg) estudiaremos pares de palabras cuyas apariciones en contextos cortos son estadísticamente dependientes. Utilizaremos varias medidas para calibrar ese grado de dependencia y emplearemos dicha información

para desambiguar. Lo haremos comparando las glosas de los sentidos del inventario de sentidos con la información contextual de cada palabra mediante estas medidas. El método empleado, resultó ser el mejor, entre los no supervisados, de los presentados a la competición internacional de desambiguación SENSEVAL-2, para el inglés.

3. Información extraída de la WWW. La información de la glosas del inventario de sentidos serán complementadas con información extraída de la Web. Esta información ha sido extraída de un sistema de clasificación de documentos realizado por voluntarios (Open Directory Project) por Celina Santamaría. Descubriremos que dicha información es de una calidad superior a la proporcionada por los ejemplos de entrenamiento en la competición SENSEVAL-2, aunque su escaso volumen no estimula por ahora un uso a gran escala.

4. Información proviniente de corpora bilingüe comparable. Partiendo de un corpus en inglés y otro en español se han buscado patrones sintácticos superficiales correspondientes a sintagmas nominales en ambos idiomas. Después se buscan correspondencias entre los sintagmas hallados en ambos idiomas, que podrían interpretarse como traducciones tentativas de dichos sintagmas. A partir de este trabajo realizado por Anselmo Peñas y Fernando López Ostenero estudiaremos si es posible aprovechar las diferencias entre ambos idiomas para detectar estos sintagmas y desambiguar mediante las capacidades translingües de una base de conocimiento léxica (EuroWordNet). Los resultados muestran que la cantidad de información perdida a lo largo del proceso impide alcanzar un rendimiento competitivo.

De todas de estas fuentes de información demostraremos la utilidad las tres primeras y descartaremos la cuarta por dar resultados poco prometedores en una primera aproximación.

Finalmente presentaremos un sistema de anotación semántica que combinará los dos primeros tipos de evidencia y compararemos su rendimiento con otros sistemas de anotación. La tercera fuente de información (la extraída de la Web) no será empleada en el sistema final porque el volumen de datos, aunque de buena calidad es escaso y mientras que no influiría significativamente en el resultado final, complicaría la tarea de combinación.

El sistema final presentado en el capítulo 7 mejora ligeramente al anteriormente mencionado en el punto 2. En definitiva, se demostrará que la anotación semántica no supervisada puede lograr buenos resultados, y que hay lineas de investigación, con un importante potencial de mejora, que merecen exploradas.

Abstract

This thesis deals with the problem of word sense disambiguation (i.e. given a dictionary, a word and a context, deciding in which of the senses of the dictionary is the word being used). When each of the words in a text is disambiguated the task is called semantic tagging. There are supervised approaches that do machine learning over hand-tagged examples and there are others that are knowledge-based and thus non-supervised.

This thesis takes the second approach. Different knowledge sources are combined in order to achieve a non-supervised semantic tagging system. The different sources of information used are :

1. Taxonomic information based on the *is-a* relation, for instance, an eagle *is a* bird. Using a lexical database that provides this information (WordNet), we will research its potential and we will show that this type of information is indeed valuable for the task at hand. Our starting point will be the original Conceptual Density algorithm by Eneko Agirre and German Rigau, which we will improve, mainly by parameterizing some aspects. The resulting algorithm, evaluated over the SemCor collection, obtains an improvement of 25 % in precision as well as recall over the original one, although it still doesn't reach the results of the most frequent sense heuristic.

2. Coocurrence information. We will start from a corpus of nearly 300 million words (Project Gutenberg) and we will study pairs of words whose occurrences in short contexts are statistically dependent. We will use several measures to calibrate that degree of dependency and employ that information in semantic tagging. We will do that by comparing the glosses for the senses in the sense inventory with the contextual information for each word. The method employed turned out to be the best among the non-supervised systems presented to the international disambiguation competition SENSEVAL-2, for the English language.

3. Information extracted from the World Wide Web. The information of the glosses of the sense inventory will be complemented with information extracted from the Web. This information has been extracted from a document classification system kept by volunteers (Open Directory Project) by Celina Santamaría. We will discover that this information is of superior quality to that provided by the training examples in the SENSEVAL-2 competition, although the scarce amount of data collected doesn't stimulate its widespread use at this moment.

4. Information coming from bilingual comparable corpora. We start with a corpus in English and another in Spanish. Shallow syntactic patterns corresponding to noun phrases have been detected in both corpora. In the next step, correspondences between these phrases in each of the languages have been established. These correspondences could be viewed as tentative translations of those phrases. From this work, undertaken by Anselmo Peñas and Fernando López Ostenero we will study if it is possible to take advantage of the differences between the two languages to detect these phrases and disambiguate by means of the translingual abilities of a lexical knowledge base (EuroWordNet). Results show that the amount of information lost along the process prevents from achieving competitive performance.

Out of all those sources of information we will show the usefulness of the first three and discard the fourth one due to the unpromising results obtained in a first approach.

Finally, we will present a semantic tagging system that will combine the first two types of evidence discussed and will compare it against other systems proposed. The third source of information (the information extracted from the Web) won't be used in the final system because the volume of data, although of a good quality, is scarce and wouldn't influence significantly the final result, but would complicate the combination task.

The final system presented in chapter 7 performs slightly better than the one previously mentioned. All in all, we will show that non supervised semantic tagging can achieve good results, and that there are research lines, with a good potential for improvement, that deserve to be explored.

Índice general

Índice de figuras

Índice de cuadros

Capítulo 1

Introducción

El problema de la desambiguación del sentido de las palabras (DSP) consiste en determinar el significado de una palabra en un contexto dado, entre las alternativas que nos ofrece un diccionario. Por ejemplo, si buscamos la palabra banco en el diccionario de la Real Academia, encontramos lo siguiente:

1. m. Asiento con respaldo o sin él, en que pueden sentarse varias personas.
2. Madero grueso escuadrado que se coloca horizontalmente sobre cuatro pies y sirve como de mesa para muchas labores de los carpinteros, cerrajeros, herradores y otros artesanos.
3. Cama del freno. Ú. m. en pl.
4. En los mares, ríos y lagos navegables, bajo que se prolonga en una gran extensión.
5. Conjunto de peces que van juntos en gran número.
6. fig. y fam. V. pata, pie de banco.
7. Arq. sotabanco de una casa.
8. Geol. Estrato de gran espesor.
9. Min. Macizo de mineral que presenta dos caras descubiertas, una horizontal superior y otra vertical.

10. Del it. banca, mesa de los cambistas. Establecimiento público de crédito, constituido en sociedad por acciones.

11. Del m. or. que la anterior. Establecimiento médico donde se conservan y almacenan órganos, tejidos o líquidos fisiológicos humanos para cubrir necesidades quirúrgicas, de investigación, etc. BANCO de ojos, de sangre.

12. Del m. or. que la anterior. p. us. El que cambia moneda.

Confrontados con la oración: *María ha ido al banco a realizar unas gestiones*, la desambiguación de la palabra banco en este contexto consistiría en señalar como sentido el referente a la mesa de los cambistas, es decir, el número 10. ¿O quizás el 12? ¿O tal vez ambos?

Este es un problema con una incidencia real; se calcula que aunque el número medio de significados por palabra en un diccionario está, típicamente, alrededor de dos. Las palabras más ambiguas son las más frecuentes, y por ello en textos reales esta cantidad se aproxima a cinco. Es un problema con interés en sí mismo pero también resulta de interés para la Inteligencia Artificial (IA) en general y el Procesamiento del Lenguaje Natural (PLN) en particular, así como para otras disciplinas como la Filosofía, la Psicología y la Lexicografía.

El problema de la desambiguación ha merecido con frecuencia la consideración de *problema intermedio* en la comprensión del lenguaje natural (Wilks and Stevenson, 1996). En esta extendida visión del PLN, la desambiguación sería una fuente más de conocimiento sobre un texto como puede serlo la proporcionada por segmentadores de palabras, anotadores sintácticos o separadores de frases. Una vez que se ha recopilado suficiente información sobre un texto se está en condiciones de pasar a una aplicación concreta. Sin embargo, la desambiguación también ha sido utilizada como banco de pruebas para medir indirectamente la efectividad de medidas de similitud léxica, siendo, en este caso, una aplicación final.

Se trata de un problema de investigación básica, es decir, no se pretende buscar una aplicación inmediata a la desambiguación del sentido de las palabras sino más bien profundizar en la naturaleza del problema e intentar resolverlo, a pesar de que su resolución puede contribuir a muchas aplicaciones prácticas interesantes.

Algunas de las posibles aplicaciones del PLN en general, y que podrían beneficiarse de la DSP son, sin pretender ser exhaustivos:

- Recuperación de Información (RI): El término *recuperación de información* se utiliza para referirse al problema de encontrar, dada una colección de documentos, todos aquellos que son relevantes para la pregunta de un usuario. Cualquier persona que haya tenido que utilizar un buscador de Internet o de una biblioteca puede comprender hasta qué punto la desambiguación semántica automática podría ayudar en la tarea de encontrar lo que se busca.

  En recuperación de información hay dos conceptos importantes de cara a la evaluación; uno es la cantidad de documentos relevantes para la consulta entre los recuperados, el otro es la proporción de los documentos relevantes para la consulta que han sido recuperados por el sistema de RI.

  Hay dos fenómenos lingüísticos especialmente problemáticos en RI; la sinonimia y la polisemia. Dos palabras en contexto, son sinónimas si son palabras distintas pero pueden tener el mismo significado, ejemplo de ello podría ser la pareja casa/hogar. Una palabra es polisémica si tiene varios significados, como hemos visto en el caso de banco.

  La sinonimia dificulta encontrar los documentos que son relevantes pero usan palabras sinónimas a las de la consulta. Supongamos que realizamos en un buscador de Internet la consulta *Vacaciones en casas rurales*. Sin la ayuda de la desambiguación semántica automática, una página web con el texto *Ofertas de vacaciones en casas de campo*, pasaría posiblemente desapercibida, al no conocer el sistema informático que *rural* y *de campo* son sinónimos en este caso.

  La polisemia facilita que se recuperen por error documentos irrelevantes que comparten palabras con la consulta. Un ejemplo de esto lo ejemplifica la siguiente situación; podríamos buscar *Horarios diarios de trenes a Chinchilla* y encontrar *Primicia en el diario de Villarriba; un hombre muerde a un perro.*

  La contribución de la desambiguación semántica a la RI permitiría, al desambiguar tanto consultas como documentos, eliminar el efecto de la sinonimia y polisemia de modo que se recuperaran todos los documentos relevantes para una consulta y sólo los relevantes.

  Sin embargo, a pesar de que sistemas de desambiguación del sentido de las palabras hay muchos, lo que no existe todavía es una evidencia de que la resolución del problema tal como se plantea actualmente resulte de utilidad en aplicaciones finales.

  En el caso de la recuperación de información, hay tanto estudios que defienden la utilidad de los sistemas de DSP (Gonzalo et al., 1998; Gonzalo et al., 1999; Schütze and Pedersen, 1995) como otros que la ponen en duda (Sanderson, 1994; Sanderson, 2000).

Los sistemas de RI con más éxito no desambiguan de forma explícita. A veces se habla de que estos sistemas realizan una *desambiguación implícita* e incluso se habla de *semántica latente* (Deerwester et al., 1990) para evitar el referirse a un proceso explícito de desambiguación. Esto quiere decir que si la consulta es corta, resulta muy difícil para un sistema informático deducir cuáles son los sentidos correctos de las palabras que la componen, por falta de contexto suficiente (recordemos que los ordenadores carecen de sentido común, como muchas personas), pero si es muy larga hay muchas palabras a buscar en los documentos de la colección, de modo que a medida que aumenta el número de palabras de la consulta más difícil es que se produzcan los efectos adversos antes mencionados de sinonimia y polisemia.

- Categorización de textos: Consiste en asignar a cada texto de una colección una o varias categorías que podrían formar parte de una estructura más compleja (típicamente una jerarquía de categorías). Esta tarea se realiza, por ejemplo, en las agencias de noticias cuando asignan a las noticias categorías periodísticas para su mejor distribución. El papel de la desambiguación semántica en esta aplicación sería similar al jugado en la recuperación de información. También son habituales las jerarquías de categorías de textos en portales de Internet, como pueden ser el de Terra [1] o el de Yahoo[2]. Un sistema concreto de categorización de textos basado en desambiguación semántica se puede encontrar en la tesis (Ureña, 2001).

- Agrupamiento de documentos de texto (*text document clustering*): Esta tarea trata de particionar una colección de textos en grupos de tal manera que cada texto pertenezca exactamente a un grupo. Cada uno de estos grupos se puede caracterizar como una respuesta a una consulta a la base de datos documental. Un ejemplo de que la aplicación de técnicas de desambiguación semántica en esta tarea mejora los resultados puede encontrarse en (Hotho et al., 2003).

- Traducción Automática (TA): La traducción automática, como su propio nombre indica, consiste en traducir un texto de un idioma a otro. En el momento actual, no se considera como una alternativa viable a la traducción humana para la mayoría de los campos de producción lingüística, sin embargo, en entornos muy especializados (por ejemplo, en las predicciones del tiempo atmosférico, o en el etiquetado de productos en múltiples idiomas), con un vocabulario limitado y de contenido más técnico que de opinión, es una linea de investigación con resultados muy satisfactorios.

[1] www.terra.es

[2] www.yahoo.es

Uno de los problemas más evidentes en traducción automática es que con frecuencia los sistemas de traducción realizan la elección equivocada a la hora de escoger el término en el idioma destino correspondiente a un cierto término en el lenguaje origen. La desambiguación previa de los términos contribuiría a reducir el número de opciones entre las que elegir. Como ejemplo, si tenemos que traducir *I asked for a loan in the bank*[3] de Inglés a Español, la desambiguación semántica nos ayudaría, al traducir *bank*, a decidirnos por banco y no por orilla. Elegido el significado, quedaría todavía el problema de elegir en el idioma destino cual de los sinónimos es el más apropiado en el contexto, pero esa es otra cuestión.

- Sistemas de Pregunta/Respuesta (Question Answering): La búsqueda de respuestas en dominios abiertos es, según la competición internacional *Text Retrieval Evaluation Conference* (TREC), la tarea de identificar, en una gran colección de documentos, un fragmento de texto donde se encuentre la respuesta a una pregunta formulada en lenguaje natural. La aplicación de técnicas de desambiguación es provechosa tanto en el caso de las preguntas como de las colecciones donde residen las respuestas. Por ejemplo, en la pregunta:

  Who was the president of Vichy France?[4]

  Para encontrar la respuesta, en este caso

  Marshall Philippe Pétain, head of Vichy France government[5]

  nos sería útil desambiguar head puesto que se trata de una palabra de una enorme polisemia (32 sentidos en la base de datos léxica WordNet-1.7).

Un criterio de clasificación de las técnicas de desambiguación automática consiste en distinguir entre aquellas que utilizan ejemplos de entrenamiento desambiguados manualmente por expertos humanos (algoritmos supervisados) y las que se limitan a usar los recursos proporcionados por un diccionario electrónico, una base de datos léxica o un tesauro (algoritmos no supervisados).

[3] Pedí un crédito en el banco.

[4] ¿Quién era el presidente de la Francia de Vichy?

[5] Marshall Philippe Pétain, cabeza del gobierno francés de Vichy.

En los primeros tiempos de las técnicas de DSP era frecuente utilizar algoritmos no supervisados por el simple motivo de que los recursos léxicos en formato electrónico eran escasos. Con la aparición de corpora anotados manualmente los algoritmos supervisados comenzaron a hacerse más populares, de suerte que en la actualidad existe una tendencia a aplicar cualquier método de aprendizaje automático que haya dado buenos resultados en otros campos. Por otra parte, los algoritmos supervisados padecen directamente el cuello de botella de adquisición del conocimiento, puesto que la producción de grandes corpora anotados manualmente resulta cara y no hay conocimiento sobre la cantidad de tal información necesaria para resolver el problema a un nivel razonable (si tal cosa fuera posible y dependiera sólo de la cantidad de información).

La distinción entre sistemas supervisados y no supervisados ha ido haciéndose mas estrecha con el paso del tiempo, puesto que gradualmente se han añadido nuevos recursos léxicos a los diccionarios y tesauros que hacen difícil trazar una línea que distinga cuándo la intervención humana ha sido suficientemente grande en la elaboración de los recursos utilizados en la construcción de un sistema de DSP como para que éste se pueda considerar supervisado. A los efectos de este trabajo, sólo llamaremos supervisados a los algoritmos que utilizan ejemplos de uso de cada sentido como fuente de aprendizaje para la tarea. La adquisición de estos ejemplos de uso anotados manualmente por expertos en lexicografía resulta muy costosa.

La dificultad del objetivo puede apreciarse por el hecho de que la DSP tiene complejas interrelaciones con otras áreas del PLN, disciplina considerada por algunos autores como Wilensky, o (Gale et al., 1993) un problema IA-completo, es decir, que si se conociera como resolver el problema se podría resolver cualquier otro problema de la Inteligencia Artificial. Otra visión de la crudeza del problema la muestran (Gale et al., 1993) citando a su vez a Masterson, "The basic problem in machine translation is that of multiple meaning"[6] (Masterson, 1967) y a Bar-Hillel, que abandonó el problema ante la certeza de la imposibilidad de desambiguar algunas palabras, notoriamente, *pen* en el siguiente texto:

Little John was looking for his toy box. Finally he found it. The box was in the pen. John was very happy.[7]

El argumento de Bar-Hillel fue el siguiente:

[6]El problema básico en traducción automática es el de la multiplicidad de sentidos.

[7]El pequeño John estaba buscando su caja de juguetes. Finalmente la encontró. La caja estaba en el corral. John estaba muy feliz.

Assume, for simplicity's sake, that pen *in English has only the following two meanings: (1) a certain writing utensil, (2) an enclosure where small children can play. I now claim that no existing or imaginable program will enable an electronic computer to determine that the word* pen *in the given sentence within the given context has the second of the above meanings whereas every reader with a sufficient knowledge of English will do this* automatically.[8]

La dificultad para desambiguar el sentido de las palabras tiene varias causas. Entre ellas:

1. Está por ver si la actual formulación del problema en términos de las definiciones de los sentidos de las palabras que proporciona un diccionario es adecuada o, al menos, como consistente consigo misma.

   Esta aproximación en la cual los sentidos de una palabra se buscan en un *inventario de sentidos*, en el cual figuran todos los posibles significados de la palabra y además cada sentido está perfectamente diferenciado de todos los demás se conoce como el modelo del *banco* de sentidos. Los experimentos demuestran que con frecuencia los seres humanos no consiguen alcanzar un consenso a la hora de decidir el sentido de una palabra en contexto según las posibilidades de un diccionario determinado. Esto pone de manifiesto algunas de las innegables carencias del modelo del banco de sentidos. En particular, en este modelo, la desambiguación semántica se reduciría a un problema muy complejo de clasificación en IA.

2. Es difícil encontrar, para una lengua, un inventario de sentidos estándar (ya sea un diccionario, tesauro o base de datos léxica). La exhaustividad del inventario es una utopía, puesto que una característica del lenguaje natural es su capacidad para crear sentidos nuevos para palabras que ya existían (y nuevas palabras con sus propios sentidos). Se dispone de diccionarios y tesauros en formato electrónico (casi siempre protegidos por estrictos *copyrights* que dificultan su accesibilidad con fines científicos) cuya compatibilidad es sólo relativa. No hay consenso sobre qué es un significado y cómo representarlo computacionalmente.

3. Para cada aplicación, el grado de refinamiento necesario al considerar sentidos es muy distinto. Por ejemplo, debería ser diferente la granularidad de sentidos

[8]Supongamos, para simplificar, que *pen* en inglés solo tiene los dos significados siguientes: (1) Un cierto utensilio de escritura. (2) Un pequeño corral donde los niños pequeños pueden jugar. Yo afirmo ahora que ningún programa actual o imaginable permitirá a un ordenador electrónico determinar que la palabra *pen* en la frase dada en el contexto dado tiene el segundo de los significados anteriores, mientras que cualquier lector con un conocimiento suficiente de inglés lo hará *automáticamente*.

de un léxico para recuperación de información que para traducción automática. Mientras que en recuperación de información nos interesa distinguir los sentidos que discriminan documentos entre sí, en traducción automática nos interesa distinguir los sentidos que reciban distintas traducciones en el idioma destino.

4. El sentido de las palabras es muy sensible al dominio concreto del texto. Esta dependencia del dominio puede explicar en parte las diferencias de rendimiento de algunos sistemas de DSP, pero no ha sido aún estudiada suficientemente.

5. A diferencia de la situación en otras tareas intermedias del PLN, como la desambiguación morfosintáctica o el análisis de estructuras sintácticas, no existen apenas colecciones anotadas de textos para entrenar y/o evaluar algoritmos de desambiguación. En particular, sólo para el inglés existen pequeñas colecciones cuya validez como referencia es discutible. Por este motivo, es necesario buscar fuentes de información alternativas a los ejemplos anotados a mano.

6. Muchos sistemas de DSP combinan diversas fuentes de información, pero no realizan un análisis de la influencia de cada factor en el rendimiento final, de manera que resulta difícil extraer conclusiones al respecto. En particular, estamos interesados en conocer el impacto en un sistema de DSP de la información de tipo taxonómico de WordNet, de la información de coocurrencias de pares de palabras en un corpus de gran tamaño, de la información extraída automáticamente de la Web y de la utilización de sintagmas bilingües alineados extraídos de corpora bilingües comparables como factores de desambiguación.

A pesar de sus limitaciones, el modelo del banco de sentidos es el más consolidado de inventario de sentidos y es la base de casi todo el trabajo empírico realizado sobre DSP, por lo cual lo adoptaremos en este trabajo de investigación. Esto permitirá la comparación de los resultados de los experimentos con los de otros investigadores.

En esta tesis se adoptará como inventario estándar la base de datos léxica EuroWordNet (Vossen, 1997), por motivos que serán expuestos más adelante. Asimismo la investigación se centrará en el uso de técnicas de desambiguación no supervisadas. Estas técnicas son interesantes por sí mismas y además evitan algunos de los problemas de los sistemas supervisados. Podemos citar algunos de ellos: El alto coste de adquisición del conocimiento en forma de ejemplos anotados, la dificultad de encontrar ejemplos de entrenamiento en cantidad suficiente para los sentidos menos habituales de palabras poco frecuentes. Tampoco se ha demostrado que exista una correlación entre número de ejemplos de entrenamiento disponibles y rendimiento de los sistemas supervisados, sin embargo hay estudios que prueban que en ocasiones cuantos más datos de entrenamiento hay, peores son los resultados (Cucchiarelli et al., 2000).

1.1. Objetivos

El objetivo general de esta tesis es el estudio de la DSP no supervisada a gran escala, a través de las siguientes tareas:

- Estudiar el papel de distintos tipos de información en los procesos de DSP por separado. En especial, la información de proximidad conceptual y la información de coocurrencia.
- Estudiar otras fuentes de información alternativas (a las supervisadas y a las del punto anterior) para caracterizar sentidos. En concreto, utilizaremos directorios de Internet y corpora comparable en varios idiomas.
- A partir de los resultados obtenidos en los puntos anteriores, proporcionar un sistema de DSP no supervisado, a gran escala y eficiente:

  1. No supervisado, de forma que no dependa de la adquisición manual de material de entrenamiento, tarea excesivamente costosa para ser asumible en posibles aplicaciones. Además, a diferencia de los algoritmos supervisados, el campo de la desambiguación no supervisada está relativamente inexplorado y tiene, todavía, un amplio margen de mejora.
  2. A gran escala: capaz de desambiguar todas las palabras de un idioma.
  3. Analizable: de modo que los resultados sean interpretables de forma clara, y la aportación de cada factor pueda ser sopesada individualmente.
  4. Eficiente: capaz de procesar grandes cantidades de texto, de forma que sea adecuado para aplicaciones sobre bases de datos documentales masivas (como la recuperación de información o la traducción automática).

1.2. Metodología

1. Creación de una plataforma de desarrollo y evaluación de sistemas de DSP, tanto para desarrollar y evaluar los propios como otros que se hayan propuesto en la literatura cuando las evaluaciones disponibles no sean directamente comparables.

   Como recurso léxico básico se utilizará WordNet/EuroWordNet (Vossen, 1997). WordNet es una base de datos léxica enriquecida con relaciones semánticas. Se

presentarán sus características con detalle en la sección sobre recursos léxicos en 2.2. Aun así, esta decisión se fundamenta en diversas razones:

- Es un recurso independiente del idioma, por lo que supone una excelente base para el desarrollo de técnicas de desambiguación independientes del idioma.
- No sólo proporciona un inventario de sentidos e información vía definiciones (glosas) de los sentidos de las palabras, sino también relaciones semánticas entre ellos. De particular interés resulta su jerarquía de la relación *es-un* entre significados.
- Se está convirtiendo en un estándar internacional de base de datos léxica (EuroWordNet). Multitud de investigadores se han decantado por él por lo que la siempre deseable comparación directa de resultados de investigación resulta posible.
- Es un recurso gratuito para fines de investigación.

2. Estudio de la utilidad de la información de tipo jerárquico en la desambiguación, para ello partiremos del algoritmo de Agirre & Rigau (Agirre and Rigau, 1995; Agirre and Rigau, 1996) y estudiaremos posibles mejoras y generalizaciones.

3. Estudio de la utilidad de la información obtenida del conteo de coocurrencias entre palabras. Para ello se recopilará la información (en inglés, por ser la más abundante) del Proyecto Gutenberg, del orden de cientos de millones de palabras para la extracción de información de coocurrencias. Esta información puede depurarse para obtener medidas como la información mutua (IM) entre dos palabras, que puede interpretarse como una medida de su relación semántica.

4. Enriquecimiento de la información asociada a algunos sentidos con etiquetas de dominio extraídas de forma automática de Internet (Santamaría et al., 2001).

5. Incorporación al sistema de la información proporcionada por las traducciones alineadas de sintagmas nominales extraídos automáticamente de corpora comparable no anotado en varios idiomas (López Ostenero et al., 2002).

6. Participación en la competición internacional de DSP SENSEVAL-2, donde se presentarán sistemas de DSP no supervisada y una extensión supervisada formada por la integración de los sistemas no supervisados con los ejemplos de entrenamiento proporcionados por los organizadores, lo cual no sólo proporcionará una evaluación externa sino al mismo tiempo una comparación con otros sistemas actuales, supervisados y no supervisados.

7. Evaluación de la aportación de cada uno de los tipos de información (jerárquica, coocurrencias, etiquetas de dominio y sintagmas alineados) por separado y conjuntamente.

1.3. Estructura del resto de la tesis

En el 2, se realiza un estudio de algunos de los factores que han marcado la investigación en desambiguación del sentido de las palabras desde el punto de vista teórico. En el 3 se realiza una clasificación de las aportaciones más relevantes en forma de sistemas concretos.

En el capítulo 4 se presenta un experimento realizado con el objeto de conocer la importancia de la información de tipo taxonómico en la DSP (la relación es-un, como existe por ejemplo entre naranja y fruta).

En el capítulo 5 se presenta un estudio del impacto de la información sobre coocurrencias de palabras en textos y las medidas de teoría de la información que pueden asociarse con ellas de manera que esta información sea puesta al servicio de la tarea.

En el capítulo 6 se describen sendos experimentos con otras fuentes de información en desambiguación; en el primero de ellos se estudiará la influencia de la información extraída de un directorio de páginas web realizado por voluntarios y en el segundo se estudiará la viabilidad de aprovechar las irregularidades de traducción de sintagmas nominales entre dos idiomas (español e inglés en este caso) para su uso en anotación semántica.

En el capítulo 7 se presentará un sistema de anotación semántica que combinará adecuadamente los diversos tipos de información empleados y en el 8 se extraerán las conclusiones pertinentes.

Parte I

El Estado del arte

Capítulo 2

Aspectos teóricos en DSP

El problema de la desambiguación del sentido de las palabras, o el problema dual de la distinción de sentidos (para poder hablar de desambiguación debe existir previamente un cierto consenso sobre cuáles son los diferentes sentidos de una palabra) ha sido estudiado por muchos investigadores; de hecho, las investigaciones se remontan al menos a Aristóteles en su obra *Topics* alrededor de 350 A.C. Otros interesados en el tema de épocas más cercanas serían Saussure, Meillet y Wittengenstein. Tres buenas visiones panorámicas sobre el tema pueden encontrarse en (Ide and Veronis, 1998; Fuji, 1998; Agirre and Martinez, 2001).

El problema (de la desambiguación propiamente dicha) se reconoce desde el inicio de la era informática (Weaver, 1955). Sin embargo, los primeros enfoques, aunque interesantes desde el punto de vista teórico, adolecen de una gran falta de recursos que lleva irremediablemente a considerar pequeñas muestras de datos, lo que hace aún más difíciles las comparaciones entre sistemas de DSP. Con el paso del tiempo, el énfasis se ha puesto en los sistemas supervisados, debido tanto al desarrollo de recursos lingüísticos de todo tipo como, por qué no decirlo, a la comodidad de aplicar técnicas estándar de aprendizaje automático desarrolladas en el último medio siglo y que han funcionado bien para otros problemas. Las aproximaciones al problema han generado aportaciones tanto de tipo introspectivo sobre la naturaleza del problema, como experimentales en la forma de algoritmos concretos de desambiguación.

Empezaremos revisando algunas de estas aportaciones de carácter más teórico sobre la naturaleza del problema y proseguiremos en el próximo capítulo con una revisión de los sistemas concretos presentados a la comunidad científica.

2.1. La naturaleza del problema

En esta sección tienen cabida las ideas y experimentos que se han realizado para acotar y profundizar en los factores que influyen en el problema, y de qué manera lo hacen.

2.1.1. Uso de ejemplos anotados

Una teoría muy influyente en la literatura fue la de *un sentido por colocación* (Yarowsky, 1993). Yarowsky nos propone la siguiente definición de *colocación*: La coaparición de dos palabras en alguna relación definida. Yarowsky estudia varias relaciones: adyacencia directa, primera palabra a la derecha o a la izquierda de una determinada *parte del discurso* (o categoría gramatical) y relaciones sintácticas directas, como verbo/objeto, sujeto/verbo y nombre/adjetivo. También diferencia las palabras con contenido de las palabras de función.

Se estudia la importancia de estas colocaciones para distinciones binarias de sentidos. En el estudio de Yarowsky, estas distinciones tienen varios orígenes: homógrafos anotados a mano, palabras en inglés que tienen dos traducciones en francés (de los Hansards canadienses, diario de sesiones del parlamento canadiense, en francés e inglés), homófonos, ambigüedades OCR (palabras que se distinguen en un carácter o dos) y pseudopalabras.

Para calcular la entropía de las distribuciones de sentidos para cada palabra se emplea validación cruzada, por ejemplo, en la distribución de aid/aide, se observa que con colocaciones como squander (squander aid/ squander aide) la proporción de aparición sería 1/0 (esto es, que *squander aid* aparece una vez y *squander aide* ninguna), pero no es aceptable considerar que sea imposible *squander aide* por lo que se realiza validación cruzada para buscar más apariciones.

La validación cruzada es una técnica comúnmente utilizada en aprendizaje automático que consiste en en dividir los datos de entrenamiento en varias partes de forma que sucesivamente una parte pueda ser utilizada como banco de pruebas y el resto como entrenamiento. Por ejemplo, en el caso de la desambiguación se podría entrenar con la mitad de una colección y evaluar sobre la otra mitad. En este caso nuestra colección de prueba tendría un tamaño del 50 % de los datos totales. La validación cruzada consistiría en dividir, por ejemplo en diez partes aproximadamente iguales la colección de manera que pudiéramos realizar diez experimentos en cada uno de los

cuales una de las partes se reservaría para la evaluación y las otras nueve para entrenar. Finalmente se realizan las medias (posiblemente ponderadas) correspondientes y el resultado es que hemos utilizado la colección entera para entrenar y al mismo tiempo nuestros resultados son más representativos, puesto que hemos utilizado como banco de pruebas toda la colección. Todo sin cometer en ningún momento el *pecado mortal* del aprendizaje automático, esto es, evaluar contra los mismos datos con los que hemos entrenado.

El algoritmo de desambiguación que surge de esto es sencillo. Para desambiguar una palabra hay que fijarse en el contexto y buscar si la colocación se encuentra en los datos de entrada. Si está, se responde el sentido que más aparezca en las colocaciones. Si hay más de una colocación aplicable se usa el método de las listas de decisión (Rivest, 1987) (la regla que tenga la verosimilitud más alta es la que se aplica) para no combinar fuentes de información no independientes. Si no hay ninguna colocación aplicable se escoge el sentido más frecuente en el entrenamiento.

En cuanto a los resultados, al coste de una cobertura más o menos baja (la cobertura es la proporción de casos de prueba para los que se ha dado una respuesta), las colocaciones dan una precisión entre el 90 y el 99 %, con una media del 92 %. Las diferencias entre categorías gramaticales son muy interesantes: Los verbos obtienen más información de desambiguación de sus objetos que de sus sujetos, los adjetivos derivan casi toda su información de desambiguación de los nombres a los que califican. Los nombres de los adjetivos adyacentes u otros nombres, y la palabra con contenido a la izquierda da el sentido correcto en el 99 % de los casos. En (Gale et al., 1993) quedó demostrado que había información relevante para la desambiguación hasta a 10000 palabras de distancia. Sin embargo, si se parte la estadística por categorías gramaticales, la caída de la precisión con la información lograda en los bordes (sólo en los bordes) de la ventana de contexto, es mucho mayor en verbos y adjetivos que en nombres.

Una aparente conclusión es que, si se dispusiera de un corpus anotado con suficientes apariciones de cada una de estas colocaciones, la DSP sería un problema casi trivial. Desgraciadamente el esfuerzo necesario para lograrlo sería enorme, incluso si la tarea está bien definida. En (Ng, 1997a), se estimó el esfuerzo necesario para crear un corpus de entrenamiento anotado manualmente con una cobertura razonable en 16-años/hombre. Sin embargo, en el caso de Ng el algoritmo asociado era menos exigente que el de Yarowsky respecto de la cantidad de ejemplos anotados necesaria, por lo que el tiempo necesario para anotar suficientes colocaciones podría ser varios órdenes de magnitud mayor.

En (Cucchiarelli et al., 2000), podemos encontrar una prueba de que no siempre el mero hecho de aumentar la cantidad de datos de entrenamiento hace que el sistema mejore. Desambiguaron 297 palabras sobre WordNet(Miller, 1995), a nivel de sentido de la *top-ontology* (la parte superior de la jerarquía, formada por unas pocas decenas de conceptos). Es decir, se trataba de una distinción de sentidos de grano grueso (una tarea obviamente más sencilla que desambiguar a nivel de sentido de WordNet completo). En una de las categorías, *location*, el aumento de la precisión se estanca en relación con el volumen de datos de entrenamiento. En *psychological_feature*, la curva toma un claro camino descendente. Es decir, que cuanto más datos de entrenamiento se suministraban, peor se comportaba el algoritmo.

Otro problema de este cuello de botella de la adquisición del conocimiento radica en que la mayoría de las aproximaciones supervisadas al problema de la DSP necesitan ejemplos para cada sentido de cada palabra. Las distribuciones estadísticas de los sentidos son muy sesgadas, de forma que es difícil encontrar ejemplos de entrenamiento para los sentidos menos habituales. Este problema se pone aún más de manifiesto si tenemos en cuenta la ley de Zipf (Zipf, 1945), que nos viene a decir que la mayoría de las palabras son muy poco frecuentes en términos relativos en cualquier texto (aunque sea muy largo). Es decir, hay un fuerte sesgo, tanto a nivel de palabra como a nivel de sentido, que impone un límite superior natural al rendimiento de los sistemas *genuinamente* supervisados.

2.1.2. Granularidad de la distinción de sentidos

En su tesis doctoral (Kilgarriff, 1992), investiga los criterios que mueven a los lexicógrafos a distinguir varias entradas (es decir, varios sentidos) en un diccionario para una misma palabra. La conclusión es que los criterios no son fijos, ni siquiera consistentes en un mismo diccionario. Un caso frecuente es de la polisemia sistemática, por ejemplo, entre una manzana como fruto y como alimento, presente en muchas otras frutas; o en una inauguración, entre el acto y el evento, otra metonimia bastante habitual. El principio que puede llevar a distinguir ese tipo de ambigüedad es el llamado SFIP (Sufficiently Frequent, Insufficiently Predictable)[1]. La palabra banco puede referirse a la compañía o al edificio donde se encuentra parte de la compañía, pero al ser una distinción predecible (según el patrón metonímico institución/edificio) no merecería una entrada específica en muchos diccionarios.

La granularidad de sentidos adecuada puede variar en gran medida dependiendo del

[1]Suficientemente Frecuente, Insuficientemente Predecible

tipo de aplicación que tengamos en mente; para sistemas de inteligencia artificial en busca de la comprensión del lenguaje natural, es preciso tener en cuenta la granularidad más fina que sea posible, sin embargo, para traducción automática dependerá por entero de la distancia conceptual entre los idiomas (varios sentidos distintos de una misma palabra pueden corresponder a una misma traducción en el idioma destino, de tal forma que la palabra no sea ambigua por lo que respecta a la tarea, mientras que para idiomas de familias distintas la ambigüedad léxica que se deberá tomar en consideración será mayor). En categorización de textos y en recuperación de información se considera que un nivel de granularidad demasiado fino puede perjudicar más que ayudar. En este sentido en (Kilgarriff, 1997) se mantiene la postura que los sentidos de las palabras no existen *per se*, sino que no adquieren existencia hasta que se haya definido una tarea que dependa de ellos.

De cara a la evaluación, sin embargo, es evidente que resulta mucho más sencillo en términos de precisión numérica considerar distinciones de grano grueso, aunque las conclusiones obtenidas son muy difícilmente extrapolables al grano fino. Podemos encontrar trabajos exitosos sobre desambiguación sobre distinciones binarias de sentidos (dos sentidos de dominios distintos en la mayoría de los casos) en (Karov and Edelman, 1996; Karov and Edelman, 1998; Schütze, 1992b; Schütze, 1998; Yarowsky, 1992; Yarowsky, 1995b; Niwa and Nitta, 1994) entre otros, con resultados en la franja del 90-99 % de precisión.

2.1.3. Un sentido por discurso

Una teoría muy interesante de cara a la desambiguación es la de *un sentido por discurso*, presentada en (Gale et al., 1992b). Esta teoría se basó en unos experimentos llevados a cabo con textos de la enciclopedia Grolier (Grolier Inc., 1991). El experimento consistió en tomar 82 pares de concordancias de 9 palabras polisémicas distintas. Una concordancia es simplemente una linea de texto de longitud indeterminada en la que aparecen una o más palabra que nos interesan. 54 pares están tomados del mismo discurso (artículo de la enciclopedia Grolier) y 28 pares de control no. Los pares de control están ahí para asegurar que ninguno de los jueces diga que todos los pares están relacionados sin tomarse el trabajo de leer las concordancias.

Cinco sujetos establecieron, para cada par, si la palabra está utilizada en las dos concordancias en el mismo sentido o no. Después se midió el acuerdo (agreement) entre cada anotador y la mayoría de los demás. Se alcanza un acuerdo medio del 96.8 %. Como además todas las palabras del estudio son polisémicas, se estima que

en texto libre (en el hay una cantidad importante de palabras no ambiguas) este porcentaje alcanzaría el 98 %. Se extrajo la conclusión de que las palabras polisémicas se utilizan, en un porcentaje de casos muy alto, en el mismo sentido dentro de una misma unidad discursiva (sin especificar su longitud, dominio u otras características).

Esta conclusión resulta bastante atrevida, si tenemos en cuenta el escaso número de palabras empleadas en el experimento y que las distinciones de sentidos son binarias y con distinciones de sentidos temáticas. La sorprendente conclusión es que *quizás el problema de la desambiguación del sentido de las palabras sea más fácil de lo que hasta ahora podría haberse pensado.*

Esta hipótesis fue puesta en duda en (Krovetz, 1998). Su experimento consistió en medir en un texto ya anotado semánticamente de forma manual, las apariciones de palabras con distintos sentidos. El inventario de sentidos elegido fue WordNet, y el corpus tanto el SemCor (Francis and Kucera, 1967) como el DSO (Ng and Lee, 1996). Es decir, se consideraron distinciones de grano finas para muchas palabras. Los resultados fueron contundentes: el 33 % de las palabras que aparecen más de una vez en un *discurso* se emplean en más de un sentido. De las 191 palabras distintas del DSO, todas mostraban sentidos distintos en algún documento. La conclusión fue que la hipótesis de (Gale et al., 1992b) es probablemente correcta para sentidos homónimos, pero que no se puede aceptar la sugerencia del mismo artículo de que para desambiguar todas las apariciones de una palabra en un discurso basta con desambiguar una aparición y asignar a todas el mismo sentido.

2.1.4. Contexto local vs. contexto amplio

El contexto de una palabra ambigua nos da la clave para su desambiguación, ¿pero cuanto contexto es conveniente considerar? Se suele distinguir entre *contexto local* (i.e. perteneciente a determinado lugar) y *contexto amplio*. El contexto local sería la frase en cuestión o unas pocas frases adyacentes. El contexto amplio puede extenderse hasta cientos de palabras de distancia.

(Choueka and Lusignan, 1985) demostraron que las personas podían desambiguar con bastante fiabilidad con un contexto de dos palabras por cada lado. Este resultado ha sido muy influyente en la literatura posterior en desambiguación.

(Gale et al., 1993) consideraron un contexto de 50 palabras a un lado y otro de la palabra que querían desambiguar. Los autores conocían el trabajo de (Choueka and Lusignan, 1985). Sin embargo comprobaron en su corpus que su método seguía

teniendo una precisión mayor que la elección aleatoria usando contextos de longitud del orden de las decenas de millar. Se realizan dos tipos de experimentos, en uno se mide hasta donde se extienden las relaciones semánticas (usando sólo los bordes del intervalo) y en otro, se mide la amplitud, esto es, se consideran todas las palabras en el interior del intervalo. Los mejores resultados se producen empleando contextos de longitud 101. Los autores concilian las tesis de Choueka con las suyas argumentando que aunque los humanos no necesitan la información del contexto amplio, eso no quiere decir que los algoritmos no puedan beneficiarse de ella.

Otros experimentos han utilizado el contexto de la frase como (Thanopoulos et al., 2000) o una mezcla de contexto local y contexto amplio (Chodorow et al., 2000).

2.2. Recursos léxicos para DSP

En esta sección seguiremos estrechamente la aproximación de (Gonzalo et al., 2004). En la investigación en DSP se han utilizado tradicionalmente dos tipos bien diferenciados de recursos; por un lado tendríamos los inventarios de sentidos, sean diccionarios electrónicos o bases de conocimiento léxicas. Por el otro, tendríamos colecciones de textos (corpora) ya sean de texto plano (sin anotaciones) o bien anotados con información semántica de algunos de los inventarios mencionados.

2.2.1. Inventarios de sentidos

Diccionarios en formato electrónico

Los diccionarios electrónicos en su versión más básica no son más que lo que el propio término indica, la información que se presenta en el papel se presenta igualmente en formato electrónico. Siguiendo un grado de sofisticación creciente, pueden llegar a incluir todo de información del tipo tradicional pero también otra imposible de encontrar en las versiones de papel, como opciones de búsquedas por formas flexionadas, lematización, pronunciación, etc...

Los diccionarios electrónicos más utilizados en la investigación en DSP son:

LDOCE - Longman Dictionary of Contemporary English Utilizado profusamente en DSP, tiene 55000 entradas de definiciones de palabras. Con el objeto

de asegurar la consistencia de dichas definiciones, éstas se escriben (al menos en principio) utilizando un conjunto cerrado de unas 2000 palabras. Este vocabulario controlado se conoce como *Vocabulario de definición de Longman*. Otra característica de LDOCE es la inclusión en algunas de las entradas de los sentidos de las palabras de etiquetas de actividad (*subject field*). Algunas entradas en LDOCE contienen también etiquetas de dominio (*box codes*), clasificadas jerárquicamente. Otra característica interesante es que los sentidos están agrupados, permitiendo diversos grados de granularidad (un sentido general puede tener subsentidos). Este hecho hace que con frecuencia los experimentos basados en LDOCE presenten resultados a nivel de grano *fino*, a nivel de sentido y también de grano *grueso*, a nivel de grupo de sentidos. Ejemplos de investigaciones en DSP con LDOCE serían (Guthrie et al., 1991; Luk, 1995; Wilks and Stevenson, 1996; Wilks and Stevenson, 1997a; Stevenson et al., 1998; Stevenson and Wilks, 1999)

COBUILD - Collins Cobuild English Language Dictionary Este diccionario fue desarrollado por el departamento de inglés de la universidad de Birmingham y la editorial Harper Collins con el propósito de representar el léxico nuclear de la lengua inglesa.

NODE - New Oxford Dictionary of English Contiene 170000 entradas que cubren todas las variedades del inglés. Está disponible en XML y SGML, incluye sintagmas y expresiones idiomáticas (frases hechas), así como relaciones semánticas y etiquetas de dominio de unos 200 tipos.

HECTOR Además de productos comerciales, existen a su vez diccionarios que han sido creados con el único fin de la investigación. Ese el caso de HECTOR, uno de los resultados del proyecto del mismo nombre, que fue utilizado como inventario de sentidos en la primera edición de la competición SENSEVAL, de la que hablaremos más adelante.

La *distancia* semántica entre los sentidos de las palabras polisémicas está reflejada en el sistema de numeración de los sentidos. Algunos sentidos son más específicos que otros, de manera que se forma una jerarquía entre ellos.

La principal ventaja de los diccionarios electrónicos es que son inventarios de sentidos creados manualmente. Sin embargo, también existen problemas respecto a su uso.

En primer lugar, las definiciones escritas por personas, especialmente si el equipo de lexicógrafos es numeroso, contienen con mayor frecuencia de lo que sería deseable inconsistencias, redundancia, circularidad o incluso errores tipográficos, que complican

su procesamiento automático. El formato electrónico permite controlar mejor todo tipo de errores. Aún así, la falta de una estructura de entrada rigurosa y, especialmente, un gran número de definiciones circulares o inconsistentes todavía constituye un problema a la hora de utilizar los diccionarios en formato electrónico para aplicaciones de procesamiento del lenguaje natural, en particular en tareas de desambiguación.

Las distinciones de sentidos son ciertamente arbitrarias, dependiendo de la experiencia y preferencias de los lexicógrafos individuales (*separadores vs. aglutinadores*), lo que lleva a un escaso grado de solapamiento entre diccionarios distintos por lo que se refiere a inventarios de sentidos.

Finalmente, a pesar de su nombre, los diccionarios en formato electrónico generalmente están orientados al uso humano. Una parte considerable de las definiciones se apoyan en la capacidad del lector para hacer inferencias y en la habilidad humana para llenar posibles vacíos con conocimiento lingüístico o incluso enciclopédico. Esta es una de las razones por las cuales hacer un uso automatizado de los diccionarios en formato electrónico no es todavía inmediato.

Tesauros

Los objetos léxicos están organizados en nuestro léxico mental tanto formalmente como conceptualmente. A diferencia de los diccionarios, que organizan las palabras alfabéticamente, los tesauros explotan otro aspecto; la organización conceptual de las palabras.

Hoy en día existe un gran número de tesauros de variados grados de cobertura y diferente audiencia. Algunos de ellos son de dominio general, otros, temáticos, se especializan en dominios concretos, ofreciendo vocabularios controlados. Todos ellos están basados en algún tipo de sistema de clasificación, y los sentidos de las palabras se distribuyen entre las categorías que forman parte de dicha clasificación. Dentro de cada categoría, los términos están agrupados básicamente por sinonimia, aunque algunos tesauros recurren a otras relaciones semánticas (v.g. hiperonimia, antonimia), para una mejor organización de los términos sinónimos. Uno de los más conocidos es el **Roget's International Thesarus** La quinta edición contiene 325000 entradas organizadas en 15 clases y 1073 categorías. Dentro de estas categorías, las ideas están separadas. Los términos extranjeros y técnicos están marcados de forma especial. No parece haber versiones electrónicas de este recurso disponibles para el público.

Bases de Conocimiento Léxicas

Durante los años 90 se ha dedicado mucho esfuerzo a la creación de diversas bases léxicas de conocimiento (LKB's): CyC (Lenat, 1995), ACQUILEX (Briscoe, 1991), COMPLEX (Grisham et al., 1994). Sin embargo, WordNet (Miller, 1995) sigue siendo la base de conocimiento léxica por antonomasia.

WordNet WordNet (WN), uno de los recursos léxicos más empleados en procesamiento del lenguaje natural, es una base de datos léxica a gran escala para el idioma inglés desarrollada por el laboratorio de ciencias cognitivas de la Universidad de Princeton. En su versión 1.7, WordNet cubre 136972 palabras o términos multipalabra que corresponden a 192460 conceptos lexicalizados, incluyendo cuatro categorías, nombres (132407), verbos (23255), adjetivos (31077) y adverbios (5721).

WN comparte algunas características con los diccionarios monolingües. Sus glosas y ejemplos de uso proporcionados para cada sentido de las palabras se parecen a las definiciones de los diccionarios. Sin embargo, WN ofrece una información mucho más rica a nivel de sentido. Basada en principios psicolingüísticos, una de las premisas implícitas en WordNet es que los conceptos lexicalizados pueden ser organizados por relaciones semánticas. La red de relaciones semánticas constituye la principal ventaja de usar WordNet en tareas de PLN como anotación semántica o recuperación de información.

El diseño de WordNet se basa en *synsets* o conjuntos de variantes lexicalizadas asociadas al mismo concepto. Un ejemplo de esto podría ser en español {casa, hogar}; un concepto ejemplificado por dos palabras distintas que sin embargo pueden aparecer en otros synsets como {casa,edificio,construcción}, por ejemplo. Estos conjuntos de sinónimos (equivalentes a sentidos de palabras) están enlazados unos con otros por medio de relaciones semánticas.

El número de relaciones consideradas en WordNet es bastante limitado. Aparte de la sinonimia, implícita en la noción de synset, hay otra relación semántica que juega un papel esencial en la estructura de WordNet: la hiperonimia. Esta relación *es-un* engloba a todos los synsets y ayuda a organizar jerárquicamente los sentidos de los nombres y los verbos. Cada synset de nombres o verbos en WordNet, excepto aquellos que son los más genéricos de su clase, tienen al menos un hiperónimo. Otras relaciones presentes en WN son la meronimia y la antonimia, aunque lógicamente no se aplican a todos los synsets.

La organización jerárquica de los conceptos sitúa a WN cerca de las ontologías. A pesar de ello, en contraste con las ontologías inferenciales (de sentido común),

WN está enfocada esencialmente sobre el conocimiento léxico, i.e., en representar unidades léxicas relacionadas con conceptos. Esto explica la falta de niveles artificiales (no lexicalizados) como los que aparecen en ontologías sobre el conocimiento general para posibilitar ciertas inferencias.

En principio, las diferentes partes del discurso están separadas en la red semántica de WordNet, así que las relaciones semánticas son casi exclusivamente entre palabras de la misma categoría gramatical. Esto se justifica en parte por el hecho de que el inventario de relaciones semánticas varía de una categoría a otra. En el caso de los nombres y de los verbos, se forman cadenas de hiperónimos.

Como ejemplo de investigaciones en DSP que han utilizado WordNet, podríamos citar, sin ánimo de ser exhaustivos a (Agirre and Martinez, 2001; Agirre and Rigau, 1995; Agirre and Rigau, 1996; Voorhees, 1993; Sussna, 1993; Chodorow et al., 2000; Cucchiarelli et al., 2000; Dini et al., 1998; Dorr and Jones, 1996; Fellbaum et al., 1997; Haynes, 2001; Krovetz, 1998; Kwong, 2001; Lin, 1997; Mihalcea and Moldovan, 2000a; Mihalcea and Moldovan, 1999; Mihalcea and Moldovan, 2000b; Moon, 2000; Ng and Zelle, 1997; Ng, 1997a; Ng and Lee, 1996; Montoyo and Suárez, 2001; Stevenson and Wilks, 1999). A partir de la segunda edición de la competición SENSEVAL, WordNet se ha afianzado aún más como recurso léxico estándar.

EuroWordNet Una manera de ver EuroWordNet es como una extensión multilingüe de WordNet. La base de datos EuroWordNet (EWN) está formada por bases de datos *à la WordNet* sobre el inglés, español, alemán, holandés, italiano, francés, estonio checo. Cada una de estas bases de datos está conectada a la principal a través del *Índice Interlingua*.

EWN ha sido desarrollada como un conjunto de módulos independientes. Cada WordNet monolingüe ha sido construido por separado, utilizando diferentes recursos disponibles para cada lengua determinada. La conectividad a través de estos sistemas autónomos y específicos al idioma se aseguró por medio de un Índice Interlingua (ILI), que representa el superconjunto de todos los conceptos que aparecen en cada lengua de EWN. Cada synset de un WordNet particular tiene al menos un enlace con un registro del ILI. De esta forma, los synsets de idiomas específicos están conectados con sus equivalentes en otros idiomas vía ILI.

La combinación WordNet/EuroWordNet proporciona un inventario de sentidos con una gran cobertura del idioma inglés, que incorpora relaciones semánticas con una larga tradición en DSP como veremos en el siguiente capítulo, y además está dotado de grandes posibilidades multilingües gracias a los índices interlingua. Además es un

inventario de sentidos muy popular por lo que resulta posible comparar directamente otros resultados de investigación. Por todos estos motivos elegidos WordNet como recurso léxico para nuestros experimentos.

2.2.2. Corpora

Los corpora pueden caracterizarse como una fuente de información de primera mano sobre el idioma en cuestión (o los idiomas en cuestión si se trata de corpora multilingües). Representan una evidencia directa sobre la frecuencia y la coocurrencia de elementos lingüísticos que no puede ser extraída con la misma fiabilidad de diccionarios u otros inventarios de sentidos.

Corpora planos

Este grupo está formado por registros de textos escritos u orales no anotados semánticamente. La carencia de cualquier distinción preestablecida de sentidos los sitúa como una fuente *pura* de conocimiento lingüístico.

El Brown Corpus Se trata de una colección de fragmentos de textos recopilados en Estados Unidos en 1961 por Francis Kucera (Francis and Kucera, 1967). El propósito real de crear dicho corpus fue el de proporcionar una colección de prueba de inglés real para hacer estudios comparativos. Se pensó en él como en un estándar contra el cual se podrían comprobar diversos trabajos de investigación.

El Brown Corpus amalgama ejemplos de prosa escrita, en una escala que va desde una variedad de artículos de prensa (noticias, críticas, reportajes), hasta fragmentos de textos científicos y de ficción, clasificados en quince categorías. El número total de documentos es 500, cada uno de ellos de aproximadamente unas 2000 palabras.

El Brown Corpus ofrece un material cuidadosamente escogido y, durante muchos años, ha sido prácticamente el único corpus disponible para los investigadores en PLN, aunque, actualmente se considera que es pequeño y está relativamente caduco.

El British National Corpus El *British National Corpus* (BNC), es el resultado de un trabajo conjunto entre editoriales de diccionarios (OUP, Longman,

Chambers-Larousse) y centros de investigación académica (Oxford University, Lancaster University y la British Library).

Esta compuesto de fragmentos de 4124 textos de inglés británico moderno que representa un amplio abanico de géneros, estilos y variedades, escritas y orales. El término *moderno* se refiere vagamente a finales del siglo XX.

Los textos están segmentados en frases y las palabras están anotadas morfológicamente. El corpus contiene más de 100 millones de palabras.

El BNC ha sido construido como un corpus relativamente equilibrado: para las fuentes escritas, se han tomado muestras de 45000 palabras de varias partes de textos de un mismo autor. Los textos más cortos, hasta de 45000 palabras han sido incluidos completos, así como los textos de varios autores como revistas y periódicos.

Proyecto Gutenberg Aparte de esos corpora usados tradicionalmente por lingüistas, otras opciones han ido abriéndose camino en los últimos años. Una de las que la ofrece un mayor potencial de crecimiento es la extracción de corpora de Internet. Con ser la fuente más inmediata, la World Wide Web no es la única ni opción, ni la menos problemática. El Proyecto Gutenberg[2] ha conseguido trasladar, gracias al trabajo de voluntarios, a formato electrónico unos 5000 libros cuyos derechos de autor han prescrito y que pueden ser por tanto redistribuidos libremente. Este esfuerzo ha producido una colección textual de un tamaño muy superior al de los corpora anteriormente mencionados.

Corpora anotados semánticamente

Los corpora anotados semánticamente son la fuente de recursos principal por definición para los métodos supervisados del sentido de las palabras y la única manera de probar y comparar sistemas de DSP en general.

SemCor SemCor (Miller et al., 1993) es todavía el mayor y más conocido corpus de disponibilidad pública anotado semánticamente. Está compuesto de documentos extraídos del Brown Corpus que fueron anotados tanto morfológicamente como semánticamente. Mientras que las etiquetas morfosintácticas fueron asignadas por el anotador de Eric Brill, el anotado semántico fue realizado de forma manual, sobre los sentidos de WordNet.

[2] http://promo.net/pg

Puesto que WN sólo comprendía en esa época categorías sintácticas abiertas, sólo nombres, verbos, adjetivos y adverbios fueron anotados semánticamente en SemCor. Dentro de los nombres, a los nombres propios se les clasificó dentro de cuatro posibles categorías, *persona*, *lugar*, *grupo* u *otro*.

Las anotaciones semánticas utilizadas constituyen atributos de elementos SGML. El número total de ficheros con todas las categorías sintácticas abiertas anotadas es de 186, también hay otros 166 ficheros en los que únicamente se han anotados los verbos (mucho más difíciles de desambiguar como es público y notorio).

El número total de sentidos de WN asignado a nombres, verbos, adjetivos y adverbios presentes en SemCor es de 50140. Hay 234136 formas léxicas anotadas. En 703 casos se asignó más de un sentido a una forma léxica.

DSO En 1996 un nuevo corpus anotado semánticamente fue compilado por la Organización de la Ciencia del Ejército (DSO) de Singapur. Contiene textos del Brown Corpus y del Wall Street Journal. Contiene 192800 términos anotados a nivel de sentido de los 121 nombres y 70 verbos que según los autores representan las palabras más ambiguas más frecuentes en inglés. Las anotaciones fueron realizadas por estudiantes de la universidad de Singapur. Los sentidos están anotados con respecto a WordNet-1.5.

Parte de los datos son documentos del Brown Corpus que también están incluidos en SemCor. A pesar de que ambos corpora fueron anotados con el mismo inventario de sentidos, el nivel de acuerdo entre los anotadores de DSO y de SemCor es sólo del 57 %. Como veremos en el siguiente capítulo es un hecho conocido que bajo ciertas condiciones los anotadores humanos pueden tener considerables diferencias de criterio, más aún si se trata de anotar las palabras más polisémicas de un idioma, como en este caso. Con todo el grado de discrepancia resulta muy considerable.

Colecciones de SENSEVAL En sus dos ediciones hasta el momento, en 1998 y en 2001, la competición SENSEVAL ha hecho públicos los datos de sus tareas de desambiguación para el inglés. En la primera edición se produjo una colección de 8448 casos de prueba para 35 palabras ambiguas usando HECTOR como inventario de sentidos. En la segunda edición se produjo una colección de 8611 ejemplos extraídos del BNC anotados con sentidos de WordNet, sobre apariciones de 73 palabras polisémicas. Estos datos se publicaron como datos de entrenamiento. También se produjo una colección para probar los sistemas participantes de esas mismas 73 palabras, en total 4328 contextos. Las anotaciones se revelaron después de que los participantes enviaran las respuestas de

sus sistemas. Esta última colección formó la tarea de *muestra léxica*, a la que aludiremos con frecuencia en esta tesis. También existe una colección de 2473 palabras anotadas tomadas del corpus Penn Treebank. Esta colección sirvió para la tarea de *todas las palabras*, puesto que se trataba de desambiguar todas las palabras de tres artículos. También haremos referencia frecuentemente a esta colección.

2.3. Evaluación de los sistemas de DSP

En el mundo de la DSP se manejan tres conceptos fundamentales a la hora de evaluar un sistema de desambiguación. Son la *precisión*, la *cobertura* y el *recall*. Se parte de la base de que se posee una colección anotada manualmente por personal competente contra la que probar los resultados de un desambiguador. Se admite que una palabra pueda ser utilizada en más de un sentido (es decir, que más de un sentido sea correcto). También es permisible que un desambiguador opte por no desambiguar una o varias palabras. En un principio era habitual definir la precisión como

$$\frac{\text{número de ejemplos correctamente desambiguados}}{\text{número total de ejemplos desambiguados}}$$

la cobertura como

$$\frac{\text{número de ejemplos desambiguados (de forma correcta o no)}}{\text{número total de ejemplos}}$$

y el recall como

$$\frac{\text{número de ejemplos correctamente desambiguados}}{\text{número total de ejemplos}}$$

de esta manera se daba la relación $\text{cobertura} \cdot \text{precision} = \text{recall}$.

Nosotros adoptaremos en buena medida la metodología de evaluación utilizada en las competiciones SENSEVAL (véase la sección anterior). La competición se divide en varias tareas. Para cada tarea, con alguna excepción como el caso del idioma japonés,

se define una serie de apariciones de palabras perfectamente identificadas. El objetivo de un sistema de desambiguación es desambiguar correctamente estas palabras. Los desambiguadores devuelven para cada palabra uno o varios de sus sentidos como correctos, repartiendo opcionalmente un peso normalizado entre ellos. Por ejemplo se puede otorgar una confianza del 70 % al primer sentido, una del 12 % al segundo y una del 18 % al tercero. Lo habitual es normalizar estos pesos a uno. Con esta forma de proceder se pueden calcular las siguientes medidas:

$$\text{precisión} = \frac{\sum_{i \in C} \text{suma de las puntuaciones de los sentidos correctos de i}}{\text{total de palabras desambiguadas}}$$

$$\text{recall} = \frac{\sum_{i \in C} \text{suma de las puntuaciones de los sentidos correctos de i}}{\text{cardinal del conjunto C}}$$

Donde C sería el conjunto de palabras por desambiguar. En esta nueva aproximación el concepto de cobertura está menos claro. Podría seguir definiéndose como en el caso anterior, pero la relación entre precisión, cobertura y recall dejaría de cumplirse. Lo único seguro es que el recall es menor o igual a la precisión.

La precisión es una medida valiosa sobre estrategias de desambiguación aisladas, pero un sistema final de DSP debe medirse por su recall. Un sistema con una precisión muy alta puede ser inútil si la cobertura es baja, mientras que un sistema con un recall alto siempre es bueno.

Se han propuesto otros métodos de evaluación como puede verse por ejemplo en (Resnik and Yarowsky, 1997; Resnik and Yarowsky, 1999). Entre otras propuestas se incluye la de calcular una matriz de distancia entre sentidos de manera que si un sistema devuelve un sentido equivocado, pero cercano al correcto, todavía consiga alguna puntuación. También se propone una forma concreta de construir esta matriz, dependiendo de si las traducciones de cada par de sentidos a un segundo idioma coinciden o no. Un ejemplo de matriz de este tipo puede encontrarse en (Chugur et al., 2002).

Capítulo 3

Revisión de sistemas de DSP

3.1. Sistemas de DSP no supervisados

Nuestro interés se centra en los algoritmos no supervisados por lo que haremos un mayor hincapié en ellos.

3.1.1. Desambiguación basada en pseudopalabras

Una pseudopalabra es una palabra artificial creada uniendo dos o más palabras de un idioma, por ejemplo: pastel/calavera. Este controvertido concepto fue introducido en (Gale et al., 1992c) como un intento de superar el cuello de botella de adquisición de ejemplos para construir conjuntos de prueba y entrenamiento. La idea era que, si se parte de un texto, se pueden crear pseudopalabras aleatoriamente y utilizar el texto original como ejemplos anotados de pseudopalabras que sirvieran de entrada a un clasificador automático basado en algún tipo de algoritmo de aprendizaje. Se pensaba que que este algoritmo de aprendizaje, con una cantidad prácticamente infinita de ejemplos de entrenamiento podría dar las claves para realizar la desambiguación real. Hay, sin embargo, un problema latente y es que mientras que las pseudopalabras se desambiguan a palabras, las palabras se desambiguan a sentidos, entes radicalmente distintos.

Probablemente nadie pensara que los algoritmos entrenados con pseudopalabras pudieran servir tal cual para desambiguar palabras auténticas, pero sí que podrían ayu-

dar a dar con un conjunto reducido de características observables en las pseudopalabras y sus contextos que permitieran ser aplicadas a la desambiguación de palabras reales. Un uso habitual de las pseudopalabras ha sido el de servir para construir una colección de prueba *barata*, para evaluar un algoritmo de desambiguación genérico (i.e. que sirviera tanto para palabras normales como para pseudopalabras).

Esta estrategia parecía prometedora tras los experimentos de (Yarowsky, 1995a) sobre la *restauración* de acentos en francés y tildes en español. La forma de trabajar consistía en tomar textos reales (i.e. con sus respectivos tildes y acentos) y entrenar listas de decisión (Rivest, 1987) sobre ellos, apoyándose en el método de decisión de (Mosteller and Wallace, 1964) y también árboles de decisión (Brown et al., 1991). Yarowsky obtuvo unos resultados muy positivos, aunque la dificultad del problema no parece demasiado elevada. La clave del éxito es que sus pseudopalabras en este caso eran las mismas palabras, a las que se había eliminado los acentos, de manera que compartían muchas características con las palabras reales. Basta con echar un vistazo a la pseudopalabra banana/kalashnikov/teletubbie para comprender que en el caso de la DSP las pseudopalabras son demasiado diferentes de las palabras como para poder evaluar características similares. Sin ir más lejos, fijarse en el final de las palabras (una característica muy influyente para los acentos en español) en el caso de las pseudopalabras no es que no resulte informativo, es que se intuye que no va a resultar muy práctico.

En cualquier caso la forma de desambiguar es la siguiente: Se escoge un conjunto de características y se evalúan sobre un conjunto de entrenamiento, así se generan las listas de decisión y se ordenan las reglas por su *verosimilitud logarítmica*, esto es:

$$\log \frac{P(\text{sentido}_1 \mid \text{expresión textual})}{P(\text{sentido}_2 \mid \text{expresión textual})}$$

Por ejemplo, supongamos que encontramos en nuestro texto ambiguo *maniqueo/desleal* y supongamos también que hemos elegido como características (features) de desambiguación, entre otras, las palabras con contenido anterior y la posterior, esto es, trigramas. Con nuestro texto original, podemos buscar las apariciones de maniqueo y desleal y ver cuáles son las palabras con contenido que la rodean.

Supongamos, además, que en nuestro texto original, no ambiguo respecto de las pseudopalabras, encontramos, descartando las palabras sin contenido, 27 veces la expresión *demagogo maniqueo trasnochado* y 35 la expresión *demagogo desleal trasnochado*. Tendríamos entonces dos reglas de decisión sobre la expresión *demagogo maniqueo/desleal trasnochado*. Una de ellas en favor del significado maniqueo, con

una verosimilitud logarítmica de $\log 27/35$. Otra en favor del significado desleal, con una verosimilitud logarítmica de $\log 35/27$, obviamente mayor. Si sólo utilizamos características de este tipo las listas de decisión no parecen muy sofisticadas pero si aumentamos el número de características (por ejemplo, podríamos añadir como característica la presencia de palabra con contenido en un radio de 20 palabras, lo que nos proporcionaría montones de reglas adicionales a las que aportarían los trigramas), entonces el método resulta mucho más versátil.

Lo importante es que estas reglas se ordenan por su verosimilitud logarítmica de mayor a menor, para ser aplicadas. La primera regla que resulte aplicable es la que nos da la solución. Las características contextuales y la forma de estimar las probabilidades escogidas son las que nos darán el número de reglas de desambiguación resultante. Es destacable el hecho de que se decide siempre en función de la primera regla que resulte aplicable, no se realiza ningún tipo de combinación de posibles reglas aplicables (si hay varias reglas, se aplica únicamente la de mayor verosimilitud logarítmica).

Tampoco se mezclan las características de desambiguación, es decir, cada regla sólo considera una característica (habitualmente, la presencia de otras palabras ya sea en posiciones relativas fijas, en ventanas de un tamaño prefijado o en forma de bigramas o trigramas).

Estas verosimilitudes logarítmicas provocan problemas en los casos en los que alguna de las probabilidades es cero. Para evitar dichos problemas se aplican técnicas de suavizado (*smoothing*), como por ejemplo la presentada en (Gale et al., 1992b).

Otra contribución interesante es que se crean clases genéricas de ambigüedad, de manera que se generan reglas de desambiguación para la clase completa. Esto es, para problemas genéricos en las tildes, como la ambigüedad -ara/ará en español (pretérito imperfecto de subjuntivo/futuro imperfecto de indicativo), se crean clases temporales como la de los días de la semana o los meses. La idea es que la cercanía de expresiones temporales podría ser productiva a la hora de discriminar el uso del futuro respecto del subjuntivo.

La idea de las clases de palabras podría ser generalizable a DSP, al crear clases de palabras cuyas interacciones a nivel semántico se consideraran similares. Esto redunda en un número menor de reglas a considerar y en una estimación más acertada (al estar los datos menos dispersos, el texto original *de entrenamiento* sirve mejor para estimar las probabilidades).

El método consigue un acuerdo con el texto original del 99.6 % cuando la heurística de la opción más frecuente consigue uno del 98.7 %. En ambos casos nos movemos en

márgenes peligrosos. Yarowsky menciona que los textos reales tienen mal colocados los acentos con bastante frecuencia. Si entrenamos un método basado en anotaciones humanas y al final el método es más eficaz que los humanos, ¿cómo podemos interpretar eso?

Además, las distinciones son binarias en la mayoría de los casos, por lo que la técnica no parece que vaya a transportarse demasiado bien a DSP sobre WordNet por ejemplo, teniendo en cuenta los comentarios de (Fellbaum et al., 1997) sobre la dificultad de la tarea de DSP cuando el número de opciones para elegir aumenta (la función que nos da la dificultad según el número de opciones no crece de forma lineal sino que pertenece a un orden de complejidad superior). Esto no se notará en la aplicación de Yarowsky a DSP porque también allí él empleará distinciones binarias relativamente sencillas de desambiguar.

En (Yarowsky, 1999), se compara este método basado en listas de decisión con un anotador basado en n-gramas y otro bayesiano y el de las listas de decisión resulta ser el mejor. En cualquier caso, el hecho es que la mayoría de las palabras son no ambiguas en lo que respecta a los acentos (hay muchas palabras en español y en francés que o bien llevan algún tipo de acento o no lo llevan, pero no existen ambas posibilidades, se podrían resolver en su mayoría con un diccionario y un analizador morfológico) por lo que el texto original y el texto con las pseudopalabras (en este caso, con los acentos eliminados) no difiere demasiado, mientras que un ligero porcentaje de pseudopalabras en DSP hace que el texto resulte muy poco comprensible o directamente incomprensible. La prueba de ello es que en los primeros sistemas de traducción automática la desambiguación se dejaba a cargo del lector, de modo que cuando una palabra era ambigua se presentaban las distintas traducciones posibles y el resultado era bastante ilegible. Un ejemplo de esto sería el ejemplo de esta traducción del ruso al inglés de un artículo sobre química, tomado de (Perry, 1955):

Saccacharification cellulose begin use/employ in/into/at techonology/technique. For what waste product (wood processing/wood working) plant/factory heat under/below pressure with/from 0.1 % solution sulfuric/sulfate acid; obtained such means/way syrup process/convert on/at wine/tartaric alcohol. (According to/Along/In accord with) other processes saccharification (accomplish/carry out) on/at cold action very strong (sp. weight 1.21) salt/hydrochloric acid. After removal acid, remain solid product being used as/how food/forage medium/means.

Otro problema con las pseudopalabras es que las características estudiadas en las palabras pueden perderse o degradarse. En el caso de los acentos es posible fijarse en los finales de las palabras sin grandes pérdidas atribuibles a la desaparición de los

acentos, o puede buscarse la presencia cercana de palabras de una lista de expresiones de tiempo (indicativas de futuro más que de subjuntivo como en la distinción esperará/esperara). Sin embargo en las pseudopalabras aplicadas a DSP el final de las palabras es sólo el final de la última palabra de la pseudopalabra y análogamente otras características se pueden fácilmente perder entre el ruido introducido. Además la ambigüedad introducida por las pseudopalabras (donde una vez elegida la cantidad de ambigüedad que se introduce, las variantes son elegidas al azar) por simple probabilidad suele ser de una naturaleza temática clara, lo cual no ocurre con tanta frecuencia en la ambigüedad natural.

(Schütze, 1998) evaluó su algoritmo de desambiguación contra diez pseudopalabras y también contra palabras *naturales*. Entre sus conclusiones destaca el hecho de que desambiguar pseudo-palabras es más fácil puesto que las distinciones de sentidos son más claras que en las palabras reales.

En (Sanderson, 1994) las pseudopalabras se utilizaron aplicadas a la recuperación de información. Las conclusiones del autor fueron que la desambiguación del sentido de las palabras ofrecía una pérdida de efectividad en la recuperación salvo que se hiciera con un alto grado de éxito. El hecho de que la ambigüedad natural sea distinta de la introducida por las pseudopalabras fue esgrimido en otros lugares contra estas conclusiones, por ejemplo (Gonzalo et al., 1999; Sanderson, 2000).

En un estudio sobre la relación entre la ambigüedad natural y las pseudopalabras (Gaustad, 2001), la autora concluye que: *Las pseudopalabras no pueden ser tomadas como un sustituto para las palabras ambiguas reales*. Sus razones están basadas en un experimento en el que se aplicaba el mismo algoritmo de desambiguación (supervisado) a palabras reales y pseudopalabras. Los resultados no eran comparables.

3.1.2. Desambiguación basada en coocurrencias de palabras

En (Church and Mercer, 1993) se hace un interesante repaso de las áreas del PLN que se han beneficiado de la creciente disponibilidad de grandes corpora, como reconocimiento de voz, reconocimiento de caracteres (OCR), anotación morfosintáctica y traducción automática entre otras disciplinas. Curiosamente el área de la desambiguación del sentido de las palabras no es explícitamente mencionado, tal vez debido a la poderosa influencia de los sistemas supervisados.

El influjo de Lesk

(Lesk, 1986) fue un pionero en la utilización de diccionarios electrónicos para hacer desambiguación. El algoritmo conocido como algoritmo de Lesk se vale de la definición de los sentidos de las palabras para contar coocurrencias de palabras entre aquellas y las palabras del contexto de la palabra objetivo de modo que se escoge el sentido de la palabra objetivo que tenga más coocurrencias con el contexto. Lesk utilizó los inventarios de sentidos de tres diccionarios; el Websters 7th Collegiate (G. and M. Merriam Company, 1971), el Collins English Dictionary y el Oxford Advanced Learner's Dictionary (Crowie and et al., 1989).

Lesk realizó una evaluación manual de su método y calculó una precisión entre un 50 % y un 70 % sobre partes de 'Pride and Prejudice' y noticias de Associated Press, sin embargo su evaluación no podría calificarse de rigurosa (la muestra de palabras era pequeña y el propio Lesk desambiguó las palabras a mano *después* de ver las respuestas de su algoritmo).

El tamaño de la ventana contextual fue probado con 10, 8, 6 y 4 palabras, con pocas diferencias. También se pesaron las coocurrencias dividiendo por la longitud de la entrada de cada sentido en el diccionario, también con un efecto limitado. Lesk consideró también pesar la influencia de las palabras en las coocurrencias según la distancia a la palabra a desambiguar, aunque no llegó a implementarlo para el artículo. El trabajo de Lesk ha sido de una notable influencia en la literatura.

Entre los influenciados se encuentran (Cowie et al., 1992). Un problema obvio del algoritmo de Lesk, es que las palabras del contexto son también ambiguas en general por lo que para evitar el ruido inducido por ese hecho habría que considerar todos las combinaciones de sentidos a la hora de calcular las coocurrencias. La solución que propusieron Cowie, Guthrie & Guthrie consistía en escoger solamente un sentido para cada palabra del contexto. Específicamente el sentido de cada palabra del contexto que maximizara la función de conteo de las coocurrencias. Al ser un problema combinatorio (de crecimiento exponencial con el tamaño del contexto), se propone la utilización de la técnica de optimización no lineal llamada *simulated annealing* (Metropolis et al., 1953; Kirkpatrick et al., 1983) para aproximar la solución. El algoritmo es bastante simple a nivel conceptual. Para cada palabra de la frase a desambiguar se elige el primer sentido del diccionario como estado inicial. Después se va pasando a otros estados seleccionando configuraciones adyacentes al azar y haciendo el cambio si disminuye la *energía* (es una función inversa del número de coocurrencias) o bien, aunque disminuyan las coocurrencias si lo permite una función probabilística que hace disminuir esta probabilidad con el tiempo (de ahí el nombre de *enfríado lento*). La

idea es que se puede ir *montaña arriba* al principio para no caer en mínimos locales cuando lo que buscamos es un mínimo absoluto (i.e. un máximo de las coocurrencias).

Los experimentos se realizaron usando como inventario el LDOCE (Procter et al., 1978), los resultados sobre 50 frases fueron de 47 % de precisión a nivel de sentido (granularidad fina) y 72 % a nivel de homógrafo (granularidad gruesa).

Mucho después, (Montoyo et al., 2001; Montoyo and Suárez, 2001; Montoyo, 2002) afinaron su sistema de Marcas de Especificidad contando coocurrencias de palabras en el contexto con las glosas de los sentidos de WordNet y con las palabras de los propios sentidos, teniendo en cuenta en ocasiones a los hipónimos e hiperónimos, en una serie de heurísticas para podar aún más los sentidos que había dejado el método de marcas de especificidad del que hablaremos más adelante.

La teoría de la Preferencia Semántica

Los experimentos de (Wilks et al., 1990) se encuadran dentro del marco teórico de la teoría de *Preference Semantics*. Según los autores: *La preferencia semántica es una teoría del lenguaje en la cual el significado de un texto está representado por una compleja estructura semántica construida a partir de componentes semánticos más pequeños. Se crean unos enlaces entre los componentes de la estructura sobre las bases de la preferencia y la coherencia. La representación semántica calculada para un texto es aquella que tiene la estructura semántica más densa entre aquellas lecturas en competición.*

Esto les lleva a afirmar los siguientes puntos:

1. La probabilidad de una relación entre dos sentidos coocurrentes en una frase es lo suficientemente alta como para extraer información útil de estadísticas de coocurrencia.
2. En qué medida esta probabilidad es mayor que la probabilidad de coocurrencia al azar es un indicador de la fortaleza de la relación.
3. Si hay más relaciones y más fuertes en una asignación de sentidos a las palabras de la frase que en otra entonces, la primera tiene más probabilidades de ser cierta.

Además de estas afirmaciones, los autores también hacen una predicción: *Dos sentidos* [se entiende que de palabras distintas] *con el mismo significado* [por ejemplo dos

sentidos de WordNet de palabras distintas pero que pertenecen al mismo synset, i.e. significan lo mismo] *tendrán la misma distribución a lo largo de una muestra representativa de textos siempre que no influyan otros factores.* A esta posición compuesta por las afirmaciones más la predicción es a lo que los autores llaman *Teoría Semántica de la Distribución.*

Esta teoría lleva a los autores a unos experimentos de desambiguación *léxica-estadística.*

En el caso del LDOCE, el recurso con el que trabajan los autores, resulta que las palabras del vocabulario controlado que se utiliza para las definiciones, es casi seis veces más polisémico que el vocabulario no controlado (el vocabulario controlado es aquel que según los creadores del LDOCE se utiliza para definir las entradas).

Se presentan seis medidas de relación entre palabras. Para ser concretos, entre las palabras del vocabulario controlado ampliado y *bank* (que es la única palabra que se desambigua). La idea es usar esas medidas de *relatedness* para calcular cual es el sentido más relacionado. Wilks et al. son los primeros en mencionar un problema endémico en este tipo de aproximaciones, y es que hay muy poco o ningún solapamiento en general entre un contexto de desambiguación y la caracterización en términos de palabras de un sentido. Es decir, no tienen apenas palabras en común (salvo quizás palabras de parada, esto es, palabras que no ofrecen contenido semántico y suelen ser ignoradas). Para solucionar este problema se propone expandir los contextos de desambiguación y las definiciones de los sentidos de alguna manera. Esta solución creará escuela entre los muchos continuadores de esta linea de investigación.

Como preparación a su experimento de desambiguación, construyeron unos vectores característicos de los sentidos y los contextos. La forma de hacerlo es tomar las palabras relacionadas de las palabras relacionadas (para la medida de relación escogida). Esta técnica la aplicarán también más tarde (Schütze, 1992b; Schütze, 1993) así como en nuestros experimentos publicados en (Fernández-Amorós et al., 2001b) y recogidos en el capítulo 5 de esta memoria, dedicado a la información proviniente de estadísticas de coocurrencia entre palabras. Finalmente diseñaron cuatro formas de juzgar la similitud entre el vector de contexto y el de sentido. El algoritmo resulta sencillo de explicar, el sentido cuyo vector sea más *similar* al vector del contexto es el elegido.

En cuanto a los resultados logrados, se asignan los sentidos manualmente a seis grupos. De las diversas combinaciones de relación, similitud y umbral, los mejores resultados en cuanto a sentido son del 45 %, y de 97 % en cuanto a grupo (se entiende que se desambiguan todos los ejemplos y la precisión y el recall son iguales). Los autores

indican que sería deseable probar el algoritmo con más palabras aparte de *bank*.

Lo más curioso es que de las 197 apariciones de *bank* estudiadas, 143 corresponden a un mismo sentido, de modo que aquí el sentido más frecuente tendría un resultado del 72.5 % a nivel de sentido, lo que supone una mejora relativa del 61 % con respecto al algoritmo de Wilks y compañía(45 %). Este es un problema bastante habitual en DSP tanto supervisada como no supervisada. En cualquier caso tomar el sentido más frecuente no es una heurística muy informativa, ya que para calcularla hay que tener el texto ya desambiguado y en tal caso el desambiguador no resulta de mucha utilidad.

Karov & Edelman vs. la dispersión de datos

El *cuello de botella de la adquisición de conocimiento* fue caracterizado por (Karov and Edelman, 1996; Karov and Edelman, 1998) de la siguiente manera: *Entrenar un desambiguador para una palabra dada W, requiere que los ejemplos en el corpus estén particionados en sentidos, lo cual, a su vez, requiere un desambiguador completamente operativo*. Los autores argumentaron que uno de los problemas habituales con los sistemas de desambiguación basados en coocurrencias es la dispersión de datos, que hace que muchas de las coocurrencias posibles no se encuentren ni siquiera en un corpus de gran tamaño (Church and Mercer, 1993) cuando es obvio que la probabilidad de tal coocurrencia no es cero (suceso imposible). Debido a ello la estimación de las palabras poco frecuentes (que son la mayoría) empeora mucho. Este problema lo atacan Karov y Edelman de varias maneras; las estadísticas que se recopilan no son a nivel de coocurrencia directa de palabras, sino que miden características de todo tipo (patrones morfológicos, relaciones sintácticas, etc. . . además de coocurrencias de las palabras), tanto entre frases como entre palabras. También incluyen pesos de las características, en las que se valoran varios factores, entre ellos para las palabras, la distancia a la palabra a desambiguar.

Se trata de un algoritmo de desambiguación completamente no supervisado, aunque la terminología original de Karov & Edelman pueda llevar a confusión por diferir de la habitual.

El modo de proceder es el siguiente: Se parte de un corpus no anotado semánticamente en el que aparece una palabra W que será nuestra palabra objetivo de desambiguación. Vamos a desambiguar todas las apariciones de esta palabra en el corpus, pero no vamos a tratar cada contexto de la palabra (que se definirá aquí como la frase u opcionalmente la frase que contiene la aparición de la palabra y las dos adyacentes) por separado, sino que el corpus participa como un todo coherente que influye en el

resultado de cada decisión de desambiguación. Esta es otra aportación muy interesante de este enfoque.

Estamos describiendo un sistema de DSP clásico, en el sentido de que se decide en primer lugar el inventario de sentidos de la palabra a desambiguar y después se anotan las apariciones según dicho inventario. En este caso, se realizaron distinciones binarias temáticas, de grano grueso y por tanto, fáciles de desambiguar a priori.

Un concepto básico es el de palabras semilla para cada sentido. Las palabras semilla de un sentido son los nombres que aparecen en las definiciones del diccionario electrónico de ese sentido. Si una misma palabra semilla pertenece a dos sentidos distintos se elimina como palabra semilla. También se emplean listas de parada para descartar como palabras semilla las palabras muy frecuentes. En el caso que nos ocupa, se empleó un recurso combinado de los diccionarios Webster's, Oxford's y WordNet (se presume que la combinación fue manual porque sólo se han desambiguado cuatro palabras).

Los contextos de la palabra objetivo formarán lo que llamaremos conjunto de entrenamiento (insistimos, no supervisado). Ahora, para cada sentido de la palabra W (en el experimento concreto, para cada uno de los dos sentidos) crearemos un conjunto de realimentación. Este conjunto constará inicialmente de los contextos en los que aparecen las palabras semilla del sentido correspondiente. De nuevo, si un mismo contexto contiene palabras semilla de más de un sentido es descartado. De esta manera vamos a utilizar para desambiguar contextos en los que no necesariamente aparece la palabra objetivo. Un ejemplo de esto es la palabra *suit*, que tiene dos significados principales, uno como prenda y otro como demanda legal. El primer sentido tendría como palabra semilla *clothes* y el otro *court*. Los contextos en los que aparece la palabra objetivo son ambiguos por naturaleza, pero no es probable que los contextos en los que aparecen *clothes* y *court* coincidan, y pueden dar la clave para desambiguar.

Ahora estamos en condiciones de desambiguar el conjunto de entrenamiento de W (i.e. las apariciones concretas de W en el corpus inicial). Cada aparición de W será desambiguada al sentido correspondiente al contexto del conjunto de realimentación más similar al contexto de la palabra objetivo. El problema es ¿como se mide la similitud entre contextos? La respuesta es que dos contextos son similares en la medida en que contienen palabras similares. Las palabras son consideradas similares en la medida en que aparecen en contextos similares. Para romper la circularidad de esta definición se emplea un mecanismo de arranque (bootstrapping). De este modo, se argumenta que palabras como *doctor* y *health* serán similares porque comparten contextos similares (no tienen por qué coocurrir directamente). Esto supone una ventaja sobre la mayoría

de las jerarquías semánticas como WordNet, donde estas palabras están muy alejadas la una de la otra.

En este enfoque las palabras y las frases juegan papeles complementarios; un frase se representa por el conjunto de palabras que contiene y una palabra se representa por el conjunto de frases a las que pertenece.

Vamos a tener, para cada palabra objetivo, W, dos tipos de sucesiones de matrices; la matriz de similitud entre palabras, y la matriz de similitud entre contextos (una por cada sentido de la palabra).

En las matrices de similitud de palabras, las filas y las columnas representan las palabras del conjunto de entrenamiento de W (recordemos que el conjunto de entrenamiento de W era simplemente el conjunto de contextos de W en el corpus), esto es, las palabras que coocurren con la palabra objetivo algún contexto. Los valores de la matriz contienen un número entre 0 y 1 que representa el grado de similitud entre las palabras. Esta sucesión de matrices se inicializa a la matriz identidad, es decir, dos palabras distintas tienen una similitud de 0 y una palabra consigo misma tiene una similitud de 1.

Para cada sentido de la palabra objetivo W vamos a tener una sucesión de matrices de similitud entre contextos. Las filas de estas matrices son siempre los contextos de la palabra objetivo. En el primero paso de la iteración, las columnas también son los contextos de la palabra objetivo, de modo que la inicialización de estas matrices es a la matriz identidad. Más adelante las filas conservarán el mismo significado pero las columnas pasarán a ser los contextos del conjunto de realimentación de cada sentido.

Ahora en cada paso de la iteración, primero actualizaremos las matrices de similitud entre contextos (para cada sentido) a partir de la matriz de similitud entre palabras y después actualizaremos la matriz de similitud entre palabras a partir de las matrices de similitud entre contextos. Este proceso se repite hasta que los cambios son suficientemente pequeños (el artículo incluye una demostración de convergencia de las sucesiones de matrices).

Para calcular estas nuevas matrices se introduce el concepto auxiliar de afinidad. La afinidad se puede medir entre una palabra y un contexto o entre un contexto y una palabra, se trata de una medida asimétrica, la afinidad de una palabra a un contexto depende de la similitud de la palabra a las palabras del contexto. Análogamente, la afinidad entre un contexto y una palabra depende de la similitud entre el contexto y los contextos que contienen a la palabra. Estas dos relaciones de afinidad también se van actualizando a medida que cambian las medidas de similitud. Las fórmulas

concretas son las siguientes:

$$\text{aff}_n(\mathcal{W}, \mathcal{S}) = \max_{\mathcal{W}_i \in \mathcal{S}} \text{sim}_n(\mathcal{W}, \mathcal{W}_i)$$

$$\text{aff}_n(\mathcal{S}, \mathcal{W}) = \max_{\mathcal{S}_j \ni \mathcal{W}} \text{sim}_n(\mathcal{S}, \mathcal{S}_j)$$

A su vez la medida de la similitud iterativa es:

$$\text{sim}_{n+1}(\mathcal{S}_1, \mathcal{S}_2) = \sum_{\mathcal{W} \in \mathcal{S}_1} \text{weight}(\mathcal{W}, \mathcal{S}_1) \cdot \text{aff}_n(\mathcal{W}, \mathcal{S}_2)$$

$$\text{sim}_{n+1}(\mathcal{W}_1, \mathcal{W}_2) = \sum_{\mathcal{S} \ni \mathcal{W}_1} \text{weight}(\mathcal{S}, \mathcal{W}_1) \cdot \text{aff}_n(\mathcal{S}, \mathcal{W}_2)$$

Una vez alcanzada la convergencia, las apariciones de la palabra objetivo son anotadas usando los conjuntos de realimentación. Concretamente, cada contexto de la palabra objetivo se anota con el sentido del conjunto del realimentación que contenga el contexto más similar.

Se realizan experimentos sobre cuatro palabras: *drug*, *sentence*, *suit* y *player*. Se comparan los resultados con los de (Gale et al., 1993), así como con los de (Schütze, 1992b). Se llega a la conclusión de que este método es mejor porque con pocos ejemplos de entrenamiento da buenos resultados, además sirve para cualquier MRD. La precisión media es del 92 %. El método aporta un enfoque muy interesante con respecto al problema de la dispersión de datos. También se intuye un coste computacional importante si el corpus de entrada es grande, aunque precisamente uno de los puntos fuertes del algoritmo es que el corpus entero influye en la desambiguación de la palabra objetivo en todos sus contextos. Otro hecho destacable de esta aproximación es que se capturan relaciones entre palabras de orden superior a la mera coocurrencia directa.

Redes de vecindad

(Guthrie et al., 1991) utilizaron también la coocurrencia de palabras de un modo similar en algunos aspectos a (Wilks et al., 1990). Se inspiraron en buena medida en

las redes Pathfinder (McDonald et al., 1990). *Grosso modo* estas redes forman grafos con pesos entre las palabras que coocurren en una colección de textos. Cuanto mayor el peso, mayor la relación entre las palabras, esto es, la red sirve para identificar las palabras más relacionadas. El problema era que las *vecindades* de coocurrencia no distinguían entre sentidos (verbigracia, qué palabras están relacionadas con qué sentidos) y que para eso era necesario usar las definiciones de sentidos, en este caso de LDOCE, (Procter et al., 1978), y expandiendo las vecindades de alguna manera.

Siguiendo el cliché de Firth, *you shall know a word by the company it keeps*[1], los autores están interesados en distinguir las palabras de la vecindad en función de la categoría de dominio asignada por LDOCE a algunos sentidos. Para ello, como LDOCE consta de un vocabulario controlado de 2187 palabras, se hace lo siguiente. Para la palabra bank, por ejemplo, algunos sentidos están clasificados como EC (economy). Se hace una matriz de 2187x2187 (simétrica) de las apariciones de las palabras del vocabulario controlado juntas en sentidos cuya categoría sea EC. Después se calcula una similitud entre las palabras x e y usando la fórmula: $f_{xy}/(f_x + f_y - f_{xy})$, donde f_x es la frecuencia de la palabra x en el diccionario y f_{xy} la frecuencia aparición de x e y en una misma definición. Así se crean vecindades de palabras. Podemos obtener una vecindad de palabras para los sentidos económicos de *bank* y otra para los sentidos de otro tipo (hay sentidos sin categoría, para ellos se considera que existe una *categoría nula*). Las vecindades para una palabra y una categoría podrían ser, por ejemplo, las diez palabras más relacionadas según la medida.

El algoritmo de DSP consiste en lo siguiente: Para uno de los códigos de dominio de los sentidos de la palabra a desambiguar hacemos la intersección entre la *vecindad* de esa categoría para esa palabra (esa fila de la matriz, vaya) y las palabras del contexto. Esta intersección debe tener un umbral mínimo de palabras para ser considerada. Si no se supera el umbral se reeemplaza la vecindad por otra ampliada que es la fila de la matriz más la primera palabra más relacionada con cada una de las palabras de la fila. Si con esto no alcanzamos lo que queremos podemos construir otra vecindad con la fila original de la matriz más las dos palabras más relacionadas con cada palabra de la fila. Se continua el proceso hasta que se supere el umbral. Entonces, se selecciona el sentido resultante. Si hay más de un sentido se usan las palabras de las definiciones como núcleos para crear vecindades discriminadoras para ellos que se vuelven a intersecar con el texto (no se dan más detalles).

El artículo no da detalles de precisión, menciona simplemente que los experimentos *están realizándose*. Más tarde (Schütze and Pedersen, 1995) tomarán una aproxima-

[1] Dime con quien andas y te diré qué palabra eres.

ción similar que explicaremos más abajo.

Algo similar realizó (Luk, 1995). La idea es sencilla, se toma el vocabulario controlado del LDOCE, se reduce un poco haciendo stemming (cortando la palabras de la misma familia por la raíz común, por ejemplo *computer* y derivados por *comp* que no es una palabra) o conflating (sustituyendo las palabras de la misma familia por una palabra concreta de la familia, por ejemplo, *computation* por *computer*) y se crea una matriz de información mutua contando coocurrencias sobre el Brown Corpus. El contexto es la frase.

Lo novedoso está en la forma de calcular la puntuación de cada sentido. Al resultado obtenido se le resta la puntuación que sacaría el sentido contra un contexto *neutro*, es decir, un suavizado de la media. Compara sus resultados con los de (Yarowsky, 1992), aunque probablemente el LDOCE(Procter et al., 1978) y el Roget's (Chapman, 1977) no sean inventarios de sentidos comparables. Obtiene un 77 % de precisión, a un nivel del 100 % de *recall* (entendemos que se refiere implícitamente a cobertura, por semejanza con la recuperación de información). Lo peor es que mide la efectividad de un sujeto humano y le sale 71 %, menos, o sea, lo que nunca debe pasar según (Kilgarriff, 1998), puesto que la desambiguación es un problema de inteligencia artificial, donde la medida de la corrección la da el hombre y por eso no debería ocurrir que el recall de un sistema automático fuera mayor que el acuerdo esperable entre anotadores humanos.

El siguiente trabajo que vamos a comentar es supervisado por lo que respecta a la caracterización de los sentidos, pero por lo demás encaja perfectamente en la sección de vecindad por coocurrencia.

En (Niwa and Nitta, 1994) se definen unas palabras representativas, *origin words* que se escogen como de la 51 a la 1050 más frecuentes en el Collins English Dictionary. El vector de distancia de una palabra es el vector de las distancias de esa palabra a las *origin words*. Las distancias se miden como el camino más corto entre las dos palabras. Dos palabras tienen una arista entre ellas si una aparece en la definición de un sentido de la otra. El peso de cada arista se calcula mediante una fórmula sensible al número de total de enlaces entrantes y salientes de cada palabra y el número de aristas entre cada par de palabras. Después esas medidas se normalizan.

Se calcula la verosimilitud de la coocurrencia de dos palabras mediante la medida de información mutua. El logaritmo de cero se considera cero (como de costumbre). Los logaritmos negativos se ignoran (también es habitual hacerlo, aunque tales casos son raros). El contexto está formado por las cincuenta palabras anteriores y las cincuenta

posteriores a la palabra a desambiguar. Como corpus se toma el CD1 de la colección del Wall Street Journal (WSJ) del 1987, unos 20MB. De las 62000 palabras con entrada propia del Collins English dictionary sólo se encuentran el 16 % de los pares cartesianos (palabra con entrada, origin-word), es decir, unos 10 millones de entradas. Se adopta el método de (Wilks et al., 1990) para calcular el vector de contexto; se suman los vectores de ocurrencia de las palabras que aparecen en el contexto. Se calcula la similitud del contexto con cada uno de los vectores de los sentidos. La similitud se mide usando el coseno del ángulo que forman estos vectores. Estos vectores de los sentidos se han formado a partir de ejemplos anotados manualmente. Se elige el sentido del ejemplo más similar. Para los contextos se elige un radio de 10 palabras antes y después. Se toma un número indeterminado de ejemplos de entrenamiento (más de 20 en cualquier caso).

Además, se intenta que las distinciones de sentidos (binarias), sean de grano grueso, es decir, se eligen palabras con significados bien diferenciados. Los resultados están entre el 80 y el 100 % de precisión. Los vectores de coocurrencia dan mejores resultados que otro experimento realizado con un método basado en diccionarios.

En el enfoque de la tesis de Resnik, (Resnik, 1993) aunque con una base teórica distinta, la desambiguación se reduce en sus aspectos prácticos a calcular la similitud entre palabras estimando dicha similitud en base a frecuencias de aparición de palabras (juntas y separadas) en corpora no anotado.

La vecindad bayesiana de Yarowsky

El trabajo de (Yarowsky, 1992) empleó la medida de la información mutua. En su experimento se toman como inventario de sentidos las categorías del Roget's Thesaurus. Para poder desambiguar se van a buscar palabras *sobresalientes* para cada categoría. Estas palabras sobresalientes se van a buscar en un corpus no anotado. En este caso concreto la enciclopedia Grolier de 1991 (Grolier Inc., 1991), que consta de unos diez millones de palabras.

Para cada una de las 1024 categorías del Roget's, se toman las palabras miembro de la categoría (las palabras que en su entrada tengan asignada la categoría) y se buscan concordancias de esas palabras en el texto (de la enciclopedia Grolier). Cada palabra puede pertenecer a más de una categoría del Roget's (es decir, son polisémicas), de manera que se introduce cierto ruido, pero se arguye que la señal se focaliza en una sola categoría, de manera que la señal supera al ruido.

Las concordancias son ventanas de 100 palabras centradas en las apariciones de las palabras de la categoría. Ahora se buscan esas palabras más *destacadas* o sobresalientes para esa categoría. Esto se realiza calculando la información mutua entre cada palabra w y cada categoría Cat, esto es

$$\log \frac{P(w \cap Cat)}{P(w)P(Cat)}$$

que se estiman contando las frecuencias correspondientes. Pese a que Yarowsky ya nos ha advertido, no deja de ser llamativo el que se considere que cualquier aparición de una palabra de una categoría, Cat, del Roget's se considere como una aparición de la categoría, cuando lo normal es que una palabra pertenezca a varias categorías.

Mediante este cálculo, lo que hemos logrado es una serie de palabras que manifiestan una cierta preferencia numérica por cada categoría del Roget's. Estas palabras no tienen por qué pertenecer a la categoría, pueden ser adjetivos o modificadores habituales de la categoría o simplemente palabras que coocurren con mayor frecuencia de la esperable con palabras de la categoría.

Como la estimación realizada mediante el estimador de máxima verosimilitud no es muy fiable, se hace una interpolación entre las probabilidades globales de w en el corpus y su probabilidad local. Esto se explica con detalle en (Gale et al., 1993; Gale et al., 1994). Es importante resaltar que una vez estimadas estas palabras destacadas, hay que aplicar una fórmula de combinación de dicha evidencia (i.e. como influyen las palabras del contexto para seleccionar un sentido). En este caso la información del contexto se combina mediante el procedimiento de decisión bayesiano (explicado más adelante, en 3.2.5). Como hemos introducido los logaritmos en la información mutua, el aspecto final de esta decisión bayesiana se reproduce a continuación. La categoría Cat, del Roget's elegida para una palabra que ocupa el centro de un contexto es la que maximiza la siguiente expresión:

$$\sum_{w \in \text{contexto}} \log \frac{Pr(w \mid Cat) \times Pr(Cat)}{Pr(w)}$$

El contexto, como en otros trabajos de Yarowsky se extiende a 50 palabras a cada lado de la palabra a desambiguar, por motivos ya expuestos. Una crítica que puede hacerse a este trabajo es que las categorías correctas en los experimentos hayan sido asignadas por el propio autor, en algunos casos uniendo diversas categorías del

Roget's Thesaurus en una sola, sin un criterio claro que justifique esta conducta. Se desambiguan ejemplos de 12 palabras polisémicas con una media de tres sentidos cada una. No se incluyen verbos entre ellas por su mala adaptación al método (y porque son muy difíciles de desambiguar, claro). Los resultados son del 92 % de precisión.

La vecindad en Schütze : Las matrices de coocurrencia

Otro interesante método de DSP basado en coocurrencia (y aplicado además a RI) puede encontrarse en (Schütze, 1992b). La hipótesis es que el significado de una palabra en contexto depende del significado de las palabras con las que coocurre. Para aprovechar esta hipótesis se utilizan unas caracterizaciones de dichas palabras como vectores. Concretamente, las dimensiones de estos vectores son palabras. El valor de cada componente es cuantas veces coocurren en un contexto del corpus la palabra de la dimensión y la palabra que se quiere caracterizar. La coocurrencia se puede medir a nivel de frase o en una ventana de tamaño prefijado.

Este enfoque permite calcular la distancia entre dos palabras calculando el coseno del ángulo que forman sus vectores. El significado de una palabra en contexto lo podemos ver como la media normalizada (o centroide) de los vectores de las palabras que aparecen en el contexto.

Para poner en marcha esta representación, según el autor, es necesario reducir la dimensionalidad, puesto que la matriz de coocurrencia de las palabras es densa (no dispersa), porque se emplean ventanas de gran longitud y los requerimientos de espacio son considerables. Con este objeto se aplica la técnica numérica de *Singular Value Decomposition* (SVD). Tras esta reducción, las nuevas representaciones son llamadas subléxicas (las dimensiones ya no corresponden exactamente a palabras).

No se utilizan diccionarios ni tesauros. Se deciden por el autor las distinciones de sentidos, en este caso binarias para 9 palabras y ternarias para otra. Se crea un conjunto de ejemplos anotado manualmente por el autor según los sentidos decididos. Estos ejemplos anotados junto con sus contextos se convierten en vectores característicos. Se realiza *clustering* sobre estos vectores. La idea es que un sentido puede corresponder a varios clusters, es decir, que los clusters constituyen una distinción de sentidos más fina que la inicial, aunque la evaluación se realiza sobre la granularidad gruesa inicial. Se supone que esta distinción interna más fina mejora el resultado final de grano más grueso. Los *clusters* son analizados, de suerte que si un cluster está formado por ejemplos no homogéneos se descarta el cluster completo. Los clusters no homogéneos son aquellos en los que hay ejemplos de sentidos diferentes de una misma palabra,

en el caso que nos ocupa, en 9 de las 10 palabras, si los dos únicos sentidos de la palabra están presentes en el mismo cluster. Es evidente que estos clusters no sirven para discriminar entre estos sentidos. No queda claro si la revisión de los clusters es supervisada o no.

El algoritmo de desambiguación consiste en asignar a cada caso de prueba el sentido correspondiente al cluster más cercano.

Las formas flexionadas se excluyen, y también los sentidos infrecuentes de las palabras de prueba. Los sentidos excluidos constan de los términos multipalabra (think-tank), sentidos metafóricos y algunos otros. Los resultados superan en casi todos los casos el 90 % de precisión.

La información derivada de la matriz de coocurrencias se puede usar para muchas cosas, es casi como un tesauro, según el autor. Hay muchas palabras que están relacionadas semánticamente con sus vecinos cercanos en el espacio de la representación vectorial. Sin embargo, hay palabras como *keeping* que no son bien caracterizadas por sus vecinas.

Se intenta determinar la mejor manera de calcular este espacio vectorial a partir de un texto no anotado de noticias del New York Times News Service. Los factores considerados son el tamaño de la ventana y el número de dimensiones a considerar (el método de SVD ordena las dimensiones de manera que uno puede cortar donde quiera, porque las dimensiones más importantes son las primeras, i.e. se quiere determinar donde es seguro pegar el corte). Los resultados no son demasiados claros. En el caso de la ventana parece que 1000 caracteres de longitud es una buena elección (aunque sólo se prueban tres posibilidades). La cuestión de la dimensionalidad no queda clara, aunque las primeras dimensiones (más importantes en términos de aproximación numérica de la matriz original) paradójicamente no son las que más información aportan en términos de desambiguación.

Para el autor, la técnica del Singular Value Decomposition, aplicada a la DSP tiene sus ventajas y sus inconvenientes. Entre los inconvenientes podemos encontrar que no mejora la precisión del sistema y que no tiene una razón de ser tan clara como en recuperación de información, donde se conoce como *Latent Semantic Indexing* (LSI), puesto que aquí no es necesario hacer suavizado como en IR. Entre las ventajas claramente está que se necesita mucho menos espacio para almacenar y utilizar la matriz.

Nuestra opinión es que se trata de un trabajo interesante pero muy complicado, en el que la cantidad de factores y algoritmos entremezclados hace que sea casi impo-

sible saber cual es el impacto de cada uno. Por otra parte, las distinciones binarias de sentidos tan distintos no son especialmente difíciles de desambiguar como se ha comprobado ya muchas veces. Los resultados son incluso algo pobres comparados con otros experimentos sobre distinciones binarias, por ejemplo (Yarowsky, 1995b). Lo cierto es, que Yarowsky se esforzó en que los sentidos fuesen equiprobables en su experimento, mientras que en el caso de Schütze había unos sentidos dominantes bastante claros, que hacían que una heurística del más frecuente diera mejores resultados que en el caso de Yarowsky. Éste último replicó en (Yarowsky, 1995b) que los experimentos de Schütze eran más fáciles de desambiguar que los suyos. Por otra parte la mera medición de coocurrencias es una medida demasiado burda, porque no se tienen en cuenta las frecuencias relativas de aparición de las palabras, de forma que por ejemplo las palabras de función, muy frecuentes, influirían mucho en el resultado. Tal vez este fenómeno esperable se contrarreste con la reducción de dimensionalidad del SVD, pero en el campo subsimbólico es difícil hacer interpretaciones.

Este trabajo fue extendido en (Schütze, 1993), las mejoras o aclaraciones concretas fueron tres:

1. Más aplicaciones de la DSP: Un sistema de reconocimiento del lenguaje para distinguir entre expresiones homónimas o parónimas, como podrían ser en inglés: *Get Prestige wreck a nice beach*[2], o *Get prestige, recognize speech*[3]. Otra, un filtro de correo, que tendría que entender el contexto para hacer bien su trabajo.

2. Toma como base de trabajo 4-gramas, en vez de palabras. Elimina los más frecuentes y los menos frecuentes, dejando los de la zona intermedia.

3. Respecto a la asignación de los clusters a los sentidos, Schütze considera que el proceso de asignación de los sentidos a los cluster no es supervisado, aunque dice que la asignación se hace por inspección manual de los diez o veinte primeros ejemplares de cada clase.

Más adelante, (Schütze and Pedersen, 1995) midieron la coocurrencia en un corpus de textos del WSJ y expandieron los contextos de desambiguación multiplicando la matriz de coocurrencia por el vector de contexto. Lo que hacen los autores es calcular una primera matriz simétrica de coocurrencias entre palabras de una ventana de radio 20 (41 palabras en total) contiguas. Después se calcula otra matriz simétrica R,

[2]Consiga que el Prestige arruine una bonita playa

[3]Consiga prestigio, reconozca el habla

basada en hacer para cada par de palabras el coseno de sus respectivos vectores de coocurrencia.

La idea que motiva esta aproximación es que dos palabras relacionadas semánticamente tendrán vecinos similares, en la medida en que sentidos relacionados sean expresados por palabras similares.

Una consecuencia interesante de calcular la similitud semántica entre palabras de esta forma es que los sinónimos, que no suelen coocurrir en contextos cortos, aparecen bastante relacionados, debido a que esta última matriz utiliza efectivamente coocurrencia de segundo orden.

Una vez que todo esto ha sido calculado, se procede al algoritmo de desambiguación en sí mismo. Si tenemos un vector de contexto v, calculamos Rv para obtener un vector *enriquecido* con la información de similitud semántica. En realidad no es eso exactamente puesto que también se hace un pesado idf[4]. Después, como en el experimento anteriormente mencionado, se asigna el sentido del cluster al cual el vector enriquecido esté más cercano. Se evalúa sobre las mismas condiciones que en el experimento de (Schütze, 1992b), con resultados similares.

El método es interesante, aunque el experimento deja preguntas abiertas: ¿escalará bien a distinciones de sentidos más finas o a muestras de palabras mucho mayores? Ya sabemos que métodos de desambiguación para distinciones binarias de grano grueso que funcionan bien hay bastantes. Por otra parte, ¿no es excesivo el coste computacional?

Otra contribución de Schütze, (Schütze, 1998), afina más sobre el problema de la DSP en general y sobre sus métodos de conteo en particular. En la introducción, Schütze hace una interesante subdivisión del problema de la DSP en dos: uno de ellos sería la *discriminación* de palabras en contexto, que sería decidir cuando dos palabras en contexto son usadas en el mismo sentido. El otro problema sería de etiquetación, que consistiría en darle una *etiqueta* a cada sentido. La tarea de DSP tradicional consiste en ambas, sin embargo, en algunas aplicaciones, por ejemplo IR no es necesaria la fase de etiquetación para que la discriminación sea de utilidad. Ello tiene además la ventaja de no depender de ningún inventario de sentidos externo.

(Miller and Walter, 1991) resumieron sus conclusiones sobre el estudio de la similitud semántica en humanos con la *Hipótesis Contextual Fuerte: Dos palabras son similares*

[4]De *inverse document frequency*, en este caso el logaritmo del inverso de la proporción de contextos en los que aparece la palabra)

semánticamente en la medida en que sus representaciones contextuales sean similares. Schütze extiende esta hipótesis con la siguiente *Hipótesis contextual para sentidos: Dos apariciones de una palabra ambigua pertenecen al mismo sentido en la medida en que sus representaciones contextuales sean similares.*

Como en otras ocasiones (Schütze, 1992b; Schütze, 1993; Schütze and Pedersen, 1995), la representación contextual de un sentido consiste en un vector contextual cuyas dimensiones corresponden a palabras, y las coordenadas a la mayor o menor coocurrencia del sentido con dichas palabras.

Para determinar qué palabras utilizar como dimensiones, Schütze empleó dos aproximaciones al problema, una local y otra global. La local consistió en utilizar un test de dependencia probabilística, el χ^2 de Pearson para determinar cuándo dos palabras no eran independientes, para incluir como palabras características sólo aquellas que fueran fuertemente dependientes con la palabra objeto de desambiguación.

La aproximación global consistió en tomar las 20000 palabras más frecuentes en el corpus (noticias del WSJ) y medir su coocurrencia con las 2000 palabras más frecuentes. Se toman las coocurrencias en una ventana de radio 25, este dato se ha determinado empíricamente como el mejor, puesto que al aumentarlo no aumenta la eficacia del sistema.

Para calcular el vector de un contexto se suman los vectores característicos de las palabras que aparecen, es una *bag of words* (bolsa de palabras; no importa el orden entre las palabras), el peso de cada palabra se pondera con un idf (frecuencia inversa de documento).

Para crear los vectores de representación de los sentidos se utilizan algoritmos de *clustering*, como Buckshot (Cutting et al., 1992) sobre los contextos de cada palabra. Una ventaja de este método es que se pueden crear tantos clusters como se quiera, es decir, que la granularidad se puede ajustar a voluntad.

Otro de los parámetros que se evalúan es el uso de la técnica de Singular Value Decomposition (Golub and van Loan, 1986), en este caso la reducción de dimensionalidad nos deja en 100 dimensiones (recordemos que inicialmente teníamos 20000).

Como es tradición en Schütze, el algoritmo asigna a una palabra el sentido del cluster cuyo centroide esté más cercano al vector del contexto.

Los experimentos se realizaron sobre diez palabras (las ya empleadas en (Schütze, 1992b), y otras diez pseudopalabras con diez clusters generados pero sólo dos sentidos

para cada una. Las características de representación fueron locales (χ^2 de Pearson), globales, y también usar la técnica de SVD y no usarla. Las distinciones de sentidos de tamaño diez (i.e. cuando se generan diez clusters para cada palabra) se evalúan indirectamente al ser estar anotados sólo dos sentidos de cada palabra. Los datos de prueba fueron desambiguados a mano por el autor.

Los datos de coocurrencia se midieron sobre 17 meses de noticias del *New York Times*, unos 435 MB de datos conteniendo unos 60 millones y medio de palabras. Como conjunto de test se tomaron también del NYT otros 2 meses de noticias. En total 46 MB, unos 5.4 millones de palabras.

Las conclusiones de Schütze son:

- Desambiguar pseudopalabras es más fácil, puesto que las distinciones temáticas son más claras que en las palabras reales.
- El uso de SVD no sólo reduce la dimensionalidad sino que mejora los resultados.
- El uso de representaciones basadas en características globales es mejor que el de características locales. Según el autor, el test de la χ^2 sufre por la dispersión de datos. En teoría, con un enfoque binomial como el de (Dunning, 1993) el efecto debería ser menos acusado.
- El *clustering* de tamaño diez da mejor resultado (aunque luego se reduzca a dos sentidos) que el de tamaño dos.

La Web como fuente de coocurrencia

El conteo de coocurrencias aplicado a la desambiguación fue aplicado con éxito por (Mihalcea and Moldovan, 1999) vía un motor de recuperación de información, concretamente Altavista. El método de desambiguación presentado tiene dos partes, una primera de filtrado de sentidos y otra de densidad conceptual. La idea es combinar pares de palabras y medir las coocurrencias de dichos pares en documentos indexados por Altavista. Se crea para cada una de las dos palabras a desambiguar (se desambiguan de dos en dos) una consulta por cada sentido de la palabra de la siguiente forma; se busca el synset de ese sentido y se toman los sinónimos, a continuación, se hace una consulta formada por parejas de palabras unidas por OR o por NEAR, como sigue: cada pareja está formada por un sinónimo perteneciente al synset del sentido que nos interesa en cuestión y un sinónimo de cualquier sentido de la otra

palabra. Se hacen todas la combinaciones posibles de parejas para formar cada consulta. Después realizan las consulta a Altavista y la que recibe más resultados es la que determina qué sentidos son mejores. Los resultados de esta primera parte ya son bastante buenos, aunque son peores que la heurística de coger el primer sentido de WordNet. Este método basado en WordNet proporciona una enorme precisión según la evaluación de los autores. Concretamente su heurística con Altavista probada sobre el primer documento del SemCor comparada con el primer sentido puede verse en el cuadro 3.1.

Los resultados son buenos, pero lo curioso es que esta heurística se utiliza como filtro para seleccionar los sentidos más prometedores, y después entre los seleccionados se aplica otra heurística. En cualquier caso, estos experimentos tienen el problema de que, debido a la dependencia del motor de búsqueda, se pierde por completo la posibilidad de reproducibilidad del experimento. No es menos cierto que los resultados dependen del algoritmo del buscador, lo que añade un factor esencialmente desconocido y potencialmente cambiante. Resulta más razonable desde el punto de vista científico usar como heurística el tomar los pares de synsets con mayor frecuencia estadística en el SemCor y proceder a la segunda heurística. Algo parecido a esto se hará más tarde, en una de las heurísticas del sistema descrito en (Mihalcea and Moldovan, 2000a). De hecho se podría proceder directamente con la segunda heurística. Un detalle interesante de este método es que no depende de glosas u otras caracterizaciones de los sentidos, aparte de los sinónimos. Este algoritmo podría ser aplicado, por ejemplo, a cualquier idioma de los considerados en el proyecto EuroWordNet, que tienen una estructura de synsets, pero carecen de glosas propias.

Categoría	Mihalcea et al.	Primer sentido
Nombres	76 %	80.3 %
Verbos	60 %	62.5 %
Adjetivos	79.8 %	81.8 %
Adverbios	87 %	84.3 %

Cuadro 3.1: DSP vía Altavista vs Primer sentido

Más tarde desarrollaron variantes de su propio algoritmo para hacer desambiguación supervisada (entrenando con SemCor) y eliminar la dependencia de los motores de búsqueda. Las ideas principales son similares pero más elaboradas, combinando diversas heurísticas de forma iterativa para mejorar los resultados. Consiguieron una muy alta precisión (92 %) a costa de un recall medianamente bueno (50 %), sin embargo, no se indica la cantidad de palabras monosémicas que participaban en la desambiguación, cuando es bien sabido que en el texto libre este porcentaje está alrededor

del 30 %. Este algoritmo fue utilizado para realizar expansión de las consultas en recuperación de información.

En contraste con estos resultados optimistas, podemos citar a (Agirre and Martinez, 2000), que crearon consultas basadas en las glosas y sinónimos de los sentidos para obtener documentos que sirvieran como ejemplos de entrenamiento no supervisado para los sentidos correspondientes. El método elegido fue el de las listas de decisión (Yarowsky, 1995a), sin embargo, la evaluación resultó completamente decepcionante, una de las palabras obtuvo una precisión de 0, y en pocas de ellas se obtenían resultados mejores que la heurística aleatoria. La conclusión de los autores respecto a estos datos de entrenamiento obtenidos automáticamente de la Web, fue que los datos de entrenamiento eran prácticamente inútiles.

3.1.3. Sistemas basados en relaciones jerárquicas

Otros enfoques hacen uso de otras fuentes de información distintas de los meros diccionarios electrónicos. WordNet es una de las más utilizadas. WordNet es una base de datos léxica que también incluye relaciones semánticas entre sus conceptos. Destaca entre ellas la relación de hiperonimia (es-un) que suele jugar un papel fundamental en el campo de las ontologías[5]. Estos métodos se basan en la hipótesis de que un fragmento de texto trata sobre alguna *cuestión* más o menos concreta, de forma que dicha cuestión nos proporciona una preferencia por algunos sentidos de las palabras del fragmento sobre otros. Cuando esta *cuestión* es un concepto de una jerarquía taxonómica, como la de la hiperonimia de WordNet, se puede habla de sistemas basados en relaciones conceptuales. Cuando la *cuestión* se entiende como un tema perteneciente a una jerarquía, como en el caso de las etiquetas de dominio asociadas a los synsets de WordNet, se puede hablar de sistemas basados en relaciones de dominio. En ambos casos las relaciones son jerárquicas, pero no se deben confundir las relaciones conceptuales con la de dominio.

Esta misma idea la expresaron con otras palabras (Manning and Schütze, 1999): *La inferencia básica en la desambiguación basada en tesauros es que las categorías semánticas de las palabras del contexto determinan la categoría del contexto como unidad, y que esta categoría a su vez determina qué sentidos de las palabras se están usando.* Esta premisa, no necesariamente acertada, es algo así como afirmar la composicionalidad semántica del contexto respecto a la semántica de las palabras, o dicho de otro modo, que la categoría semántica del contexto es la suma de las categorías

[5]Aunque quizás un experto en ontologías lo negaría.

semánticas de las palabras que lo componen.

Relaciones de dominio

Un intento de explotar esta hipótesis podemos encontrarla en el trabajo de (Walker, 1987). Walker aplicó el siguiente algoritmo. Tenemos un tesauro en el que cada palabra puede tener varias etiquetas de dominio (subject codes). Se identifica cada sentido de una palabra con cada posible etiqueta distinta. Para desambiguar calculamos la *puntuación* para cada sentido contando cuantas palabras del contexto tienen la etiqueta de dominio del sentido como posibilidad. El sentido con más puntuación es el elegido.

Este algoritmo fue retomado en (Black, 1988) y obtuvo un éxito moderado puesto que consiguió una precisión de alrededor del 50 % sobre una muestra de cinco palabras polisémicas. Las palabras en cuestión eran consideradas difíciles y muy polisémicas.

En estos casos la jerarquía sería degenerada a un sólo nivel; a primera vista puede parecer una clasificación frívola, pero lo cierto es que los sistemas diseñados más adelante con ayuda de jerarquías más profundas generalizan perfectamente este primer enfoque. También cierto que esta aproximación guarda un cierto parecido con el experimento de (Yarowsky, 1992).

Otro sistema basado en relaciones jerárquicas de dominio es descrito en (Magnini et al., 2001). Utiliza una extensión de WordNet llamada *WordNet domains* (Magnini and Cavagliá, 2000). En esta extensión se asignan a cada concepto (synset) de WordNet una o varias etiquetas de dominio. A la hora de desambiguar a nivel de dominio, se tiene en cuenta la frecuencia relativa de aparición de cada sentido (estimado según las apariciones en SemCor) perteneciente a cada palabra del contexto. Esto tiene su importancia porque esta distribución de sentidos suele ser muy sesgada. Se combina la información para cada palabra del contexto y se extraen unos dominios posibles a los que pertenece el contexto. Con estos contextos resulta sencillo descartar los sentidos de la palabra que no pertenezcan al dominio y realizar así la desambiguación.

Los dominios de Magnini están tomados de la *Dewey Decimal Classification*[6], son de tipo jerárquico, con una distribución en tres niveles. Sin embargo, de cara a los experimentos, (el sistema fue presentado a la competición de desambiguación SENSEVAL-2) se utilizó una simplificación plana de los dominios. Al desambiguar a nivel de domi-

[6]http://www.oclc.org/dewey

nio podríamos hablar de una desambiguación de grano grueso. En cualquier caso, la precisión a nivel de sentido es muy alta, lo que parece avalar la hipótesis del dominio de la frase. Una estrategia similar, aunque más sencilla había sido empleada con anterioridad en (Wilks and Stevenson, 1996), donde se utilizaban los códigos de dominio de LDOCE (Procter et al., 1978) para desambiguar las palabras después de haber determinado su categoría gramatical, con un anotador de partes de discurso (en concreto el anotador basado en reglas de Eric Brill (Brill, 1992).

WordNet y la densidad conceptual

La jerarquía de hiperonimia de WordNet ha generado toda una corriente de investigación respecto de sus posibilidades en DSP en general, y sobre la fórmula de la *Densidad Conceptual* en particular. Revisaremos aquí algunas de ellas.

Las capuchas de Voorhees

En el trabajo de (Voorhees, 1993) se parte de la idea de que los nombres de una frase tienen un tema común. Se proponen como motivación las palabras: *base*, *bat*, *glove* y *hit*. El tema o dominio común aquí sería el béisbol. Con esta idea se propone un algoritmo de marcas en la jerarquía de la hiperonimia de WordNet (1.2 en este caso). Este algoritmo parece seminal del trabajo de (Agirre and Rigau, 1995; Agirre and Rigau, 1996; Montoyo, 2002) entre otros. El algoritmo de desambiguación puede describirse como sigue.

Se empieza definiendo lo que es una *capucha* (hood) para un sentido de una palabra: Una capucha de un synset *s*, se define como el grafo conexo más grande que contiene a s, y que contiene sólo descendientes de un ascendente de s y que no contiene ningún synset que tenga descendientes que incluyan a algún miembro de s (como conjunto de sinónimos). La idea es que la capucha de un synset es la zona donde una palabra no es ambigua (idea debida a Miller). Como los synsets pueden tener más de un padre (i.e. hiperónimo) pueden tener varias capuchas. También podrían no tener ninguna. Según la autora se pueden utilizar estas capuchas como las categorías de otros diccionarios como LDOCE o el Roget's. El indicador de clase de cada capucha es su raíz.

El algoritmo considera un procedimiento de marcado para un nombre: Se coge cada sentido de dicho nombre y se recorren los hiperónimos, llevando la cuenta de cuántas veces visitamos cada nodo. Con este procedimiento claro, el algoritmo consiste en lo

siguiente: Se coge la colección (de textos), y para cada palabra se aplica el procedimiento de marcado. La cuenta de las veces que visitamos cada nodo la llamamos visitas globales. Ahora, para un texto de la colección, tomaremos cada palabra que queremos desambiguar volvemos a marcar y llamamos al resultado visitas locales. Definimos los siguientes conceptos para cada sentido de una palabra de un texto concreto:

$$\text{Proporción de visitas locales} = \frac{\text{visitas locales al indicador de la capucha del sentido}}{\text{llamadas locales al procedimiento de marcado}}$$

$$\text{Proporción de visitas globales} = \frac{\text{visitas globales al indicador de la capucha del sentido}}{\text{llamadas al procedimiento de marcado global}}$$

Y por tanto la puntuación de cada sentido será:

$$\text{Puntuación sentido} = \text{Proporción de visitas locales} - \text{Proporción de visitas globales}$$

En otras palabras, se calcula la diferencia entre la proporción de veces que el elemento representativo de la capucha ha contado para el texto y se le resta la proporción en que ha contado para la colección. Sólo se consideran las diferencias positivas. Después se escoge como sentido correcto el sentido con mayor valor positivo de esa cantidad.

No se ha realizado una evaluación exhaustiva del algoritmo de desambiguación (se ha utilizado como paso intermedio para recuperación de información). La conclusión que saca la autora es: *La relación es-un define una jerarquía de generalización/especialización que no es suficiente para seleccionar el sentido correcto de un nombre a partir del conjunto de distinciones de grano fino de WordNet.* Estos experimentos fueron realizados con una de las primeras versiones de WordNet, tal vez sus conclusiones fuesen ligeramente distintas si se repitieran los experimentos con alguna versión actual.

La distancia mínima de Sussna

(Sussna, 1993) afronta la desambiguación semántica desde el punto de vista de la mejora de la recuperación de información. Como defensa de su aproximación cita los

clásicos problemas de la sinonimia y la polisemia. Por otra parte no realiza ningún experimento de recuperación, de modo que la recuperación es una mera justificación de la utilidad de la desambiguación. Antes de desambiguar nada se calculan unos pesos para los arcos de WordNet. Se toman en cuenta todas las relaciones entre nombres en WordNet y se calcula el *coste* de cada arco teniendo en cuenta que cuantos más arcos de un cierto tipo de relación salen de un nodo más se *dispersa* el significado de dicho nodo.

Después se hace un tratamiento de los textos por indexar. En dicho procesamiento se pasa un anotador morfosintáctico, se pasa el texto a minúsculas, se eliminan los signos de puntuación y las palabras de parada, las palabras que no son nombres y por último todo lo que no esté en WordNet como nombre.

Con respecto a la desambiguación, la hipótesis es que si tomamos un conjunto de nombres que aparecen cercanos unos de otros en el texto, cada uno de los cuales puede tener múltiples significados, minimizando la distancia entre los significados seleccionamos los sentidos correctos.

Se utilizan tres medidas alternativas para realizar la desambiguación. En la primera de ellas, restricción mutua, para un conjunto de nombres se toman todas las maneras de seleccionar un sentido de cada uno de ellos. Para cada una de estas posibilidades se forman todos los pares de sentidos posibles y se calcula la media de sus distancias. La *energía* es el valor que minimiza esta media entre todas las posibilidades. La segunda medida consiste en seleccionar primero con el método anterior los sentidos para un conjunto inicial de 2 nombres y después desambiguar el siguiente variando sólo los sentidos de este último nombre y considerando las otras selecciones *congeladas*. Este último método, obviamente, es menos costoso computacionalmente.

Como tercera estrategia, se consideran diversos métodos mixtos como la estrategia congelada, pero cuando el conjunto inicial (una ventana de texto de tamaño prefijado) se desambigua con la restricción mutua. En todos estos casos hay una ventana móvil de texto que va avanzando una palabra cada vez. En el caso de la restricción mutua es de destacar que solamente se anota la palabra central de la ventana, pero para la siguiente ventana la palabra desambiguada anteriormente puede tomar otro valor en la desambiguación de la siguiente palabra, ofreciendo así una posibilidad de uso más flexible, interesante en casos en que la distinción de sentidos es muy fina. Además, de esta manera se evita el efecto bola de nieve que padecen muchos algoritmos de desambiguación en el sentido de que una cantidad alta de fallos en una zona determinada del texto no provoca una degradación progresiva de los resultados.

Se distinguen tres casos básicos a la hora de desambiguar un nombre. Cuando hay un sentido correcto, cuando hay más de un sentido correcto y cuando no hay ninguno. Después se definen dos medidas, una de ellas sería la precisión polisémica y la otra una medida más compleja donde entra la dificultad de desambiguar cada palabra en el contexto en el que se encuentra. El *gold standard* (las soluciones que se consideran correctas) se entiende que lo calcula a mano el autor con los primeros cinco documentos de la colección *Time*. Para estimar la cantidad de información que recibe el algoritmo comparada con el texto completo se utiliza a unos anotadores humanos en condiciones similares y obtienen una precisión del 78 %.

Los resultados son consistentemente superiores a una heurística aleatoria, primero se intenta determinar cuál es el tamaño óptimo de ventana y se llega a la conclusión de está alrededor de 41 palabras. La evaluación confirma que es mejor tomar en consideración todas las relaciones de WordNet y no sólo la de hiperonimia/hiponimia. Por otra parte, el esquema de pesado de los arcos basado en el flujo de salida no parece resultar tan importante de cara a los resultados. El tener en cuenta la profundidad en la jerarquía sí tiene efectos beneficiosos en los resultados. Por último, la restricción mutua parece bastante mejor que la heurística congelada a iguales tamaños de ventana, pero, por motivos de eficiencia (la complejidad algorítmica es exponencial), no se puede llegar muy lejos con la primera posibilidad.

En conclusión, se trata de un artículo interesante, inspirador de (Agirre and Rigau, 1995; Agirre and Rigau, 1996) entre otros. Los resultados corroboran la idea de que probar todas las combinaciones posibles mejora los resultados en comparación con hacer las desambiguaciones *congeladas*, permitiendo además que el sentido elegido para la misma aparición de la palabra cambie según las circunstancias. Probablemente esta sea la conclusión empírica más importante, puesto que la comparación con otros métodos de desambiguación no es factible. Una precisión alta no era de esperar dado que es un algoritmo con poca información (sólo los nombres con entradas en WordNet) cuando es obvio que muchos nombres no tienen ninguna relación especial entre ellos. Quizás algún tipo de umbral para no comparar nombres con distancias mínimas muy grandes podría mejorar los resultados.

La aportación de Resnik

Encontramos otro algoritmo de desambiguación de nombres en (Resnik, 1998). El artículo empieza planteando que sería interesante caracterizar los sentidos como vectores de coocurrencia con palabras o aún mejor con otros sentidos, aunque se argumenta que esto hoy por hoy no es posible dada la escasa cantidad de corpora anotados

semánticamente.

Entre las aproximaciones *distribucionales* al problema cita a (Hearst and Schütze, 1993), que usaron el algoritmo de coocurrencia de (Schütze, 1993), que no era otra cosa que el algoritmo descrito en (Schütze, 1992b) y también (Resnik, 1993).

Resnik hace también la observación, que más tarde recogerán (Budanitsky and Hirst, 2000), de que la similitud es una noción más especializada que la de asociación o relación. Como ejemplo de ello argumenta que los doctores y las enfermedades podrían estar altamente asociados, pero que no por ello son particularmente similares.

Se plantea un algoritmo de desambiguación, que no está orientado a todo tipo de palabras, sino que desambigua grupos de nombres que están relacionados ya por algún motivo. Concretamente se mencionan grupos de nombres que pueden encontrarse en un tesauro (suelen ser sobre temas especializados) o bien nombres agrupados por algún algoritmo de *clustering* distribucional. Resnik dedica buena parte del artículo a clarificar las diferencias entre su algoritmo y el de (Sussna, 1993). El algoritmo en cuestión funciona de la siguiente manera: Se toman las palabras del grupo de dos en dos (todos los pares simétricos posibles). A continuación se calcula la similitud entre los sentidos de las dos palabras utilizando una medida descrita en (Resnik, 1995). Más adelante, se busca el concepto más informativo que subsuma a las dos palabras según la misma medida. Luego, se recorren los sentidos de las dos palabras y si algún sentido es descendiente del concepto subsumidor máximo, se le aumenta su puntuación. Por último, cuando esto se ha realizado para todos los pares de palabras, se normaliza la puntuación de los sentidos de cada palabra.

Resnik compara esta aproximación con (Lesk, 1986) y sobre todo con (Sussna, 1993), por considerar que es el trabajo más parecido al suyo. El artículo contiene también una evaluación formal del algoritmo, aplicado a desambiguar (a sentidos de WordNet) palabras dentro del grupo de palabras de la misma categoría del Roget's Thesaurus, esto es, exactamente el tipo de aplicación para la que el algoritmo ha sido creado. Esto permitirá al autor más tarde, en (Resnik, 1999) la posibilidad de enlazar las categorías de WordNet, el Wordsmyth English Dictionary Thesaurus[7] y el CETA (Chinese English Dictionary) (Group, 1982).

(Budanitsky and Hirst, 2000) realizaron unos experimentos que intentaban medir cuál era la mejor medida de similitud entre varias candidatas. La medida de Resnik no fue de las mejores, de manera que sería interesante saber qué pasaría si se aplicara la medida que quedó mejor situada, descrita en (Jiang and Conrath, 1997), a este

[7] http://www.wordsmyth.net

mismo algoritmo, al de (Agirre and Rigau, 1996) o incluso al algoritmo descrito en el capítulo 4.

Una objeción importante a estos algoritmos es que su representación de la información tiene una expresividad bastante limitada. Sussna comenta que los anotadores humanos tenían serias dificultades para anotar los sentidos correctos de las palabras después de que sólo quedaran los nombres lematizados (Sussna quería ver a los humanos trabajar con la misma información que el algoritmo). La aproximación de Agirre & Rigau, que veremos a continuación, también ignora la información proviniente del resto de categorías gramaticales, debido a que WordNet no dispone de relaciones nombre-verbo.

Por otra parte, desde el punto de vista lingüístico no es menos cierto que las relaciones entre los nombres de una misma frase no son siempre claras. Como ejemplo, el mero concepto de complemento circunstancial explica que un sujeto y un complemento no tengan por qué estar relacionados de una forma directa. Hay muchas estructuras lingüísticas donde esto ocurre. Si se considera por ejemplo la oración, *El niño de Burgos toca la trompeta*, no parece haber grandes relaciones entre sus nombres. Conscientes de estas limitaciones, otros investigadores siguieron caminos que tomaran en cuenta también las relaciones léxico-semánticas entre las palabras de una oración.

Probablemente, la relación de pertenencia más o menos vaga de un trozo de texto a un dominio sea más clara que la proporcionada por la relación de hiperonimia en WordNet, lo que explicaría en parte los buenos resultados de (Magnini et al., 2001).

Agirre, Rigau y sus seguidores

El concepto de *densidad conceptual* se remonta al menos a (Wilks et al., 1990). En (Agirre and Rigau, 1995) también se aprovecharon las relaciones semánticas entre conceptos (synsets) en WordNet para definir el concepto de densidad conceptual y después aplicarlo a DSP en nombres. La forma de hacerlo era en buena medida tributaria de las ideas de (Sussna, 1993) y también de (Voorhees, 1993). En este caso su algoritmo consistía en tomar una ventana de texto alrededor de cada palabra por desambiguar y calcular el concepto que *dominaba* a la ventana de texto en términos de densidad conceptual. Los sentidos de la palabra objetivo que quedaban fuera del área de influencia del concepto *dominante* eran descartados. La aportación más valiosa consistió en la medida concreta de densidad conceptual, que fue diseñada para ser sensible a un buen número de fenómenos lingüísticos que los autores consideraron relevantes. Los resultados proporcionados eran muy positivos, sin embargo sus

medidas de evaluación (orientadas a número de sentidos y no a número de palabras como es habitual) no permitían una comparación directa. El conjunto de evaluación se consideraría, con los parámetros de hoy en día, de tamaño reducido.

Los mismos autores, en (Agirre and Rigau, 1996) aportaron una comparación entre sus resultados obtenidos para cuatro artículos escogidos al azar del SemCor y los de (Sussna, 1993) y (Yarowsky, 1992). Agirre & Rigau no dudan en considerar superior su algoritmo. En (Fernández-Amorós et al., 2001a) realizamos una serie de mejoras a este algoritmo. Volveremos sobre este punto en el capítulo 4.

El algoritmo de densidad conceptual de Agirre y Rigau ha sido de una considerable influencia en la literatura científica sobre desambiguación. El algoritmo fue utilizado en la competición SENSEVAL-2 para la tarea del estonio por (Vider and Kaljurand, 2001). Por su parte (Peh and Ng, 1997) reimplementará el algoritmo y lo comparará con la heurística de tomar el primer sentido de WordNet (los sentidos de WordNet vienen ordenados por frecuencia de aparición en SemCor, de modo que la heurística de tomar el primer sentido es en este caso equivalente a la de tomar el sentido más frecuente a priori) quedando mejor situada la heurística del primer sentido. Otro algoritmo con fuertes reminiscencias de la densidad conceptual sería el algoritmo de marcas de especificidad (Montoyo et al., 2001; Montoyo and Suárez, 2001; Montoyo, 2002).

El algoritmo de marcas de especificidad resuelve algunos de los problemas del algoritmo original de Agirre & Rigau. Dicho algoritmo consideraba cada posible sentido de una palabra como un punto con *masa* para calcular la densidad. Esto plantea problemas como veremos en el capítulo 4, puesto que las palabras más polisémicas cuentan con mucho más peso en la densidad. El algoritmo de marcas de especificidad resuelve en buena medida el problema al contar sólo cuántas palabras distintas aportan sentidos en cada parte de la jerarquía. Este enfoque resulta similar al que aplicaremos en el capítulo 4. Desgraciadamente, la evaluación del algoritmo en (Montoyo et al., 2001) utiliza la misma métrica de evaluación de (Agirre and Rigau, 1995; Agirre and Rigau, 1996). En todo caso, disponemos de la evaluación de su aplicación en SENSEVAL-2. Otro problema que resuelven (Montoyo et al., 2001) es que el en el algoritmo de densidad conceptual de Agirre & Rigau era frecuente que varios sentidos de una palabra empataran en densidad conceptual. Para poder desambiguar en estos casos, Montoyo, Palomar & Rigau emplearon unas heurísticas basadas en coocurrencia con los sinónimos que conforma los *synsets* de WordNet y también jugaron con las glosas de los mismos. Combinando estas características de los synsets con los hiperónimos e hipónimos lograron mejorar los resultados del algoritmo de densidad conceptual original, simplificando en buena medida su formulación inicial.

En el ámbito del agrupamiento de textos, (Hotho et al., 2003) se inspiró en el algoritmo de densidad conceptual de Agirre & Rigau para determinar si la desambiguación podía contribuir a mejorar los resultados en dicha tarea (con resultados positivos).

La densidad conceptual, calculada de manera diferente, se utiliza también en la segunda fase de la desambiguación en (Mihalcea and Moldovan, 1999; Mihalcea and Moldovan, 2000b). A modo de ejemplo, una descripción de este algoritmo para el caso de desambiguación de un par nombre-verbo sería la siguiente. Se forman todos los pares posibles entre los sentidos del nombre y del verbo y se calcula la densidad conceptual. Los sentidos elegidos son aquellos que maximizan la densidad conceptual. La medida de densidad conceptual se basa en contar coocurrencias de nombres en las glosas de los conceptos de las subjerarquías del sentido del nombre y del verbo, con la influencia de otros factores como el número de nombres en común y la profundidad a la que se encuentran (la profundidad del synset en cuya glosa se encuentran). Los resultados ahora sí son superiores al primer sentido. También se comparan los resultados con los de (Stetina et al., 1998). Por último, se explica como generalizar el algoritmo para pares verbo-verbo y nombre-nombre.

Categoría	Filtro	Primer sentido	Resultado final
Nombres	76 %	80.3 %	86.5 %
Verbos	60 %	62.5 %	67 %
Adjetivos	79.8 %	81.8 %	79.8 %
Adverbios	87 %	84.3 %	87 %

Cuadro 3.2: Resultados finales de Mihalcea & Moldovan

La desambiguación emplea WordNet-1.6 junto con Altavista sobre el primer artículo del SemCor. El algoritmo tiene dos fases, primero filtrado de sentidos con Altavista, como vimos en la subsección anterior, y después densidad conceptual. Se desambiguan todas las palabras de clases abiertas y en el cuadro 3.2 se puede ver una comparación entre el filtro previo, el primer sentido y el resultado final.

3.1.4. Sistemas basados en Restricciones/Preferencias Selectivas

En la técnica de las restricciones selectivas, se dispone de información acerca de las relaciones entre los sentidos de una palabra y las demás palabras. Por ejemplo, si queremos desambiguar *emplear*, distinguiendo entre dos sentidos (contratar y utilizar) podríamos encontrarnos con las dos oraciones siguientes:

1. La compañía empleará nuevos candidatos (contratar).

2. El comité empleará su propuesta (utilizar).

Las restricciones selectivas (Katz and Fodor, 1964) nos informarían en este caso de que para el primer sentido de emplear (contratar), el sujeto está restringido a las categorías Humano/Organización y el objeto a Humano. En el segundo sentido (emplear), las restricciones serían Humano/Organización para el sujeto e Idea para el objeto. Como candidato tiene sentido como Humano y no como Idea en el primer caso podríamos seleccionar el sentido correcto. Análogamente, si tenemos la información de que *propuesta* es una *idea* en mayor medida que un *humano* podríamos desambiguar el verbo en la segunda oración.

(Resnik, 1997) interpretó estas preferencias selectivas como asociación estadística y desarrolló un algoritmo no supervisado de DSP, con la base teórica de las restricciones. En la práctica, sin embargo, el algoritmo resultante es similar a otros basados en coocurrencia, puesto que al ser no supervisado, las relaciones entre sentidos se *deducen* de aquellas entre palabras. El recurso elegido fue una vez más WordNet + SemCor. Los resultados demostraron que, efectivamente esta fuente de información proporciona evidencia sobre DSP, sin embargo, también parece claro que por sí sola no resulta suficiente para una desambiguación de calidad.

3.1.5. Sistemas basados en corpora multilingües

Estos sistemas se basan en aplicar la información obtenida de corpora paralelos (el mismo corpus traducido) o corpora más o menos comparables (corpora sobre temas similares). En el caso de corpora paralelos podríamos hablar de desambiguación supervisada y en el de los comparables de DSP no supervisada.

En (Dagan et al., 1991; Dagan and Itai, 1994) los idiomas de los corpora son el inglés y el hebreo. Está enfocado a la traducción automática, sin embargo la técnica tiene evidentes implicaciones sobre DSP. En traducción automática la ambigüedad léxica tiene una doble vertiente, una para elegir el sentido en el que se utiliza cada palabra en el idioma de partida y otra para saber cual de las posibles traducciones del sentido correcto es la conveniente. Si uno se centra en el segundo problema, puede verse que las estructuras sintácticas del idioma destino son muy importantes en esta tarea. Una posibilidad sería ver, de entre las diferentes traducciones, cuál es la palabra que más aparece en el corpus. Eso sería una aproximación ingenua. Lo que los autores hacen es

más sofisticado, al aplicarlo a pares de palabras, y no a pares cualesquiera, sino pares de palabras relacionadas sintácticamente, como sujeto-verbo, verbo-objeto o similar. Es decir, a aquello que Yarowsky denominó *collocations*.

También se puede aplicar este enfoque también para el primer problema. Una vez que hemos encontrado la traducción correcta de la palabra resulta que podemos descartar sentidos en el idioma origen. Así de esta manera podemos desambiguar. Esto es justamente lo que hacen los autores con un corpus en hebreo y otro en inglés. Según los autores, la cobertura es del 70 % y la precisión del 92 %. El método de inferencia es que las asociaciones (*mappings*) de las palabras a significados difieren en ambos idiomas. Se arguye que si todavía nos quedaran varios sentidos candidatos podríamos recurrir a un tercer idioma donde esos sentidos produjeran traducciones distintas.

Estas diferencias entre las asociaciones de palabras a sentidos entre idiomas será estudiada más a fondo posteriormente por (Resnik and Yarowsky, 1997) y (Resnik and Yarowsky, 1999). Se realizan unos experimentos en los que se utilizan 12 idiomas distintos para discernir sentidos. Se postula que puede obtenerse mediante la técnica descrita una granularidad de sentidos *universal*, la más fina necesaria desde el punto de vista del lenguaje, e independiente del idioma.

Por lo que respecta a la traducción automática Resnik & Yarowsky hacen la reflexión de que mediante las traducciones posibles de una palabra de un idioma en el otro se puede determinar el grado de granularidad en la distinción de sentidos necesario para una correcta traducción. Es decir, que para traducción automática se puede conocer con relativa facilidad la granularidad más adecuada porque no es necesario diferenciar sentidos de una palabra en el idioma origen que se traducen por la misma palabra en el idioma destino.

3.2. Sistemas supervisados

3.2.1. Introducción

Los sistemas supervisados son aquellos que se ayudan de la información proviniente de ejemplos anotados mediante intervención humana, frecuentemente con el objeto de entrenar un desambiguador mediante algún algoritmo estándar de aprendizaje automático, o *ad hoc*. En otras ocasiones, estos ejemplos anotados se emplean en algoritmos de razonamiento basado en casos o en memoria.

Los sistemas supervisados suelen obtener mejores resultados que los no supervisados, por motivos que ya se han expuesto anteriormente en esta tesis, y por tanto hay que tener cuidado al compararlos con los no supervisados. Sin embargo, por completitud, y dado que nada impide a un algoritmo supervisado utilizar técnicas no supervisadas en alguno de sus componentes, enumeraremos las técnicas de decisión más habituales dentro de esta extensa familia de algoritmos.

3.2.2. Árboles de decisión

Los algoritmos de aprendizaje inductivo basados en árboles de decisión se han utilizado en varias ocasiones en DSP. (Mooney, 1996) comparó siete algoritmos diferentes de DSP entre los que se contaba uno basado en árboles de decisión. (Pedersen and Bruce, 1997a) también comparó varios sistemas entre los que se incluía uno basado en el algoritmo C4.5 de (Quinlan, 1993).

3.2.3. Redes neuronales

Las redes neuronales también han sido objeto de investigación en DSP. Un problema considerable de este enfoque es que el tratamiento sub-simbólico hace muy difícil ver qué es lo que se gana y qué es lo que se pierde con estos métodos. La base teórica resulta confusa. Como ejemplo de estos enfoques y problemas puede consultarse (Veronis and Ide, 1990; Véronis and Ide, 1995; Higinbotham, 1990; Towell and Voorhees, 1998).

3.2.4. Razonamiento basado en casos o en memoria

El razonamiento basado en casos no ha sido utilizado en muchas ocasiones en DSP, sin embargo, un algoritmo de este tipo obtuvo unos resultados espectaculares en la primera competición SENSEVAL, el algoritmo TiMBL.

En este algoritmo se devuelve el sentido correcto de un ejemplo anotado manualmente que está una base de datos y resulta estar más cercano al contexto de desambiguación, según una cierta métrica, al contexto de desambiguación que los demás ejemplos anotados. Los detalles concretos de esta métrica se encuentran en (Veenstra et al., 1998).

3.2.5. Probabilísticos Bayesianos

Otra familia importante de algoritmos supervisados es la de los modelos de decisión probabilísticos bayesianos. La idea es seleccionar aquel sentido de la palabra objetivo que tenga la máxima probabilidad de ser correcto, condicionado a la aparición de la palabra por desambiguar en un cierto contexto. Esto es, para el significado s de la palabra x, empleada en un contexto c,

$$P(s|c) = \frac{P(s)P(c|s)}{P(c)}$$

usando el teorema de Bayes. Omitiendo el factor P(c), que es constante para todos los sentidos en competición, el problema se reduce a calcular $P(s)P(c|s)$ para todo s. La probabilidad de s se suele estimar a partir de su frecuencia de aparición en alguna colección anotada. La probabilidad $P(c|s)$, esto es, de que se produzca el contexto c condicionado a que la palabra x se emplea en el sentido s se calcula basándose en un vector de características $< F_1 = f_1, F_2 = f_2, \ldots, F_N = f_N >$. Estas características son habitualmente la presencia de palabras *destacadas* con la que queremos desambiguar, sus lemas, sus partes del discurso y cualquier otro tipo de información en la que tengamos alguna confianza teórica. Aquí es donde los caminos se bifurcan.

El modelo bayesiano *ingenuo* consiste en considerar las características como probabilísticamente independientes unas de otras, de tal modo que $P(c|s) = \prod_{i=1}^{i=n} P(F_i = f_i|s)$ y estas probabilidades condicionadas simples se pueden estimar sobre un corpus anotado. Esta aproximación la tomaron entre otros (Gale et al., 1993; Mooney, 1996; Leacock et al., 1993; Ng and Lee, 1996; Ng, 1997b; Pedersen and Bruce, 1997a; Pedersen and Bruce, 1997b; Escudero et al., 2000; Chodorow et al., 2000). También, como ya vimos, (Yarowsky, 1992).

Los modelos descomponibles, por el contrario, consideran que dentro del conjunto de características existen dependencias entre algunas de ellas. Se crean entonces subconjuntos de características tales que dos características son dependientes si y sólo si pertenecen al mismo subconjunto. El problema aquí es que el tamaño de datos de entrenamiento para estimar de forma fiable cada subconjunto crece enormemente. Esta aproximación la han tomado (Bruce and Wiebe, 1994; Pedersen et al., 1997; Tom O'Hara and Bruce, 2000) aunque no está claro que sus resultados superen cualitativamente al método *ingenuo*.

No hay nada en la toma bayesiana de decisiones que implique que el algoritmo tenga que ser supervisado. Simplemente ha sido habitual que las características elegidas sean estimadas según un corpus anotado, pero podrían tomarse conjuntos de carac-

terísticas cuyas probabilidades pudieran estimarse en base a un corpus no anotado. Como ejemplo de uso no supervisado podríamos citar de nuevo el trabajo de (Yarowsky, 1992). Por otra parte, al producir una ordenación relativa de los sentidos, el algoritmo podría equivocarse gravemente en las estimaciones que hace de las probabilidades de cada sentido y aún así ser un algoritmo de decisión óptimo, en el sentido de otorgar la mayor probabilidad al sentido correcto.

Los algoritmos supervisados suelen obtener mejores resultados cuantitativos en DSP que los no supervisados. De hecho, parece poco discutible que a la hora de desambiguar es preferible hacer uso de toda la información disponible. Sin embargo estos algoritmos padecen problemas estructurales que no los hacen igualmente adecuados para la desambiguación de todos los sentidos de todas las palabras, por lo que la investigación en anotación semántica no supervisada todavía tiene mucho que aportar al problema de la anotación en conjunto. En particular para reducir o eliminar la dependencia de ejemplos anotados en la medida de lo posible.

3.3. Otros sistemas

Hay otros sistemas de DSP que no encajan exactamente en una ninguna de estas categorías, o que encajan en varias pero que han supuesto una contribución importante al campo de estudio de la DSP. Dedicaremos ahora nuestra atención a algunos de ellos.

3.3.1. Mihalcea & Moldovan

El sistema presentado en (Mihalcea and Moldovan, 2000a), utiliza ocho heurísticas en cascada para desambiguar y después hace arranque (*bootstrapping*) (las heurísticas combinan información sobre palabras ya desambiguadas y no desambiguadas) sobre ellas hasta que no se puede desambiguar nada más. Dos de las heurísticas son supervisadas, lo que convierte al algoritmo en conjunto en supervisado. Los resultados, fueron de un 92 % de precisión y un 55 % de recall. Se evaluó sobre seis artículos de SemCor al azar. No queda claro si esos seis artículos fueron excluidos de los datos utilizados por las heurísticas supervisadas (ni que decir tiene que de no ser así la metodología del experimento dejaría mucho que desear).

El algoritmo consiste en ocho heurísticas (o procedimientos) entre un conjunto de

palabras no anotadas todavía que consta de todas las palabras al principio y otro de palabras desambiguadas que comienza vacío. Son las siguientes:

1. Un componente de reconocimiento de entidades reconoce nombres de personas, lugares, nombres de empresas y otras. En SemCor estas entidades están anotadas como *person*, *place*, *group* y *other*. Hay quien no desambigua ni evalúa sobre estas entradas de SemCor pero esta es una novedad interesante.

2. A continuación se anotan las palabras monosémicas.

3. Se forman pares de palabras sucesivas, (W_{i-1}, W_i) y (W_i, W_{i+1}). Se busca en SemCor si esas palabras aparecen juntas. Si aparecen juntas, en todas las apariciones W_i tiene el mismo sentido y el número de apariciones supera un cierto umbral, se anota la palabra con ese sentido. Las conjunciones y los determinantes no pueden formar parte de las parejas. Esta heurística es claramente supervisada y parece estar inspirada en (Mihalcea and Moldovan, 1999).

4. En esta heurística, se determina el contexto nominal de cada sentido de una palabra. Esto se hace tomando los nombres que aparecen en los synsets hiperónimos de un sentido. Además, para ese sentido, se toman del SemCor los nombres de un entorno de 10 palabras alrededor de una aparición de ese sentido. Ahora para cada sentido contamos las coocurrencias entre el contexto nominal y el contexto a desambiguar. Ordenamos los sentidos por orden decreciente de coocurrencias. Si el primer sentido está a suficiente distancia del segundo (hay un umbral que parametriza esto) se anota la palabra con dicho sentido. Esta heurística también es supervisada.

5. Después, buscamos palabras no anotadas que estén en el mismo synset que alguna palabra ya desambiguada. Las anotamos con el sentido correspondiente al synset (esto sería algo así como *un synset por discurso*).

6. En la siguiente heurística, buscamos dos palabras no desambiguadas que pertenezcan al mismo synset. Anotamos dichas palabras con el sentido correspondiente al synset donde coocurren.

7. Luego, se buscan una palabra desambiguada y otra no desambiguada que, o bien pertenecen al mismo synset, o bien la desambiguada está marcada con un synset hiperónimo o hipónimo directo de un sentido de la palabra no anotada. Se anota la palabra con ese sentido (no se explica lo que ocurre si hay varios sentidos en esas condiciones).

8. Finalmente, se hace lo propio con dos palabras no anotadas que tengan sentidos hiperónimos o hipónimos directos uno del otro. Tampoco se explica qué pasa si hay varios sentidos en esa situación.

Antes de aplicar estas heurísticas, se tokeniza y se aplica una versión modificada del anotador morfosintáctico Eric Brill, (Brill, 1992). En otro paso preparatorio se detectan los *complex nominals*, que no son otra cosa que los términos multipalabra de WordNet. También hay una lista de palabras que no se deben desambiguar: en este caso consta de los verbos *have*, *be* y *do*. En WordNet1.7 *have* tiene 21 sentidos, *be* 13 y *do* otros 13, son pues, difíciles de desambiguar y además extremadamente comunes. Esta lista de parada beneficia a la precisión en perjuicio de la cobertura. Aparentemente, se aplican las heurísticas iterativamente (pasando de la última otra vez a la primera) hasta que ya no se puede desambiguar nada más.

No se menciona si hacen validación cruzada (no hacerlo sería poco riguroso, el *pecado mortal* de los métodos supervisados según (Manning and Schütze, 1999)). Los resultados, un 92 % de precisión y un 55 % de cobertura (para nombres y verbos), son poco tajantes: el porcentaje de palabras monosémicas y de nombres propios, que también se desambiguan con un componente de reconocimiento de entidades, en SemCor es enorme.

El recall sería del 50.6 %, que es bastante modesto, sobre todo si tenemos en cuenta que se trata de un método supervisado y además que la heurística del primer sentido obtiene un recall del 75 %. Una crítica que podría hacérsele al artículo es que no evalúa los resultados de cada heurística concreta. Si lo hicieran podríamos saber cuales son las más productivas.

3.3.2. El algoritmo de arranque de contexto amplio de Yarowsky

En (Yarowsky, 1995b) se explotan las propiedades de *Un sentido por colocación* y *un sentido por discurso*, para, a partir de unas semillas, hacer arranque (bootstrapping) no supervisado que entrena unas listas de decisión al estilo de (Yarowsky, 1995a). Los resultados serían espectaculares si no se tratara de distinciones de sentidos binarias, relativamente sencillas de desambiguar.

El algoritmo se basa en tres hipótesis:

1. Un sentido por colocación (Yarowsky, 1993)
2. Un sentido por discurso (Gale et al., 1992b)
3. Que el lenguaje es muy redundante.

El algoritmo parte de unas semillas para, en base a unas características, generar reglas de desambiguación basadas en listas de decisión con una *verosimilitud logarítmica* (log-likelihood) asociada a cada una. Esta verosimilitud logarítmica se estima sobre unas semillas, como en el artículo anterior de Yarowsky. Si las semillas se obtienen de forma supervisada, el algoritmo será supervisado, en otro caso será no supervisado. El algoritmo hace arranque para incorporar nuevas semillas, en base a las restricciones de un sentido por colocación y un sentido por discurso, y para descartar otras con esas mismas restricciones, pero no de forma determinista, sino sólo en base a unos umbrales. Esas nuevas semillas se obtienen de un corpus no anotado de gran tamaño (460 millones de palabras), mediante el arranque aplicando esas restricciones. También se utilizan un par de métodos para evitar el sobreaprendizaje (overfitting); ampliar la ventana de contexto para deducir nuevas reglas y modificar aleatoriamente el umbral de verosimilitud logarítmica de aceptación de nuevas reglas, de forma similar a la que se usa en *simulated annealing* (Metropolis et al., 1953; Kirkpatrick et al., 1983).

Las semillas se pueden adquirir de forma supervisada, i.e. seleccionar manualmente colocaciones adecuadas para cada sentido de cada palabra a desambiguar, o bien usando claves como palabras *sobresalientes* en las definiciones del diccionario.

La desambiguación se prueba sobre 12 palabras más o menos representativas o interesantes. De cada palabra se evalúan en promedio 3936 apariciones. Las distinciones de sentidos son binarias. El aprendizaje no supervisado se realiza sobre un corpus de 460 millones de palabras de artículos de noticias, *abstracts* científicos, transcripciones de textos orales, y novelas. Según el autor, constituye el corpus de mayor tamaño usado en la literatura sobre desambiguación. Los mejores resultados son los no supervisados que dan una precisión y recall de 96.5 % y son mejores que los supervisados.

La evaluación es muy detallada pero lo más interesante es que los resultados se comparan con los de (Schütze, 1992b). De las 12 palabras evaluadas, algunas son comunes con los experimentos de Schütze y otras vienen motivadas por el interés de Yarowsky en las palabras que se traducen de forma distinta al francés según su significado en contexto. El algoritmo de Yarowsky supera al de Schütze, y eso que según Yarowsky, las palabras de Schütze eran más sencillas de desambiguar, dado que la heurística del sentido más frecuente daba mejores resultados en el caso de Schütze (Yarowsky

utiliza sentidos equiprobables en el corpus de prueba). Dos ventajas que esgrime Yarowsky sobre Schütze es que su algoritmo es más sensible a matices amplios del lenguaje, puesto que no utiliza la representación de bolsa de palabras (bag of words); por ejemplo distancia colocacional, secuencias de palabras y la existencia de relaciones predicado-argumento entre palabras. Por otro lado, la adecuación a los sentidos de un diccionario se realiza de forma natural al principio del proceso en el caso de Yarowksy, mientras que Schütze hace esta adaptación después de realizar el clustering y asigna manualmente los clusters a sentidos del diccionario.

En cualquier caso, ambos métodos realizan desambiguación sobre unas pocas palabras y las distinciones de Yarowsky son binarias, o sea, de grano muy grueso, por lo que unos buenos resultados eran de esperar. También Yarowsky compara su método con una heurística de *major sense*, o sea, de sentido más frecuente. Dado lo grueso de las distinciones de sentidos, esta heurística sólo logra un 63.9 %, pero mientras esta heurística escala bastante bien al grano fino, es muy posible que el algoritmo de Yarowsky no lo hiciera tan bien, entre otros motivos por la dificultad de encontrar semillas adecuadas para la elevada tasa de sentidos por palabra que presenta WordNet. Este algoritmo será utilizado en (Stevenson and Wilks, 1999), como veremos más adelante.

3.3.3. La influencia de la teoría de la información

Otro método difícil de clasificar es el algoritmo flip-flop de (Brown et al., 1991). El sistema usa los Hansards canadienses y el algoritmo flip-flop para determinar, para cada palabra, el informante que maximiza la información mutua entre una palabra y sus traducciones. Los informantes son palabras del contexto de la palabra que se quiere desambiguar. Para simplificar el algoritmo de flip-flop, los informantes se restringen sólo a una palabra. Las formas de escoger el informante son; la palabra inmediata a la izquierda, la inmediata a la derecha, el primer nombre a la izquierda, el tiempo de la palabra si es un verbo, el tiempo del primer verbo a la izquierda y combinaciones por el estilo.

Se asignan dos sentidos (haciendo sólo distinciones útiles para la traducción) para las 500 palabras más frecuentes en inglés y las 200 más frecuentes en francés. Para las palabras francesas se escogen siete informantes (plantillas) y para las inglesas dos. Para cada palabra francesa se busca el informante (la plantilla) que maximiza la información mutua media con su traducción en los Hansards. Como ejemplo, para *prendre*, el informante con que maximiza la información mutua es el nombre a la

derecha con una información mutua de 0.381 bits.

Para desambiguar una palabra concreta se mira el valor de su informante y se calcula a qué sentido corresponde habitualmente dicho valor del informante. Por último, dado el sentido buscamos cual de las traducciones en inglés es la más habitual.

Se tomaron 100 frases al azar de los Hansards y se tradujeron correctamente 45, cuando el mismo algoritmo sin esta característica de desambiguación, traducía correctamente 37.

3.3.4. Sistemas basados en sintaxis

Xerox Europa presentó a la primera competición SENSEVAL un sistema de desambiguación basado en reglas. En primer lugar se extraen con un analizador sintáctico superficial (shallow parser) relaciones sintácticas (sujeto-objeto, modificador-nombre, verbo-objeto) y después se realiza una desambiguación basada en preferencia semántica.

Se entrena sobre la parte del Brown Corpus anotada con 45 etiquetas semánticas (se entiende que el corpus es el SemCor y las etiquetas los ficheros de lexicógrafo de WordNet) y se extraen unas reglas. Se obtiene una precisión de 97 % con un recall del 25 %. Los resultados siguen la tónica habitual, buenos resultados para desambiguación de grano grueso; en este caso, además, con baja cobertura.

(Lin, 1997) realizó un algoritmo de desambiguación no supervisado basado en relaciones sintácticas y estadísticas. La metodología es curiosa, pero los resultados aun lo son aún más. Hay una forma de evaluación parametrizada en la que, en términos clásicos, los resultados son peores que la heurística del primer sentido, pero variando los valores del parámetro pueden ser mejores.

La idea de este método se basa en el hecho, no ya hipótesis como en la época de (Wilks et al., 1990): *Dos apariciones de la misma palabra tienen significados idénticos si tienen contextos locales similares.* Según Lin, la mayor parte de los sistemas de DSP que se basan en esto intentan desambiguar una palabra aprovechando usos anteriores de la misma palabra. Dicho enfoque tiene tres problemas:

1. Una palabra debe aparecer miles de veces en el corpus antes de que un desambiguador aprenda a desambiguarla. Pone como ejemplo los trabajos de Yarowsky y Ng. Si no hay esos miles de ejemplos, la cosa no funciona.

2. Lo que se ha aprendido para una palabra no se aprovecha para las otras.
3. Por último, estos sistemas no pueden desambiguar palabras para las cuales no se ha entrenado un clasificador.

En el caso de Lin, su sistema no posee ninguna de estas limitaciones. El ejemplo que pone es que para desambiguar facility en la frase: *The new facility will employ 500 of the existing 600 employees*, usamos los sujetos de employ vistos con anterioridad y vemos que son más similares a *installation* que a *toilet* etc...

Por lo que respecta al contexto, Lin extrae información del corpus basándose en el contexto local, lo que en su caso quiere decir las palabras de las que depende sintácticamente en la frase. Así el contexto local de una palabra W está formado por ternas de la forma (*tipo palabra posición*) donde *tipo* es el tipo de relación sintáctica (sujeto, verbo, adjunto, complemento...), *palabra* es la palabra de la que depende W y *posición* puede ser núcleo o modificador. Con esto se construye una base de datos de contextos locales cuyas entradas son un contexto local, lc, y un conjunto asociado LC(lc). El conjunto asociado es de ternas (palabra, frecuencia, verosimilitud). La palabra es una palabra que en algún momento ha tenido como contexto local a lc. Frecuencia es con cuanta frecuencia (absoluta) apareció la palabra como gancho de la relación con lc y la verosimilitud es tomar la *verosimilitud logarítmica* como si la palabra y lc fueran bigramas y aplicando la fórmula de (Dunning, 1993). Esto supondría, para el ejemplo anterior, contar cuantas veces *corporation* es el sujeto de *employ* en el corpus y su ratio de verosimilitud. Cuando tomamos una palabra W, y un contexto concreto, C, tenemos unas palabras que son los selectores, esto es, las palabras que se han relacionado con W en las mismas condiciones sintácticas. Para estos selectores vamos a aplicar una medida de similitud para determinar qué sentido de W escogemos.

La explicación sobre la elección de la medida de similitud es la siguiente. Se determina, para cada synset de WordNet, la probabilidad de que un nombre escogido al azar, tenga un sentido bajo la subjerarquía del synset. Se menciona que el cálculo se realiza estimando en base a la información de SemCor. Como ejemplo, la probabilidad de entity, es de un escaso 39.5 %. La medida de similitud entre dos synsets de WordNet es:

$$\mathrm{sim}(C_1, C_2) = 2\frac{\log P(\text{primer concepto común})}{\log P(C_1) + \log P(C_2)}$$

Donde primer concepto común es el primer hiperónimo común (se entiende que el

dividendo vale cero en otro caso). Según el ejemplo, *hill* y *coast* (como synsets) hijos ambos de *geological_formation*, tienen una similitud de 59 %. Este concepto vuelve a tener reminiscencias de la idea de densidad conceptual.

Se desambiguan los nombres de siete artículos del SemCor, del mismo género que los datos del corpus de 25 millones de palabras del Wall Street Journal. Se aplica un analizador sintáctico de gran cobertura al corpus y se crea la base de datos de contextos locales. Después se aplica el algoritmo descrito.

La forma de evaluar es muy curiosa. Desambiguar correctamente un sentido significa que sea lo suficientemente similar al sentido correcto. Si esa medida de similitud es estricta, es cuando la similitud es igual a 1. Si se considera que la similitud sea mayor que cero, entonces significa que los dos sentidos tienen un ancestro común. La similitud media entre dos sentidos de una palabra se estima con el Roget's Thesaurus en 0.27. Los resultados entre este sistema y la heurística del primer sentido pueden verse en el cuadro 3.3.

Similitud	Lin	Primer sentido
> 0	73.6 %	67.2 %
≥ 0.27	68.5 %	64.2 %
$=1$	56.1 %	58.9 %

Cuadro 3.3: El papel de la medida de similitud en la evaluación

Estos resultados constituyen un buen argumento a favor de que la heurística del primer sentido podría no ser tan interesante de cara a aplicaciones finales. De todas formas tampoco con está métrica resulta fácil de batir.

El trabajo de (Dorr and Jones, 1996) trata de la correspondencia entre sintaxis y semántica para los verbos, en particular entre la sintaxis y las clases de Levin (Levin, 1993). Se termina presentando un algoritmo para asignar apariciones de verbos no clasificados por Levin a las clases existentes o bien a alguna nueva. Este algoritmo realiza una desambiguación para verbos vía WordNet y LDOCE hasta llegar a las clases de Levin.

Se describen en primer lugar unos experimentos para comprobar la importancia de la DSP en la clasificación semántica (con las clases de Levin como inventario) de verbos basadas en las signaturas sintácticas de dichos verbos en la frases que Levin utilizó como ejemplo para su clasificación. Se mide si las clases formadas por los verbos que comparten una misma signatura sintáctica coinciden con las clases de Levin. En el caso en el que se ignora la desambiguación, la correspondencia es del 6.3 %. En el caso

en el que se toma en cuenta la desambiguación (utilizando las signaturas sintácticas de los sinónimos en la clase de Levin) se logra una correspondencia del 97.9 %. Estos índices de correspondencia se calculan en base al coeficiente de Dice aplicado a la clase de Levin con la clase de la signatura sintáctica. Esto, según los autores, valida la tesis principal de Levin, es decir, que la semántica de los verbos puede deducirse a partir de la sintaxis.

Para terminar se describe un algoritmo para asignar a una aparición de un verbo a una clase de Levin o bien a una nueva clase haciendo un análisis sintáctico superficial de la frase, esto es, asignando una signatura sintáctica y probando con los sinónimos de WordNet y las clases semánticas del LDOCE. Este segundo algoritmo consigue una eficacia del 82 %. Este trabajo resulta tanto más interesante por cuanto que es notorio que los verbos constituyen la categoría gramatical más difícil de desambiguar.

3.3.5. Sistemas basados en combinación de fuentes de información

En (Hearst, 1991) se combinan varios tipos de evidencia de cara a la desambiguación. Se buscan ciertas propiedades en el nombre a desambiguar y otras distintas en las palabras del contexto distintas de la palabra objetivo. Estas propiedades son las siguientes:

Para la palabra objetivo :

- El objetivo está en mayúsculas.
- El modificador del objetivo está en mayúsculas.
- El objetivo está modificado.
- El objetivo modifica otro ítem.
- El objetivo pertenece a un sintagma preposicional encabezado por *in*, *on* u *of*.
- El objetivo pertenece a un sintagma preposicional encabezado por otra preposición.
- Un sintagma preposicional adyacente al objetivo está encabezado por *in*, *on* u *of*.

- Hay un sintagma preposicional adyacente al objetivo encabezado por otra preposición.

Para todas las palabras:

- La palabra modifica al objetivo.
- La palabra es modificada por el objetivo.
- La palabra es el núcleo de una construcción adyacente al objetivo .
- La palabra es modificador en una construcción adyacente al objetivo.
- La palabra actúa como verbo en una construcción adyacente.

Más que cuáles son las características concretas que se buscan en cada frase, lo importante es percatarse de que son tanto de tipo sintáctico como de tipo de coocurrencia, es decir, como se explicará en el capítulo 7, ambas fuentes de información se combinan. Esta combinación se realiza de una forma diferente a la de otros autores, por ejemplo (McRoy, 1992).

En el artículo se mencionan ejemplos que apoyan la tesis de que: *Los algoritmos que confían solamente en las palabras con contenido vecinas* [i.e. coocurrencia] *probablemente cometerán errores que podrían ser evitados si se tuviera en cuenta información simple de tipo sintáctico*. Como existen significativamente menos marcos sintácticos locales que posibles palabras con contenido en la vecindad (i.e. el algoritmo se encontrará con más frecuencia con palabras desconocidas que con marcos sintácticos desconocidos) los indicadores sintácticos proveen una información más generalmente aplicable, combinada con información de coocurrencia léxica, que la coocurrencia léxica por si sola.

El algoritmo necesita unos cuantos ejemplos anotados manualmente como entrenamiento. Con estos ejemplos se rellenan dos tablas, una para las características indicadas para las palabras objetivo y otra para las características de todas las palabras.

Como nota al respecto de la dependencia del dominio de entrenamiento, Hearst dice que una ventaja de hacer el arranque desde una enciclopedia es que se cubre un amplio rango de temas, así que es probable encontrar un nombre en una buena variedad de sus posibles contextos. Esto es compatible con las observaciones de (Marchment, 2002) sobre el trabajo de (Yarowsky, 1992).

El enfoque y la ideas resultan innovadoras e interesante pero los resultados no resultan de especial interés, puesto que sólo se evalúa sobre cinco nombres con distinciones binarias de sentidos y de grano grueso (son homógrafos, como bien dice el título del artículo).

(McRoy, 1992) es un artículo con un título autoexplicativo: *Using Multiple Knowledge Sources for Word Sense Discrimination*[8]. Tiene dos inconvenientes importantes, uno es que el recurso está construido a mano (con los problemas que ello acarrea en términos de cobertura y transportabilidad a otros idiomas) y el otro que no se ha realizado una evaluación cuantitativa del método. Una de las cosas que destaca la autora es que la forma de combinar las distintas evidencias no responde a un orden prefijado entre ellas, sino a la especificidad de aplicación de cada una.

Las fuentes de información que se combinan son: Etiquetas sintácticas, frecuencias de sentidos, asociaciones de palabras, colocaciones, contexto semántico (clusters), restricciones selectivas e indicios sintácticos. Concretamente se citan las siguientes características:

- El análisis de cada palabra en su raíz y afijos, o sea, la morfología.
- Las etiquetas sintácticas, i.e., la categoría morfosintáctica de cada palabra.
- Para cada sentido de una palabra, si es preferido o está obsoleto (en general basándose en su frecuencia o en particular dependiendo del dominio del contexto concreto).
- Si la palabra forma parte de una colocación (*soda-cracker*) o una relación predicativa (*take action*).
- El contexto semántico. Si las palabras del contexto comparten una categoría semántica, una situación o un tema.
- Si la entrada satisface las expectativas creadas por evidencias sintácticas (algunos sentidos sentidos sólo toman argumentos de un determinado tipo sintáctico).
- Si se satisfacen ciertas expectativas asociadas a roles (expectativas sobre las relaciones semánticas entre objetos unidos sintácticamente).
- Si se referencia algo anteriormente mencionado en el foco del discurso.

[8]Uso de múltiples fuentes de conocimiento para discriminación de los sentidos de las palabras.

Según la autora, las fuentes de información más importantes de cara a la DSP, son: Categoría gramatical, morfología, colocaciones y asociaciones de palabras. Se menciona que conocer la parte del discurso a menudo es suficiente para identificar el sentido correcto. Esto hace sospechar de un inventario de sentidos de granularidad gruesa. Estas sospechas se confirman más adelante cuando dice que: *El léxico, por diseño, sólo incluye distinciones gruesas entre sentidos.*

Otro detalle interesante es que las entradas están en orden de frecuencia de aparición y no en orden histórico como, por ejemplo, en el LDOCE.

Se realiza también una interesante distinción entre léxico estático, cuyos sentidos siempre se consideran posibles, y léxico dinámico, sentidos que necesitan palabras gatillo (triggers-words) para activarse. El léxico incluye también una jerarquía de conceptos y unos clusters categóricos (basados en la hiperonimia, en la meronimia y situacionales).

El recurso léxico, creado a mano a partir de otro comercial, cuenta con una cobertura de unos 10000 sentidos raíz y unas 10000 derivaciones.

Los resultados de cobertura son los siguientes: Se evalúa sobre una muestra de 25000 palabras del Wall Street Journal con una cobertura del sistema del 91 % en general y 98 % quitando nombres propios. No hay resultados de precisión. Los motivos aducidos son que un experto humano tenía muchas dificultades en identificar los sentidos en contexto y que la tarea era más tediosa de lo esperado. La autora expresa su preferencia por intentar hacer una evaluación indirecta a través de una aplicación como recuperación de información.

En (Mahesh et al., 1997) se describe un sistema de DSP basado en un léxico con restricciones selectivas recogidas a mano más una ontología donde aplicarlas. También se utiliza un sistema de relajación de las restricciones para los casos (frecuentes) en que las restricciones selectivas no se cumplen estrictamente.

El artículo es parco en detalles, el recurso léxico utilizado parece ser propio. Desambiguan en español. Lo más novedoso es que utilizan una mezcla entre el algoritmo de Dijkstra y el A* para encontrar el camino de coste mínimo para unir dos conceptos (se supone que las relaciones están pesadas) cuando no hay una restricción selectiva que se cumpla directamente. Utilizan el analizador sintáctico de español del proyecto Pangloss. Los pesos de los arcos (de las relaciones) se determinan experimentalmente haciendo *simulated annealing* con un conjunto de entrenamiento, así que el método tiene algo de supervisado, pero no demasiado. Los resultados son espectaculares, pero no sabemos qué granularidad de sentidos se considera. Precisión del 97 % para 4

documentos de una colección de 400 documentos de la agencia EFE.

En una saga de artículos (Wilks and Stevenson, 1996; Wilks and Stevenson, 1997b; Wilks and Stevenson, 1997a; Wilks and Stevenson, 1998a; Stevenson et al., 1998; Stevenson and Wilks, 1999) se va desarrollando un desambiguador. En un primer intento, se limita a la elección del homógrafo correcto en LDOCE de las palabras de cinco artículos de la colección del Wall Street Journal. Requiere una anotación morfosintáctica previa que se realiza con el Brill Tagger. Se consigue un 92 % de precisión a nivel de homógrafo.

Después se combina esta información con: la heurística del primer sentido, una basada en los códigos de dominio del LDOCE (Procter et al., 1978) y otra basada en *Simulated Annealing* (Metropolis et al., 1953; Kirkpatrick et al., 1983). Se evalúan los algoritmos contra 14 frases desambiguadas a mano del WSJ, en total 314 palabras. Los resultados pueden verse en el cuadro 3.4. Es digno de mención que la mejora que produce la combinación de fuentes de información es más bien modesta, como ocurrirá también en nuestro caso en el capítulo 7.

Heurística	Homógrafo	Sentido
Primer sentido	80 %	54 %
Simulated Annealing	86 %	57 %
Códigos de dominio	86 %	58 %
Combinación supervisada	88 %	60 %

Cuadro 3.4: Resultados a nivel de homógrafo y a nivel de sentido

El trabajo realizado se completa en (Stevenson and Wilks, 1999). El sistema está formado por un reconocedor de entidades más el anotador de Brill, (Brill, 1992). Después, como heurísticas a combinar, simulated annealing (47 % de precisión en solitario), y un método de restricciones selectivas (44 % de precisión). También se reimplementa y se añade como heurística el algoritmo de contexto amplio de (Yarowsky, 1995b), que por cierto consigue un 79 % de precisión al nivel de sentidos, y un algoritmo de aprendizaje automático basado en memoria (Veenstra et al., 1998). Por cierto, este algoritmo basado en memoria tiene información sobre ciertas características contextuales y también información sobre frecuencias. Si el algoritmo basado en memoria devuelve más de un sentido se devuelve el primer sentido para romper el empate. Se entrena sobre el SemCor, con una asociación (mapping) entre los sentidos de WordNet y el LDOCE realizado para SENSUS (Knight and Luk, 1994) . Se realiza el entrenamiento haciendo validación cruzada 10 veces. Los resultados son del 90 % de precisión a nivel de sentido (94 % a nivel de homógrafo). Los autores extraen la conclusión de que el hecho de que el desambiguador final sea mejor que cada una de las partes

demuestra que los diferentes *anotadores parciales* o heurísticas aplicadas tienen un contenido de información diferente.

En realidad no es sorprendente que el algoritmo final supervisado obtenga mejores resultados que las distintas heurísticas (algunas supervisadas y otras no).

En (Montoyo and Suárez, 2001; Suárez and Montoyo, 2001) se toma como punto de partida que para desambiguar palabras de categorías gramaticales distintas puede ser interesantes aplicar técnicas diferentes. Se describe el sistema presentado a SENSEVAL-2 por la Universidad de Alicante. Para los nombres se ha utilizado el método de marcas de especificidad tal como se explica en (Montoyo et al., 2001; Montoyo, 2002) y para las demás categorías gramaticales un método supervisado basado en máxima entropía. Los resultados ocupan un lugar discreto en tabla oficial de resultados.

3.4. Conclusión

A modo de resumen se presentan algunas carencias frecuentes de los sistemas estudiados que se intentarán resolver en este trabajo:

- La integración de diversos recursos y técnicas se ha lleva a cabo de tal forma que no se conoce la aportación individual de cada uno de ellos, lo que sería muy deseable para poder distinguir lo que es de utilidad en la tarea de lo que no lo es.
- Se dejan de lado las posibles contribuciones de información extraída de forma no supervisada a la DSP en favor de sistemas puramente supervisados. Esta tendencia parece ir en claro aumento.
- Se realizan sistemas *de juguete* que no pueden desambiguar más que unas pocas palabras (o sólo una en casos extremos), o con una complejidad algorítmica que no permite su uso más que en cantidades testimoniales de texto. Este inconveniente va cayendo en desuso. Sin embargo, incluso en la competición SENSEVAL-2 varios sistemas no pudieron desambiguar más que una pequeña porción de las palabras, y ello sin que la colección sea muy extensa, como por ejemplo los presentados en (Haynes, 2001; Tugwell and Kilgarriff, 2001).
- Se evalúan los sistemas con métricas *ad hoc* que hacen imposible su comparación con otros sistemas.

En términos generales podemos decir que a la investigación en DSP le queda todavía un largo camino por recorrer. En primer lugar podría citarse que no hay apenas pruebas de aplicaciones finales en las que la DSP haya jugado un papel decisivo. En el ámbito de la evaluación, se carece en general de metodologías fiables de anotación manual y estimación de error (hecho absolutamente decisivo como comentaremos en el último capítulo) o bien dichas metodologías no son públicas, situación igualmente indeseable.

En el caso de los algoritmos supervisados hay quien argumenta, por un lado, como (Daelemans and Hoste, 2002) que a menudo hay más diferencias en los resultados según el ajuste de parámetros de un mismo algoritmo supervisado que entre dos algoritmos distintos. Este fenómeno puede advertirse con toda claridad en los resultados oficiales del SENSEVAL-2 [9] en el caso de los sistemas Duluth (Pedersen, 2001). Otros investigadores, como (Banko and Brill, 2001), son de la opinión de que la elección de un algoritmo supervisado u otro es mucho menos importante que contar con un tamaño de datos de entrenamiento suficientemente grande.

Otro punto delicado de la DSP es que los estudios realizados en relación al acuerdo entre anotadores no pueden ser más dispares; en (Jorgensen, 1990) se relata un experimento en el cual la correlación entre los anotadores es de un 68 %, cifra que se antoja penosamente baja. (Gale et al., 1992a) propusieron como cota superior del rendimiento de un sistema de DSP el acuerdo entre anotadores humanos, y como cota inferior (deseable, se entiende, que no siempre posible) el sentido más frecuente, de manera que se llega a la contradictoria situación de que la cota inferior puede ser en determinadas condiciones superior a la cota superior. Como explica (Kilgarriff, 1992): *Sería inútil esperar que los ordenadores coincidieran más con el corpus de referencia que los anotadores humanos entre ellos.* La conclusión más clara que puede extraerse es que hay mucho que investigar todavía sobre la propia naturaleza del problema.

Nuestra contribución consistirá en aportar estrategias razonadas y convenientemente parametrizadas de desambiguación no supervisada, principalmente de tipo jerárquico conceptual y de coocurrencias.

La información de tipo jerárquico es muy poco sensible al cambio de colección de prueba, factor que resulta devastador como veremos para otras estrategias (en particular la del primer sentido), de hecho probaremos en el capítulo 4 que, al menos para los nombres, la información jerárquica es una heurística no demasiado inferior a la heurística de tomar el primer sentido del diccionario por lo que a la precisión se

[9] Los resultados están disponibles en http://www.senseval.org.

refiere.

La información extraída calculando medidas de similitud entre palabras mediante métodos no supervisados que investigaremos en el capítulo 5, resulta mucho más dependiente del sesgo en la distribución de sentidos para las palabras que la información conceptual, pero posee a cambio la ventaja de que la información de entrenamiento (texto sin ningún tipo de anotación) se puede conseguir casi sin limitaciones.

Finalmente, defenderemos la necesidad fundamental de tener más colecciones de prueba para evaluar los algoritmos de desambiguación, dadas las notables diferencias que puede llegar a haber en algunos casos.

Parte II

Experimentos con distintas fuentes de información

Capítulo 4

Información jerárquica

4.1. Introducción

Un nivel competitivo en DSP, como ilustraron los participantes en el primer SENSEVAL (Kilgarriff and Rosenzweig, 2000) sólo puede ser alcanzado mezclando fuentes de conocimiento de todo tipo: Información de coocurrencia, información sintáctica, colocaciones, información adicional de los diccionarios tal como etiquetas de dominio, restricciones selectivas y toda clase de heurísticas, véase por ejemplo (Ng and Lee, 1996; Rigau et al., 1997; Wilks and Stevenson, 1998b; Stevenson and Wilks, 1999). Un problema con tales sistemas híbridos es que resulta difícil discernir cuál es el poder discriminador de cada uno de los diferentes tipos de conocimiento sobre el contexto de la palabra a desambiguar. Nuestra opinión es que un estudio separado, detallado, de cada fuente de conocimiento es un paso necesario para entender los retos de la DSP.

En este capítulo, nos concentraremos en las relaciones conceptuales como una fuente de información para los sistemas de DSP. La hipótesis básica es que los sentidos correctos para las palabras en una expresión en lenguaje natural tendrán unas relaciones más cercanas (en una red semántica) que combinaciones incorrectas de los sentidos. Por ejemplo, en *Spring is my favorite season*, el sentido *springtime* de spring tiene una relación de hiponimia con el significado *season of the year* de *season*, mientras que cualquier otra combinación de sentidos (por ejemplo, *spring* como fuente y *season* como temporada de deportes) tiene unas relaciones semánticas más débiles.

Nuestra intención es llevar a cabo un estudio en profundidad (a través de una eva-

luación empírica exhaustiva) del papel que las relaciones conceptuales pueden jugar de cara a una DSP precisa. Como punto de partida elegimos uno de los algoritmos más prometedores de DSP, basado en una medida de densidad conceptual (Agirre and Rigau, 1996). Como en su trabajo, hemos utilizado la red semántica de WordNet (Fellbaum, 1998), como base de datos léxica que proporciona sentidos para las palabras y relaciones semánticas entre ellos. WordNet-1.7 incluye 192460 sentidos de palabras para el inglés, y también existen versiones a gran escala para muchos otros idiomas (Vossen, 1998).

Comenzaremos por generalizar el algoritmo, parametrizando muchos aspectos del sistema original, incluyendo la propia fórmula de la densidad conceptual. Las estrategias incorporadas al algoritmo incluyen todas las posibilidades para explotar las relaciones semánticas de WordNet en las que pudimos pensar. Finalmente, llevaremos a cabo una evaluación exhaustiva, ejecutando el sistema en más de cien configuraciones diferentes contra todos los nombres en la colección SemCor (Francis and Kucera, 1967), la colección de prueba anotada de mayor tamaño que conocemos. El algoritmo original no había sido probado previamente contra toda la colección SemCor.

Dado que el algoritmo presentado depende de varios parámetros, hay una gran variabilidad de posibles pares (precisión,recall). En general hemos optado por optimizar el recall, para comprobar qué resultados se obtendrían si utilizaramos únicamente esta fuente de información como un sistema completo, pero también nos ha interesado averiguar qué niveles de precisión se pueden lograr, al precio de un recall bajo, con el objeto de dilucidar si puede usarse como heurística de gran precisión en un sistema más complejo.

Dado que esta evaluación ofrece resultados un tanto parciales (es importante saber si los resultados son transportables a otras colecciones y otros dominios) decidimos re-evaluar el algoritmo con las colecciones prueba de SENSEVAL-2, con unos resultados dispares entre la tarea de *todas las palabras* y la de la *muestra léxica* que complementan a la evaluación realizada sobre SemCor.

En la Sección 2, explicamos el algoritmo principal y todas las variantes. En la Sección 3 describimos la evaluación realizada y los resultados obtenidos. Finalmente la Sección 4 describe las conclusiones principales.

4.2. Descripción del algoritmo

Los elementos básicos para el algoritmo son una Base de Conocimiento Léxica (BCL) con información conceptual (tal como los *s*ynsets de WordNet, o conjuntos de términos sinónimos), una relación binaria R (habitualmente la relación es-un en una taxonomía) entre los conceptos en la BCL y una fórmula de densidad conceptual (ver más abajo) que devuelva la densidad conceptual de un concepto con una cierta cantidad de subconceptos *activados* (con respecto a R).

Para desambiguar una palabra hacemos lo siguiente: Primero, tomamos el contexto que rodea a la palabra y formamos una ventana de un radio fijado dado. Después clasificamos por orden los sentidos de la palabra central siguiendo estos pasos:

- Buscamos los sentidos de todas las formas de la ventana. Para cada sentido de cada palabra, tomamos un número de conceptos relacionados a través de la relación R, y los pesamos de acuerdo a alguna fórmula.
- Para cada sentido de la palabra central de la ventana, el concepto (relacionado con el sentido mediante una aplicación transitiva de R) que tenga la densidad conceptual más alta nos da la densidad conceptual de ese sentido.
- Después normalizamos la clasificación de los sentidos de la palabra y tomamos los valores resultantes como salida del algoritmo.

Estos pasos definen una *plantilla* de algoritmos de densidad conceptual con un amplio rango de posibilidades. En la siguiente sección discutiremos los parámetros que hemos considerado y los valores que hemos probado.

4.2.1. Parámetros

Relación Transitiva R La más obvia es tal vez la relación de hiperonimia, pero también hemos considerado la unión de relaciones semánticas tales como la de la hiperonimia y la meronimia (la relación *es-parte-de*).

Medida de densidad conceptual Hemos probado cuatro medidas diferentes de densidad conceptual:

1. La fórmula original de densidad conceptual de Agirre-Rigau (Agirre and Rigau, 1996):

$$CD(c,m) = \frac{\sum_{i=0}^{m-1} adesc^{i^{\alpha}}}{\sum_{i=0}^{h-1} adesc^{i}} \tag{4.1}$$

   donde *adesc* es el número medio de descendientes del concepto c según $R^{*^{-1}}$, m es el número de marcas (o synsets activados) en la subjerarquía de c. Y h es la profundidad de la subjerarquía bajo c. Hemos llamado a esta fórmula Agirre-Rigau Estricta (SAR). El valor α =0.2 optimiza los resultados para WordNet 1.4.

2. La misma fórmula sin α (que fue optimizado por Agirre y Rigau para una colección de prueba diferente, mucho más pequeña, y para otra versión de WordNet). La hemos llamado Agirre-Rigau (AR).

3. La *fórmula logarítmica* (LF):

$$LF(c,m) = \frac{1}{desc_c} log_2 d \sum_{i=0}^{m-1} adesc^{i}$$

   Donde d es la profundidad del concepto c en la jerarquía. Esta es una fórmula similar a la de Agirre-Rigau con un factor de corrección que favorece a los conceptos más específicos (más profundos en la jerarquía).

Tamaño de la ventana Hemos experimentado con varios tamaños de ventana.

Selección de sentidos relacionados Por lo que respecta a seleccionar los conceptos relacionados con un sentido a través de R hemos tomado en cuenta diversas posibilidades.

- Primero, tenemos un parámetro para eliminar las relaciones entre los niveles superiores de la jerarquía inducida por el cierre transitivo de R (que hemos denotado por $R*$, como es acostumbrado). La razón que motiva esta sección del experimento es que los niveles superiores en jerarquías conceptuales amplias tienden a ser altamente subjetivos. Si hay un concepto representativo del tema discutido en la ventana de palabras y este concepto va a ser de utilidad en la tarea de desambiguación, no debería ser demasiado abstracto o genérico (como suelen ser los conceptos de los niveles superiores) en relación con los sentidos de las palabras que se están

desambiguando. Las fórmulas de densidad conceptual están diseñadas para reflejar este hecho pero para tamaños grandes de ventana parece inevitable que los *synsets* de la *top ontology*[1] de WordNet obtengan densidades altas. Un valor de cero en este parámetro representa considerar la jerarquía entera.

- Introducimos otro parámetro, l, para considerar sólo los l conceptos más cercanos a través de la aplicación transitiva de R. En otras palabras, cuando calculemos la densidad conceptual de un concepto c, no consideraremos el peso de un subconcepto s si tenemos que iterar sobre R más de l veces hasta llegar a c. La idea que hay detrás de esta restricción es que un concepto c y su hiperónimo inmediato estarán estrechamente relacionados semánticamente (como sería el caso entre highway_1 [2] y road_1 en WordNet); por el contrario, aunque una autopista es ciertamente una entidad no está claro que esta información vaya a tener algún impacto en la tarea de desambiguación. Podría parecer que este parámetro y el descrito en el punto precedente proporcionan resultados similares, pero en nuestros experimentos muestran una conducta muy diferente. Un valor de cero en este parámetro representa tomar en consideración todos los conceptos relacionados a través de R^*.

Pesado de los sentidos Para calcular la densidad conceptual de un concepto c, en la jerarquía inducida por R hemos considerado tres posibilidades para contar y pesar cuántas marcas o *synsets activados*, m, caen bajo él:

synsets Contar cada sentido de las palabras de la ventana relacionado con c como una marca. Esta es la formulación original de Agirre y Rigau. El problema aquí es que las palabras interfieren severamente entre ellas. Si tomamos como ejemplo la palabra *end*, que tiene 14 sentidos en WordNet[3], y dibujamos la jerarquía de la hiperonimia para los sentidos bajo *entity* (con algunos nodos intermedios omitidos por claridad) obtenemos los resultados de la figura 4.1.

Es fácil ver aquí que los restantes ocho sentidos de *end* (que no son hipónimos de *entity*) probablemente se verán discriminados con respecto a estos porque, en ausencia de contexto, el concepto *object* en la figura consigue una alta densidad. Si añadimos más palabras como contexto a la ventana,

[1] La top ontology de WordNet es un reducido conjunto de synsets situado en la cima de la jerarquía de la hiperonimia.

[2] Seguimos la convención de que w_i es el i-ésimo sentido de WordNet de la palabra w.

[3] El ejemplo está tomado de WordNet-1.5

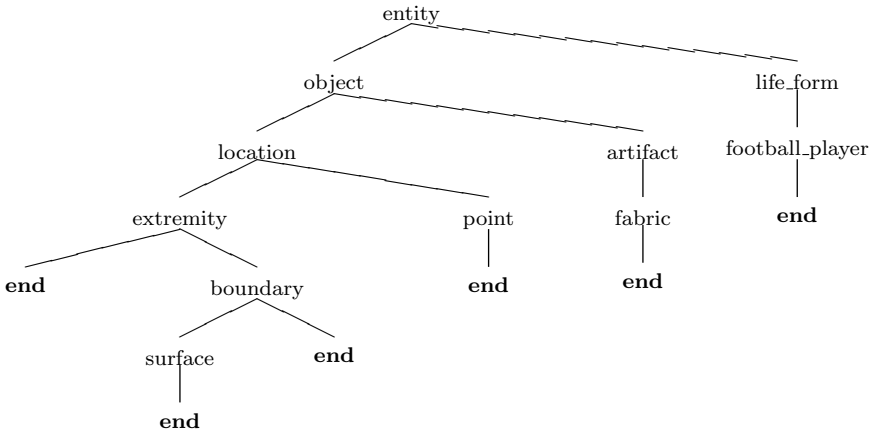

Figura 4.1: La jerarquía de *end*

lo más probable es que la mayoría de los sentidos caigan bajo la subjerarquía de *entity* (es la más grande con mucha diferencia) y el algoritmo descarte los otros sentidos. Otro efecto adverso de las palabras altamente polisémicas es que tienden a dominar las medidas de densidad conceptual. Por ejemplo, *end* tiene 14 sentidos y por tanto 14 marcas en las medidas de densidad, y eso parece un poco inapropiado teniendo en cuenta que alrededor de un tercio de las palabras en texto corriente son monosémicas. Con el objeto de minimizar estos efectos, hemos probado dos formas más de pesar los sentidos:

fraccional Contar para cada sentido de una palabra en la ventana $1/m$ (donde m es número total de sentidos de esa palabra) para prevenir que una palabra altamente polisémica sesgue la densidad conceptual, aunque probablemente esto no impedirá que algunas palabras se desambigüen a sí mismas.

palabras Contar como marcas de la subjerarquía de un concepto c, sólo el número de palabras diferentes en la ventana que contribuyen con sentidos bajo c. De este modo, todas las palabras en la ventana contribuirán en la misma medida y también una alta densidad intrapalabra (usualmente derivada de la finura de grano de WordNet) no debería discriminar los sentidos de esa palabra fuera de esa área. Esta forma de pesado comparte mucho con el enfoque de Marcas de Especificidad expuesto en (Montoyo et al., 2001), desarrollado de forma independiente en la misma época.

4.3. Evaluación

Como hemos indicado, los parámetros del algoritmo han sido ajustados para optimizar el recall sobre SemCor, aunque también estamos interesados en conocer su potencial de precisión. Primero presentaremos los resultados sobre esta colección y más adelante los completaremos con los resultados de evaluar sobre las colecciones de SENSEVAL-2.

4.3.1. Evaluación sobre SemCor

La evaluación se ha realizado sobre la colección SemCor (Miller et al., 1993), un conjunto de 187 documentos donde todas las palabras con contenido semántico están anotadas con el sentido más apropiado de WordNet. En nuestra evaluación, cada una de las configuraciones probadas del algoritmo de DSP ha sido evaluada sobre cada nombre de cada documento de SemCor y exclusivamente sobre los nombres, por lo que los resultados no son extrapolables a rendimiento en general.

El comportamiento del sistema se aporta medido sobre *precisión* y *recall*, tal como fue definido para la primera edición del SENSEVAL:

El sistema de puntuación permite puntuaciones entre 0 y 1, cuando el sistema devuelve más de un sentido por palabra, con la masa de probabilidad compartida entre ellos. La precisión se calcula dividiendo las puntuaciones del sistema sobre los sentidos correctos entre número de items respondidos. El recall se calcula dividiendo las puntuaciones del sistema sobre los sentidos correctos entre el número total de items a ser desambiguados.

Esta medida compara correctamente desambiguaciones contra todos los nombres en la colección, por tanto, un sistema que sea muy preciso pero que tenga poca cobertura tendrá también un recall bajo.

4.3.2. Rendimiento sobre nombres

El cuadro 4.1 compara el algoritmo de Agirre-Rigau original, nuestro mejor sistema de densidad conceptual (al que hemos llamado ARF) y una medida de referencia: l sentido más frecuente (que define el primer sentido de WordNet, que en este caso no puede considerarse una heurística porque esta información sólo se conoce después de

desambiguar a mano). La precisión y el recall del sentido más frecuente no coinciden porque en 51 nombres la anotación manual del lema es errónea y no se encuentra en WordNet. Por ese motivo la cobertura no es del 100 %.

Algoritmo de DSP	Cobertura	Precisión	Recall
ARF	99.94 %	46.31 %	46.31 %
Agirre-Rigau	93.31 %	39.92 %	37.25 %
Sentido más frecuente	99.94 %	78.39 %	78.34 %

Cuadro 4.1: Rendimiento sobre los nombres

En otras ocasiones hemos comparado con la heurística aleatoria, pero en esta ocasión hemos decidido no emplearla. La razón para ello es que consideramos que un sistema aleatorio no tiene ningún conocimiento sobre los datos. En SemCor, los términos multipalabra (monosémicos casi siempre) vienen anotados manualmente, de forma que repartir el peso a partes iguales entre los sentidos da unos resultados relativamente buenos. En nuestra opinión, en una situación real, la detección de términos multipalabra es un mérito con un impacto muy importante para un sistema, un mérito que un sistema *aleatorio* no puede tener. De ahí que no comparemos con una heurística aleatoria que podría parecer que tiene un rendimiento notable, cuando en un caso real no sería así. Del mismo modo, el sentido más frecuente no puede entenderse como una heurística, puesto que es necesario tener desambiguado el texto para poder aplicarlo.

En (Gale et al., 1992a) se defiende que la cota inferior de un sistema de DSP debería ser el rendimiento de la *heurística* del sentido más frecuente, y la cota superior el acuerdo entre los anotadores humanos. En este punto disentimos de Gale et al. puesto que la precisión del sentido más frecuente en este caso (sobre los nombres de SemCor) es de un 78 %, cifra muy elevada, que parece más bien una cota superior de lo que se puede conseguir que una cota inferior. Los creadores de SemCor estimaban la tasa de error en la anotación manual *en torno a un 10 %*. No tenemos conocimiento de la existencia de sistemas evaluados sobre SemCor en esa franja de recall (entre 78 % y 90 %).

Hay que mencionar también que en nuestro algoritmo hemos decidido aglutinar todo el peso de la respuesta sobre el sentido con más densidad conceptual, para evitar dispersar el peso de la respuesta. Hemos hecho esto porque pensamos que es beneficioso para los resultados según el tipo de evaluación de SENSEVAL. El algoritmo original

de Agirre y Rigau es anterior a la competición y no realizaba esta consideración, lo que puede influir en los resultados de la comparación.

Los resultados presentados en (Agirre and Rigau, 1996) eran más prometedores que los obtenidos en esta evaluación, pero fueron obtenidos sobre una colección de prueba casi 50 veces menor que toda la colección SemCor (sólo utilizaron cuatro documentos). Además las definiciones de precisión, cobertura y recall utilizadas eran diferentes.

Nuestro sistema consigue un recall del 46.31 %, una mejora relativa del 24.32 % sobre el sistema original de Agirre-Rigau. El algoritmo de estos últimos emplea un tamaño de ventana de 35, que explica la cobertura ligeramente inferior a la obtenida por nuestro sistema (nuestro sistema utiliza una ventana de tamaño 271). Se trata de una mejora considerable con respecto al algoritmo original, pero los resultados siguen estando muy por debajo de la heurística del sentido más frecuente. La comparación con el recall del 78 % de esta sencilla heurística podría llevar a descartar las relaciones conceptuales como una fuente de información para la DSP. Esto sería, sin embargo, un error, por varias razones:

- Las anotaciones manuales, tomadas como norma de corrección, están sesgadas en favor del primer sentido de WordNet, que corresponde al sentido más frecuente. Los anotadores humanos, en un tarea de anotar todas las palabras de un corpus, tienen que seleccionar el sentido apropiado para una palabra distinta cada vez. Cada palabra tiene más de cinco sentidos en media. Inevitablemente, el anotador tiende a seleccionar el primer sentido que parece encajar en el contexto, y esto produce un sesgo en favor de los primeros sentidos. Estudios sobre la evaluación de sistemas de DSP (Resnik and Yarowsky, 1999; Kilgarriff and Rosenzweig, 2000) hablan a favor de una tarea de anotación sobre una muestra léxica, en la que el anotador anota repetidamente apariciones de una misma palabra, alcanzando de tal manera una mínima familiaridad con los sentidos de la palabra escogida.

- Dejando de lado los problemas de la anotación manual, la tarea de todas las palabras implica que el sistema debe intentar desambiguar repetidamente apariciones de palabras muy comunes, que pueden tener veinte sentidos distintos o más en la base de datos. Estos términos son casi imposibles de desambiguar, y probablemente su correcta desambiguación es casi inútil para la mayoría de las aplicaciones.

- La comparación con el sentido más frecuente no puede hacerse sin hacer la reflexión de que el sentido más frecuente está muy cerca del límite de acuerdo entre

anotadores humanos (intertagger agreement), es decir, de lo que se considera una desambiguación óptima. En el caso del SemCor, además, el sentido más frecuente es mucho más que una heurística supervisada.

- Nuestro algoritmo asigna probabilidades a los sentidos (a diferencia de la heurística del más frecuente) y la distribución global de probabilidades produce mejores resultados en un sistema de recuperación textual basado en recuperación conceptual que la heurística del sentido más frecuente, como se ha comunicado previamente en (Vossen et al., 1999). Esto es una indicación de que la medida de recall en una tarea de DSP pura podría no reflejar la utilidad de la DSP en aplicaciones finales de lenguaje natural. Esta medida indirecta es una prueba de la utilidad potencial de las medidas de densidad conceptual para DSP.

- Nuestros resultados sugieren que el algoritmo debe ser muy eficiente para el tipo de tarea para la que lo diseñaron sus autores; la desambiguación del genérico en entradas de diccionario. Las entradas de los diccionarios a menudo constan en el caso de los nombres de una o dos frases que describen el sentido del nombre en cuestión haciendo referencia al *genérico* y a sus propias características diferenciadoras respecto de éste. Para que la definición sea de alguna utilidad el genérico en cuestión suele ser un hiperónimo, y no uno muy lejano (una frase como *Un plátano es una fruta* parece ciertamente más informativa que *Un plátano es una entidad*). Desambiguando con contexto de frase y limitando los niveles de exploración a tres la situación parece ajustarse bastante a lo que proporcionan muchas entradas de diccionario. Aquí nos hemos limitado a desambiguar texto plano, pero dada la alta precisión alcanzada a nivel de frase (un 65 %, como veremos en la siguiente subsección) pensamos que el algoritmo resultaría muy eficiente en la desambiguación del genérico en entradas del diccionario. De hecho, si el recall no es mayor es debido a que la presencia de hiperónimos cercanos en la misma frase no es corriente en el texto plano. Y porque este algoritmo no permite discriminar entre dos sentidos de la misma palabra que sean hermanos en la jerarquía, situación no infrecuente en WordNet.

Una conclusión más apropiada sería, entonces, que las medidas conceptuales aisladamente son insuficientes para llevar a cabo una desambiguación del sentido de las palabras en texto plano precisa. En este punto coincidimos con (Voorhees, 1993).

Una mejora que hemos introducido en el sistema es que hemos comprobado que con cierta frecuencia, el algoritmo asigna la misma puntuación a todos los sentidos. Esto se explica en parte porque el algoritmo no puede discriminar entre sentidos

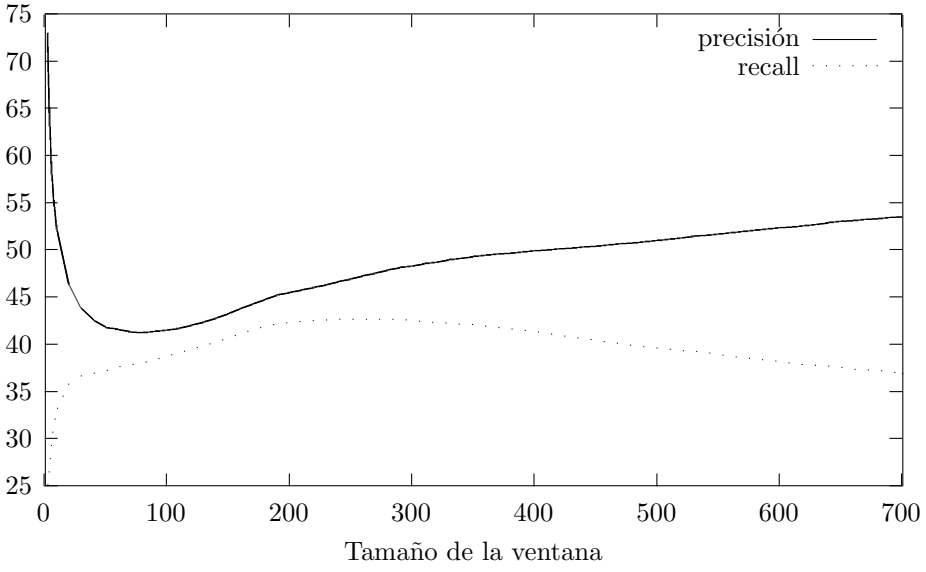

Figura 4.2: Efecto de la variación del tamaño de la ventana

que son hijos del mismo padre (hiperónimo) en la jerarquía de WordNet y empatan sistemáticamente.

Una posible solución a este hecho sería utilizar unas heurísticas de coocurrencia como las empleadas en (Montoyo, 2002) que se aplicaban para puntuar los sentidos que empataban después de aplicar el método de marcas de especificidad.

Por esta razón hemos decidido eliminar las respuestas que coincidían con una heurística de repartir el peso de la respuesta a partes iguales. Lógicamente, al hacerlo ha disminuido el recall y aumentado la precisión. Los resultados del sistema ahora son los siguientes: Se desambiguan 72896 casos (un 82.78 %), con una precisión de 49.9 % y un recall de 41.31 %. Todos los datos de evaluación que siguen se han tomado eliminando este tipo de respuestas.

Nos ocuparemos ahora en la evaluación por separado de todas las variantes introducidas en el algoritmo original, que nos llevaron hasta la mejor combinación de parámetros, que explicaremos a continuación.

4.3.3. Tamaño de la ventana

La figura 4.2 muestra el comportamiento del algoritmo con tamaños de ventana que oscilan entre 1 y 701 nombres. Los resultados nos resultan completamente inesperados. En los experimentos anteriores realizados sobre este mismo algoritmo (Fernández-Amorós et al., 2001a), la evaluación se realizaba sobre otra versión de WordNet (1.6) y sin eliminar las respuestas pseudoaleatorias. Tampoco se evaluó la precisión del sistema. En aquel trabajo se pensaba que el recall se estabilizaba a partir de una ventana de 150 nombres más o menos. Lo cierto es que ahora hemos comprobado el espectro de posibilidades casi completo. La evaluación proporcionada en esta tesis, aunque es hasta cierto punto complementaria de la aportada en (Fernández-Amorós et al., 2001a) proporciona una mayor introspección en el problema, por lo que no merece la pena detallar aquí los hallazgos de este trabajo previo.

El número medio de nombres por documento es de 422 y el máximo es de 649, nosotros hemos llegado hasta una ventana de 700. Alguien podría preguntarse por qué la gráfica no es constante entre los tamaños de ventana 651 y 701. La respuesta se encuentra en la forma de calcular la ventana en los extremos del documento. Si hablamos por ejemplo, de una ventana de tamaño 51, sería en general una ventana de radio 25 por cada lado. Esto no es posible en los extremos del documento: el primer nombre de un documento, sin ir más lejos, no tiene contexto por la izquierda, el segundo tampoco tiene un posible contexto de 25 palabras y así sucesivamente. Con los nombres del final pasa lo propio. Hemos elegido ajustarnos lo mejor posible al *contexto lateral*. Para el primer nombre cogemos ese y los 25 nombres siguientes como ventana. Para el segundo, el contexto por la izquierda sería de un nombre y el contexto por la derecha de 25, para el tercero, 2 por la izquierda y 25 a la derecha (si la longitud del documento lo permite, claro) así hasta que haya 25 nombres en el contexto por la izquierda, la ventana esté completa y podamos avanzar una posición la ventana. La misma idea se repite al final del documento. De esta forma la ventana no siempre es del tamaño marcado pero hay un equilibrio en cuanto a los contextos laterales.

La forma de lograr la gráfica completa, por tanto, sería ir hasta un tamaño de ventana de 1299, para que en la primera ventana de desambiguación del documento más largo, la ventana abarque ya el documento completo. De todas formas, en valores de esta escala, los cambios son cada vez menos apreciables.

A diferencia de lo que pensábamos, el recall no se estabiliza a partir de un tamaño de ventana grande. Ahora que hemos ampliado el rango de los experimentos y descartado las respuestas pseudoaleatorias, comprobamos que disminuye para tamaños

de ventana grandes, alcanzando un valor óptimo con una amplitud de la ventana de 271 nombres.

La precisión alcanza un valor óptimo con una ventana de tres nombres, aunque con una bajísima cobertura, como indica la gran separación entre la curva de la precisión y del recall en ese punto. Si solo permitimos saltar un nivel, los resultados son 28.35 % de cobertura, 74.16 % de precisión y 21.03 % de recall para un tamaño de ventana tres.

Esto ha llamado nuestra atención, de modo que hemos querido calcular también los resultados a nivel de frase. Se desambiguan 33600 nombres (un 38.16 % del total), con una precisión de 65.13 % y un recall de 24.85 %.

Es conveniente tener en cuenta que el porcentaje de expresiones monosémicas (ya que hablamos de nombres y de términos multipalabra de WordNet anotados como nombres en SemCor) es alto, un 20.26 %.

4.3.4. Tipo de relación conceptual

El cuadro 4.2 muestra los resultados del algoritmo usando distintos tipos de relaciones semánticas. Se ha considerado la puntuación de cada caso de prueba entre 0 y 100. Aparentemente, las relaciones de meronimia/holonimia no añaden ninguna información útil a la hiperonimia, sin embargo, por sí solas obtienen una precisión notable, aunque con un recall bajo debido a una baja cobertura.

Relación	Precisión	Recall
Hiperonimia	49.90 %	41.31 %
Hiperonimia + Meronimia	49.90 %	41.31 %
Hiperonimia + Holonimia	49.90 %	41.31 %
Meronimia	61.32 %	25.07 %
Holonimia	60.89 %	25.18 %

Cuadro 4.2: Precisión y recall con diferentes relaciones conceptuales

4.3.5. Fórmula de Densidad Conceptual

Los efectos de la fórmula de densidad pueden verse en el cuadro 4.3. Los nombres de los sistemas son el prefijo ARF + el nombre de la fórmula. La formulación LF

Sistema	Intentadas	Cobertura	Precisión	Recall	Puntuación
ARF-AR	72896/88058	82.78 %	49.90 %	41.31 %	3637486
ARF-LF	84685/88058	96.17 %	38.81 %	37.32 %	3286283
ARF-ARS	84658/88058	96.14 %	36.12 %	34.73 %	3058018

Cuadro 4.3: Efecto de las medidas de densidad conceptual

(fórmula logarítmica) se comporta peor que la fórmula original de Agirre y Rigau. Por otra parte, el parámetro α, que fue ajustado a 0.2 con el objeto de optimizar la desambiguación sobre cuatro documentos particulares del SemCor en WordNet 1.4 es claramente inadecuado para evaluar contra todos los documentos del SemCor en WordNet 1.7: $\alpha = 1$ (AR) produce una mejora del 19 % sobre α =0.2 (ARS).

Las diferentes fórmulas dan cifras de recall entre 34.73 % y 41.31 %, demostrando que escoger una fórmula adecuada tiene un impacto directo en los resultados. Tal vez una fórmula más adecuada podría mejorarlos aún más.

4.3.6. Selección de los synsets

Eliminación de los niveles superiores

La figura 4.3 muestra los efectos de eliminar las relaciones entre los niveles superiores de la jerarquía. Contrariamente a nuestras suposiciones, eliminar solamente los dos niveles superiores afecta negativamente al recall del sistema. Eliminar más de seis niveles produce una conducta prácticamente aleatoria, ya que la mayoría de la información nominal de WordNet se encuentra en esos primeros niveles.

Límite superior sobre las cadenas jerárquicas

Los efectos de limitar la inspección de las cadenas de hiperonimia se muestran en la figura 4.4. La gráfica muestra que el algoritmo es inútil sin dicha limitación y que el límite óptimo es tres.

Este criterio confirma que subir en la jerarquía sin límites introduce ruido, debido a conceptos muy genéricos, que arruinan la actuación del algoritmo.

Estos resultados parecen indicar que el algoritmo de DSP no se está comportando co-

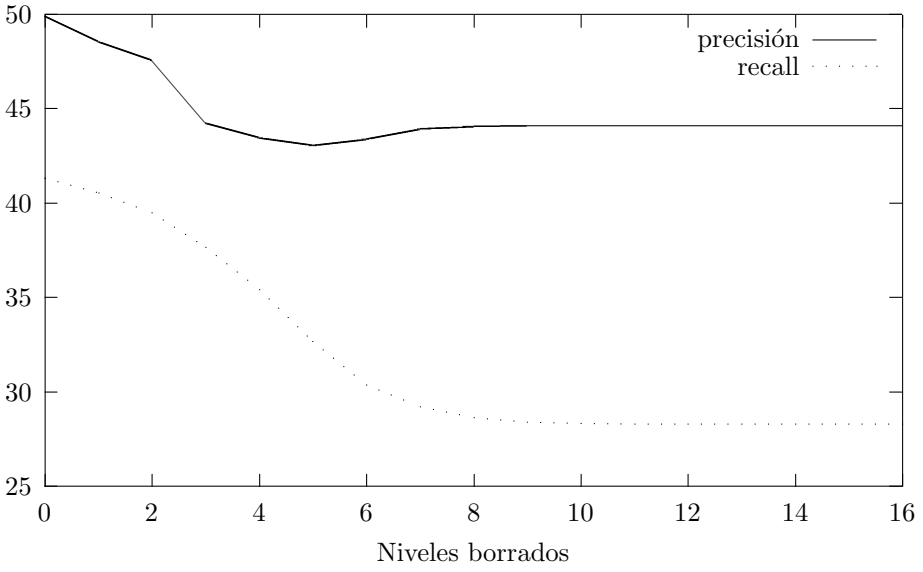

Figura 4.3: Efecto del borrado de relaciones en los niveles superiores de la jerarquía

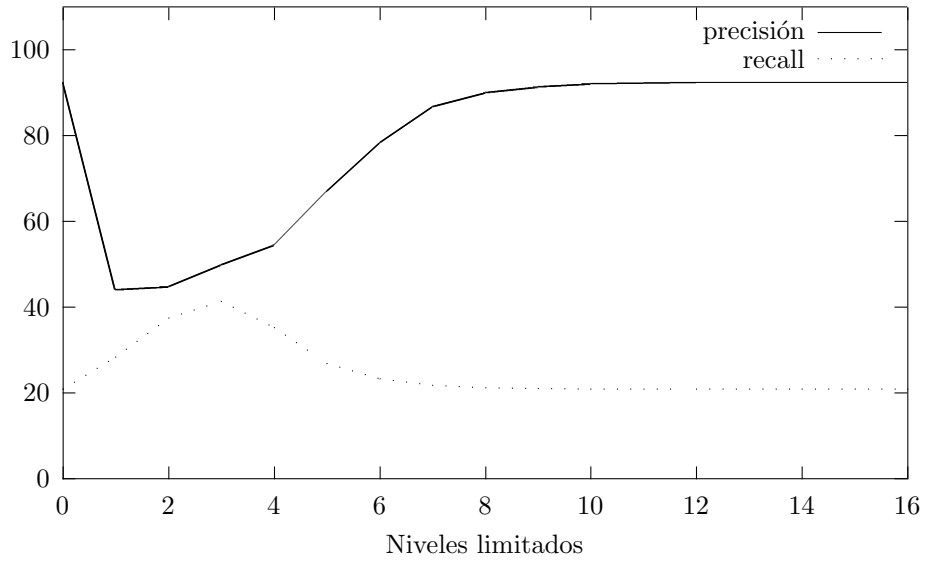

Figura 4.4: Efecto de la limitación de niveles sobre el algoritmo

mo se esperaba: Los niveles superiores participan activamente en la desambiguación, y por tanto la medida de densidad conceptual está usando conceptos que son demasiado generales para ser significativas para la desambiguación. Esto puede explicar en parte por qué las medidas de densidad no producen mejores resultados.

4.3.7. Pesado de sentidos

El cuadro 4.4 muestra el recall para las tres aproximaciones al pesado de los sentidos. Sorprendentemente, asignar pesos bajos a los sentidos de las palabras muy ambiguas (*fraccional*) no mejora en precisión el rendimiento sobre la aproximación estándar (*synsets*). Tomar el número de palabras distintas en la medida de densidad (*palabras*) consigue una precisión casi tan buena como contar synsets, con un recall considerablemente mayor.

Criterio	Precisión	Recall
Palabras	49.91 %	41.31 %
Fraccional	40.61 %	39.13 %
Synsets	51.58 %	30.81 %

Cuadro 4.4: Efectos del pesado de sentidos

4.3.8. Comportamiento sobre diferentes categorías de texto

Los documentos de SemCor, una fracción del Brown Corpus (Francis and Kucera, 1982) están clasificados según un conjunto predefinido de dominios (Prensa, Ficción general, Rosa, Humor, etc . . .) Es interesante ver como el rendimiento de la DSP varía a lo largo de las diferentes categorías documentales. En el cuadro 4.5, el rendimiento general está desglosado según dichas categorías. Las categorías donde la densidad conceptual funciona mejor están colocadas al principio de la tabla.

Los resultados son dignos de comentario. Mientras que la polisemia media no varía excesivamente en función de la categoría de documento, el sistema de DSP funciona mejor sobre las categorías de no ficción (*Prensa: Reportajes, Editoriales , Cultura general y Miscelánea*, etc . . .), y peor sobre las categorías de ficción (*Ficción de misterio y detectivesca y ficción de aventuras y del oeste*). Esto parece corroborar la hipótesis de que la DSP tiene más aplicabilidad en documentos técnicos, donde

Categoría textual	Polisemia	Precisión	Recall
A. Prensa: Reportajes	5.74	53.15 %	36.33 %
F. Cultura popular	5.38	52.86 %	46.65 %
B. Prensa: Editoriales	5.75	52.29 %	45.33 %
J. Profesional	5.31	52.13 %	45.95 %
H. Miscelánea	5.29	50.52 %	44.97 %
R. Humor	5.81	49.70 %	43.62 %
E. Habilidades y aficiones	5.61	49.68 %	45.77 %
G. Literatura, biografías, ensayos	5.58	49.12 %	42.43 %
D. Religión	5.59	48.65 %	42.07 %
P. Rosa e historias de amor	6.68	47.63 %	37.76 %
K. Ficción general	6.37	46.89 %	39.18 %
L. Ficción de misterio y detectivesca	6.53	46.08 %	34.07 %
C. Prensa: Crítica	5.44	45.81 %	38.49 %
M. Ciencia ficción	5.91	44.62 %	36.86 %
N. Ficción de aventuras y del oeste	6.67	44.52 %	37.34 %

Cuadro 4.5: Rendimiento de la DSP en diferentes categorías de texto

los sentidos de las palabras tienen distinciones más clara, las metáforas son menos comunes, y el contexto proporciona una información de dominio más acertada que en los textos de ficción.

4.3.9. Evaluación contra colecciones SENSEVAL

En este caso hemos decidido, en vez de optimizar para precisión, buscar un compromiso entre la precisión y el recall de modo que hemos maximizado la precisión con el tamaño de ventana a nivel de frase.

Para poder comparar con los otros sistemas hemos tenido que re-evaluar de todos los sistemas participantes en SENSEVAL-2 para obtener los resultados únicamente sobre nombres. La evaluación sólo a nivel de nombre de la tarea de *todas las palabras*[4] puede observarse en la figura 4.5. El sistema, ARF, se sitúa hacia la mitad de la tabla, justo entre los dos sistemas que presentamos a la competición (david_fa_UNED-AW-U y david_fa_UNED-AW-T). Hemos abreviado algunos de los nombres de los sistemas

[4]Véase el apartado 2.2.2.

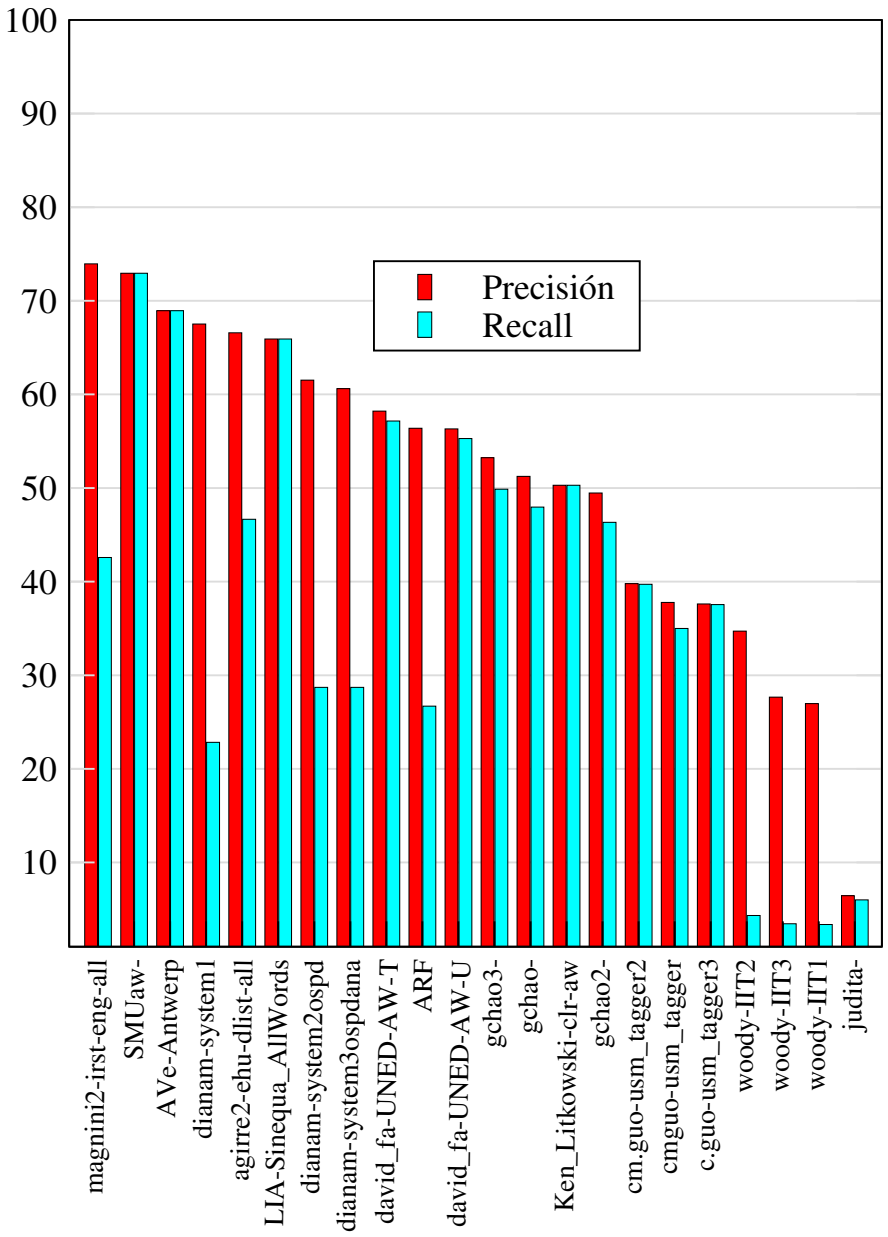

Figura 4.5: Comparación del sistema con los de la tarea de todas las palabras

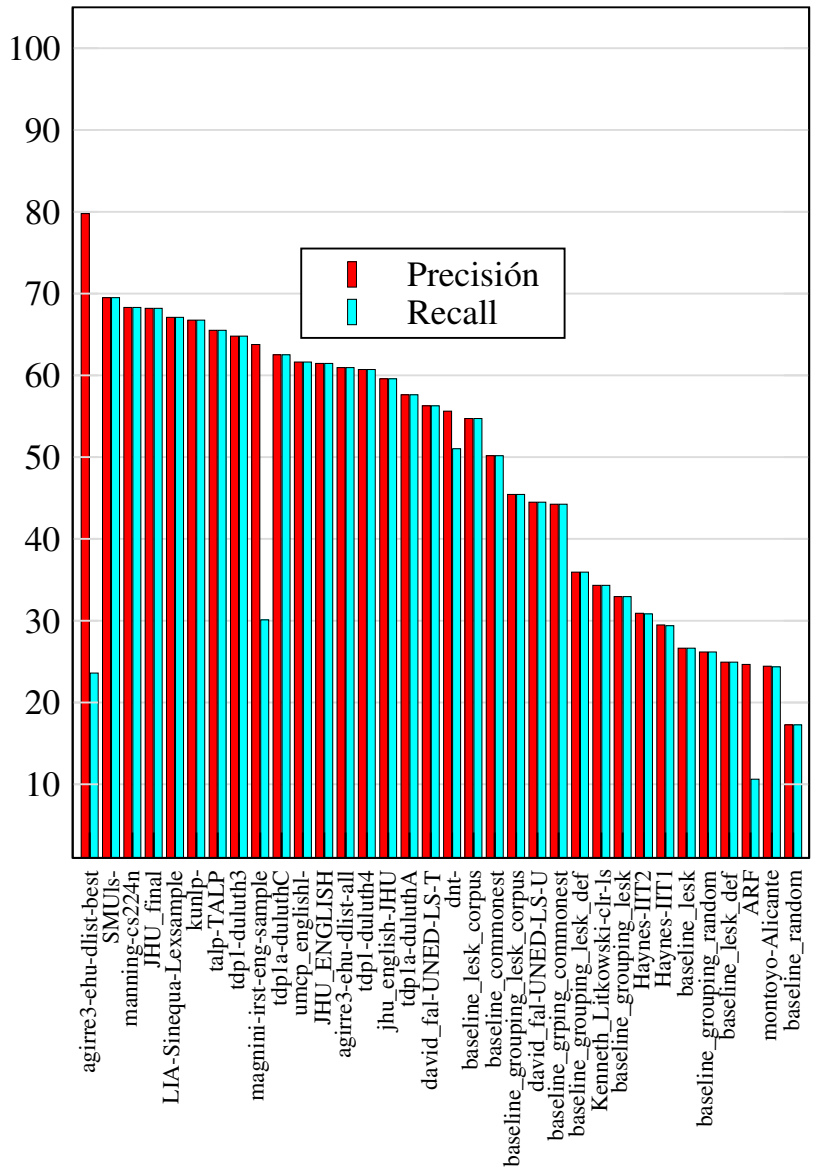

Figura 4.6: Comparación del sistema con los de la muestra léxica

por claridad.

Puede comprobarse como el sistema, que hemos denominado ARF, tiene una precisión bastante alta para ser no supervisado y mejora en precisión al sistema no supervisado que presentamos a la competición.

Para la muestra léxica los resultados pueden observarse en la figura 4.6. El sistema ARF queda el tercero por la cola. Hemos eliminado variantes de algunos sistemas por claridad.

Aunque la mejora de la precisión sobre la heurística aleatoria es considerable, es evidente que el algoritmo no produce buenos resultados sobre estas palabras con un nivel tan alto de polisemia.

Merecería la pena estudiar si las palabras para las que el algoritmo ha tenido un rendimiento más pobre pueden desambiguarse mejor usando información de coocurrencias. De ser así podría ser un indicio de que hay palabras cuyo sentido es más dependiente del dominio contextual que de la jerarquía. Es bien sabido que la jerarquía de WordNet es una clasificación taxonómica que no asocia dominios semánticos. Por ejemplo, *tennis_racket* y *tennis_shoe* pertenecen al dominio del tenis pero no están ligados en WordNet. Esa ligazón estaría asociada al dominio, no a la taxonomía.

4.4. Conclusiones

Hemos aportado una evaluación exhaustiva de varios algoritmos distintos de DSP que se apoyan únicamente en las relaciones conceptuales entre los sentidos candidatos de las palabras. Nuestro punto de partida ha sido el algoritmo de Agirre-Rigau, basado en una medida de densidad conceptual sobre toda la jerarquía nominal de WordNet. Este algoritmo, que tenía un rendimiento competitivo sobre una colección de prueba más pequeña, se comporta peor en una evaluación completa contra todos los documentos del SemCor. Hemos experimentado con varias clases de mejoras al algoritmo, y ajustado todos los parámetros asociados a ellos, obteniendo evaluaciones para más de cien variantes del algoritmo de DSP, incluida la que es idéntica al algoritmo original.

Algunas de las principales conclusiones que hemos sacado de nuestros experimentos son:

- Nuestro sistema se comporta un 24.32 % mejor en cuanto a recall que el algo-

ritmo original de Agirre-Rigau. Esta mejora se obtiene con una implementación de complejidad lineal en tiempo y ha sido usada para desambiguar grandes colecciones de textos en tres idiomas distintos (inglés, español y catalán) dentro del ámbito del proyecto ITEM (Verdejo et al., 2000b).

- Hemos demostrado que, en la práctica, el algoritmo original usa largas cadenas jerárquicas para desambiguar, que están asociadas con asociaciones conceptuales vagas que dan resultados ruidosos. Nuestra configuración ideal usa cadenas de hiperónimos de longitud máxima tres, combinadas con otras optimizaciones para mantener la cobertura del sistema.

- En la cuestión de la longitud del contexto, las cosas están claras respecto al recall. Alcanza un máximo con una ventana de tamaño 271, es decir, que el contexto amplio tiene información útil para la desambiguación. Sin embargo, el comportamiento de la precisión es curioso, es muy alta en contextos muy cortos, especialmente a tamaño de ventana 3 (cuando el primer nombre a la izquierda/derecha es un hiperónimo/hipónimo), aunque esta situación es poco frecuente (cobertura 34.37 %, precisión 72.97 %, recall 25.08 %). Después sufre un brusco bajón, con un mínimo en tamaño de ventana 81 de 41.22 % y más allá se recupera aunque sin volver a los niveles iniciales. Una posible explicación a esto es que los documentos cortos de SemCor están compuestos por varios trozos relacionados pero independientes (en el caso de las noticias, es frecuentes que varias noticias breves compongan el mismo documento), mientras que los largos son más homogéneos (como los documentos que son extractos de una novela).

- El sorprendente comportamiento en cuanto a precisión de las relaciones de meronimia y holonimia por sí solas parece apoyar una hipótesis según la cual un método efectivo de desambiguar consistiría en detectar multitud de fenómenos lingüísticos poco habituales en términos absolutos pero de alta precisión y, en caso de necesidad, estudiar la forma de combinarlos en los casos en los que se den varios fenómenos de este tipo en el contexto de una misma palabra en sentidos contradictorios.

- Hemos proporcionado evidencia cuantitativa que demuestra que la DSP da mejores resultados sobre textos de no ficción, de dominio específico, que sobre ficción general con esta técnica.

El rendimiento de las relaciones conceptuales es relativamente bajo en términos de recall, indicando que las relaciones conceptuales deberían ser combinadas con otros

tipos de información (contextual, sintáctica, información de dominio, etc...). Por este motivo en el siguiente capítulo explotaremos las posibilidades de DSP de una fuente de información que complementará a esta: La información de coocurrencias extraídas de un corpus no anotado.

Capítulo 5

Información de coocurrencias extraídas de un corpus

5.1. Introducción

En este capítulo describiremos las heurísticas desarrolladas para explotar la información de coocurrencia de palabras en un corpus no anotado.

Para ello vamos a utilizar un modelo de lenguaje muy simple. En este caso consistiría en estimar la probabilidad, dada la aparición de una cierta palabra w_1, de la aparición de una segunda palabra w_2 dentro de una ventana de palabras de un tamaño prefijado. Esta información es útil porque las palabras no se distribuyen al azar, sino que hay dependencias entre ellas. Supongamos que tenemos que desambiguar *iglesia* en la frase *Me gusta contemplar la torre de la iglesia* y que tenemos dos sentidos para iglesia, uno caracterizado como *El conjunto de los fieles* y otra como *Edificio donde se reunen los fieles*. Se podría medir la verosimilitud de la coocurrencia de edificio con torre[1] y ver si es mayor que la de conjunto con torre[2]. Si torre está mas relacionada con edificio que con conjunto, como es de suponer, entonces tendría evidencia en favor del primer sentido mencionado. La palabra fieles, al ser común a ambas definiciones no aportaría nada.

Dicha información es fácil de extraer hoy en día de corpora de gran tamaño y pro-

[1] En inglés, building y tower, la verosimilitud es 218.68.

[2] En inglés la verosimilitud de set y tower es 32.73.

porciona una representación sobre las *vecindades* de palabras que puede ser utilizada para desambiguar. En este capítulo demostraremos que dicha información es útil y que debe ser tenida en cuenta a la hora de combinar distintas fuentes de información para la tarea de DSP. Los sistemas que presentaremos al final de este capítulo fueron presentados a la competición internacional de desambiguación SENSEVAL-2. El correspondiente a la tarea de todas las palabras logró situarse en el cuarto puesto en términos de recall (cosa destacable tratándose de un algoritmo no supervisado, puesto que los tres primeros eran supervisados). El presentado a la tarea de la muestra léxica fue el mejor en recall entre los no supervisados (en esta tarea, los resultados oficiales han distinguido entre algoritmos supervisados y no supervisados).

Ya hemos demostrado que la información de tipo jerárquico puede ser muy útil para desambiguar. En este capítulo veremos que la información de las glosas de los sentidos que proporciona WordNet, que no fue utilizada en ningún momento en el capítulo anterior, también es una fuente de información vital si la utilizamos conjuntamente con el componente contextual de las coocurrencias de palabras en textos.

La idea central es que, puesto que las palabras no aparecen al azar, existen relaciones entre ellas de manera que la aparición de una palabra del contexto puede ser la clave para discriminar entre varios de los sentidos de una palabra, esto es, suponer una evidencia positiva en favor de unos, y neutra o negativa con respecto a otros. Un ejemplo sería el siguiente: Queremos desambiguar la palabra *banco* y encontramos a dos posiciones de distancia la palabra *dinero*. Esto nos inclinaría a pensar que la palabra *banco* está siendo utilizada en alguno de los sentidos asociados a la entidad financiera más bien que a un banco del parque. Pero si encontraramos *parque* un poco más allá tendríamos información conflictiva[3]. Puesto que no siempre hay un gran solapamiento entre las palabras de las glosas y las palabras del contexto, a menudo se hace necesario expandir estas glosas con términos relacionados. Es necesario formalizar estas ideas intuitivas, cosa que haremos más adelante en este capítulo.

Hemos incluido en el sistema un reconocedor de las expresiones multipalabra recogidas en WordNet, el inventario de sentidos utilizado en dicha competición. Por motivos de claridad en la evaluación hemos distinguido varias heurísticas que se aplican en cascada (i.e. si una heurística no devuelve resultado para una palabra se pasa a la siguiente heurística), aunque todas ellas se basan en información de coocurrencia extraída de un corpus no anotado junto con las glosas de los sentidos de WordNet.

[3]En inglés, el ejemplo clásico de desambiguación es bank. La verosimilitud de bank y money es 91.98, la de bank y river es 522.40. Parece que river es una palabra mucho más decisiva que money para desambiguar bank. Esto podría ser debido a que el dinero se relaciona fuertemente con muchos conceptos, mientras que los ríos lo hacen en mucha menor medida.

Hemos empleado también la información proporcionada por el inventario de sentidos (WordNet) sobre las frecuencias de aparición de los diferentes sentidos en textos. En la siguiente sección, explicaremos el origen y el modo de procesamiento del corpus no anotado del que se ha extraído la información de coocurrencia y la forma de construir una matriz de coocurrencias primero y después una matriz de vecindad de palabras. En la tercera sección explicaremos las heurísticas que forman nuestra cascada. En la cuarta sección explicaremos los distintos sistemas de desambiguación que hemos construido mediante esas heurísticas. En la quinta veremos la evaluación y los resultados de los sistemas tal como fueron presentados en SENSEVAL-2 con la información desglosada por heurísticas y en la sexta presentaremos nuestras conclusiones.

5.2. Construcción de la matriz de vecindad

5.2.1. Procesamiento del corpus

Antes de construir nuestros sistemas de DSP hemos desarrollado un recurso léxico llamado *matriz de vecindad de palabras*. Los datos de entrada para construir la matriz provienen del Proyecto Gutenberg (PG).

En el momento en que la matriz fue creada, el PG constaba de más de 3000 libros sobre diversos géneros. Hemos adaptado estos libros a nuestros propósitos: En primer lugar, una heurística para detección del idioma fue utilizada para tomar solamente libros escritos en inglés (el idioma en cuyas tareas de desambiguación participamos en la competición); aplicamos una heurística sencilla para determinar el porcentaje de palabras de parada en el texto. Este método de detección del idioma está considerado como de una precisión aceptable para textos largos y este es el caso. Después hemos cortado los *disclaimers* para quedarnos con los textos de los libros, tal cual. El resultado es una colección de alrededor de 1.3GB de texto plano.

Para continuar, hemos segmentado el corpus usando un sencillo autómata finito, después hemos lematizado usando reglas del estilo de WordNet (reglas con excepciones), eliminado los signos de puntuación y las palabras de la lista de parada. También hemos marcado dos clases de palabras: Los nombres propios y los números, con el objeto de reducir un poco la cantidad de items distintos resultantes. Esto redunda en una cierta reducción de la dimensionalidad de la matriz final, además de que las relaciones de asociación por coocurrencia son menos vagas; nuestras medidas de asociación darán un valor más adecuado a (<NUMERO>, dinero) que a (7,dinero) donde

<NUMERO> sería la clase que aglutina a los números, que sufriría más el problema de la dispersión de datos.

5.2.2. La matriz de coocurrencia

Hemos construido un vocabulario de las 20000 palabras o etiquetas más frecuentes en el texto (después de lematizar y eliminar las palabras de parada) ya que hemos etiquetado como tales los nombres propios y números. Hemos construido una matriz simétrica de coocurrencia dentro de un contexto de 61 palabras entre esos 20000 términos (hemos pensado que un contexto amplio de radio 30 sería apropiado, puesto que estamos intentando capturar relaciones semánticas vagas).

5.2.3. La matriz de vecindad

En un segundo paso, hemos construido otras matrices simétricas, que hemos llamado *matrices de vecindad*, usando varias medidas. La más interesante de ellas es una medida similar a la de información mutua entre dos palabras o etiquetas, de tal forma que para dos palabras a y b, la entrada de la matriz para ellas sería $\frac{P(a \cap b)}{P(b)P(a)}$, donde $P(a)$ es la probabilidad de encontrar la palabra a en un contexto cualquiera de la longitud especificada, y $P(a \cap b)$ es la probabilidad de encontrar juntas a a y a b en un contexto cualquiera de ese tamaño. Hemos aproximado dichas probabilidades utilizando las frecuencias de aparición en el corpus usando el estimador de máxima verosimilitud (EMV).

En este caso el cálculo sería el siguiente: Formamos ventanas de contexto de un tamaño prefijado, que vamos moviendo una posición cada vez, empezando por el principio de cada documento y terminando cuando el borde derecho de cada ventana coincida con el final del documento. Esta claro que una aparición de una palabra pertenece en general a varios contextos (salvo que esté en el borde de un documento). Si llamamos f_a a la frecuencia absoluta de a en estos contextos, f_b a la frecuencia absoluta de b en estos contextos, f_{ab} a la frecuencia absoluta de la coocurrencia de a y b en el mismo contexto y N al número total de contextos, el estimador de máxima verosimilitud aplicado a estas probabilidades nos daría que:

$$\frac{P(a \cap b)}{P(b)P(a)} \simeq N \cdot \frac{f_{ab}}{f_a \cdot f_b}$$

Es un hecho conocido que la medida de la información mutua sobreestima la relación entre palabras que ocurren con poca frecuencia o con frecuencias muy diferentes (Dunning, 1993). Para evitar ese problema hemos tomado una postura similar a la de (Church and Hanks., 1989) e ignorado las entradas donde la frecuencia de la intersección era menor que cincuenta.

También hemos introducido un umbral por debajo del cual consideramos la entrada de la matriz como cero por motivos prácticos (sólo estamos interesados pares de palabras fuertemente relacionadas). Hemos fijado este umbral en 2 de forma arbitraria para no perder muchas entradas y que la matriz resultante no quede demasiado dispersa. Pensamos que esta matriz es un recurso valioso que podría ser de interés para muchas otras aplicaciones además de la DSP. Además, podrá crecer en calidad tan pronto como se le suministren nuevos datos de entrada en suficiente cantidad.

La elección de la medida de asociación se tomó por su sencillez, pero experimentos posteriores revelaron que en estos experimentos resultaba más eficaz en términos de precisión que las medidas de la χ^2 y la medida de asociación binomial propuesta por (Dunning, 1993), como veremos en la sección 5.4.4.

Puede parecer un contrasentido que por un lado estemos buscando identificar *relaciones semánticas vagas* como decimos en la subsección anterior y por otro *pares de palabras fuertemente relacionadas*, pero lo cierto es que emplearemos técnicas de combinación de la información con el objeto de ponderar la influencia de cada palabra del contexto sobre cada sentido de manera proporcional a su medida de asociación.

Un ejemplo de las palabras con mayor valor de asociación para algunas de las palabras de la tarea de la muestra léxica de SENSEVAL-2 puede verse en el cuadro C.1 en el apéndice C. Es destacable el hecho de que, a menudo, una de las palabras más asociadas es la palabra misma.

Esta aproximación tiene ciertas similitudes con la de (Schütze, 1992b; Schütze, 1993; Schütze and Pedersen, 1995) pero también presenta características propias:

- Schütze calculó sólo algunas partes de la matriz de coocurrencias (en particular ignoró la parte de coocurrencias entre palabras poco frecuentes, cuando la mayoría de las palabras que aparecen en un texto son poco frecuentes. Aquí esa matriz se calculado completa.

- Hemos aplicado listas de parada, cosa que no se menciona en los artículos de Schütze.

- Schütze realizó experimentos para diez palabras distintas. En esta tesis se ha tomado la opción de desambiguar todas las palabras de un texto.
- Hemos probado diversas posibilidades para calcular la *asociación* entre palabras; información mutua, χ^2 y binomial de Dunning.
- Se ha tenido en cuenta la distancia a la palabra por desambiguar como un factor más a considerar, frente a la aproximación de la bolsa de palabras.
- Las distinciones de sentidos son las de WordNet, de grado fino, mientras que Schütze utilizó sobre todo distinciones de sentidos binarias y ternarias. Schütze realizaba *discriminación de sentidos*, es decir, elegía cuantos sentidos quería para cada palabra sin atenerse a un diccionario, sino especificando un parámetro en sus algoritmos de *clustering*.
- Las asignaciones de los *clusters* a los sentidos de un diccionario (sin este paso es imposible evaluar la desambiguación de forma directa) se realizaban de forma manual, mientras que los métodos presentados en este capítulo son completamente no supervisados.

5.3. Cascada de heurísticas

Hemos desarrollado un lenguaje muy simple para sistematizar los experimentos. Este lenguaje permite la construcción de sistemas de DSP compuestos de diferentes heurísticas que se aplican en cascada, de forma que cada palabra a desambiguar pasa por la primera heurística y si ésta no puede desambiguarla pasa la segunda heurística y así sucesivamente. Por motivos de eficiencia el lenguaje permite ejecutar y evaluar varios de estos sistemas en paralelo. Para las heurísticas que devuelven una puntuación numérica para cada sentido lo que hemos hecho es devolver como respuesta sólo el sentido con mayor puntuación. Es muy difícil conseguir buenos resultados si la puntuación se reparte entre más de un sentido. Sólo lo permitimos en caso de empate entre dos sentidos (es decir, si en la misma heurística dos sentidos obtienen la misma puntuación repartimos al 50 %, si empatan más de dos no desambiguamos). A continuación describiremos las heurísticas que construyen los sistemas.

- **Expresiones monosémicas**.

  Las expresiones monosémicas son simplemente palabras no ambiguas (en el caso de la tarea de todas las palabras) o bien, en la muestra léxica expresiones

multipalabra de WordNet detectadas con éxito y que suelen tener un único significado (una excepción a esto sería *Virgin Mary*, que por un lado es la Virgen María y por otro el equivalente no alcohólico de un *Bloody Mary*, segun WordNet).

Hemos implementado un módulo para detectar estos términos multipalabra. Tomamos las expresiones del fichero de índices de WordNet y utilizamos un algoritmo de backtracking multinivel que tiene en cuenta las capacidades flexivas de las palabras de las expresiones para detectar el mayor número posible de ellas. Hemos probado este algoritmo contra el PG y hemos encontrado millones de estos términos multipalabra.

Parece existir un problema en la metodología de anotación manual de las expresiones multipalabra de WordNet. A menudo expresiones multipalabra que contienen a una palabra pueden constituir una anotación correcta para esa palabra, por ejemplo, *work_of_art %1:06:00::* se considera a veces en la muestra léxica de SENSEVAL-2 como el sentido correcto de *art*. Sin embargo, no siempre ocurre esto. Un ejemplo sería el sentido *short_circuit %1:06:00::*, que nunca es el correcto para *circuit*. Nuestro sistema siempre considera que las expresiones multipalabra aportan nuevos sentidos a considerar en la desambiguación. El rendimiento se ha visto beneficiado por ello en algunos casos y perjudicado en otros, pero los resultados han sido decepcionantes comparados con la eficacia de la detección sobre SemCor (los términos multipalabra detectados por nuestro algoritmo y los lemas anotados coincidían en un porcentaje mucho mayor que en la muestra léxica)[4].

Tras la participación en la competición pensamos que tal vez el motivo por el que nunca se consideraban los sentidos de *short_circuit* para *circuit* o de *the_good_old_days* para *day* podría ser que en WordNet no hay relación de hiperonimia entre estos pares de expresiones (i.e. un cortocircuito no es un tipo de circuito).

Por esa razón, probamos limitando la detección de estos términos multipalabra sólo a aquellos que contienen a palabras de la muestra léxica y además son hipónimos de alguno de sus sentidos.

Desgraciadamente, los resultados empeoraron, lo que hace pensar en algún problema de metodología a la hora de anotar las palabras que están incluidas en expresiones idiomáticas. Dicha metodología debería tener en cuenta, a la hora de anotar manualmente una palabra, que algunos de sus sentidos podrían estar

[4]Estos experimentos no están descritos en esta tesis, en las evaluaciones sobre SemCor publicadas aquí se ha utilizado la lematización de los anotadores humanos.

contenidos en expresiones multipalabra (situación que no siempre se da, como sabemos por nuestra experiencia como anotadores en la tarea del español).

Finalmente el algoritmo de detección de términos multipalabra se aplicó con la lista completa de términos multipalabra de WordNet, es decir, como estaba al principio.

En la colección SemCor las expresiones multipalabra ya están marcadas y no es necesario detectarlas.

- **Filtro estadístico**

  WordNet viene con un fichero llamado cntlist, literalmente *el fichero que lista el número de veces que cada sentido anotado aparece en SemCor*. Utilizamos está información presente en el diccionario para calcular la probabilidad de que una palabra se use en un sentido concreto. Con dicha información eliminamos los sentidos de la palabra por desambiguar que tengan una probabilidad menor a 1/10. Si nos queda un sólo sentido después de esto devolvemos ese sentido como resultado de la heurística, si no, pasamos a otra heurística. Dicho de otra manera, no hemos aplicado técnicas más complejas para palabras que tienen una distribución de sentidos altamente sesgada. En cualquier caso los sentidos eliminados por una heurística pueden ser recuperados por las las siguientes. El filtro estadístico, por ejemplo, rechaza sentidos con una baja frecuencia relativa en SemCor, sin embargo, la heurística de los sentidos enriquecidos (posterior en la cascada) vuelve a considerar esos sentidos.

  Alguien podría argüir que esta heurística es supervisada. En nuestra opinión, utilizar la información presente en cntlist no convierte *per se* a un sistema en supervisado, puesto que se trata de información que se distribuye como parte de la base de datos léxica. De hecho cualquier diccionario debería proporcionar esa información. Los sentidos de las palabras no se inventan, o no deberían inventarse. Es costumbre hacer los diccionarios basándose en las apariciones de las palabras en un gran corpus. A pesar de ello, de los 192460 sentidos de palabras en WordNet-1.7, sólo el 19.7 % de ellos aparece al menos una vez en SemCor y sólo el 11.3 % más de una vez. Es elemental que esta información debe ser tenida en cuenta para no dificultar aún más una tarea ya de por si compleja. Este es un problema que los sistemas supervisados no padecen, puesto que para los sentidos de frecuencias despreciables no hay en la práctica ejemplos de entrenamiento, de forma que pueden ser ignorados sin mayores complicaciones. Por otro lado nuestros sistemas, esta heurística incluida, no hacen uso de ejemplos anotados como entrenamiento ni entrada de ninguna clase.

- **Filtro de vecindad**

  Esta heurística hace uso de la matriz de vecindad. Para asignar una puntuación a un sentido, contamos las coocurrencias de palabras entre el contexto (de la palabra por desambiguar) y las palabras de la definición de cada sentido (las glosas de WordNet en este caso), pesando cada coocurrencia con la entrada en la matriz de vecindad para el par (palabra por desambiguar, palabra coocurrente). Para no favorecer a los sentidos con glosas más largas hemos decidido dividir este peso por la longitud de la glosa, medida en número de palabras. Finalmente, para cada palabra del contexto hemos hecho un último pesado de su contribución a la desambiguación mediante una función de la distancia a la palabra por desambiguar, de manera que las palabras más cercanas obtengan un peso mayor que las lejanas. Esta función debería ser sensible a la categoría gramatical, según lo expuesto en (Yarowsky, 1993), pero por el momento hemos empleado la misma para todas las partes del discurso. También hemos empleado otro factor tomado prestado de la recuperación de información, la *frecuencia inversa de documento* (inverse document frequency) para discriminar que términos son típicos de las glosas y por tanto no significativos para la desambiguación. De esta manera hemos considerado cada glosa como un documento.

  Si s es un sentido de la palabra α y C es el contexto de desambiguación de α, (considerado como conjunto), la puntuación de s sería:

$$\sum_{w \in C} \text{VECINDAD}_{w\alpha} \text{freq}(w, C) \text{pesado}(w, \alpha) \text{freq}(w, S) \text{idf}(w, \alpha)$$

  Donde $\text{VECINDAD}_{\alpha\beta}$, es el valor de la matriz de vecindad para las palabras α y β, obtenido aplicando alguna de las medidas de asociación a la matriz de coocurrencia, $\text{idf}(w, \alpha) = \log \frac{N}{d_w}$, es la frecuencia de documento inversa, siendo N el número de sentidos para la palabra α y d_w el número de glosas en las que aparece w. $\text{freq}(w, C)$ es la frecuencia de la palabra w en el contexto C, $\text{freq}(w, S)$ es la frecuencia de w en la glosa del sentido S y

$$\text{pesado}(w, \alpha) = 0{,}1 + e^{-\text{posic\_distancia}_{(w,\alpha)^2/2\sigma^2}}$$

  donde a su vez $\text{posic\_distancia}(w, \alpha)$ nos dice cuantas posiciones de distancia hay entre las palabras w y α. A σ le hemos asignado el valor 2.17, el valor óptimo determinado empíricamente en nuestros experimentos con los datos de SENSEVAL-1. La función pesado es una campana de Gauss con media cero y desviación típica σ, a la que hemos sumado 0.1 para no perder por completo la información de las palabras lejanas. Su aspecto puede observarse en la figura 5.1

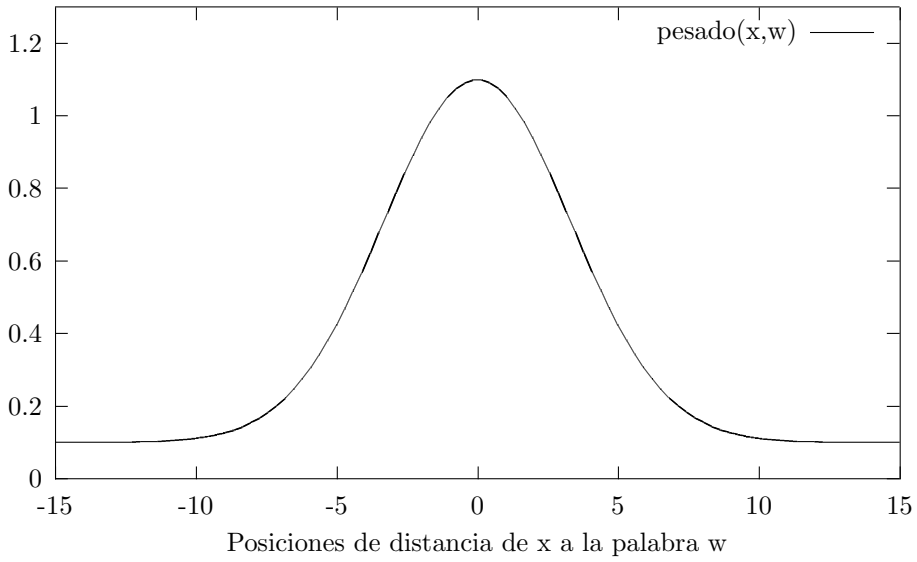

Figura 5.1: Campana de Gauss de pesado de palabras del contexto

La idea es pesar más las apariciones de las palabras que son relevantes según la matriz respecto a la palabra que queremos desambiguar y dar poco peso (posiblemente ninguno) a las palabras del contexto que guardan poca relación con la palabra objetivo.

Además, en la tarea de todas las palabras (que proporciona las anotaciones de categoría morfosintáctica del Penn Treebank) hemos considerado sólo las palabras de contexto que tienen una categoría morfosintáctica *sensible* a la de la palabra por desambiguar. Las categorías compatibles entre sí son: nombres y nombres, nombres y verbos, nombres y adjetivos, verbos y verbos, verbos y adverbios y viceversa en todos los casos. Simplificando un poco serían las parejas de categorías susceptibles de tener una relación intra-sintagmática; hemos asumido que las palabras cercanas y que posiblemente pertenecen al mismo sintagma deben consideradas antes que las lejanas y las que probablemente no pertenecen al mismo sintagma por motivos estadísticos. Obviamente un trabajo futuro interesante sería utilizar un analizador sintáctico, para extraer las palabras del mismo sintagma y utilizar únicamente éstas como contexto.

Dado que en la tarea de la muestra léxica no había disponible información morfosintáctica hemos reimplementado el algoritmo de anotación de Eric Brill. Confiamos en que cuando esta característica esté integrada en el sistema los resultados mejorarán ligeramente.

También hemos permitido como parámetro la posibilidad de filtrar los sentidos con pocas apariciones en SemCor como en la heurística anterior. Además hemos limitado la posibilidad de elección a los primeros seis sentidos de cada palabra, puesto que en término medio estos primeros seis sentidos acumulan más del 90 % de las apariciones de cada palabra. Esta heurística no ha sido incluida en los sistemas que evaluamos en la siguiente sección por los problemas que se discuten en la siguiente heurística, pero ha sido incluida aquí para facilitar la comprensión de la discusión del problema y porque se utiliza en la definición de las dos siguientes heurísticas.

- **Enriquecimiento de los vectores característicos de los sentidos**

  El problema con el filtro de relevancia es que hay muy poco solapamiento entre las definiciones de los sentidos y los contextos en términos de coocurrencia (después de eliminar las palabras de la lista de parada y calcular el idf) lo que significa que la heurística anterior, pese a su alta precisión no es capaz de desambiguar muchas palabras. Este problema se señaló por vez primera en (Wilks et al., 1990).

  Para solucionar este problema, enriquecemos los vectores característicos de los sentidos añadiendo, en cada componente, información sobre las palabras relacionadas con cada palabra de la glosa, yendo más allá de la coocurrencia directa. Esto se corresponde desde un punto de vista algebraico con multiplicar la matriz de vecindad y el vector característico. En otras palabras, si VECINDAD es la matriz de vecindad y v nuestro vector característico, el vector enriquecido sería VECINDAD $\cdot\, v$.

  Este cálculo para cada sentido puede ser muy costoso computacionalmente tanto en términos de tiempo como de espacio, de modo que si lo que nos interesa es calcular la puntuación de un sentido, sólo será necesario calcular los valores que vayan a ser utilizados. De este modo, para calcular la puntuación de un sentido nos bastaría calcular:

$$\sum_{i \in C} \sum_{j \in S} \text{VECINDAD}_{ij} \text{freq}(i, C) \text{pesado}(i, \alpha) \text{freq}(j, S) \text{idf}(i, \alpha)$$

  donde, como antes, VECINDAD es la matriz de vecindad de palabras, s es un sentido de la palabra α, que es la palabra por desambiguar, cuya glosa es S y cuyo contexto de desambiguación es C. Sólo tenemos que sumar una cantidad de términos que es el producto de la longitud de C por la longitud de S. El cálculo ahora resulta perfectamente manejable.

Una consecuencia curiosa de que la matriz de vecindad sea simétrica es que se puede probar fácilmente que el efecto de enriquecer los vectores de los sentidos es exactamente igual al de enriquecer los vectores de los contextos, por lo que no merece la pena multiplicar ambos vectores por la matriz de vecindad.

Esto debería incrementar el número de palabras desambiguadas, supuesto que apliquemos correctamente la *frecuencia inversa de documento*, puesto que ahora los vectores de sentidos ya no son dispersos como antes y el calcular el idf de cada término directamente lo anularía en la mayoría de los casos. Esto puede verse en los cuadros 5.1, 5.3 y 5.5.

- **Filtro mixto**

  También hemos incorporado como parámetro a la heurística de los sentidos enriquecidos el poder eliminar sentidos en función de una baja frecuencia relativa, como hemos hecho con las heurísticas anteriores. En caso de utilizar dicha característica llamamos a esta heurística *filtro mixto*.

- **Estrategia de soporte**

  Para aquellos casos en los que no hayamos podido desambiguar una palabra utilizando las heurísticas anteriores hemos empleado el primer sentido de WordNet como heurística *soporte*, ya que nuestro objetivo siempre ha sido maximizar la medida de recall, para lo cual es conveniente presentar resultados para todas las palabras de la tarea. Aún así las diferencias son escasas porque se utiliza en muy pocos casos, como puede comprobarse en los cuadros 5.1, 5.3 y 5.5.

5.4. Sistemas y resultados

5.4.1. Tarea de todas las palabras

La puntuación recogida en los cuadros refleja una puntuación para cada palabra de la tarea entre 0 y 100. Las heurísticas utilizadas para construir el sistema UNED–AW–U de la categoría de todas las palabras que participó en SENSEVAL-2 y los resultados obtenidos para cada una de ellas pueden verse en el cuadro 5.1. Vemos como curiosamente la precisión de las expresiones monosémicas no es del 100 %. esto se debe tanto a posibles errores en la anotación como a que la anotación permitía para cualquier palabra las etiquetas U (no asignable/desconocido) y P (nombre propio). Entre los sentidos enriquecidos y el filtro mixto hay una pugna clara entre precisión y recall que veremos repetida a lo largo de todos los resultados.

Heurística	Intentadas	Cobertura	Precisión	Recall	Puntuación
Monosémicas	451/2473	18 %	89 %	16 %	40500
Filtro Estadístico	587/2473	23 %	68 %	16 %	40100
Filtro Mixto	860/2473	34 %	38 %	13 %	33200
Sentidos Enriquecidos	531/2473	21 %	50 %	10 %	26600
Primer Sentido	27/2473	1 %	59 %	0 %	1600
Total	2456/2473	99 %	57 %	57 %	142000

Cuadro 5.1: Cascada de heuristicas no supervisadas para todas las palabras

Sistema	Intentadas	Cobertura	Precisión	Recall	Puntuación
Primer Sentido	2456/2473	99 %	60 %	59 %	148000
Sistema no supervisado	2456/2473	99 %	57 %	57 %	142000
Filtro Mixto	1898/2473	76 %	59 %	46 %	113800
Sentidos Enriquecidos	2410/2473	97 %	46 %	45 %	111400
Aleatorio	2456/2473	99 %	36 %	35 %	88489
Filtro Estadístico	1038/2473	41 %	77 %	32 %	80600
Monosémicas	451/2473	18 %	89 %	16 %	40500

Cuadro 5.2: Sistema no supervisado vs heurísticas sobre todas las palabras

Si las heurísticas individuales se hubieran utilizado como sistemas para desambiguar todas las palabras los resultados serían los mostrados en el cuadro 5.2. Es digno de mención el hecho de que el primer sentido no consigue desambiguar todas las palabras. Algunas palabras están escritas con faltas de ortografía, otras no se encuentran en WordNet.

5.4.2. Tarea de muestra léxica

Heurística	Intentadas	Cobertura	Precisión	Recall	Puntuación
Monosémicas	208/4328	4 %	58 %	2 %	12200
Filtro Estadístico	1099/4328	25 %	43 %	11 %	48100
Filtro Mixto	1941/4328	44 %	34 %	15 %	66100
Sentidos Enriquecidos	1036/4328	23 %	47 %	11 %	49200
Primer Sentido	1/4328	0 %	100 %	0 %	100
Total	4285/4328	99 %	41 %	40 %	175700

Cuadro 5.3: Heurísticas no supervisadas para la muestra léxica

Los resultados de las heurísticas correspondientes a la participación en la muestra léxica pueden verse en el cuadro 5.3. Los resultados de la detección de términos multipalabra (correspondiente a la heurística *monosémicas*) resulta claramente decepcionante y en nuestra opinión achacable a posibles errores de anotación en la anotación manual. Una vez más, parece que un intercambio en el orden de la cascada entre los sentidos enriquecidos y el filtro mixto sería beneficioso. La comparación del rendimiento de las heurísticas como sistemas individuales se encuentra en el cuadro 5.4. El primer sentido vuelve a no conseguir desambiguar todas las palabras. Esto es debido a un error de la organización en la anotación de la palabra *colourless* y su confusión entre la grafía británica y la americana.

5.4.3. Evaluación sobre SemCor

Los resultados de la evaluación sobre SemCor corresponden a los cuadros 5.5 (el sistema en sí) y 5.6 (las heurísticas tomadas como sistemas individuales comparadas con el sistema).

Los resultados muestran que el sistema obtiene muy buenos resultados, el problema es que estos resultados no se extrapolan a otras colecciones por el hecho de que la

Sistema	Intentadas	Cobertura	Precisión	Recall	Puntuación
Primer sentido	4285/4328	99 %	43 %	42 %	185400
Sistema no supervisado	4285/4328	99 %	41 %	40 %	175700
Filtro Mixto	3248/4328	75 %	38 %	29 %	126400
Sentidos Enriquecidos	4284/4328	98 %	35 %	34 %	150800
Aleatorio	4285/4328	99 %	18 %	18 %	79715
Filtro Estadístico	1307/4328	30 %	46 %	13 %	60300
Monosémicas	208/4328	4 %	58 %	2 %	12200

Cuadro 5.4: Sistema no supervisado vs heurísticas sobre la muestra léxica

Heurística	Intentadas	Cobertura	Precisión	Recall	Puntuación
Monosémicas	40240/192639	20 %	100 %	20 %	4024000
Filtro Estadístico	55028/192639	28 %	83 %	23 %	4614800
Filtro Mixto	63770/192639	33 %	42 %	13 %	2686800
Sentidos Enriquecidos	30181/192639	15 %	46 %	7 %	1400500
Primer Sentido	3420/192639	1 %	61 %	1 %	211300
Total	192639/192639	100 %	67 %	67 %	12937400

Cuadro 5.5: Cascada de heurísticas no supervisadas para SemCor

estimación de las frecuencias relativas de aparición de los sentidos aquí es exacta, y en otras colecciones solo resulta orientativa.

Sistema	Intentadas	Cobertura	Precisión	Recall	Puntuación
Primer Sentido	192639/192639	100 %	75 %	75 %	14524900
Sistema no supervisado	192639/192639	100 %	67 %	67 %	12937400
Filtro Mixto	159038/192639	82 %	71 %	58 %	11325600
Sentidos Enriquecidos	186980/192639	97 %	50 %	49 %	9469600
Filtro Estadístico	95268/192639	49 %	90 %	44 %	8638800
Aleatorio	192639/192639	100 %	40 %	40 %	7731393
Monosémicas	40240/192639	20 %	100 %	20 %	4024000

Cuadro 5.6: Sistema no supervisado vs heurísticas sobre SemCor

5.4.4. Efecto de la medida de asociación

Los contrastes de hipótesis de dependencia estadística suelen estar basados en comparar lo que podría obtenerse si se cumple la hipótesis de independencia (hipótesis nula) con lo que se observa. Si la hipótesis nula resulta improbable, i.e., si su probabilidad es menor que un nivel de confianza α (habitualmente 0.05 o 0.005) entonces se rechaza la hipótesis nula y se considera que las variables aleatorias (las frecuencias de las dos palabras en cuestión) son dependientes. En otro caso los resultados no son concluyentes. La forma de calcular esta probabilidad consiste en crear otra variable aleatoria cuya distribución asintótica se conoce por medio de algún teorema. De este modo es posible calcular los valores de la variable aleatoria y comprobar si son compatibles con los esperados según las hipótesis planteadas.

La medida χ^2 es un contraste de hipótesis que sirve para determinar la dependencia estadística de dos sucesos. Por esa razón también puede utilizarse como medida de asociación. Tiene la ventaja sobre otros contrastes de hipótesis de dependencia que no presupone que los datos se distribuyan según una distribución normal. En este caso nuestra hipótesis nula sería que la coaparición de a y b en un mismo contexto es independiente. Para comprobar esa hipótesis podemos observar el cuadro 5.7, en el que compararemos la dependencia entre *judge* y *jury*.

Si observamos la tabla de arriba como una matriz O_{ij}, la variable aleatoria definida como:

	Aparece *jury*	No aparece *jury*
Aparece *judge*	1032	3971
No aparece *judge*	30307	92742864

Cuadro 5.7: Tabla de contingencias para *jury* y *judge*

$$X^2 = \sum_{i=1}^{2}\sum_{j=1}^{2}\frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

se distribuye asintóticamente como una χ^2 de modo que si fijamos un nivel de confianza de α =0.05, cualquier valor de X mayor de 3.84 nos lleva a descartar la hipótesis nula e inferir que las apariciones de dos palabras son estadísticamente dependientes. Hemos usado el valor de X^2 como medida de asociación entre ambas palabras. Por supuesto, es conveniente fijar un umbral bajo el cual la medida sea considerada como cero.

En el caso de *judge* y *jury*, X vale 628160, de modo que parece haber una clara dependencia estadística. Como ejemplo de palabras estadísticamente independientes podríamos mencionar la pareja (*banana*, *chip*), con un valor de X de 0.015 no aparecen juntas jamás en el corpus. La información mutua le asignaría a esta pareja un valor de cero.

La medida de verosimilitud de (Dunning, 1993) promete ser más apropiada como contraste de hipótesis sobre la dependencia estadística de la coocurrencia de palabras que el χ^2 según las investigaciones empíricas de Dunning. Dunning calculó bigramas con fuerte dependencia y encontró que su fórmula basada en una distribución binomial asociaba unos bigramas más interesantes a priori que aquellos discriminados con más confianza por el contraste χ^2 de Pearson.

Una vez más, consideramos las palabras a y b en una serie de N contextos y queremos calcular su dependencia estadística. Supondremos que el resultado de encontrar o no las palabras en un contexto es independiente del contexto (en la linea de la mejor tradición empirista), con el objeto de modelar la aparición de cada una de las palabras con una distribución binomial.

Nuestra hipótesis nula H_0, será:

$$H_0 : P(b|a) = p = P(b|\neg a)$$

Esto es, que a y b son independientes. La otra hipótesis H_1 será :

$$H_1 : P(b|a) = p_1 \gg p_2 = P(b|\neg a)$$

La aparición de una palabra a menudo tiene un impacto sobre la aparición o no de otras, pero la no aparición de una palabra raramente influye sobre la aparición o no de otras.

Adaptando las fórmulas de (Manning and Schütze, 1999) para bigramas a nuestro problema de coocurrencias en contextos de longitud fija, si utilizamos de nuevo los EMV's correspondientes para estimar las probabilidades, tendríamos que:

$$p = \frac{f_b}{N}, \quad p_1 = \frac{f_{ab}}{f_b}, \quad p_2 = \frac{f_b - f_{ab}}{N - f_a}$$

Suponiendo una distribución binomial:

$$b(k, n, x) = \binom{n}{k} x^k (1 - k)^{(n-k)}$$

La probabilidad de que las frecuencias observadas f_a, f_b y f_{ab} corresponda a la hipótesis H_0 será entonces:

$$L(H_0) = b(f_{ab}, f_a, p) \cdot b(f_b - f_{ab}, N - f_a, p)$$

La probabilidad de la hipótesis H_1 será por tanto:

$$L(H_1) = b(f_{ab}, f_a, p_1) \cdot b(f_b - f_{ab}, N - f_a, p_2)$$

La verosimilitud logarítmica se define entonces como:

$$\begin{aligned}
\log \lambda &= \log \frac{L(H_0)}{L(H_1)} \\
&= \log \frac{b(f_{ab}, f_a, p) \cdot b(f_b - f_{ab}, N - f_a, p)}{b(f_{ab}, f_a, p_1) \cdot b(f_b - f_{ab}, N - f_a, p_2)} \\
&= \log L(f_{ab}, f_a, p) + \log L(f_b - f_{ab}, N - f_a, p) \\
&\quad - \log L(f_{ab}, f_a, p_1) - \log L(f_b - f_{ab}, N - f_a, p_2)
\end{aligned}$$

Donde $L(k, n, x) = x^k(1-x)^{n-k}$. Entonces $-2\log\lambda$, que será nuestra *medida de Dunning*, se distribuye asintóticamente como una χ^2.

Probamos a sustituir la medida de asociación utilizada para calcular vectores característicos enriquecidos. Además de la fórmula similar a la información mutua, empleamos la χ^2 y la fórmula sugerida por (Dunning, 1993). Los resultados de esta comparación pueden verse en el cuadro 5.8. En el apéndice C, pueden observarse las peculiaridades de cada medida con respecto a la mayoría de las palabras de la muestra léxica de SENSEVAL-2.

Es evidente que la medida de información mutua da mejores resultados. Tal vez sea debido a que se ajusta de manera natural al modelo aditivo de puntuación que hemos establecido, mientras que la interpretación de la χ^2 y de la medida de Dunning proporcionan cantidades de órdenes de magnitud tan elevada (en la medida binomial de Dunning interviene el factorial de 92 millones, el número de contextos después de eliminar las palabras de parada de los documentos) que probablemente un modelo aditivo no sea adecuado para apreciar las diferencias de asociación.

Fórmula	Intentadas	Cobertura	Precisión	Recall	Puntuación
Información mutua	1706/4328	39 %	36 %	14 %	62100
χ^2	1700/4328	39 %	28 %	11 %	48800
Dunning	1683/4328	38 %	28 %	11 %	48800

Cuadro 5.8: Comparación de medidas de asociación

5.5. Discusión y conclusiones

Los resultados obtenidos apoyan la afirmación de (Wilks et al., 1990) de que la información de coocurrencia es una fuente de evidencia para la DSP. Incorporar información de tipo sintáctico sería también muy deseable puesto que su influencia de cara a la desambiguación se estima como importante.

Pensamos que también merecería el esfuerzo tratar de usar algún tipo de técnica estadística de suavizado para la matriz de vecindad, tal como se hace por ejemplo en (Gale et al., 1993), puesto que hemos experimentado el mismo tipo de problemas al mezclar las frecuencias de aparición de palabras con frecuencias de aparición muy desiguales.

Teníamos bastante confianza en que el filtro de vecindad daría buenos resultados, puesto que ya lo habíamos evaluado contra los datos de SENSEVAL-1 y SemCor. Dada su escasa cobertura decidimos no utilizarlo y en su lugar mejorar la cobertura enriqueciendo los vectores de los sentidos a través de la multiplicación por la matriz, con la esperanza de no perder demasiada precisión, al estilo de (Yarowsky, 1992; Schütze and Pedersen, 1995). Estamos relativamente satisfechos con los resultados, aunque pensamos que la incorporación de la información de coocurrencia *de orden superior* puede mejorarse. Es interesante resaltar que la heurística de los sentidos enriquecidos obtiene a veces mejor precisión que la del filtro mixto. Esto es debido al hecho de que las palabras altamente sesgadas en su distribución de sentidos son generalmente más polisémicas. A pesar de ello, intercambiar de orden estas dos heurísticas hace que los resultados empeoren, como puede comprobarse en el cuadro 5.2.

Por lo que respecta a la puntuación en relación con los otros participantes en SENSEVAL-2, el sistema presentado para la tarea de muestra léxica no supervisado obtuvo el recall más alto entre los sistemas no supervisados. En la tarea de todas las palabras, el sistema UNED-AW-U2 también obtuvo el recall más alto entre los sistemas no supervisados como puede comprobarse en la web de SENSEVAL[5] y fue el cuarto en términos absolutos (sistemas supervisados y no supervisados). Este sistema es una ligera modificación del UNED-AW-U. Los detalles de estas diferencias pueden encontrarse en (Fernández-Amorós et al., 2001b).

La calidad de la desambiguación parece ser muy sensible a la información utilizada puesto que multiplicar por la matriz de vecindad las caracterizaciones de los sentidos basadas únicamente en las glosas produce unos resultados aceptables mientras que hacer lo mismo con la información de las glosas junto con la de los ejemplos de entrenamiento (presuntamente menos informativos que las glosas) hace que la precisión se degrade considerablemente.

Por lo que respecta a la medida de asociación, a pesar de su sencillez, la basada en información mutua es la que mejores resultados ha proporcionado. Además tiene un coste computacional muy razonable, sobre todo si lo comparamos con el coste de calcular la asociación de Dunning. Desde un punto de vista teórico tanto la χ^2 como la de Dunning ofrecían la solución a los problemas de dispersión de datos y errores de estimación de las coocurrencias con baja frecuencia. En la práctica no ha sido así.

[5] www.senseval.org

Capítulo 6

Otras fuentes de información

6.1. Información extraída de la Web

En esta sección exploramos la incidencia sobre la desambiguación de datos de entrenamiento no supervisado extraídos automáticamente de la World Wide Web. Utilizaremos dichos datos para enriquecer las descripciones de los sentidos, con el objeto de dilucidar si se trata de una información que pueda contribuir a la mejora de la de la DSP.

Los datos, como se describe en (Santamaría et al., 2001; Santamaría et al., 2003) son extraídos de las categorías del Open Directory Project (ODP)[1]. Se realiza una comparación entre las palabras descriptivas de cada sentido (glosas de WordNet en este caso) y las palabras del camino de un directorio en el ODP. En la evaluación manual realizada por los autores, el alto porcentaje de acierto obtenido en la asignación de directorios a sentidos hizo que se planteara la posibilidad de utilizar documentos de los directorios en cuestión como ejemplos de entrenamiento para realizar DSP. En (Santamaría et al., 2003) se describe un experimento en el que se utilizan dichos datos con el algoritmo descrito en (Pedersen, 2001) y se comparan los resultados con entrenar el mismo algoritmo con los datos oficiales de SENSEVAL-2. Los resultados indican que la calidad de la información extraída es comparable aunque el volumen de datos de entrenamiento obtenidos del ODP no es muy grande (unos 120KB de datos en texto plano)

[1] http://dmoz.org

Nuestra aproximación aquí será similar, utilizaremos los datos de entrenamiento no supervisado obtenidos por (Santamaría et al., 2001; Santamaría et al., 2003) como información complementaria a la caracterización de los sentidos de WordNet presente en las glosas, con el objeto de determinar si esta información puede ser beneficiosa en la tarea de DSP.

Finalmente compararemos las dos aproximaciones mediante una evaluación sobre las palabras del SENSEVAL-2 para las que ha sido posible extraer información.

6.1.1. Evaluación y resultados

Hemos decidido comparar la utilidad de los datos desde una perspectiva no supervisada sencilla y clara. Por ello, se ha empleado como heurística de desambiguación el considerar como sentido elegido aquel que tenga más palabras en común entre su caracterización y el contexto. Es una heurística cuya precisión es un buen indicador de lo que puede esperarse de una caracterización de sentidos en términos de desambiguación no supervisada, de hecho en nuestro caso es una buena estimación de la cota superior de la precisión que se puede alcanzar con otras heurísticas, aunque su cobertura es bastante limitada. Las caracterizaciones de sentidos que compararemos entre si son las siguientes:

Web Contiene los datos recogidos de la web.

Wn Contiene los datos de las glosas de WordNet.

Training Los datos de entrenamiento de la muestra léxica para SENSEVAL-2.

Wn_web Los datos combinados de las dos primeras caracterizaciones, ponderadas al mismo peso.

Wn_2hiper La caracterización de cada sentido contiene las palabras de su glosa más las de las glosas los dos primeros hiperónimos del *synset* correspondiente.

Wn_Training Los datos de las glosas de WordNet más los datos de entrenamiento proporcionados por la organización del SENSEVAL-2 para la tarea de la muestra léxica.

Wn_2hiper_Web Los datos de WordNet, los dos primeros hiperónimos y la Web, tras calcular la media ponderada.

Wn_Web_Training Los datos de WordNet, la Web y los datos de entrenamiento ponderados.

La evaluación se realiza solamente sobre las diez palabras de la muestra léxica para las cuales existen datos extraídos de la Web para al menos dos sentidos diferentes. Los resultados pueden verse en el cuadro 6.1. También en la figura 6.1. Hemos incluido también como puntos de comparación las heurísticas aleatoria (con el nombre de RND) y la del primer sentido (con el nombre FIRST).

Caracterización	Intentadas	Cobertura	Precision	Recall	Puntuación
Web	369/672	54 %	41 %	22 %	15200
WN	234/672	34 %	50 %	17 %	11750
Training	666/672	99 %	27 %	26 %	18100
WN-Web	448/672	66 %	42 %	28 %	19000
WN-2hiper	345/672	51 %	46 %	24 %	16200
WN-2hiper-Web	498/672	74 %	39 %	29 %	19500
WN-Training	666/672	99 %	30 %	30 %	20200
WN-Web-Training	666/672	99 %	29 %	29 %	19900
RND	672/672	100 %	21 %	21 %	14709
FIRST	672/672	100 %	38 %	38 %	26200

Cuadro 6.1: Comparativa datos web vs. otros

Dado que que los datos extraídos de la web obtienen mejores resultados que utilizar las glosas de WordNet en términos de recall, hemos decidido que sería interesante analizar en detalle qué ocurre en las diez palabras en cuestión, los resultados pueden ver en los cuadros 6.2 y 6.3.

Los resultados parecen indicar que los ejemplos de entrenamiento extraídos de forma automática de la Web son mejores en términos de precisión que los ejemplos anotados a mano proporcionados por los organizadores del SENSEVAL-2. Los resultados se circunscriben a diez palabras para las que había datos de entrenamiento de la Web para más de un sentido. Son pocas palabras y pocos contextos como para extraer conclusiones generales. En cualquier caso la técnica resulta muy prometedora aunque está limitada a los sentidos susceptibles de asociarse con un directorio web, es decir, han de tener un cierto grado de *especificidad de dominio*.

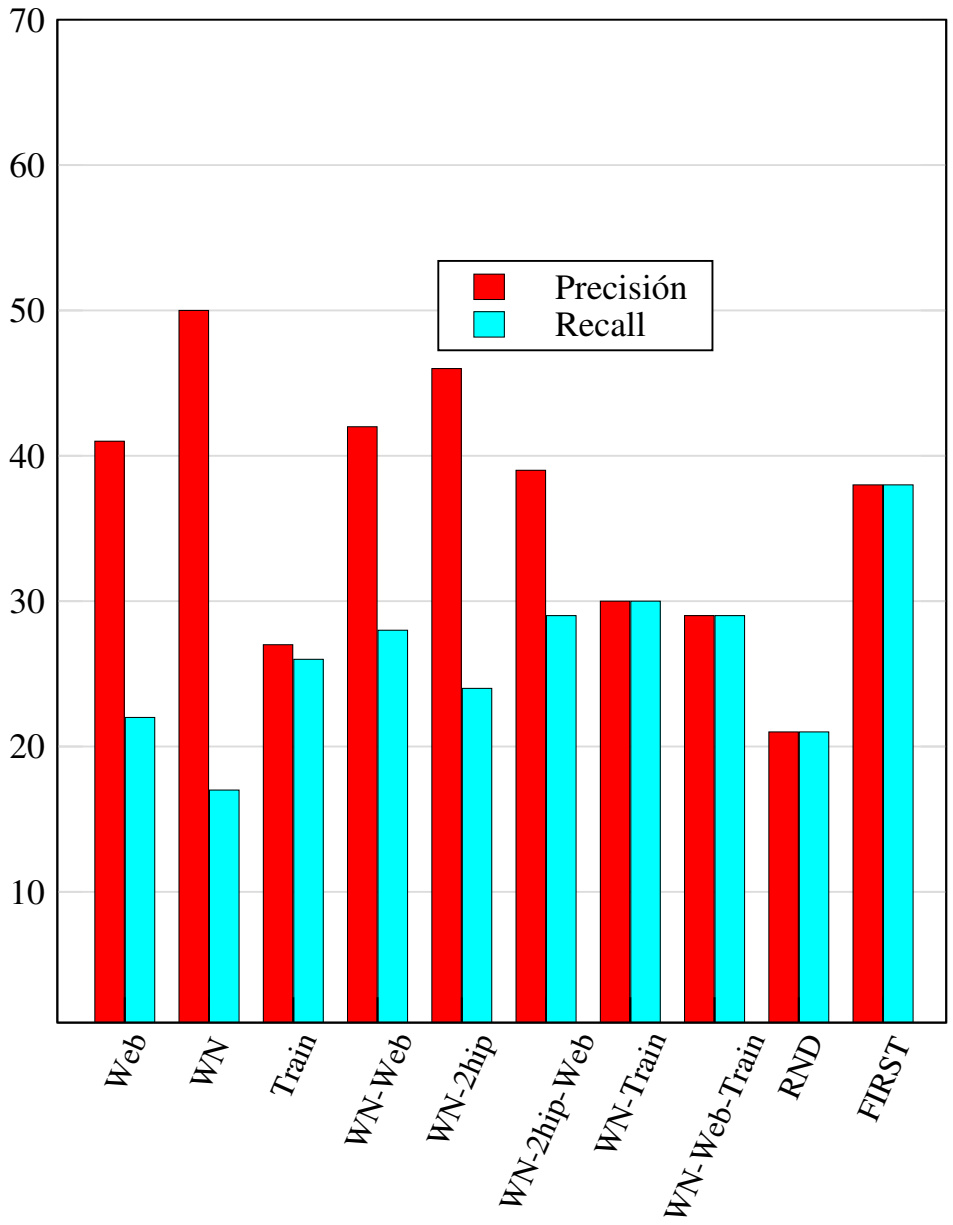

Figura 6.1: Caracterizaciones de sentidos comparadas mediante intersección, junto con las heurísticas de comparación aleatoria y del primer sentido

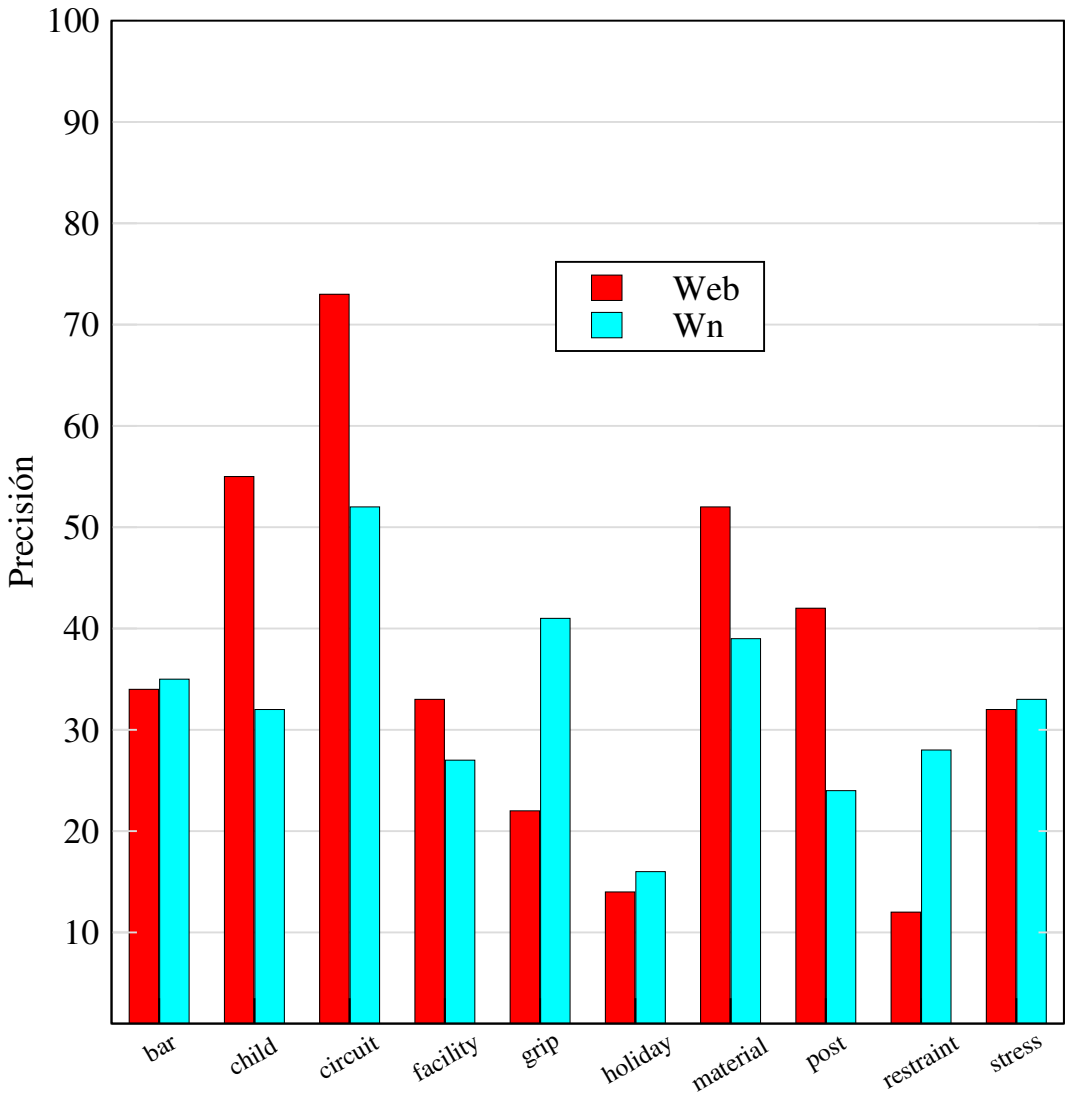

Figura 6.2: Datos Web vs. WordNet palabra por palabra

Palabra	Intentadas	Cobertura	Precision	Recall	Score
bar	34/151	22 %	47 %	10 %	1600
child	55/64	85 %	47 %	40 %	2600
circuit	73/85	85 %	56 %	48 %	4100
facility	33/58	56 %	42 %	24 %	1400
grip	22/51	43 %	9 %	3 %	200
holiday	14/31	45 %	71 %	32 %	1000
material	52/69	75 %	36 %	27 %	1900
post	42/79	53 %	21 %	11 %	900
restraint	12/45	26 %	50 %	13 %	600
stress	32/39	82 %	28 %	23 %	900
Total	369/672	54 %	41 %	22 %	15200

Cuadro 6.2: Datos web, por palabras

Palabra	Intentadas	Cobertura	Precision	Recall	Score
bar	54/151	35 %	42 %	15 %	2300
child	21/64	32 %	33 %	10 %	700
circuit	45/85	52 %	53 %	28 %	2400
facility	16/58	27 %	56 %	15 %	900
grip	21/51	41 %	57 %	23 %	1200
holiday	5/31	16 %	80 %	12 %	400
material	27/69	39 %	59 %	23 %	1600
post	19/79	24 %	44 %	10 %	850
restraint	13/45	28 %	61 %	17 %	800
stress	13/39	33 %	46 %	15 %	600
Total	234/672	34 %	50 %	17 %	11750

Cuadro 6.3: Datos WordNet, por palabras

6.2. Sintagmas alineados

En esta sección incorporaremos a nuestro sistema de DSP una nueva fuente de información: la información perteneciente a sintagmas extraídos de corpora comparables bilingües.

La idea básica no difiere mucho de la aproximación tomada por (Dagan et al., 1991; Dagan and Itai, 1994), aunque aquí la unidad fundamental a partir de la cual se extraerá la información no será la palabra, sino el sintagma, una expresión mucho menos ambigua como conjunto, y que permite descartar sentidos de las palabras individuales via traducción con un diccionario bilingüe.

Los sintagmas bilingües utilizados se han tomado de (Peñas, 2002) y la alineación entre ellos de (López Ostenero et al., 2002; López Ostenero, 2002). El algoritmo de alineación utilizado tiene la ventaja de que no es necesario disponer de grandes corpora paralelos (i.e. traducciones de los mismos textos), sino que basta con que los textos de los corpora sean comparables (en este caso lo son por tratarse de noticias de agencias de la misma época).

Dado que la precisión del reconocimiento y el alineamiento de los sintagmas es alta para los sintagmas altamente frecuentes, se ha evaluado esta información en forma de filtro, es decir, aquellos sentidos de palabras pertenecientes a sintagmas alineados que no tengan una correspondencia en las traducciones del segundo idioma son descartados, y no son presentados a las heurísticas de desambiguación.

Un ejemplo de esta técnica sería el siguiente [2]:

Queremos desambiguar la palabra *issue*. La palabra issue se puede traducir en español como: asunto, tema, número, emisión, expedición, descendencia, publicar, emitir, expedir, dar y promulgar. En este punto detectamos que el contexto de la palabra nos indica que forma parte del sintagma *abortion issue*. Este sintagma se ha alineado mediante el algoritmo descrito en (López Ostenero, 2002) con el sintagma español *tema del aborto*. Si estuviéramos haciendo traducción automática nos quedaríamos satisfechos con esta traducción, sin embargo, en el marco de la DSP, quisiéramos descartar los sentidos de issue que no correspondan a la traducción de tema. Por desgracia, la estructura de WordNet no permite conocer con facilidad esta información.

La solución que hemos desarrollado consiste en utilizar los índices interlingua (ILI)

[2] Adaptado de (López Ostenero et al., 2002)

de EuroWordNet. Estos índices relacionan los conceptos (synsets) de un idioma de EuroWordNet con otro, de manera que nos serviría, en el ejemplo presentado para encontrar los synsets asociados a *issue* en español y ver en cuales de ellos se encuentra la palabra tema. Un pequeño inconveniente de esta aproximación es que la taxonomía de EuroWordNet está ligada en el inglés a los conceptos de WordNet-1.5 (existe también una versión de SemCor anotada con los sentidos de WordNet-1.5 pero en caso de éxito querríamos aplicar la técnica a las colecciones de SENSEVAL-2, anotadas con WordNet-1.7). Para solucionar esta dificultad utilizaremos una conversiones de la versión 1.5 a la 1.6 y de la 1.6 a la 1.7 desarrolladas en (Daudé et al., 2000; Daudé et al., 2001).

Cuando queremos desambiguar una palabra, primero vamos a mirar su contexto para determinar si pertenece a uno de los sintagmas alineados de nuestra base de conocimiento. Para esto utilizaremos el mismo algoritmo de backtracking multinivel que empleamos para detectar los términos multipalabra. En caso afirmativo tomamos la lista de sintagmas alineados en español (ordenada por frecuencia) y recorremos estos sintagmas en español. Para las palabras de los sintagmas en español buscamos sus sentidos y, utilizando la asociación, sus synsets correspondientes en inglés. Si alguno de estos synsets contiene a la palabra original a desambiguar entonces consideramos este sentido como candidato y lo añadimos al conjunto de sentidos a considerar.

Una vez realizado este proceso para todas la alineaciones, habremos descartado posiblemente algunos sentidos de la palabra. De este modo, utilizamos este tipo de información como un filtro, más que como una heurística de desambiguación propiamente dicha, puesto que después de filtrar sentidos podemos aplicar las otras heurísticas de que disponemos.

La forma de proceder ha consistido en reanotar automáticamente la versión de SemCor que habíamos traducido a XML para los experimentos de los capítulos anteriores. Hemos añadido para las palabras correspondientes un atributo, *phrase*, que indica el sintagma (phrase) que se ha detectado y otro, *alineación*, que nos proporciona una lista de sentidos candidatos junto con la frecuencia de alineación del sintagma alineado en español que más veces se ha alineado con el atributo *phrase*. Esta información de frecuencia es importante, porque se supone que la fiabilidad de la alineación depende directamente de su frecuencia.

De este modo, un fragmento de documento de SemCor que anteriormente presentaba este aspecto:

```
<wf cmd="done" pos="NN" lemma="number" wnsn="2" lexsn="1:23:00::">number</wf>
```

```
<wf cmd="ignore" pos="IN">of</wf>
<wf cmd="done" pos="NN" lemma="voter" wnsn="1" lexsn="1:18:00::">voters</wf>
```

Ahora presenta este otro:

```
<wf alineacion="number%1:07:00:: 51 number%1:10:00:: 51 number%1:10:01:: 51
number%1:10:02:: 51 number%1:10:03:: 51 number%1:10:04:: 51 number%1:10:05:: 51
number%1:23:00:: 51" cmd="done" lemma="number" lexsn="1:23:00::"
phrase="number of voters" pos="NN" wnsn="2">number</wf>
<wf cmd="ignore" pos="IN">of</wf>
<wf cmd="done" lemma="voter" lexsn="1:18:00::" pos="NN" wnsn="1">voters</wf>
```

Hemos reetiquetado toda la colección SemCor con el parser Xerces/C++ de la Apache Foundation y el interfaz DOM, para construir el árbol sintáctico y poder detectar los sintagmas y anotar los sentidos de las alineaciones y la frecuencia de estas.

6.2.1. Evaluación y resultados

Hemos decidido evaluar esta técnica contra la colección SemCor, puesto que se trata de una colección de un tamaño suficiente como para poder extraer conclusiones interesantes. En el laborioso proceso de la reetiquetación, de las 192480 palabras susceptibles de ser desambiguadas en las colecciones brown-1 y brown-2 del SemCor, se han detectado 10787 sintagmas del inglés, esto es tenemos alineaciones para el 5.6 % de las palabras. Estos sintagmas se han alineado con sintagmas en español en 5290 ocasiones, de modo que hemos filtrado sentidos para este número de palabras (el 2.74 %). Entre éstas, el sentido correcto se ha conservado a través del proceso de filtrado en 3922 casos. Es decir, el proceso de filtrado tiene una precisión potencial del 74.33 %.

Un ejemplo en el que el algoritmo no funcionó como se esperaba sería en la desambiguación de *friend*. Encontramos como sintagma en inglés *friend_of_mine* y la alineación en español fue *conocido_de_las_minas*. Es evidente que la técnica de las alineaciones todavía necesita progresar, sin embargo, la alineación entre estos dos sintagmas sólo se producía una vez de manera que sentimos la curiosidad de comprobar el grado de correlación entre la frecuencia de las alineaciones y la precisión potencial susceptible de ser obtenida. Para ello hemos repetido el experimento anterior utilizando un umbral, para considerar sólo los sentidos para los que el número de alineaciones del sintagma más alineado supere el umbral. Los resultados pueden verse en la figura 6.3.

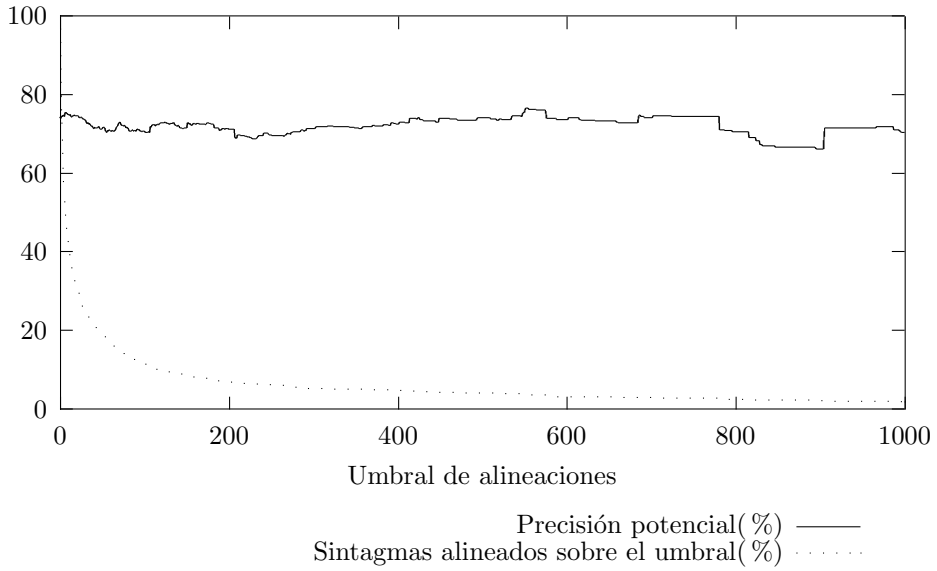

Figura 6.3: Relación entre el umbral, la cobertura y la precisión potencial

Los resultados son bastante sorprendentes: No parece haber una relación clara entre el aumento del umbral y el aumento de la precisión potencial. Hasta un valor de umbral de 3000 apenas si se aprecian diferencias. A partir de ahí el número de sintagmas alineado es tan bajo (con un umbral de 3000 el número de sintagmas con alineaciones en SemCor es de 38) que la información no tiene mucha utilidad. La precisión potencial de 100 % que se produce a partir de un umbral de 8713 hasta 8836 corresponde íntegramente a alineaciones del sintagma *year_old*, que además apoyan por igual los cuatro sentidos de *year*, es decir, que en ese punto la información de los sintagmas alineados es completamente irrelevante.

La conclusión sobre este tipo de información a día de hoy es clara, pese a ser un filtro interesante a priori en términos de precisión, la bajísima cobertura del método no permite un nivel de desambiguación mínimamente apreciable. Tal vez un aumento del tamaño de los corpora y una mejora del diccionario de alineamiento permitan aprovechar este tipo de información para la DSP en el futuro.

Por otra parte, la alineación de sintagmas entre idiomas de procedencia más heterogénea probablemente mejoraría la situación. En el caso del inglés y el español, se pueden esperar casos patológicos como que muchos de los sintagmas que contienen *art* en inglés se alineen con otros que contengan *arte* en español, cosa que no parece muy productiva a la hora de filtrar sentidos: Casi todos los sentidos de *art* se pueden

traducir como arte.

Parte III

Sistema de DSP final y conclusiones

Capítulo 7

El sistema final de anotación semántica

7.1. Introducción

De cara a realizar un sistema final a partir de los experimentos realizados, es conveniente hacer una pequeña revisión de los métodos de combinación de heurísticas que se han aplicado en la literatura sobre el tema. Básicamente, se pueden distinguir dos grandes corrientes que describiremos a continuación.

7.1.1. Combinación informada

En esta corriente, cada heurística, además de proporcionar unos resultados de desambiguación, produce también un resultado secundario que permite asociar un grado de confianza a la solución propuesta.

En el caso de (McRoy, 1992) este criterio era el de especificidad. Según este criterio para cada heurística y sentido a desambiguar, se asigna un valor en entre -10 y 10 según la especificidad de cada heurística. Se suman estás cantidades para cada sentido de la palabra a desambiguar y el que mayor puntuación obtenga es el elegido. Desgraciadamente no se proporciona una evaluación cuantitativa, ni de las heurísticas por separado, ni de su combinación.

Otro caso interesante es el de (Hearst, 1991). El criterio decisivo en estos experimentos fue llamado *evidencia comparativa*. Se calculaba para sentido y dependía del número de *características de evidencia* halladas para cada sentido. Estas característica de evidencia se determinaban a priori y estaban estructuradas en forma de reglas.

A pesar de todo, posiblemente el método con más éxito de combinación de heurísticas para anotación semántica es el basado en combinaciones lineales de los resultados. Es decir, a cada heurística o desambiguador parcial se le asigna un escalar, de manera que para una palabra y un sentido concreto se calcula su puntuación realizando la combinación lineal correspondiente. Hay que decir que pese a carecer de una justificación teórica hasta ahora en el caso de la desambiguación, es un método mediante el cual pueden conseguirse modestas mejoras respecto a los desambiguadores parciales.

Esta técnica se puede utilizar tanto desde un punto de vista supervisado como no supervisado. En el caso supervisado la idea está clara, se determinan cuáles son los coeficientes óptimos para cada desambiguador y por definición el resultado final tiene que ser mejor que el del mejor desambiguador parcial. Por este motivo se suelen utilizar algoritmos de cálculo numérico de tipo gradiente máximo para determinar dichos coeficientes. En este grupo se podría englobar el trabajo de (Ilhan et al., 2001; Alshawi and Carter, 1994; Harley and Glennon, 1997; Pedersen and Bruce, 1997b; Montoyo et al., 2002).

7.1.2. Combinación ciega

En otros casos, como (Chodorow et al., 2000), la propia naturaleza del algoritmo, probabilístico bayesiano en este caso facilitaba la combinación. Se consideraba cada una de las heurísticas como generadoras de resultados basados en evidencia de distinta naturaleza y después se consideraban estas heurísticas estadísticamente independientes, con lo cual la fórmula para el cálculo final resultaba inmediata.

Otro método habitual es el de considerar que hay heurísticas inherentemente mejores que otras , de manera que se aplican *en cascada*, pasando, para cada palabra objetivo, de las presuntamente mejores a las peores, hasta que una heurística sea capaz de devolver un resultado. Sistemas de este tipo podemos encontrarlos en (Mihalcea and Moldovan, 2000a; Fernández-Amorós et al., 2001b).

Otros métodos comunmente utilizados y que no precisan más información que aquella que proporcionan desambiguadores parciales sobre sus preferencias son los basados en votación y en combinación lineal. Un ejemplo de sistema que combina heurísticas

según un sistema de votación se puede encontrar en (Rigau et al., 1997); hay diferentes heurísticas para desambiguar entradas del diccionario respecto del genérico. La forma de hacerlo es tipo votación/combinación lineal con coeficientes positivos. Cada sentido y heurística obtiene una puntuación entre 0 y 1, luego se divide por el número de heurísticas. Otra estrategia similar es la de (Kwong, 2001), dadas varias heurísticas, se suman las puntuaciones para cada sentido de cada palabra y después se divide por el máximo valor parcial para obtener la puntuación final de cada sentido.

En (Stevenson et al., 1998; Wilks and Stevenson, 1997a) se presenta un sistema que tiene dos heurísticas básicas, simulated annealing y códigos de temáticos (subject codes). La forma de combinarlas es la siguiente: Si coinciden en escoger el mismo sentido se devuelve ese, si no, se toma el primer sentido del diccionario. También (Ilhan et al., 2001) utiliza un sistema de votación con elección por mayoría.

Otro enfoque de combinación consiste en aplicar heurísticas diferentes según la categoría gramatical y después simplemente unir los resultados. En (Montoyo and Suárez, 2001; Suárez and Montoyo, 2001) se utilizaron dos algoritmos distintos, uno no supervisado de marcas de especificidad aplicado a los nombres y otro supervisado de *Máxima entropía* aplicado a verbos y adjetivos (se aplicó a la muestra léxica de SENSEVAL-2, en la que no había adverbios para desambiguar).

7.2. El sistema final

El sistema propuesto tiene como objetivo combinar los distintos tipos de información que se han estudiado en los capítulos precedentes. De estos tipos de información dos de ellos serán excluidos. La información extraída de la web plantea el problema de que en una evaluación seria y exhaustiva la escasez de los datos obtenidos sería un factor más a considerar y las diferencias serían prácticamente despreciables.

Por lo que respecta a la desambiguación basada en sintagmas alineados, la escasa frecuencia de las alineaciones nos ha llevado a descartarlo asimismo del sistema final.

Esto nos deja en una situación en la que las posibilidades de combinación no resultan demasiado grandes. El algoritmo basado en relaciones conceptuales del capítulo 3 sólo resulta aplicable a nombres, de manera que para las otras partes del discurso la opción está en ordenar adecuadamente la cascada de heurísticas. El caso de los nombres resulta más complejo.

Un primer experimento interesante consistiría en saber cuáles son los grados de correlación de las heurísticas de los capítulos 3 y 4 entre ellas y también con el primer sentido de WordNet. La colección elegida para calcular esta correlación ha sido la tarea de todas las palabras del SENSEVAL-2. La colección SemCor, pese a su mayor longitud padece un defecto considerable: No se conoce cual es el grado de error en la anotación manual. Este punto es esencial, puesto que si bien los autores estiman este error *en torno a un 10 %*, no sabemos nada sobre la forma de realizar dicha estimación. Este dato es fundamental, puesto que el primer sentido de WordNet para cada palabra está basado en el sentido más frecuente según los anotadores humanos en SemCor. Como se ha señalado en otros lugares, SemCor es una parte del Brown Corpus, que nació con el objeto de ser una muestra del inglés americano de la época, de todos los campos. En nuestra opinión no se debió lograr una independencia completa del dominio del discurso, tarea tal vez imposible. Si se hubiese alcanzado dicha independencia del dominio, la degradación de la medida del primer sentido en otras colecciones no debería ser tan marcada.

Lo cierto es que la posibilidad de que la tasa de error en SemCor sea más elevada de lo previsto hace que cualquier comparación de una heurística no supervisada con la del primer sentido resulte bastante desigual. De hecho cualquier heurística queda mal parada con respecto al primer sentido en el ámbito de SemCor, con la excepción de las supervisadas (y no todas) que utilizan la propia información del SemCor.

Una prueba del problema del posible error de anotación o cambio de dominio puede hallarse comprobando que en SemCor el recall alcanzado por la heurística del primer sentido es del 75 %, mientras que en la tarea de todas las palabras se reduce a un 59 %. En el caso de la muestra léxica, el primer sentido se queda en un recall de 39 %. Es evidente que la tarea de la muestra léxica es más difícil puesto que todas las palabras que la integran son altamente polisémicas, mientras que en SemCor y en la tarea de todas las palabras hay palabras de diversos grados de polisemia, incluso muchas monosémicas. Lo que esto viene a probar es que hay una gran diferencia entre el concepto de sentido más frecuente, que no puede considerarse una heurística ya que por definición, para conocerlo hay que tener el texto desambiguado, sino más bien un punto de comparación, y el de primer sentido, que sí puede considerarse una heurística, aunque en el caso de SemCor ambos conceptos coinciden.

Los problemas que se plantean al utilizar la muestra léxica como colección de prueba para evaluar un sistema están relacionados también con la metodología de la anotación manual. Como ya indicamos en el capítulo dedicado a la información de coocurrencia consideramos que el rendimiento de nuestro detector de términos multipalabra es injustamente bajo debido a posibles problemas de dicha metodología. La tarea de

todas las palabras es más representativa de la ambigüedad en un texto real que la de la muestra léxica y los filtros estadísticos utilizados en las heurísticas de coocurrencia no están basados en esta colección por lo que el dominio de evaluación resulta diferente del de conteo de frecuencias. Por otro lado, los malos resultados obtenidos para la muestra léxica obtenidos por nuestra heurística jerárquica no hacen prever que la combinación de heurísticas produzca mejoras apreciables en esa colección.

Otra cuestión de relevancia a la hora de intentar combinar diversas heurísticas de desambiguación es conocer cual es la cota máxima de éxito que se puede esperar. Si una estrategia de combinación se halla suficientemente cerca de esa cota, posiblemente el esfuerzo necesario para mejorarla no compense. En nuestro caso, hemos calculado esta cota para la combinación de la información de tipo jerárquico y de tipo de coocurrencia.

Recordamos que la heurística de información jerárquica, excluyendo los casos en los que se comporta de la misma forma que la heurística aleatoria obtenía una precisión de 55.76 %. El primer sentido y la información jerárquica coinciden en un 58.33 % de los casos, un poco más de la mitad, de modo que la correlación no es muy alta (por poner un ejemplo, la correlación entre el sistema SMU-AW de la Southern Methodist University con el primer sentido es del 77.82 %). Una correlación muy alta no era esperable puesto que la precisión es baja comparada con la del primer sentido.

La heurística basada en coocurrencia obtenía una precisión de 55.52 %. El primer sentido y la heurística de coocurrencia coinciden en un 73.71 % de los casos. Esto es razonable puesto que ambas heurísticas usan información sobre frecuencias que proviene de la misma fuente (SemCor).

Las relaciones entre las heurísticas jerárquicas y de coocurrencia, restringiéndonos sólo a los nombres son las siguientes:

De los 1153 nombres en la tarea, la información jerárquica desambigua 546 (47.35 %) y la de coocurrencia 1132 (98.17 %). En 544 casos de prueba (47.18 %) han respondido ambas heurísticas, y han coincidido en 317 casos (58.27 %). Entre estos últimos casos son correctos 270 (85.17 %), de modo que obtendríamos una precisión de 85.17 % si la medida de combinación fuese desambiguar sólo los casos coincidentes. Sin embargo, 239 de estos casos corresponden a expresiones monosémicas, de forma que el grado de acuerdo entre ambas heurísticas es realmente bastante bajo.

Puesto que ambas heurísticas (jerárquicas y coocurrencia) sólo coinciden en un 58 %, podría parecer que hay mucho margen para la mejora pero no es así porque la heurística jerárquica sólo se aplica a nombres mientras que la de coocurrencias se aplica a

todas las categorías. De hecho, la cota superior para el caso de una combinación óptima es de una precisión y recall de 57.78 %, cifras bastante cercanas a la de la información de coocurrencias.

La combinación lineal de ambas heurísticas con iguales pesos proporciona los siguientes resultados reunidos en el cuadro 7.2.

Sistema	Proporción	Cob(%)	Prec(%)	Rec(%)	Puntuación
Sistema final	2448/2473	98.99	55.56	54.99	136000
H. Coocurrencias	2446/2473	98.91	55.52	54.91	135800
H. Jerárquica	552/2473	22.32	55.76	12.45	30781

Cuadro 7.1: Resultados finales de la combinación

Dada la escasa mejora alcanzada no merece la pena comparar de nuevo el sistema resultante con los demás participantes en SENSEVAL-2, de modo que las gráficas al respecto del capítulo 3 siguen siendo bastante aplicables ya que no se produce ningún cambio relativo de posición.

Queda claro que la información jerárquica y la de coocurrencias tienen bastante poca correlación en las palabras polisémicas, de modo que un método de combinación más satisfactorio (que utilice coeficientes de confianza para cada respuesta, por ejemplo) debería proporcionar una mayor mejoría. También es cierto que sería deseable una colección de prueba mayor para comprobarlo con claridad.

7.3. Comparación con sistemas no supervisados de SENSEVAL-2

En la tarea de la muestra léxica se distinguía entre sistemas supervisados y no supervisados, de manera que describiremos someramente estos otros sistemas y compararemos su rendimiento con el del nuestro.

IIT1, IIT2, IIT3 Sherwood Haynes es el autor de estos tres sistemas descritos en (Haynes, 2001). El algoritmo de desambiguación es bastante modular; se trata de asignar puntuación a cada uno de los sentidos plausibles de la palabra según su categoría gramatical. La clave está en utilizar los ejemplos de las glosas de WordNet puesto que estos ejemplos han sido incluidos con el objeto de orientar

a los anotadores humanos a la hora de distinguir entre sentidos que son difíciles de distinguir únicamente en función de su definición.

Los ejemplos que se toman no son solamente los del sentido en concreto al que se quiere asignar una puntuación, sino que también se añaden los ejemplos de sentidos relacionados. WordNet proporciona numerosas relaciones entre sentidos (realmente entre synsets, que son los que contienen glosas con ejemplos). Algunas de estas relaciones son hiperonimia, la antonimia, la hiponimia, la similitud, la meronimia, la holonimia, etc... En total se mencionan 18 relaciones de este tipo. Algunas de estas relaciones se toman directamente; por ejemplo, en la hiponimia, sólo se toma el primer synset relacionado. En otras, como la hiperonimia, se toma el cierre transitivo (es decir, se aplica la relación repetidamente para añadir más synsets y más ejemplos).

Después se aplica una función de puntuación para cada sentido. Esta función puntúa las relaciones entre las palabras del contexto y las de los ejemplos. No cuenta únicamente coocurrencias de las palabras sino que es sensible al hecho de que las palabras tengan ancestros comunes en alguna relación, la distancia a la palabra *ancla* del ejemplo (la palabra ancla del ejemplo es la palabra que ha llevado a elegir el ejemplo, puede ser la propia palabra objetivo o alguna de las relacionada por WordNet), longitud de los ejemplos y el contexto, etc... Esto por lo que respecta a los sistemas IIT1 y IIT2. La fórmula exacta es un modelo aditivo de puntuaciones por cada característica. La diferencia entre IIT1 y IIT2 está en el conjunto de características que se consideran.

El sistema IIT3 además limita los sentidos posibles de las palabras en función de los sentidos anotados anteriormente, aunque la explicación resulta un poco confusa. Los resultados son claramente inferiores a los de nuestros sistemas. Debido a problemas de tiempo, sólo el 12 % de los casos de prueba de la tarea de todas las palabras fue anotado. La precisión del mejor sistema fue de 33.5 %, muy inferior al 55.5 % de nuestro sistema para esta tarea. En la tarea de todas las palabras, su recall del 24.4 % está también muy por debajo de nuestro 40.1 %.

CL-Research-DIMAP Kenneth Litkowski presentó en (Litkowski, 2001) este sistema a la competición. En un primer paso se realizó una asociación (mapping) entre las entradas del diccionario nativo del sistema (NODE) y WordNet. Después se procede a puntuar los sentidos de manera que el que mayor puntuación obtenga sea el elegido. Si varios sentidos empatan se toma el primer sentido de WordNet. Este sistema también se considera no supervisado, lo que refuerza nuestros argumentos de que la información de frecuencias es de naturaleza no supervisada.

La puntuación de cada sentido se considera de forma aditiva. Si se encuentra un término multipalabra de WordNet se suman diez puntos. La presencia de palabras contextuales clave (partículas o preposiciones) se premiaba con dos puntos más. Cada palabra con contenido que apareciera tanto en el contexto de la palabra objetivo como en la glosa se premia con otros dos puntos.

En el diccionario NODE existen etiquetas de dominio asociadas a los sentidos. La intersección de etiquetas de dominio entre los sentidos de las palabras del contexto y las de la glosa también mejora la puntuación de un sentido (un punto por cada coincidencia).

En la tarea de todas las palabras, los resultados son de una precisión y recall de 46.1 %, muy inferior al 55.5 % de nuestro sistema. En la muestra léxica, los resultados son de una precisión y recall de 29.3 %, muy alejado de del 40.1 % de nuestro sistema.

ITRI-WASPS-Workbench Este sistema, explicado en (Tugwell and Kilgarriff, 2001) presenta numerosas peculiaridades con respecto a los anteriores. No solamente es un algoritmo de DSP sino también una herramienta lexicográfica. Está coautorado por uno de los organizadores de la competición, aunque se siguió una estricta metodología que impidió cualquier ventaja sobre los demás sistemas que participaron en la tarea de la muestra léxica.

El algoritmo en sí funciona como sigue. Un usuario elige, mediante la interfaz web de la herramienta, cuáles son las asignaciones de relaciones sintácticas características a un sentido (por ejemplo, el usuario podría escoger un sentido para *bank* cuando es el sujeto de *lend* y otro distinto cuando es modificador nominal de *river*). El sistema sugiere una lista de tales posibilidades y es el usuario quien puede asociar estas propuestas con sentidos concretos de WordNet.

Dada esta información, se construye para cada palabra una serie de listas de decisión con una *adecuación* que en este caso resulta del producto entre la información mutua entre dos palabras w_1 y w_2 con una determinada relación sintáctica R y una nueva multiplicación por el logaritmo de la frecuencia de la relación, con el objeto de evitar sobreestimar los items poco frecuentes. Este cálculo se lleva a cabo ayudándose del análisis automático realizado sobre el British National Corpus (para estimar las probabilidades). Se distinguen en total 26 tipos de relaciones. La base de datos calculada a partir del BNC consta de 70 millones de dichas relaciones.

El algoritmo sigue después los pasos del algoritmo de contexto amplio explicado en (Yarowsky, 1995b). Los autores defienden que la interacción inicial del usuario con el sistema para detectar estas semillas es de naturaleza no supervisada, puesto que no se utilizan en ningún caso los ejemplos de entrenamiento

proporcionados por la organización. Los autores estiman que para cada palabra se tarda unos quince minutos de media en proporcionar semillas para todos los sentidos de una palabra. Como puede apreciarse, la distinción entre sistemas supervisados y no supervisados a veces es realmente estrecha. En cualquier caso la interacción humana posterior a la construcción del inventario de sentidos es considerable.

Además de las claves proporcionadas por la adecuación de las expresiones también se consideran otras características como la aparición de palabras con contenido en una ventana de tamaño fijado respecto a la palabra objetivo. El tamaño por defecto de esta ventana es de 30 palabras. También se considera información de bigramas y trigramas sobre la palabra objetivo. El algoritmo está programado en Perl lo cual podría ser relevante con respecto al hecho de que los autores aducen *severas restricciones temporales* que les impidieron desambiguar los verbos (la parte del discurso más polisémica por cierto) y el nombre *day*, debido al tiempo necesario para procesar los más de 93000 ejemplos presentes en el BNC.

Como era lógico ante un sistema que necesita un cuarto de hora de interacción humana por palabra, el sistema sólo compitió en la tarea de la muestra léxica. Los resultados fueron de una precisión de 58.1 % y un recall de 31.9 %. Los autores aducen que los resultados de precisión son los que cuentan, debido a que el recall es bajo por no haber podido desambiguar más que algunos de los casos de prueba. Comparan su sistema con el mejor no supervisado (el nuestro) y concluyen que puesto que su precisión es *significativamente mayor*, esto demuestra que la relativamente pequeña interacción humana resulta muy beneficiosa.

Por supuesto, queda la duda de saber qué resultados obtendría nuestro sistema si fuera evaluado únicamente sobre las mismas palabras que el ITRI-WASP-Workbench. Para despejar esa duda hemos desambiguado 1756 palabras (todas las posibles, incluida colourless/colorless, imposible de desambiguar correctamente debido a cuestiones de lexicografía dialectal en la organización de la competición) y la precisión es de 50.89 %, que mejora la anterior de 40.2 %, cosa perfectamente predecible debido a la desaparición de los muy polisémicos verbos. La precisión de nuestro sistema sigue siendo inferior a la de Tugwell & Kilgarriff, pero el recall de nuestro sistema sigue siendo mayor, la interacción humana inexistente en nuestro caso y, finalmente, nuestra aproximación tiene la indiscutible ventaja de que es viable para cualquier palabra y mucho más eficiente que la suya.

La conclusión de la comparación es que nuestro sistema es claramente el mejor sistema no supervisado presentado a la competición; cuantitativamente en todos los casos en

lo que respecta al recall y cualitativamente al menos en el caso del sistema de (Tugwell and Kilgarriff, 2001).

Capítulo 8

Conclusiones y trabajo futuro

La anotación semántica, a pesar de ser una tarea de carácter controvertido, y de los problemas de definición y evaluación que plantea, es indudablemente un problema real. La existencia de multitud de diccionarios o inventarios de sentidos para la mayoría de las lenguas no pone en entredicho la ambigüedad semántica de las palabras. El hecho de que los anotadores humanos tengan diferencias de criterio (bajo ciertas condiciones incluso considerables) a la hora de anotar semánticamente un texto tampoco lo hace. En este sentido es fundamental contar tanto con metodologías de anotación manual claras como con estimaciones de error respecto a dicha tarea. Otra necesidad que sigue sin estar suficientemente cubierta es la de tener más colecciones de prueba anotadas manualmente, de forma que las evaluaciones de los algoritmos de DSP sean más fiables, por volumen de datos y por adaptación a diferentes dominios. En el capítulo dedicado a las relaciones conceptuales pudimos observar un comportamiento sorprendente de la precisión del sistema en función del tamaño de la ventana de desambiguación que parece achacable en principio a la forma de construir la colección. Para poder estar seguros de esto necesitamos más colecciones de gran tamaño anotadas manualmente.

Lo cierto es que aún no sabemos suficientemente cómo integrar de forma satisfactoria la desambiguación semántica con las aplicaciones actuales y futuras del procesamiento del lenguaje natural, pero la conveniencia de dicha integración es un hecho poco discutido.

La literatura científica sobre desambiguación ha tenido dos lineas principales no excluyentes. En una de ellas se presentan sistemas concretos de desambiguación basados en intuiciones respecto a la naturaleza de la ambigüedad semántica, pero sin analizar

el impacto concreto de cada una de la decisiones tomadas durante el diseño. La otra se ha centrado en tratar de acotar experimentalmente qué características contextuales son más influyentes en la ambigüedad y en qué medida, sin presentar necesariamente sistemas concretos. Desgraciadamente la primera de las lineas expuestas ha sido seguida mucho más frecuentemente que la segunda.

En esta tesis hemos intentado buscar un equilibrio entre ambas tendencias, de manera que se ha presentado un sistema de anotación semántica no supervisado, que ha obtenido buenos resultados comparativamente, pero también hemos querido profundizar en algunos de los fenómenos lingüísticos que permiten llevar a cabo dicha desambiguación.

La información taxonómica presentada en el capítulo 4 tiene a su favor el explotar unos supuestos bastante intuitivos y que funcionan muy bien en algunos casos, como en el de las frases en las que hay relaciones de hiponimia/hiperonimia entre algunos de los posibles sentidos. También funcionan bien las relaciones de holonimia y meronimia aisladamente. Desgraciadamente, esos casos no son muy frecuentes. Es posible que la combinación de la información jerárquica de WordNet con información sobre frecuencias de aparición de los distintos sentidos ayudara a mejorar la situación (la forma de la taxonomía de WordNet ayuda a la desambiguación en algunos casos y perjudica en otros). Para propósitos específicos como desambiguar los nombres de entradas de un diccionario la técnica parece muy aplicable, sin embargo, de cara a la desambiguación de textos genéricos las limitaciones son bastante evidentes. Una de ellas es que no se puede integrar de forma directa la información contextual de las categorías gramaticales distintas de los nombres, al menos en el caso del recurso léxico primario utilizado. Otros investigadores como Rada Mihalcea y Andrés Montoyo han desarrollado técnicas indirectas ya mencionadas para tratar este problema. En cualquier caso, la información de tipo jerárquico siempre se puede considerar como una fuente de información más que se puede combinar con otras técnicas para intentar mejorar los resultados individuales de cada una.

Hemos mejorado en un 25 % aproximadamente tanto en precisión como en recall el algoritmo original de densidad conceptual de Agirre y Rigau. Hemos demostrado que el algoritmo original veía perjudicado su rendimiento por la utilización de largas cadenas de hiperónimos y que la limitación de estas cadenas a los tres hiperónimos más cercanos mejora sensiblemente el recall. También hemos comprobado que los niveles superiores de la jerarquía son muy importantes, contrariamente a nuestras intuiciones *a priori*. Hemos mejorado también la forma de pesar los sentidos en la jerarquía, contando las palabras que intervienen en cada subjerarquía en vez contar una marca por cada sentido de una palabra. Por lo que respecta a la fórmula de

la densidad conceptual, la fórmula original de Agirre y Rigau ha demostrado ser la mejor de las probadas, una vez eliminados los factores de optimización empírica que aplicaron en sus condiciones experimentales concretas. Hemos comprobado asimismo que añadir otras relaciones a la de la hiperonimia no produce mejoras apreciables del rendimiento. Hemos establecido también que con estas modificaciones, el tamaño de ventana óptimo se sitúa en 271 nombres, que indica que los nombres se benefician de información muy lejana en el contexto. Por último, hemos comprobado que el algoritmo de desambiguación obtiene mejores resultados sobre textos de no ficción que sobre textos de ficción. Un interesante trabajo futuro sería aplicar la medida de similitud de sentidos presentada en (Jiang and Conrath, 1997), que resultó ser la mejor de las evaluadas en (Budanitsky and Hirst, 2000).

El uso de la información sobre coocurrencia tiene una larga historia en este campo; quizás por ello la cantidad de variaciones posibles resulta abrumadora. En esta tesis hemos demostrado que dicha información puede ser utilizada con buenos resultados comparativos para realizar anotación semántica. Parece claro que el empleo de unos modelos de lenguaje más avanzados, basados en n-gramas por ejemplo, más que en coocurrencia en una ventana de texto, o bien de tuplas de palabras relacionadas sintácticamente (lo que Yarowsky llamó *collocations*), debería mejorar ostensiblemente los resultados.

El problema es que, hoy por hoy, la capacidad de cálculo y de adquisición de textos para realizar el aprendizaje no supervisado apenas permiten diseñar experimentos teóricos de este estilo, aunque es de esperar que la situación vaya mejorando a medida que la capacidad de cálculo y la disponibilidad de textos lo permitan. Si algo ha quedado claro en la historia del PLN, es que para avanzar es necesario hacer experimentos a gran escala. La época en que una técnica de PLN se aplicaba a unas pocas palabras y de ahí se extraían conclusiones debería pasar definitivamente a la historia. Es por ello que realizar experimentos de coocurrencia con modelos de lenguaje estadísticamente complejos para aplicarlos después a una pequeña muestra de palabras posiblemente no aportaría gran cosa a la comunidad científica.

Hemos realizado experimentos con diversas medidas de asociación de palabras para formar una matriz de vecindad que explote la coocurrencia de palabras de *orden superior*, de manera que palabras que nunca han aparecido juntas en el corpus de conteo de coocurrencias, pero que están relacionadas, puedan contribuir a la desambiguación. Lo hemos hecho multiplicando en sentido algebraico la matriz de vecindad por las caracterizaciones de las glosas de los sentidos como forma de enriquecer dichas caracterizaciones, aunque manteniendo diferencias importantes con los trabajos de otros investigadores de estas técnicas.

Los mejores resultados con técnicas de asociación han sido obtenidos con la medida de información mutua. En el apéndice C puede comprobarse como el tipo de información proporcionada por cada medida resulta distinto, pero en todos los casos interesante. La medida más interpretable intuitivamente es sin lugar a dudas la información mutua (en el caso de *judge* y *jury*, su información mutua nos indica que es 620 veces más habitual que aparezcan en el mismo contexto que si fueran palabras estadísticamente independientes). La información mutua no está exenta de problemas. Como puede observarse en el cuadro C.1 del apéndice C, *ocellus* tiene una información mutua de 340 con *simple*. El extraño comportamiento de esta medida es bien conocido en palabras con frecuencias de aparición no comparables. En este caso *simple* aparece 29041 veces en el corpus y *ocellus* escasamente 150 (hay un par de órdenes de magnitud de diferencia). La medida χ^2 también proporciona palabras relacionadas, de otra manera, con las palabras objetivo, como puede comprobarse en el cuadro C.2 del apéndice C. Por último, la medida binomial debida a Dunning también aporta palabras estadísticamente dependientes, aunque parece tener el inconveniente de favorecer en exceso a las palabras o etiquetas más comunes (<NUMBER>, <PROPER_NOUN>, *say*, *like*, etc...). Esto puede verse en el cuadro C.2 del apéndice C. Es posible que la aplicación de frecuencia inversa de documento pudiera contribuir a eliminar este problema. También sería interesante probar combinaciones de las tres medidas, puesto que las relaciones que encuentran parecen complementarias, así como sus condiciones óptimas de confianza. Para poder hacerlo sería imprescindible realizar algún tipo de normalización, puesto que la medida χ^2 y la de Dunning se mueven en órdenes de magnitud muy alejados de la información mutua y se adaptan mal al modelo aditivo utilizado en esta tesis. Otro problema detectado es que, incluso después de una intensa optimización, el cálculo de la medida de Dunning resulta muy costoso computacionalmente.

Nos hemos aproximado también a otras posibles fuentes de conocimiento para apoyar nuestra búsqueda de claves para la desambiguación. La extracción automática de datos de la WWW ha tenido resultados francamente interesantes por la alta calidad de la información obtenida; sin embargo, estas técnicas deben progresar aún en la dirección de lograr una cantidad cuantitativamente mayor de información para jugar un papel más activo en DSP. El uso de sintagmas bilingües alineados ha producido unos resultados un tanto desalentadores. De nuevo, la capacidad de cálculo y de adquisición de corpora actuales han hecho que tal vez esta técnica se haya adelantado a su tiempo. Cuando los corpora de adquisición de alineaciones sean mayores será posible utilizar sólo pares de sintagmas que hayan sido alineados en un alto número de ocasiones. Se ha comprobado que bajo esas condiciones la precisión de alineación es alta. Lo que faltará comprobar entonces es si la reducción de la polisemia compensa la pérdida de sentidos correctos.

La combinación ciega de fuentes de información para DSP es otro problema abierto. En nuestro caso particular podría seguramente haberse mejorado la técnica final asignando factores de confianza a cada fuente de información. El hecho de haber utilizado sólo la información jerárquica y de coocurrencia en el sistema final ha dejado poco margen para probar modelos de combinación más complejos. Hemos decidido optar por no incluir la información sobre sintagmas alineados por sus malos resultados y la información extraída de la WWW tampoco ha sido incluida por motivos de claridad ya que la escasa cantidad de información añadida habría hecho extremadamente difícilmente interpretar su influencia en el sistema final. La correlación moderadamente alta entre las decisiones tomadas usando información de la jerarquía de WordNet y la tomada en base a la información sobre coocurrencias hacía un tanto estéril esforzarse en buscar modelos más complejos de combinación cuando la cota superior estaba marcada de antemano y no demasiado lejana.

En términos relativos, la DSP no supervisada aún obtiene resultados pobres comparada con los sistemas supervisados, pero la mejora de esos resultados debe pasar, entre otros aspectos, por una definición más acertada de la tarea y, con ello, de los mecanismos de evaluación asociados.

En suma, hemos presentado a la comunidad científica un sistema de anotación semántica no supervisado, a gran escala, que utiliza diversas fuentes de información, que descarta otras, y que obtiene, por comparación con otros sistemas, unos buenos resultados. También hemos profundizado en lo que estas fuentes de información pueden aportar potencialmente.

Bibliografía

Agirre, E. (1999). *Formalization of concept-relatedness using ontologies: Conceptual Density*. PhD thesis, Universidad del País Vasco.

Agirre, E. and Martinez, D. (2000). Exploring automatic word sense disambiguation with decision lists and the web. *Proceedings of the Semantic Annotation And Intelligent Annotation workshop organized by COLING*.

Agirre, E. and Martinez, D. (2001). Knowledge sources for word sense disambiguation. *Lecture Notes in Computer Science*, 2166.

Agirre, E. and Rigau, G. (1995). A Proposal for Word Sense Disambiguation using Conceptual Distance. In *Proceedings of the First International Conference on Recent Advances in Natural Language Processing. – Tzigov Chark, Bulgaria, September*.

Agirre, E. and Rigau, G. (1996). Word sense disambiguation using conceptual density. In *Proceedings of the International Conference in Computational Linguistics COLING'96, Copenhagen, Denmark.*, pages 16–22.

Allen, J. (1995). *Natural Language Understanding*. Addison-Wesley Publishing Co.

Alshawi, H. and Carter, D. (1994). Training and scaling preference functions. *Computational Linguistics*, 20(4):635–648.

Atkins, S. (1992). Tools for Computer-Aided Corpus Lexicography: The Hector Project. In *Papers in Computational Lexicography (COMPLEX)*, pages 1–60, Budapest.

Banko, M. and Brill, E. (2001). Scaling to Very Very Large Corpora for Natural Language Disambiguation. In *Proceedings of 39th Annual Meeting of the Association for Computational Linguistics*, pages 26–33.

Bar-Hillel (1960). Automatic Translation of Languages. *Advances in Computers.* Donald Booth and R.E Meaghers, eds. Academic New York.

Berger, A. L., Pietra, S. A. D., and Pietra, V. J. D. (1996). A Maximum Entropy Approach to Natural Language Processing. *Computational Linguistics*, 22(1):39–74.

Black, E. (1988). An experiment in computational discrimination of English word senses. *IBM Journal of Research and Development*, 32:185–194.

Brill, E. (1992). A simple rule-based part-of-speech tagger. In *Proceedings of ANLP-92, 3rd Conference on Applied Natural Language Processing*, pages 152–155, Trento, IT.

Briscoe, E. (1991). Lexical issues in Natural Language Processing. In *Proceedings of the Symposium on Natural Language and Speech, Berlin*, pages 39–68.

Brown, P. F., Della Pietra, S. A., Della Pietra, V. J., and Mercer, R. L. (1991). Word Sense Disambiguation Using Statistical Methods. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 264–270, Berkeley, California, USA.

Bruce, R. and Guthrie, L. (1992). Genus Disambiguation: A Study in Weighted Preference. In *Proc. of the 14th COLING*, pages 1187–1191, Nantes, France.

Bruce, R. and Wiebe, J. (1994). Word sense disambiguation using decomposable models. In *Proceedings of the ACL-94, 32nd Annual Meeting of the Association for Computational Linguistics*, pages 139–145, Las Cruces, New Mexico (USA).

Budanitsky, A. (1999). Lexical Semantic Relatedness and its application to Natural Language Processing. Technical report, Department of Computer Science, University of Toronto.

Budanitsky, A. and Hirst, G. (2000). Semantic Distance in WordNet : An experimental, application-oriented evaluation of five measures. In *Workshop on WordNet and Other Lexical Resources, in the North American Chapter of the Association for Computational Linguistics (NAACL-2000), Pittsburgh, PA.*

Carroll, J. and McCarthy, D. (2000). Word sense disambiguation using automatically acquired verbal preferences. In *Computers and the Humanities. Senseval Special Issue*. Kluwer Academic Publishers.

Chapman, R. (1977). *Roget's International Thesaurus,* 4^{th} *Edition*. Harper and Row, New York.

Chodorow, M., Leacock, C., and Miller, G. (2000). A Topical/Local Classifier for WSD. *Computers and the Humanities*, 34:115–120.

Choueka, Y. and Lusignan, S. (1985). Disambiguation by short contexts. *Computers and the Humanities*, 19:147–158.

Chugur, I., Gonzalo, J., and Verdejo, F. (2002). Polysemy and Sense Proximity in the Senseval-2 Test Suite. In *Proceedings of the ACL-2002 Workshop on Word Sense Disambiguation: Recent Successes and Future Directions, Pennsylvania*.

Church, K. W. and Gale, W. A. (1990). Poor Estimates of Context are Worse than None. *Proceedings of Third DARPA Speech and Natural Language Workshop*.

Church, K. W. and Hanks., P. (1989). Word association norms, mutual information and lexicography. In *27th Annual Conference of the Association of Computational Linguistics*, pages 76–82.

Church, K. W. and Hanks, P. (1990). Word Association Norms, Mutual Information and Lexicography. *Computational Linguistics*, 16(1).

Church, K. W. and Mercer, R. L. (1993). Introduction to the Special Issue on Computational Linguistics using Large Corpora. *Computational Linguistics*, 19(1):1–24.

Cohen, J. (1995). A coefficient of agreement for nominal scales. In *Educational and Psychological measurement*, volume 20, pages 37–46.

Collins, M. (1999). *Head-Driven Statistical Models for Natural Language Parsing*. PhD thesis, University of Pennsylvania.

Cowie, J., Guthrie, J., and Guthrie, L. (1992). Lexical disambiguation using simulated annealing. In *Proceedings of the International conference in computational linguistics (COLING), Nantes*, pages 359–365.

Crestan, E., El-Bèze, M., and Loupy, C. (2001). Improving WSD with Multi-Level View of Context Monitored by Similarity Measure. In *Second International Workshop on Evaluating Word Sense Disambiguation Systems (SENSEVAL), Toulouse*, pages 67–70.

Crowie, A. and et al. (1989). *Oxford's Advanced Learner's Dictionary*. Oxford University Press, 4th edition.

Cucchiarelli, A., Faggioli, E., and Velardi, P. (2000). Will Very Large Corpora Play For Semantic Disambiguation The Role That Massive Computing Is Playing For Other AI-Hard Problems? In *Proceedings of the* 2^{nd} *International Conference on Language Resources & Evaluation (LREC)*.

Cutting, D., Karger, D., Pedersen, J. O., and Tukey, J. (1992). Scatter/gather: A cluster-based approach to browsing large document collections. In *Proceedings of SIGIR'92, Copenhagen, Denmark* , pages 318–329.

Daelemans, W. and Hoste, V. (2002). Evaluation of Machine Learning Methods for Natural Language Processing Tasks. In *Proceedings of LREC-2002, the third International Conference on Language Resources and Evaluation, Las Palmas, Spain*, pages 755–760.

Daelemans, W., van den Bosch, A., Buchholz, S., Veenstra, J., and Zavrel, J. (1998). Memory-Based Word Sense Disambiguation for Senseval. In *Proceedings of CLIN98*.

Dagan, I. and Itai, A. (1994). Word Sense Disambiguation Using a Second Language Monolingual Corpus. *Computational Linguistics*, 20(4):563–596.

Dagan, I., Itai, A., and Schwall, U. (1991). Two Languages are More Informative than One. In *Proceedings of the* 29^{th} *meeting of the Association for Computational Linguistics*, pages 130–137.

Dagan, I., Lee, L., and Pereira, F. (1999). Similarity-Based Models of Word Coocurrence Probabilities. *Machine Learning*, 34:43–69.

Dagan, I., Marcus, S., and Markovitch, S. (1993). Contextual word similarity and estimation from sparse data. In *Proceedings of the* $33^{r}d$ *Meeting of the Association for Computational Linguistics*, pages 164–171.

Daudé, J., Padró, L., and Rigau, G. (2000). Mapping WordNets using structural Information. In *Proceedings of the* 38^{th} *Annual Meeting of the Association for Computational Linguistics (ACL), Hong Kong*.

Daudé, J., Padró, L., and Rigau, G. (2001). A Complete WN1.5 to WN1.6 Mapping. In *Proceedings of the NAACL Workshop WordNet and Other Lexical Resources : Applications Extensions and Customization, Pittsburg*.

Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., and Harshman, R. A. (1990). Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407.

Dempster, A., Laird, N., and Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39:1–38.

Dini, L., Vittorio Di Tomaso, and Segond, F. (1998). Error driven word sense disambiguation. In Boitet, C. and Whitelock, P., editors, *Proceedings of the Thirty-Sixth Annual Meeting of the Association for Computational Linguistics and Seventeenth International Conference on Computational Linguistics*, pages 320–321, San Francisco, California. Morgan Kaufmann Publishers.

Domingos, P. and Pazzani, M. J. (1997). On the optimality of the simple bayesian classifier under zero-one loss. *Machine Learning*, 29(2-3):103–130.

Dorr, B. and Jones, D. (1996). Role of word-sense disambiguation in lexical acquisition. Predicting Semantics from Syntactic Cues. In *Proceedings of International Conference in Computational Linguistics (COLING), Copenhagen.*

Duda, R. and Hart, P. (1973). *Pattern Classification and Scene Analysis.* John Wiley and Sons, New York.

Dunning, T. (1994). Statistical identification of language. In *Technical report CRL MCCS-94-273, Computing Research Lab, New Mexico State University.*

Dunning, T. E. (1993). Accurate Methods for the Statistics of Surprise and Coincidence. *Computational Linguistics*, 19(1):61–74.

Escudero, G., Márquez, L., and Rigau, G. (2000). Naive Bayes and Exemplar-Based Approaches to Word Sense Disambiguation Revisited. In *Proceedings of the 14th European Conference on Artificial Intelligence, ECAI, Berlin, Germany.*

Fellbaum, C. (1998). A semantic network of English: the mother of all wordnets. *Computers and the Humanities, Special Issue on EuroWordNet.*

Fellbaum, C., Joachim, G., and Landes, S. (1997). Analysis of a Hand-Tagging Task. In *Proceedings of the Applied Natural Language Processing (ANLP) Workshop on Tagging Text with Lexical Semantics, Why, What and How?*, Washington D.C., USA.

Fernández-Amorós, D., Gonzalo, J., and Verdejo, F. (2001a). The Role of Conceptual Relations in Word Sense Disambiguation. In *Proceedings of the 6th International Workshop on Applications of Natural Language for Information Systems (NLDB), Madrid*, volume 3 of *LNI Series*, pages 87–98. GI Publishers.

Fernández-Amorós, D., Gonzalo, J., and Verdejo, F. (2001b). The UNED systems at SENSEVAL-2. In *Second International Workshop on Evaluating Word Sense Disambiguation Systems (SENSEVAL), Toulouse*, pages 75–78.

Francis, S. and Kucera, H. (1967). Computational Analysis of present-day American English. *Providence, Rhode Island: Brown University Press*.

Francis, W. and Kucera, H. (1982). *Frequency Analysis of English Usage: Lexicon and Grammar*. Houghton Mifflin Company, Boston.

Fuji, A. (1998). *Corpus-based Word Sense Disambiguation*. PhD thesis, Tokyo Institute of Technology.

G. and M. Merriam Company, S. M. (1971). *Webster's Third New International Dictionary*. G and M. Merriam Company, Springfiled MA.

Gale, W., Church, K. W., and Yarowsky, D. (1994). Discrimination Decisions for 100,000-Dimensional Spaces. In A. Zampolli, N. C. and Palmer, M., editors, *Current Issues in Computational Linguistics: In honour of Don Walker*, pages 429–550. Kluwer Academic Publishers, The Netherlands.

Gale, W. A., Church, K. W., and Yarowsky, D. (1992a). Estimating upper and lower bounds on the performance of word-sense disambiguation programs. *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics*.

Gale, W. A., Church, K. W., and Yarowsky, D. (1992b). One Sense per Discourse. *Proceedings of DARPA Speech and Natural Language Workshop*.

Gale, W. A., Church, K. W., and Yarowsky, D. (1992c). Work on statistical methods for word sense disambiguation. In *AAAI Fall Symposium on Probabilistic Approaches to Natural Language*, pages 54–60, Cambridge, MA.

Gale, W. A., Church, K. W., and Yarowsky, D. (1993). A method for disambiguating word senses in a large corpus. *Computers and the Humanities*, 26(5):415–439.

Gaustad, T. (2001). Statistical Corpus-Based Word Sense Disambiguation: Pseudowords vs. Real Ambiguous Words. In *Proceedings of the ACL 2001 Student Research Workshop*, pages 61–66.

Golub, G. H. and van Loan, C. F. (1986). *Matrix Computations*. The Johns Hopkins University Press, Baltimore and London.

Gonzalo, J., Chugur, I., and Verdejo, F. (2004). *Word Sense Disambiguation: Algorithms, Applications, and Trends*, chapter Resources for Word Sense Disambiguation. Kluwer Academic Publishers, In Preparation.

Gonzalo, J., Peñas, A., and Verdejo, F. (1999). Lexical ambiguity and Information Retrieval revisited. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-99), MaryLand*.

Gonzalo, J., Verdejo, F., Chugur, I., and Cigarrán, J. (1998). Indexing with WordNet synsets can improve Text Retrieval. In *Proceedings of the COLING/ACL '98 Workshop on Usage of WordNet for NLP*, pages 38–44, Montreal, Canada.

Good, I. J. (1953). The population frequencies of species and the estimation of population parameters. *Biometrika*, 40:237–264.

Grisham, R., MacLeod, C., and Meyers, A. (1994). COMPLEX syntax : Building a Computational Lexicon. In *Proceedings of the* 15^{th} *International Conference in Computational Linguistics COLING'94. Kyoto, Japan*, pages 268–272.

Grolier Inc. (1991). *New Grolier's Electronic Encyclopedia*. Grolier Inc.

Group, C.-E. T. (1982). *Chinese Dictionaries : An extensive Bibliography of Dictionaries in Chinese and Other Languages*. Greenwood Publishing.

Guthrie, J., Guthrie, L., Wilks, Y., and Aidinejad, H. (1991). Subject-Dependent Co-occurrence and Word Sense Disambiguation. In *Proceedings of the* 29^{th} *Meeting of the Association for Computational Linguistics*, pages 146–152, Berkeley.

Guthrie, L., Pustejovsky, J., Wilks, Y., and Slator, B. M. (1996). The Role of Lexicons in Natural Language Processing. *Communications of the ACM*, 39(1):63–72.

Harabagiu, S., Miller, G., and Moldovan, D. (1999). WordNet 2 - A Morphologically and Semantically Enhanced Resource. In *In Proceedings of the ACL SIGLEX Workshop*.

Harley, A. and Glennon, D. (1997). Sense Tagging in Action : Combining Different Tests with Additive Weights. In *Proceedings of the SIGLEX Workshop Tagging Text with Lexical Semantics, Association for Computational Linguistics*, pages 74–78, Washington D.C.

Haynes, S. (2001). Semantic tagging using wordnet examples. In Yarowsky, D. and Preiss, J., editors, *Proceedings of the Second SENSEVAL Workshop*, pages 79–82.

Hearst, M. A. (1991). Noun homograph disambiguation using local context in large corpora. In *Proceedings of the 7th Annual Conference of the University of Waterloo Centre for the New Oxford English Dictionary*, pages 1–22, Oxford, UK.

Hearst, M. A. and Schütze, H. (1993). Customizing a Lexicon to Better Suit a Computational Task. In *Proceedings of ACL SIGLEX Workshop, Acquisition of Lexical Knowledge from Text*.

Higinbotham, D. W. (1990). *Semantic Coocurrence Networks and the Automatic Resolution of Lexical Ambiguity in Machine Translation*. PhD thesis, The University of Texas at Austin.

Hirst, G. and St-Onge, D. (1998). *WordNet : An electronic lexical database*, chapter Lexical chains as representations of context for the detection and correction of malapropisms, pages 305–332. The MIT Press, Cambridge, MA.

Hotho, A., Staab, S., and Stumme, G. (2003). Wordnet improves text document clustering. In *Proceedings of the Semantic Web Workshop at SIGIR-2003, 26th Annual International ACM SIGIR Conference, July 28-August 1, 2003, Toronto, Canada*.

Ide, N. and Veronis, J. (1998). Introduction to the Special Issue on Word Sense Disambiguation: The State of the Art. *Computational Linguistics*, 24(1):1–40.

Ilhan, H. T., Kamvar, S. D., Klein, D., Manning, C., and Toutanova, K. (2001). Combining heterogeneous classifiers for word-sense disambiguation. In Yarowsky, D. and Preiss, J., editors, *Proceedings of the Second SENSEVAL Workshop*, pages 87–90.

Jiang, J. J. and Conrath, D. W. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of the Conference on Research in Computational Linguistics*, Taiwan.

Jorgensen, J. (1990). The Psychological Reality of Word Senses. In *Journal of Psycholinguistic Research*, volume 19, pages 167–190.

Karov, Y. and Edelman, S. (1996). Learning similarity based word sense disambiguation from sparse data. In *Fourth Workshop on Very Large Corpora, pages 42–55, August.*

Karov, Y. and Edelman, S. (1998). Similarity-Based Word Sense Disambiguation. *Computational Linguistics*, 24(1):41–59.

Katz, J. J. and Fodor, J. A. (1964). The Structure of a Semantic Theory. In Fodor, J. A. and Katz, J. J., editors, *The Structure of Language: Readings in the Philosophy of Language*, pages 479–518. Prentice-Hall, Englewood Cliffs, New Jersey.

Kelly, E. and Stone, P. (1975). *Computer recognition of English Word Senses*. North-Holland, Amsterdam.

Kilgarriff, A. (1992). *Polysemy*. PhD thesis, University of Sussex.

Kilgarriff, A. (1997). I don't believe in word senses. *Computers and the Humanities. Kluwer Academic Publishers, The Netherlands*, 31:91–113.

Kilgarriff, A. (1998). SENSEVAL: An exercise in evaluating word sense disambiguation programs. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, pages 581–588, Granada, Spain.

Kilgarriff, A. and Rosenzweig, J. (1998). Senseval: Report and Results. In *Proceedings of the Language and Resources Evaluation Conference (LREC), Athens*.

Kilgarriff, A. and Rosenzweig, J. (2000). Framework and results for English SENSEVAL. *Computers and the Humanities*, 34(1-2).

Kirkpatrick, S., Gelatt, C., and Vecchi, M. (1983). Science. volume 220, pages 671–680.

Knight, K. and Luk, S. (1994). Building a Large Knowledge Base for Machine Translation. In *Proceedings of the American Association of Artificial Intelligence Conference AAAI-94. Seattle, WA*.

Krovetz, R. (1998). More than one sense per discourse. Technical report, NEC Princeton New Jersey Labs. Research Memorandum.

Krovetz, R. and Croft, B. (1989). Word sense disambiguation using machine-readable dictionaries. In Belkin, N. J. and van Rijsbergen, C. J., editors, *Proceedings of the 12th annual international ACM SIGIR conference on Research and development in information retrieval.*, pages 127–136, Cambridge, Massachussets. ACM.

Kwong, O. (2001). Word Sense Disambiguation with an Integrated Lexical Resource. In *Proceedings of the Workshop WordNet and Other Lexical Resources, of the Second Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

Leacock, C. and Chodorow, M. (1998). *WordNet : An electronic lexical database*, chapter Combining local context and WordNet similarity for word sense identification, pages 265–284. The MIT Press, Cambridge, MA.

Leacock, C., Towell, G., and Voorhees, E. (1993). Corpus-Based Statistical Sense Resolution. In *Proceedings ARPA Human Language Technology Workshop*, pages 260–265, Princeton, NJ.

Lenat, D. (1995). Cyc : A large scale investment in knowledge infrastructure. *Communications of the ACM*, 38(1).

Lesk, M. (1986). Automatic Sense Disambiguation using Machine Readable Dictionaries: How to Tell a Pine Cone from An Ice Cream Cone. In *Proceedings of the 5th annual international conference on Systems documentation*, pages 24–26. ACM Press.

Lesk, M. (1988). They Said True Things, but Called Them By Wrong Names. Vocabulary Problems Over Time in Retrieval. In *Proceedings of the 1988 Waterloo OED conference*, pages 1–10, Waterloo, Ontario.

Levin, B. (1993). *English Verb Classes and Alternation: A Preliminary Investigation.* The University of Chicago Press.

Lin, D. (1997). Using syntactic dependency as local context to resolve word sense ambiguity. In *Meeting of the Association for Computational Linguistics*, pages 64–71.

Lin, D. (1998). An information-theoretic definition of similarity. In *Proceedings of the* 15^th *International Conference on Machine Learning*, pages 296–304. Morgan Kaufmann, San Francisco, California.

Litkowski, K. C. (2001). Use of machine readable dictionaries for word-sense disambiguation in senseval-2. In Yarowsky, D. and Preiss, J., editors, *Proceedings of the Second SENSEVAL Workshop*, pages 107–110.

Luk, A. K. (1995). Statistical Sense Disambiguation with Relatively Small Corpora Using Dictionary Definitions. In *Proceedings of the* 33^{rd} *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 181–188.

López Ostenero, F. (2002). *Un Sistema Interactivo para la Búsqueda de Información en Idiomas Desconocidos por el Usuario.* PhD thesis, Universidad Nacional de Educación a Distancia (UNED).

López Ostenero, F., Gonzalo, J., Anselmo, P., and Verdejo, F. (2002). Noun phrase translations for Cross-Language Document Selection. In *Cross Language Evaluation Forum (CLEF)*, number 2406 in Lecture Notes in Computer Science. Springer-Verlag.

Magnini, B. and Cavagliá, G. (2000). Integrating Subject Field Codes into WordNet. In *Proceedings of the* 2^{nd} *International Conference on Language Resources & Evaluation (LREC), Athens.*

Magnini, B., Strapparava, C., Pezzulo, G., and Gliozzo, A. (2001). Using Domain Information for Word Sense Disambiguation. In *Second International Workshop on Evaluating Word Sense Disambiguation Systems (SENSEVAL), Toulouse*, pages 111–114.

Mahesh, K., Nirenburg, S., Beale, S., Viegas, E., Raskin, V., and Onyshkevych, B. (1997). Word Sense Disambiguation : Why Statistics When We Have These Numbers? In *Proceedings of the International Conference on Theoretical and Methodological Issues in Machine Translation. MT Yesterday, Today and Tomorrow*, Santa Fe.

Manning, C. and Schütze, H. (1999). *Foundations of Statistical Natural Language Processing.* MIT Press, Cambridge, MA.

Marchment, G. (2002). Word sense disambiguation. Technical report, Department of Computer Science. University of Sheffield.

Masterson, M. (1967). Mechanical Pidgin Translation. *Machine Translation.* Donald Booth, ed. Wiley.

McCarthy, D. (2002). Lexical Substitution as a Task for Word Sense Disambiguation Evaluation. In *Proceedings of the ACL Workshop on Word Sense Disambiguation: Recent Successes and Future Directions.*

McCarthy, D., Carroll, J., and Preiss, J. (2001). Disambiguating noun and verb senses using automatically acquired selectional preferences. In *Second International Workshop on Evaluating Word Sense Disambiguation Systems (SENSEVAL), Toulouse*, pages 119–122.

McDonald, J. E., Plate, T., and Schvaneveldt, R. W. (1990). *Using Pathfinder to extract information from texts. In Pathfinder associative networks : Studies in Knowledge Organization.* Ablex, Norwood NJ.

McQuitty, L. (1966). Similarity analysis by reciprocal pairs for dicrete and continuous data. *Educational and Psychological Measurement*, 26:825–831.

McRoy, S. W. (1992). Using Multiple Knowledge Sources for Word Sense Discrimination. *Computational Linguistics*, 18(1):1–30.

Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., and Teller, E. (1953). J. Chem. Phys. volume 21, page 1087.

Mihalcea, R. and Moldovan, D. (1999). A Method for Word Sense Disambiguation of Unrestricted Text. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics, Maryland, NY.*

Mihalcea, R. and Moldovan, D. (2000a). An Iterative Approach to Word Sense Disambiguation. In *Proceedings of FLAIRS*, pages 219–223.

Mihalcea, R. and Moldovan, D. (2000b). Using WordNet and Lexical Operators to Improve Internet Searches. *IEEE Internet Computing 4*, 1:34–43.

Mihalcea, R. and Moldovan, D. I. (2001). A Highly Accurate Bootstrapping Algorithm for Word Sense Disambiguation. *International Journal on Artificial Intelligence Tools*, 10(1-2):5–21.

Miller, G. (1990a). Five papers on WordNet. *Special Issue of International Journal of Lexicography.*

Miller, G. (1990b). Nouns in WordNet: A Lexical Inheritance System. *International Journal of Lexicography.*, 3(4).

Miller, G. and Fellbaum, C. (1991). Semantic networks of english. In *Cognition*, volume 41, pages 197–229.

Miller, G., R., B., Fellbaum, C., Gross, D., and Miller, K. (1990). Introduction to WordNet: An On-line Lexical Database. *Journal of Lexicography*, 3(4):234–244.

Miller, G. A. (1985). Dictionaries of the Mind. In *Proceedings of* 23^{rd} *Annual Meeting of the Association for Computational Linguistics*, pages 305–314.

Miller, G. A. (1995). WordNet: A Lexical Database for English. *Communications of the ACM*, 38(11):39–41.

Miller, G. A., Leacock, C., Randee, T., and Bunker, R. (1993). A semantic concordance. In *Proceedings of the* 3^{rd} *DARPA Workshop on Human Language Technology, Plainsboro New Jersey*, pages 303–308.

Miller, G. A. and Walter, C. G. (1991). Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 1:1–28.

Montoyo, A. (2002). *Desambiguación léxica mediante marcas de especificidad*. PhD thesis, Universidad de Alicante.

Montoyo, A., Palomar, M., and Rigau, G. (2001). Lexical Enrichment of WordNet with Classification Systems Using Specification Marks Method. In *Proceedings of the NLDB'01*, pages 109–119.

Montoyo, A., Suárez, A., and Palomar, M. (2002). Combining Supervised-Unsupervised Methods for Word Sense Disambiguation. In Gelbukh, A. F., editor, *Computational Linguistics and Intelligent Text Processing, Third International Conference, CICLing 2002, Mexico City, Mexico, February 17-23, 2002, Proceedings*, volume 2276 of *Lecture Notes in Computer Science*, pages 156–164. Springer.

Montoyo, A. and Suárez, A. (2001). The University of Alicante Word Sense Disambiguation System. In Yarowsky, D. and Preiss, J., editors, *Second International Workshop on Evaluating Word Sense Disambiguation Systems (SENSEVAL), Toulouse*, pages 131–134.

Moon, R. (2000). Lexicography and Disambiguation : The Size of the Problem. In *Computers and the Humanities.*, volume 34, pages 99–102. Kluwer Academic Publishers.

Mooney, R. J. (1996). Comparative Experiments on Disambiguating Word Senses: An Illustration of the Role of Bias in Machine Learning. In Brill, E. and Church, K., editors, *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 82–91. Association for Computational Linguistics, Somerset, New Jersey.

Morris De Groot (1986). *Probability and Statistics (second edition)*. Addison-Wesley. Reading, Massachussets.

Mosteller, F. and Wallace, D. (1964). *Inference and Disputed Authorship : The Federalist*. Addison-Wesley, Reading, Massachussets.

Ng, H. (1997a). Getting serious about word sense disambiguation. In *Tagging Text with Lexical Semantics: Why, What, and How? ANLP Workshop, Washington, D.C.*

Ng, H. T. (1997b). Exemplar-Based Word Sense Disambiguation: Some Recent Improvements. In Cardie, C. and Weischedel, R., editors, *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 208–213. Association for Computational Linguistics, Somerset, New Jersey.

Ng, H. T. and Lee, H. B. (1996). Integrating Multiple Knowledge Sources to Disambiguate Word Sense: An Exemplar-Based Approach. In Joshi, A. and Palmer, M., editors, *Proceedings of the Thirty-Fourth Annual Meeting of the Association for Computational Linguistics*, pages 40–47, San Francisco. Morgan Kaufmann Publishers.

Ng, H. T. and Zelle, J. (1997). Corpus-Based Approaches to Semantic Interpretation in Natural Language Processing. In *Artificial Intelligence Magazine, Special Issue on Natural Language Processing*, volume 18(4), pages 45–64. American Association for Artificial Intelligence.

Niwa, Y. and Nitta, Y. (1994). Co-ocurrence vectors from corpora vs. distance vectors from dictionaries. In *Proceedings of the* 15^{th} *Conference on Computational Linguistics (COLING)*, pages 304–309.

Pedersen, T. (2001). Machine Learning with Lexical Features : The Duluth Approach to SENSEVAL-2. In *Second International Workshop on Evaluating Word Sense Disambiguation Systems (SENSEVAL), Toulouse*, pages 139–142.

Pedersen, T. and Bruce, R. (1997a). A New Supervised Learning Algorithm for Word Sense Disambiguation. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence*, pages 604–609, Providence, RI.

Pedersen, T. and Bruce, R. (1997b). Distinguishing Word Senses in Untagged Text. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 197–207, Providence, RI.

Pedersen, T., Bruce, R., and Wiebe, J. (1997). Sequential Model Selection for Word Sense Disambiguation. In *Proceedings of the Fifth Conference on Applied Natural Language Processing*, pages 388–395, Washington, DC.

Peh, L. S. and Ng, H. T. (1997). Domain-Specific Semantic Class Disambiguation Using WordNet. In *Proceedings of the Fifth Workshop on Very Large Corpora, Beijing*, pages 55–64.

Perry, J. W. (1955). Translation of Russian Technical Literature by Machine. *Mechanical Translation*, 2:15–26.

Peñas, A. (2002). *Website Term Browser. Un sistema interactivo y multilingüe de búsqueda textual basado en técnicas lingüísticas*. PhD thesis, Universidad Nacional de Educación a Distancia (UNED).

Peñas, A., Gonzalo, J., and Verdejo, F. (2001). Cross-language information access through phrase browsing.

Procter, P., Ilson, R., and Ayto, J. (1978). *Longman Dictionary of Contemporary English.* Longman Group Limited, Harlow, UK.

Quinlan, J. (1993). *Programs for machine learning*. San Mateo, CA: Morgan Kaufmann.

Resnik, P. (1993). *A Class-based Approach to Lexical Relationships*. PhD thesis, University of Pennsylvania.

Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the IJCAI*, pages 448–453.

Resnik, P. (1997). Selectional preference and sense disambiguation. *Proceedings of the ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How?*

Resnik, P. (1998). Disambiguating noun groupings with respect to wordnet senses. In *Proceedings of the third Workshop on Very Large Corpora, MIT*, pages 95–130.

Resnik, P. (1999). Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language. *Journal of Artificial Intelligence Research*, 11:95–130.

Resnik, P. and Yarowsky, D. (1997). A perspective on WSD methods and their evaluation. In *Proceedings of the ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How?*, Washington D.C., USA.

Resnik, P. and Yarowsky, D. (1999). Distinguishing Systems and Distinguishing Senses: New Evaluation Methods for Word Sense Disambiguation. In *Journal of Natural Language Engineering*, volume 5(2), pages 113–134.

Rigau, G., Atserias, J., and Agirre, E. (1997). Combining Unsupervised Lexical Knowledge Methods for Word Sense Disambiguation. In Cohen, P. R. and Wahlster, W., editors, *Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics (ACL-EACL)*, pages 48–55, Somerset, New Jersey.

Rivest, R. L. (1987). Learning Decision Lists. *Machine Learning*, 2:229–246.

Sanderson, M. (1994). Word Sense Disambiguation and Information Retrieval. In *Proceedings of the* 17^{th} *annual international ACM SIGIR conference on Research and development in information retrieval*, pages 142–151. Springer-Verlag New York Inc.

Sanderson, M. (2000). Retrieving with Good Sense. *Information Retrieval*, 2(1):49–69.

Santamaría, C., Gonzalo, J., and Verdejo, F. (2001). Internet como fuente de información léxica: Extracción de etiquetas de dominio y detección de nuevos sentidos. In *Procesamiento del Lenguaje Natural*.

Santamaría, C., Gonzalo, J., and Verdejo, F. (2003). Automatic association of web directories to word senses. *Computational Linguistics*, 29(3).

Schuler, W. (2001). Computational Properties of Environment-based Disambiguation. In *Proceedings of the* 39^th *Meeting of the Association for Computational Linguistics*, pages 466–473.

Schütze, H. (1993). Word space. In Hanson, S., Cowan, J., and Giles, C., editors, *Advances in Neural Information Processing Systems 5*. Morgan Kaufmann Publishers.

Schütze, H. (1998). Automatic Word Sense Discrimination. *Computational Linguistics*, 24(1):97–124.

Schütze, H. (1992a). Context Space. In *AAAI Fall Symposium on Probabilistic Approaches to Natural Language*, pages 113–120, Cambridge, MA.

Schütze, H. (1992b). Dimensions of meaning. In *Proceedings of the ACM/IEEE Conference on Supercomputing*, pages 787–796. IEEE Computer Society Press.

Schütze, H. and Pedersen, J. (1995). Information Retrieval Based on Word Senses. In *Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175.

Schütze, H. and Silverstein, H. (1997). Projections for Efficient Document Clustering. In *Proceedings of SIGIR'97, Philadelphia*, pages 74–81.

Shannon, C. (1948). A Mathematical Theory of Communication. *Bell Systems Technical Journal*, 27(3):623–656.

Simpson, J. A. and Weiner, E., editors (1989). *The Oxford English Dictionary, Second Edition*. Clarendon Press, Oxford.

Small, S. and Rieger, C. (1982). Parsing and Comprehending with Word Experts (A Theory and its Realization). In Lehnert, W. G. and Ringle, M. H., editors, *Strategies for Natural Language Processing*, pages 89–147. Erlbaum, Hillsdale, NJ.

Stetina, J., Kurohashi, S., and Nagao, M. (1998). General Word Sense Disambiguation Method Based on A Full Sentential Context. In Harabagiu, S., editor, *Use of WordNet in Natural Language Processing Systems: Proceedings of the Conference*, pages 1–8. Association for Computational Linguistics, Somerset, New Jersey.

Stevenson, M., Cunningham, H., and Wilks, Y. (1998). Sense Tagging and Language Engineering. In *European Conference on Artificial Intelligence (ECAI)*, pages 185–189.

Stevenson, M. and Wilks, Y. (1999). Combining Weak Knowledge Sources for Sense Disambiguation. In *IJCAI*, pages 884–889.

Stroustrup, B. (2000). *The C++ Programming Language (Special Edition)*. Addison-Wesley. Reading, Massachussets.

Sussna, M. (1993). Word Sense Disambiguation for Free-text Indexing Using a Massive Semantic Network. In *Proceedings of the Second International Conference on Information and Knowledge Management (CIKM)*, pages 67–74.

Suárez, A. and Montoyo, A. (2001). Estudio de cooperación entre métodos de desambiguación léxica : Marcas de especificación vs. máxima entropía. *Procesamiento del Lenguaje natural, Revista de la Sociedad Española de Procesamiento del Lenguaje Natural*, 27(1):207–214.

Thanopoulos, A., Fakotakis, N., Patras, R., and Kokkinakis, G. (2000). Automatic Extraction of Semantic Similarity of Words from Raw Technical Texts. In *Proceedings of the* 2^{nd} *International Conference on Language Resources & Evaluation (LREC)*.

Tom O'Hara, J. W. and Bruce, R. (2000). Selecting Decomposable Models for Word Sense Disambiguation : The grling-sdm System. In *Computers and the Humanities*, volume 34, pages 159–164.

Towell, G. and Voorhees, E. M. (1998). Computational Linguistics : Special Issue on Word Sense Disambiguation. In *Disambiguating Highly Ambiguous Words*, volume 24(1), pages 125–145.

Tugwell, D. and Kilgarriff, A. (2001). Wasp-bench : A lexicographic tool supporting word sense disambiguation. In Yarowsky, D. and Preiss, J., editors, *Second International Workshop on Evaluating Word Sense Disambiguation Systems (SENSEVAL), Toulouse*, pages 151–154.

Ureña, L. A. (2001). *Resolución de la Ambigüedad Léxica en Tareas de Clasificación de Documentos*. Colección de Monografías. Sociedad Española para el Procesamiento del Lenguaje Natural.

Veenstra, J., Bosch, A. V., Buchholz, S., Daelemans, W., and Zavrel, J. (1998). Memory-based Word Sense Disambiguation. In *Computers and the Humanities, Special issue on SENSEVAL*.

Verdejo, M., Gonzalo, J., Fernández-Amorós, D., Peñas, A., and López, F. (2000a). ITEM: un motor de búsqueda multilingüe basado en indexación semántica. In *Primeras Jornadas de Bibliotecas Digitales (JBIDI), Valladolid*, pages 139–148.

Verdejo, M., Gonzalo, J., Peñas, A., López, F., and Fernández-Amorós, D. (2000b). Evaluating Wordnets in Cross-Language Text Retrieval: The ITEM Multilingual Search Engine. In *Second Language Resources and Evaluation Conference (LREC'2000), Athens*, pages 1769–1774.

Veronis, J. and Ide, N. (1990). Word Sense Disambiguation with Very Large Neural Networks Extracted from Machine Readable Dictionaries. In *9th European Conference on Artificial Intelligence, ECAI'90,Stockholm*, pages 366–368.

Vider, K. and Kaljurand, K. (2001). Automatic WSD : Does it make sense for Estonian? In *Second International Workshop on Evaluating Word Sense Disambiguation Systems (SENSEVAL), Toulouse*, pages 159–162.

Voorhees, E. M. (1993). Using WordNet to Disambiguate Word Senses for Text Retrieval. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 171–180. ACM Press.

Vossen, P. (1997). Eurowordnet: A multilingual database for information retrieval. In *Proceedings of the DELOS workshop on Cross-language Information Retrieval*.

Vossen, P. (1998). *EuroWordNet: a multilingual database with lexical semantic networks*. Kluwer Academic Publishers.

Vossen, P., Peters, W., and Gonzalo, J. (1999). Towards a universal index of meaning. In *Proceedings of SIGLEX (Special Interest Group on the Lexicon), Association for Computational Linguistics*.

Véronis, J. (1998). A study of polysemy judgements and inter-annotator agreement. *Programme and advanced papers of the Senseval workshop. Herstmonceux Castle, England*.

Véronis, J. and Ide, N. (1995). Large Neural Networks for the Resolution of Lexical Ambiguity. In *Computational Lexical Semantics. Natural Language Processing Series*, pages 251–270. Cambridge University Press.

Walker, D. E. (1987). Knowledge Resource Tools for Accessing Large Text Files. *Machine Translation : Theoretical and Methodological Issues*, pages 247–261.

Wallis, P. (1993). Information retrieval based on paraphrase. In *Proceedings of PACLING Conference, 1993*.

Ward, J. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58:236–244.

Weaver, W. (1955). Translation. *William N. Locke and A. Donald Booth, editors, Machine translation of languages : Fourteen Essays. MIT Press*.

Wilks, Y. (1993). Providing Machine Tractable Dictionary Tools. In Pustejovsky, J., editor, *Semantics and the Lexicon*, pages 341–401. Kluwer Academic Publishers, London.

Wilks, Y., Fass, D., Guo, C., McDonald, J., Plate, T., and Slator, B. (1990). Providing Machine Tractable Dictionary Tools. In *Machine Translation 5(2), 99-151.*

Wilks, Y. and Stevenson, M. (1996). The Grammar of Sense : Is word sense tagging much more than part-of-speech tagging? Technical report, University of Sheffield, UK.

Wilks, Y. and Stevenson, M. (1997a). Combining Independent Knowledge Sources for Word Sense Disambiguation. In *Proceedings of the Conference Recent Advances in Natural Language Processing*, pages 1–7, Tzigov Chark, Bulgaria.

Wilks, Y. and Stevenson, M. (1997b). Sense tagging: Semantic tagging with a lexicon. In *Proceedings of the SIGLEX Workshop Tagging Text with Lexical Semantics: What, why and how?, Washington, D.C.*, pages 47–51.

Wilks, Y. and Stevenson, M. (1998a). Word Sense Disambiguation using Optimised Combinations of Knowledge Sources. In *COLING-ACL*, pages 1398–1402.

Wilks, Y. and Stevenson, M. (1998b). Word sense disambiguation using optimised combinations of knowledge sources. *Proceedings of COLING (International Conference in computational linguistics) - ACL (Association for Computational Linguistics) in Montreal, Quebec, Canada*.

Yarowsky, D. (1992). Word-Sense Disambiguation using Statistical Models of Roget's Categories Trained on Large Corpora. In *Proceedings of International Conference in Computational Linguistics (COLING)*, pages 454–460, Nantes, France.

Yarowsky, D. (1993). One Sense per Collocation. In *Proceedings, ARPA Human Language Technology Workshop*, pages 266–271, Princeton.

Yarowsky, D. (1995a). Decision Lists For Lexical Ambiguity Resolution : Application to Accent Restoration in Spanish and French. In *Proceedings of the* 32^{nd} *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 88–95, Las Cruces, New Mexico.

Yarowsky, D. (1995b). Unsupervised Word Sense disambiguation Rivaling Supervised Methods. In *Proceedings of the* 33^{rd} *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 189–196.

Yarowsky, D. (1999). Corpus-based Techniques for Restoring Accents in Spanish and French Text. In *Natural Language Processing Using Very Large Corpora*, pages 99–120. Kluwer Academic Publishers.

Zipf, G. (1945). The Meaning-Frequency Relationship of Words. *General Psychology*, 33:251–266.

Parte IV

APÉNDICES

Apéndice A

La arquitectura de desarrollo y evaluación PIXIE-DIXIE

A.1. Introducción

Para sistematizar el desarrollo de estas heurísticas de desambiguación, de forma que fueran efectivas y reutilizables para las tres colecciones utilizadas (SemCor y las tareas de todas las palabras y la muestra léxica de SENSEVAL) y también para posibilitar su evaluación y la postproducción de los resultados para su estudio, ha sido necesario desarrollar una plataforma de programación adecuada. Se ha optado por una arquitectura modular con un cliente (PIXIE) y un servidor (DIXIE).

¿Por qué una arquitectura cliente-servidor? Hay dos buenas motivaciones para ello:

- Aunque la mayoría de las veces tanto el cliente como el servidor se han ejecutado en una misma máquina, es muy deseable poder ejecutarlos en maquinas distintas y que se comuniquen a través de la red. El servidor puede acaparar gran cantidad de recursos de modo que poder dedicar una máquina a ser servidor y realizar las consultas mediante el cliente resulta muy práctico.

- El servidor incorpora un intérprete de comandos limitado para ejecutar instrucciones, sin embargo, resultaría absurdo replicar una *shell* entera cuando ya existen muchas disponibles. La ventaja de la arquitectura cliente-servidor es que permite utilizar *scripts* nativos y sólo llamar a las funciones específicas del

servidor cuando sea necesario (por ejemplo, si queremos ir variando un parámetro de un desambiguador podemos realizar un bucle *for* en *shell script* dentro del cual hagamos una petición al servidor especificando ese parámetro, eso es mucho más simple que incorporar estructuras de control y variables al servidor).

Por lo que respecta a la modularidad es importante ser capaz de separar los servicios y recursos concretos de los desambiguadores individuales. Por ello, el servidor admite una serie de comandos para cargar servicios que pueden necesitarse para manejar la entrada y también dentro de los desambiguadores. Estos servicios son de segmentación, lematización, separación de frases, anotación morfosintáctica con el Brill tagger, utilización de listas de parada, detección de términos multipalabra, carga y manejo de las matrices de vecindad, caracterizaciones de sentidos como vectores característicos y otros que se salen del ámbito de estos artículos como una matriz de distancia entre sentidos para realizar evaluación de grano más grueso que el habitual.

Estos servicios se encuentran desligados tanto de los desambiguadores como de los manejadores de la entrada y existen comandos para descargarlos y para recargarlos con otros parámetros según sea necesario, todo ello sin alterar los desambiguadores definidos y sin reiniciar el servidor.

Los sistemas de desambiguación y las heurísticas individuales también son módulos independientes que se pueden cargar y descargar sin reiniciar el servidor. Esto es vital para tener una flexibilidad suficiente en el desarrollo. Por ejemplo, si se está desarrollando un desambiguador que utiliza la matriz de vecindad del capítulo 5, tener que reiniciar el servidor y cargar la tabla (de 800 MB) otra vez sólo porque se han modificado unas lineas en el desambiguador resultaría en unos tiempos de desarrollo inaceptables. Para evitar esto los desambiguadores han sido implementados como librerías dinámicas que se enlazan y desenlazan cada vez que son necesarias, para poder actualizarlas sin necesidad de tener que recargar todos los recursos léxicos. Además, no hay que olvidar que servicios que pueden ser interesantes para una combinación de colección y desambiguador, pueden no serlo para otra (solo la muestra léxica necesita la anotación morfosintáctica, y heurísticas como el primer sentido o el sentido aleatorio no necesitan tener caracterizaciones de los sentidos, por ejemplo).

Además de todo ello cada heurística admite una serie de parámetros (a modo de ejemplo, en desambiguación jerárquica el tamaño de la ventana, o en desambiguación basada en coocurrencias el tipo de medida; información mutua, χ^2 o la de Dunning, etc...) La arquitectura permite definir diversos sistemas que pueden utilizar las mismas heurísticas con distintos parámetros.

Otro componente importante son los manejadores de entrada. Para proporcionar una interfaz uniforme a las heurísticas y que puedan aceptar datos que pueden provenir de tres colecciones de textos en formatos distintos hemos utilizado unos manejadores de colección. Para el caso del SemCor la conversión de los ficheros resulta inmediata de manera que estos manejadores (uno por colección) los hemos implementado como *parsers* de XML utilizando la herramienta Xerces/C++ de la Apache Foundation. Estos manejadores están al tanto de proporcionar el contexto especificado para cada heurística (puede ser de longitud fija, a nivel de frase, de párrafo o de todo un documento), como de buscar las expresiones multipalabra adecuadas si es necesario y también de buscar los sentidos correspondientes a las palabras objetivo, etc. Se trata de proporcionar una interfaz uniforme a las heurísticas para que funcionen correctamente sin saber de qué colección proviene el texto. Obviamente, un manejador para texto plano también sería deseable de cara a ofrecer un servicio de prueba de un desambiguador o del propio servidor. En general estos manejadores completan la información que no viene codificada en cada colección. Una excepción a esto sería la información sintáctica que viene en la colección de la tarea de todas las palabras de SENSEVAL-2 (extraída del PennTreeBank) que no hemos utilizado en estos experimentos, motivo por el cual los otros manejadores no la intentan calcular.

A.2. Comandos del intérprete

Estos son algunos de los comandos que admite el intérprete del servidor junto con una breve descripción

carga_diccionario_bilingue Carga un diccionario bilingüe español-inglés. (También pensado para la desambiguación basada en sintagmas pero descartado finalmente).

carga_matriz_proximidad Carga una matriz de proximidad entre sentidos que permite hacer una evaluación de grano grueso.

desambigua Aplica la heurística o sistema complejo de heurísticas definido previamente a una colección concreta.

genera_tabla Genera una tabla LaTeX con la evaluación de un sistema, desglosada por las heurísticas de la componen.

radio Permite fijar una ventana de tamaño fijo para la desambiguación de cada categoría gramatical.

carga_brill_tagger Carga los datos para utilizar el POS tagger de Eric Brill.

carga_euromapping Carga un *mapping* para pasar de lemas en español a synsets de WordNet-1.7 (empleado en el experimento de los sintagmas alineados).

carga_tabla Carga la tabla de coocurrencias, a partir de esta tabla se calculan las medidas de asociación entre palabras.

detector_de_nombres Carga una lista de nombres propios de persona para intentar detectarlos en el texto mediante heurísticas. También se detectan números.

encola Los sistemas se forman encolando heurísticas en cascada, si una heurística no consigue desambiguar una palabra se pasa a la siguiente. Este comando encola una nueva heurística a un sistema.

entrena Produce caracterizaciones de sentidos basadas en la información de entrenamiento de la muestra léxica.

evalua Evalúa un sistema ya ejecutado, un fichero de respuestas, posiblemente de terceros o un directorio de ficheros de resultados. Permite seleccionar/excluir partes del discurso y desglosar los resultados por palabra (útil para la muestra léxica).

fichero_sentidos Carga la caracterizacion de los sentidos de WordNet. Normalmente las glosas, pero a veces son descripciones enriquecidas con información de la web, ejemplos de entrenamiento o con glosas de hiperónimos.

forma_base Carga el lematizador, basado en el del WordNet inglés pero simplificado.

genera_comparativa Genera una tabla LaTeX comparando las evaluaciones de varios sistemas.

lematiza Toma una palabra en inglés (el idioma en el que están todas las colecciones) y nos devuelve el lema.

multiwords Carga el conjunto de términos multipalabra de WordNet que se quiere detectar. Solo es útil en la muestra léxica, en las otras colecciones ya vienen fijados.

phrases_alineadas Carga los sintagmas junto con su alineación español-inglés.

raiz_wordnet Para especificar qué wordnet queremos utilizar. La API de wordnet no se utiliza de modo que la información se extrae directamente de los ficheros de la distribución.

stop_words Carga una lista de palabras de parada que usan en los manejadores de la entrada.

traduce Toma una palabra y nos da las traducciones según el diccionario bilingüe que debe estar cargado previamente.

umbral Umbral por debajo del cual se considera como cero una entrada de la tabla de vecindad.

Apéndice B

Publicaciones generadas durante la realización de la tesis

David Fernández-Amorós, (2000) *Efficient and Unsupervised Multilingual Semantic Tagging*, Informe técnico proyecto ITEM (TIC96-1243-C03-01).

Felisa Verdejo, Julio Gonzalo, Anselmo Peñas, Fernando López Ostenero, David Fernández Amorós, (2000) *Evaluating WordNets in a Cross-Language Retrieval Environment : The ITEM Search Engine*, Proceedings of the Second Language Resources and Evaluation Conference, Atenas, pp 1769-1774.

Felisa Verdejo, Julio Gonzalo, David Fernández-Amorós, Anselmo Peñas y Fernando López Ostenero, (2000). *ITEM: Un Motor de Búsqueda Multilingüe basado en Indexación Semántica*. Primeras Jornadas de Bibliotecas Digitales, Valladolid, pp 139-148. ISBN: 84-8448-066-6.

David Fernández-Amorós, Julio Gonzalo y Felisa Verdejo, (2001) *The Role of Conceptual Relations in Word Sense Disambiguation*, Proceedings of the 6^{th} International Workshop on Applications of Natural Language for Information Systems (NLDB). ISBN : 3-88579-332-6. Editores Ana María Moreno y Reind P. Van De Riet, volumen 3, pp 87-98, editorial GI.

David Fernández-Amorós, Julio Gonzalo y Felisa Verdejo, (2001) *The UNED Systems at SENSEVAL-2*, Proceedings of the Senseval-2 International Workshop, held in conjunction with the ACL Conference, Toulouse 2001, editores David Yarowsky y Judita Preiss, volumen 1, pp 75-78.

Apéndice C

Muestra de las medidas de asociación

En este apéndice presentamos unas tablas de las veinte palabras más relacionadas según la medida de asociación utilizada con las palabras de la muestra léxica de SENSEVAL-2 excepto *see*, que forma parte de nuestra lista de palabras de parada. Los números entre paréntesis indican el valor numérico de la asociación. El cuadro C.1 trata la información mutua, el C.2 la χ^2 y el C.3 la medida binomial de Dunning.

C.1. Información mutua

Cuadro C.1: Palabras más asociadas con respecto a una muestra de palabras según la medida de información mutua

palabra	vecindad de palabras
art	decorative (625) pictorial (619) thou (595) proficient (529) imitative (501) hub (489) archaeology (485) whistler (425) healing (414) angling (408) culinary (405) sculpture (400) corruptible (400) photography (388) handicraft (373) adept (337) aesthetic (335) rhetoric (328) icelander (326) statuary (312) connoisseur (311)
authority	vested (1013) abrogate (704) municipal (664) judiciary (648) legislative (639) usurped (611) subordination (570) marital (569) parental (556) guaranty (538) federal (535) papal (517) centralized (504) concurrent (485) unquestioned (483) ecclesiastical (480) judicial (468) unimpeachable (460) despotic (439) competent (438) subvert (429)

Continúa en la siguiente página

Cuadro C.1 – continuado de la página anterior

palabra	vecindad de palabras
bar	capstan (1914) shuttered (1028) barkeeper (839) bartender (817) transverse (795) visor (753) barmaid (712) bolt (693) iron (675) cage (673) padlock (603) stanchion (561) socket (554) gridiron (554) whitish (547) croup (540) overlie (525) barricaded (503) zebra (494) indict (493) stripe (482)
begin	wane (187) thicken (180) afresh (164) anew (161) ending (159) drowse (139) undress (136) collaborate (130) sprout (129) whimper (119) slacken (116) lengthen (116) crackle (115) bombardment (115) omega (114) snore (114) pacing (111) rummage (110) unroll (108) gulden (108) agitate (108)
blind	deaf (2208) lame (1179) maimed (1071) buff (977) unreasoning (955) sightless (868) paralytic (693) leper (675) dumb (665) slat (643) eyesight (605) crustacean (602) groping (581) venetian (581) necked (576) wilfully (573) insensate (570) blindness (565) alley (565) arethusa (560) toothless (506)
bum	bum (44417) tapster (41532) disruption (12453) booze (9085) topple (6447) algorithm (6307) punk (5835) gauntlets (5762) synonym (5044) meter (4872) configuration (4823) clarity (4780) med (4614) malcontent (4368) jawbone (3919) mediate (3834) hardware (3701) loafing (3625) beck (3595) decimal (3529) dime (3483)
call	beck (438) switchboard (333) phone (292) bugle (277) clarion (241) neo (209) processor (207) billing (200) quota (195) slanderer (184) crier (177) curlew (169) logician (166) caller (163) ganymede (162) fbi (155) flu (154) goer (153) strep (152) dalliance (148) quince (147)
carry	stretcher (538) loads (365) portage (289) suitcase (268) satchel (258) swag (256) knapsack (249) valise (246) sedan (240) litter (230) petrol (224) momentum (223) connotation (216) sling (215) basket (203) baggage (194) fetching (192) pom (191) impetus (188) babylon (181) load (180)
chair	bottomed (1499) sedan (1472) wicker (1303) swivel (1223) upholster (1178) cushioned (1062) washstand (985) horsehair (916) rocker (699) seating (694) rickety (674) tilted (665) mahogany (640) plush (614) vacate (606) carven (588) limply (566) wheeled (564) chintz (559) footstool (527) bedstead (524)
channel	dredge (3832) bayou (2086) tortuous (1896) buoy (1665) irrigation (1474) islet (1443) irrigate (1288) shoal (1220) estuary (1184) formosa (1162) trireme (1159) beagle (1149) tv (1138) bristol (1123) navigable (1111) diverted (982) bottomed (977) atoll (969) streamlet (920) torpedo (918) mainland (917)
child	motherless (463) procreation (323) fatherless (317) israel (309) grandchild (308) weaned (302) nursemaid (297) unborn (284) adult (276) illegitimate (276) kindergarten (274) josue (264) pregnancy (259) librarian (254) heth (251) plaything (250) romp (249) parent (248) educate (239) suckling (225) playground (222)
church	congregational (1713) anglican (1343) methodist (1302) militant (1273) presbyterian (1195) romish (1151) lutheran (1119) episcopal (1082) steeple (996) liturgy (954) wesleyan (893) catholic (811) methodists (737) spire (723) baptists (716) chancel (705) unitarian (697) calvinistic (690) schism (686) dignitary (650) apostolic (643)
circuit	node (10147) relay (5441) integrated (3691) transmitter (3575) mac (2984) testing (2807) circuit (2719) billed (2703) stadium (2321) carbon (2181) installation (2178) tandem (2006) microphone (1957) platinum (1780) id (1653) generator (1616) multiple (1570) assize (1561) tuned (1517) induction (1511) antenna (1383)

Continúa en la siguiente página

Cuadro C.1 – continuado de la página anterior

palabra	vecindad de palabras
collaborate	fabrication (26467) materialize (20409) melodrama (16514) maidenhood (8549) taxing (8400) eventual (7918) disseminate (7822) domineering (7346) descartes (7318) philology (7291) festal (7209) lapel (6755) sameness (6685) aphorism (6685) stanford (6549) dormer (6133) financially (6075) hartford (5948) calcareous (5945) physiologist (5666) amicable (5222)
colourless	bladed (9865) daub (4057) sulphuric (3811) vapid (3440) entree (3337) prism (3264) oxide (3144) ammonia (3127) crystallize (3120) protoplasm (3035) nutrition (2981) unshaven (2929) chloride (2842) inane (2641) liquid (2627) centred (2595) diluted (2570) puffy (2521) pert (2351) pungent (2325) bromine (2297)
cool	deliciously (1952) cucumber (1620) refreshing (1222) fevered (1115) leftist (1041) shady (987) bracing (807) translucent (797) vanilla (789) breeze (744) texan (651) fragrant (644) sultry (611) ewer (597) delightfully (591) breezy (580) invigorating (556) porridge (553) plash (534) rippled (531) ornate (503)
day	livelong (476) sabbath (247) ides (231) unleavened (196) seventh (178) pentecost (177) dawning (170) eventful (164) passover (159) rainy (158) anniversary (153) thanksgiving (152) fortieth (152) windless (148) dawn (138) sultry (137) sunshiny (133) thirtieth (129) eighth (129) gala (124) michaelmas (123)
detention	corrective (41508) segregation (12465) preventive (8031) italicize (4662) bennie (4662) quarantine (4652) repressive (4546) illegal (4405) detention (4278) abduction (4241) forcible (3497) clamorous (3452) forfeiture (3407) gamma (3367) reformatory (3276) unpromising (2921) unburied (2841) fraudulent (2510) penitentiary (2497) cellular (2377) chattel (2369)
develop	telecommunication (2415) centralized (1766) utilize (1321) mobility (1285) rep (1281) tadpole (1155) thyroid (1147) concentration (1104) tic (1100) algorithm (1082) ramp (1075) pneumonia (1027) spontaneously (999) larva (993) processing (983) inconspicuous (962) horsepower (955) adolescence (952) theorem (942) experimenter (927) implementation (918)
draw	nigher (583) inference (558) demarcation (521) scabbard (456) bowstring (365) curtain (352) nigh (350) nearer (340) sheath (339) closer (329) horoscope (317) lottery (298) saber (277) vortex (270) bead (270) suction (269) breath (263) holster (256) portiere (253) sword (249) poniard (219)
dress	gingham (1698) flounce (1150) muslin (1003) calico (906) bodice (869) necked (690) gala (674) serge (643) silk (634) looped (632) homespun (610) chemise (588) satin (577) bedraggled (570) lace (560) frills (557) cashmere (556) slovenly (550) brocade (548) bridesmaid (529) attired (524)
drift	electronics (2164) incidence (1316) floe (1159) derelict (1031) blench (1023) shrift (901) iridescent (841) mooring (838) southerly (805) aimlessly (795) detrimental (769) innuendo (751) leeward (737) casing (721) drift (663) rafts (650) glacial (648) winnow (640) video (617) oarsman (609) hawser (605)
drive	hansom (690) buggy (621) barouche (604) gig (585) cabriolet (578) cabman (562) phaeton (525) coachman (516) canaanite (516) wedge (491) postillion (452) sleigh (449) limousine (440) cab (431) beater (430) taxicab (426) chaise (420) landau (415) horsepower (409) harnessed (393) charioteer (392)

Continúa en la siguiente página

Cuadro C.1 – continuado de la página anterior

palabra	vecindad de palabras
dyke	dyke (9683) ogle (4506) outlawry (4336) porphyry (3390) tam (3085) witchery (3054) derrick (2942) fen (2661) van (2608) lush (2471) caribbean (2333) landward (2275) intrenchment (2104) newmarket (2080) princeton (2004) midland (1998) quartz (1951) acadian (1931) causeway (1918) cinch (1899) dune (1873)
face	wry (654) upturned (582) expressionless (545) blanched (441) careworn (434) seamed (423) pallor (403) immobile (395) contorted (393) sallow (388) cadaverous (381) puffy (371) impassive (360) ashen (356) bronzed (356) livid (353) overspread (353) haggard (351) flushed (345) sunburned (338) suffused (328)
facility	milieu (4356) hollander (3709) unequalled (2513) micro (2482) transportation (1888) abrogate (1815) reciprocal (1813) improperly (1796) locomotion (1789) lab (1516) reproductive (1480) cayenne (1480) tracking (1322) acquiring (1303) incidental (1272) poesy (1251) segregation (1241) looker (1221) dissemination (1215) centralization (1184) conduit (1180)
faithful	unfaithful (1063) servitor (1003) follower (955) rewarding (944) acceptation (723) adherent (712) necked (700) caliph (663) chesterfield (640) bagdad (631) helpmate (620) faithless (579) henchman (579) retainer (574) commander (555) portraiture (514) faithfulness (507) dispenser (507) firkin (488) milieu (473) collaborate (470)
fatigue	sleeplessness (1368) recuperate (1174) inured (1121) privation (1076) hunger (912) drowsiness (886) serried (760) exhausted (743) jaded (707) inclement (684) satiety (673) footsore (672) exertion (653) lateness (642) delve (631) exhaustion (631) overpowered (628) inhibition (621) languor (591) thirst (585) bodily (584)
feeling	limbed (3465) feeling (2524) osier (1985) tripe (1955) squirt (1755) coco (1702) romaine (1674) begrimed (1659) finland (1605) indissoluble (1438) snipe (1427) haystack (1381) spiritless (1367) gesticulate (1330) stoicism (1295) cayenne (1266) effrontery (1239) wen (1234) inaccurate (1219) cube (1200) tarpaulin (1172)
ferret	hutch (12203) weasel (11434) ferret (6814) coney (6287) punk (5629) simmer (5405) ping (4554) curfew (4332) yap (4284) incitement (4146) chateaubriand (4102) impale (4073) mink (3934) tee (3837) unwarrantable (3762) unlawfully (3484) overlie (3397) cartwright (3281) aura (2932) reproductive (2780) stainless (2764)
find	cathay (159) lodgment (150) rummage (132) mailbox (124) whereto (121) outlet (114) ransacked (112) pottery (112) search (111) footmark (103) clew (101) ransack (101) palatable (98) autopsy (97) grope (95) searching (93) punctuation (91) congenial (90) purchaser (88) clue (88) celebes (88)
fine	cambric (521) linen (492) punishable (436) needlework (427) broider (360) voucher (355) imprisonment (345) neo (342) twined (332) flour (322) argonaut (315) physique (312) weather (307) drizzle (298) forfeiture (298) shekels (295) drachma (285) arts (282) texture (279) broadcloth (275) aquiline (263)
fit	epileptic (1240) apoplectic (1094) coughing (1062) survival (945) horsepower (643) snugly (624) sneezing (586) ague (536) pock (462) apoplexy (453) aptly (445) hysterics (375) colic (357) biplane (356) hysteric (349) piston (333) valve (330) immoderate (329) uncontrollable (326) epilepsy (311) magneto (311)
free	unfettered (653) shackle (587) thraldom (534) villein (524) incumbrance (515) punctuation (478) coinage (464) transvaal (458) congo (418) bondman (409) basuto (404) trapper (393) bondage (379) enslaved (374) taint (374) expertise (373) wrenching (353) ingress (349) fetter (347) durance (304) archive (300)

Continúa en la siguiente página

Cuadro C.1 – continuado de la página anterior

palabra	vecindad de palabras
graceful	lithe (1959) minuet (1514) leaved (1498) shapely (1464) queenly (1453) drape (1356) contour (1273) curve (1237) frond (1119) undulation (1078) wolfish (1047) slender (945) festoon (925) supple (915) exquisitely (904) roundness (892) argonaut (864) suiting (822) tinsel (812) luxuriance (812) sinuous (807)
green	baize (3615) bowling (1742) sward (1155) lush (802) sheen (778) velvety (764) foliage (760) translucent (749) frond (704) emerald (698) turf (663) moss (645) watermelon (644) unripe (619) mossy (604) billowy (587) leek (582) verdant (576) sedge (550) hedgerow (548) herbage (547)
grip	tightened (2873) vise (2808) racquet (2740) tightening (1846) scruff (1595) relaxing (1371) tightly (1269) magically (1240) talon (1147) malemute (1115) sinewy (1081) wrist (1056) strangling (1053) squirming (981) loosen (959) convulsive (950) relax (928) norseman (907) relaxed (902) convulsively (870) bulldog (844)
hearth	ember (3254) cricket (2757) smoulder (2689) crackled (2001) diffusing (1832) peat (1703) untasted (1655) afire (1538) veined (1480) cinder (1450) rug (1445) smouldering (1365) acadian (1277) pestle (1275) fender (1214) chirp (1208) firelight (1196) slut (1160) blazing (1122) faggot (1039) saucepan (1034)
holiday	easter (1402) christmas (1234) holiday (1099) bur (1027) picker (970) schoolboy (949) outing (943) acadian (889) jocund (856) egregious (839) attire (798) lave (779) midsummer (764) jaunt (746) touchy (734) selectman (719) unmannerly (675) vacation (664) recuperate (661) teuton (645) polynesia (642)
keep	lookout (319) unspotted (315) aloof (277) inviolate (268) passover (267) donjon (263) commandment (249) tryst (222) tab (220) vigil (211) mum (202) afloat (198) undefiled (165) bosc (162) intact (160) religiously (153) vigilant (152) unleavened (151) assignation (147) repeating (143) tabernacles (142)
lady	gorgon (376) brunhild (332) viscountess (322) ladyship (287) helena (280) russell (280) abbess (220) patroness (215) dowager (214) blanch (213) gentlewoman (209) griffin (204) bevy (204) thrum (203) virgilia (201) curtsy (197) elderly (197) palfrey (195) kinswoman (186) silvia (177) chesterfield (169)
leave	unsay (302) untouched (195) lurch (187) unanswered (184) legacy (179) ajar (179) petiole (175) untried (169) unfinished (166) unburied (144) unprotected (138) unsolved (134) rustling (131) unmolested (126) unexplained (124) stranded (123) rustle (120) indelible (119) uncut (118) aspen (118) footmark (117)
live	virtuously (302) respectably (293) happily (236) wherewithal (227) vegetate (208) consultant (202) prosperously (191) peaceably (190) pittance (185) economically (183) righteously (182) cheaply (174) comfortably (158) anchorite (152) retirement (150) silurian (149) widowed (137) elysium (131) die (131) contentedly (125) realist (124)
local	option (2799) centralized (2106) fbi (2078) switchboard (1677) global (1394) bbs (1385) locally (1024) celebrity (1023) magnate (956) relay (950) centralization (950) quarantine (929) electronics (872) leftist (865) transmitter (808) federation (794) federal (769) manicure (756) personnel (741) stadium (739) modem (738)
match	tinder (2107) boxing (1756) wrestling (1654) phosphorus (1582) lucifer (1557) kerosene (1272) ignited (1150) football (1111) cricket (979) mugger (975) tennis (833) fumble (815) flared (808) cigarette (793) blanch (782) baseball (719) wrangler (700) championship (688) flare (677) sulphur (637) brimstone (609)

Continúa en la siguiente página

Cuadro C.1 – continuado de la página anterior

palabra	vecindad de palabras
material	inflammable (1799) raw (1610) combustible (1474) immaterial (1263) porous (883) subsection (806) inorganic (779) unpromising (763) workmanship (651) multi (649) collecting (610) floppy (566) footmark (541) collate (539) upholster (538) compilation (526) plastic (522) manufacture (505) durable (502) cheapness (485) header (484)
mouth	agape (1647) toothless (891) puckered (860) rinse (839) gag (780) naris (691) gaping (656) toothpick (538) nostrils (518) suckling (515) gape (465) twitch (463) orifice (456) aquiline (438) yellowstone (415) wry (411) foaming (411) distended (410) spoonful (400) screwing (384) estuary (383)
nation	civilized (659) afro (593) ish (572) germanic (571) barbarous (472) solidarity (459) barbarism (429) democratic (399) ploughshare (382) maritime (380) disarmament (368) teutonic (335) jurisprudence (319) pontus (319) engross (319) autocracy (313) byword (308) territorial (305) redound (304) jewish (301) centralized (301)
natural	selection (1798) aptitude (583) genealogical (547) artificial (483) causation (478) habitat (467) disuse (454) classification (453) explicable (439) divergence (430) transitional (402) botany (388) concomitant (388) variability (369) ephemeris (367) zoology (364) sterility (358) supernatural (358) recapitulation (346) variation (341) modification (340)
nature	implanted (478) chatterer (390) inorganic (372) niggard (356) jumper (350) freak (320) dual (317) vacuum (313) immutable (312) conditioned (312) domestication (298) human (280) inanimate (278) chipmunk (272) organic (271) inherent (270) sentient (251) endowed (245) elemental (238) personify (234) impressionable (233)
oblique	longitudinal (10703) beady (7161) oblique (7144) ocellus (6266) zodiac (5402) rectangular (4876) curdling (4476) refraction (4392) hunched (4331) cabaret (3664) radial (3643) nib (3405) median (3263) zoroaster (3246) kor (3071) begrudge (3041) cuckold (2955) pigtail (2946) stripes (2734) morale (2655) fold (2522)
play	cliche (1027) billiards (1016) chess (923) backgammon (905) collaborate (842) croquet (814) sonata (774) role (720) whist (669) epilogue (650) ibsen (648) golf (645) tennis (635) banjo (635) chopin (606) dramatist (562) quince (552) eavesdropper (549) fiddle (545) prank (499) pianist (495)
post	lucrative (1003) sentinel (906) sinecure (822) trading (761) placard (672) houston (636) lintel (615) sentry (596) postal (585) detroit (557) santee (534) postmark (525) seattle (520) georgetown (518) posting (511) minneapolis (494) fbi (479) postmaster (447) chaise (445) mailing (440) arkansas (422)
pull	trigger (3171) bobbin (993) forelock (903) centrifugal (865) stunt (791) oar (708) jiffy (700) scull (699) gaff (688) halyard (685) pull (674) ransack (645) pu (605) dinghy (596) gravitation (584) woozy (574) mote (558) limber (555) reliant (553) acheron (516) pristine (497)
replace	comma (2389) flooring (1358) processor (1204) periodic (1039) oilcloth (897) tab (862) teepee (820) tights (813) literacy (792) tahitian (775) stopper (766) integrated (763) loam (759) hermetically (755) affability (739) processing (734) implementation (732) revolutionize (723) rinse (696) vial (693) coverlet (682)

Continúa en la siguiente página

Cuadro C.1 – continuado de la página anterior

palabra	vecindad de palabras
restraint	durance (2317) imposed (2062) sisterhood (1823) immoderate (1635) retention (1368) salutary (1308) licentiousness (1297) emancipated (1271) conventionality (1196) impose (1181) restraint (1140) importation (1125) chafed (1097) enchant (1092) censorship (1074) satiate (1032) shackle (967) coupling (933) hyena (926) inhibition (923) naively (918)
sense	insecurity (979) unreality (671) literal (640) mainstream (564) unfitness (489) unworthiness (480) fitness (468) figurative (414) humor (414) proprietorship (410) nearness (394) suffocation (377) loneliness (366) helplessness (364) inverse (341) decency (330) aesthetic (329) inferiority (325) pervading (311) inadequacy (309) propriety (304)
serve	apprenticeship (1412) mammon (747) faithfully (577) sauce (482) loyally (466) truffle (418) tablespoonful (400) dessert (394) parsley (380) viands (376) garnish (373) gravy (363) waitress (333) collation (331) deepen (315) icelander (313) tomato (295) juror (295) accentuate (294) teaspoonful (293) omelet (281)
simple	straightforward (746) inexpensive (580) complex (569) unsophisticated (536) rugby (511) artless (477) unaffected (443) directness (417) elemental (392) unteach (387) guileless (374) unlettered (373) arithmetical (344) complicated (344) ocellus (340) unassuming (315) elementary (313) trustful (311) sophisticated (305) cellular (302) untutored (298)
solemn	conclave (1016) chanting (933) requiem (830) ceremonious (802) dirge (709) oath (633) dong (621) baseless (621) chorister (591) impressive (579) ceremonial (561) knell (559) rite (539) protestation (538) choral (532) chant (530) procession (528) festival (517) disembody (508) unleavened (505) oracular (503)
spade	mattock (15590) pickaxe (12514) ace (7813) spade (5638) ploughshare (5247) trowel (4276) hoe (3885) shovel (3220) dig (3021) wheelbarrow (2953) frier (2389) pruning (2227) plow (2064) sexton (1916) barrow (1786) rake (1784) digger (1708) scythe (1680) plough (1604) solstice (1579) sickle (1549)
stress	compression (4096) billed (2540) enunciation (1980) pathological (1478) syllable (1269) purvey (1188) frontiersman (1178) unassisted (1077) disarmament (1065) cark (1016) misrule (1002) carmelite (977) outlawry (971) bonus (953) epsilon (913) stress (900) senile (892) decreasing (889) revulsion (885) veda (867) recounting (851)
strike	clocks (649) chord (603) keynote (528) parallelism (473) forcibly (430) striker (400) gong (349) pickaxe (327) tinder (327) fist (315) thunderbolt (298) doodle (290) pullman (278) blow (271) lightning (265) dumb (254) cobra (253) clock (248) teamster (245) unpleasantly (239) rebound (239)
train	subway (988) pullman (932) siding (932) locomotive (913) depot (744) freight (725) conductor (709) omaha (692) station (690) compartment (648) railway (604) platform (569) brakes (559) titania (542) steamed (518) trestle (508) platforms (503) utica (493) junction (492) rails (488) tidal (468)
treat	disrespect (914) soiree (896) regale (773) ordinate (696) rightfully (659) civilly (601) unfairly (570) unkindly (552) outrageously (532) contumely (530) harshly (499) explicitly (488) humorously (483) treat (475) brutally (467) treated (462) wesleyan (452) depute (447) empowered (434) decently (421) burma (421)

Continúa en la siguiente página

Cuadro C.1 – continuado de la página anterior

palabra	vecindad de palabras
turn	grindstone (488) somersault (454) pivot (385) upside (378) twist (235) monger (225) winch (212) wheel (203) axis (191) tubal (183) ploughshare (183) tapster (174) modulated (171) gryphon (164) bight (161) lathe (157) dally (153) reliant (153) weathercock (148) meandering (146) capstan (146)
use	commercially (2254) prohibited (1193) distributed (718) disuse (708) distribution (564) mailbox (438) commercial (427) mainstream (373) pronoun (373) binary (335) acronym (291) update (283) micro (274) modem (268) ftp (268) graphics (264) appropriated (258) slang (258) processor (258) plural (255) format (253)
vital	cohesion (1534) constituency (1242) perpetuity (1181) druid (1123) assimilation (1121) nutriment (1116) engulf (1038) appreciable (1014) digress (989) organs (975) vital (972) inorganic (902) radiation (867) phantasy (852) currently (852) entente (845) thyroid (824) causation (785) atrophy (780) organism (776) importance (763)
wander	aimlessly (3626) restlessly (1051) disconsolately (869) bottomed (852) pathless (686) proserpine (642) byway (620) streamlet (611) irresolutely (576) maze (566) labyrinth (550) homeless (548) purposeless (547) disconsolate (521) eventide (509) flowery (507) afield (502) andalusia (489) brier (489) trackless (464) briny (454)
wash	ewer (4358) pock (2519) rinse (2032) scour (1753) unclean (1572) soap (1521) cleanse (1382) foulness (1372) towel (1236) bathe (1202) lethe (1172) ironing (1163) ablution (1143) fulsome (1113) tub (1053) anoint (1009) basin (872) dish (854) doomsday (853) squirt (850) scupper (850)
work	havoc (244) electronics (238) crochet (219) drudgery (212) shirk (204) joiner (204) subsection (189) laundry (186) artificer (186) journeyman (182) ironing (179) foundry (177) thoroughness (176) harmoniously (176) lathe (175) michelangelo (170) completion (168) uphill (164) miracle (162) experimental (161) efficiently (159)
yew	cypress (12695) shooter (10846) yew (9352) juniper (7767) sliver (7404) alley (5006) aspen (4831) clip (4697) dryad (4577) hedge (3697) variegated (3612) nook (3570) yolk (3428) faerie (3407) churchyard (3402) sonny (3283) spenser (3234) chrysanthemum (3203) fir (3138) arbour (3079) bowstring (2996)

C.2. χ^2

Cuadro C.2: Palabras más asociadas con respecto a una muestra de palabras según la medida χ^2

palabra	vecindad de palabras
art	thou (27876326) art (1678962) literature (518831) science (429678) <NUMBER> (427302) artist (333471) <PROPER_NOUN> (326771) works (323251) say (319701) wilt (310466) nature (304253) man (230313) painting (207721) god (203718) work (174899) lord (162135) healing (156331) make (155631) learn (149129) know (137855) life (136883)
authority	government (442232) exercise (395075) supreme (310251) federal (290045) power (276589) legislative (249501) civil (243819) <NUMBER> (236534) vested (227588) authority (225747) municipal (212650) local (201607) state (198292) ecclesiastical (184878) united (168075) executive (164087) military (150961) <PROPER_NOUN> (150231) sovereign (127786) church (126680) judicial (124683)
bar	iron (1467428) window (332361) bolt (309943) cage (276602) door (243841) bar (243141) gate (205132) <NUMBER> (195982) harbor (123727) capstan (112828) <PROPER_NOUN> (111856) prison (89388) way (85035) grate (63759) wall (63142) stand (58638) lock (56372) socket (54693) prisoner (54117) shutter (49967) grating (49937)
begin	<PROPER_NOUN> (1500448) <NUMBER> (1214464) say (772402) soon (620821) end (600998) time (552412) think (498200) come (496035) feel (484635) make (437540) little (405556) man (392605) grow (356921) work (351451) talk (319599) look (311612) just (305336) know (298072) day (287946) long (273328) new (271853)
blind	deaf (2610790) lame (545451) blind (449563) dumb (350256) man (225704) eye (201885) window (160064) alley (136900) maimed (125182) venetian (114174) buff (104370) <NUMBER> (101300) beggar (100389) <PROPER_NOUN> (86283) blindness (72734) groping (70160) fury (68001) unreasoning (67729) grope (58499) lead (57125) sight (55102)
bum	bum (2576094) tapster (996745) disruption (124514) pompey (94002) topple (77348) program (65191) troth (62040) booze (54500) meter (48701) efficient (44336) code (37828) hardware (36994) med (36902) synonym (30253) beastly (29249) beck (28746) tune (25489) algorithm (25223) gauntlets (23040) dime (20890) configuration (19286)
call	<PROPER_NOUN> (985146) say (514922) <NUMBER> (494834) call (387079) man (318606) come (263684) shall (247834) know (222472) names (204447) make (202645) don (194806) hear (188785) thou (173091) like (152462) good (146109) think (143468) let (141998) people (140762) phone (137450) attention (134882) time (134844)
carry	away (1004321) <NUMBER> (836838) <PROPER_NOUN> (551773) man (338920) far (309384) basket (253234) bag (206569) plan (202455) arms (196879) say (193476) make (178565) trade (163224) burden (162020) little (160639) execution (159913) bundle (150676) home (149973) come (148536) great (144796) time (144174) hand (144059)

Continúa en la siguiente página

Cuadro C.2 – continuado de la página anterior

palabra	vecindad de palabras
chair	sit (3406687) table (2193720) lean (1265280) seated (1108048) sink (726860) room (561772) <PROPER_NOUN> (533274) rise (328605) wicker (290358) floor (286616) placed (280774) sofa (279279) stand (264599) say (262100) furniture (258596) draw (256866) desk (252464) wheeled (245173) push (224916) drop (208796) arm (207063)
channel	narrow (303212) flow (247437) river (234270) stream (189826) dredge (183854) island (162100) islet (147001) channel (140431) water (139018) reef (117008) shoal (116963) tortuous (113688) buoy (109812) bayou (91717) bristol (86395) waters (84529) diverted (84370) current (83330) sea (80139) shallow (78687) coast (76649)
child	israel (2234889) mother (1961926) <NUMBER> (1943583) wife (1515093) woman (1420955) parent (1352107) child (1173126) <PROPER_NOUN> (1102560) little (1021085) father (963547) say (883715) poor (683209) like (553034) man (510065) bear (456465) dear (437814) come (430816) old (380012) know (340249) young (335852) nurse (330123)
church	catholic (2453397) church (1113067) methodist (744481) sunday (740593) presbyterian (719080) christ (685841) parish (666658) episcopal (631082) steeple (577972) <NUMBER> (560036) holy (491664) spire (471250) christian (428022) <PROPER_NOUN> (427643) protestant (422023) anglican (414785) pastor (414429) england (388566) congregational (364646) bishop (353883) congregation (349947)
circuit	node (1623333) circuit (584205) mac (566779) relay (277409) testing (182344) court (168103) judges (144091) current (136039) transmitter (135790) battery (119586) make (108198) wire (106941) carbon (102421) maintenance (92091) mile (77913) id (74324) <NUMBER> (66624) integrated (66410) wide (55697) installation (54417) edison (52044)
collaborate	fabrication (158794) materialize (122448) melodrama (99072) hartford (71354) play (29442) fashions (27481) dickens (21207) novelist (18111) maidenhood (17094) academic (17015) taxing (16797) novel (16017) eventual (15833) disseminate (15640) inexhaustible (14727) domineering (14689) descartes (14633) philology (14578) festal (14415) civilian (14317) lapel (13507)
colourless	liquid (128678) boiling (39221) face (27088) prism (26099) ammonia (25003) crystallize (24945) complexion (24811) daub (24333) sulphuric (22857) chloride (22724) grey (22136) lip (21920) transparent (21762) centred (20745) moustache (20743) crystal (20481) bladed (19726) alcohol (19546) oxide (18855) pungent (18586) eye (16092)
cool	breeze (427647) hot (392382) air (278726) refreshing (212385) shady (210983) deliciously (191181) water (166936) shade (145154) cucumber (142442) fragrant (99676) warm (87627) evening (85488) summer (84265) fresh (84094) green (83369) calm (82234) wind (73050) <PROPER_NOUN> (72112) keep (68807) night (65603) sweet (64593)
day	<NUMBER> (4273561) night (3279457) <PROPER_NOUN> (2053083) come (1519331) day (1121302) say (835750) following (774398) time (714695) shall (672540) morning (638323) make (612914) spend (600592) dawn (581764) work (581567) long (573265) evening (526158) man (522081) week (494264) hour (485667) hours (483011) tomorrow (463206)

Continúa en la siguiente página

Cuadro C.2 – continuado de la página anterior

palabra	vecindad de palabras
detention	corrective (1743305) segregation (211880) preventive (112409) illegal (110083) imprisonment (83032) forcible (55922) quarantine (55810) prison (52023) detention (51322) clamorous (41410) lunatic (35014) abduction (33915) forfeiture (27240) penitentiary (24959) damages (23714) punishment (21424) seizure (20253) asylum (19553) arrest (19393) chattel (18939) bennie (18643)
develop	concentration (52902) resource (51931) power (46744) develop (40898) tend (37666) germ (31957) telecommunication (31381) character (29604) faculty (28855) type (28296) social (28234) latent (27873) talent (27412) development (26822) developed (26737) industry (25021) begin (24559) utilize (23759) tendency (23272) quality (22929) spontaneously (21940)
draw	sword (1637891) near (1595394) breath (1491409) nearer (1452065) curtain (1085431) <PROPER_NOUN> (730492) <NUMBER> (613667) closer (598075) pocket (566116) inference (539595) nigh (521244) hand (417517) say (373706) conclusion (372105) line (355789) close (305663) man (272093) aside (263738) chair (256866) eye (253727) face (247387)
dress	wear (1887035) silk (858641) costume (359834) muslin (355716) white (353606) lace (341860) hair (330944) clothe (312292) black (278115) dress (245234) satin (227939) gingham (220545) calico (216243) evening (216175) woman (207699) skirt (202696) <PROPER_NOUN> (193211) hat (184560) lady (160765) flounce (159749) blue (157302)
drift	snow (309514) drift (222176) current (199488) wind (104177) cloud (83163) tide (81573) boat (72581) sea (64813) away (64711) river (62323) lift (59638) aeroplane (56540) raft (53600) ice (51846) shore (49076) electronics (43256) <PROPER_NOUN> (43199) smoke (42391) canoe (38656) surface (38462) incidence (38126)
drive	carriage (1301642) away (762738) <PROPER_NOUN> (534602) coachman (486807) cab (463078) horse (429389) <NUMBER> (394076) buggy (324316) cattle (310208) home (285045) cart (225011) mad (222674) wagon (220484) car (207838) wind (194826) man (193673) road (186615) drive (184611) hansom (170257) gig (164512) station (150045)
dyke	dyke (658368) van (497885) ogle (72077) tam (61664) derrick (47041) henry (36504) porphyry (33881) fen (29253) outlawry (26008) ditch (25289) inn (24754) witchery (24419) quartz (19491) causeway (19166) sandstone (19132) engineer (18136) dam (17754) dune (16845) <PROPER_NOUN> (15833) ranch (14375) freight (14250)
face	eye (3860855) look (3739293) expression (2511280) pale (2216675) <PROPER_NOUN> (1586983) turned (1495153) hair (1420626) smile (1369116) flushed (1140292) hands (1039502) say (985816) white (962536) face (799474) looking (776214) man (764287) light (680849) like (633186) hide (593543) <NUMBER> (592354) stand (581338) lip (565534)
facility	afford (157733) transportation (120754) communication (87648) milieu (78383) facility (65279) acquiring (54662) reciprocal (50737) greater (49038) unequalled (40179) equal (37739) acquired (35853) incidental (33046) hybrid (30524) hollander (29658) research (26751) locomotion (25031) species (23908) storage (23861) reproductive (23650) transport (23585) penetration (22726)

Continúa en la siguiente página

Cuadro C.2 – continuado de la página anterior

palabra	vecindad de palabras
faithful	servant (752678) commander (284518) friend (219643) <PROPER_NOUN> (131474) <NUMBER> (129148) follower (91596) remain (89405) faithful (82048) adherent (79615) caliph (79367) unfaithful (77464) followers (71245) true (66175) love (57123) god (55757) reward (53991) service (53841) servitor (50096) duty (46606) master (46113) devoted (44966)
fatigue	hunger (374298) journey (237568) exhausted (231369) sleep (152631) overcome (125835) exertion (108117) thirst (106814) privation (96718) endure (87412) undergo (85511) bodily (80412) asleep (70766) rest (67652) day (55337) weariness (52702) night (52216) excitement (52133) repose (51143) wear (49932) hours (45242) exhaustion (42811)
feeling	feeling (60539) recognition (15589) elementary (10194) destination (9442) <PROPER_NOUN> (9037) sure (9010) snipe (8555) uneasy (8263) osier (7933) tripe (7815) effrontery (7426) squirt (7015) limbed (6927) caress (6716) finland (6415) acutely (5552) haystack (5517) spiritless (5460) gesticulate (5312) stoicism (5173) feel (5098)
ferret	weasel (331551) hutch (122014) ferret (115815) rabbit (83239) coney (37712) simmer (32424) ping (27315) burrow (25715) yap (25696) impale (24429) mink (23597) punk (22510) otter (18828) chide (17706) cicero (17416) curfew (17323) reproductive (16673) incitement (16578) chateaubriand (16401) tee (15340) unwarrantable (15040)
find	<PROPER_NOUN> (2344989) <NUMBER> (2304422) man (910085) say (901485) way (892325) come (693498) search (687139) place (648085) time (614506) make (559498) seek (532082) know (526478) little (517237) shall (475428) difficult (471401) soon (468613) think (467979) ain (447205) try (412429) house (410723) great (410549)
fine	weather (689513) linen (630382) <NUMBER> (415392) <PROPER_NOUN> (349797) fine (271748) arts (270655) fellow (248715) say (246362) clothe (213588) make (176999) man (166322) imprisonment (163850) gold (151329) old (149391) day (144892) like (138226) hair (135021) handsome (126561) flour (123846) gentleman (120945) young (117319)
fit	survival (367232) coughing (348974) think (213717) laughter (195849) <PROPER_NOUN> (192269) <NUMBER> (183769) fit (161173) epileptic (149902) man (137900) make (135096) say (115398) apoplectic (98396) horsepower (94300) place (81836) violent (70957) burst (70870) time (65801) find (61734) like (59035) work (57343) ague (53418)
free	slave (725050) set (532718) free (473420) <NUMBER> (268273) man (266777) slavery (255678) <PROPER_NOUN> (206804) state (196916) freedom (187236) liberty (186996) make (174245) independent (146728) leave (141501) government (133225) shall (120360) people (118812) say (114545) bondage (112686) access (111610) bond (103585) life (101764)
graceful	figure (396184) curve (317637) slender (310316) lithe (207549) tall (204319) beautiful (120081) elegant (114072) movement (92778) shapely (84807) attitude (72848) outline (70398) delicate (70116) handsome (65597) contour (63583) form (62399) exquisitely (61360) slim (57842) manners (53813) charming (53107) supple (47494) manner (46117)

Continúa en la siguiente página

Cuadro C.2 – continuado de la página anterior

palabra	vecindad de palabras
green	grass (1362662) baize (1120609) blue (957240) yellow (885914) tree (866064) meadow (557218) foliage (534911) white (530869) green (397831) bowling (381230) moss (364490) emerald (342245) turf (331459) bright (296392) flower (291802) field (289515) purple (283423) bough (274925) willow (256285) sward (254970) hills (253107)
grip	tightened (425010) hand (313160) tightly (271208) arm (256212) wrist (252114) throat (242977) tight (182141) hands (161148) hold (150530) grip (123057) finger (117659) vise (106648) firm (102854) relaxed (102652) tightening (88518) clutch (78999) shoulder (76366) racquet (73944) iron (58388) <PROPER_NOUN> (54990) wrench (53039)
hearth	afire (740579) ember (523634) cricket (402327) ash (199473) log (197390) chimney (179664) rug (171757) burn (164276) blaze (151896) blazing (150196) sit (148978) chair (112069) fireplace (90645) smoulder (69886) cinder (69512) glow (62632) wood (62541) light (59986) grate (57412) warm (54470) crackled (53976)
holiday	christmas (607669) holiday (360148) easter (173694) spend (127368) summer (79785) schoolboy (79602) attire (74077) school (65743) home (65427) day (57879) <PROPER_NOUN> (54754) week (50130) festival (41834) sunday (40763) <NUMBER> (37858) vacation (37082) enjoy (35408) midsummer (35080) work (32821) time (32266) boy (30615)
keep	<PROPER_NOUN> (1148101) secret (1042423) <NUMBER> (994138) say (738520) watch (533312) eye (526716) time (482455) man (437311) quiet (435890) alive (407576) make (405746) commandment (384765) away (384731) know (380146) promise (367053) till (361194) long (338602) good (325868) lookout (322966) little (298652) house (292678)
lady	young (4673156) <PROPER_NOUN> (2923422) say (1499755) gentleman (946868) old (941332) lady (935733) mrs (789128) miss (644718) sir (572342) dear (558851) <NUMBER> (531920) lord (476059) come (457010) helena (414243) mr (413920) know (410420) fair (362724) ladyship (341802) daughter (327790) think (323201) tell (321794)
leave	<PROPER_NOUN> (2564274) <NUMBER> (1911800) right (1491237) room (1294868) say (1136947) house (735092) come (734166) hand (702392) away (632825) man (543512) time (541165) little (534166) home (519665) make (507227) door (433118) return (416572) place (409462) day (404244) find (392153) think (378502) soon (377162)
live	die (1467613) life (1116885) <NUMBER> (898880) years (883100) long (831652) <PROPER_NOUN> (712804) man (586768) live (580078) house (520426) say (384421) shall (380847) know (357098) people (331172) old (299615) world (294185) time (291356) happily (277946) like (253515) come (243489) think (203267) age (198868)
local	option (472860) local (365716) federal (204966) authority (201607) government (167784) national (164021) administration (122313) fbi (118346) celebrity (110356) phone (100473) police (93327) network (89411) state (88573) centralized (84184) municipal (83909) <NUMBER> (83727) organization (77746) central (76578) district (70923) area (68676) committee (53117)

Continúa en la siguiente página

Cuadro C.2 – continuado de la página anterior

palabra	vecindad de palabras
match	light (383300) match (356131) strike (312512) cigarette (288142) wrestling (266018) box (155759) lucifer (155533) tinder (147375) candle (145128) cricket (135820) <PROPER_NOUN> (120760) boxing (119292) pipe (115222) football (99911) cigar (95976) tennis (91506) make (83182) <NUMBER> (75201) phosphorus (61630) flame (57854) pocket (57062)
material	raw (1488427) spiritual (289599) supply (178374) manufacture (166265) inflammable (159999) collecting (143666) material (141786) work (138622) immaterial (137554) furnish (135710) build (129108) moral (117736) combustible (110492) prosperity (109564) furnished (109279) make (104066) <NUMBER> (97190) use (92312) form (86710) substance (85774) product (78269)
mouth	open (989291) eye (770639) nose (672579) opened (603986) tooth (483171) pipe (431353) <NUMBER> (410894) lip (409855) tongue (356811) chin (338610) corner (337149) river (334277) word (320786) wide (301916) shut (293816) nostrils (285300) cave (223325) gaping (219976) face (212020) agape (210726) mouth (202805)
nation	nation (729414) <NUMBER> (546865) civilized (487054) people (343808) war (340661) foreign (275491) great (258575) government (253311) europe (239012) shall (200219) barbarous (189541) democratic (181518) european (165013) national (161401) god (152952) country (148562) individual (143949) united (140490) power (139268) earth (135075) political (132082)
natural	selection (8700032) history (777724) artificial (345659) science (306381) species (295533) instinct (268355) philosophy (240180) variation (214242) phenomenon (202553) <NUMBER> (197150) man (177027) natural (171587) simple (142933) nature (140665) <PROPER_NOUN> (138297) perfectly (137077) theory (128956) modification (126347) tendency (125877) law (122568) supernatural (120204)
nature	human (3176505) man (529709) laws (366721) nature (352047) art (304253) <NUMBER> (298956) things (252279) life (239383) make (219257) law (214486) <PROPER_NOUN> (188749) power (183157) good (178876) true (160522) divine (152745) beauty (150553) love (144493) know (144285) knowledge (143897) god (143304) natural (140665)
oblique	longitudinal (235435) fold (196578) oblique (192862) eyebrow (143319) slightly (89660) stripes (73771) beady (57277) zodiac (54001) ray (49457) rectangular (38997) ocellus (37586) glance (32265) horizontal (30077) direction (29651) line (28097) refraction (26342) hunched (25977) perpendicular (24438) cuckold (23631) slight (23400) furrow (22717)
play	game (2046560) play (615465) trick (510993) cards (487158) <PROPER_NOUN> (460945) role (433182) piano (376875) billiards (334025) actor (283198) chess (272877) theatre (269143) tune (262228) music (248484) <NUMBER> (245557) played (239981) tennis (217520) fiddle (209867) violin (205946) child (197932) say (193797) player (188107)
post	office (526714) letter (450336) sentinel (354834) <NUMBER> (256232) <PROPER_NOUN> (225035) post (223556) trading (184569) lucrative (121205) sentry (114131) morning (89035) send (88527) write (82552) fort (68883) guard (63190) beat (58077) military (57208) placard (55669) come (53752) occupy (51294) return (49403) man (47353)

Continúa en la siguiente página

Cuadro C.2 – continuado de la página anterior

palabra	vecindad de palabras
pull	trigger (830477) pull (362027) oar (194996) rope (128204) try (72791) <PROPER_NOUN> (64552) boat (59405) string (53097) <NUMBER> (50698) say (49024) pulling (47386) don (42122) away (40320) tug (40153) tooth (36375) hard (34846) pulled (34499) cord (34431) bell (32990) man (32921) let (31926)
replace	article (98131) <NUMBER> (59648) comma (57305) carefully (49499) literacy (45044) new (39712) pocket (39448) flooring (36641) removed (36579) replace (36318) <PROPER_NOUN> (29429) table (25673) following (24092) capital (23415) box (21035) hydrogen (17445) receiver (17326) envelope (16491) sheath (16356) coverlet (16326) old (15923)
restraint	imposed (482268) restraint (142341) impose (93193) free (58775) freedom (55519) salutary (49639) durance (41671) sisterhood (36439) liberty (35937) discipline (33719) emancipated (30475) importation (29213) impatience (27118) immoderate (22871) power (22076) impatient (22022) chafed (21916) conventional (21104) presence (20820) self (20500) constitutional (20052)
sense	common (1538670) humor (538037) good (335347) feel (323420) word (304182) humour (284558) duty (271738) moral (271471) man (226200) literal (205688) <PROPER_NOUN> (205625) sense (205602) responsibility (192628) term (190004) <NUMBER> (175338) keen (168326) loneliness (152618) propriety (144593) life (124288) make (121394) vague (120824)
serve	purpose (561351) apprenticeship (486838) <NUMBER> (375260) faithfully (344640) serve (226981) <PROPER_NOUN> (193877) dinner (170295) god (165179) dish (135526) man (131049) meal (110798) sauce (108559) table (104091) say (99054) shall (98152) lord (95090) years (92132) make (91378) king (82462) thou (82292) master (81470)
simple	complex (278243) pure (191000) straightforward (165282) <NUMBER> (160780) <PROPER_NOUN> (149270) natural (142933) fact (130068) simple (126192) life (125551) man (102010) plain (98845) form (98726) complicated (96308) make (90379) truth (86819) nature (85938) word (83275) easy (82109) say (82092) explanation (81707) process (80940)
solemn	oath (473914) procession (181213) vow (180337) silence (107783) feast (107326) festival (99532) rite (93016) impressive (87731) promise (84566) ceremony (83607) tone (81064) voice (73620) chanting (72663) covenant (66769) stillness (62397) <PROPER_NOUN> (60455) <NUMBER> (51225) face (48473) conclave (46659) slow (44621) make (44542)
spade	dig (1307629) mattock (1091237) spade (1031444) pickaxe (875882) ace (687463) hoe (302942) shovel (286500) rake (115870) plough (86539) ploughshare (83922) trowel (76939) ax (69942) pick (68198) implement (66887) gardener (59831) wheelbarrow (59026) plow (57749) sexton (53604) barrow (49954) garden (47882) scythe (43643)
stress	compression (163789) syllable (93784) strain (79946) lie (68441) weather (68431) storm (54248) stress (50299) accent (41424) great (36645) mental (34699) pathological (25098) circumstances (24307) enunciation (23737) word (23691) especial (22858) emotion (18728) fact (18576) tension (17918) financial (16850) particular (16271) pitch (15718)

Continúa en la siguiente página

Cuadro C.2 – continuado de la página anterior

palabra	vecindad de palabras
strike	blow (1708877) clock (1093595) <NUMBER> (501487) <PROPER_NOUN> (436247) match (312512) fist (312190) chord (301923) lightning (287137) clocks (212633) man (210241) strike (204241) head (202956) dumb (194007) contrast (189647) say (189062) forcibly (185148) fall (180108) hand (176925) bullet (158280) face (145774) time (145751)
train	station (2021118) railway (614954) platform (454249) car (398989) passenger (387040) train (371212) conductor (273201) locomotive (264470) freight (252732) <PROPER_NOUN> (239772) railroad (239091) <NUMBER> (216384) stop (195461) start (187631) depot (171504) catch (160309) clock (150851) wagon (150088) leave (148430) compartment (147396) morning (126413)
treat	treated (265444) treat (255034) respect (101126) subject (100892) like (69381) contempt (66068) <PROPER_NOUN> (64971) way (62160) say (57366) man (50814) tome (49954) kindly (49540) manner (49390) <NUMBER> (42391) disrespect (40136) harshly (37318) lightly (33562) friend (33550) badly (31144) kindness (30558) child (30390)
turn	<PROPER_NOUN> (453875) <NUMBER> (387263) round (302580) say (302071) away (300199) make (228253) come (214349) man (188892) aside (187541) head (176013) eye (174543) wheel (170200) way (162973) look (152280) face (148821) turn (145132) know (131763) good (130763) shall (126105) let (121439) like (116917)
use	commercially (3237229) prohibited (1828067) distributed (1580028) make (1059226) distribution (855634) commercial (698852) <NUMBER> (480664) word (381839) language (373132) <PROPER_NOUN> (304196) used (268121) say (262453) personal (261971) maybe (226947) use (223457) term (219001) disuse (217774) weapon (212927) mean (185595) phrase (175643) man (172899)
vital	importance (346582) vital (133993) organs (119791) force (86923) spark (73827) energy (72620) principle (67414) life (43828) organism (41830) point (36921) factor (33453) question (30665) function (27682) power (26100) element (24845) apart (24156) chemical (21597) problem (20865) appreciable (20254) statistics (20063) human (19322)
wander	aimlessly (884570) eye (157003) restlessly (104979) away (82600) far (81149) street (71855) night (66764) garden (60343) wander (60276) wilderness (59418) lose (58371) woods (56142) <PROPER_NOUN> (53177) search (49352) find (46536) path (46422) maze (45755) seek (43706) <NUMBER> (43612) till (39498) restless (38401)
wash	water (480743) clean (358755) hands (327271) dish (318740) unclean (266975) soap (259870) ewer (243967) wash (243166) clothe (153986) basin (153237) washed (146036) towel (133385) bathe (120038) stain (102245) tub (99879) anoint (90706) blood (87368) linen (85709) dirty (71906) scour (70053) cleanse (69031)
work	hard (1629143) <NUMBER> (1296073) <PROPER_NOUN> (953906) work (717467) man (645528) day (581567) set (523427) make (515525) time (410699) beat (364292) begin (351451) say (349762) great (328123) good (318117) miracle (261987) finished (252939) find (250830) life (240789) know (223873) think (220571) come (217436)

Continúa en la siguiente página

Cuadro C.2 – continuado de la página anterior

palabra	vecindad de palabras
yew	cypress (787037) alley (480485) hedge (428667) yew (420790) tree (251290) clip (244153) churchyard (176816) shooter (162670) nook (149893) bow (139542) juniper (124242) fir (103493) holly (74143) aspen (72449) sliver (59221) variegated (50552) spenser (42023) arbour (36931) garden (33411) lawn (28417) elm (28026)

C.3. Medida binomial

Cuadro C.3: Palabras más asociadas con respecto a una muestra de palabras según la medida Binomial de Dunning

palabra	vecindad de palabras
art	thou (600718) <NUMBER> (118145) <PROPER_NOUN> (91708) say (68094) art (62411) man (46063) make (32716) know (29320) god (25112) nature (24787) come (23802) life (22313) work (21866) great (21619) lord (20726) science (18594) shall (18450) good (18321) literature (17949) love (17541) works (16505)
authority	<NUMBER> (63885) <PROPER_NOUN> (43085) government (19847) power (19178) state (15130) man (15085) make (14323) say (13608) great (12401) exercise (12366) people (12169) authority (10536) church (9640) law (9371) time (8914) united (8471) civil (8343) public (8322) subject (8262) supreme (8204) king (8090)
bar	<NUMBER> (49571) <PROPER_NOUN> (30707) iron (24616) door (16859) window (15194) man (10842) way (10810) say (10745) make (8981) gate (8720) come (8287) stand (7758) bar (7629) like (7034) little (5638) wall (5298) thea (5273) find (5249) bolt (5034) room (5002) time (4958)
begin	<PROPER_NOUN> (389740) <NUMBER> (349395) say (180917) come (104565) time (103769) make (97058) think (96192) man (95966) little (77723) know (72251) feel (71652) end (70202) soon (69274) look (61067) like (56346) tell (55597) day (54546) work (52354) just (51532) life (51491) long (50628)
blind	man (31314) <NUMBER> (29674) <PROPER_NOUN> (25038) eye (18934) deaf (16432) say (13288) blind (10746) know (9611) window (9500) make (9130) like (7814) old (7715) come (7124) think (6639) love (6630) light (6345) dumb (5887) look (5811) lame (5733) lead (5587) draw (5565)
bum	bum (1145) <NUMBER> (757) <PROPER_NOUN> (615) tapster (468) pompey (426) program (404) say (376) come (356) code (307) know (287) tell (269) tune (268) don (259) troth (258) just (253) efficient (246) instructions (233) like (218) make (212) ain (205) lot (189)
call	<PROPER_NOUN> (232910) <NUMBER> (145540) say (107466) man (65314) come (52702) know (46295) make (44456) shall (39299) call (33647) think (31639) like (31421) time (30422) hear (30323) don (29574) good (27921) thou (25374) mr (24193) tell (24152) people (23958) let (23798) little (22502)
carry	<NUMBER> (223029) <PROPER_NOUN> (154546) away (90630) man (70497) say (54891) make (42015) far (40675) come (36398) little (33028) time (32950) great (29082) hand (24816) like (24475) house (24226) leave (23412) know (22183) day (22127) home (21363) way (21185) think (20888) ain (20573)
chair	sit (127935) <PROPER_NOUN> (121008) table (65918) say (52945) <NUMBER> (51455) room (36518) lean (33616) stand (27324) seated (24720) sink (24720) look (24559) rise (24483) mr (22912) little (22400) draw (20478) old (19639) eye (19177) hand (19175) come (19118) face (17810) man (17637)
channel	<NUMBER> (17581) river (8478) water (7011) <PROPER_NOUN> (6463) narrow (6193) island (5205) stream (5203) flow (4675) sea (4588) run (4356) deep (3717) make (3613) mile (3245) time (2947) find (2932) ship (2716) bank (2693) english (2681) new (2676) waters (2646) shore (2643)

Continúa en la siguiente página

Cuadro C.3 – continuado de la página anterior

palabra	vecindad de palabras
child	<NUMBER> (483163) <PROPER_NOUN> (297125) say (190306) mother (142288) little (138449) woman (138117) wife (110833) man (110256) child (109924) father (100005) like (93389) come (90315) know (75703) israel (73259) make (73179) poor (68072) old (63979) think (56953) shall (54326) bear (53102) young (52203)
church	<NUMBER> (146433) <PROPER_NOUN> (113172) church (49598) catholic (36336) say (30486) england (24424) man (24004) old (23584) christ (23275) make (22993) holy (22624) sunday (22594) great (22327) come (21005) people (20280) build (19916) time (19422) christian (18670) god (18193) ain (17103) state (16873)
circuit	<NUMBER> (14746) make (10106) <PROPER_NOUN> (4784) court (4758) circuit (3005) mile (2947) node (2694) mac (2691) current (2294) judges (1939) wide (1917) round (1854) trouble (1619) wire (1609) wall (1590) judge (1534) battery (1426) long (1257) host (1250) town (1146) district (1143)
collaborate	play (416) <NUMBER> (387) write (224) hartford (186) begin (175) novel (150) story (150) work (141) friendship (140) fabrication (110) materialize (107) hard (105) melodrama (105) friend (100) agree (99) obscure (98) know (93) twain (93) fashions (89) dickens (86) professor (85)
colourless	face (1427) <NUMBER> (1234) eye (1148) <PROPER_NOUN> (763) lip (698) liquid (678) hair (581) little (512) water (432) grey (428) look (415) like (402) cold (371) long (368) man (361) white (346) light (333) life (330) clear (330) voice (327) stand (324)
cool	<PROPER_NOUN> (20076) air (12849) <NUMBER> (11752) hot (10134) say (9965) water (9781) come (7615) keep (7121) night (6877) little (6767) breeze (6549) man (5738) evening (5722) wind (5211) day (5134) sit (4914) feel (4910) look (4898) like (4814) eye (4808) make (4783)
day	<NUMBER> (1010161) <PROPER_NOUN> (543111) night (267584) come (256272) say (213477) day (155821) time (139827) make (138026) man (131398) shall (113879) know (95400) long (95046) think (86031) work (83561) little (80483) leave (79088) find (78606) tell (78310) morning (75870) great (73427) good (70092)
detention	<NUMBER> (1605) <PROPER_NOUN> (967) corrective (823) prison (777) house (682) imprisonment (488) palace (418) time (383) punishment (382) years (380) cause (375) sentence (371) illegal (371) place (361) make (348) person (302) great (296) say (289) segregation (288) arrest (274) long (267)
develop	<NUMBER> (4972) man (3014) power (2869) begin (2432) life (2399) <PROPER_NOUN> (2365) new (1970) make (1916) time (1885) character (1749) nature (1643) work (1544) form (1406) idea (1396) mind (1371) grow (1353) condition (1181) social (1169) human (1149) resource (1090) great (1075)
draw	<PROPER_NOUN> (192328) <NUMBER> (177332) say (89061) near (85662) sword (61974) man (60977) breath (54024) hand (51152) nearer (43104) come (42333) little (41960) make (41279) eye (39827) away (35890) face (35495) look (34712) time (33111) long (33028) curtain (31381) line (30643) head (30189)
dress	<PROPER_NOUN> (52128) wear (50165) <NUMBER> (36826) say (23231) white (22397) woman (21382) make (19880) look (17642) black (17378) hair (16821) like (15890) lady (15643) come (15570) silk (15133) little (14958) man (14692) clothe (14527) evening (14133) ain (12007) dress (11145) girl (10901)

Continúa en la siguiente página

Cuadro C.3 – continuado de la página anterior

palabra	vecindad de palabras
drift	<PROPER_NOUN> (12256) <NUMBER> (10926) snow (7200) away (6332) wind (5534) like (4470) man (4353) sea (4239) current (4204) river (3937) drift (3728) time (3633) far (3567) come (3541) boat (3474) cloud (3303) let (3186) little (3184) find (3113) make (2869) say (2774)
drive	<PROPER_NOUN> (134084) <NUMBER> (112106) away (63957) man (41279) carriage (38913) say (38155) horse (31999) come (31253) home (27621) make (22859) time (21821) house (21577) like (21041) day (19395) little (18055) wind (17340) think (17119) know (16365) leave (16082) drive (16026) way (15008)
dyke	<PROPER_NOUN> (3415) van (2676) <NUMBER> (1531) dyke (1124) henry (871) mrs (715) make (603) come (599) little (564) wall (538) water (518) man (483) inn (481) sea (449) stone (427) thea (389) house (389) run (371) great (371) know (365) way (363)
face	<PROPER_NOUN> (411121) look (329349) eye (308622) say (218188) <NUMBER> (204340) man (155675) turned (119983) come (111472) expression (110476) like (109331) hands (98657) face (98560) smile (95666) little (93660) pale (91394) hair (88348) white (86684) stand (80562) light (77436) know (75634) head (73448)
facility	<NUMBER> (4022) afford (2756) <PROPER_NOUN> (2199) great (1843) greater (1816) make (1800) communication (1453) equal (1306) country (1215) man (1158) new (1125) offer (1044) ain (993) species (972) language (964) time (930) learn (902) acquired (854) write (842) transportation (842) find (816)
faithful	<NUMBER> (34087) <PROPER_NOUN> (32734) servant (20491) friend (15850) say (10470) man (9491) good (7010) love (6667) god (6638) old (6613) know (6282) remain (6175) true (5877) make (5854) commander (5530) shall (5490) life (5105) king (4959) thou (4822) find (4818) time (4482)
fatigue	<NUMBER> (10363) <PROPER_NOUN> (7763) sleep (6616) journey (6235) day (5989) hunger (4835) night (4832) long (4434) rest (4357) feel (4276) say (3614) exhausted (3534) man (3246) time (3166) wear (3051) fall (3035) little (2993) overcome (2835) make (2783) hours (2754) morning (2505)
feeling	<PROPER_NOUN> (2307) <NUMBER> (1241) little (714) say (642) sure (612) feel (608) think (422) find (418) way (406) time (388) hand (387) make (383) stand (382) hands (353) know (344) ask (339) eye (333) look (329) man (329) feeling (329) come (299)
ferret	<PROPER_NOUN> (1100) <NUMBER> (969) man (782) rabbit (580) like (536) eye (521) weasel (487) face (465) little (434) say (431) know (314) french (307) ferret (267) find (260) time (233) corner (216) old (213) keep (208) bring (205) mystery (186) boy (183)
find	<NUMBER> (652238) <PROPER_NOUN> (628319) say (237304) man (205832) come (156932) way (139048) make (136556) time (133112) know (127731) little (109708) think (109428) place (104836) shall (95078) great (88438) ain (85961) good (85217) leave (81978) tell (81399) day (78606) house (76897) old (76651)
fine	<NUMBER> (113190) <PROPER_NOUN> (94389) say (55123) man (35701) make (34819) like (25175) old (23470) day (22513) weather (21839) think (20756) come (19807) good (19278) little (18913) fellow (18702) fine (18656) ain (18436) young (17662) eye (17497) great (16313) look (15696) know (15666)

Continúa en la siguiente página

Cuadro C.3 – continuado de la página anterior

palabra	vecindad de palabras
fit	<NUMBER> (55859) <PROPER_NOUN> (55082) think (31842) say (28934) man (28123) make (25866) time (14248) come (12885) know (12662) like (12522) place (12489) find (11925) fit (10269) little (10182) good (9425) life (9065) work (8956) shall (8549) ain (8214) great (8084) laughter (8034)
free	<NUMBER> (81457) <PROPER_NOUN> (63838) man (49892) set (39679) make (34610) say (31925) free (26786) slave (26016) leave (22666) shall (20626) state (18817) people (18275) come (18081) life (17817) time (17008) think (16148) know (14557) let (14180) hand (13535) country (13242) like (13229)
graceful	figure (9866) <PROPER_NOUN> (8557) <NUMBER> (6902) tall (5177) beautiful (5084) eye (4509) form (3935) slender (3889) woman (3516) make (3495) young (3492) head (3478) little (3437) movement (3394) like (3385) look (3291) curve (3178) manner (3072) face (2989) hair (2658) girl (2653)
green	<PROPER_NOUN> (57667) <NUMBER> (57591) tree (37048) white (31622) grass (30709) blue (30564) little (25395) like (25202) yellow (23331) leave (21065) eye (18144) green (18067) look (16671) come (16460) light (15926) field (15505) bright (14631) flower (14602) old (14529) say (14018) thea (13783)
grip	hand (16251) <PROPER_NOUN> (14083) hold (9010) hands (8621) arm (7523) <NUMBER> (6495) say (6281) man (6039) throat (4818) feel (4437) finger (4184) come (3773) shoulder (3559) eye (3330) face (3289) like (3190) wrist (2876) firm (2859) tight (2736) tightly (2656) hard (2423)
hearth	sit (7753) afire (6207) <NUMBER> (5890) <PROPER_NOUN> (5625) burn (4801) stand (4444) light (4055) chair (3675) room (3041) old (2712) log (2690) ash (2666) thea (2629) home (2468) little (2417) lie (2312) ember (2311) wood (2235) come (2235) chimney (2163) cricket (2043)
holiday	<PROPER_NOUN> (13712) <NUMBER> (10922) christmas (6116) day (5919) make (5042) come (5005) time (4974) home (4722) spend (4541) holiday (3980) say (3459) work (3361) summer (3052) little (2966) school (2942) week (2907) days (2853) boy (2752) like (2674) man (2178) ain (2176)
keep	<PROPER_NOUN> (311402) <NUMBER> (293461) say (170653) man (100865) time (91826) make (89549) know (83474) eye (78639) come (69437) secret (64573) think (64556) little (61117) good (60618) away (59112) long (56894) way (51957) like (51936) day (51391) tell (50858) house (49337) let (46267)
lady	<PROPER_NOUN> (610824) young (282023) say (267499) <NUMBER> (174456) old (113166) come (90794) lady (87173) know (83299) mrs (78579) make (68672) gentleman (68284) sir (66880) think (65431) mr (64160) little (59962) miss (59630) lord (58156) tell (57398) dear (55571) look (54261) good (53761)
leave	<PROPER_NOUN> (650824) <NUMBER> (548219) say (270431) right (164852) come (157090) man (137291) room (131169) make (122603) time (117045) house (109560) little (107780) hand (105054) away (98580) know (91961) think (90613) find (81978) day (79088) like (78743) place (73453) tell (72721) home (71820)
live	<NUMBER> (240283) <PROPER_NOUN> (191307) life (110387) man (106031) say (92258) die (91104) long (87028) years (76547) know (68260) house (60971) shall (55956) time (55165) come (52870) live (50264) like (47421) old (47026) people (45413) make (44733) think (42793) little (37892) world (37768)

Continúa en la siguiente página

Cuadro C.3 – continuado de la página anterior

palabra	vecindad de palabras
local	<NUMBER> (22426) <PROPER_NOUN> (13999) government (7212) authority (6468) state (6119) local (5987) national (4581) general (4182) man (3841) make (3806) new (3690) police (3197) people (3147) country (3128) time (3093) federal (3043) ain (2993) town (2871) administration (2702) public (2605) great (2549)
match	<PROPER_NOUN> (29036) <NUMBER> (21375) light (18827) make (12835) strike (12447) say (11562) man (6961) match (6625) think (5567) good (5400) box (5362) find (5252) know (4672) cigarette (4178) ain (4050) candle (3898) come (3861) little (3794) burn (3649) like (3547) hold (3499)
material	<NUMBER> (29257) make (17500) <PROPER_NOUN> (13888) work (12690) raw (12152) life (8331) man (7937) build (7797) write (7623) find (7582) new (7506) use (7369) form (7192) supply (7010) spiritual (6963) world (6770) time (6619) things (6243) material (5644) great (5510) moral (4912)
mouth	<NUMBER> (106805) eye (61958) <PROPER_NOUN> (58281) open (45427) say (40575) word (32664) opened (26817) man (25627) like (24173) face (24016) come (21985) lip (21394) hand (21221) river (21180) little (20904) nose (20428) make (19985) look (17852) head (17727) speak (16983) tooth (16902)
nation	<NUMBER> (128860) people (32300) great (31342) nation (25974) shall (25567) man (25065) war (21470) <PROPER_NOUN> (19076) make (19008) god (17754) king (15437) government (15304) world (15071) country (15026) power (13435) time (13100) lord (12032) earth (11595) say (11572) state (11203) foreign (10853)
natural	selection (68479) <NUMBER> (61349) <PROPER_NOUN> (44807) man (35226) history (31318) say (21146) make (20517) life (16909) think (16046) great (13988) nature (13646) science (13524) species (13236) time (12942) quite (12140) natural (12111) know (11841) find (11766) like (11415) feel (11355) law (10803)
nature	human (110185) <NUMBER> (98950) man (92735) <PROPER_NOUN> (66771) make (46835) life (37023) say (36830) know (34059) things (32111) good (31971) nature (31419) like (25924) great (25583) art (24787) think (24135) god (22679) love (22558) time (22529) power (21690) find (21539) mind (20888)
oblique	<NUMBER> (2463) eye (1164) fold (1077) line (962) slightly (923) glance (886) <PROPER_NOUN> (828) eyebrow (820) direction (749) ray (614) slight (519) straight (500) light (497) make (477) oblique (427) man (419) stripes (375) longitudinal (366) sun (363) direct (356) angle (339)
play	<PROPER_NOUN> (112824) <NUMBER> (73728) game (47850) say (44887) play (28450) come (27442) make (25944) like (25910) man (25564) child (22064) little (21923) know (20427) time (18134) think (17957) good (16809) begin (16059) trick (14798) let (14642) boy (14375) write (13230) old (13102)
post	<NUMBER> (62667) <PROPER_NOUN> (53534) letter (23744) office (17957) come (11210) man (11114) say (8955) send (8935) write (8687) morning (8278) make (8163) post (7955) beat (7801) leave (7776) time (7723) return (6108) day (5941) ain (5878) place (5860) stand (5704) find (5462)
pull	<PROPER_NOUN> (17533) <NUMBER> (15346) say (10813) man (6973) pull (6003) try (5834) make (5240) don (5174) come (5042) away (5023) like (4692) let (4289) little (4115) long (3921) begin (3875) hand (3849) trigger (3794) think (3448) hold (3433) rope (3416) time (3402)

Continúa en la siguiente página

Cuadro C.3 – continuado de la página anterior

palabra	vecindad de palabras
replace	<NUMBER> (15338) <PROPER_NOUN> (8530) new (3864) article (3056) man (2643) old (2541) say (2539) shall (2493) make (2237) time (2177) table (2004) carefully (1883) pocket (1855) find (1854) lose (1705) following (1654) away (1578) leave (1445) little (1442) removed (1408) come (1251)
restraint	<PROPER_NOUN> (4451) imposed (3138) <NUMBER> (3075) free (2647) man (2084) feel (1906) life (1829) power (1813) freedom (1587) break (1562) say (1543) restraint (1519) keep (1480) liberty (1391) make (1250) time (1224) fear (1194) presence (1187) know (1173) think (1158) law (1012)
sense	<PROPER_NOUN> (63689) <NUMBER> (59046) common (54658) man (44589) good (41957) feel (35506) word (34242) say (32975) make (26953) know (21175) think (20531) life (20460) duty (17617) time (16153) sense (15907) come (15831) mean (15338) ain (15293) like (14804) mind (14618) great (13540)
serve	<NUMBER> (96657) <PROPER_NOUN> (55643) man (27266) say (26033) purpose (25033) make (19795) god (18856) shall (16001) time (15605) serve (12880) lord (12572) king (12239) years (12176) good (11870) thou (11677) right (11450) come (11263) know (11094) dinner (10754) little (9598) table (9566)
simple	<NUMBER> (49436) <PROPER_NOUN> (44580) man (22190) say (22099) make (19053) life (17668) know (13891) think (13570) like (12639) word (12381) fact (12324) little (12134) way (10472) people (10298) find (10217) natural (9819) form (9668) good (9306) time (9135) nature (9064) pure (8841)
solemn	<PROPER_NOUN> (17309) <NUMBER> (16018) say (9985) make (8396) oath (8289) voice (6357) face (5814) man (5709) silence (5625) day (4978) promise (4965) look (4791) come (4697) great (4666) word (4501) like (4347) tone (4341) vow (4063) eye (3904) procession (3659) time (3647)
spade	dig (6217) <PROPER_NOUN> (3715) <NUMBER> (3560) spade (2835) man (1982) say (1767) garden (1584) pick (1552) come (1478) ace (1419) work (1407) aa (1274) shovel (1269) little (1245) mattock (1237) earth (1214) pickaxe (1201) hoe (1142) hand (1033) make (998) strike (992)
stress	lie (3994) great (3596) <NUMBER> (3357) <PROPER_NOUN> (3174) word (2307) time (1996) weather (1666) man (1633) say (1574) storm (1504) fact (1442) strain (1356) life (1327) circumstances (1058) feel (1039) make (1027) little (1016) war (952) mental (937) syllable (914) particular (889)
strike	<NUMBER> (138027) <PROPER_NOUN> (118005) blow (60149) say (48683) man (45082) clock (41091) time (29644) make (27301) come (27022) head (26921) like (25587) think (25442) hand (25201) fall (23120) face (21763) know (19471) hear (18697) eye (18599) strike (18413) great (17802) look (16936)
train	<PROPER_NOUN> (59173) <NUMBER> (57768) station (33436) come (20946) leave (18212) say (15614) time (15577) think (14871) car (13680) stop (13049) train (12844) long (11531) start (11343) railway (11228) morning (11201) catch (10443) man (10081) night (10039) run (9473) way (9427) make (9057)
treat	<PROPER_NOUN> (18403) <NUMBER> (13966) say (12718) man (9954) like (9764) way (7626) subject (6262) treated (5983) tome (5969) know (5940) treat (5617) make (5436) think (5093) respect (4746) shall (4527) come (4509) friend (4385) manner (4031) child (4020) time (3864) great (3808)

Continúa en la siguiente página

Cuadro C.3 – continuado de la página anterior

palabra	vecindad de palabras
turn	<PROPER_NOUN> (124331) <NUMBER> (115845) say (69900) make (45831) come (43264) man (42942) away (37187) know (30435) round (28757) eye (27905) look (27443) way (26507) time (25564) think (25477) head (25238) like (24771) good (24492) shall (23462) face (22736) leave (21858) little (21786)
use	<NUMBER> (138201) make (133281) <PROPER_NOUN> (93198) say (64123) word (44829) man (40939) know (35614) time (29013) maybe (27908) mean (26450) good (26161) distributed (26025) used (23914) think (23685) language (21823) use (21591) little (20786) prohibited (20692) like (20311) long (19939) find (18877)
vital	<NUMBER> (5722) importance (5204) life (4340) force (3655) <PROPER_NOUN> (2923) point (2490) man (2438) principle (2279) question (2252) power (2114) great (1880) make (1867) energy (1822) vital (1633) body (1559) fact (1460) organs (1456) human (1446) time (1445) apart (1287) say (1209)
wander	<PROPER_NOUN> (15815) <NUMBER> (14257) eye (14047) away (8528) far (8195) night (6862) find (6836) come (6320) like (5811) little (5550) time (5372) long (5369) day (5301) lose (5148) street (5040) man (4877) way (4834) leave (4708) place (4679) mind (4624) say (4446)
wash	water (15016) <NUMBER> (14307) hands (13457) <PROPER_NOUN> (10718) say (6890) shall (6059) clean (5980) clothe (5336) away (5327) come (4469) face (4398) dish (4346) blood (4261) make (4150) wash (3432) foot (3119) thou (2953) time (2680) little (2648) let (2604) dry (2589)
work	<NUMBER> (353678) <PROPER_NOUN> (266513) man (130103) make (103819) say (97747) hard (96959) day (83561) time (80489) work (79709) great (60428) set (58750) good (58402) know (56316) come (55738) beat (53553) begin (52354) think (51904) find (50018) like (47750) little (47526) life (45907)
yew	tree (3858) hedge (1698) <NUMBER> (1640) bow (1541) alley (1461) <PROPER_NOUN> (1129) cypress (1059) garden (874) old (808) clip (781) churchyard (747) yew (739) dark (682) nook (606) house (532) wood (530) green (498) white (494) fir (467) walk (462) like (458)