Departamento de Computación
UNIVERSIDADE DA CORUÑA

TESIS DOCTORAL CON MENCIÓN INTERNACIONAL

Adquisición y representación del conocimiento mediante procesamiento del lenguaje natural

Doctorando: Milagros FERNÁNDEZ GAVILANES

Directores: Dr. Manuel VILARES FERRO
Dr. Éric VILLEMONTE DE LA CLERGERIE

A Coruña, Octubre 2012

TESIS DOCTORAL:	Adquisición y representación del conocimiento mediante procesamiento del lenguaje natural
AUTOR:	Milagros Fernández Gavilanes
DIRECTORES:	Dr. Manuel Vilares Ferro Dr. Éric Villemonte de la Clergerie
TUTOR:	Dr. Miguel Ángel Alonso Pardo
FECHA:	9 de Octubre de 2012

TRIBUNAL:

PRESIDENTE:

VOCAL 1^{O}:

VOCAL 2^{O}:

VOCAL 3^{O}:

SECRETARIO:

CALIFICACIÓN:

Agradecimientos

Es difícil entender la importancia de los agradecimientos de una tesis doctoral hasta que una no la finaliza. En ese momento es cuando te das cuenta de lo mucho que tienes que agradecer. Seguramente que cuando termine de escribirlos me falten muchos nombres, pero todos los que aquí aparecen tienen un hueco merecido. Es difícil resumir la gratitud que siento hacia las personas que han estado presentes en esta etapa. Sin el apoyo, tanto profesional como personal de los que aquí aparecen, este trabajo no hubiese llegado a ser realidad. Podría retocar estas líneas millones de veces, pero el significado final seguiría siendo el mismo. Sea como sea, simplemente gracias.

Especial reconocimiento merecen las dos personas sin las cuales esta tesis no hubiese tenido razón de ser. Éstos son mis directores. Al Dr. Manuel Vilares, tengo que agradecerle que me haya abierto las puertas de su grupo, dándome la oportunidad de tener una visión más amplia del mundo de la investigación. Pero sobre todo darle las gracias por su paciencia infinita y por sus sabios consejos, que aunque crea que no, siempre están presentes. Al Dr. Éric Villemonte de la Clergerie, gracias por permitirme realizar esa primera estancia. Ahí empezó todo. Luego vendrían más. Nunca olvidaré el recibimiento por parte de los integrantes del que en otros tiempos fue el grupo ATOLL, ya hoy grupo ALPAGE. Todos ellos de algún modo han puesto su granito de arena.

A ambos tengo que agradecerles todo lo que he aprendido en este proceso, pero sobre todo el apoyo recibido a lo largo de estos años.

El ambiente de trabajo en el cual se ha desarrollado esta tesis es responsabilidad de mis compañeros de laboratorio del grupo COLE. Gracias a todos ellos, a los que han estado desde el principio (Víctor y Fran), a los que pasaron por aquí (Juan, Moli, Sara, Erica, Nieves, Vanesa, Gonzalo, Cristina y Josefina) y, también a los que han ido llegando a lo largo de estos años (Daniel, Santi y Adrián). Concretamente, un agradecimiento muy especial a Adrián por ayudarme en la etapa final de pruebas. Sin él aún no hubiese terminado ;-) .

Gracias también a todos los integrantes del grupo LYS, por recibirme con los brazos abiertos, en especial a Miguel, Jesús y Jorge. En esa etapa de docencia, nunca me pusieron ningún inconveniente en lo que a horario se refiere y es de agradecer. A Margarita le debo la preocupación, siempre ha estado pendiente de saber como lo llevaba. Y a Carlos, algún día el 3D llegará a estos grafos.

A mis padres Julia y Eladio, qué decirles. No tengo palabras. Ellos son los responsables de lo que soy. Gracias por no haberme detenido nunca en el ansia por estudiar, aprender y trabajar en aquéllo que me gusta. Gracias por animarme siempre a seguir adelante y aguantar, con infinita paciencia, mis continuos cambios de humor durante este tiempo. Gracias por estar siempre ahí.

A David, mil gracias por haberme hecho la vida más fácil, estando a mi lado, en los buenos y malos momentos, animándome siempre a continuar. Le doy las gracias por todos los esfuerzos que ha hecho, por haberme hecho creer cada día que podía hacerlo, por toda su ayuda, aunque a veces no entendiese nada. Pero principalmente, gracias por hacerme feliz.

Y una lista infinita de nombres: a toda mi familia, del primero al último, que siempre se han preocupado de alguna u otra manera por saber como estaba. A las personas que, aunque no aparecen aquí con nombres y apellidos, han estado presentes de alguna forma durante el desarrollo de este trabajo y han hecho posible que hoy vea la luz.

A todos, mi eterno agradecimiento.

Resumen corto

Este trabajo introduce un marco para la recuperación de información combinando el procesamiento del lenguaje natural y conocimiento de un dominio, abordando la totalidad del proceso de creación, gestión e interrogación de una colección documental. La perspectiva empleada integra automáticamente conocimiento lingüístico en un modelo formal de representación semántica, directamente manejable por el sistema. Ello permite la construcción de algoritmos que simplifican las tareas de mantenimiento, proporcionan un acceso más flexible al usuario no especializado, y eliminan componentes subjetivas que lleven a comportamientos difícilmente predecibles.

La adquisición de conocimientos lingüísticos parte de un análisis de dependencias basado en un formalismo gramatical suavemente dependiente del contexto. Conjugamos de este modo eficacia computacional y potencia expresiva.

La interpretación formal de la semántica descansa en la noción de grafo conceptual, sirviendo de base para la representación de la colección y para las consultas que la interrogan. En este contexto, la propuesta resuelve la generación automática de estas representaciones a partir del conocimiento lingüístico adquirido de los textos y constituyen el punto de partida para su indexación.

Luego, se utilizan operaciones sobre grafos así como el principio de proyección y generalización para calcular y ordenar las respuestas, de tal manera que se considere la imprecisión intrínseca y el carácter incompleto de la recuperación. Además, el aspecto visual de los grafos permiten la construcción de interfaces de usuario amigables, conciliando precisión e intuición en su gestión. En este punto, la propuesta también engloba un marco de pruebas formales.

Resumo curto

Este traballo introduce un marco para a recuperación de información combinando procesamento da linguaxe natural e o coñecemento dun dominio, abordando a totalidade do proceso de creación, xestión e interrogación dunha colección documental. A perspectiva empregada integra automáticamente coñecementos lingüísticos nun modelo formal de representación semántica, directamente manexable polo sistema. Isto permite a construción de algoritmos que simplifican as tarefas de mantemento, proporcionan un acceso máis flexible ao usuario non especializado, e eliminan compoñentes subxectivos que levan a comportamentos difícilmente predicibles.

A adquisición de coñecementos lingüísticos parte duhna análise de dependencias basada nun formalismo gramatical suavemente dependente do contexto. Conxugamos deste modo eficacia computacional e potencia expresiva.

A interpretación formal da semántica descansa na noción de grafo conceptual, servindo de base para a representación da colección e para as consultas que a interrogan. Neste contexto, a proposta resolve a xeración automática destas representacións a partires do coñecemento lingüístico adquirido dos textos e constitúe o punto de partida para a súa indexación.

Logo, empréganse operacións sobre grafos así como o principio de proxección e xeneralización para calcular e ordenar as respostas, de tal maneira que se considere a imprecisión intrínseca e o carácter incompleto da recuperación. Ademáis, o aspecto visual dos grafos permiten a construción de interfaces de usuario amigables, conciliando precisión e intuición na súa xestión. Neste punto, a proposta tamén engloba un marco de probas formais.

Short abstract

This thesis introduces a framework for information retrieval combining natural language processing and a domain knowledge, dealing with the whole process of creation, management and interrogation of a documental collection. The perspective used integrates automatically linguistic knowledge in a formal model of semantic representation directly manageable by the system. This allows the construction of algorithms that simplify maintenance tasks, provide more flexible access to non-specialist user, and eliminate subjective components that lead to hardly predictable behavior.

The linguistic knowledge adquisition starts from a dependency parse based on a midly context-sensitive grammatical formalism. In this way, we combine computational efficiency and expressive power.

The formal interpretation of the semantics is based on the notion of conceptual graph, providing a basis for the representation of the collection and for queries that interrogate. In this context, the proposal addresses the automatic generation of these representations from linguistic knowledge acquired from texts and constitute the starting point for indexing.

Then operations on graphs are used and the principle of projection and generalization to calculate and manage replies, so that is considered the inherent inaccuracy and incompleteness of the recovery. In addition, the visual aspect of graphs allow the construction of user-friendly interfaces, balancing precision and intuition in management. At this point, the proposal also includes a framework for formal testing.

Índice general

Índice de figuras

PARTE I

Preliminares

CAPÍTULO I

Introducción

El texto, junto con la palabra, constituye uno de los canales de comunicación más poderosos. En particular permite remontarnos a los orígenes de la historia humana en la búsqueda de información. Sin embargo, su simplicidad de acceso universalmente aceptada y potenciada por la irrupción generalizada de las nuevas tecnologías, resulta ser aún un reto abierto en el ámbito de su gestión computacional. Así, hemos llegado a la paradoja de poder disponer de cantidades prácticamente ilimitadas de información, aunque su consulta por usuarios no especializados no ha avanzado en la misma medida, limitando sus aplicaciones prácticas.

En este sentido, las técnicas de *recuperación de información* (RI) han permitido flexibilizar las tareas de acceso y gestión, pero no responden totalmente a nuestros requerimientos como interlocutores humanos. Existe de hecho una necesidad real no sólo de localizar información, sino de extraerla y sintetizarla a partir de diferentes fuentes, en un proceso interactivo con el usuario.

Independientemente del nivel de detalle deseado para la herramienta informática en la determinación de la información que nos interese, su acceso requiere de la conjunción de diferentes capacidades, habitualmente contempladas en lo que conocemos como ámbito de trabajo relativo al *procesamiento del lenguaje natural* (PLN). Esta disciplina, íntimamente ligada a la *inteligencia artificial* (IA) y a la lingüística computacional, se ocupa de la formulación e investigación de mecanismos eficaces para la comunicación entre personas, o entre personas y máquinas por medio de lenguajes de comunicación humana, también denominados *lenguajes naturales* (LN's). De este modo, los modelos aplicados se enfocan no sólo a la comprensión del lenguaje en sí, sino a aspectos generales cognitivos humanos y a la organización de la información.

Este proceso de contextualización requiere fundamentalmente de tres consideraciones, cuya resolución constituye el núcleo de la presente tesis. La primera, un conocimiento previo, lo más profundo posible, por parte del sistema del ámbito de trabajo al que

las interrogaciones se refieren. La segunda, un análisis lo más detallado posible, de la estructura de dependencias sintáctico/semánticas de la interrogación. La tercera, la disponibilidad de un mecanismo que permita no sólo poner en relación las representaciones formales de conocimiento relativas a la interrogación y a la colección documental, sino también evaluar cualitativamente dicha relación e interpretarla.

A este respecto, la mayor parte de los sistemas de RI actuales basan su funcionamiento en motores de búsqueda cuyos mecanismos de localización de información se sitúan muy lejos de la filosofía que hemos descrito, para basarse en datos de naturaleza casi exclusivamente léxica. En este sentido, tales herramientas se basan en la capacidad que poseen para discernir con respecto de una consulta qué contenidos de la colección documental resultan relevantes de los que no lo son. En concreto, la *relevancia* de un documento viene determinada por la correspondencia entre la representación de su contenido y la de la consulta.

Sin embargo, la mayoría de estos sistemas usan el bien conocido *modelo de espacio vectorial* [274]. Éste se centra en el concepto de recuperación basada en *conjuntos de términos*[1] [129], donde la consideración de estructuras de dependencias sintáctico/semánticas es casi anecdótica. De hecho, consideran que la representación interna de los documentos, se basa en una interpretación del denominado *principio de composición* [155], según el cual la semántica de un documento reside exclusivamente en los términos que lo forman, sin tener en cuenta el sentido que sus autores quieren transmitir, lo que se traduce en la falta de consideración de sus significados en un contexto dado. Este hecho resulta sorprendente, pues los trabajos iniciales de investigación en el campo de las representaciones conceptuales [281] asociadas al tratamiento semántico de la información, datan de la misma época en la que se publicaban las primeras propuestas sobre RI [90, 282].

Ello supone en sí mismo una cierta contradicción. En efecto, dado que se podría considerar que la RI [331] es una tarea propia del PLN, lo más sensato sería incorporar algún conocimiento por parte del usuario a este nivel y alguna capacidad de razonamiento para mejorar la precisión en el procesamiento de las consultas. La respuesta a esta aparente incoherencia debe de buscarse en el rendimiento mostrado por los enfoques basados en *correspondencia de palabras*[2] lo que, de alguna manera, compensa la imprecisión derivada de considerar la recuperación como una función calculada sobre una secuencia de términos que aproxima la relación entre una consulta y un documento [57].

En el contexto descrito, aunque estas técnicas han demostrado ser sólidas y eficaces para una gran variedad de textos, los motores de búsqueda necesitan que el usuario indique de forma muy precisa y casi textual la consulta en relación al contenido que se pretende localizar, a riesgo en otro caso de obtener una avalancha de resultados sin interés. Esto implica que el usuario debiera conocer perfectamente, para asegurar un mínimo de

[1] en terminología anglosajona, *bag-of-words*.
[2] en terminología anglosajona, *word matching*.

precisión en las consultas, no sólo el ámbito de conocimiento en el marco del cuál ésta se realiza, sino también el protocolo de funcionamiento del propio motor. En el peor caso, consultar la colección documental puede convertirse en una tarea frustrante [101], fuera del control del interlocutor, que a veces no entiende cuales son los mecanismos y criterios que debe tener en cuenta para obtener resultados razonables. En consecuencia, un entorno de RI debiera facilitar el acceso al conocimiento no sólo cuando el usuario es un experto en la materia, sino también cuando se trata de no iniciados. Esto es, debería corresponder al sistema el acercarse al lenguaje humano y no al revés.

La consecuencia inmediata de la aplicación de este tipo de estructuras es una pérdida sustancial de precisión en las consultas, ya que las palabras no pueden considerarse por sí solas como detentoras del significado de la frase de la que forman parte, sino simples constituyentes de la misma, cuya naturaleza, significado y función sólo pueden determinarse en relación a las demás. Esto es, los conceptos asociados a una frase no pueden considerarse como la simple suma de significados de los términos que la componen, sino como el resultado del conjunto de restricciones que las relaciones semánticas entre palabras aplican sobre dicha condición. Actualmente, sin embargo, el creciente tamaño y complejidad de las colecciones documentales puede conducir a una ruptura de este inestable equilibrio, incrementando la exhaustividad en detrimento de la precisión, de tal manera que esta clase de técnicas difícilmente podrán mantener su interés práctico.

En particular, se ha argumentado en no pocas ocasiones que la integración de técnicas de PLN [331] podría contribuir a mejorar las prestaciones en sistemas de RI más sofisticados [281] gracias a una representación adecuada de los documentos. Ello pasa por considerar como punto de partida a las oraciones, y no sólo a las palabras, además de una herramienta capaz de identificar la semántica subyacente al texto. Esto implica a su vez poder disponer de técnicas eficientes para estudiar la naturaleza compleja de los lenguajes humanos, incluyendo el tratamiento de ambigüedades [139]. En efecto, el análisis de éstas constituye una preocupación fundamental ya que los sistemas de RI parecen ser más sensibles a la desambiguación errónea que a la propia ambigüedad [275]. En consecuencia, una elección inadecuada en la implementación podría hacer peligrar las teóricas ventajas de una arquitectura basada en el conocimiento real presente en el discurso.

Atendiendo a las estructuras lingüísticas del texto, sin necesidad de un conocimiento predefinido del dominio concreto a analizar, la mayoría de los autores consideran una combinación de capacidades para la resolución del problema planteado. Nos referimos en concreto al análisis sintáctico robusto, permitiendo la identificación de relaciones semánticas, y a algún tipo de estrategia estadística y/o heurística a fin de escoger las más relevantes entre éstas. En especial, los acercamientos estadísticos/heurísticos se aplican a menudo como complemento de los análisis léxicos y/o sintácticos realizados con un propósito de agrupación del significado. El objetivo es simplificar los conjuntos iniciales de enlaces semánticos, eliminando en la medida de lo posible las interpretaciones

ambiguas. Dado que estas técnicas se basan en un algoritmo distribucional [130] destinado a ser aplicado a un *corpus*, el tiempo y el espacio consumidos en su ejecución se convierten en factores esenciales de complejidad en su diseño.

Por este motivo, la hipótesis de la que partimos es que con una representación adecuada de los documentos e incorporando conocimientos semánticos limitados, es posible mejorar la eficacia de un sistema de RI. Esto requiere en primer lugar un análisis del texto en profundidad, lo que sitúa de lleno el problema en el marco de PLN, aunque con dos características propias. La primera tiene que ver con la cantidad de texto con el que el sistema ha de tratar, y que puede resultar tan grande y heterogéneo que resulte poco práctico para llevar a cabo un análisis exhaustivo. La segunda característica viene a suavizar los requerimientos derivados de la primera por cuanto un análisis semántico detallado y preciso no es necesario para las tareas de RI [147], lo que las distingue de otras más estrechamente relacionadas con el PLN como la traducción automática, las búsquedas de respuestas o los resúmenes automáticos [306].

1.1 | Contribución de la propuesta

El objeto principal de esta tesis ha sido el desarrollo y evaluación de un marco de RI combinando el PLN y el conocimiento de un dominio. Estos dos aspectos son en los que se han centrado nuestros mayores esfuerzos. Por un lado, hemos abordado la totalidad del proceso de creación, gestión e interrogación de la base de datos documental, desde una perspectiva que integra de forma automática el conocimiento lingüístico en un modelo formal de representación semántica directamente manejable por el sistema. En este sentido, creemos que nunca nadie antes había conseguido obtenerla de un modo automático y práctico, más allá de simples ejemplos de laboratorio. Por el otro, hemos planteado un marco formal novedoso para la evaluación de este sistema de RI basado en conocimiento. A continuación, trataremos brevemente estos dos aspectos.

1.1.1 | Desarrollo del marco de RI

El objetivo ha sido establecer un protocolo de actuación que permita extraer de forma automática la semántica atesorada en el texto, a la vez que asegurar un compromiso óptimo entre el rendimiento computacional y el lógico. De esta manera, las representaciones de los documentos obtenidos integran conocimientos básicos a partir del *corpus*, explotando tanto información lingüística como sintáctica. A este propósito, si nos centramos en la fase de análisis del texto, hemos considerado necesaria una estrategia en dos pasos para tratarlo. El primero se refiere a la adquisición de conocimiento léxico a nivel de frase, tarea para la que nos hemos inspirado de la arquitectura *Alexina* [262], cuyo núcleo se basa en un analizador de estados finitos. Éste integra un pre-procesador [264] que asume la separación de cadenas de caracteres, la corrección ortográfica y el

reconocimiento de entidades nombradas (REN), y que tiene como principal recurso un léxico a gran escala [263]. La salida incluye todas las posibles interpretaciones para cada forma léxica en un *grafo acíclico dirigido* (GAD) que es posteriormente utilizado en una fase de análisis sintáctico, y que constituye el segundo paso. Al respecto, hemos elegido un formalismo *suavemente dependiente del contexto* [321], que proporciona la potencia suficiente para su aplicación sobre LN's, sin por ello renunciar a la eficacia computacional.

En cualquier caso, para rentabilizar las ventajas asociadas al análisis de texto en tareas de RI, también es necesario disponer de una notación formal que sirva como intermediario entre el humano y el ordenador. Concretamente, los *grafos conceptuales* (GC's) [293] poseen la potencialidad necesaria para describir el significado de los datos de acuerdo con la visión del usuario, a la vez que podemos asociarlos con procedimientos que permiten acceder a los datos en la máquina. Estamos así en disposición de evitar el tener que recibir una formación específica para acceder a ellos e interpretar tanto resultados finales como parciales, algo de lo que también adolece la recuperación basada en conjuntos de términos. Por otra parte, la consideración de un mecanismo de inferencia conceptual como el señalado nos permite estimar la *granularidad semántica* de un documento [351], la cual hace referencia al nivel de detalle que conlleva un elemento de información [100]. De esta manera, se abren las puertas para abordar tareas que implican búsqueda de consultas ambiguas, colecciones documentales incompletas y RI aplicada en dominio específico. Todo ello justifica que nos hayamos decantado por la elección de este tipo de estructura como formalismo de representación semántica.

Formalmente, los GC's obtenidos son derivados de acuerdo con un modelo de dependencias. Concretamente, la colección documental se analiza sintácticamente en un primer momento con el fin de generar un *grafo inicial de dependencias* (GID's) que más tarde será traducido en uno de dependencias *gobernante/gobernado* (GDGG's), es decir, relacionando el núcleo de un sintagma con sus modificadores. A partir de aquí y mediante la aplicación de un conjunto de valores iniciales proporcionados por el programador para las clases semánticas (los tipos), marcadores lingüísticos y patrones sintácticos, podemos aproximar y extender de forma fehaciente ambos conjuntos iniciales de dependencias y clases. Una cuidadosa implementación en programación dinámica permite posponer el tratamiento de las ambigüedades tanto de tipo léxico como sintáctico a una posterior fase de definición semántica, donde un protocolo de adquisición de conocimiento iterativo sirve para filtrar interpretaciones irrelevantes con el fin de obtener los GC's. Esto es lo que nos va a permitir realizar una formulación simple de la tarea de recuperación. Así, cuando un usuario realiza una pregunta en LN, el sistema la traduce a un GC y luego trata de buscar en la colección documental otros GC's que sean relevantes con respecto al primero. Una vez encontrados, se pueden utilizar para acceder a su información y calcular las respuestas.

Más tarde, necesitaremos incorporar una *función de ordenación*[3] con el fin de clasificar los documentos recuperados en base a su relevancia con respecto a la consulta.

[3]también llamada *función de recuperación* por Fuhr and Buckley [103].

El objetivo es evitar que el usuario pierda el tiempo buscando en las listas de resultados obtenidas, entendiendo que en ellas se encuentran numerosos documentos irrelevantes, especialmente cuando sabemos de antemano que quién las revisa rara vez va más allá de la primera página del conjunto recuperado [117], lo cual constituye una causa mayor en la falta de satisfacción asociado a los sistemas de RI [94] y puede llegar a desvirtuar la capacidad real del propio buscador [115]. Para resolver este problema, nos hemos inspirado en trabajos anteriores, donde la función de ordenación se caracteriza mediante una relación de orden parcial sobre el conjunto de transformaciones aplicadas a la consulta con el fin de satisfacer su cometido en la colección documental [111]. La idea consiste en asignar diferentes pesos a estas transformaciones dependiendo de su naturaleza estructural, lo cual nos permitirá centrarnos en criterios de búsqueda lejos de las preferencias personales, descartando los enfoques basados en aprendizaje supervisado debido a su elevado coste en términos humanos.

1.1.2 | Evaluación del marco de RI

En relación al segundo aspecto, existe una preocupación primordial en el campo de la RI que es la evaluación. En este sentido, nuestra propuesta define un marco formal de pruebas que permite la consideración de diferentes técnicas de ordenación para estos sistemas, como son la aplicación o no de *juicios de relevancia* (JREL's), a menudo almacenados en un fichero denominado *relevancia de la consulta* (CREL), y la selección de un conjunto representativo de *consultas o tópicos* en función de las necesidades de información. De un modo más detallado, en el caso de la tarea de ordenación nuestro punto de partida ha sido el protocolo clásico empleado en la conferencia *Text REtrieval Conference*[4] (TREC), un congreso de carácter anual organizado por el *National Institute of Standards and Technology* (NIST) y la *Information Technology Office* de la *Defense Advanced Research Projects Agency* (DARPA), y basado en JREL's [332]. Pero también hemos estimado una simple variación de éstos usando *pseudo*-JREL's (PJREL's), propuestos por Soboroff *et al.* [290] y una alternativa algo diferente, incorporando los JREL's y/o PJREL's pero considerando un criterio algo distinto para la realización de la ordenación. Para ello, hemos retomado una técnica inspirada en la noción de *autoridad del sistema* descrita por Mizzaro *et al.* [208]. En cuanto a las técnicas de ordenación que no tienen en cuenta los JREL's, se optó por evaluar nuestra propuesta mediante un método inspirado en Wu *et al.* [347], que parece ser uno de los más populares en su tipo y que se basa en la idea de comparar la efectividad del motor de búsqueda con los resultados proporcionados por un conjunto de sistemas de RI que sirvan como referencia.

Con respecto a la elección del conjunto de consultas, hemos combinado una serie de trabajos anteriores en torno a dos preguntas complementarias. La primera se refiere a la selección de una consulta para un sistema de RI individual, aplicando el concepto de *precisión media* (PM) [37]. El siguiente consiste en la selección, pero esta vez de un

[4] `http://trec.nist.gov/`.

conjunto de tópicos para un determinado sistema [121]. A partir de estas técnicas, y a falta de soluciones definitivas y específicas en el estado del arte, proponemos un método razonado de selección a partir de un conjunto de sistemas de RI, inspirado tanto en la valoración basada en el tipo humano como en la noción de *conectividad del tópico*[5] propuesto por Mizzaro *et al.* [208].

Expuestas nuestras aportaciones, creemos necesario puntualizar que, para la realización de las pruebas y experimentos, se ha utilizado un *corpus* botánico que describe la flora del África Occidental. En este sentido, la presente tesis tiene su origen en BIOTIM [258], una iniciativa de investigación sobre la gestión integral de este tipo de documentos. En particular, nos hemos centrado en el trabajo «*Flore du Cameroun*», publicada entre 1963 y 2001, fruto del trabajo de varios autores, el cual está compuesto de aproximativamente 40 volúmenes escritos en francés, donde cada uno de ellos consta de unas 300 páginas. Este *corpus* se encuentra descrito con más detalle en el Apéndice A, y lo denotaremos como *corpus* $\mathscr{B}$. Debido a su utilización, casi la mayoría de los ejemplos existentes a lo largo de esta tesis están inspirados en él.

1.2 | Ámbito de la tesis

El trabajo desarrollado en esta tesis doctoral se enmarca en dos áreas de investigación: el PLN cuyo objetivo fundamental es facilitar la comunicación entre las personas y las máquinas mediante el lenguaje humano y la RI, cuya tarea es localizar dentro de una colección de documentos aquéllos que son relevantes a una consulta.

En lo que respecta al contexto dentro del cual se ha desarrollado el trabajo de investigación de esta tesis, éste se ha llevado a cabo dentro de diferentes becas y proyectos que recogemos a continuación.

Becas de investigación

- *Beca para estancias del Centre Français pour l'acceuil et les échanges Internationnaux* (EGIDE) del Ministère des Affaires Étrangères et Européenne, Francia, del 27/02/2006 al 30/07/2006.
- *Beca para estancias en el extranjero* del programa Recursos Humanos del Plan Gallego de Investigación, Desarrollo e Innovación Tecnológica de la Xunta de Galicia. *Beca predoctoral*, del 03/11/2008 al 05/12/2008.
- *Beca-contrato María Barbeito* del programa de Recursos Humanos del Plan Gallego de Investigación, Desarrollo e Innovación Tecnológica de la Xunta de Galicia. *Beca predoctoral*, del 28/12/2007 al 30/06/2010.

[5] en terminología anglosajona *topic hubness*.

- *Beca de investigación* del programa de Ayudas a Grupos de investigación de la Universidad de Vigo, del 15/10/2011 al 14/11/2011.

Contratos de investigación

- *Promoción y coordinación de prácticas socio-sanitarias en geriatría* (PGIDIT 03SIN30501PR) de la Xunta de Galicia, del 16/11/2005 al 28/02/2006 y del 16/08/2006 al 15/10/2006.
- *Extracción de información económica multilingüe* (TIN2004-07246-C03-01) del Ministerio de Educación y Ciencia, del 01/11/2006 al 27/12/2007.
- *Análisis robusto para la búsqueda de respuestas* (ARBORE) (HUM2007-66607-C04-03), del Ministerio de Educación y Ciencia, del 01/07/2010 al 31/12/2010.
- *Procura de respostas mediante grafos conceptuales*, de la Universidad de Vigo, del 01/10/2011 al 14/10/2011.
- *Análisis de textos y recuperación de información para la minería de opiniones: extracción de conocimiento* (ATRIO) (TIN2010-18552-C03-01), del Ministerio de Educación y Ciencia, del 01/01/2011 al 31/05/2011 y del 15/11/2011 al 31/08/2012.

Proyectos de I+D de ámbito internacional

- *Automatic design of a proper noun ontology for question-answering system* (acción integrada hispano-lusa HP2007-0061)
- *ESF Research Networking Programme: Evaluating Information Access Systems*, de la European Science Foundation, del 01/06/2011 al 31/06/2016.

Proyectos de I+D de ámbito nacional

- *Búsqueda de respuestas empleando metagramáticas* (HUM2007-66607-C04-02), del 01/01/2004 al 31/12/2007.
- *Análisis de textos y recuperación de información para la minería de opiniones: análisis de enunciados y extracción de relaciones* (ATRIO) (TIN2010-18552-C03-02), del 01/01/2010 al 31/12/2013.

Proyectos de I+D de ámbito autonómico

- *Consolidación y estructuración de unidades competitivas* (INCITE08 ENA305025ES), del 01/01/2008 al 31/12/2008, de la Xunta de Galicia.

- *Consolidación y estructuración de unidades competitivas* (INCITE09 EIR305070ES), del 01/01/2009 al 31/12/2009, de la Xunta de Galicia.
- *Consolidación y estructuración de unidades competitivas* (INCITE845B-2010/067), del 01/01/2010 al 31/12/2010, de la Xunta de Galicia.
- *Mejora en la recuperación de noticias y en el acceso a la información financiera: recuperación de textos sobre bases documentales de agencias de noticias* (PGIDIT07SIN005206PR) de la Xunta de Galicia, del 01/01/2007 al 31/12/2010.

Proyectos de I+D local y de la universidad

- *Entorno abierto para la recuperación de información semántica en colecciones textuales sobre dominios acotados* (2009-INOU-7) de la Universidad de Vigo, del 15/05/2009 al 15/05/2010.

Redes temáticas

- *Red Gallega de Procesamiento del Lenguaje y recuperación de información* (REDPLIR) de la Xunta de Galicia, del 01/01/2006 al 31/12/2010.
- *Red Gallega de Lingüística de Corpus* de la Xunta de Galicia, del 01/01/2009 al 31/12/2010.
- *Red Gallega de Recursos Lingüísticos para una Sociedad del Conocimiento* (RELISCO) de la Xunta de Galicia, del 01/01/2011 al 31/12/2012.

Estancias de investigación

- *Institut National de Recherche en Informatique et Automatique*, Francia. Se han realizado dos estancias de cerca de siete meses de duración en el grupo ATOLL dirigido por el *Dr. Éric Villemonte de la Clergerie*, investigador de reconocido prestigio en el campo del PLN. Estas estancias se realizaron bajo su tutela y el tema han sido el desarrollo de metodologías para la extracción de ontologías a partir de descripciones botánicas analizadas sintácticamente.
- *Universidad de Paris Diderot-Paris VII - Institut National de Recherche en Informatique et Automatique*, Francia. Se han realizado dos estancias de cerca de tres meses de duración total en el grupo ALPAGE que dirige la *Dra. Laurence Danlos*, bajo la tutela del *Dr. Éric Villemonte de la Clergerie*. El tema de estas estancias han sido la estabilización de las metodologías para la extracción de ontologías a partir de descripciones botánicas analizadas sintácticamente.

CAPÍTULO II

Estado del arte

La investigación en lo que a RI se refiere no es algo nuevo. Concretamente, es anterior a los años 60, momento en el cual se introdujeron por primera vez sistemas dedicados a la recuperación de textos. En aquellos años, estos documentos eran considerados como un mero conjunto de términos [20] que se indexaban en su totalidad. Dicho de otro modo, se trataba de grafías cuya semántica [112] no se consideraba. Tampoco se tenían en cuenta los contextos en los que aparecían ni el orden seguido, ya que se suponían independientes unos de otros. La única información considerada eran sus frecuencias y el peso que se estimaba debían de poseer en base a ellas, es decir, la aplicación de técnicas cuantitativas [192].

Sin embargo, desde entonces se han llevado a cabo numerosos trabajos que han destacado las limitaciones existentes con este tipo de representación [27, 209, 259, 284, 345], y se ha tenido que optar por utilizar, además de las palabras, otro tipo de datos. En este sentido, los investigadores siempre han mostrado cierta fascinación por la incorporación de técnicas de IA y PLN a la RI. Se trata de conseguir la integración de técnicas de interpretación de la semántica del texto, con el fin de identificar a un conjunto de elementos con unas determinadas cualidades, llamados *descriptores* y que serán empleados en la generación de estructuras de datos, que darán acceso a los documentos y que denominaremos *índices*. El objetivo no es otro que aprovechar esta información para realizar las tareas de recuperación [147].

En este punto, el estado del arte nos sitúa en un marco genérico de trabajo al que se refiere de diferentes formas. Así, algunos autores hablan de *indexación motivada lingüísticamente* [171, 218], mientras que otros consideran como más apropiado el término *indexación semántica* [167, 168, 285]. Algunos trabajos recurren incluso a la expresión de *recuperación inteligente* [76, 112, 288, 290, 306] para subrayar la interacción entre la mente humana y la IA a través de redes y tecnología.

Pero además, la naturaleza determinista de los sistemas de RI propicia su necesidad

intrínseca de evaluación. Surge entonces un amplio campo de trabajo dedicado específicamente a la calibración de medidas que permitan valorar su efectividad.

2.1 | Indexación semántica

Se podría decir que el primer autor que aportó luz sobre la indexación automática del contenido de documentos, fue Luhn [192]. Este autor consideraba que la importancia de una palabra en un texto estaba estrechamente ligada a su frecuencia, por lo que en base a ella las clasificaba en orden descendente. Partiendo de esta ordenación, estimaba que las frecuencias medias eran las más adecuadas, obviando las demás. En este sentido, las frecuencias elevadas correspondían a palabras frecuentes en exceso, que no permitían discriminar entre los diferentes textos, mientras que las poco comunes correspondían a términos de escaso poder expresivo.

Posteriormente, en la Universidad de Cornell se desarrolló uno de los primeros sistemas de RI basado en indexación automática, denominado SMART [269]. Éste contribuyó a avances en el estado del arte que incluyen desde el modelo vectorial, esquemas de ponderación y diferentes medidas de similitud, hasta métodos de clasificación. Este trabajo se retomaría más tarde con aportaciones extra al desarrollo de las ponderaciones y de métricas de proximidad [253, 273, 314].

Una alternativa al modelo vectorial es el *modelo probabilístico*, propuesto por Maron y Kuhn en [200], cuyas principales contribuciones se sugirieron más tarde por Robertson y Sparck Jones en [253]. Pero tampoco nos podemos olvidar de otro de los modelos clásicos, el denominado *modelo booleano*, basado en *álgebra de Boole* [71], y que se ha utilizado con éxito durante muchos años. A este respecto, se han propuesto numerosas extensiones [85].

Cuando se habla de la incorporación de técnicas de IA y PLN en RI, se consideran diferentes niveles de actuación, pero siempre con un doble objetivo: integrar técnicas de interpretación de textos para identificar el conjunto de descriptores [27, 223, 284], y proporcionar las características de la estructura interna de los índices asociados [233, 147]. Tradicionalmente, estas estructuras de indexación pueden ir desde simples palabras[1] hasta unidades multipalabra[2]. Por lo tanto, sobre ellos se suele aplicar un leve análisis lingüístico, utilizando léxicos para lograr una simple descomposición morfológica y la reducción de las palabras a su raíz, eliminando sufijos, afijos y demás de un modo superficial[3] [114, 142, 163]. Pero también se puede aplicar un análisis algo más profundo, que revele la estructura interna de las palabras[4]. Debido a la abundancia de información

[1] también llamados *términos simples*.

[2] también llamados *términos compuestos*.

[3] en terminología anglosajona se denomina *stemming*.

[4] por medio de la lematización o de las familias morfológicas sin tener en cuenta la información sintáctica.

disponible, estos métodos siguen siendo de los más empleados, y son capaces de hacer frente a algunos fenómenos lingüísticos complejos tales como los pronombres clíticos, contracciones y reconocimiento de nombres propios [10].

Sin embargo, nuestro principal interés se centra en captar la esencia de los documentos mediante la utilización de técnicas de análisis algo más elaboradas, tales como el uso de sintagmas significativos, pero también de frases como condición para la categorización automática de los documentos. Se trata en definitiva de una vieja idea que debiera marcar una mejora sobre el uso de palabras sueltas, aunque en la práctica exista poca evidencia de ello. De hecho, la convicción generalmente aceptada durante mucho tiempo [288, 147] era que sólo las técnicas lingüísticas superficiales podían resultar de interés en el desarrollo de este tipo de aplicaciones [288], aunque, en el mejor de los casos, su efecto positivo sobre la precisión era pequeño [171]. No obstante, la característica que define a estos métodos es que explotan conocimientos léxicos, morfológicos y/o sintácticos, con el fin de detectar relaciones de dependencia lingüística entre palabras, su representación formal y posterior definición de un mecanismo de localización de información en base a ésta.

En este sentido, podemos diferenciar [171, 355] dos niveles de complejidad en el tratamiento de dependencias en textos. El nivel más bajo se orienta al léxico, lingüísticamente menos sofisticado y representado por un grupo de técnicas conocidas como *modelado de dependencias*. Por lo general, estos sistemas consideran las dependencias existentes entre determinados pares o ternas de palabras [270], a menudo asociadas a un modelo probabilístico [58, 180, 193, 291] con el fin de clasificar las relaciones más plausibles. En este sentido, la mayoría de las estrategias de extracción de términos compuestos se basan en el uso de métodos estadísticos [93], que comprueban el grado de relación[5] existente entre los términos simples que constituyen el par o también en un reconocimiento simple de patrones [156, 283], en lugar de considerar las relaciones estructurales entre los elementos que conforman la oración. Más recientemente, algunos autores propusieron la utilización de técnicas de análisis superficial para la detección de estos pares [10] y/o ternas [171] de palabras relacionados mediante algún tipo de dependencia sintáctica. Es el caso, por ejemplo para el francés, del desarrollo de herramientas que permiten realizar la tarea de extracción, tales como TERMINO, LEXTER y ACABIT. En el caso de TERMINO[6], se trata de una de las primeras desarrolladas con el fin de adquirir automáticamente sintagmas nominales [80], y está construida sobre la base de un formalismo para la expresión de gramáticas del LN, denominado *Atelier FX*[7], es decir, está centrado en la aplicación de un análisis morfosintáctico basado en reglas. Del

[5] existen diversos tipos de medidas estadísticas que tratan de cuantificar el grado en el que estos pares se relacionan, tales como las *frecuencias de Lebart y Salem* [179] y la *medida de la información mutua de Church* [62].

[6] la versión actual se llama NOMINO [234].

[7] es un entorno de programación dedicado a la concepción de sistemas de análisis lingüístico, de extracción de información en textos y de puesta a punto de paquetes de programas informáticos a base de conocimiento.

mismo modo, LEXTER[8] también se centra en la extracción de sintagmas nominales [33] susceptibles de ser términos compuestos, pero con la diferencia de que el *corpus* debe de estar previamente anotado y desambiguado [34, 35], organizando los resultados bajo una forma de red. Finalmente, ACABIT retoma las ideas desarrolladas en TERMINO y LEXTER, agrupando variantes para extraer secuencias nominales, siguiendo patrones, tales como [Sustantivo - Adjetivo], [Sustantivo - Sustantivo], [Sustantivo - Preposición - (Determinante) - Sustantivo] y [Sustantivo - à - Infinitivo] [79]. En una segunda fase, utiliza medidas estadísticas para determinar el grado de relación entre los componentes de los términos binarios obtenidos, empleando para ello un *corpus* especializado y una lista de términos válidos extraídos del mismo. Todos estos trabajos muestran la mejora obtenida con respecto al modelo basado en palabras con independencia del idioma[9], en particular cuando se trata de un lenguaje rico en léxico y morfología. Sin embargo, el principal problema radica en la dificultad de integrar la proximidad de términos en el marco descrito. El espacio de parámetros puede volverse muy amplio considerando directamente las dependencias, haciendo la estrategia sensible a la escasa información y al ruido, lo que podría contrarrestar relativamente las pequeñas ventajas que se podrían obtener y sobre las que justificar el interés en modelos de proximidad del lenguaje [355].

Por este motivo, el nivel superior en el tratamiento de dependencias en textos trata de incorporar unidades mayores a las palabras a la hora de afrontar su representación, de modo que las dependencias existentes entre términos pueden ser capturadas indirectamente. Al igual que ocurría en el caso anterior, existen técnicas para la extracción de frases directamente relacionadas con métodos estadísticos [69, 108], con reconocimiento de patrones [246], pero también con técnicas de análisis sintáctico profundo [98, 300]. Sin embargo, aunque no se requiere de un análisis semántico muy detallado y preciso para la realización de tareas de RI [288], con el crecimiento desmesurado de la información, resulta difícil recuperar los documentos relevantes únicamente mediante métodos estadísticos [306]. El origen del problema se sitúa en el excesivo número de términos susceptibles de ser de interés para la descripción de una colección documental, pero a su vez también está relacionada con la dificultad de hacer frente a la escasez de datos en este contexto. En este sentido, las representaciones de textos basadas en grafos etiquetados parecen ser capaces de detectar enlaces no siempre evidentes entre los conceptos [145, 197, 285], independientemente del tamaño del *corpus* considerado. El acercamiento no sólo resulta prometedor, sino que posee el potencial de mejorar el modelo estándar de conjuntos de términos, sobre todo en respuestas a consultas largas [187], una idea en torno a la cuál el consenso es muy amplio [70], siendo varias las estrategias propuestas. Por tanto, aunque hasta hace poco el más conocido de estos

[8] la versión actual de LEXTER es SYNTEX [77] y permite la extracción, a partir de un *corpus*, de sintagmas nominales, verbales y adjetivales.

[9] en la práctica, en los entornos de recuperación, normalmente se supone que las palabras asignadas a los documentos de una colección aparecen de manera independiente las unas de las otras [270]. La hipótesis de independencia entre ellas no es realista en muchos de los casos, pero su uso conlleva la utilización de un algoritmo de recuperación simple.

acercamientos eran las *redes semánticas* [182], probablemente ninguno de ellos ha sido tan popular en los últimos tiempos como los GC's [293]. En realidad, los GC's son una extensión de las anteriores, introduciendo la noción de dependencia entre nodos. Éstos poseen tres ventajas principales como método de descripción formal. En primer lugar, pueden apoyar una correspondencia directa a partir de una base de datos relacional [67]. En segundo, pueden ser usadas como base semántica para el LN. Finalmente, basándonos en las transformaciones sobre grafos, permiten dar soporte a inferencias automáticas para calcular las relaciones que no son explícitamente mencionadas [112].

Esta aparente versatilidad del modelo basado en grafos debe además dar respuesta a la búsqueda de aquellos documentos que se encuentran representados de un modo incompleto, incluso a partir de consultas confusas. Este fenómeno, que ha justificado durante bastante tiempo la consideración de estrategias basadas en lógica probabilística, crece ahora de manera exponencial como consecuencia de la imposibilidad de integrar la cantidad total de información disponible en tareas de RI. Se trata en definitiva de formalizar la implementación del *principio de incertidumbre lógica de van Rijsbergen's* [315], según el cual la relevancia es una cuestión de grado y el problema central de la RI radica en como modelarlo y medirlo. Como consecuencia, asumir que dicho proceso puede ser mejorado mediante coincidencias exactas o por medio de la lógica clásica es un intento vano [112]. Este desajuste ha servido en cierto modo de campo propiciatorio para difundir ese sentimiento de que la mejora utilizando frases como índices no parece que sea la alternativa que mejor se ajuste al tratamiento de este tipo de problemas [10] de RI [109].

En este contexto, algunos autores adoptan una posición intermedia, investigando técnicas que hacen uso de conocimientos semánticos limitados, los cuales a su vez pueden ser fácilmente representables a partir del texto usando un formalismo en forma de GC [294]. Esto permite expresar el sentido de la colección documental de una manera lógicamente precisa, humanamente entendible y computacionalmente manejable. Gracias a la correspondencia directa existente entre este tipo de representación y el lenguaje, los GC's desempeñan el papel de lenguaje intermedio para la traducción entre los formalismos orientados a la máquina y el LN. Pero además, este tipo de representación gráfica sirve de lenguaje de especificación y de modelo legible por el usuario, a la vez que formal. Esto justifica que la noción de consulta conceptual date de los primeros tiempos de la investigación en el campo de la RI [295], así como el esfuerzo llevado a cabo en los últimos años con el fin de reemplazar las nociones clásicas probabilísticas por transformaciones formales de grafos [112], o simplemente de completarlas [288, 306].

[10] en algunos casos, la mejora de la eficacia se logra mientras que para otros, se alcanzan unos resultados marginales o negativos.

2.2 | Estrategia de ordenación

Tradicionalmente, la relevancia de los documentos se ha venido estimando usando una variedad de funciones de ordenación basadas en la similitud, que, en la práctica, no dejan de ser simples estrategias empleadas por los motores de búsqueda para ajustar los pesos asociados a los términos de indexación con el fin de optimizar su rendimiento[11]. Más recientemente, las funciones basadas en la popularidad han ganado cierta notoriedad. Estos modelos explotan la existencia de una correlación cercana entre la popularidad y la relevancia, principalmente en el caso de sistemas de RI que gestionan gran cantidad de datos y accesos por parte de los usuarios, como en el ejemplo típico de las búsquedas Web [165, 230]. Sin embargo, aunque en la actualidad los algoritmos encargados de la evaluación de la popularidad de los documentos se han vuelto cada vez más sofisticados, es necesario aplicar un esfuerzo específico para evitar algunos problemas inherentes a esta técnica. Nos referimos concretamente al tratamiento de contenidos de nueva incorporación que poseen pocos accesos [22, 47, 54, 86, 91, 173, 219], al hecho de que los documentos más populares tienden a serlo cada vez más [19, 59, 60, 113] o a la eliminación de posibles manipulaciones en las ordenaciones mediante la utilización de enlaces promovidos artificialmente [14, 16, 53, 148, 194, 226, 301].

A pesar de ello, ambos modelos de ordenación, los basados en la similitud y en la popularidad, no parecen ser por sí solos lo suficientemente eficaces como para dar apoyo en la RI aplicada a un dominio general o incluso a uno específico [351]. Este es el motivo por el que se justifica la consideración de propuestas híbridas, ya ampliamente aplicadas [87, 132, 230], incluso cuando las basadas en similitud parecen ser el punto de partida determinante para la obtención de la eficiencia en la recuperación. Con respecto a esto, una alternativa para mejorar su rendimiento consiste en medir directamente la similitud conceptual, la cual puede ser estimada de diferentes maneras. Así, algunos trabajos la calculan mediante el *concepto de menor ancestro común* (CMAC) a partir del contenido de información, algo que parece acercarse a las funciones de ordenación implícitas ejercidas por los humanos [243]. La idea original se debe a Cohen *et al.* [68] que describen un método para calcular la CMAC entre un par de conceptos, el cual nos permite relacionarlos a través de una descripción más específica que integra las respectivas estructuras. De esta manera, podemos inferir relaciones de subconcepto/superconcepto (resp. si un determinado individuo pertenece a un concepto determinado), proporcionando una herramienta para obtener elementos explícitos comunes y derivar conocimiento implícito usando técnicas orientadas a la inclusión en una categoría (resp. instancia) [175]. El estado del arte retoma este estudio con el objetivo de utilizar el contenido de la información para evaluar la similitud semántica en las taxonomías [243], y que más tarde serviría de inspiración para lidiar de diferentes maneras con las tareas de computación en el contexto de la tecnología en RI. Es el caso de algunos autores [215] que se aprovechan

[11] algunos autores hablan indistintamente de estrategias de ponderación de términos y de funciones de ordenación [94].

directamente de esta técnica para ampliar las medidas clásicas para la comparación de textos, como por ejemplo en el caso del coeficiente Dice [84]. De la misma manera, se consideran otras técnicas diferentes de las que utilizan CMAC, incluyendo a su vez extensiones alternativas a la medida Dice [216], así como relaciones de generalización asociadas a un dominio de conocimiento específico [259]. En cualquier caso, estas propuestas necesitan en primer lugar disponer de una estructura ontológica basada en conocimiento para representar estos conceptos, así como la tecnología estadística basada en *corpus* para generarlos y gestionarlos, situándonos de este modo en el contexto de la RI conceptual [295].

Desde el punto de vista operativo, sea cual sea el criterio de relevancia considerado, una función de ordenación se puede clasificar atendiendo a tres criterios complementarios relacionados con su fase de generación: la capacidad de adaptación al contexto, la naturaleza supervisada y la consideración de un modelo basado en aprendizaje [183]. En relación al primero de ellos, la mayoría de los sistemas de RI utilizan una estrategia fija para apoyar la tarea de clasificación definiendo su contexto de trabajo, independientemente de la heterogeneidad de los usuarios, de las consultas y de las colecciones [94]. Es el llamado *consenso de búsqueda*, en el que la relevancia calculada para toda la población se supone apropiada para todos los individuos y, como consecuencia, todos obtienen los mismos resultados. A pesar de que podríamos interpretar esta uniformidad como una ventaja, debido a que permite la comparación de los resultados de búsqueda entre los diferentes usuarios, lo cierto es que la idea de adecuar las características del proceso de recuperación a nuestras propias preferencias resulta siempre atractiva. Se habla entonces de *búsquedas personalizadas* [235], un enfoque que parece no aplicarse de forma consistente en diferentes contextos [271, 357].

Por otro lado, la RI tradicional se centra principalmente en modelos de ordenación sin supervisión, generalmente basados en el grado de correspondencia entre la consulta y el documento. Es el caso del booleano [314], del vectorial [271], del probabilístico [249], y de los asociados al modelado del lenguaje [236]. Teóricamente resultan sencillos e intuitivos, funcionan razonablemente bien y no requieren de datos etiquetados, una ventaja que no excluye la posibilidad de asociar un número de parámetros de ajuste mediante el uso de alguna técnica de entrenamiento, lo que no es inusual. Sin embargo, como los modelos de ordenación han visto incrementada su sofisticación, el conseguir ajustarlos convenientemente se ha convertido en una cuestión cada vez más difícil [350] y, en la práctica, estos enfoques empíricos sólo disponen de unos pocos parámetros que permitan ser afinados [17].

En contraste con los enfoques no supervisados, los supervisados disfrutan de una mayor precisión y una mejor adaptabilidad, al tiempo que requieren de un esfuerzo humano más importante, lo que durante muchos años limitó el interés práctico en este tipo de estrategias. Sin embargo, la disponibilidad actual de conjuntos etiquetados de evaluación de la relevancia realizados por grupos de expertos ofrecen una alternativa práctica para incorporar técnicas de aprendizaje automático en el diseño de modelos

de ordenación. La idea consiste en usar estos recursos etiquetados como medio de entrenamiento para estimar la proximidad semántica entre las consultas y los documentos [351] a través de la minimización de una *función de pérdida* indirectamente relacionada con determinadas medidas de rendimiento de la RI, como el *promedio de la precisión media*[12] (PPM) o la *la ganancia acumulativa reducida normalizada*[13] (GAARN), aunque también existen propuestas que permiten optimizar cualquiera de ellas [349]. En este sentido, se han descrito una gran variedad de estrategias de aprendizaje, tales como las redes neuronales [40, 46], las máquinas soportadas por vectores [45, 135, 136, 146, 316, 352], el «boosting» [102, 184, 349] o la programación genética [75, 81, 95, 311]. En la práctica, aunque estos métodos parecen funcionar mejor que los no supervisados tradicionales [183, 350], se pueden observar algunas diferencias importantes dependiendo del tipo de instancias utilizados en el aprendizaje. Más en detalle, se han abordado tres modelos diferentes de instanciación: punto a punto, por parejas, y por lista.

En el acercamiento punto a punto [184, 222], cada par de entrenamiento consulta-documento asocia una puntuación de manera independiente, lo que implica que no se consideran las preferencias relativas entre dos documentos recuperados para una misma consulta. Como consecuencia, el método ha demostrado tener un bajo rendimiento durante la fase de aprendizaje de la ordenación, transformando el problema en uno de regresión o de clasificación de un único documento [174]. En cambio, los basados en parejas [40, 45, 102, 136, 146, 174, 312, 351, 354] parecen ser los más populares. Los pares de documentos recuperados dada una consulta, en los que se ha determinado cuál de ellos es el más relevante, constituyen aquí las instancias del conjunto de entrenamiento. Así, el objetivo del proceso de aprendizaje es reducir al mínimo el número medio de inversiones en la ordenación, con el fin de obtener un clasificador binario que pueda indicar qué documento es mejor en un par dado. Esto implica que, dada una consulta, debemos inducir una ordenación total para un conjunto de documentos recuperados a partir de uno parcial entre pares, lo que limita severamente las posibilidades prácticas de este enfoque [32]. Por último, el modelo por lista [32, 41, 46, 176, 238, 348, 349, 352] también ha visto incrementado su popularidad en los últimos años. Considera el conjunto de documentos recuperados para una consulta como instancias en la fase de entrenamiento. Esto debería ser suficiente para superar los problemas anteriormente mencionados en relación con las técnicas punto a punto y por parejas y, de hecho, los resultados prácticos sugieren que éstas poseen cierta superioridad sobre las demás. Sin embargo, la definición de una función de pérdida puede convertirse en una tarea compleja porque la mayoría de las medidas de evaluación en RI no son magnitudes continuas con respecto a los parámetros del modelo de ordenación.

Finalmente, existe un amplio espectro de técnicas básicas de ordenación disponibles. Cada una de ellas tiene su propio conjunto de ventajas que deberíamos tratar de reconciliar mediante propuestas mixtas, y tener claro cuáles son los inconvenientes que se quieren

[12] en terminología anglosajona se denomina *mean average precision*.
[13] en terminología anglosajona se denomina *normalized discounted acumulative gain*.

evitar o al menos minimizar. A este respecto, probablemente la combinación de factores óptimos depende de la naturaleza de la tarea de búsqueda con la que queremos tratar. En nuestro caso, se refiere al tratamiento de un dominio específico. La afirmación de la existencia de claros beneficios derivados de la utilización de la similitud basada para la fase de búsqueda nos sitúa directamente en el contexto de algunos trabajos recientes [351], incorporando una dimensión de popularidad cuando el entorno de trabajo puede garantizar un número suficiente de accesos.

2.3 | Evaluación de la recuperación de la información

A la hora de evaluar un sistema de RI existen múltiples aspectos a tener en cuenta [20]: su eficiencia referida a sus costes espacio-temporales asociados, su efectividad a la hora de devolver el mayor número de documentos relevantes minimizando a la vez el número de no relevantes devueltos [314], el esfuerzo realizado por el usuario a la hora de formular o modificar su consulta; y la cercanía del interfaz de presentación de resultados en relación al esfuerzo requerido por el usuario para su interpretación.

Para calcular la relevancia, el acercamiento más simple es establecer valores binarios: un documento es relevante, es decir, sirve como respuesta a nuestra pregunta, (valor 1) o no sirve (valor 0), aunque también se puede fijar una gradación, y establecer una escala ordinal para medir la relevancia de los documentos [74]. El problema de determinar una escala es que no hay una guía clara para elaborarla. Por ejemplo Keen [161, 269], usa cuatro valores de escala, para dividir del más relevante al menos relevante. Saracevic [278] da tres valores a su escala: relevante, parcialmente relevante y no relevante, pero en la práctica distinguir entre un documento relevante y uno parcialmente relevante es muy difícil.

En este sentido, las técnicas basadas en JREL's y popularizadas por el TREC [332, 333] son consideradas como un estándar *de facto* para la evaluación en RI. Los eventos realizados por el TREC enfocan esta cuestión tomando como fondo común los 100 primeros documentos devueltos por cada sistema participante. Más tarde, estos documentos se revisan por especialistas que juzgan su relevancia, inspirándose en la metodología *Cranfield* [65, 64]. En definitiva, se trata de comparar los sistemas de RI con un conjunto de tópicos o consultas, una serie de documentos referidos a cada uno de ellos, y un conjunto de JREL's por cada documento. Este tipo de experimentación a gran escala ha sido el referente en este campo durante más de veinte años, denominándose *selección profunda*[14]. Sin embargo, el incremento del tamaño, de la complejidad y de la heterogeneidad de las colecciones documentales; así como del conjunto de consultas, lo han hecho inviable.

Por ello, se han propuesto una serie de enfoques alternativos para estimar el

[14] en terminología anglosajona se denomina *depth pooling*.

rendimiento de los sistemas de RI con recursos limitados de JREL'S, con el fin de reducir el esfuerzo humano en la creación de colecciones de prueba. El primero trata de conseguirlo seleccionando el mejor conjunto de documentos para ser evaluado y teniendo en cuenta medidas de calidad en aquellos casos en los que se pueden realizar pocos juicios. En esta categoría, podemos incluir como primera tentativa las técnicas de *selección*[15] [296], las cuales se centran en aquellos textos que menos probabilidades tienen de ser no relevantes. Sin embargo, trabajos recientes sugieren que el crecimiento en el tamaño de los *corpora* está superando incluso la capacidad de esta técnica para encontrar y juzgar suficientes documentos [39], ya que si se consideraran menos, las estimaciones de las medidas de evaluación tendrían una mayor varianza. En este sentido, algunos autores [52] tratan de reducir el esfuerzo necesario para juzgar a la vez que mantienen un gran número de tópicos, aunque reconocen que analizar los fallos resulta más complejo, por lo que esta vía necesita todavía seguir siendo explorada.

Una segunda alternativa relaja la carga de la valoración de tipo humano de la generación de JREL para introducir la noción de PJREL, los cuales se crean o bien aleatoriamente, seleccionando una correspondencia entre los documentos sobre los tópicos [290], o bien haciendo una lectura rápida de los situados en las posiciones más altas en la ordenación devuelta a partir de un subconjunto de representaciones de tópicos [89].

Por su parte, Mizzaro *et al.* [208] proponen un método de análisis de datos recogidos a partir de recursos de evaluación basados en JREL's o a partir de sistemas de RI similares, como es el caso de los PJREL's. Mediante la introducción de dos versiones normalizadas de PM que los autores usan para construir un grafo bipartito ponderado de motores de búsqueda y tópicos, encontraron que las medidas sobre la autoridad del sistema sirven para medir su rendimiento y que la conectividad revela la sencillez o complejidad de un tópico.

Finalmente, algunas propuestas [347] prescinden del concepto de JREL's, utilizando el solapamiento de los resultados obtenidos. Brevemente, la técnica pasa por interpretar la relación entre los documentos recuperados a partir de un grupo de sistemas de RI, donde dicha estructura de superposición parece proporcionar un fuerte impacto sobre los resultados. Así, se suele argumentar [298] que este tipo de métodos pueden producir malos resultados en los sistemas con mejor rendimiento cuando éstos se clasifican junto con los de menor rendimiento, a la vez que parece que obtienen peores resultados que el grupo anterior de técnicas.

Otro aspecto a tener en cuenta para definir un marco de pruebas formal en sistemas de RI es la elección adecuada de un conjunto de tópicos o consultas, con el fin de determinar cuáles son los mejores en la predicción de la relevancia real. El trabajo de investigación desarrollado al respecto es escaso y los resultados prácticos se limitan, *de facto*, a algunas ideas relacionadas con la hipótesis del trabajo y propuesta de estrategias de selección

[15] en terminología anglosajona se denomina *pooling*.

cuya validación requiere todavía una seria experimentación. En el apartado de hipótesis ya confirmadas, Mizzaro [207, 208] demuestra formalmente que algunos tópicos son más fáciles que otros y que existen diferencias entre los sistemas a la hora de distinguir entre los fáciles y los difíciles. Sin embargo, aunque podemos decir que no todos ellos son igualmente informativos sobre los sistemas de RI, no tenemos evidencias en cuanto a qué criterio podría ser mejor para calificar esta afirmación.

Estos trabajos en el campo de la evaluación de la RI sugieren de manera reiterada que los tópicos individuales varían enormemente en su capacidad para discriminar entre sistemas, lo cual provoca que se extienda la atención también en la construcción del propio conjunto de tópicos. Se trataría no sólo de discernir cuando un conjunto de este tipo es más útil que otro, siempre con un propósito de evaluación, sino también de seleccionar un número de ellos lo más pequeño posible sin que por ello pierdan esa cualidad. Ello permitiría reducir la carga de trabajo en una metodología cuyo principal problema es el coste, lo que justifica el interés práctico de este tipo de estrategias. Sin embargo, aunque desde hace muchos años ha existido preocupación por esta cuestión, no se han producido contribuciones relevantes hasta hace poco tiempo [121]. Los trabajos anteriores se basan exclusivamente en lo que debe ser la selección profunda, tomando como base metodológica algún tipo de enfoque heurístico [37, 276, 296, 336, 340, 356] que, por desgracia, proporciona para cada caso un resultado diferente. Con respecto a esto, aunque la propuesta de Guiver en [121], no intenta conseguir de inmediato una solución completa al problema de la identificación de conjuntos adecuados de tópicos, demuestra formalmente la existencia de fenómenos de complementariedad entre éstos y su influencia en la calidad de la evaluación, desechando la hipótesis de que se trate de un efecto aleatorio. El método se basa en el PPM [127]. Más en detalle, se aplica una búsqueda exhaustiva sobre todos los posibles subconjuntos de tópicos en un intervalo de cardinalidad. Para cada subconjunto, se calcula el correspondiente PPM, así como la correspondiente correlación sobre todos los tópicos con PPM. Los autores argumentan que los mayores valores de correlación (resp. menor) corresponden con los mejores (resp. los peores) conjuntos de tópicos. Sin embargo, el principal obstáculo para la aplicación directa de este método es el complejo análisis combinatorio que requiere, lo que implica poseer un amplio conjunto de tópicos evaluados y de sistemas asociados ejecutándose. De esta manera, la ganancia de tal reducción, puede ser relativamente pequeña para un esfuerzo importante y es necesario prever algún tipo de estrategia heurística a fin de evitar búsquedas completas en este espacio de trabajo.

PARTE II

Conceptos previos

CAPÍTULO III

Teoría de autómatas y lenguajes formales

El desarrollo de los ordenadores en la década de los 40, con la introducción de los programas en la memoria principal y posteriormente con los lenguajes de programación de alto nivel, propiciaron la distinción entre lenguajes formales, con reglas sintácticas y semánticas concretas y bien definidas, de los LN's o humanos, donde la sintaxis y la semántica no se pueden controlar tan fácilmente. En este sentido, el creciente interés en el tratamiento de estos últimos llevó a la construcción de gramáticas formales como un modo para su descripción, utilizando para ello reglas clásicas. Pero además de su formalización, también fue necesario el diseño de las máquinas abstractas adaptadas a su reconocimiento.

La descripción de una clase de lenguaje es equivalente a la de la clase de gramáticas que lo genera. En este sentido, existen diversas perspectivas. Chomsky [61] propuso su organización inicial en base a cuatro tipos de lenguajes, siguiendo la hoy denominada *Jerarquía de Chomsky*. Los cuatro tipos básicos son: *gramáticas recursivamente enumerables*, *gramáticas dependientes del contexto*, *gramáticas independientes del contexto* y *gramáticas regulares*. Las reglas son en sí mismas también escritas en un lenguaje formal definido por un vocabulario y una sintaxis.

3.1 | Definiciones básicas

Para llegar a la definición de los diferentes lenguajes incluidos en la Jerarquía de Chomsky, debemos introducir primero una serie de conceptos. Comenzamos con el más simple y a la vez uno de los más importantes, ya que a partir del mismo se definen y construyen buena parte de los demás.

Definición 3.1 *Un* alfabeto Σ *es un conjunto finito de elementos llamados* símbolos.

■

Como no puede ser de otra manera, la definición de alfabeto no difiere de nuestra concepción intuitiva. Tampoco lo hace la definición de cadena o palabra de un lenguaje que será una agrupación de símbolos del alfabeto.

Definición 3.2 *Una* cadena w *sobre un alfabeto* Σ *es una secuencia de cero o más símbolos del alfabeto. La cadena que no contiene símbolos se denomina* cadena vacía *y se representa como* ϵ*. El conjunto de todas las cadenas definidas sobre* Σ*, incluida* ϵ*, se designa por* Σ^**; su cierre transitivo.*

■

De esta manera, hemos definido satisfactoriamente los componentes básicos de un lenguaje, los símbolos y cadenas que lo forman. Podemos, entonces, centrarnos en el concepto mismo de lenguaje.

Definición 3.3 *Sea* Σ *un alfabeto, definimos un* lenguaje sobre Σ *como un subconjunto finito o infinito, de* Σ^*.

■

Ejemplo 3.1 *Con el alfabeto* $\Sigma = \{a, b\}$*, tenemos que* Σ^* *es el conjunto de cadenas formadas por los símbolos* «a» *y* «b». *Un posible lenguaje sobre* Σ^* *será el que está formado por cadenas de símbolos terminadas por* «b».

■

Estamos pues en posesión de una definición de lenguaje general. Ahora bien, una cosa es conocer lo que es un lenguaje y otra bien diferente es obtener una representación particular manejable. Un modo de lograr este objetivo es enumerar las cadenas que los forman. Pero este procedimiento no resulta muy práctico cuando el lenguaje consta de numerosas o infinitas cadenas, o pretendemos definir propiedades entre diferentes lenguajes. Por este motivo, surge la necesidad de establecer algún mecanismo para generar y representar lenguajes con una notación finita. Estos generadores son los que se denominan *gramáticas*, representaciones formales adaptadas al tratamiento computacional, que pasamos a definir.

Definición 3.4 *Una* gramática *se representa mediante una cuádrupla* $\mathscr{G} = (N, \Sigma, P, S)$*, donde:*

- *N es un alfabeto finito de* símbolos no terminales, *también denominados* variables. *Cada una de estas variables representan una* categoría sintáctica *de la gramática.*

- Σ *es un alfabeto finito de* símbolos terminales*, cada uno de los cuales representa una* categoría léxica *de la gramática. Por ejemplo las palabras* «hojas», «verde», «textura».
- P *es un conjunto finito de* reglas de producción[1] *de la gramática.*
- $S \in N$ *es el denominado* símbolo inicial, categoría inicial, raíz *o* axioma *de la gramática.*

■

En adelante, y para unificar criterios, se utilizará la notación que sigue para representar el conjunto de símbolos de una gramática:

- $V = N \cup \Sigma$, el conjunto total de símbolos gramaticales.
- $a, b, c, \ ... \ \in \Sigma$, los símbolos terminales.
- $A, B, C, \ ... \ \in N$, los símbolos no terminales.
- $..., \ X, Y, Z \in V$, símbolos arbitrarios.
- $..., \ x, u, v \in \Sigma^*$, cadenas de terminales.
- $\alpha, \beta, \gamma, \ ... \ \in V^*$, cadenas arbitrarias de símbolos terminales y no terminales.
- ϵ, la cadena vacía.

En este punto, se puede introducir el concepto de *derivación* de un símbolo no terminal. Se trata, en definitiva, de expresar la noción de descomposición de una categoría sintáctica compleja en otras más simples e incluso en categorías léxicas. Esto lleva al concepto de *derivación directa*.

Definición 3.5 *Se dice que* $\alpha\beta\gamma$ deriva directamente $\alpha\sigma\gamma$ *si y sólo si* $\beta \to \sigma \in P$*, y se usará la notación* $\alpha\beta\gamma \Rightarrow \alpha\sigma\gamma$

■

Extendiendo ahora el concepto, consideramos la noción de *derivación indirecta*.

Definición 3.6 *Se dice que* $\alpha\beta\gamma$ deriva indirectamente*, o simplemente que* deriva, $\alpha\sigma\gamma$ *si y sólo si* $\beta \Rightarrow \sigma_1 \Rightarrow \sigma_2 \Rightarrow \cdots \Rightarrow \sigma_n \Rightarrow \sigma$*, que notaremos como* $\alpha\beta\gamma \overset{+}{\Rightarrow} \alpha\sigma\gamma$.

■

[1] es un par ordenado que se compone de lado izquierdo (α) y de lado derecho (β), en la forma $\alpha \to \beta$.

3.2 | Jerarquía de Chomksy

Con lo expuesto anteriormente, ya podemos definir cada una de las gramáticas que componen la Jerarquía de Chomsky, comenzando con las que se sitúan en nivel más alto, y por tanto con un ámbito de aplicación más genérico, las gramáticas recursivamente enumerables, o también llamadas *gramáticas sin restricciones* [137].

Definición 3.7 *Formalmente, una* gramática recursivamente enumerable *(*GRE*) se define mediante una cuádrupla, $\mathscr{G} = (N, \Sigma, P, S)$, donde sus reglas de producción son de la forma:*

$$\alpha A \gamma \rightarrow \alpha w \gamma \; con \; A \in N, \; \alpha, \gamma, w \in (N \cup \Sigma)^*$$

Los lenguajes generados por este tipo de gramáticas se llaman lenguajes recursivamente enumerables *(*LRE*).*

■

Puede probarse que, una GRE [137] es una gramática formal para la cual además existe una *máquina de Turing* [313] (MT) que acepta cualquier cadena del lenguaje por ellas generado, pero que puede parar para aceptar o rechazar, o bien iterar indefinidamente, según la cadena pertenezca o no al lenguaje, o simplemente sea una cuestión no decidible. No existe ninguna restricción sobre las producciones. Más adelante, se dará una definición de MT.

Definición 3.8 *Formalmente, una* gramática dependiente del contexto (GDC) *se define mediante una cuádrupla, $\mathscr{G} = (N, \Sigma, P, S)$, donde las reglas de producción tienen una de las dos formas siguientes:*

- $\alpha A \gamma \rightarrow \alpha w \gamma \; con \; A \in (N \cup \{S\}), \; \alpha, \gamma \in (N \cup \Sigma)^*, \; w \in (N \cup \Sigma)^* - \{\epsilon\}$
- $S \rightarrow \epsilon$

con $|\alpha A \gamma| \leqslant |\alpha w \gamma|$, siendo $|\alpha A \gamma|$ el número de símbolos en $\alpha A \gamma$.

Los lenguajes generados por este tipo de gramáticas se llaman lenguajes dependientes del contexto (LDC).

■

Operacionalmente, este tipo de gramáticas necesita de un *autómata linealmente acotado*[2] [137, 221] (ALA) para su tratamiento, lo que aún supone unos niveles de

[2] en terminología anglosajona se denomina *linear bounded automaton*.

complejidad temporal y espacial para su tratamiento que son elevados y poco prácticos. Como en el caso de las MT's, también se va a proporcionar más adelante una definición de ALA.

El hecho de que la parte izquierda de la producciones sólo pueda contener una cadena de símbolos terminales y no terminales de longitud menor o igual que la parte derecha asegura que, al aplicar una derivación sobre una *forma sentencial*[3], se obtiene otra de igual o mayor longitud. A continuación, vamos a ilustrarlo mediante el Ejemplo 3.2.

Ejemplo 3.2 *Sea* $\mathscr{G} = (\{a, b, c\}, \{S, M\}, P, S)$ *donde:*

$$P = \left\{ \begin{array}{l} S \rightarrow aMc|aSMc, \\ aM \rightarrow ab, \\ bM \rightarrow bb, \\ cM \rightarrow Mc, \end{array} \right\}$$

En las GDC*'s existen reglas en las que un símbolo no terminal puede derivar a formas sentenciales distintas, según los símbolos que aparezcan en su contexto. Si observamos la segunda regla de producción, el no terminal* «M» *puede ser sustituido por el terminal* «b» *manteniendo el contexto que poseía, que no es otro que* «a»*, ya que coincide tanto en el lado derecho como en el izquierdo. Lo mismo ocurre con las otras reglas de producción. Además de esto, se cumple que la longitud de la cadena de la parte izquierda es inferior o igual a la de la parte derecha.*

La gramática $\mathscr{G}$ *genera el* LDC $\mathscr{L}(\mathscr{G}) = \{a^n b^n c^n / n > 0\}$*. Un ejemplo de derivación sería:*

$$S \Rightarrow aSMc \Rightarrow aaMcMc \Rightarrow aabcMc \Rightarrow aabMcc \Rightarrow aabbcc$$

■

Por debajo de las GDC's, Chomsky sitúa las gramáticas independientes del contexto [137], incapaces de mostrar derivaciones contextuales, pero muy eficaces computacionalmente al poder implementar su reconocimiento mediante un *autómata de pila* (AP) [137] que definiremos en la siguiente sección.

Definición 3.9 *Formalmente una* gramática independiente del contexto (GIC) *se define mediante una cuádrupla* $\mathscr{G} = (N, \Sigma, P, S)$*, donde los elementos de* P *son de la forma:*

$$A \rightarrow \gamma, \ con \ A \in N, \ \gamma \in (\Sigma \cup N)^*$$

[3]se dice que $\alpha \in V^*$ es una *forma sentencial* para $\mathscr{G}$, si puede obtenerse de una secuencia de derivaciones $S \stackrel{+}{\Rightarrow} \alpha$. Decimos que $x \in \Sigma^*$ es una *sentencia* si y sólo si $S \stackrel{+}{\Rightarrow} x$.

Los lenguajes generados por este tipo de gramáticas se llaman lenguajes independientes del contexto (LIC).

■

El hecho de que sus producciones tengan un único símbolo no terminal en la parte izquierda asegura que, a la hora de realizar un paso de derivación directo, es posible decidir qué símbolo no terminal queremos reescribir, independientemente del contexto que lo rodea. Una muestra de GIC es la mostrada en el Ejemplo 3.3.

Ejemplo 3.3 *Sea* $\mathscr{G} = (\{a, b, c, d\}, \{S, A, B, C\}, P, S)$ *donde:*

$$P = \left\{ \begin{array}{l} S \to AB|aCd, \\ A \to ab|aAb, \\ B \to cd|cBd, \\ C \to AbcB|bc, \end{array} \right\}$$

La gramática $\mathscr{G}$ *genera el* LIC $\mathscr{L}(\mathscr{G}) = \{a^n b^n c^m d^m / \ n, m \geqslant 1\}$. *De este modo podemos obtener para la sentencia «*aabbccdd*» los árboles de la Fig. 3.1.*

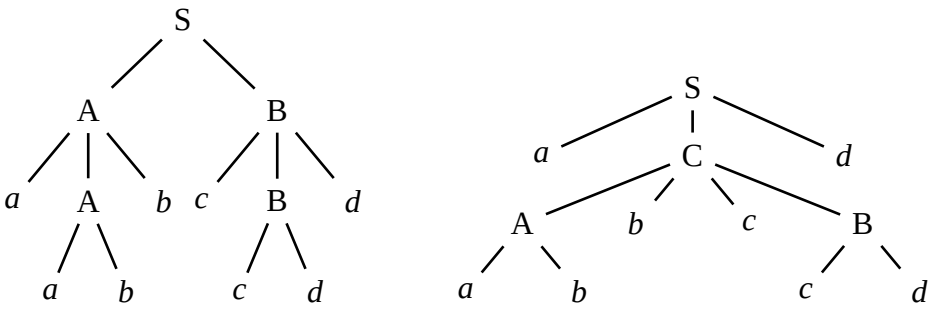


Figura 3.1: Algunos árboles derivados $\{a^n b^n c^m d^m / \ n, m \geqslant 1\}$.

Resultado de las derivaciones siguientes:

$S \Rightarrow AB \Rightarrow aAbB \Rightarrow aabbB \Rightarrow aabbcBd \Rightarrow aabbccdd$
$S \Rightarrow aCd \Rightarrow aAbcBd \Rightarrow aabbccdd$

■

En el nivel más bajo de su jerarquía, Chomsky sitúa a las gramáticas regulares, cuyo reconocimiento requiere tan sólo de un *autómata finito* (AF) [137] y, por tanto, extremadamente eficaces desde el punto de vista computacional. Por otro lado, hay que destacar su limitada expresividad debido a la carencia de estructuras memorísticas asociadas, como las pilas, lo que le impide realizar operaciones triviales en otros formalismos. Es, por ejemplo, el caso de las contabilizaciones en el número de elementos derivados.

Definición 3.10 *Formalmente, una* gramática regular (GR) *se define mediante una cuádrupla* $\mathscr{G} = (N, \Sigma, P, S)$*, donde sus reglas de producción son de la forma:*

- *En caso de ser regulares por la derecha,* $A \to aB$ *ó* $A \to a$
- *En caso de ser regulares por la izquierda,* $A \to Ba$ *ó* $A \to a$
- $S \to \epsilon$

con $a \in \Sigma^*$*,* $A, B \in N$*.*

Los lenguajes generados por este tipo de gramáticas se llaman lenguajes regulares (LR)*.*

■

Ejemplo 3.4 *Sea* $\mathscr{G} = (\{a, b\}, \{S, A, B\}, P, S)$*, donde:*

$$P = \left\{ \begin{array}{l} S \to aA, \\ A \to aA|bB, \\ B \to bB|\epsilon, \end{array} \right\}$$

La gramática $\mathscr{G}$ *genera el* LR $\mathscr{L}(\mathscr{G}) = \{a^n b^m / n, m \geqslant 1\}$*. Ejemplos de árboles resultantes de la derivación serían los mostrados en la Fig. 3.2:*

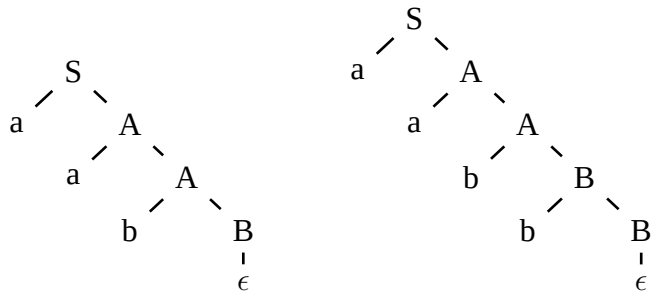


Figura 3.2: Algunos árboles derivados $\{a^n b^m \ / \ n, m \geqslant 1\}$

■

Si nos centramos en los lenguajes que acabamos de describir en esta Jerarquía, podemos clasificarlos de mayor a menor genericidad de forma que cada nivel es incluido en los lenguajes del nivel anterior. Dicho esto, pasemos ahora a comentar conceptos básicos de la teoría de autómatas.

3.3 | Teoría de autómatas

La teoría de autómatas es una rama de las ciencias de la computación que estudia las máquinas abstractas y los problemas que éstas son capaces de resolver. Los *autómatas* son, por tanto, reconocedores para las estructuras gramaticales previamente descritas.

3.3.1 | Autómata finito

De este modo comenzaremos describiendo un AF como un modelo matemático de una máquina, con entradas y salidas discretas [137, 229] sobre un alfabeto Σ, que se define de la siguiente manera.

Definición 3.11 *Un* autómata finito (AF) *se define como una 5-tupla* $\mathcal{A} = (Q, \Sigma, \delta, q_0, Q_F)$*, donde:*

- Q *es un conjunto finito de* estados*, no vacío.*
- Σ *es un alfabeto finito de* símbolos terminales *de entrada.*
- δ *es una* función de transición*, definible como un conjunto de arcos o* transiciones *que constan de un estado origen, un estado destino y un símbolo terminal de entrada, es decir:*
  - $\delta : Q \times \Sigma \to Q$*, si el autómata es* finito y determinista (AFD)*, es decir, si es* $\forall q \in Q, \forall a \in \Sigma, |\delta(q, a)| \leq 1$.
  - $\delta : Q \times \Sigma \to \mathcal{P}(Q)$*, siendo* $\mathcal{P}(Q)$ *el conjunto de partes de* Q*, si el autómata es* finito no determinista (AFND)*. Esto es,* $\exists q \in Q, a \in \Sigma, |\delta(q, a)| > 1$.
- q_0 *es el* estado inicial *del autómata, donde* $q_0 \in Q$.
- Q_F *es el conjunto de* estados finales*, no vacío, donde* $Q_F \subseteq Q$.

Se denota por $\mathscr{L}(\Sigma)$ *el lenguaje reconocido por el* AF*, es decir, el conjunto de todas las palabras* «w» *tales que* $\delta(q_0, w) \in Q_F$.

■

Ejemplo 3.5 *Supongamos que tenemos un* AF *con los siguientes componentes:*

- $Q = \{q_0, q_1, q_2\}$,
- $\Sigma = \{0, 1\}$,
- *La función de transición se describe a continuación en la Tabla 3.1,*

δ	0	1
q_0	$\{q_1\}$	$\{q_1, q_2\}$
q_1	$\varnothing$	$\varnothing$
q_2	$\{q_1\}$	$\varnothing$

Tabla 3.1: Función de transición de un AF

- *El estado inicial es* q_0,
- $Q_F = \{q_1\}$.

Es posible representar un AF *como un grafo dirigido en el que los estados serán los nodos y las transiciones desde un estado* «p» *a uno* «q» *mediante el símbolo de entrada* «a» *se representan mediante un arco dirigido desde el nodo que representa al estado* «p» *hacia el nodo que representa al estado* «q», *etiquetada con el símbolo* «a». *Para distinguir los estados finales, éstos se representan con doble círculo mientras que el estado inicial se marcará mediante la punta de una flecha, como se puede observar en la Fig. 3.3.*

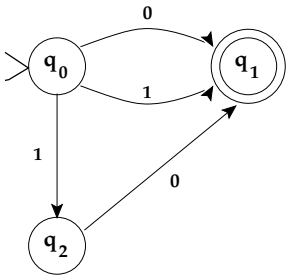


Figura 3.3: AF de ejemplo

■

Intuitivamente, un AF no es más que un conjunto de estados interconectados y transitables. Podemos decir que el proceso de reconocimiento de una palabra consiste en encontrar su traza, de forma que si ésta termina en un estado final, la palabra es reconocida, y si no rechazada. Si ocurriese que en algún momento es posible transitar a más de un estado con el mismo carácter entonces se tratará de un AFND. En caso contrario, será un AFD.

3.3.2 | Autómata de pila

Tomando como referencia la Jerarquía de Chomsky y sus formalismos gramaticales, el siguiente nivel de complejidad en lo que a tratamiento operacional se refiere es el de los AP's [137].

Definición 3.12 *Formalmente, un* autómata de pila (AP) *se define como una tupla* $(Q, \Sigma, \Gamma, q_0, \delta, Z, Q_F)$*, donde:*

- Q *es un conjunto finito de* estados.
- Σ *es el* alfabeto de terminales de entrada.
- Γ *es el* alfabeto de la pila.
- q_0 *es el* estado inicial*, donde* $q_0 \in Q$.
- δ *es una* función de transición *que define las transiciones del autómata de* $Q \times (\Sigma \cup \{\epsilon\}) \times \Gamma$ *en* $Q \times \Gamma^*$*, que define las transiciones válidas del autómata.*
- Z *es el* símbolo inicial de pila*, donde* $Z \in \Gamma$.
- Q_F *es el conjunto de* estados finales*, donde* $Q_F \subseteq Q$.

■

De este modo, un AP cuenta con una cinta de entrada y un mecanismo de control que puede encontrarse en uno de entre un número finito de estados. A diferencia de los AF's, éstos cuentan con una memoria auxiliar llamada *pila*, donde se pueden insertar o extraer símbolos.

Ejemplo 3.6 *Supongamos que tenemos un* AP *con los siguientes componentes:*

- $Q = \{q_0, q_1, q_2, q_3\}$,
- $\Sigma = \{a, b\}$,
- $\Gamma = \{A, Z\}$,
- *El estado inicial es* q_0,
- *La función de transición se describe a continuación:*
  - $\delta(q_0, a, Z) = (q_1, A\ Z)$
  - $\delta(q_1, a, Z) = (q_1, A\ A)$
  - $\delta(q_1, b, A) = (q_2, \epsilon)$
  - $\delta(q_2, b, A) = (q_2, \epsilon)$
  - $\delta(q_2, \epsilon, Z) = (q_3, Z)$
- *El símbolo inicial es* Z

- $Q_F = \{q_3\}$.

Al igual que en el Ejemplo 3.5, es posible representar un AP *como un grafo dirigido en el que los estados serán los nodos y los arcos las transiciones, como se ve en la Fig. 3.4.*

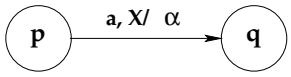


Figura 3.4: AP de ejemplo

Si existe una transición entre el estado «p» *a uno* «q»*, y la cabeza lectora apunta a un símbolo* «a»*, y el tope de la pila es* «X»*, entonces cambiar al nuevo estado* «q» *consiste en avanzar la cabeza lectora y sustituir el símbolo del tope* «X» *en la pila por la cadena* α*. Concretamente, la Fig. 3.5 ilustra para los componentes descritos, el* AP *resultante.*

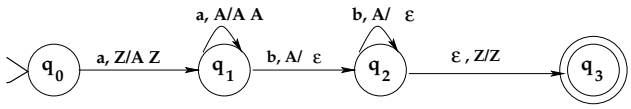


Figura 3.5: AP de ejemplo

Por ejemplo, si el estado actual es q_0 *y la cabeza lectora apunta sobre el símbolo* «a»*, y la cima de la pila es* «A»*, entonces cambiar al nuevo estado* q_1 *implica avanzar la cabeza lectora, y sustituir el símbolo del tope* «A» *en la pila por la cadena* «A Z».

■

Intuitivamente, y de forma similar a un AF, un AP es un conjunto de estados interconectados y transitables, en los que el proceso de reconocimiento de una cadena se hace efectiva en función de si la secuencia de transiciones, comenzando en el estado inicial y con pila vacía, conduce a un estado final y con pila también vacía, después de leer toda la cadena.

3.3.3 | Autómata linealmente acotado

Un ALA [137] es un autómata que incluye en su alfabeto de entrada dos símbolos especiales más: el de inicio de cinta (#) y el de fin ($), denominados *marcadores finales* izquierdo y derecho respectivamente. Además, en los ALA's no existen movimientos a la izquierda de # ni a la derecha de $, ni siquiera se puede escribir otro símbolo sobre ellos. En definitiva, se trata de que en lugar de tener un cinta infinita sobre la cual escribir, se restringe a la porción de la cinta que contiene a la entrada, más los dos marcadores finales.

Definición 3.13 *Formalmente, un* autómata linealmente acotado (ALA) *se define como una 7-tupla* $(Q, \Sigma, \Gamma, q_0, \delta, \lambda, \#, \$, Q_F)$ *donde:*

- Q *es un conjunto finito de estados.*
- Σ *es el* alfabeto de terminales de entrada.
- Γ *es el* alfabeto de la cinta.
- q_0 *es el* estado inicial, *donde* $q_0 \in Q$.
- λ *es el* símbolo blanco, *donde* $\lambda \notin \Sigma,\ \lambda \in \Gamma$.
- $\delta : Q \times \Gamma \to Q \times \Gamma \times \{I, D\}$ *es una función de transición, donde* I *es un movimiento a la izquierda y* D *es el movimiento a la derecha.*
- Q_F *es un conjunto de estados finales.*
- $\#$ *es el símbolo inicial de la cinta, con* $\delta(q_n, \#) = (q_n, \#, I)$, *y* $\$$ *es el símbolo final de la cinta, con* $\delta(q_n, \$) = (q_n, \$, D)$.

■

En la práctica vamos a usar este tipo de autómata para reconocer LDC's, realizando el cálculo en las únicas celdas de la cinta que están originalmente ocupadas por la cadena de entrada.

Ejemplo 3.7 *Supongamos que consideramos el* ALA *definido por:*

- $Q = \{q_0, q_1, q_2, q_3, q_4, q_5\}$,
- $\Sigma = \{a, b\}$,
- $\Gamma = \{a, b, \vdash, \#, \$\}$,
- $Q_F = \{q_4\}$,
- δ *se define como:*
  - $\delta(q_0, \#) = (q_0, \#, D)$
  - $\delta(q_0, a) = (q_1, \#, D)$
  - $\delta(q_1, a) = (q_1, a, D)$
  - $\delta(q_1, b) = (q_2, b, D)$,
  - $\delta(q_2, b) = (q_2, b, D)$,
  - $\delta(q_2, \$) = (q_3, \$, I)$,

- $\delta(q_3, b) = (q_4, \$, I)$,
- $\delta(q_4, b) = (q_5, b, I)$,
- $\delta(q_5, a) = (q_5, b, I)$,
- $\delta(q_5, \#) = (q_0, \#, D)$,

Este ALA *acepta el lenguaje* $\mathscr{L}(\mathscr{G}) = \{a^n b^n / \ n \in \mathbb{N}\}$*, representado en el grafo de la Fig. 3.6.*

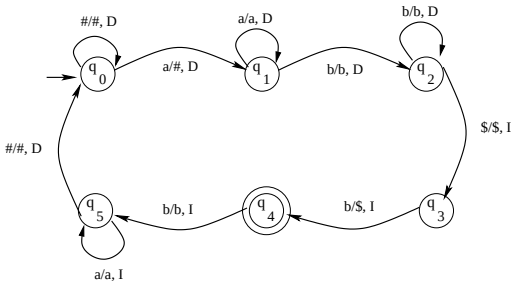


Figura 3.6: ALA de ejemplo

Hay que recalcar que aunque puede reconocer y trabajar sobre los símbolos $ *y* #*, no puede reemplazarlos o moverse más allá de ellos. Además, si suponemos que w=aabb, entonces aplicando los pasos que se muestran en la Fig. 3.7, se concluye que el autómata acepta dicha cadena.*

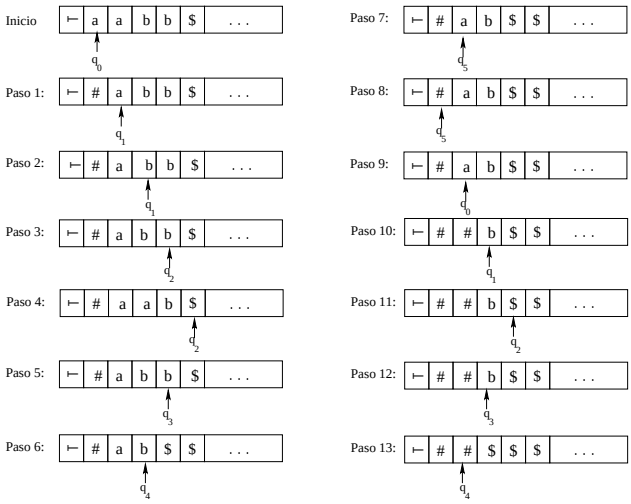


Figura 3.7: Pasos seguidos por un ALA de ejemplo

■

3.3.4 | Máquina de Turing

El modelo general de MT [137] permite aceptar los LRE que incluyen todo el conjunto de lenguajes que describen procedimientos computacionales. Su modelo básico tiene un mecanismo de control, una cinta de entrada que se divide en celdas, y una cabeza de lectura/escritura que lee un sólo símbolo de la cinta a la vez. La cinta tiene una celda de inicio, situada en la posición más a la izquierda e infinitas a la derecha. La diferencia fundamental con el AP y el AF, es que se puede leer un símbolo y reescribirlo por otro símbolo, y además la cabeza de lectura/escritura puede desplazarse a la izquierda o a la derecha. En principio todas las celdas que no se hayan escrito antes contienen un carácter especial nulo o blanco (que se representa por λ).

Definición 3.14 *Formalmente, una* máquina de Turing (MT) *se define como una 7-tupla* $(Q, \Sigma, \Gamma, q_0, \delta, \lambda, Q_F)$ *donde:*

- Q *es un conjunto finito de estados.*
- Σ *es el* alfabeto de terminales de entrada.
- Γ *es el* alfabeto de la cinta.
- q_0 *es el* estado inicial*, donde* $q_0 \in Q$.
- λ *es el* símbolo blanco*, donde* $\lambda \notin \Sigma,\ \lambda \in \Gamma$.
- $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{I, D\}$ *es una función de transición, donde* I *es un movimiento a la izquierda y* D *es el movimiento a la derecha.*
- Q_F *es un conjunto de estados finales.*

■

Ejemplo 3.8 *Queremos construir una máquina que verifique si el número de 0s en una palabra es par:*

- $Q = \{q_0, q_1\}$.
- $\Sigma = \{0, 1\}$.
- $\Gamma = \{0, 1, \vdash, \lambda\}$.
- $Q_F = \{q_0\}$.
- δ *se define como:*
  - $\delta(q_0, 0) = (q_1, \lambda, D)$.

- $\delta(q_0, 1) = (q_0, \lambda, D)$.
- $\delta(q_1, 0) = (q_0, \lambda, D)$.
- $\delta(q_1, 1) = (q_1, \lambda, D)$.

El alfabeto sólo dispone de dos símbolos: el 0 *y el* 1*. La máquina puede adoptar dos estados diferentes, donde el primero es el inicial, que a la vez hace de estado final. También es posible representar la función de transición mediante el grafo de la Fig. 3.8.*

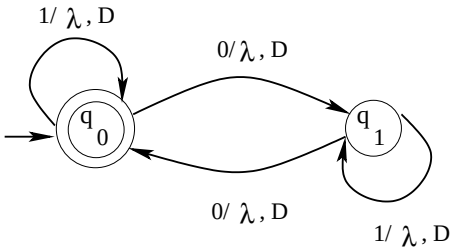


Figura 3.8: MT de ejemplo

Si suponemos que w=00010, entonces aplicando los pasos que se muestran en la Fig. 3.9, se concluye que la máquina acepta dicha cadena.

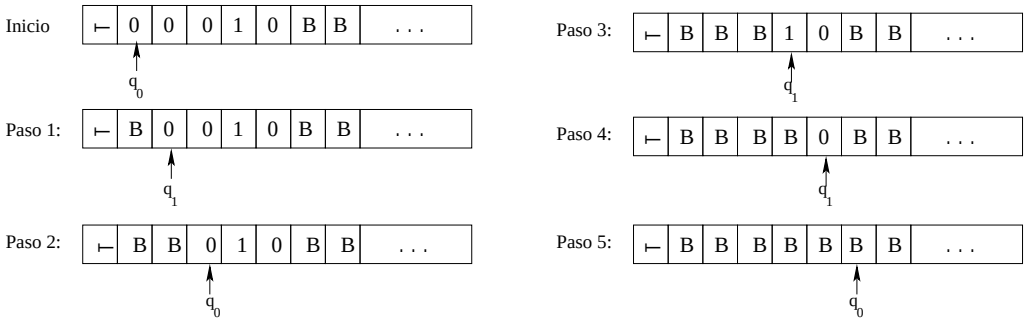


Figura 3.9: Pasos seguidos por una MT de ejemplo

■

CAPÍTULO IV

Teoría de grafos

Introducimos una serie de nociones y notaciones relacionadas con esta teoría. Empezaremos por las definiciones básicas asociadas a los grafos en relación a sus componentes, a la variedad existente y a su representación. Más tarde, con estos elementos estaremos en disposición de definir cómodamente el formalismo catalogado dentro de las estructuras que permiten representar conocimiento por medio de conceptos y descripciones a través de símbolos lógicos, es decir, los GC's.

4.1 | Definiciones básicas

Informalmente, un *grafo* [342] es un conjunto de objetos llamados *vértices* o *nodos* a partir de los cuales es posible representar relaciones binarias entre ellos.

Definición 4.1 *Un* grafo *se representa mediante un par* $\mathcal{G} = (\mathcal{V}, \mathcal{A})$*, donde:*

- $\mathcal{V}$ *es un conjunto finito, tal que* $\mathcal{V} \neq \emptyset$*, llamado* vértices o nodos.
- $\mathcal{A}$ *es un conjunto de pares de nodos de la forma* $\{x, y\}$*, tal que* $x, y \in \mathcal{V}$ *y* $\mathcal{V} \cap \mathcal{A} = \emptyset$*, llamados* aristas *o* arcos.

Además, se dice que un vértice y una arista son incidentes *si el vértice es uno de los extremos de la arista. También se dice que en una arista* $\{x, y\}$*, los dos vértices «*x*» e «*y*» son* adyacentes.

Por otro lado, se dice que dos aristas del grafo son independientes *si no tienen vértices en común. Finalmente, se le llama* orden *de* $\mathcal{G}$ *al número de vértices* $|\mathcal{V}|$.

■

Definición 4.2 *Sea $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ un grafo. Se dice que $\mathcal{G}$ es un* grafo no dirigido *si $\mathcal{A}$ es un conjunto de pares no ordenados de nodos $\{x, y\}$, tal que $x, y \in \mathcal{V}$ y $\mathcal{V} \cap \mathcal{A} = \emptyset$.*

■

Si $a = \{x, y\}$ es una arista entonces se dice que los vértices «*x*» e «*y*» son los *extremos* de «*a*». Al ser $\mathcal{A}$ un conjunto de pares no ordenados, la arista $\{x, y\} = \{y, x\}$.

Ejemplo 4.1 *Sea $\mathcal{V} = \{a, b\}$ y $\mathcal{A} = \{\{a, b\}\}$. Entonces $(\mathcal{V}, \mathcal{A})$ es un grafo con dos vértices y una arista. La Fig. 4.1 es su representación gráfica. Así, el par $\{a, b\}$ representa a la misma arista que $\{b, a\}$. Del mismo modo, los nodos* «a» *y* «b» *son a su vez, los extremos de dicha arista.*

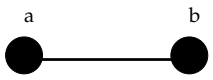


Figura 4.1: Grafo no dirigido de ejemplo

■

Otra generalización del concepto de grafo es el que hace referencia al sentido de las aristas. Definamos entonces *grafo dirigido* o *digrafo*.

Definición 4.3 *Sea $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ un grafo. Se dice que $\mathcal{G}$ es un* grafo dirigido *o* digrafo *si $\mathcal{A}$ es un conjunto de pares ordenados de nodos $\{x, y\}$, tal que $x, y \in \mathcal{V}$ y $\mathcal{V} \cap \mathcal{A} = \emptyset$, llamados* arcos*, tal que $\{x, y\} \neq \{y, x\}$. En este sentido, a* «x» *se le llama* origen *y a* «y» *se le llama* extremo*.*

■

Todo digrafo tiene un grafo no dirigido subyacente, que se obtiene olvidando el sentido de los arcos y considerándolos como aristas no orientadas. Ilustrémoslo mediante el Ejemplo 4.2.

Ejemplo 4.2 *Consideremos el digrafo $\mathcal{G} = (\{a, b, c, d, e, f, g, h\}, \{\{b, a\}, \{a, h\}, \{b, g\}, \{h, g\}, \{g, f\}, \{d, b\}, \{c, d\}, \{f, d\}, \{e, c\}\})$ de la Fig. 4.2.*

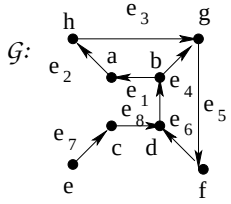


Figura 4.2: Grafo dirigido de ejemplo

En él, se puede observar como los arcos dirigidos están indicados mediante flechas. El arco e_1 está asociado al par ordenado de vértices $\{b, a\}$ por lo que se escribe $e_1 = \{b, a\}$ y el arco e_7 con el par ordenado $\{e, c\}$.

En este sentido, el grafo subyacente que se obtiene de $\mathcal{G}$, si obviamos el sentido de los arcos, es el que se muestra en la Fig. 4.3.

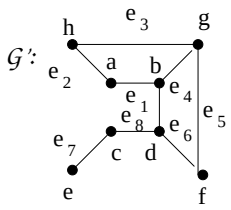


Figura 4.3: Grafo no dirigido obtenido a partir de un digrafo de ejemplo

■

Si quisiéramos utilizar sólo una parte de un grafo, sería necesario echar mano de la noción de *subgrafo*. Veamos su definición, y a continuación ilustrémoslo con el Ejemplo 4.3.

Definición 4.4 *Si $\mathcal{G} = (\mathcal{V}_1, \mathcal{A}_1)$ y $\mathcal{H} = (\mathcal{V}_2, \mathcal{A}_2)$ son grafos tales que $\mathcal{V}_2 \subset \mathcal{V}_1$ y $\mathcal{A}_2 \subset \mathcal{A}_1$, entonces se dice que $\mathcal{H}$ es un* subgrafo *de $\mathcal{G}$ y, en correspondencia, que $\mathcal{G}$ es un* supergrafo *de $\mathcal{H}$.*

■

Ejemplo 4.3 *Sea el grafo no dirigido $\mathcal{G} = (\{a, b, c, d, e, f, g, h\}, \{\{a, b\}, \{a, c\}, \{b, d\}, \{c, d\}, \{a, h\}, \{b, g\}, \{d, f\}, \{c, e\}, \{h, e\}, \{h, g\}, \{f, g\}, \{e, f\}\})$ de la Fig. 4.4.*

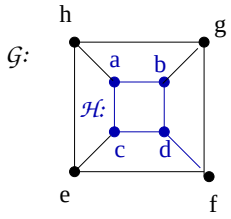


Figura 4.4: Subgrafo de ejemplo

Entonces $\mathcal{H} = (\{a, b, c, d\}, \{\{a, b\}, \{b, d\}, \{a, c\}, \{c, d\}\})$ es un subgrafo de $\mathcal{G}$, porque $\{a, b, c, d\} \subset \{a, b, c, d, e, f, g, h\}$, pero también $\{\{a, b\}, \{b, d\}, \{a, c\}, \{c, d\}\} \subset \{\{a, b\}, \{a, c\}, \{b, d\}, \{c, d\}, \{a, h\}, \{b, g\}, \{d, f\}, \{c, e\}, \{h, e\}, \{h, g\}, \{f, g\}, \{e, f\}\}$.

■

4.1.1 | Valencia o grado de un vértice

Cada uno de los vértices del grafo pueden poseer un número propio de aristas que inciden en ellos, de donde el concepto de *valencia o grado*. En este sentido, es necesario proporcionar dos definiciones en función de si se trabaja con grafos dirigidos o no. Comencemos entonces en el caso de que el grafo no sea dirigido.

Definición 4.5 *Llamamos* valencia o grado *de un vértice v en un grafo no dirigido $\mathcal{G}$ al número $g(v)$ de aristas incidentes con él. Si $g(v) = 0$ se dice que x es un vértice* aislado.

■

En el caso de tratar con grafos dirigidos, es necesario definir dos conceptos más: las *valencias de entrada* y *de salida*.

Definición 4.6 *Sea $\mathcal{G}$ un grafo dirigido. Llamaremos* valencia o grado de salida *de un vértice v, y lo denotaremos por $gs(v)$, al número de arcos salientes de v. Llamaremos* valencia o grado de entrada *de un vértice v, y lo denotaremos por $ge(v)$, al número de arcos entrantes en v. Finalmente, se denominará* valencia o grado *de un vértice v, denotado por $g(v)$, a la suma de estos dos grados, es decir*

$$g(v) = gs(v) + ge(v)$$

■

En el grafo del Ejemplo 4.1 se tiene $g(a) = 1$ y $g(b) = 1$. La *sucesión de valencias* de un grafo se obtiene ordenando en forma no decreciente las valencias de todos los vértices. En ese ejemplo, la sucesión de valencias es {1, 1}.

4.1.2 | Camino y conexión de un grafo

Si en un grafo tratamos de transitar por los diversos vértices a través de las aristas (resp. arcos) que inciden en ellos, recorremos lo que se denomina el *camino* [342].

Definición 4.7 *Se denomina* camino de longitud n *de un grafo $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ no dirigido, a una sucesión de vértices unidos por las aristas $x_0a_0x_1 \ldots x_{k-1}a_{k-1}x_k$, donde $x_i \in \mathcal{V}$ y $a = \{x_i, x_{i+1}\} \in \mathcal{A}$ para $0 \leq i \leq k-1$, de forma que no se repite ninguna de ellas. Los vértices x_0 y x_k son los extremos del camino. Observar que un grafo con un solo vértice es un camino de longitud 0.*

Si $x_0 = x_k$, el camino se dice cerrado, *de lo contrario se dice* abierto. *Decimos que un camino es un* ciclo *si todos los vértices (excepto los extremos) son distintos.*

■

En grafos simples, en los que no existe ambigüedad, los caminos suelen omitir en su sucesión a las aristas. Sin embargo, en el caso de que el grafo sea más complejo, es decir, que permita múltiples aristas entre un mismo par de vértices, sí resultan necesarias.

Estos conceptos son los mismos para grafos dirigidos salvo que las direcciones de los arcos deben concordar con la dirección del camino. En el caso dirigido, el ciclo recibe el nombre de *circuito*. Dicho esto, podemos dar una definición de todos aquellos grafos que no posean ciclos (resp. circuitos) e ilustrarlo mediante el Ejemplo 4.4.

Definición 4.8 *Un grafo $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ es* acíclico *si no contiene ningún ciclo.*

■

Definición 4.9 *Sea $\mathcal{G}$ un grafo. Se dice que dos vértices «u» y «v» están* conectados *si existe un camino de «u» a «v».*

■

Ejemplo 4.4 *El grafo $\mathcal{G}$ no dirigido de la Fig. 4.5 contiene seis ciclos: $\{abgha\}, \{bdfgb\}, \{cdfec\}, \{abdfgha\}, \{bdcefgb\}$ y $\{abdcefgha\}$.*

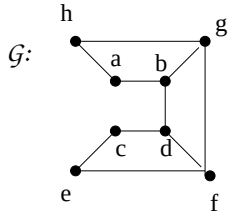


Figura 4.5: Ciclos en grafo de ejemplo

Del mismo modo, el vértice «a» está conectado al vértice «c» ya que existen varios caminos entre ellos. Un ejemplo podría ser: $\{abdc\}$.

■

Definición 4.10 *Sea un grafo $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ no dirigido. Se dice que $\mathcal{G}$ es* conexo*, si para cualquier par de vértices de $\mathcal{G}$, existe al menos un camino entre ellos, es decir, si existen una o más aristas que lleven del primer vértice al segundo.*

■

Definición 4.11 *Un grafo dirigido $\mathcal{G}$ es* débilmente conexo *si su grafo no dirigido asociado es conexo.*

■

Ejemplo 4.5 *Si tomamos las Fig. 4.1 y 4.7, éstas muestran grafos conexos. En cambio, la Fig. 4.6 no lo es, ya que no existe ningún camino que lleve del vértice «*b*» al vértice «*d*».*

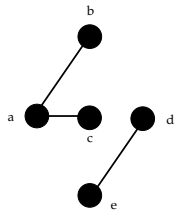


Figura 4.6: Grafo no conexo de ejemplo

■

4.1.3 | Grafos particulares

A continuación, describimos algunos tipos de grafos con los que nos podemos encontrar, tales como, los *grafos bipartitos*, los *grafos simples* y *multigrafos*, así como sus características.

Definición 4.12 *Un* grafo bipartito *es un grafo no dirigido cuyos vértices se pueden separar en dos conjuntos disjuntos $\mathcal{V}_1$ y $\mathcal{V}_2$, denotándose por $\mathcal{G} = (\mathcal{V}_1 \cup V_2, \mathcal{A})$ y cuyas aristas siempre unen vértices de un conjunto, con vértices de otro, es decir:*

- $\mathcal{V}_1 \cup \mathcal{V}_2 = \mathcal{V}$ *y* $\mathcal{V} \neq \emptyset$.
- $\mathcal{V}_1 \cap \mathcal{V}_2 = \emptyset$.
- $\forall x \in \mathcal{V}_1, \forall y \in \mathcal{V}_2$ *las aristas son del tipo* $\{x, y\}$ *ó* $\{y, x\}$.
- $\forall x_1, x_2 \in \mathcal{V}_1, \forall y_1, y_2 \in \mathcal{V}_2$ *no existe ninguna arista del tipo* $\{x_1, x_2\}$ *ni* $\{y_1, y_2\}$.

Si los dos subconjuntos $\mathcal{V}_1$ y $\mathcal{V}_2$ tienen la misma cantidad de elementos, esto es $|\mathcal{V}_1| = |\mathcal{V}_2|$, decimos que el grafo bipartito $\mathcal{G}$ es balanceado.

Del mismo modo, un grafo dirigido es bipartito si lo es su grafo no dirigido asociado.

■

Ejemplo 4.6 *En la Fig. 4.7 se presenta un grafo bipartito, donde los conjuntos de vértices (●) $a, b, c \in \mathcal{V}_1$ y (■) $x, y, z \in \mathcal{V}_2$, son disjuntos y no vacíos. Asimismo, vemos como las*

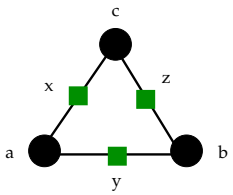


Figura 4.7: Grafo bipartito de ejemplo

relaciones existentes siempre van de un elemento de $\mathcal{V}_1$ a $\mathcal{V}_2$, ó de $\mathcal{V}_2$ a $\mathcal{V}_1$.

■

El concepto de *grafo* admite restricciones. Una de ellas consiste en admitir una única arista (resp. arco) con los mismos extremos, dando lugar a los denominados *grafos simples*; o por el contrario, con más de una arista (resp. arco) con los mismos extremos, referenciando así a los denominados *multigrafos*.

Definición 4.13 *Un* grafo simple *es un grafo $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ que no posee aristas (resp. arcos) cuyos extremos son el mismo vértice, y en el que no existen dos aristas (resp. arcos) que unan el mismo par de vértices.*

Si el grafo es dirigido, además deberá de cumplir que no existan dos arcos uniendo el mismo par de vértices con la misma dirección.

■

Definición 4.14 *Formalmente, un* multigrafo *es una terna $\mathcal{G} = (\mathcal{V}, \mathcal{A}, \psi)$, donde:*

- *$\mathcal{V}$ es un conjunto finito, tal que $\mathcal{V} \neq \emptyset$.*
- *$\mathcal{A}$ es un multiconjunto*[1] *de pares de vértices de la forma $\{x, y\}$ tal que $x, y \in \mathcal{V}$.*
- *ψ es una función, tal que $\psi : \mathcal{A} \to \{\{x, y\}/x, y \in \mathcal{V}, x \neq y\}$. La función ψ se llama función de* incidencia. *Para cada arista (resp. arco)* a $\in \mathcal{A}$, *$\psi(a)$ contiene los extremos de* a.

Se dice que las aristas (resp. arcos) $a_1, a_2 \in \mathcal{A}$ son aristas múltiples *(resp.* arcos múltiples*) si y sólo si $f(a_1) = f(a_2)$.*

[1]un multiconjunto difiere de un conjunto en que cada miembro del mismo tiene asociada una multiplicidad, un número natural indicando cuántas veces el elemento es miembro del conjunto, Por ejemplo, en el multiconjunto {a, a, a, b, b, c}, las multiplicidades de los miembros *«a»*, *«b»*, y *«c»* son 3, 2, y 1, respectivamente.

En el caso de tratarse de un multiconjunto de pares no ordenados de vértices, al multigrafo *se le denomina* no dirigido. *En caso contrario, se le llama* multigrafo dirigido.

■

Por lo tanto, un multigrafo es un grafo que tiene múltiples aristas (resp. arcos) sobre un mismo par de vértices, de este modo, dos de ellos pueden estar conectados por más de una arista (resp. arco).

Ejemplo 4.7 *Supongamos que tenemos el grafo no dirigido que se observa en la Fig. 4.8. En este caso se trata de un multigrafo ya que para los vértices* b,c $\in \mathcal{V}$*, existe una función* f *tal que* $f(e_1) = f(e_2) = f(e_3)$.

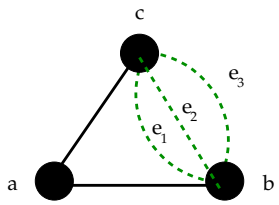


Figura 4.8: Multigrafo de ejemplo

■

4.1.4 | Morfismos de grafos

Mediante el concepto de *morfismo* pretendemos poner enfásis en la relación que existe entre las extremidades del grafo.

Definición 4.15 *Un* morfismo *de un grafo* $\mathcal{G} = (\mathcal{V}_\mathcal{G}, \mathcal{A}_\mathcal{G})$ *en un grafo* $\mathcal{H} = (\mathcal{V}_\mathcal{H}, \mathcal{A}_\mathcal{H})$ *es una aplicación* $f : \mathcal{V}_\mathcal{G} \to \mathcal{V}_\mathcal{H}$ *que conserva las aristas (resp. arcos), es decir, para toda arista (resp. arco)* {x,y} $\in \mathcal{A}_\mathcal{G}$, {f(x),f(y)} $\in \mathcal{A}_\mathcal{H}$.

■

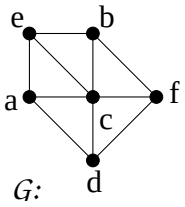
Para ilustrar este concepto vamos a utilizar el Ejemplo 4.8, que podría ser considerado como una instancia del *problema de la coloración de grafos*[2] [158].

Ejemplo 4.8 *Sea* $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ *un grafo no dirigido y* K *un conjunto de cardinalidad* $k \geqslant 2$*, de colores. Una coloración por* K *de* $\mathcal{G}$ *es dada por un morfismo* $f : \mathcal{V} \to K$*, de*

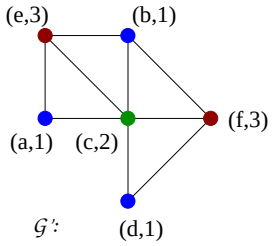
[2]este problema consiste en buscar la menor cantidad posible de colores para poder colorear los nodos de un grafo, de tal forma que los nodos adyacentes nunca tengan el mismo color. Este problema también se puede plantear para aristas o para las caras del plano de un grafo.

modo que si $\{v, w\} \in \mathcal{A}$, entonces $f(v) \neq f(w)$. En otras palabras, si cada elemento de K representa un color diferente, entonces una coloración para K consiste en dotar de un color a cada vértice de manera que vértices vecinos no tengan el mismo color.

Supongamos que las etiquetas de los colores están representados por enteros en $\{1, ..., k\}$ y que el grafo que queremos colorear es el que viene a continuación.



Consideremos que $K = \{1, 2, 3\}$ son colores. Concretamente, no existe ningún morfismo que lleve de $\mathcal{V}$ en K, ya que no es posible que con tres colores todos los vértices adyacentes posean colores diferentes. En cambio, esto sí se consigue si eliminamos una de las aristas, como por ejemplo, $\{a, d\}$. De este modo, aplicando un morfismo se podrían asignar los siguientes colores a los vértices de $\mathcal{G}$: $\{(a, 1), (e, 3), (c, 2), (b, 1), (d, 1), (f, 3)\}$, obteniendo el grafo $\mathcal{G}'$ que queda coloreado de la siguiente manera:



■

Siguiendo con la idea de que lo esencial de la parte visual del grafo no son las aristas sino sus extremidades, la posición de dichos vértices tampoco importa, y se puede variar para obtener un grafo más fácilmente comprensible. Estos cambios se denominan *isomorfismos* de grafos.

Definición 4.16 *Dos grafos $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ y $\mathcal{G}' = (\mathcal{V}', \mathcal{A}')$ son* isomorfos *si existe una biyección $f : \mathcal{V} \to \mathcal{V}'$ que preserva la relación de adyacencia, es decir, tal que:*

$$\{x, y\} \in \mathcal{A} \text{ si y sólo si } \{f(x), f(y)\} \in \mathcal{A}'$$

Denotamos que $\mathcal{G}$ y $\mathcal{G}'$ son isomorfos mediante $\mathcal{G} \approx \mathcal{G}'$.

■

Dos grafos isomorfos deben tener el mismo número de vértices. Más aún, todas las propiedades que se deriven de la relación de adyacencia deben ser idénticas en ambos. En particular deben tener el mismo número de aristas (resp. arcos), el mismo número de vértices aislados y la misma sucesión de valencias o grados. En este sentido, dos grafos isomorfos se consideran asimilables, como lo ilustra el Ejemplo 4.9.

Ejemplo 4.9 *Los dos grafos representados en la Fig. 4.9 son isomorfos, ya que la función* «f» *que lleva* «a» *en* «a'» , «b» *en* «b'» , «c» *en* «c'» *y* «d» *en* «d'» *es una biyección y preserva las aristas adyacentes.*

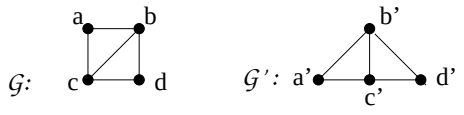


Figura 4.9: Grafos isomorfos de ejemplo

Dos grafos con idénticas sucesiones de valencias tienen el mismo número de vértices y de aristas, pero esto no es suficiente para que los grafos sean isomorfos, como muestra la Fig. 4.10.

Figura 4.10: Grafos no isomorfos de ejemplo

Ambos tienen sucesión de valencias 1, 1, 1, 2, 3, 3, pero no son isomorfos ya que en $\mathcal{G}'$ el único vértice de valencia 2 es adyacente a un vértice de valencia 1 y a otro de valencia 3, mientras que en el grafo $\mathcal{G}$ el único vértice de valencia 2 es adyacente a dos vértices de valencia 3.

■

4.2 | Grafos conceptuales

Los GC's son un formalismo de representación del conocimiento en el que los objetos del universo del discurso son modelados mediante conceptos y relaciones conceptuales, asociados unos con otros. Introducidos por Sowa [295], se basan en la *teoría de grafos* y la *lógica de primer orden* (LPO). Se trata básicamente de *grafos bipartitos* [190] sobre los que se distinguen dos conjuntos de vértices o nodos denominados *conceptos* y *relaciones*. Su principal ventaja radica en que permiten estructurar la mayor parte de la información expresada en forma de LN, permitiendo su estandarización. Ello significa que, a través de la aplicación de algoritmos, ésta pueda ser procesada para su interpretación.

Si nos centramos en en la Fig. 4.11, se puede observar un ejemplo de la notación empleada por Sowa. Se trata en definitiva de un tipo de representación esencialmente gráfico, donde los rectángulos representan a los nodos *concepto*, mientras que las elipses hacen lo propio con las *relaciones* entre conceptos. Un arco que apunta hacia una elipse muestra cual es el primer concepto ($Concepto_1$). Un arco que sale de una elipse muestra cual es el segundo ($Concepto_2$). Si una relación sólo tiene un argumento concepto, entonces la flecha sólo entra en la elipse, pero no sale. Si una relación tiene más de dos argumentos, la punta de la flecha se sustituye por números naturales.

Figura 4.11: Grafo conceptual de Sowa de ejemplo

Aunque la representación visual es extremadamente útil para un humano, ésta puede ser traducida en texto lineal, siendo los rectángulos abreviados por corchetes y las elipses por paréntesis

$$[Concepto_1] \rightarrow (relación) \rightarrow [Concepto_2]$$

y donde para recordar la dirección de las flechas, el grafo anterior puede leerse como

$$"La\ relación\ del\ Concepto_1\ es\ Concepto_2"$$

Sowa diferencia dos grupos de conceptos: un *tipo conceptual* y un *referente* [190]. El primero hace referencia a la clase de elemento que representa al concepto, o dicho de otro modo, se trata de la clase semántica a la que pertenece. Éstos se organizan en una jerarquía de tipos, es decir, un ordenamiento parcial definido sobre dicho conjunto y denotado por el símbolo $\leq$. El segundo indica la instancia específica y puede ser de dos clases: *genérico e individual*. Los referentes genéricos identifican a conceptos no especificados. Por ejemplo, un concepto [*Texture*] (`[Textura]`) indica que existe un concepto de ese tipo, pero no indica cual. En cambio los referentes individuales funcionan como sustitutos de elementos específicos del mundo real. A este respeto, para separar el tipo conceptual del referente individual usaremos «,». Es el caso del concepto [*Organe,tige*] (`[Órgano,tallo]`), cuyo tipo conceptual es *Organe* (`Órgano`) y su referente individual es «*tige*» («`tallo`»). En el caso de los referentes genéricos, éstos se pueden representar mediante «*», como por ejemplo en [*Texture,**] (`[Textura,*]`).

Finalmente, las relaciones conceptuales indican la manera en la que los conceptos se relacionan entre sí. Constan de un *tipo relacional*, que sugiere el papel semántico de los conceptos ligados a la relación, y una *valencia* igual al grado del tipo relacional, que señalará el número de conceptos unidos al nodo relación.

Ejemplo 4.10 *Utilizando la aproximación propuesta por Sowa [295], supongamos que tenemos definidos tres conceptos:* `[Evento,morder]`, `[Entidad,Eva]`, *y*

`[Lugar,oficina]`. *Usándolos, Sowa define el tipo relacional* «agente» (AGNT) *que liga una entidad con un evento, y el tipo relacional* «localidad» (LOC) *que lo hace con una entidad y un lugar, tal como se observa en la Fig. 4.12.*

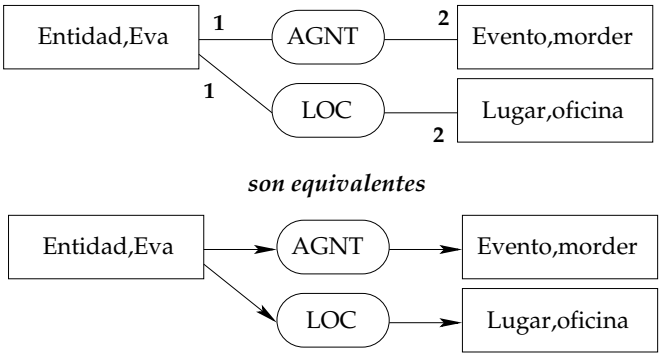


Figura 4.12: GC según Sowa de ejemplo

■

Una vez detallada a grandes rasgos la notación gráfica que se va a emplear, estamos en disposición de definir formalmente los GC's más simples. En este sentido, la mayoría del contenido que describimos a continuación está inspirado de Chein *et al.* [56] y Genest *et al.* [112].

4.2.1 | Grafos conceptuales básicos

Sobre el modelo relativamente sencillo de GC de Sowa [295] se ha ido añadiendo, a lo largo de estos años, nociones cada vez más complejas. Por este motivo, a día de hoy, existe una gran variedad de tipos de GC's [56]. Sin embargo, todos ellos mantienen algo en común: existe una separación y una estructuración del conocimiento en dos niveles.

Por un lado, es necesario disponer de un mapa del conocimiento básico del dominio[3] y de sus restricciones. Es lo que comúnmente se denomina *soporte* [56].

Definición 4.17 *Un* soporte *es una tripla* $\mathcal{S} = (\mathcal{T_C}, \mathcal{T_R}, \mathcal{I})$ *de conjuntos disjuntos, donde:*

- $\mathcal{T_C}$ *y* $\mathcal{T_R}$ *son conjuntos finitos parcialmente ordenados*[4] *de* tipos conceptuales *y* tipos relacionales, *respectivamente, donde el orden que los rige es interpretado como una relación de especialización. Entonces,* $t \leq r$ *se lee como que* r *es una generalización de* t*, o que* t *es una especialización de* r*.*

[3] de hecho, Sowa indicaba que un GC no tenía ningún sentido de forma aislada, sino que sólo a través de las diversas redes que enlazaban los conceptos y relaciones de podía establecer un contexto.

[4] en este caso el conjunto parcialmente ordenado no es más que una jerarquía de tipos.

- *Los tipos de $\mathcal{T}_\mathcal{C}$ poseen un* tipo universal *que generaliza a todos los demás, denotado por $\top$. Del mismo modo, los tipos de $\mathcal{T}_\mathcal{R}$ pueden tener cualquier aridad*[5] *superior o igual a 1, y sólo aquéllos con misma aridad serán comparables.*

- *$\mathcal{I}$ es un conjunto numerable de* referentes individuales*, con un* referente genérico *denotado por $* \notin \mathcal{I}$. El conjunto $\mathcal{I} \cup \{*\}$ está ordenado parcialmente y sus elementos son dos a dos no comparables entre sí, siendo $*$ el más general.*

■

En definitiva, un soporte consiste en una jerarquía de tipos conceptuales, una jerarquía de tipos relacionales y un conjunto de referentes individuales que pueden ser identificados mediante un diccionario, cuyos elementos se asociarán más tarde con tipos conceptuales. En la práctica, este diccionario representa formas léxicas de un tesauro o de un *corpus*, mientras que los tipos conceptuales se referirán a sus clases semánticas, y los relacionales al nexo que los une.

Una vez introducidos los conceptos y las relaciones que formarán parte del mapa general del dominio, podemos enlazarlos entre sí con el fin de describir hechos en los que estamos interesados. Para ello, usaremos la noción de *grafo conceptual básico* [56] (GCB) sobre un soporte $\mathcal{S}$. Se trata de una simple variante de la noción original del GC de Sowa [295] en el que se representa la información factual interpretable en el contexto de $\mathcal{S}$ sin negación, y que describe tanto los conceptos como sus relaciones. En este sentido, un GCB representa una plantilla que va a ser cumplimentada con las instancias específicas del ámbito de trabajo para un contexto determinado, o lo que es lo mismo, con los referentes individuales de los conceptos y las relaciones, todos ellos tomados a partir del soporte.

Definición 4.18 *Formalmente, un* GCB *definido sobre un soporte $\mathcal{S} = (\mathcal{T}_\mathcal{C}, \mathcal{T}_\mathcal{R}, \mathcal{I})$, es una cuádrupla $\mathcal{G} = (\mathcal{C} \cup \mathcal{R}, \mathcal{A}, \mathcal{E})$ que satisface las siguientes condiciones:*

- *$(\mathcal{C} \cup \mathcal{R}, \mathcal{A})$ es un multigrafo bipartito, no necesariamente conexo, donde $\mathcal{C}$ y $\mathcal{R}$ son conjuntos disjuntos de* nodos conceptos *y* nodos relaciones*, respectivamente.*

- *$\mathcal{A}$ es el multiconjunto de* aristas.

- *$\mathcal{E}$ es una función de etiquetado de nodos y relaciones del grafo $\mathcal{G}$ que verifica:*

  - *Un nodo concepto $c \in \mathcal{C}$ se etiqueta con un par* $[\text{tipo}(c), \text{ref}(c)] \in \mathcal{T}_\mathcal{C} \times (\mathcal{I} \cup \{*\})$.
  - *Un nodo relación $r \in \mathcal{R}$ se etiqueta mediante* $\text{tipo}(r) \in \mathcal{T}_\mathcal{R}$*, y su valencia debe ser igual a la aridad de* $\text{tipo}(r)$.
  - *Una arista $a \in \mathcal{A}$, etiquetada mediante $i \in \mathbb{N}$, que conecta un nodo $r \in \mathcal{R}$ con un nodo $c \in \mathcal{C}$, se denota por (r, i, c). Las aristas $(r, 1, c_1), ..., (r, k, c_k)$*

[5] la aridad de un operador matemático o de una función es el número de argumentos necesarios para que dicho operador o función se pueda calcular.

que inciden sobre r *son totalmente ordenados y se etiquetan de* 1 *a la aridad de* tipo(r). *Generalmente, se emplea la notación* $r = \text{tipo}(r)(c_1, ..., c_k)$.

■

Intuitivamente, un GCB se puede ver como un grafo bipartito que proporciona un conjunto de punteros semánticos sobre dos jerarquías del dominio de conocimiento, uno para conceptos y otro para las relaciones entre estos conceptos. Como existe un orden parcial sobre los tipos conceptuales $\mathcal{T}_\mathcal{C}$, también existirá un orden parcial sobre las etiquetas de los nodos conceptos. Es decir, dadas dos etiquetas $\mathcal{E}(c_1) = [\textit{tipo}(c_1), \textit{ref}(c_1)]$ y $\mathcal{E}(c_2) = [\textit{tipo}(c_2), \textit{ref}(c_2)]$ sobre dos conceptos $c_1, c_2 \in \mathcal{C}$, se dice que $\mathcal{E}(c_1) \leq \mathcal{E}(c_2)$ si y sólo si $\textit{tipo}(c_1) \leq \textit{tipo}(c_2)$ y $\textit{ref}(c_1) \leq \textit{ref}(c_2)$.

Ejemplo 4.11 *Un ejemplo concreto de un* GCB *es el que se muestra en la Fig. 4.13. Se observa como la etiqueta de un nodo concepto* $c \in \mathcal{C}$ *es un par* (*tipo*(c), *ref*(c)). *Así, por ejemplo, existen dos nodos conceptos cuyo tipo es* «Organe» (`«Órgano»`) *y cuyos referentes individuales son* {tige, tépale}. *Lo mismo ocurre si consideramos como nodos concepto aquéllos que son de tipo* «Forme» (`«Forma»`), *cuyos referentes son* {oblong, tétragone, dense, obovale}.

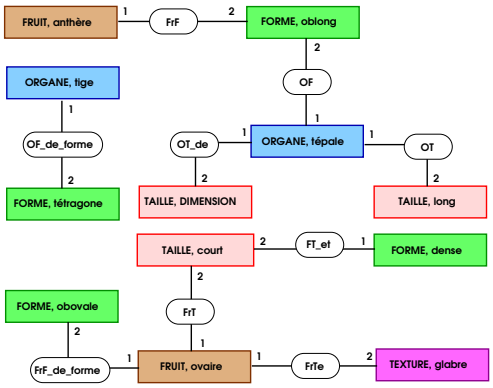


Figura 4.13: GCB de ejemplo

A su vez, existe un tipo relacional «OF» *que liga un nodo concepto de tipo* «Organe» (`«Órgano»`) *con uno de tipo* «Forme» (`«Forma»`). *Del mismo modo, el tipo relacional* «FrTe» *liga un nodo concepto de tipo* «Fruit» (`«Fruto»`) *con* «Texture» (`«Textura»`).

■

En este punto, y una vez formalizada la estructura de GCB, podemos ya introducir un aspecto fundamental como es la comparación de GCB's. De hecho, esta noción viene dada en la definición de relación de *especialización/generalización* [21, 56] ($\leq$ / $\geq$) sobre el conjunto de GCB's, ya que el orden parcial establecido es uno de los mecanismos para su comparación.

Así, cuando hablamos de la *especialización* de un GCB entendemos la restricción de determinados aspectos, con el fin de conseguir una estructura más específica. De la misma manera, cuando hablamos de su *generalización*, tratamos de que el GCB resultante aporte informaciones más generales. A partir de estos conceptos, introduciremos luego el de *proyección*, un tipo especial de morfismo que nos permitirá especializar conceptos y relaciones sobre los grafos [56].

Con el fin de simplificar la notación, en adelante representaremos los nodos relación mediante el símbolo ♦.

4.2.2 | Especialización

Para establecer una relación de *especialización* ($\leq$), es necesario introducir operaciones internas sobre el conjunto de los GCB's definidos sobre el soporte $\mathcal{S}$. Concretamente existen diversas *operaciones elementales*: cuatro unarias[6] y una binaria[7].

4.2.2.1 | Operaciones unarias

Sea $\mathcal{G}$ un GCB de partida. Se puede obtener otro GCB $\mathcal{H}$ más específico o igual a partir de $\mathcal{G}$, es decir, $\mathcal{H} \leq \mathcal{G}$, aplicando:

- *Restricción de concepto*. Sea $c \in \mathcal{C}$ un nodo concepto, donde $\mathcal{E}(c) = [\mathit{tipo}(c), \mathit{ref}(c)]$. $\mathcal{H}$ se obtiene en este caso cuando sustituimos $\mathcal{E}(c)$ por $[\mathit{tipo}'(c), \mathit{ref}'(c)]$, donde $\mathit{tipo}'(c) \leq \mathit{tipo}(c)$ y $\mathit{ref}'(c) \leq \mathit{ref}(c)$. Hay que decir que si $\mathit{ref}'(c) < \mathit{ref}(c)$, entonces $\mathit{ref}(c)$ es un referente genérico y $\mathit{ref}'(c)$ un referente individual.

Ejemplo 4.12 *Supongamos $\mathcal{G}$, el* GCB *mostrado en la Fig. 4.14.*

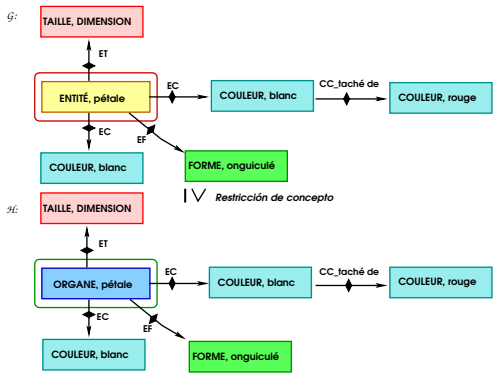


Figura 4.14: Restricción de concepto

[6] en el sentido en el que se realizan a partir de un único GCB.
[7] en el sentido en el que se realizan a partir de dos GCB's.

El concepto [Entité, pétale] (`[Entidad, pétalo]`) *de* $\mathcal{G}$ *se puede restringir a* [Organe: pétale] (`[Órgano,pétalo]`), *considerando que en la jerarquía de conceptos* Organe$\leq$Entité (`Órgano`$\leq$`Entidad`), *por lo que aplicando esta operación se obtiene el* GCB $\mathcal{H}$ *más específico que* $\mathcal{G}$.

■

- *Restricción de relación*. Sea $r \in \mathcal{R}$ un nodo relación. En este caso, el GCB $\mathcal{H}$ se obtiene sustituyendo el tipo relacional *tipo*(r) por *tipo*$'(r)$, donde *tipo*$'(r) \leq$ *tipo*(r).

Ejemplo 4.13 *Partamos del* GCB *que se obtuvo en la restricción de concepto del Ejemplo 4.12. Se observa como la relación entre* [Organe,pétale] (`[Órgano, pétalo]`) *y* [Forme,onguiculé] (`[Forma,ungulado]`) *está etiquetada por un nodo relación* EF. *Si consideramos que* OF $\leq$ EF, *se podría restringir* EF *al tipo más general* OF.

También se puede realizar el mismo proceso con la relación entre los nodos [Organe, pétale] (`[Órgano, pétalo]`) *y* [Taille, DIMENSION] (`[Tamaño, DIMENSIÓN]`), *etiquetada por el nodo relación* ET *y, con las relaciones entre* [Organe, pétale] (`[Órgano, pétalo]`) *y los dos nodos conceptos* [Couleur, blanc] (`[Color, blanco]`) *etiquetadas por nodos relación* EC. *Si consideramos que los tipos relacionales* OT $\leq$ ET *y* OC $\leq$ EC, *obtenemos que, aplicando estas restricciones, dan lugar a los tipos más generales* OT *y* OC, *respectivamente. De este modo,* $\mathcal{H} \leq \mathcal{G}$, *es decir,* $\mathcal{H}$ *es más específico que el de partida* $\mathcal{G}$, *como vemos en la Fig. 4.15.*

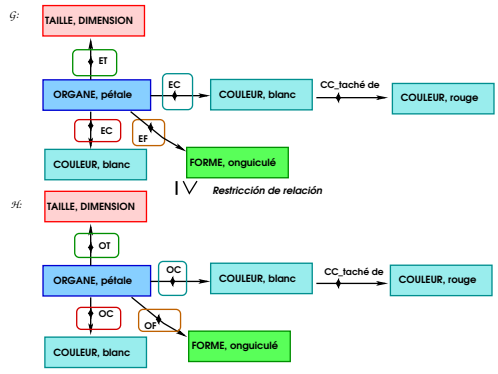


Figura 4.15: Restricción de relación

■

- *Ligadura interna*. Sean $c_1, c_2 \in \mathcal{C}$ dos nodos conceptos, con la misma etiqueta. En este caso, el GCB $\mathcal{H}$ se obtiene de fusionar c_1 y c_2.

Ejemplo 4.14 *Partamos del* GCB *que se obtuvo en el Ejemplo anterior 4.13. En la Fig. 4.16 se puede observar como el concepto* [Couleur,blanc]

`([Color,blanco])` *aparece dos veces, por lo que fusionando el resultado es el* GCB $\mathcal{H}$.

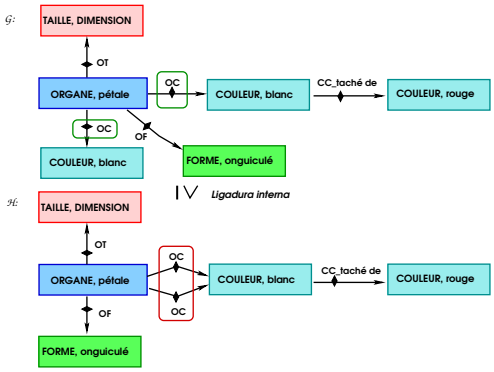


Figura 4.16: Ligadura interna

Observar que, en este caso se obtienen como resultado dos relaciones de tipo OC *que van desde* [Organe,pétale] `([Órgano,pétalo])` *hasta* [Couleur,blanc] `([Color,blanco])`. *Más tarde, usando otra operación denominada* simplificación, *podemos eliminar una de las dos relaciones.*

■

- *Simplificación*. Sean $r_1, r_2 \in \mathcal{R}$ dos relaciones del mismo tipo, con los mismos nodos conceptos vecinos, y en el mismo orden. En este caso, el GCB $\mathcal{H}$ se obtiene suprimiendo o bien r_1, o bien r_2.

  Ejemplo 4.15 *Partamos del* GCB *obtenido en el Ejemplo 4.14. Se observa como existen dos relaciones del mismo tipo entre* [Organe,pétale] `([Órgano,pétalo])` *y* [Couleur,blanc] `([Color, blanco])` *ambas con tipo relacional* OC. *Fusionándolas, obtenemos el* GCB $\mathcal{H}$ *de la Fig. 4.17.*

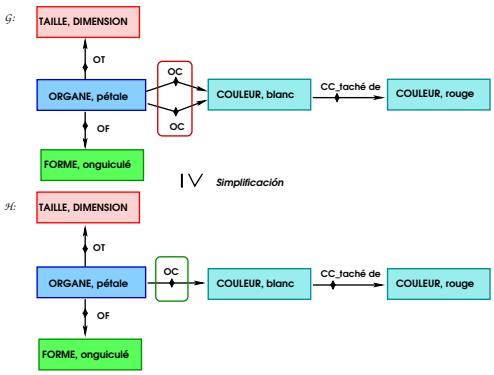


Figura 4.17: Simplificación

■

4.2.2.2 | Operaciones binarias

Sean $\mathcal{G}_1$ y $\mathcal{G}_2$ dos GCB's distintos de partida. Se puede obtener $\mathcal{H}$, un nuevo GCB más específicos a partir de $\mathcal{G}_1$ y $\mathcal{G}_2$, aplicando:

- *Ligadura externa*. Sean $c_1 \in \mathcal{C}_{\mathcal{G}_1}$ y $c_2 \in \mathcal{C}_{\mathcal{G}_2}$ dos nodos concepto con la misma etiqueta. En este caso, el GCB $\mathcal{H}$ se obtiene fusionando los nodos concepto c_1 y c_2.

  Ejemplo 4.16 *Supongamos dos* GCB*'s $\mathcal{G}_1$ y $\mathcal{G}_2$, mostrados en la Fig. 4.18. Se ve como el nodo concepto* [Organe,pétale] (`[Órgano,pétalo]`) *se repite tanto en $\mathcal{G}_1$ como en $\mathcal{G}_2$. Por lo tanto, tras la fusión de ambos conceptos se obtiene el* GCB *más específico $\mathcal{H}$.*

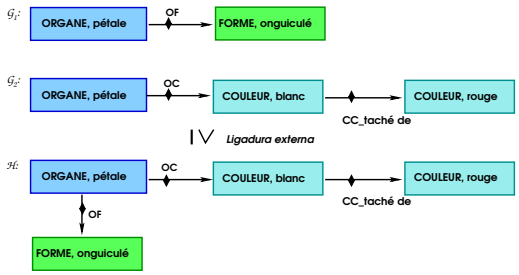


Figura 4.18: Ligadura externa

■

Una vez definidas las operaciones elementales necesarias para la *relación de especialización*, estamos en disposición de dar la siguiente definición.

Definición 4.19 *Sean $\mathcal{G}$ y $\mathcal{H}$ dos* GCB*'s. Se dice que $\mathcal{H}$ es una* especialización *de $\mathcal{G}$, denotado por $\mathcal{H} \leq \mathcal{G}$, si y sólo si existe una secuencia de operaciones elementales de especialización que permiten transformar $\mathcal{G}$ en $\mathcal{H}$.*

■

4.2.3 | Generalización

Para establecer una relación de *generalización* ($\geq$), es necesario introducir operaciones internas sobre el conjunto de los GCB's definidos sobre el soporte $\mathcal{S}$. En este sentido, se puede considerar a las *operaciones elementales de generalización* como las recíprocas de las ya definidas de *especialización*. Concretamente, existen cinco operaciones que permiten obtener un grafo más general a partir de uno más específico. Así, sea $\mathcal{G}$ un GCB de partida. Se puede obtener el GCB más general $\mathcal{H}$ a partir de $\mathcal{G}$, aplicando:

- *Generalización de concepto*. Sea $c \in \mathcal{C}$ un nodo concepto, donde $\mathcal{E}(c) = [tipo(c), ref(c)]$. En este caso, el GCB $\mathcal{H}$ se obtiene sustituyendo $\mathcal{E}(c)$ por $\mathcal{E}'(c) = [tipo'(c), ref'(c)]$, donde $tipo'(c) \geq tipo(c)$ y $ref'(c)$ es un referente genérico, es decir, $ref'(c) = *$.

  Ejemplo 4.17 *Supongamos un* GCB $\mathcal{G}$ *tal como el que se muestra en la Fig. 4.19. El concepto* [Organe,pétale] `([Órgano,pétalo])` *se puede generalizar a todos los posibles, es decir a* [Organe,*] `([Órgano,*])`. *Además, se siguen cumpliendo las restricciones* Organe $\geq$ Organe, *por lo que aplicando esta operación se obtiene el* GCB *más general* $\mathcal{H}$.

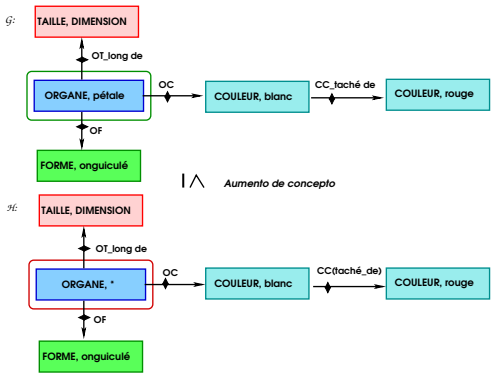


Figura 4.19: Generalización de concepto

■

- *Generalización de relación*. Sea $r \in \mathcal{R}$ un nodo relación. En este caso el GCB $\mathcal{H}$ se obtiene sustituyendo el tipo relacional de r, es decir, $tipo(r)$, por uno más general $tipo'(r)$. Dicho de otro modo, $tipo'(r) \geq tipo(r)$. Además, no existen restricciones sobre esta operación.

  Ejemplo 4.18 *Partamos del* GCB *que se obtuvo en el Ejemplo 4.17. Se observa como la relación entre* [Organe,*] `([Órgano,*])` *y* [Taille,DIMENSION] `([Tamaño,DIMENSION])` *está etiquetada con un nodo relación* OT_long de `(OT_largo de)`. *Si consideramos que* OT $\geq$ OT_long de, *y aplicamos la generalización de la relación, este tipo se convierte en* OT, *tal y como se observa en la Fig. 4.20.*

  Lo mismo ocurre si realizamos el mismo proceso con las relaciones entre [Organe,*] `([Órgano,*])` *y* [Couleur,blanc] `([Color,blanco])` *etiquetada por un nodo relación* OC, *y entre* [Organe,*] `([Órgano,*])` *y* [Forme,onguiculé] `([Forma,ungulado])` *etiquetada por un nodo relación* OF. *Si consideramos que* EC $\geq$ OC *y* EF $\geq$ OF, *el resultado aplicando la generalización de relación será en el primer caso* EC, *y en el segundo* EF. *El* GCB *resultante es* $\mathcal{H}$.

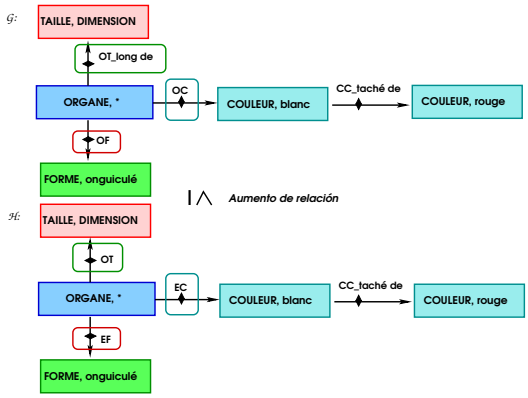


Figura 4.20: Generalización de relación

■

- *Duplicación*. Sea $r \in \mathcal{R}$ un nodo relación. En este caso, se obtiene el GCB $\mathcal{H}$ al añadir un nodo relación gemelo de r.

  Ejemplo 4.19 *Partamos del* GCB *obtenido en el Ejemplo 4.18. Se observa como existe una relación del tipo* EC *entre* [Organe,*] (`[Órgano,*]`) *y* [Couleur,blanc] (`[Color,blanco]`). *Duplicándola, obtenemos el* GCB $\mathcal{H}$ *de la Fig. 4.21.*

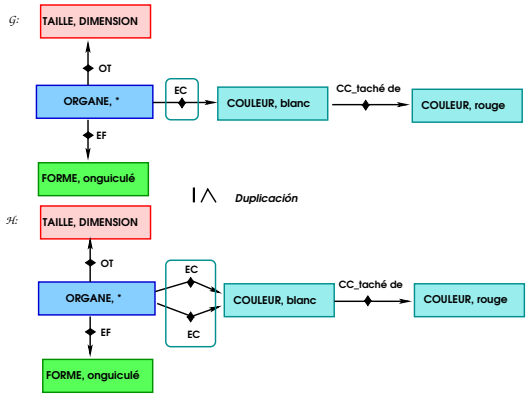


Figura 4.21: Duplicación

■

- *Desdoblamiento*. Sean $c \in \mathcal{C}$ un nodo concepto y $r_1, r_2 \in \mathcal{R}$ dos relaciones que inciden sobre c. En este caso, el GCB $\mathcal{H}$ se obtiene separando c en dos nodos concepto c_1 y c_2 con misma etiqueta que c, y relacionando c_1 con la relación r_1, y c_2 con la relación r_2. En este sentido, r_1 ó r_2 pueden ser relaciones vacías.

  Ejemplo 4.20 *Partamos del* GCB *del Ejemplo 4.19. Se observa como existen dos relaciones que unen los conceptos* [Organe:*] (`[Órgano,*]`) *y* [Couleur,blanc]

`([Color,blanco])` *etiquetada por* EC, *por lo que desdoblando el nodo* [Organe,*] `([Órgano,*])` *se obtiene el* GCB $\mathcal{H}$ *de la Fig. 4.22.*

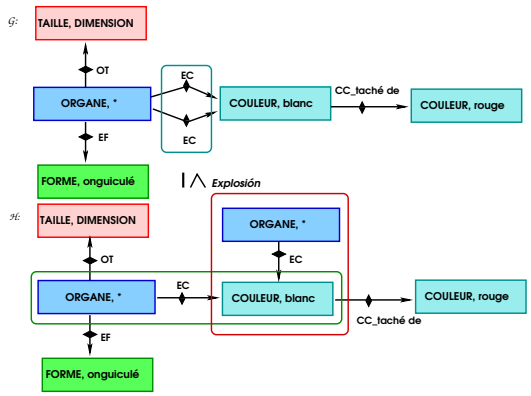


Figura 4.22: Desdoblamiento

■

- *Descomposición*. Sea $c \in \mathcal{C}$ un nodo concepto. En este caso, los GCB's $\mathcal{H}_1$ y $\mathcal{H}_2$ se obtienen suprimiendo ciertas componentes conexas de $\mathcal{G}$, es decir, creando dos nodos concepto c_1 y c_2 con misma etiqueta que c, donde c_1 estará en $\mathcal{H}_1$ y c_2 en $\mathcal{H}_2$. Algunas de las relaciones que se establecían con c se harán ahora con c_1, y las demás con c_2.

Ejemplo 4.21 *Partamos del* GCB $\mathcal{G}$ *del Ejemplo 4.19. Vamos a descomponer en dos nodos el nodo concepto* [Organe,*] `([Órgano,*])`. *Se ve como en el* GCB $\mathcal{H}_1$ *se han repartido las relaciones entre* [Organe,*] `([Órgano,*])` *y* [Taille,DIMENSION] `([Tamaño,DIMENSION])` *y entre* [Organe,*] `([Órgano,*])` *y* [Forme,onguiculé] `([Forma,ungulado])`. *En cambio en el* GCB $\mathcal{H}_2$ *se repartieron las relaciones entre* [Organe,*] `([Órgano,*])` *y* [Couleur,blanc] `([Color,blanco])`. *El resultado de* $\mathcal{H}_1$ *y* $\mathcal{H}_2$ *se muestra en la Fig. 4.23.*

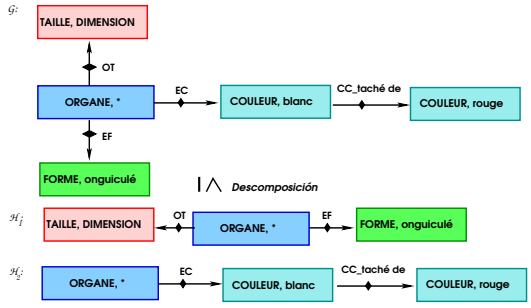


Figura 4.23: Descomposición

■

Una vez definidas las operaciones elementales necesarias para la *relación de generalización*, estamos en disposición de dar la siguiente definición.

Definición 4.20 *Sean $\mathcal{G}$ y $\mathcal{H}$ dos* GCB*'s. Se dice que $\mathcal{H}$ es una* generalización *de $\mathcal{G}$, denotado por $\mathcal{H} \geq \mathcal{G}$, si y sólo si existe una secuencia de operaciones elementales de generalización que permiten transformar $\mathcal{G}$ en $\mathcal{H}$.* ■

La relación de especialización es un preorden parcial[8] [55]. Pero al mismo tiempo, existe una relación de reciprocidad entre las operaciones elementales de la generalización y las de la especialización [111].

Teorema 4.1 *Se dice que el* GCB *$\mathcal{H}$ es una especialización del* GCB *$\mathcal{G}$, es decir, $\mathcal{H} \leq \mathcal{G}$, si y sólo si $\mathcal{G}$ es una generalización de $\mathcal{H}$, es decir, $\mathcal{G} \geq \mathcal{H}$.*

Demostración: Ver en [56]. ■

4.2.4 | Proyección

En este punto, podemos ya introducir la *proyección*, un morfismo que permite especializar conceptos y relaciones sobre los GCB's.

Definición 4.21 *Sean $\mathcal{G}_1 = (\mathcal{C}_1, \mathcal{R}_1, \mathcal{A}_1, \mathcal{E}_1)$ y $\mathcal{G}_2 = (\mathcal{C}_2, \mathcal{R}_2, \mathcal{A}_2, \mathcal{E}_2)$ dos* GCB*'s definidos sobre un soporte $\mathcal{S} = (\mathcal{T}_\mathcal{C}, \mathcal{T}_\mathcal{R}, \mathcal{I})$. Una* proyección *de $\mathcal{G}_1$ en $\mathcal{G}_2$ es una correspondencia π de $\mathcal{C}_1$ en $\mathcal{C}_2$, y de $\mathcal{R}_1$ en $\mathcal{R}_2$ que verifica:*

$$(r, i, c) \in \mathcal{A}_1 \Rightarrow (\pi(r), i, \pi(c)) \in \mathcal{A}_2$$

y

$$x \in \mathcal{C}_1 \cup \mathcal{R}_1 \Rightarrow \mathcal{E}_2(\pi(x)) \leq \mathcal{E}_1(x)$$

donde, si $x \in \mathcal{C}_1$, $\leq$ hace referencia al producto cartesiano del orden en $\mathcal{T}_\mathcal{C}$ y $\mathcal{I} \cup \{*\}$[9]*. En el caso de que $x \in \mathcal{R}_1$, entonces $\leq$ hace referencia al orden de $\mathcal{T}_\mathcal{R}$.*

Del mismo modo, se dice que $\mathcal{G}_1$ es el origen y que $\mathcal{G}_2$ es el destino, pero también se dice que $\mathcal{G}_1$ subsume *a $\mathcal{G}_2$ o que $\mathcal{G}_1$ es* más general *que $\mathcal{G}_2$, usando la notación $\mathcal{G}_1 \succeq \mathcal{G}_2$. El conjunto de proyecciones de $\mathcal{G}_1$ en $\mathcal{G}_2$ se denota por* proy*$(\mathcal{G}_1, \mathcal{G}_2)$.*

■

[8] cuando un conjunto contiene los elementos de otro conjunto, se dice que es menor o igual. Con todo, hay conjuntos que no son comparables, puesto que cada uno puede contener algún elemento que no esté presente en el otro. Por lo tanto, la inclusión de subconjuntos usando una relación de preorden, es decir, una relación que es reflexiva y transitiva, pero no necesariamente antisimétrica, se llama *preorden parcial*.

[9] esto es, $[\textit{tipo}(\pi(x)), \textit{ref}(\pi(x))] \leq [\textit{tipo}(x), \textit{ref}(x)]$ si y sólo si $\textit{tipo}(\pi(x)) \leq \textit{tipo}(x)$ y $\textit{ref}(\pi(x)) \leq \textit{ref}(x)$.

Intuitivamente, una *proyección* también puede definirse usando el concepto de *homomorfismo*[10] aplicado a GCB's, que permite especializar las etiquetas de los nodos conceptos y de los nodos relacionales. La búsqueda de una proyección de un grafo $\mathcal{G}$ en un grafo $\mathcal{H}$ puede ser visto como la búsqueda del embebimiento de la información representada por $\mathcal{G}$ en $\mathcal{H}$, es decir, permite calcular si un grafo está más especializado que otro. Si ese es el caso, se dice que $\mathcal{H}$ *es una especialización de* $\mathcal{G}$ ó que $\mathcal{G}$ *es una generalización de* $\mathcal{H}$.

Esta definición conlleva el hecho de que la proyección [112] es un morfismo de los grafos que conserva la bipartición. Esto es, la imagen de un vértice concepto es un vértice concepto y la imagen de un vértice relación es otro vértice relación. La condición sobre las etiquetas hace intervenir los conocimientos representados en el soporte. De este modo, la etiqueta de la imagen de un vértice es una especialización de la etiqueta de su origen.

Ejemplo 4.22 *Consideremos los dos* GCB*'s* $\mathcal{G}$ *y* $\mathcal{H}$ *de la Fig. 4.24. Existe una proyección de* $\mathcal{G}$ *en* $\mathcal{H}$ *denominada* h_1*, mostrada en la figura mediante flechas discontinuas. En este contexto, se observa como* [Fruit,anthère] `([Fruto,antera])` *en* $\mathcal{G}$ *se proyecta en* [Fruit,anthère] `([Fruto,antera])` *de* $\mathcal{H}$*, mediante* h_1*. En el otro nodo concepto ocurre exactamente lo mismo. El nodo concepto* [Forme,oblong] `([Forma,oblongo])` *de* $\mathcal{G}$ *se proyecta en* [Forme,oblong] `([Forma,oblongo])` *de* $\mathcal{H}$*. Pero además, la relación que une a ambos nodos conceptos, es decir, la relación* FrF *de* $\mathcal{G}$ *se proyecta en la misma de* $\mathcal{H}$*.*

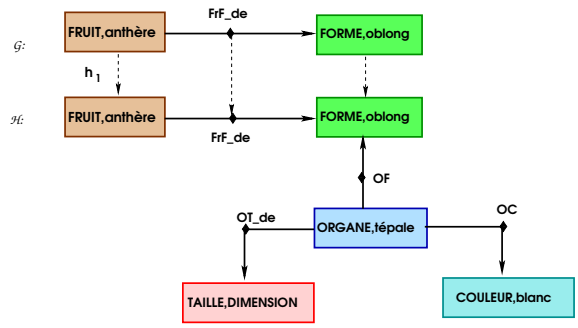


Figura 4.24: Homomorfismo o proyección de $\mathcal{G}$ en $\mathcal{H}$

■

Ejemplo 4.23 *Consideremos los dos* GCB*'s* $\mathcal{G}$ *y* $\mathcal{H}$ *de la Fig. 4.25. Existe una proyección de* $\mathcal{G}$ *en* $\mathcal{H}$*, llamada* h_1*, mostrada en la figura mediante flechas discontinuas. En este contexto se observa como* [Fruit,*] `([Fruto,*])` *en* $\mathcal{G}$ *se proyecta en* [Fruit,anthère] `([Fruto,antera])` *de* $\mathcal{H}$*, mediante* h_1*. En este sentido, el referente genérico es más general que el referente individual* «anthère» `(«antera»)`*, es decir,* * $\geq$ antera*. Al igual que en el Ejemplo 4.22, el nodo concepto* [Forme,oblong] `([Forma,oblongo])` *de* $\mathcal{G}$ *se*

[10] es un morfismo que preserva las aristas.

proyecta en [Forme,oblong] (`[Forma,oblongo)`] *de* $\mathcal{H}$*, así como la relación que los une a ambos.*

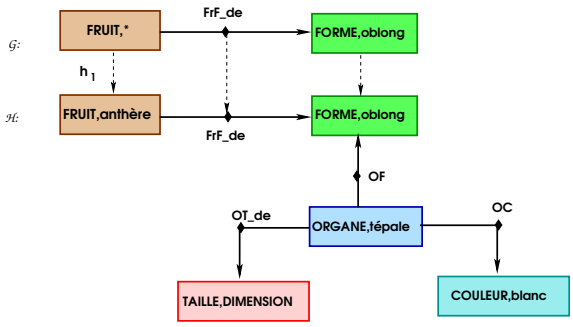


Figura 4.25: Homomorfismo o proyección de $\mathcal{G}$ en $\mathcal{H}$ usando un referente genérico

■

Ejemplo 4.24 *Consideremos los dos* GCB*'s* $\mathcal{G}$ *y* $\mathcal{H}$ *de la Fig. 4.26. Existe dos proyecciones de* $\mathcal{G}$ *en* $\mathcal{H}$*, llamadas* h_1 *y* h_2*, mostradas en la figura mediante flechas discontinuas en el primer caso, y discontinuas y punteadas en el segundo.*

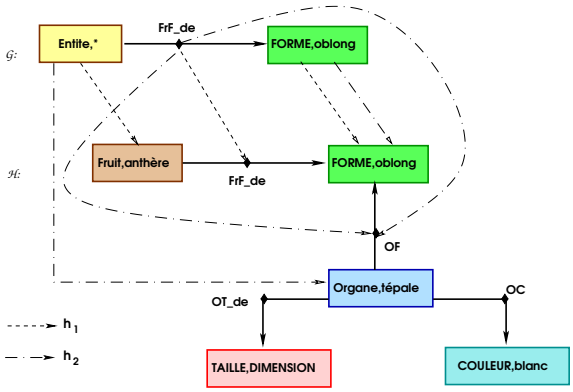


Figura 4.26: Homomorfismos o proyecciones de $\mathcal{G}$ en $\mathcal{H}$, donde $\mathcal{G} \succeq \mathcal{H}$

En este contexto, si consideramos que Fruit $\leq$ Entité (`Fruto` $\leq$ `Entidad`) *se observa como* [Entité,*] (`[Entidad,*]`) *en* $\mathcal{G}$ *se proyecta en* [Fruit,anthère] (`[Fruto,antera]`) *de* $\mathcal{H}$*, mediante* h_1*. A su vez, si consideramos que* Organe $\leq$ Entité (`Órgano` $\leq$ `Entidad`)*, se puede proyectar* [Entité,*] (`[Entidad,*]`) *de* $\mathcal{G}$ *en* [Organe,tépale] (`[Órgano,tépalo]`) *de* $\mathcal{H}$*, mediante* h_2*. Retomando los Ejemplos 4.22 y 4.23, el nodo concepto* [Forme,oblong] (`[Forma,oblongo]`) *de* $\mathcal{G}$ *se proyecta en* [Forme,oblong] (`[Forma,oblongo]`) *de* $\mathcal{H}$*, así como la relación que une a ambos nodos conceptos, tanto con* h_1 *como con* h_2*.*

■

Así, se pueden aplicar proyecciones sobre GCB's en los que tanto las etiquetas de los nodos conceptos como los nodos relaciones están totalmente especificadas, denominadas

proyecciones totales, como se observa en el Ejemplo 4.22. Pero también, se pueden aplicar proyecciones sobre GCB's llamadas *proyecciones parciales*. Esto es, en el caso en el que alguno de los nodos concepto o relación posean en su etiqueta un tipo más general, o incluso un referente genérico, tal como se puede ver en los Ejemplos 4.23 y 4.24.

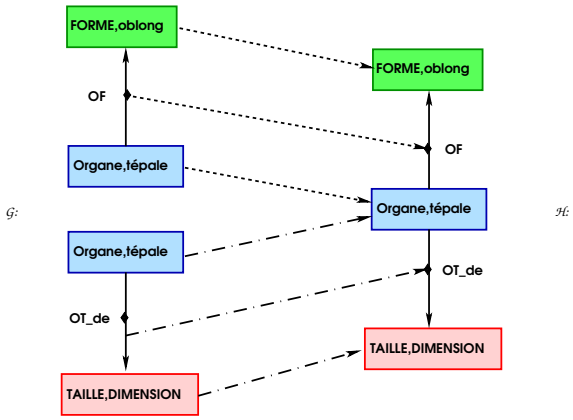


Figura 4.27: $\mathcal{G} \succeq \mathcal{H}$ y $\mathcal{H} \not\succeq \mathcal{G}$

Observemos ahora que ocurre con la proyección cuando un GCB posee dos nodos concepto con el mismo referente individual, como ocurre en el GCB $\mathcal{G}$ de la Fig. 4.27, donde existen dos nodos [*Organe,tépale*] (`[Órgano,tépalo]`). Si consideramos también el GCB $\mathcal{H}$, vemos como existe claramente una proyección de $\mathcal{G}$ en $\mathcal{H}$. Por un lado, podemos proyectar en $\mathcal{H}$ la relación existente entre [*Organe,tépale*] (`[Órgano,tépalo]`) y [*Forme,oblong*] (`[Forma,oblongo]`), pero también entre [*Organe,tépale*] (`[Órgano,tépalo]`) y [*Taille,DIMENSION*] (`[Tamaño,DIMENSION]`). En cambio, no existe ninguna proyección en el otro sentido, es decir de $\mathcal{H}$ en $\mathcal{G}$, a pesar de que ambos poseen intuitivamente el mismo significado: «*el referente individual* «tépale» (`«tépalo»`) *cuyo tipo conceptual es* «Entité» (`«Entidad»`) *tiene como propiedades a* [Forme,oblong] (`[Forma,oblongo]`) *y* [Taille,DIMENSION] (`[Tamaño,DIMENSION]`)». Esto se solventaría si ambos nodos conceptuales [*Organe,tépale*] (`[Órgano,tépalo]`) fuesen considerados el mismo elemento, por lo que se unirían para formar uno único.

Teorema 4.2 *Sea* $\mathcal{G}_1 = (\mathcal{C}_1, \mathcal{R}_1, \mathcal{A}_1, \mathcal{E}_1)$ *y* $\mathcal{G}_2 = (\mathcal{C}_2, \mathcal{R}_2, \mathcal{A}_2, \mathcal{E}_2)$ *dos* GCB*'s definidos sobre un* $\mathcal{S} = (\mathcal{T}_\mathcal{C}, \mathcal{T}_\mathcal{R}, \mathcal{I})$*, entonces* $\mathcal{G}_1 \succeq \mathcal{G}_2$ *si y sólo si* $\exists\ \pi$*, una proyección de* $\mathcal{G}_1$ *en* $\mathcal{G}_2$*.*

Demostración: Trivial a partir de la Definición 4.21.

■

CAPÍTULO V

Procesamiento del lenguaje natural

Uno de los aspectos fundamentales del comportamiento humano es el lenguaje. Se trata de la herramienta que posibilita al hombre expresar sus ideas y pensamientos, en función del conocimiento que éste posea sobre el mundo y transmitirlos a sus semejantes. Estos lenguajes pueden materializarse mediante la utilización de signos que producen la comunicación [343]. De hecho, no sólo sirven para comunicarnos oralmente, sino que también son el vehículo para almacenar información en forma escrita. No es, por tanto, extraño que todas las civilizaciones hayan desarrollado disciplinas encargadas de estudiar el lenguaje, pudiendo clasificarse en grupos bien diferenciados [128]. En esta línea, investigadores y estudiosos se han planteado desde los albores del conocimiento humano la tarea de reflejar la organización y funcionamiento de las estructuras tanto de procesos lingüísticos como cognitivos [217].

En este marco surge el PLN, la disciplina encargada de aglutinar los esfuerzos para producir sistemas informáticos que posibiliten la comunicación entre hombre y máquina, por medio de la voz o del texto. Una disciplina tan antigua como el uso de los ordenadores [339], y que relaciona técnicas de modelado en diferentes campos [195], incluyendo, por ejemplo:

- La computación, que provee métodos para representar modelos, diseñar e implementar algoritmos para herramientas de *software*.
- La lingüística, que contribuye con nuevos modelos lingüísticos y procesos.
- La matemática, encargada de proponer modelos formales y métodos de análisis.
- La neurociencia, que explora los mecanismos mentales.

Siguiendo este modelo, para que la comunicación entre personas y/o sistemas informáticos funcione, tiene que existir una interoperabilidad semántica [134]. Es por

tanto necesario algún tipo de protocolo bien definido en el que partiendo de una representación tangible del lenguaje del emisor, el receptor sea capaz de extraer de forma fehaciente y precisa los componentes o conceptos contenidos en dichas representaciones.

Para cumplir este objetivo, un sistema de PLN necesita hacer uso de una cantidad considerable de información acerca de las estructuras del lenguaje que permitan construir esa representación semántica del texto, por lo que otro factor de interés a tener en cuenta es el propio conocimiento lingüístico. En este sentido, las estructuras de cualquier lenguaje humano se pueden organizar naturalmente en tres niveles [73]: un *nivel léxico*, uno *sintáctico* y uno *semántico*.

5.1 | Nivel léxico

Como primer paso, abordaremos el estudio de la *morfología*, la parte de la lingüística que se ocupa de la estructura interna de las palabras y de sus procesos de formación.

Definición 5.1 *Un* morfema *es la unidad distintiva mínima de la gramática, es decir, la unidad mínima de significado.*

■

La idea fundamental es que estos morfemas pueden ser combinados (o no) para formar palabras. Así, en función del significado que transmiten, éstos se dividen en dos clases: los *morfemas léxicos* y los *gramaticales* [138, 353].

Definición 5.2 *Un* morfema léxico *es la unidad mínima con significado léxico (relacionado con el mundo real). Se suele denominar como* lexema *o* raíz de la palabra.

■

Como quiera que siempre es posible añadir, con relativa facilidad, nuevos morfemas léxicos a una lengua, decimos que estos morfemas constituyen una clase «abierta» de palabras.

Definición 5.3 *Un* morfema gramatical *es aquél cuyo significado y función son intralingüísticos, es decir, aportan contenido gramatical. Más concretamente, podemos distinguir:*

- morfemas libres*: Son los que pueden aparecer como palabras independientes, pero sin aportar información semántica. Se trata en definitiva de las preposiciones, conjunciones y artículos.*
- morfemas dependientes*: Son los elementos que acompañan a la raíz para completar su significado, denominados* afijos*. Los más comunes son los* prefijos,

es decir, aquéllos que preceden a la raíz; y los sufijos*, es decir, aquéllos que se encuentran pospuestos al lexema. Hay que destacar que dichos morfemas pueden sufrir variaciones en su forma como consecuencia del contexto fonológico, denominándose* alomorfo*.*

■

Debido al hecho de que casi nunca se pueden añadir nuevos morfemas gramaticales a una lengua, se dice que constituyen una clase «cerrada» de palabras.

Ejemplo 5.1 *La palabra francesa* «soleil» (`«sol»`) *es en sí una raíz ya que no presenta morfemas gramaticales. Pero, si consideramos la palabra también francesa, que a su vez es de la misma familia que la anterior,* «ensoleiller» (`«solear»`)*, ésta sí posee diversos morfemas que permiten descomponerla.*

■

Ejemplo 5.2 *Supongamos que tenemos la palabra en francés* «inutile» (`«inútil»`)*. En este caso, el prefijo viene dado por* «in-» *y la raíz es* «utile» (`«útil»`)*.*

Supongamos ahora que tenemos la palabra también en francés «rhomboïde» (`«romboidal»`)*. Aquí, la raíz será* «rhomb-» (`«romb-»`)*, donde el sufijo es* «-oïde» (`«-oidal»`) *y expresa la idea de semejanza y forma.*

■

Ejemplo 5.3 *El morfema gramatical* «in-»*, en francés, tiene tres alomorfos:* «i-» *ante* /l/ *o* /r/*, duplicando la consonante en francés:* «illégal» (`«ilegal»`)*,* «irréel»» (`«irreal»`)*;* «im-» *ante* /p/ *o* /b/*:* «impossible» (`«imposible»`)*, e* «in-» *en el resto de casos.*

■

De este modo, la morfología permite delimitar, definir y clasificar unidades, de tal manera que cada una de ellas pueda ser combinadas para formar palabras. Podemos aquí diferenciar tres procesos:

- La *flexión*: Es la alteración que experimentan las palabras mediante morfemas gramaticales para expresar sus distintas funciones dentro de la oración y sus relaciones de dependencia o concordancia con las demás palabras. Así, los afijos de flexión no cambian la categoría sintáctica de las raíces a las se conectan. Por ejemplo, el lexema francés «*plante*» (`«planta»`), que en sí es un sustantivo, adquiere un significado más específico si se le añade el morfema flexivo «*-s*», indicador del plural, dando lugar a «*plantes*» (`«plantas»`) que a su vez sigue siendo un sustantivo.

  Concretamente, a la flexión verbal se le denomina *conjugación*, y a la nominal, *declinación*, que se suele aplicar a sustantivos, pronombres y adjetivos.

- La *derivación*: Describe como son creadas nuevas palabras con la ayuda de afijos. Por ejemplo, el adjetivo en francés «*dentelé*» («`dentado`») se deriva del sustantivo «*dent*» («`diente`»). Otro ejemplo es el adjetivo también francés «*verdâtre*» («`verdoso`») que se deriva del también adjetivo «*vert*» («`verde`»).

  Esto permite tener un léxico que designa diferentes sentidos a partir de un número mucho más reducido de raíces o lexemas.

- La *composición*: Se ocupa de la construcción de palabras nuevas combinando morfemas léxicos, como en «*girasol*», de «*gira*» y «*sol*». Resulta curioso, pero esta palabra también consiste en una composición en francés. Así, «*tournesol*» procede de «*tourne*» («`gira`») y «*sol*».

Hasta ahora hemos descrito la estructura interna de las palabras y su proceso de formación. Llegados a este punto, uno de los primeros pasos en cualquier aplicación PLN consiste en transformar el flujo de caracteres de entrada en un flujo de unidades léxicas de más alto nivel. El proceso de identificación de estas unidades, denominado *análisis morfológico*, y la asignación de las etiquetas candidatas a cada una de ellas [299], tales como su género, número o persona, llamado *etiquetación*, es lo que denominamos *análisis léxico*. Así, por ejemplo, dado el término francés «*aiguillons*» («`aguijones`»), se indicará que se trata de un nombre masculino plural.

5.1.1 | Análisis morfológico

Desde un punto de vista computacional, el *análisis morfológico* suele ligarse a la denominada *morfología de dos niveles* [169, 170], un modelo general aplicable a cualquier idioma, que permite considerar las palabras como una correspondencia entre su nivel superficial, representando su forma gráfica, y el nivel léxico o profundo que incluye la concatenación de morfemas almacenados en un sistema de diccionario. Éstas se clasifican según sus posibles encadenamientos, de tal manera que se regulen las secuencias posibles de raíces y afijos. A la forma gráfica que adopta es a lo que se conoce como *forma*.

Definición 5.4 *Se denomina* forma *a la unidad lingüística sintácticamente atómica, es decir, una unidad considerada como no descomponible desde el punto de vista sintáctico.*

■

Definición 5.5 *Se denomina* forma compuesta *a aquélla que se compone de varias cadenas de caracteres separadas de sus vecinos por espacios o por alguna otra marca tipográfica, como la puntuación. Por convención se representan uniendo cada una de ellas mediante el símbolo* «_». *Un ejemplo sería* «al_contrario». *Del mismo modo, se denomina* forma simple *a aquélla compuesta por únicamente una de estas cadenas de caracteres, como por ejemplo,* « nervadura».

■

Definición 5.6 *Se le llama* amalgama *o* contracción *a una cadena que es el resultado de la fusión de varias formas, como en el ejemplo francés* «du» (`«del»`)*, cuyas formas son* «de + le» (`«de + el»`).

■

Definición 5.7 *Se le llama* forma especial *a una forma ausente del léxico, como por ejemplo, nombres científicos, fechas y dimensiones. También se suelen denominar* entidades nombradas*. Por convención, las formas especiales poseen una etiqueta que las identifica, cuyo símbolo de comienzo es el* «_» *y están constituidas por mayúsculas, como por ejemplo _SCIENTIFIC_NAME, _DATE y _DIMENSION, usadas para identificar el tipo de entidad nombrada.*

■

En la morfología de dos niveles, el nivel léxico consiste en una representación abstracta, donde cada entrada consta de raíz, morfemas que pueden concatenarse con la entrada, y el rasgo morfológico que se quiera expresar. El superficial refleja la realización del nivel anterior en forma de palabra concreta. A continuación, el Ejemplo 5.4 muestra esa correspondencia.

Ejemplo 5.4 *En la Fig. 5.1 se representa la palabra en francés* «nervures» (`«nervaduras»`) *como una correspondencia entre el nivel léxico, que representa una concatenación de morfemas con la raíz, y el nivel superficial, que representa la concatenación de letras que conforman la actual palabra. Así, a partir de la raíz francesa* «nerf» (`«nervio»`)*, si se le concatena los morfemas adecuados, se obtiene a nivel superficial la palabra* «nervures» (`«nervaduras»`).

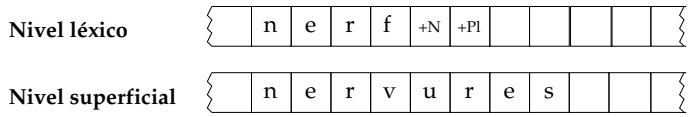


Figura 5.1: Nivel léxico y superficial en la morfología de dos niveles

En este sentido, el símbolo +N *representa el rasgo morfológico de nombre y* «+Pl» *representa el de plural.*

■

Definición 5.8 *Se le llama* lema *a la forma canónica de una palabra, es decir, la forma por la que aparece en el diccionario. Por ejemplo el lema de* «nervaduras» *es* «nervadura».

■

Para realizar la correspondencia entre ambos niveles se necesita disponer de una información mínima [155]. Por un lado, un *lexicón* que recoja las raíces o los *lemas*, y sus afijos a emplear, junto con la información básica acerca de los mismos. Por el otro, un modelo de ordenación para la aplicación de los morfemas, conocido como *morfosintaxis*.

Pero además, una serie de reglas ortográficas que modelen los cambios que se producen en la palabra durante la adjunción de los morfemas, y que actúan directamente como restricciones.

Ejemplo 5.5 *Retomando el Ejemplo 5.4, si el léxico posee dos niveles y la palabra francesa en el nivel superficial es* «nervures» (`«nervaduras»`)*, existe una entrada en el léxico que permite hacer las correspondencias siguientes*

$$n{:}n \quad e{:}e \quad r{:}r \quad f{:}v \quad +N{:}ure \quad +Pl{:}s$$

Cada bloque separado por espacios se identifica con la correspondencia entre el/los carácter/es del nivel léxico (a la izquierda) y el/los del nivel superficial (a la derecha). Esto quiere decir que, para formar la palabra francesa «nervures» (`«nervaduras»`) *sólo es necesario aplicar una modificación de la letra* f *en* v *y añadir los morfemas gramaticales derivativo sufijo* «-ure» *y flexivo de plural* «-s».

De este modo, para conseguir formar el plural de todos los posibles nombres regulares a nivel superficial, será necesario que el lexicón incluya para cada uno de ellos su nivel léxico, es decir, su raíz, así como todos los morfemas gramaticales utilizados.

■

Ejemplo 5.6 *Siguiendo con el Ejemplo 5.5, la variación de la raíz francesa* «nerf» `«nervio»` *en función del morfema derivativo sufijo viene dado en base a la regla*

$$f{:}v \Leftrightarrow \_[aeiouáàéè]$$

Esto expresaría que el carácter «f» en la forma léxica se sustituye por «v» en la forma superficial si y sólo si va seguido de una vocal acentuada o no. Mediante esta regla sería posible dar cuenta de la formación de palabras a las que se le añade el morfema «-ure» *del tipo* «nervure» (`«nervure»`) *o el morfema* «-ation» *del tipo* «nervation» (`«nervación»`).

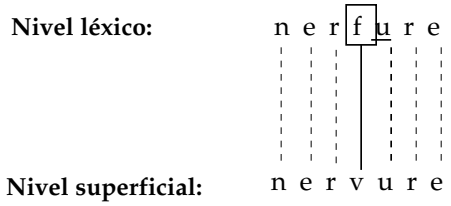


Figura 5.2: Aplicación de reglas en la morfología de dos niveles

■

Ejemplo 5.7 *Supongamos que queremos aplicar el prefijo* «in-» *sobre una palabra en francés. Es necesario considerar todas las posibles variaciones en la forma de dicho morfema en función del contexto. De este modo, será necesario incluir la regla*

$$n{:}m \Leftrightarrow \_\, p$$

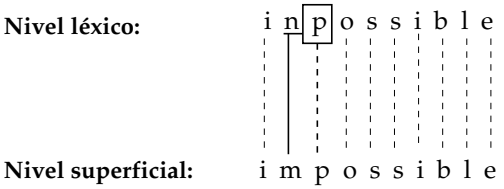


Figura 5.3: Aplicación de reglas en la morfología de dos niveles

Esto expresaría que el carácter «n» en la forma léxica se sustituye por «m» en la forma superficial si y sólo si va seguido de una «p». Mediante esta regla sería posible dar cuenta de la formación de palabras a las que se le añade el morfema «in-» *del tipo* «impossible» (`«imposible»`).

■

De esta forma, el análisis morfológico de una palabra es un conjunto de reglas que hacen corresponder secuencias de letras del nivel superficial a secuencias de morfemas y rasgos morfológicos del nivel léxico.

5.1.2 | Etiquetación

La etiquetación del LN es un proceso que consiste en marcar las palabras de un texto, asignando a cada una de ellas una categoría léxica basándose tanto en su definición como en su relación con las palabras adyacentes relacionadas en la frase [229]. Una forma simplificada de etiquetación es la que identifica las palabras de una frase por su categoría léxica: nombre, verbo, adjetivo o determinante. Sin embargo, este proceso resulta sensiblemente más complejo que manejar un diccionario de palabras con su correspondiente etiqueta, ya que algunos términos pueden pertenecer a diferentes categorías dependiendo del papel que jueguen en una frase concreta. Es lo que se conoce como *ambigüedad léxica*.

Así, si preguntásemos a alguien acerca de la categoría léxica de la palabra «*rosa*», es muy probable que la respuesta fuese que depende del contexto y como ejemplo ilustrativo podríamos analizar la frase: «*Pon la rosa al lado de la blusa rosa*», en la que la palabra «*rosa*» desempeña diferente función sintáctica dependiendo de su posición: sustantivo femenino singular o adjetivo femenino singular. La elección de la categoría correcta en

casos como el del ejemplo sólo es posible a partir del estudio del contexto de la palabra que presenta la ambigüedad.

Conocer la etiqueta correcta de cada palabra de una oración será de ayuda en la fase de desambiguación sintáctica, pero la desambiguación a nivel morfológico requiere, a su vez, cierta clase de análisis sintáctico, ya que es necesario en ocasiones determinar los contextos de las palabras. En cualquier caso, el proceso de etiquetación debe resolver este tipo de ambigüedades, determinando cuál de las alternativas resulta ser la que mejor encaja en el contexto en el que aparece.

5.2 | Nivel sintáctico

Una vez identificadas y analizadas individualmente las palabras que componen un texto a nivel léxico, el siguiente paso consiste en establecer cómo se organizan y relacionan, y cual es la función de cada cual, es decir, identificar la estructura sintáctica.

Siguiendo esta idea, se tiene tendencia a pensar que las palabras que componen una frase lo hacen como una progresión siguiendo una sola dimensión. Pero una propiedad del LN es que la sintaxis tiene dos dimensiones: una explícita y otra implícita. La primera hace referencia al orden lineal de las palabras. La segunda se centra en la estructura jerárquica que presentan dichos vocablos mostrándolos, la mayoría de las veces, como una dependencia [128] tal y como se observa en el Ejemplo 5.8. En cualquier caso, la estructura viene determinada por un modelo gramatical que la describe y delimita, y que permite generar una representación de la misma en forma arborescente.

Ejemplo 5.8 *Supongamos que tenemos las frases de la Fig. 5.4. En la primera, el grupo de palabras «*de un rosal*» está unido al grupo «*Una hoja*», considerando el orden lineal de las palabras en la frase. Por otro lado, y pensando en como se relacionan ambos grupos implícitamente, se puede establecer una relación entre ellos mediante una dependencia que nos indica cuál es el tipo de hoja al que se hace referencia.*

```
Una hoja de un rosal
Una hoja teñida de un rosal
```

Figura 5.4: Diferencia entre dimensión implícita y explícita de la sintaxis

*Sin embargo, en el segundo caso, este mismo grupo de palabras ya no se une explícitamente con «*Una hoja*» sino que lo hace con «*teñida*», siempre considerando el orden de las palabras. Aquí, a diferencia del caso anterior, no existe una relación de dependencia entre «*teñida*» y «*de un rosal*» debido a que ambas están en relación con «*Una hoja*», permitiendo así indicar cuál es el tipo de hoja, pero también cuál es su color.*

Como vemos, lo que hace la diferencia entre las dos interpretaciones no es, el orden lineal de las palabras, puesto que el grupo «de un rosal» *se encuentra en ambos casos al final de la frase, sino las relaciones de dependencia implícitas que se establecen entre ellas.*

■

Llegados a este punto, resulta necesario, introducir el concepto de *ambigüedad sintáctica*, que se produce cuando para una misma frase existe más de una estructura válida de reconocimiento.

Definición 5.9 *Se dice que una gramática* $\mathscr{G} = (N, \Sigma, P, S)$ *es una* gramática ambigua *si y sólo si* $\exists x \in \mathscr{L}(\mathscr{G})$*, para la cual existen al menos dos análisis sintácticos válidos. Asimismo, diremos que un* lenguaje $\mathscr{L}$ no es ambiguo *si y sólo si existe una gramática* $\mathscr{G}$ *no ambigua tal que* $\mathscr{L}(\mathscr{G}) = \mathscr{L}$*. En caso contrario, diremos que* $\mathscr{L}$ *es un* lenguaje ambiguo.

■

En lo que se refiere a la complejidad descriptiva de un LN, aún hoy se discute cuál sería la posición real que en la Jerarquía de Chomsky [61] ocuparían este tipo de lenguajes, si bien se cree que deberían de situarse entre los LIC's y los LDC's, posiblemente más cerca de los segundos que de los primeros. En la práctica, muchas aplicaciones en PLN usan las GIC's como esqueleto gramatical, proporcionando la estructura jerárquica interna de las propias oraciones. Gracias a ellas, se pueden describir construcciones recursivas que no podían ser tratadas a través de las GR's, así como expresar la alternancia y la opcionalidad. Además, poseen propiedades formales que facilitan el diseño de algoritmos de análisis sintáctico eficaces. Sin embargo, los LIC's no parecen ser lo suficientemente potentes como para expresar en su totalidad los LN's puesto que existen construcciones básicas [11], tales como por ejemplo la *replicación*[1] o las *concordancias*[2], que no pueden ser tratadas desde la óptica de una GIC.

Es importante también señalar que una buena parte de las construcciones sintácticas que se pueden obtener a través de los LN's sólo van a depender débilmente del contexto en el cual son aplicadas. Si no fuera así, la semántica asociada sería de una complejidad tal que su comprensión y utilización por un humano sería poco práctica. De este modo, formalismos como los *lenguajes suavemente dependientes del contexto* (LSDC's), situados entre los LIC's y los LDC's, suponen intuitivamente un buen compromiso entre potencia expresiva y eficacia computacional en su análisis. Además de esto, parece razonable pensar que si la estructura sintáctica asociada a las frases es jerárquica y se representa habitualmente por un esquema de tipo arborescente, el mecanismo descriptivo para la sintaxis de los LN's debiera ser un formalismo gramatical que utilice explícitamente árboles.

[1] como ocurre en ciertas variantes del alemán, y que producen lenguajes de la forma $\{ww\}$ [341].
[2] como ocurre en el holandés, y que producen lenguajes de la forma $\{a^n b^m c^n d^m / n, m > 0\}$ [110]

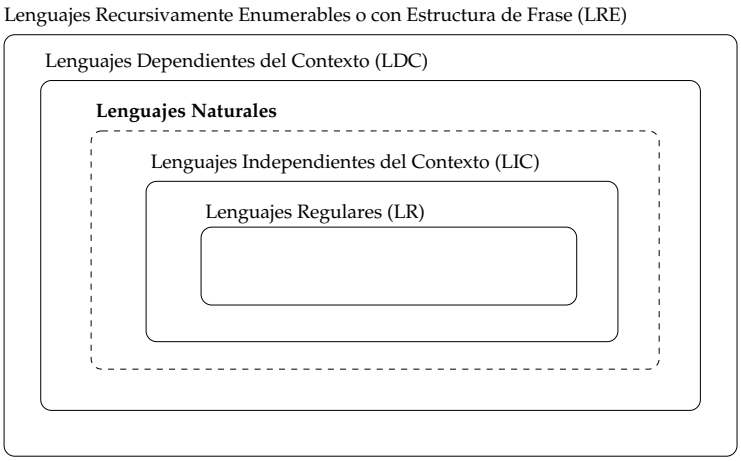


Figura 5.5: Diagrama de Venn correspondiente de la Jerarquía de Chomsky

En este sentido, las *gramáticas de adjunción de árboles* (GA's)[3] [150] se han mostrado adecuadas en el tratamiento de los fenómenos sintácticos que aparecen en el LN [151]. Los lenguajes por ellas generadas, los *lenguajes de adjunción de árboles* (LA's), constituyen además una de las subclases más populares de los LSDC's.

Una vez señalados los conceptos de lenguaje como conjunto de cadenas y el de gramática como formalismo descriptivo, el objetivo del *análisis sintáctico* es reconocer si una cadena pertenece al lenguaje generado por la gramática y proponer una representación apropiada de dicho proceso de reconocimiento. Los algoritmos que realizan sólo la primera de las dos acciones se denominan *reconocedores*, mientras que a aquéllos capaces de generar además una representación del proceso, es decir, capaces de obtener el árbol sintáctico de la cadena procesada, se les denomina *analizadores sintácticos*. En este punto, podemos introducir una primera clasificación de este tipo de algoritmos [7, 43, 88, 99, 119, 322, 323, 324, 325] en razón del tipo de estrategia a aplicar en la construcción de árboles:

- Los *algoritmos ascendentes* son aquéllos que construyen el árbol desde las hojas hasta la raíz, y se corresponden con una derivación por la derecha de las reglas gramaticales.
- Los *algoritmos descendentes* actúan en sentido contrario a los ascendentes, de la raíz a las hojas, y se corresponden con una derivación por la izquierda de las reglas gramaticales.
- Las *estrategias mixtas* combinan los dos enfoques anteriores, habitualmente con una fase descendente estática que predice el conjunto de posibles derivaciones gramaticales, para luego aplicar una arquitectura ascendente en la interpretación efectiva del texto, guiada esta última por el análisis descendente previo.

[3] el Apéndice C trata con más detalle las GA's.

Podemos igualmente establecer clasificaciones basándonos en otros criterios. Es el caso del tratamiento del posible no determinismo en el análisis, factor de especial importancia en el caso de los LN's:

- *Algoritmos basados en retroceso*. Cuando varias alternativas son posibles [7], se escoge sólo una y, si ésta resulta infructuosa, se retrocede hasta el último punto de no determinismo y se escoge otra. Los cálculos realizados en las alternativas exploradas anteriormente se desechan. Este enfoque es sencillo, pues economiza espacio y recursos, pero presenta varios problemas:
  - Los cálculos realizados en las alternativas exploradas anteriormente se desechan. Por tanto, si éstos vuelven a ser necesarios en una alternativa posterior, deberán ser calculados de nuevo.
  - El criterio de selección de las alternativas puede no ser óptimo, llevándonos a una elección incorrecta que no conduzca a una solución y, por tanto, a cálculos innecesarios.
- *Algoritmos basados en programación dinámica*. Mediante estas técnicas [43, 88, 325], se almacenan los cálculos ya realizados de forma que no sea necesario repetirlos en caso de que se vuelvan a necesitar. Esto nos permite compartir cálculos entre las diversas alternativas de análisis derivadas de una gramática ambigua, solucionando en parte los problemas de los algoritmos basados en retroceso, en particular la multiplicación innecesaria de cálculos y los problemas de no terminación, cuyo origen se sitúa en la presencia de ciclos de análisis.

En el contexto del LN, especialmente complejo, cobran protagonismo frente a las técnicas clásicas de análisis sintáctico completo o convencional, otros acercamientos alternativos en el objetivo de asegurar el proceso de análisis sintáctico frente a los problemas de cobertura gramatical incompleta y/o presencia de errores sintácticos:

- *Análisis sintáctico robusto*. Al contrario que ocurre con los lenguajes formales, en el LN no siempre es posible analizar correcta y completamente una cadena de entrada, debido a la dificultad de diseñar una gramática exhaustiva que cubra todas las posibles sentencias del lenguaje a reconocer o a la presencia de construcciones no gramaticales introducidas por el propio usuario. Esto nos obliga a realizar un análisis sintáctico en presencia de lagunas gramaticales. A este tipo de análisis se le califica de robusto [99, 322, 323].
- *Análisis sintáctico parcial*. Emplearemos este término para referirnos a las técnicas de análisis capaces de obtener, a ser posible, el análisis completo de una entrada, y, en su defecto, posibles subanálisis de menor entidad [254, 292].
- *Análisis sintáctico superficial*. No siempre es necesario realizar un análisis detallado de la estructura sintáctica del texto. Para algunas tareas basta realizar

un análisis superficial de la misma [119], identificando únicamente las estructuras de mayor entidad, tales como frases nominales, grupos preposicionales, etc. En este contexto es común la utilización de cascadas de autómatas o traductores finitos [3, 4].

5.3 | Nivel semántico

Nuestro propósito es llegar a identificar el significado de las frases, retomando los de las palabras en el contexto de su estructura sintáctica. Un punto esencial a abordar es el de las representaciones semánticas, ya que para el caso de los elementos lingüísticos, como por ejemplo las palabras o sintagmas, éstas deben ser capturadas mediante estructuras formales para su posterior tratamiento. En este sentido, cualquier teoría que pretenda abordar la comprensión de textos debería dar cuenta de cómo el sistema cognitivo humano es capaz de reproducir su estructura jerárquica, junto con las relaciones que define. En caso contrario, no sería posible la asimilación cabal del mensaje que el autor trata de transmitir.

En este proceso, se trata en definitiva de imitar las estrategias de asimilación del conocimiento puestas en marcha por un humano, elaborando una representación interna de la semántica del texto en cuestión. Durante este proceso de decodificación, la información externa se divide en pequeños fragmentos [231] que se vuelven a agrupar en función de sus exigencias.

5.3.1 | Representaciones semánticas

En este sentido, han surgido acercamientos varios. Tomando como base la clasificación realizada por Laurière [178], la Fig. 5.6 presenta distintos formalismos que van de lo más estructurado a lo declarativo.

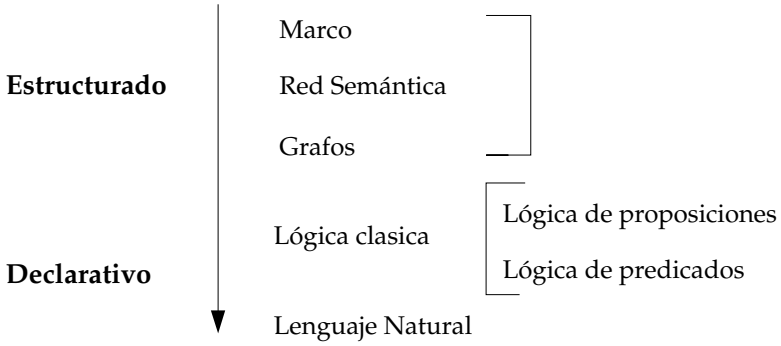


Figura 5.6: Clasificación del conocimiento basada en la realizada por Laurière

5.3.1.1 | Representación declarativa

Recoge una aproximación que permite la representación por separado del conocimiento y de las técnicas para su procesado. De esta forma, cuando se trata de textos con incertidumbre, se pueden ensayar distintas representaciones de conocimiento para resolver uno o varios problemas relacionados y, en función del rendimiento y resultados observados, su representación puede refinarse hasta alcanzar un alto grado de eficiencia. En este sentido, un ejemplo de representación declarativa sería la proporcionada por la *lógica formal* [107].

En definitiva, estas representaciones se basan en la utilización de razonamientos que sean efectivos y que respondan de un modo *categórico*[4]. A dichos modelos de razonamiento se les denomina *cálculo*[5] [122], y vienen dados por una estructura sintáctica que no constituirá un lenguaje hasta que no se le haya aportado la interpretación *semántica*. Para ello, se les deberá de incorporar un vocabulario, unas reglas de formación y las reglas de transformación.

La lógica, por tanto, se estructura en cálculos, que no dejan de ser una simple estructura sintáctica. En este sentido, habrá que cuidar que las sentencias aseguren su validez formal, lo que se consigue aceptando sólo fórmulas bien construidas y reglas de inferencia que sean lógicamente válidas [122]. En cualquier caso, la potencia y expresividad de la representación dependerán del tipo de lógica considerada y ésta, a su vez, viene determinada por la sintaxis de esos cálculos. Así, la lógica formal puede caracterizarse mediante un diagrama de Venn, tal como se observa en la Fig. 5.7, dónde sobre un cálculo se incorpora otro que contiene más recursos expresivos y que necesita de nuevos elementos, o incluso que evita restricciones del uso de estos recursos. Esta figura debe de interpretarse de una forma monótona ascendente en lo que a expresividad se refiere. Es decir, la expresividad de los sistemas crece en los de nivel superior, pero no en sentido contrario. Teniendo esto presente, podemos clasificar las lógicas formales como sigue [122]:

- *Lógica de proposiciones* (LP). Es el cálculo básico de la lógica formal, cuyas fórmulas representan proposiciones. En este cálculo, la deducción se establece en una relación de implicación entre las premisas y la conclusión. Debido a que las variables son únicamente booleanas, y que no permite el uso de cuantificadores, el estudio de la LP es sencillo, aunque resulta difícil generalizar los razonamientos si no es por enumeración de la totalidad de los casos individuales, lo que imposibilita el tratamiento de dominios de definición infinitos. Ello justifica en sí la consideración de un formalismo más potente, la lógica de primer orden.

[4] de manera general, *categórico* hace referencia al discurso en el que se afirma algo como verdadero y sin condiciones. Por asimilación también a un enunciado afirmativo.

[5] según el diccionario de la RAE, *cálculo* es un sistema lingüístico formal en el que lo esencial son las reglas sintácticas y que permite realizar operaciones sin necesidad de interpretar los símbolos.

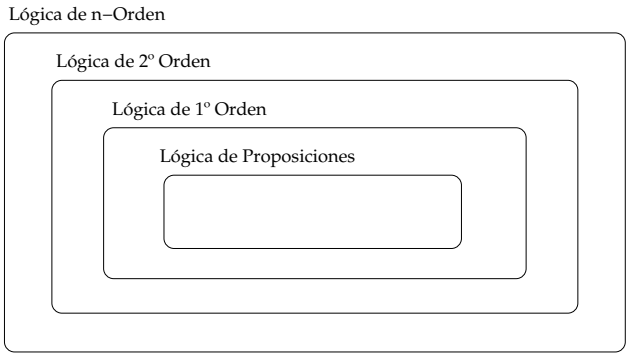


Figura 5.7: Diagrama de Venn de la lógica moderna

- *Lógica de primer orden* (LPO). Caracterizada por la introducción del concepto de variable y porque permite usar cuantificadores sobre los elementos individuales, lo que posibilita expresar la pertenencia o posesión de propiedades por parte de los distintos individuos y también las relaciones entre ellos.

  Ejemplo 5.9 *Un ejemplo, usando un cuantificador universal, se aplicaría en la frase «*Todos los Cynometras son árboles*» pudiendo formalizarse empleando los predicados:*

  Cynometra(x)=«x es Cynometra» y Árbol(x)=«x es árbol»

  como:

  $$\forall x,\ (\textit{Cynometra(x)} \Rightarrow \textit{Árbol(x)})$$

  donde x es el término, Cynometra(x) y Árbol(x) son fórmulas atómicas y (Cynometra(x) ⇒ Árbol(x)) también es una fórmula.

  ■

  En este sentido, la aproximación semántica basada en LPO es de las soluciones más extendidas [344], sobre todo con el fin de aplicarse a sistemas de RI.

- *Lógicas de 2º (3º, 4º, ... , n) orden*. Usando como base la LPO, podremos cuantificar sobre los predicados (propiedades o relaciones) obteniendo una lógica de segundo orden, o sobre los predicados de predicados obteniendo una lógica de tercer orden, y así sucesivamente.

Otra forma de clasificar las lógicas formales, igualmente en relación directa con su poder expresivo y capacidad de representación, viene determinada por el número de *valores de verdad*[6], es decir, el significado o interpretación de una proposición. En este

[6]el conjunto de valores que indican en qué medida una declaración es verdadera, que se acepten en los cálculos.

sentido, hay que distinguir entre la lógica clásica y la no clásica. Cuando los cálculos lógicos son *bivalentes*, es decir, que sus fórmulas pueden ser verdaderas o falsas, y no puede ocurrir que lo sean a la vez, se le llama *lógica clásica* [260]. Si en los cálculos lógicos se contemplan más valores de verdad que lo verdadero y lo falso, entonces se habla de *lógica no clásica* [31, 106, 242, 358], y surgen por la limitación expresiva de la LPO. Es el caso de:

- La *lógica modal*. Incorpora como operadores aquellos modificadores relativos a lo que es necesario y lo que es posible [31]. En este tipo de lógica se podrían expresar formalmente cosas como *«posiblemente los Cynometras que tienen dos pares de foliolos son Cynometras Sanagaensis»*.
- La *lógica temporal*. Incorpora parámetros temporales [106]. Para muchas sentencias su verificación depende del momento en que se produce, como cuando nos referimos al color de los pétalos de una flor en función de la época del año que la estamos describiendo.
- Las *lógicas multivaloradas*. Aquéllas que contemplan un número finito de valores de verdad. Por un lado, aquéllas que pueden tener tres o más valores; lo verdadero, lo falso y otros valores intermedios considerados desconocidos o inciertos. Se les denomina *lógicas finitamente valoradas* [358]. Por ejemplo, el enunciado *«la Cynometra Manii crece en entornos húmedos»* puede ser verdadero, falso o incierto si la Cynometra Manii se da en entornos de humedad intermedia. Por otro lado, las que consideran infinitos valores, generalmente establecidos el intervalo [0, 1], se llaman *lógicas infinitamente valoradas* [242]. Si tomamos como ejemplo *«un árbol x, cuya altura es de 5 metros»* podría poseer un grado de pertenencia $0'6$ para el valor *«alto»* y un grado de $0'4$ para el conjunto *«bajo»*, aunque también tendría un grado de pertenencia de 1 para el valor *«mediano»*.
- *La lógica borrosa*. Considera valores de verdad difusos como *«muy verdadero»*, *«bastante verdadero»*, *«poco verdadero»*, *«poco falso»*, *«bastante falso»* o *«muy falso»*, que se representan mediante el uso de números borrosos y a los que subyace toda una aritmética con este tipo de números [310].

5.3.1.2 | Representación estructurada

La lógica, aunque constituye una buena formalización del conocimiento, no siempre resulta definitiva cuando tenemos que describir una estructura compleja como parte de un diseño de implantación, comprometiendo su aplicación fundamentalmente por falta de legibilidad y por el tipo de relaciones a modelizar.

En concreto, resulta a menudo útil representar aspectos como estructuras y relaciones que permiten agrupar las propiedades de los objetos del mundo en unidades de descripción. Esto permite al sistema focalizar su atención en un objeto concreto, sin

considerar el resto de hechos que conoce. Ello es importante para evitar la explosión combinatoria que supone explorar la totalidad del espacio de cálculo.

En este sentido, las representaciones estructuradas tienen una gran potencia expresiva y permiten una fácil interpretación del mismo. Entre las más populares podemos considerar las *redes semánticas*, basadas en el uso de grafos y destinadas a la comprensión del LN [82]. Se trata de una estructura de representación del conocimiento lingüístico, donde los nodos pueden representar objetos, entidades, atributos, eventos o estados; y los arcos representan las relaciones existentes entre ellos. En particular, pueden agruparse en dos tipos:

- *Sistemas asertivos*. Permiten realizar afirmaciones particulares. En ellas no existen definiciones de conceptos ni clasificaciones jerárquicas, sino solamente afirmaciones concretas. Son sistemas que no excluyen la posibilidad lógica de una contradicción. Para ello se requiere formalizar las relaciones mediante etiquetas que representarán conocimiento declarativo. En este tipo de sistemas se pueden incluir, entre otros, los denominados *modelos de memoria semántica* o *grafos relacionales* [239], y los *grafos de dependencias conceptuales de Schank* [280, 281].

- *Sistemas taxonómicos*. Permiten relacionar los conceptos mediante jerarquías. Los tipos de relaciones que incluyen serán relaciones de instancias, entre conjuntos y subconjuntos, incluyendo relaciones de pertenencia y de propiedades. Es la denominada *jerarquía de conceptos* [36].

Otra de las representaciones más utilizadas son los *marcos*. En el campo de la IA, este término se refiere a una forma concreta de representación de conceptos, llamadas *clases*, y *situaciones estereotipadas*[7]. Fueron propuestos inicialmente por Minsky [206], quién consideraba que la resolución de problemas humanos era el proceso de rellenar huecos en descripciones mentales. Por este motivo, se usan con la finalidad de representar el conocimiento mediante el rellenado de espacios vacíos [289]. En este sentido, permiten superar las limitaciones de la lógica a la hora de abordar problemas como la visión artificial [126], la comprensión del LN [82] o el razonamiento basado en el sentido común [82]. Los marcos son, de hecho, una evolución de las redes semánticas donde el nodo es sustituido por una estructura de datos que representa una situación estereotipada a partir de sus elementos más significativos.

En cualquier caso, el conocimiento expresado mediante cualquiera de estas representaciones estructuradas puede ser traducido a LPO [123].

[7] imagen mental muy simplificada y con pocos detalles acerca de una situación concreta que comparte ciertas cualidades características.

5.3.2 | Análisis semántico

Una vez se dispone de una estructura de representación adecuada, el objetivo es obtener la representación semántica de las frases en un texto. Uno de los enfoques más utilizados es el denominado *análisis dirigido por la sintaxis* [155], basado en el *principio de composición*, según el cual la semántica del todo puede ser obtenida a partir de las de sus partes. Fue Montague [214] quien mostró que el enfoque composicional podía ser aplicado al PLN, introduciendo la estructura de modelos teóricos en la teoría lingüística, y dando lugar de este modo a una integración mucho más fuerte entre las teorías de la sintaxis formal y un amplio rango de estructuras semánticas.

Pero también es cierto que el significado de una frase no puede obtenerse sólo a partir de las palabras y sintagmas que la componen de un modo individual. Es necesario considerar la forma en la que estas estructuras se relacionan. En otras palabras, el significado de la frase depende sustancialmente de su arquitectura sintáctica. En este sentido, el análisis semántico resulta sensiblemente más complejo ya que una frase puede tener asignadas diferentes interpretaciones, lo que constituye un nuevo factor de ambigüedad. Por ejemplo, en la frase «*Voy a darles un pastel a los niños*» puede pensarse en que sólo se dispone de un pastel y se va a repartir entre todos los niños, o por el contrario, en que se dispone de uno para cada uno de ellos.

Del mismo modo, puede existir la posibilidad de que una palabra pueda tener diversos significados según el contexto en el se encuentre y que constituye uno de los principales problemas del análisis semántico. Así, por ejemplo, en la frase «*Juan se sentó en el banco*», se entiende que éste lo hizo en un asiento, mientras que en «*Juan entró en el banco*» se refiere a una entidad financiera. Teniendo esto en cuenta, existen herramientas susceptibles de ser utilizadas en tareas de procesamiento semántico, como son el uso de bases de datos lexicográficas, tipo *WordNet* [97, 125, 205] para el caso del inglés, o su equivalente *EuroWordNet* [338], en el caso de otras lenguas europeas. Las técnicas de *desambiguación del sentido de las palabras* [203] tratan de resolver el problema seleccionando el sentido adecuado de cada palabra en una frase, cuestión especialmente compleja dada la potencial presencia de palabras homónimas y polisémicas. En esencia, se aplican técnicas similares a las utilizadas para realizar la etiquetación de las palabras en el nivel morfológico, pero en lugar de considerar etiquetas morfosintácticas se usan otras de carácter semántico que identifican el significado de los términos.

CAPÍTULO VI

Recuperación de información

Para satisfacer su necesidad de información, el usuario ha de disponer de herramientas capaces de localizar los contenidos de interés, procesarlos, integrarlos y generar una respuesta acorde a los requerimientos expresados. Además, el entorno debería ser capaz de incorporar el LN en su interfaz, permitiendo así la interacción también a aquéllos inexpertos en el manejo de ordenadores.

La globalización y fiabilidad en el acceso a la información ha justificado la popularización de sistemas de RI, haciendo de su diseño e implementación uno de los mayores retos para la comunidad científica, lo que propició el desarrollo de las líneas de investigación específicas que conocemos como RI, *extracción de información* (EI) y *búsqueda de respuestas* (BR).

De un modo general, se puede decir que los sistemas de RI tratan del acceso a la información a partir de una consulta del usuario, así como de la presentación, almacenamiento y organización de sus respuestas [20, 314]. Como resultado proporcionan una lista de documentos [90, 282] que suelen presentarse ordenadamente en función de valores que pretenden reflejar en qué medida cada uno de ellos resulta pertinente a esa consulta.

En una línea análoga, la EI consiste en recuperar aquellos documentos que se ajusten a una consulta dada, aunque añadiendo a esta funcionalidad la de extraer la información y presentarla en un formato de grano más fino, susceptible de ser tratado posteriormente [141]. De este modo, su finalidad es la de realizar tareas de búsqueda de información muy concretas pasando, por ejemplo, del nivel de documento al de párrafo o frase, considerando que las técnicas de PLN aplicables, tales como la lematización, son esencialmente comunes a las de la RI. Este tipo de herramientas se diseñan generalmente de forma específica para la realización de una tarea determinada, en un dominio de conocimiento también concreto.

Si la RI y la EI han facilitado el tratamiento de grandes cantidades de información, la BR persigue una interacción más cercana al usuario, relacionando su pregunta con una respuesta explícita construida a partir de la información disponible en una colección de documentos [72, 105, 317].

A nivel ilustrativo, en esta tesis nos hemos centrado en los sistemas de RI, aunque los resultados obtenidos sean a nuestro entender de interés también en EI y BR. Por ello, el primer paso es introducir una serie de conceptos de uso común que vamos a emplear a lo largo de todo el trabajo, empezando por los más elementales, ilustrados en la Fig. 6.1. Así, la noción de *documento* hace referencia a una unidad de texto almacenado por el sistema y que contiene datos de interés disponibles para su recuperación [324]. Por su parte, denominaremos *colección* o *base documental* a un repositorio de documentos que denotamos por $\mathscr{C}$.

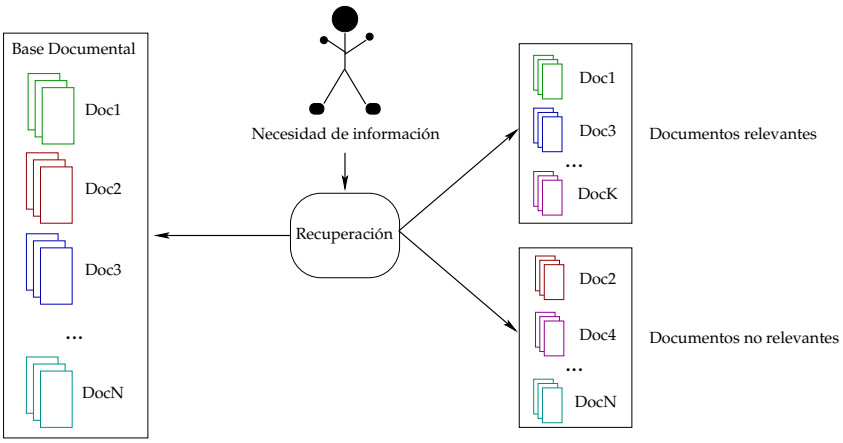


Figura 6.1: Proceso de RI

En cuanto a los usuarios, éstos expresan sus necesidades de información, mediante consultas. Como respuesta, el sistema devuelve referencias a documentos que estima relevantes [141], es decir, que satisfacen la necesidad expresada en la consulta, generalmente de forma ordenada [271].

6.1 | Arquitectura de un sistema de RI

Idealmente, un entorno de RI debería procurar únicamente respuestas consideradas relevantes por el usuario a su consulta, pero en la práctica esto no es así, fundamentalmente porque resulta extremadamente complejo trasladar fielmente el sentido de la misma al sistema. Además, existe una carga de subjetividad subyacente que depende de los usuarios, lo que dificulta aún más si cabe dicha tarea.

Por este motivo, a la hora de diseñar un entorno de RI es necesario establecer previamente cual será el tipo de consultas a las que el usuario pretende hacer frente.

En función de ello será preciso definir, por un lado, la forma de representación de los documentos y consultas y, por el otro, el modo de comparación de ambas, es decir, definir el propio modelo de recuperación. Este proceso es el que se muestra en la Fig. 6.2, retomando [20] para formalizar el concepto de modelo de RI.

Definición 6.1 *Un* modelo de RI *es una cuádrupla* $[\mathcal{D}, \mathcal{Q}, \mathcal{F}, sim(d_i, c_j)]$*, donde:*

- $\mathcal{D} = \{d_i\}_{i \in I}$ *es el conjunto de representaciones de los documentos de la colección.*
- $\mathcal{Q} = \{c_j\}_{j \in J}$ *es el conjunto de representaciones de las consultas.*
- $\mathcal{F}$ *es la función que modeliza las representaciones de documentos, consultas y relaciones entre ambas.*
- $sim(d_i, c_j)$ *es una función de ordenación que asocia un número real con los diferentes pares* (d_i, c_j)*, donde* $d_i \in \mathcal{D}, i \in I$ *y* $c_j \in \mathcal{Q}, j \in J$*. Ésta define la* similitud *entre las representaciones de la consulta y el documento, a saber, el valor con el que estimamos la pertinencia del documento* d_i *en relación a la consulta* c_j*.*

■

Dado que los documentos no se almacenan en el sistema de RI y que hemos de realizar operaciones sobre ellos, es necesario obtener primero su representación formal. Por este motivo, tendrán que ser preprocesados y modelizados por un conjunto de descriptores obtenidos mediante una función de representación y que pretenden reflejar la semántica del contenido, tal y como se ilustra en la Fig. 6.2.

Sobre la base de la representación formal de los documentos, aplicamos un proceso denominado *indexación* que generará unas estructuras de datos, llamadas *índices*, que permitirán dar acceso a los descriptores que modelizan el contenido de los documentos. La consulta, redactada mediante un lenguaje de consulta específico, es analizada y transformada de acuerdo al mismo procedimiento utilizado con los documentos, es decir, a través de la función de representación.

Una vez que documentos y consulta están formalmente representados, podremos estimar su proximidad semántica gracias a una *función de comparación*. El conjunto de documentos recuperados se divide en dos grupos. Por un lado, los relevantes recuperados, cuyo contenido posee algún significado relativo a la consulta. Por el otro, los no relevantes, que son aquéllos que se han recuperado erróneamente, provocando ruido en la salida. Los documentos no recuperados pueden dividirse a su vez en relevantes, rechazados por el sistema de manera errónea; y en no relevantes, rechazados de manera correcta. En cualquier caso, resulta necesario formalizar el concepto de relevancia.

Definición 6.2 *Sean* $\mathcal{D} = \{d_i\}_{i \in I}$ *una colección documental y* $\mathcal{Q} = \{c_j\}_{j \in J}$ *un conjunto finito de consultas. Se dice que un documento* $d_i \in \mathcal{D}$ *es* relevante *con respecto a una*

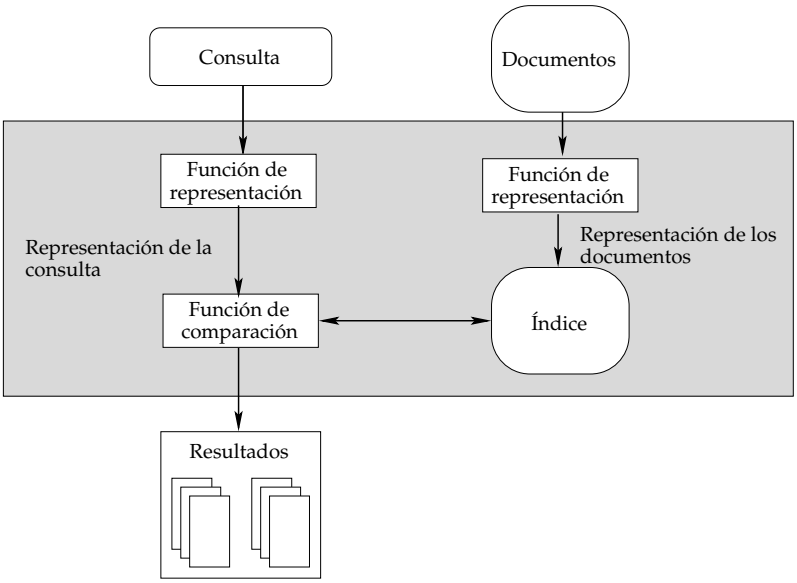


Figura 6.2: Sistema de RI

consulta $c_j \in \mathcal{Q}$ si y sólo si un experto humano considera que dicho documento posee información relativa a la misma. Si no es así, se dice que $d_i \in \mathcal{D}$ no es relevante *a $c_j \in \mathcal{Q}$. Al conjunto de documentos de $\mathcal{D}$ que son relevantes a $c_j \in \mathcal{Q}$, lo denotamos por* $\text{rel}(c_j, \mathcal{D})$*, y por* $\text{nrel}(c_j, \mathcal{D})$ *a los que no lo son.*

■

Una vez comparada la representación de la consulta con la de los documentos, se utiliza la *función de ordenación* para establecer los criterios que van a determinar hasta que punto el documento recuperado puede contener la respuesta buscada. Finalmente, es necesario disponer de una interfaz con el objetivo de facilitar la tarea de consulta, así como la visualización de los resultados.

6.2 | Modelos de RI clásicos

En este sentido, siguiendo el trabajo realizado en [188], los modelos de RI clásicos consideran que un documento está representado por un conjunto de palabras claves como posibles descriptores. La finalidad es utilizarlos para crear los índices, pero también para resumir la semántica del documento, generalmente eliminando aquellos términos denominados *palabras vacuas* [192], es decir, que no poseen significado, entre los que figuran artículos, pronombres, preposiciones, conjunciones o números.

Dado un documento, y un conjunto de términos índices, cada uno de estos últimos puede presentar una relevancia distinta respecto al documento. Por este motivo, introducimos la noción de *peso de un término en un documento*.

Definición 6.3 *Sean* $\{t_i\}_{i\in I}$ *la colección de términos índice y* $\mathcal{D} = \{d_j\}_{j\in J}$ *la colección documental. Denotamos por* $W(t_i, d_j), i \in I, j \in J$ *al* peso asociado al término t_i en el documento d_j*, de tal forma que*

$$\begin{aligned} W(t_i, d_j) &> 0, \ si\ t_i \in d_j \\ W(t_i, d_j) &= 0, \ \text{en otro caso} \end{aligned} \tag{6.1}$$

Dado un documento $d_j, j \in J$ *introducimos* $\overrightarrow{d_j} = [W(t_1, d_j), W(t_2, d_j), \cdots, W(t_p, d_j)]$*,* $p \in I$*, como el* vector de pesos asociados al documento d_j *y a* los términos $\{t_i\}_{i=1}^{p}$*. En este punto, podemos definir a su vez la función*

$$\begin{aligned} g_{t_i} : \{\overrightarrow{d_j}, j \in J\} &\to \mathbb{R}^+, i \in I \\ \overrightarrow{d_j} &\longmapsto W(t_i, d_j) \end{aligned} \tag{6.2}$$

que devuelve el peso del término t_i en $\overrightarrow{d_j}$.

■

Una vez definidos estos conceptos, ahora introduciremos brevemente los modelos teóricos más populares. Se trata del *booleano*, del *vectorial* y del *probabilístico*. Con el fin de utilizar la misma estructura describiremos, por un lado, la estrategia de representación de los documentos y la de las consultas en el espacio de indexación[1], y por el otro, la función de correspondencia empleada para estimar la pertinencia de cada documento con respecto a una consulta dada.

6.2.1 | Modelo booleano

Fue uno de los primeros en desarrollarse. Se basa en el *álgebra de Boole* [71], y permite tratar representaciones generadas a partir de proposiciones, combinando operadores lógicos [314].

6.2.1.1 | Representación de textos

Este modelo considera que los términos están bien presentes o bien ausentes en un documento. Es por ello que la función de representación se consigue asociando un peso binario a cada uno de los términos extraídos en la colección documental: 1 si el término aparece en el documento y 0 cuando no es el caso. Esto es

$$W(t_i, d_j) \in \{0, 1\} \tag{6.3}$$

[1] considerando que tanto documentos como consultas se expresan mediante un vector de pesos de términos índice.

Además, asumimos que los pesos de los términos son mutuamente independientes. Las consultas se representan de manera análoga a la de los documentos.

Sin embargo no resulta sencillo trasladar un concepto de usuario a una expresión booleana. De hecho, en este tipo de modelo, las consultas se pueden componer de términos relacionados entre sí mediante conectores lógicos AND, OR y NOT. Así, una consulta es esencialmente una expresión booleana convencional que puede ser representada como una disyunción de vectores conjuntivos [20, 188], esto es, como una *forma normal disyuntiva*[2].

Ejemplo 6.1 *Supongamos que queremos realizar la consulta expresada de la siguiente manera* $c = t_1 \wedge (t_2 \vee \neg t_3)$*, donde* t_1, t_2 *y* t_3 *son términos índice. Para simplificar su verificación, vamos a reducirla a forma normal disyuntiva. El primer paso es aplicar la propiedad distributiva, por lo que*

$$c = (t_1 \wedge t_2) \vee (t_1 \wedge \neg t_3)$$

Una vez hecho esto, vamos a asociar a $(t_1 \wedge t_2)$ *una expresión booleana que siempre es cierta, es decir, una tautología del tipo* $(t_3 \vee \neg t_3)$*, obteniendo*

$$(t_1 \wedge t_2) \equiv (t_1 \wedge t_2) \wedge (t_3 \vee \neg t_3)$$

Teniendo en cuenta las leyes distributivas, es decir, $A \wedge (B \vee C) \equiv (A \wedge B) \vee (A \wedge C)$ *y* $A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$*, concluimos que*

$$(t_1 \wedge t_2) \equiv (t_1 \wedge t_2) \wedge (t_3 \vee \neg t_3) \equiv (t_1 \wedge t_2 \wedge t_3) \vee (t_1 \wedge t_2 \wedge \neg t_3)$$

Por otro lado, asociando a $(t_1 \wedge \neg t_3)$ *otra expresión booleana que siempre es cierta del tipo* $(t_2 \vee \neg t_2)$*, se obtiene*

$$(t_1 \wedge \neg t_3) \equiv (t_1 \wedge \neg t_3) \wedge (t_2 \vee \neg t_2)$$

aplicamos de nuevo las leyes distributivas

$$(t_1 \wedge \neg t_3) \equiv (t_1 \wedge \neg t_3) \wedge (t_2 \vee \neg t_2) \equiv (t_1 \wedge t_2 \wedge t_3) \vee (t_1 \wedge \neg t_2 \wedge \neg t_3)$$

sustituyendo ahora ambas expresiones en $(t_1 \wedge t_2)$ *y en* $(t_1 \wedge \neg t_3)$*, tenemos que*

$$(t_1 \wedge t_2 \wedge t_3) \vee (t_1 \wedge t_2 \wedge \neg t_3) \vee (t_1 \wedge t_2 \wedge t_3) \vee (t_1 \wedge \neg t_2 \wedge \neg t_3)$$

Aplicando las leyes de idempotencia, es decir, $A \equiv A \wedge A$ *y* $A \equiv A \vee A$*, la* forma normal disyuntiva *de la consulta* c*, denotada por* c_{fnd}*, resulta ser*

$$c_{fnd} = c_1 \vee c_2 \vee c_3$$

[2]una fórmula F se dice que está en *forma normal disyuntiva* si y sólo si es de la forma $F = F_1 \vee F_2 \vee \cdots \vee F_n, n \in \mathbb{N}$, donde cada F_p, siendo $p \leq n$, es una conjunción de operandos.

donde $c_1 = (t_1 \wedge t_2 \wedge t_3)$*,* $c_2 = (t_1 \wedge t_2 \wedge \neg t_3)$ *y* $c_3 = (t_1 \wedge \neg t_2 \wedge \neg t_3)$*. Se observa como cada uno de ellos poseen la misma cantidad de términos. Si extendemos de un modo natural la disyunción de booleanos a la disyunción de vectores tendremos que*

$$\overrightarrow{c_{fnd}} = \overrightarrow{c_1} \vee \overrightarrow{c_2} \vee \overrightarrow{c_3}$$

con $\overrightarrow{c_1} = (1, 1, 1)$*, ya que el documento posee los tres términos;* $\overrightarrow{c_2} = (1, 1, 0)$ *debido a que no posee el último y* $\overrightarrow{c_3} = (1, 0, 0)$ *ya que sólo posee el primero.*

■

Técnicamente la forma normal disyuntiva simplifica la verificación de una fórmula, reduciéndola a la de alguno de sus términos. En este sentido, cada uno de sus componentes es a su vez un vector binario de pesos asociados con la tupla en cuestión. A cada uno de estos vectores binarios se les denominan *componentes conjuntivos de la formal normal disyuntiva*.

6.2.1.2 | Función de comparación y ordenación

En este modelo, la función de comparación se basa en criterios de inclusión/exclusión de términos [20], lo que provoca que su resultado sea binario, es decir, se considera que un documento es relevante a una consulta cuando su valor es 1. De lo contrario, el documento no tiene ninguna relevancia y el valor de la función será 0, lo que significa que no existen gradaciones en este modelo. Retomamos ahora los trabajos realizados en [20, 188] a efectos descriptivos.

Definición 6.4 *Sean* $\{t_i\}_{i \in I}$ *la colección de términos índice,* $\mathcal{D} = \{d_j\}_{j \in J}$ *la colección documental, y* $c \in \mathcal{Q}$ *una consulta cualquiera, respectivamente. Sea* $c_{fnd} = c_1 \vee c_2 \vee \cdots \vee c_n, n \in \mathbb{N}$ *la forma normal disyuntiva de* c*, y* $\overrightarrow{c_{fnd}} = \overrightarrow{c_1} \vee \overrightarrow{c_2} \vee \cdots \vee \overrightarrow{c_n}$ *su vector asociado. La similitud entre el documento* d_j *y la consulta* c *se define como*

$$sim(d_j, c) := \begin{cases} 1, \text{ si } \exists\ \overrightarrow{c_p}, p \in \mathbb{N} \text{ tal que } \overrightarrow{c_p} \in \overrightarrow{c_{fnd}} \text{ y } \forall\ t_i, i \in I,\ g_{t_i}(\overrightarrow{d_j}) = g_{t_i}(\overrightarrow{c_p}) \\ 0, \text{ en otro caso} \end{cases} \tag{6.4}$$

Si $sim(d_j, c) = 1$*, el modelo booleano predice que el documento* d_j *es relevante para la consulta* c*. De otra forma, la predicción es que el documento es irrelevante.*

■

Así, el conjunto de documentos recuperado estará formado por aquéllos que, aplicando la consulta deseada y una vez evaluada la expresión booleana, obtengan 1 como resultado de la función de comparación.

Ejemplo 6.2 *Siguiendo con el Ejemplo 6.1, vamos a retomar el valor de* $\overrightarrow{c_{fnd}} = (1,1,1) \vee (1,1,0) \vee (1,0,0)$. *Para que la similitud entre un documento* d *y la consulta* c *tenga el valor 1, uno de los componentes de* $\overrightarrow{c_{fnd}}$ *debe ser equivalente al vector asociado a* d.

Así, si suponemos que el vector asociado al documento es $\overrightarrow{d} = (0,1,0)$*; se observa que* $sim(c,d) = 0$*, por lo que el documento no es relevante. En efecto, no existe ningún componente de* $\overrightarrow{c_{fnd}}$ *que tenga su representación igual a la de* $\overrightarrow{d}$*. Sin embargo, si el vector asociado al documento fuera* $\overrightarrow{d} = (1,1,0)$*, la similitud entre ambos sería de 1, por lo que en este caso sí sería relevante.*

■

La ventaja del modelo booleano es su simplicidad. Una desventaja fundamental reside en la imposibilidad de facilitar una ordenación de los documentos en función de un valor de relevancia respecto a la consulta. Los documentos son o bien relevantes o bien irrelevantes, pero no existe la posibilidad de indicar que un documento es más relevante que otro. Con el objeto de paliar esta carencia, se han desarrollado nuevas variantes del modelo mediante la asignación de pesos a los operadores booleanos [85], que no detallamos aquí.

6.2.2 | Modelo vectorial

Es seguramente el más popular en el ámbito de la RI [271]. Al igual que en el booleano, representa las consultas y documentos mediante vectores de pesos de términos. Sin embargo, aquí se propone estimar dichos términos en base a la importancia de cada uno de ellos en el documento. Es lo que se conoce como *ponderación del término*. Desde un punto de vista geométrico, si ambos vectores están próximos, se puede asumir que el documento es similar a la consulta. En otras palabras, el documento es posiblemente relevante.

6.2.2.1 | Representación de textos

La función de representación de los documentos se construye asociando un peso positivo no binario a cada uno de los términos índice empleados en la colección documental. Es decir, en este caso el peso asociado a un término índice $t_i, i \in I$ y a un documento $d_j,\ j \in J$ toma valores entre 0 y 1. Esto es

$$W(t_i, d_j) \in [0,1] \tag{6.5}$$

Los pesos asociados a los términos de $d_j \in \mathcal{D}$ se calculan identificando aquéllos que aparecen con frecuencias altas en algunos de los documentos individuales y, a la vez, que se hayan observado en contadas ocasiones en la colección completa. Estos términos serán los que tendrán mayor capacidad de discriminación en el modelo. Así, el peso final viene

dado en función de dos variables: la primera hace referencia al intervalo de variación del término t_i en el documento d_j, más conocido como *frecuencia de aparición del término*, representado por $\text{FT}(t_i, d_j)$, y la segunda al valor de discriminación de t_i en la colección $\mathcal{D}$, conocida como la *frecuencia documental inversa* y denotado por $\text{FDI}(t_i)$ [191, 271]. Dicho peso se representa gracias a la expresión

$$W(t_i, d_j) = \text{FDI}(t_i) \cdot \text{FT}(t_i, d_j) \tag{6.6}$$

con $\text{FDI}(t_i)$ dada por

$$\text{FDI}(t_i) = \log(\frac{|J|}{n(t_i)}) + 1 \tag{6.7}$$

donde $n(t_i)$ es el número de documentos en los que se menciona al término t_i. De este modo, el valor $\text{FDI}(t_i)$ decrece conforme $n(t_i)$ crece, variando desde 1 hasta $\log(|J|) + 1$. Por tanto, cuantas menos veces aparezca el término en la colección, más alto será su $\text{FDI}(t_i)$, describiendo una forma de estimar el impacto global del término en toda la colección. El hecho de introducir un logaritmo se justifica para suavizar en los cálculos el crecimiento del tamaño de la colección. Las consultas se representan de forma análoga.

6.2.2.2 | Función de comparación y de ordenación

Existen diferentes funciones para medir la similitud entre documentos y consultas. Todas están basadas en considerar ambos como puntos en un espacio n-dimensional. Por lo tanto, sean $d \in \mathcal{D}$, $c \in \mathcal{Q}$ y $\{t_k\}_{k=1}^n$ un documento, una consulta cualesquiera y el conjunto de los términos índice respectivamente. Entre las funciones más populares, citaremos las siguientes:

- *Producto escalar*: Se trata en definitiva de calcular la intersección de los términos coincidentes en la consulta y en el documento. Esto es, multiplicamos escalarmente ambas representaciones vectoriales

$$sim_{\text{escalar}}(d, c) := \overrightarrow{c} \bullet \overrightarrow{d} := \sum_{k=1}^{n} W(t_k, c) \cdot W(t_k, d) \tag{6.8}$$

  Dado que el producto escalar de dos vectores es mayor cuanto mayor es la proyección del primero sobre el segundo, y ello a su vez se corresponde con su proximidad sobre el plano euclídeo, parece razonable considerarlo como una función de similitud.

  Ejemplo 6.3 *Supongamos que tenemos una consulta c y un documento d, cuyos vectores asociados son los siguientes*

$$\overrightarrow{c} = (1, 0, 1, 0, 1, 0)$$

$$\overrightarrow{d} = (1, 0, 1, 1, 0, 0)$$

El producto escalar se calcula en función de la Tabla 6.1, por lo que

$\overrightarrow{c}$	1	0	1	0	1	0
$\overrightarrow{d}$	1	0	1	1	0	0
$W(t_k, c) \cdot W(t_k, d)$	1 · 1 =1	0 · 0 =0	1 · 1 =1	0 · 1 =0	1 · 0 =0	0 · 0 =0

Tabla 6.1: Cálculos para la similitud usando el producto escalar

$$sim_{escalar}(d, c) = \overrightarrow{c} \bullet \overrightarrow{d} = \sum_{k=1}^{n} W(t_k, c) \cdot W(t_k, d) = 2$$

■

- *Medida del coseno*: La similitud entre una consulta c y un documento d se obtiene estableciendo la correlación entre los vectores $\overrightarrow{c}$ y $\overrightarrow{d}$. Dicha correlación puede ser estimada calculando el coseno del ángulo que forman ambos vectores representados en el espacio n-dimensional, a partir de la definición de producto escalar, tal como se observa en la Fig. 6.3. Cuanto más paralelo sea el vector del documento al de la consulta, más relevante se considerará. Este cálculo se realiza aplicando la formulación [271]

$$sim_{\cos}(d, c) := \frac{\overrightarrow{c} \bullet \overrightarrow{d}}{||\overrightarrow{c}|| \cdot ||\overrightarrow{d}||} := \frac{\sum_{k=1}^{n} W(t_k, c) \cdot W(t_k, d)}{\sqrt{\sum_{k=1}^{n} W(t_k, c)^2} \cdot \sqrt{\sum_{k=1}^{n} W(t_k, d)^2}} \tag{6.9}$$

donde $||\overrightarrow{c}||$ y $||\overrightarrow{d}||$ son las normas de los vectores representando la consulta y

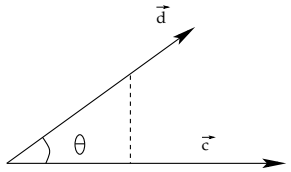


Figura 6.3: El coseno de θ adoptado como similitud $sim_{\cos}(d, c)$

el documento. Observar que el valor de $||\overrightarrow{c}||$ no afectará a la ordenación de los documentos relevantes debido a que es el mismo para todos ellos, algo que no ocurre con $||\overrightarrow{d}||$.

Ejemplo 6.4 *Siguiendo con los mismos vectores c y d del Ejemplo 6.3, la medida del coseno se calcula siguiendo la Tabla 6.2, por lo que*

$\overrightarrow{c}$	1	0	1	0	1	0
$\overrightarrow{d}$	1	0	1	1	0	0
$W(t_k,c)\cdot W(t_k,d)$	1 · 1 =1	0 · 0 =0	1 · 1 =1	0 · 1 =0	1 · 0 =0	0 · 0 =0
$W(t_k,c)^2$	1 · 1 =1	0 · 0 =0	1 · 1 =1	0 · 0 =0	1 · 1 =1	0 · 0 =0
$W(t_k,d)^2$	1 · 1 =1	0 · 0 =0	1 · 1 =1	1 · 1 =1	0 · 0 =0	0 · 0 =0

Tabla 6.2: Cálculos para la similitud usando la medida del coseno

$$sim_{cos}(d,c) = \frac{\sum_{k=1}^{n} W(t_k,c)\cdot W(t_k,d)}{\sqrt{\sum_{k=1}^{n} W(t_k,c)^2}\cdot\sqrt{\sum_{k=1}^{n} W(t_k,d)^2}} = \frac{2}{\sqrt{3}\cdot\sqrt{3}} = \frac{2}{3}$$

■

Después de observar el modo en el que se calculan el producto escalar y el coseno, se puede decir que estas medidas de similitud favorecen a aquellos documentos de mayor extensión. Esto se debe a que es más probable que posean una mayor cantidad de términos considerados de interés, por lo que al realizar el sumatorio éstos resultarán en un valor más alto.

- *Índice Jaccard*: Este índice [140] está basado en la asociación entre dos términos, calculando el coeficiente de intersección de los dos conjuntos respecto a su unión. En este sentido, resulta útil para estudiar la similitud entre objetos constituidos de atributos binarios, es decir, cuando los vectores que representan a c y a d, posean los valores 0 ó 1. De este modo, cuando la intersección de los vectores sea nula, el índice valdrá 0, y cuando ambos sean idénticos, será igual a 1. Formalmente

$$sim_{Jac}(d,c) := \frac{|c\cap d|}{|c\cup d|} := \frac{M_{11}}{M_{01}+M_{10}+M_{00}} := \tag{6.10}$$

$$\frac{\sum_{k=1}^{n} W(t_k,c)\cdot W(t_k,d)}{\sum_{k=1}^{n}[|1-W(t_k,c)|\cdot W(t_k,d)] + [W(t_k,c)\cdot|1-W(t_k,d)|] + [|1-W(t_k,c)|\cdot|1-W(t_k,d)|]}$$

donde:

- $M_{11} = W(t_k,c)\cdot W(t_k,d)$ representa el número total de términos índice coincidentes en los vectores $\overrightarrow{c}$ y $\overrightarrow{d}$, ambos con valor 1.
- $M_{01} = |1-W(t_k,c)|\cdot W(t_k,d)$ representa el número total de términos índice cuyo peso en el primer vector es 0, y en el segundo es 1.
- $M_{10} = W(t_k,c)\cdot|1-W(t_k,d)|$ representa el número total de términos índice cuyo peso en el primer vector es 1 y en el segundo es 0.

- $M_{00} = |1 - W(t_k, c)| \cdot |1 - W(t_k, d)|$ representa el número total de términos índice cuyo peso en ambos vectores es 0.

De este modo, la intersección de la consulta y del documento quedará representada por M_{11} mientras que la unión se hará mediante la suma de todo lo no común, es decir, $M_{01} + M_{10} + M_{00}$.

Ejemplo 6.5 *Siguiendo con los mismos vectores c y d del Ejemplo 6.3, el índice Jaccard se calcula en función de la Tabla 6.3*

$\overrightarrow{c}$ $\overrightarrow{d}$	1 1	0 0	1 1	0 1	1 0	0 0
M_{11}	1 · 1=1	0 · 0=0	1 · 1=1	0 · 1=0	1 · 0=0	0 · 0=0
M_{01}	|1-1| · 1=0	|1-0| · 0=0	|1-1| · 1=0	|1-0| · 1=1	|1-1| · 0=0	|1-0| · 0=0
M_{10}	1 · |1-1|=0	0 · |1-0|=0	1 · |1-1|=0	0 · |1-1|=0	1 ·|1-0|=1	0 ·|1-0|=0
M_{00}	0 · 0=0	1 · 1=1	0 · 0=0	1 · 0=0	0 · 1=0	1 · 1=1
Σ	0+0+0=0	0+0+1=1	0+0+0=0	1+0+0=1	0+1+0=1	0+0+1=1

Tabla 6.3: Cálculos para la similitud usando el índice Jaccard

donde $\Sigma = M_{01} + M_{10} + M_{00}$, por lo que

$$sim_{Jac}(d, c) = \frac{\sum_{k=1}^{n} W(t_k, c) \cdot W(t_k, d)}{\sum_{k=1}^{n} [|1 - W(t_k, c)| \cdot W(t_k, d)] + [W(t_k, c) \cdot |1 - W(t_k, d)|] + [|1 - W(t_k, c)| \cdot |1 - W(t_k, d)|]} = \frac{1}{2}$$

■

- *Índice de Tanimoto*: Este índice [303] es una extensión del de Jaccard, que le permite ser aplicado sobre valores no binarios. Tiene propiedades intermedias entre la medida del coseno y la distancia euclídea, que detallaremos más adelante. Se calcula mediante la fórmula

$$sim_{Tan}(d, c) := \frac{\overrightarrow{c} \bullet \overrightarrow{d}}{||\overrightarrow{c}||^2 + ||\overrightarrow{d}||^2 - (\overrightarrow{c} \bullet \overrightarrow{d})} := \frac{\sum_{k=1}^{n} W(t_k, c) \cdot W(t_k, d)}{\sum_{k=1}^{n} [W(t_k, c)^2 + W(t_k, d)^2 - W(t_k, c) \cdot W(t_k, d)]} \tag{6.11}$$

Ejemplo 6.6 *Siguiendo con el Ejemplo 6.3, el índice Tanimoto se calcula en función de la Tabla 6.4, dando lugar a*

$$sim_{Tan}(d, c) = \frac{\sum_{k=1}^{n} W(t_k, c) \cdot W(t_k, d)}{\sum_{k=1}^{n} [W(t_k, c)^2 + W(t_k, d)^2 - W(t_k, c) \cdot W(t_k, d)]} = \frac{2}{4} = \frac{1}{2}$$

$\overrightarrow{c}$	1	0	1	0	1	0
$\overrightarrow{d}$	1	0	1	1	0	0
$W(t_k,c)\cdot W(t_k,d)$	1 · 1=1	0 · 0=0	1 · 1=1	0 · 1=0	1 · 0=0	0 · 0=0
$W(t_k,c)^2$	1 · 1=1	0 · 0=0	1 · 1=1	0 · 0=0	1 · 1=1	0 · 0=0
$W(t_k,d)^2$	1 · 1=1	0 · 0=0	1 · 1=1	1 · 1=1	0 · 0=0	0 · 0=0

Tabla 6.4: Cálculos para la similitud usando el índice Tanimoto

■

- *Índice Dice*: Calcula [84] un ratio de la intersección de dos conjuntos y del número total de entradas distintas de cero. Al igual que el anterior, es utilizado para datos cualitativos de presencia/ausencia. Está diseñado para ser igual a 1 en casos de similitud completa, e igual a 0 en el caso de no poseer ningún valor en común.

$$sim_{Dice}(d,c) := \frac{2\cdot|c\cap d|}{|c|+|d|} := \frac{2\cdot\sum_{k=1}^{n} W(t_k,c)\cdot W(t_k,d)}{\sum_{k=1}^{n}[W(t_k,c)+W(t_k,d)]} \tag{6.12}$$

Ejemplo 6.7 *Siguiendo con el Ejemplo 6.3, el índice Dice se calcula utilizando los valores intermedios de la Tabla 6.5, por lo que*

$\overrightarrow{c}$	1	0	1	0	1	0
$\overrightarrow{d}$	1	0	1	1	0	0
$W(t_k,c)\cdot W(t_k,d)$	1 · 1=1	0 · 0 =0	1 · 1 =1	0 · 1 =0	1 · 0 =0	0 · 0 =0
$W(t_k,c)+W(t_k,d)$	1 + 1 =2	0 + 0 =0	1 + 1 =2	0 + 1 =1	1 + 0 =1	0 + 0 =0

Tabla 6.5: Cálculos para la similitud usando el índice Dice

$$sim_{Dice}(d,c) = \frac{2\cdot\sum_{k=1}^{n} W(t_k,c)\cdot W(t_k,d)}{\sum_{k=1}^{n}[W(t_k,c)+W(t_k,d)]} = \frac{4}{6} = \frac{2}{3}$$

■

Existe una relación entre el índice Dice y el Jaccard:

$$sim_{Dice}(d,c) := \frac{2\cdot sim_{Jac}(d,c)}{(1+sim_{Jac}(d,c))} \tag{6.13}$$

- *Distancia euclídea*: Un concepto muy relacionado con el de similitud es el de distancia, que trata de expresar la proximidad o lejanía entre dos objetos. En este sentido, asumimos que dos vectores distantes son aquéllos que poseen entre ellos un escaso valor de similitud. Considerando vectores n-dimensionales asociados al documento y a la consulta, la distancia euclídea entre ambos vendrá dada por

$$sim_{euclidea}(d,c) := \sqrt{\sum_{k=1}^{n}[W(t_k,d) - W(t_k,c)]^2} \tag{6.14}$$

Ejemplo 6.8 *Retomando los vectores asociados c y d del Ejemplo 6.3, la distancia euclídea se calcula usando los valores intermedios de la Tabla 6.6, por lo que*

$\overrightarrow{c}$	1	0	1	0	1	0
$\overrightarrow{d}$	1	0	1	1	0	0
$W(t_k,c) - W(t_k,d)$	1 - 1=0	0 - 0=0	1 - 1=0	0 - 1=-1	1 - 0=1	0 - 0=0
$[W(t_k,c) - W(t_k,d)]^2$	0	0	0	1	1	0

Tabla 6.6: Cálculos para la similitud usando la distancia euclídea

$$sim_{euclidea}(d,c) = \sqrt{\sum_{k=1}^{n}[W(t_k,d) - W(t_k,c)]^2} = \sqrt{2}$$

■

A diferencia del *modelo booleano*, el vectorial no se limita a comprobar si los términos especificados en la consulta están o no presentes en el documento, sino que va un paso más allá. Su principal ventaja es que permite ordenar los resultados en base a su relevancia. Sin embargo, su principal inconveniente es que no incorpora la noción de *correlación*[3] entre términos. En efecto, la presencia de un término «*a*» en un texto no necesariamente provoca la presencia de un término «*b*» en el mismo, pero puede aumentar la probabilidad de que ocurra. En este sentido, el modelo vectorial considera que todos los términos son independientes unos de otros.

6.2.3 | Modelo probabilístico

Definido por Robertson y Jones [253], se fundamenta en la idea de que dada una consulta, existe exactamente un conjunto de documentos, y no otro, que satisface la respuesta a la misma y que se conoce como *conjunto de respuesta ideal* [252].

[3] ocurre cuando existen relaciones entre los elementos, es decir, si los cambios en uno influyen también en el otro.

Si tuviéramos la descripción de ese conjunto no tendríamos problemas para recuperar los documentos relevantes. Luego podemos pensar que el proceso de generación de consultas es el de la especificación de las propiedades de dicho conjunto ideal. Pero el problema es que tampoco conocemos cuáles son estas propiedades exactamente. Todo lo que sabemos es que existen términos cuya semántica podría utilizarse para caracterizarlas [188]. Por este motivo, en un principio es necesario realizar un esfuerzo por aproximar estas propiedades, ya que se desconocen totalmente en el momento de la consulta. Con el objeto de generar una descripción probabilística preliminar, echaremos mano de un conjunto de hipótesis iniciales, que servirán para recuperar una primera serie de documentos, que el usuario analizará para decidir cuáles son relevantes y cuáles no. Luego, el sistema utilizará esta información para refinar la descripción del conjunto de respuesta ideal. El proceso se repetirá hasta que la descripción se acerque a la real.

6.2.3.1 | Representación de textos

La función de representación asocia un peso binario a cada término índice del documento considerado. Este será 1 si el término aparece al menos una vez en el documento y 0 en caso contrario. En consecuencia, el peso inicial asociado a un término índice $t_i, i \in I$ en un documento de la colección documental $d_j \in \mathcal{D} = \{d_j\}_{j \in J}$ tendrá los siguiente valores

$$W(t_i, d_j) \in \{0, 1\} \tag{6.15}$$

Las consultas se representarán de manera análoga.

6.2.3.2 | Función de comparación y de ordenación

Para estimar la similitud entre un documento y una consulta, el modelo mide la correspondencia entre las probabilidades de que dicho documento sea relevante o no para esa consulta, minimizando la probabilidad de un juicio erróneo [104, 314]. En esencia, los documentos son devueltos ordenados en orden decreciente de acuerdo a su probabilidad de relevancia respecto a la consulta [20, 248].

Definición 6.5 *Sean $\{t_i\}_{i \in I}$ una colección de términos índice, $\mathcal{D} = \{d_j\}_{j \in J}$ la colección documental, $c \in \mathcal{Q}$ una consulta, y $\overrightarrow{d_j}$ el vector asociado a un documento $d_j \in \mathcal{D}$. Definimos la similitud de una consulta c con respecto a un documento d_j como la relación siguiente*

$$sim(d_j, c) := \frac{P(\text{rel}(c, \mathcal{D}) \,|\overrightarrow{d_j})}{P(\text{nrel}(c, \mathcal{D}) \,|\overrightarrow{d_j})} \tag{6.16}$$

donde denotamos por $P(\text{rel}(c, \mathcal{D}) \,|\overrightarrow{d_j})$ la probabilidad de relevancia de $\text{rel}(c, \mathcal{D})$ dado $\overrightarrow{d_j}$, y por $P(\text{nrel}(c, \mathcal{D}) \,|\overrightarrow{d_j})$ la probabilidad de no relevancia de $\text{nrel}(c, \mathcal{D})$ dado $\overrightarrow{d_j}$.

■

Aplicando el *Teorema de Bayes* [26, 144, 188] y tras una serie de simplificaciones, el valor de la similitud, expresada en la Definición 6.5, es el siguiente

$$sim(d_j, c) = \frac{\frac{P(\overrightarrow{d_j}|\textit{rel}(c,\mathcal{D}))\cdot P(\textit{rel}(c,\mathcal{D}))}{P(\overrightarrow{d_j})}}{\frac{P(\overrightarrow{d_j}|\textit{nrel}(c,\mathcal{D}))\cdot P(\textit{nrel}(c,\mathcal{D}))}{P(\overrightarrow{d_j})}} = \frac{P(\textit{rel}(c,\mathcal{D}))}{P(\textit{nrel}(c,\mathcal{D}))} \cdot \frac{P(\overrightarrow{d_j}|\textit{rel}(c,\mathcal{D}))}{P(\overrightarrow{d_j}|\textit{nrel}(c,\mathcal{D}))} \tag{6.17}$$

donde $P(\overrightarrow{d_j}|\textit{rel}(c,\mathcal{D}))$ representa la probabilidad de elegir a d_j conocido el conjunto $\textit{rel}(c,\mathcal{D})$. Respectivamente, $P(\overrightarrow{d_j}|\textit{nrel}(c,\mathcal{D}))$ representa la probabilidad análoga de elegir a d_j conocido el conjunto $\textit{nrel}(c,\mathcal{D})$. Podemos expresar el valor de similitud en función de cada uno de los términos que componen $\overrightarrow{d_j}$ si asumimos su inter-independencia y aplicamos la *hipótesis de independencia condicional de Bayes simplista*[4], para obtener

$$sim(d_j, c) = \frac{P(\textit{rel}(c,\mathcal{D}))}{P(\textit{nrel}(c,\mathcal{D}))} \cdot \frac{\prod_{i}^{|I|} P(W(t_i, d_j)|\textit{rel}(c,\mathcal{D}))}{\prod_{i}^{|I|} P(W(t_i, d_j)|\textit{nrel}(c,\mathcal{D}))} \tag{6.18}$$

con $W(t_i, d_j)$ el i-ésimo componente de $\overrightarrow{d_j}$. Se trata en definitiva del peso asociado al término t_i, que indica su presencia o ausencia en el documento d_j.

Dado que $\textit{rel}(c,\mathcal{D}))$ y $\textit{nrel}(c,\mathcal{D}))$ son constantes para una consulta y una colección documental determinada, podemos simplificar la expresión anterior de tal forma que, si bien los valores obtenidos difieren, la ordenación se mantiene. Además, gracias a la propiedad conmutativa podemos igualmente agrupar aquellos operandos correspondientes a términos t_i que aparecen en el documento d_j, esto es, donde $W(t_i, d_j) = 1$. Del mismo modo, también se puede agrupar aquéllos correspondientes a términos t_i que no aparecen en el documento d_j, esto es, donde $W(t_i, d_j) = 0$.

$$sim(d, c) \sim \prod_{\substack{t_i \in d_j \\ t_i \in c}} \frac{P(W(t_i, d_j) = 1|\textit{rel}(c,\mathcal{D}))}{P(W(t_i, d_j) = 1|\textit{nrel}(c,\mathcal{D}))} \cdot \prod_{\substack{t_i \notin d_j \\ t_i \in c}} \frac{P(W(t_i, d_j) = 0|\textit{rel}(c,\mathcal{D}))}{P(W(t_i, d_j) = 0|\textit{nrel}(c,\mathcal{D}))} \tag{6.19}$$

con los siguientes valores:

- $p_i := P(W(t_i, d_j) = 1|\textit{rel}(c,\mathcal{D}))$ es la probabilidad de que un término t_i aparezca en un documento d_j relevante para una consulta c.

[4] la denominada *hipótesis de independencia condicional de Bayes simplista* [314] indica que dado un suceso A compuesto por varios sucesos K_i independientes entre sí, $A = \{K_i \in A\}$, ocurre que $P(A|B) = \prod_{K_i \in A} P(K_i|B)$.

- $u_i := P(W(t_i, d_j) = 1|\textit{nrel}(c, \mathcal{D}))$ es la probabilidad de que un término t_i aparezca en un documento d_j no relevante para una consulta c.
- $1 - p_i := P(W(t_i, d_j) = 0|\textit{rel}(c, \mathcal{D}))$ es la probabilidad de que un término t_i no aparezca en un documento d_j relevante para una consulta c.
- $1 - u_i := P(W(t_i, d_j) = 0|\textit{nrel}(c, \mathcal{D}))$ es la probabilidad de que un término t_i no aparezca en un documento d_j no relevante para una consulta c.

Así, haciendo las correspondientes sustituciones, obtenemos

$$sim(d, c) \sim \prod_{\substack{t_i \in d_j \\ t_i \in c}} \frac{p_i}{u_i} \cdot \prod_{\substack{t_i \notin d_j \\ t_i \in c}} \frac{1 - p_i}{1 - u_i} \tag{6.20}$$

Por otra parte, si introducimos en una expresión un valor distinto de cero, multiplicando y dividiendo simultáneamente, el valor de la expresión no varía. De igual modo, reordenando y reagrupando los factores, mediante las propiedades conmutativa y asociativa, el valor de la expresión tampoco varía, obteniendo

$$\begin{aligned} sim(d, c) \sim \prod_{\substack{t_i \in d_j \\ t_i \in c}} \frac{p_i}{u_i} \cdot \prod_{\substack{t_i \notin d_j \\ t_i \in c}} \frac{1 - p_i}{1 - u_i} \cdot \Big(\prod_{\substack{t_i \in d_j \\ t_i \in c}} \frac{1 - p_i}{1 - u_i} \cdot \prod_{\substack{t_i \in d_j \\ t_i \in c}} \frac{1 - u_i}{1 - p_i} \Big) = \\ = \prod_{\substack{t_i \in d_j \\ t_i \in c}} \frac{p_i \cdot (1 - u_i)}{u_i \cdot (1 - p_i)} \cdot \prod_{t_i \in c} \frac{1 - p_i}{1 - u_i} \end{aligned} \tag{6.21}$$

Si aplicamos logaritmos[5], con el fin de emplear el valor resultante en la ordenación de los documento devueltos, ya que éstos aminoran suavemente su valor al mismo tiempo que incrementan su posición en dicha ordenación, el resultado toma la forma

$$sim(d, c) \sim \log \prod_{\substack{t_i \in d_j \\ t_i \in c}} \frac{p_i \cdot (1 - u_i)}{u_i \cdot (1 - p_i)} = \sum_{\substack{t_i \in d_j \\ t_i \in c}} \log \frac{p_i \cdot (1 - u_i)}{u_i \cdot (1 - p_i)} \tag{6.22}$$

Por lo tanto, si un término de la consulta tiene la misma probabilidad de aparecer en un documento relevante que la de aparecer en un no relevante ($p_i = u_i$), el cociente será 1 y su logaritmo será 0. Por otra parte, si la probabilidad de aparecer en un documento relevante es mayor que la de aparecer en un no relevante, el numerador será mayor que el denominador ($p_i > u_i$), el cociente será mayor que 1, y su logaritmo mayor que 0. Por

[5] se trata de una función monótona por lo que la ordenación se mantiene.

el contrario, si la probabilidad de aparecer en un documento relevante es menor que la de aparecer en un no relevante ($p_i < u_i$), el numerador será menor que el denominador, el cociente será menor que 1, y su logaritmo menor que 0. La cuestión, ahora, es cómo estimar los parámetros p_i y u_i para así poder calcular la proporción

$$\log \frac{p_i \cdot (1 - u_i)}{u_i \cdot (1 - p_i)}$$

En este punto, se hace necesario hacer un cierto número de suposiciones para avanzar en nuestro objetivo de aproximar el conjunto ideal, aunque en el caso de disponer de información sobre la relevancia de algunos textos [20], dicha proporción puede estimarse fácilmente [104, 287]. Éstas suposiciones pueden ser las siguientes [188, 324]:

- El conjunto de respuesta ideal es el que maximiza la probabilidad de relevancia para la consulta. Por lo que se asume que sus elementos serán relevantes, mientras que el resto no.

- El hecho de juzgar un documento dado como relevante o no, no aporta información alguna sobre el carácter de otros documentos, lo que denominamos *hipótesis de independencia* [324].

Por ello, supongamos que el sistema ha devuelto un conjunto inicial de documentos para la consulta y que el usuario ha examinado algunos, identificando cuáles son relevantes y cuáles no. Tal conjunto inicial, denotado por V, puede aproximarse (por ejemplo) tomando los r mejores de la ordenación resultante de las respuestas obtenidas, siendo r un umbral definido previamente. Sean entonces $V_{t_i} \subset V$ aquéllos documentos que contienen el término t_i. Lo que haremos será aproximar p_i mediante la distribución del término t_i en V. De forma similar, se puede aproximar u_i considerando que los documentos no recuperados son irrelevantes. Numéricamente, esto implica que

$$p_i \sim \frac{|V_{t_i}|}{|V|} \qquad\qquad u_i \sim \frac{n_{t_i} - |V_{t_i}|}{|J| - |V|} \tag{6.23}$$

donde n_{t_i} representa la cantidad de documentos que contienen el término t_i y $|J|$ el total de documentos de la colección documental, respectivamente. Partiendo de estas suposiciones iniciales, se pueden recuperar documentos que contienen los términos de la consulta y brindan una ordenación probabilística inicial. Luego se mejora la ordenación repitiendo este proceso recursivamente. Hay que destacar que la Ecuación 6.23 tiene algunos problemas prácticos con valores pequeños de V y V_{t_i}. Para resolverlos se pueden agregar factores de ajuste, de forma que

$$p_i \sim \frac{|V_{t_i}| + 0{,}5}{|V| + 1} \qquad\qquad u_i \sim \frac{n_{t_i} - |V_{t_i}| + \frac{n_{t_i}}{|J|}}{|J| - |V| + 1} \tag{6.24}$$

De este modo, sustituyendo dichas estimaciones en la expresión de similitud 6.22 obtenemos

$$sim(c_i, d_j) \sim \sum_{\substack{t_i \in d_j \\ t_i \in c}} \log \frac{(|V_{t_i}| + 0{,}5)/(|V| - |V_{t_i}| + 0{,}5)}{(n_{t_i} - |V_{t_i}| + 0{,}5)/(|J| - n_{t_i} - |V| - |V_{t_i}| + 0{,}5)} \quad (6.25)$$

donde cada uno de los logaritmos que componen el sumatorio se denominan *pesos de Robertson-Sparck Jones* [253].

Al igual que en el modelo vectorial, el probabilístico obtiene un conjunto resultante que proporciona una ordenación de los documentos en base a su relevancia estimable. En relación a las desventajas, cabe destacar la necesidad de realizar una separación inicial entre documentos relevantes y no relevantes, que no siempre es simple. Por otro lado, se considera la presencia o ausencia de los términos, pero no el número de veces que éstos aparecen en el documento a la hora de evaluar su relevancia. En este sentido, existen trabajos que si consideran este parámetro, tales como el modelo *Okapi* BM*25* [250, 251, 253] o el paradigma DFR [133, 232], que no abordaremos aquí.

6.3 | Modelo de RI mediante GC's

Los modelos descritos hasta el momento se centraban en representar documentos y consultas mediante un conjunto de palabras, considerados como los descriptores e índices del sistema. En cambio, en el caso de un enfoque basado en GC's, la indexación se basa en la consideración de las relaciones entre términos. En adelante como marco general de trabajo llamaremos $\mathcal{D} = \{d_i\}_{i \in I}$ a la colección documental y $\mathcal{Q} = \{c_j\}_{j \in J}$ a la de las consultas relativas a un *corpus* $\mathscr{C}$.

6.3.1 | Representación de textos

En este caso utilizaremos GCB's, un caso particular de GC's, tanto para representar la colección documental como las consultas. A nuestro conocimiento, no se han presentado ni documentado, hasta ahora, herramientas o algoritmos encargados de la generación automática de estos GCB's a partir de texto. Todos ellos parecen obtenerse de un modo manual [56, 112]. En este sentido, dicho proceso específico de generación automática constituye una de las contribuciones de esta tesis, razón por la cual no lo abordaremos en esta sección, sino que lo ilustraremos en detalle más adelante.

Ejemplo 6.9 *Un ejemplo de representación de la consulta* «je cherche une tige tétragone» (`«busco un tallo tetragonal»`) *en forma de* GCB *es el que se observa en la Fig. 6.4.*

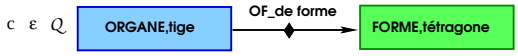


Figura 6.4: Una consulta $c \in \mathcal{Q}$ en forma de GCB de ejemplo

Del mismo modo, supongamos que disponemos de un conjunto de documentos $\mathcal{D}$ procedentes de un corpus *$\mathscr{C}$. Un ejemplo de representación de un documento $d \in \mathcal{D}$ es el que se muestra en la Fig. 6.5.*

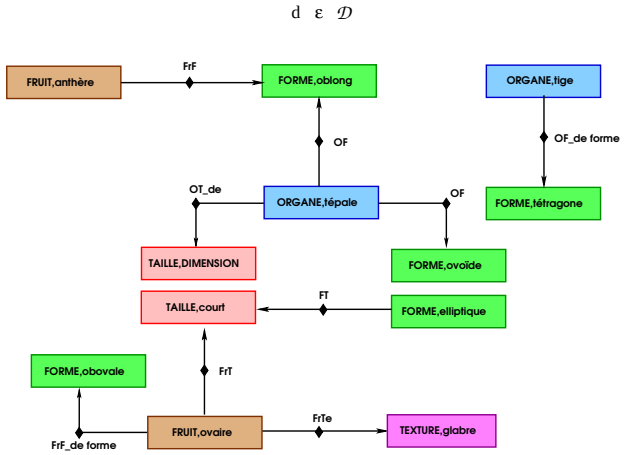


Figura 6.5: Un documento $d \in \mathcal{D}$ en forma de GCB de ejemplo

■

6.3.2 | Función de comparación y de ordenación

El modelo basado en la utilización de GCB's plantea realizar la comparación entre una consulta $c \in \mathcal{Q}$ y la colección documental $\mathcal{D}$ utilizando el concepto de *proyección*. Cada proyección de c sobre un documento $d \in \mathcal{D}$ conduce a una respuesta o, como veremos, c es deducible de la colección documental $\mathcal{D}$.

Como paso preliminar a la formalización de este proceso, estableceremos una correspondencia semántica Φ que asigne una fórmula en LPO $\Phi(\mathcal{G})$ a cada GCB $\mathcal{G}$ [295] definido sobre el soporte $\mathcal{S} = (\mathcal{T}_\mathcal{C}, \mathcal{T}_\mathcal{R}, \mathcal{I})$, donde $\Phi(\mathcal{G})$ es una fórmula positiva, conjuntiva y cerrada existencialmente. En otras palabras, Φ asigna un conjunto de fórmulas $\Phi(\mathcal{S})$ sobre un soporte $\mathcal{S}$, lo cual corresponde con una interpretación de orden parcial de $\mathcal{T}_\mathcal{R}$ y $\mathcal{T}_\mathcal{C}$. Para todo tipo t y t', tal que $t \geq t'$, se tiene la siguiente fórmula:

$$\forall\, C_1,\ ...,C_k,\ t'(C_1,\ ...,C_k) \rightarrow t(C_1,\ ...,C_k)$$

donde $k = 1$ para los tipos conceptuales, y en cualquier otro caso k es la aridad de los tipos relacionales. Esto implica que las consultas $c \in \mathcal{Q}$ y los documentos $d \in \mathcal{D}$ pueden

ser interpretados como fórmulas lógicas, y que el proceso de búsqueda se corresponde con un proceso de inferencia lógica.

Ejemplo 6.10 *Supongamos el* GCB $\mathcal{G}$ *de la Fig. 6.6. Las fórmulas en* LPO *de* $\Phi(\mathcal{G})$ *se construirán siguiendo los siguientes pasos:*

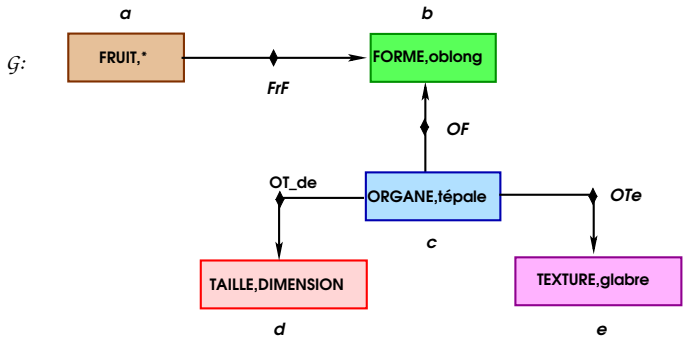


Figura 6.6: Construcción del modelo semántico $\Phi(\mathcal{G})$ a partir del GCB $\mathcal{G}$

1. *En primer lugar, asociamos a cada uno de los nodos conceptos de la figura los siguientes términos:* a= Fruit(x), b= Forme(oblong), c= Organe(tépale), d= Taille(DIMENSION) y e= Texture(glabre), *donde* x *representa a la única variable. La conjunción de las fórmulas asociadas a estos nodos conceptos es:*

   $\mathcal{C}$ = Fruit(x) $\wedge$ Forme(oblong) $\wedge$ Organe(tépale) $\wedge$ Taille(DIMENSION) $\wedge$ Texture(glabre).

2. *Luego, asociamos a cada uno de los nodos relación de la figura los siguientes átomos:* FrF(a,b) = FrF(x,oblong), OF(c,b) = OF(tépale,oblong), OT(c,d) = OT(tépale,DIMENSION) y OTe(c,e) = OTe(tépale,glabre), *donde* x *representa a la única variable. La conjunción de las fórmulas asociadas a los nodos relaciones es:*

   $\mathcal{R}$ = FrF(x,oblong) $\wedge$ OF(tépale,oblong) $\wedge$ OT(tépale,DIMENSION) $\wedge$ OTe(tépale,glabre).

3. *Finalmente,* $\Phi(\mathcal{G})$ *es el cierre existencial aplicado sobre las variables libres*[6] *de la conjunción de fórmulas asociadas a todos los nodos de* $\mathcal{C}$ *y* $\mathcal{R}$*:*

   $\Phi(\mathcal{G}) = \exists x,$ Fruit(x) $\wedge$ Forme(oblong) $\wedge$ Taille(DIMENSION) $\wedge$ Organe(tépale) $\wedge$ Texture(glabre) $\wedge$ OF(tépale,oblong) $\wedge$ FrF(x,oblong) $\wedge$ OTe(tépale,glabre) $\wedge$ OT(tépale,DIMENSION)

   Como sólo existe una variable libre x*, el cierre existencial se realizará sobre ella.*

■

[6]una variable x es libre en una fórmula si x no aparece ligada, es decir, si esa variable no tiene un radio de acción de un cuantificador.

Si la representación de conocimiento bajo forma de GCB favorece la lectura a los no familiarizados con las notaciones lógicas, la representación gráfica del razonamiento producido por el GCB del Ejemplo 6.10 es también más fácil de interpretar que una basada en fórmulas obtenidas de la manera ahora descrita.

Dicho esto, ya estamos en condiciones de razonar formalmente en base a los conocimientos representados mediante grafos en la colección documental y en las consultas.

Teorema 6.1 *(Suficiencia y completitud) Sean $d \in \mathcal{D}$ y $c \in \mathcal{Q}$ dos* GCB*'s definidos sobre el soporte $\mathcal{S}$, entonces*

$$c \succeq \text{nf}(d) \Leftrightarrow \Phi(\mathcal{S}),\ \Phi(d) \models \Phi(c)$$

donde $\models$ denota la deducción en LPO*; y* $\text{nf}(d)$ *es la* forma normal *de d, a saber, aquélla que se obtiene fusionando los nodos concepto con mismo referente individual*[7]*. En definitiva, se trata de aplicar la operación binaria de ligadura externa.*

Demostración: Ver [220].

■

Se puede demostrar que la generación de respuestas a consultas mediante GCB's en el marco descrito es un problema NP-*completo* [55]. En este sentido, el problema de la decisión[8] se puede resolver en un tiempo polinómico [56, 137], dando un sentido computacional a nuestro planteamiento.

6.3.2.1 | Transformaciones

Desde un punto de vista práctico hemos de dotar, además, a las proyecciones de la flexibilidad necesaria para la localización de respuestas cuya estructura no se corresponda exactamente con la proyección de la correspondiente pregunta. En este sentido, será necesario organizar la búsqueda de secuencias de *transformaciones* que permitan a la pregunta o a la colección documental relajar sus estructuras de forma tal que dicha proyección sea posible.

Definición 6.6 *Sean $d, d' \in \mathcal{D}$ y $c \in \mathcal{Q}$, tres* GCB*'s definidos sobre un soporte $\mathcal{S}$, y ς una correspondencia del conjunto de* GCB*'s definidos sobre $\mathcal{S}$ en él mismo, tal que $\varsigma(d) = d'$. Si $\pi \in \text{proy}(c, d')$, entonces (π, ς) es una* proyección de c en d modulo ς.

■

[7] esto es, un GCB está en forma normal si cada referente individual con un tipo conceptual aparece una única vez en él.

[8] esto es, saber si es resoluble, no o simplemente es no decidible.

Intuitivamente, la idea es la de proveer un conjunto de transformaciones que permitan determinar la pertinencia de un documento en relación a una pregunta, cuando la información contenida en ambos guarde algún tipo de relación. Formalmente consideraremos tres mecanismos de transformación aplicables a un GCB. Comenzaremos por el de *sustitución*.

Definición 6.7 *Sea* $\mathcal{G} = (\mathcal{C} \cup \mathcal{R}, \mathcal{A}, \mathcal{E})$ *un* GCB *definido sobre un soporte* $\mathcal{S} = (\mathcal{T}_\mathcal{C}, \mathcal{T}_\mathcal{R}, \mathcal{I})$. *Una* sustitución en $\mathcal{G}$ *es un par* $(t, t') \in (\mathcal{C} \times (\mathcal{T}_\mathcal{C} \times (\mathcal{I} \cup \{*\}))) \cup (\mathcal{R} \times \mathcal{T}_\mathcal{R})$. *Si se puede afirmar que un término concepto (resp. relación)* t *puede ser sustituido por uno* t'*, se dice que* (t, t') *son* términos compatibles.

■

Como acabamos de ver, una transformación por sustitución puede afectar tanto a los referentes individuales de los conceptos como a las etiquetas de las relaciones. En este sentido, esta transformación hace uso de las operaciones de restricción de concepto y de relación.

Ejemplo 6.11 *En la Fig. 6.7 se presenta la transformación del nodo concepto* [Forme, oblong] `([Forma,oblonga])` *del grafo* $\mathcal{G}$*, en el nodo concepto* [Forme,lancéolé] `([Forma,lanceolada])`*, dando lugar al grafo* $\mathcal{H}$.

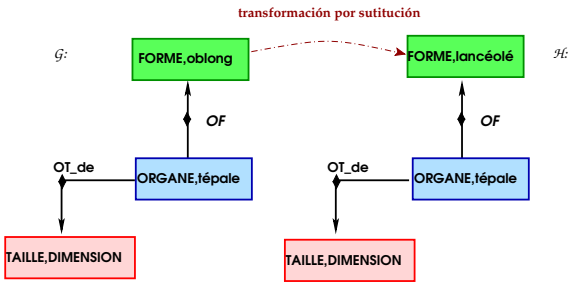


Figura 6.7: Aplicación de transformación sustitución

■

La siguiente transformación a definir, hace referencia al conjunto de operaciones de ligadura interna para producir la unión de nodos aplicables a un grafo.

Definición 6.8 *Sea* $\mathcal{G} = (\mathcal{C} \cup \mathcal{R}, \mathcal{A}, \mathcal{E})$ *un* GCB *definido sobre un soporte* $\mathcal{S} = (\mathcal{T}_\mathcal{C}, \mathcal{T}_\mathcal{R}, \mathcal{I})$. *El resultado de aplicar una* unión de los conceptos $c, c' \in \mathcal{T}_\mathcal{C}$*, tal que* $\mathcal{E}(c) = \mathcal{E}(c')$*, es el* GCB *obtenido a partir de* $\mathcal{G}$ *mediante la identificación de* c *y* c'.

■

Ejemplo 6.12 *En la Fig. 6.8 se presenta la transformación por unión de conceptos sobre el grafo $\mathcal{G}$, es decir, los nodos conceptos* [Forme, obovoïde] (`[Forma,obovoide]`) *del grafo $\mathcal{G}$, se transforman en uno único en $\mathcal{H}$, conservando las relaciones que existían.*

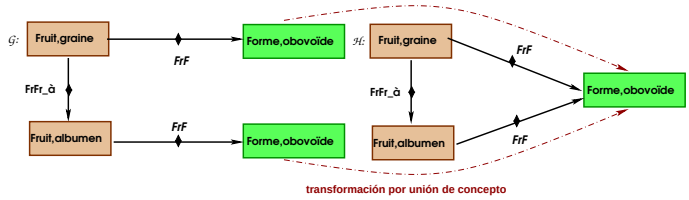


Figura 6.8: Aplicación de transformación de unión de conceptos

■

Como una unión puede cambiar sustancialmente la estructura de un GCB, se considera que provoca más distanciamiento que las sustituciones. Finalmente, la última transformación hace referencia a la agregación de nodos tanto concepto como relación.

Definición 6.9 *Sea $\mathcal{G} = (\mathcal{C} \cup \mathcal{R}, \mathcal{A}, \mathcal{E})$ un* GCB *definido sobre un soporte $\mathcal{S} = (\mathcal{T}_\mathcal{C}, \mathcal{T}_\mathcal{R}, \mathcal{I})$. El resultado de* agregar un nodo $n \in \mathcal{C} \cup \mathcal{R}$, *tal que $\mathcal{E}(n) = v$, es el nuevo* GCB *$\mathcal{G} + \mathcal{N}$, donde $\mathcal{N}$ es el grafo reducido a n. Si $n \in \mathcal{R}$, entonces es necesario especificar sus aristas vecinas.*

■

Ejemplo 6.13 *En la Fig. 6.9 se presenta la transformación por agregación de nodos sobre el grafo $\mathcal{H}$. Esto es, el nodo concepto* [Forme,oblong] (`[Forma,oblongo]`) *y el nodo relación* OF *del grafo $\mathcal{G}$ son agregados en $\mathcal{H}$, así como las aristas vecinas de* OF.

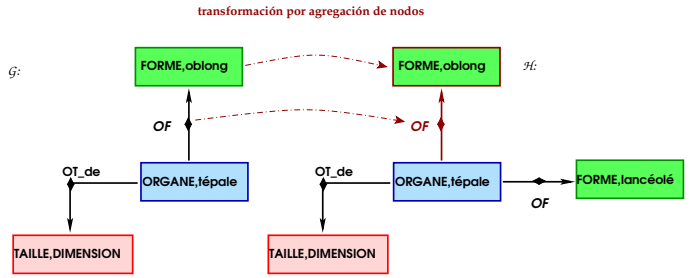


Figura 6.9: Aplicación de transformación de agregación

■

Dado que una agregación no sólo varía la estructura del GCB original, sino que además introduce un elemento externo al mismo, esta transformación se considera más compleja que una unión y, en consecuencia, también posee un impacto mayor que el de una sustitución. De este modo, es posible establecer un preorden sobre las secuencias de transformaciones a la hora de comparar el GCB asociado a una consulta c con el de un documento d, tal que los resultados se encuentren ordenados. La recuperación vendrá determinada por la secuencia de transformaciones necesarias para que exista una proyección de la consulta sobre éstos. Así, los documentos recuperados serán clasificados en base al orden seguido en las secuencias de transformaciones sobre los índices, para que exista la proyección.

Por otra parte, y en función de la necesidad o no de combinar las transformaciones definidas, se pueden considerar cuatro posibles tipos de respuestas a una pregunta dada, que introducimos de forma incremental en consideración a la complejidad de su proceso de cálculo. En este sentido, las respuestas más simples serán aquéllas cuyo contenido se refiere de forma exacta a la interrogación planteada.

6.3.2.2 | Tipos de respuestas

Para comenzar, las *respuestas exactas* son aquéllas que satisfacen plenamente a una consulta. Dicho de otro modo, al plantear ésta, todos y cada unos de los conceptos y relaciones que se encuentran en el grafo creado se pueden proyectar en su totalidad en el grafo de un documento. De esa manera, dicho documento debiera dar respuesta exacta a la consulta planteada.

Definición 6.10 *Sean $d \in \mathcal{D}$ y $c \in \mathcal{Q}$ dos* GCB*'s definidos sobre un soporte $\mathcal{S}$. Entonces d es una* respuesta exacta de *c si y sólo si* $\text{proy}(c, d) \neq \emptyset$.

■

Ejemplo 6.14 *Supongamos que realizamos una consulta $c \in \mathcal{Q}$ cuyo* GCB *asociado es $\mathcal{G}$, ilustrada en la Fig. 6.10.*

Si suponemos que nuestro documento $d \in \mathcal{D}$ es el GCB *$\mathcal{H}$ de la misma figura, observamos como se han podido proyectar todos y cada uno de los elementos de $\mathcal{G}$ en $\mathcal{H}$. Por lo tanto, el documento representado por $\mathcal{H}$ proporciona una respuesta exacta.*

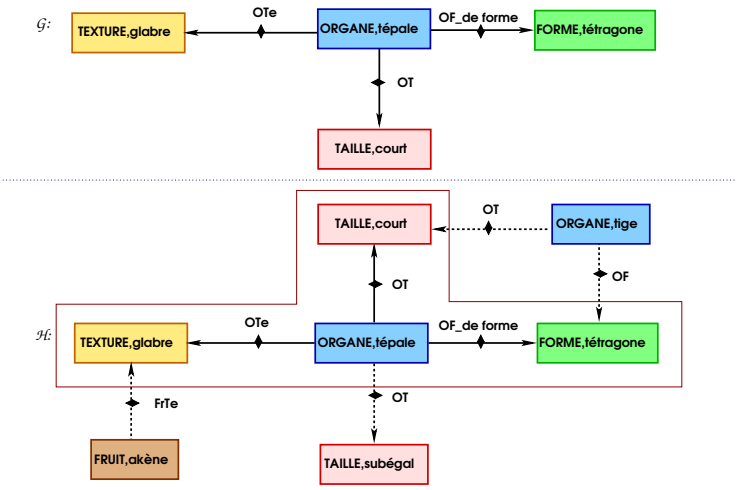


Figura 6.10: Respuesta exacta

■

A menudo la ausencia de una respuesta exacta es previsible, bien por la falta de información específica en la base de datos documental, bien por la falta de concretud de la propia pregunta. En el primer caso, hablaremos de *incompletitud documental* y en el segundo de *ambigüedad de la consulta*. Con el fin de tratar estos casos, primero tenemos que capturar formalmente la noción de respuesta no exacta y situarla en el marco ya definido para los GCB's. A este respecto, en esta tesis adoptamos la estrategia de búsqueda descrita en [112], a su vez inspirada en la implementación de la *segunda forma del Principio de incertidumbre de van Rijsbergen's* [315] propuesto en [157]:

"Sean $d \in \mathcal{D}$ y $c \in \mathcal{Q}$ dos proposiciones, una medida de incertidumbre relativa de $d \rightarrow c$ a una base de conocimiento está determinada por la transformación mínima de d en d', tal que se verifique $d' \rightarrow c$."

donde, en nuestro caso, la transformación de d en d' está basada en las operaciones de grafos, que podrían también ser usadas para transformar una consulta c. En este sentido, cabría preguntarse por qué no transformar c en c' con el fin de conseguir verificar que $d \rightarrow c'$. Con respecto a esto, se puede ver que $d' \rightarrow c$ se verifica si y sólo si $d \rightarrow c'$, donde c' se obtiene a partir de c mediante una transformación dual de una transformación de d en d'. La ventaja usualmente argumentada [112] para modificar la colección documental $\mathcal{D}$ en lugar de las preguntas $\mathcal{Q}$ es que los contenidos de la primera pueden enriquecerse mediante relevancia retroalimentada por el sistema de RI. En cualquier caso, ello permite establecer el marco formal que necesitábamos para flexibilizar el protocolo de interrogación antes introducido en los GCB's. Comenzaremos por describir el caso más simple. Se trata de las *respuestas aproximadas*.

Definición 6.11 *Sean $d \in \mathcal{D}$ y $c \in \mathcal{Q}$ dos* GCB*'s definidos sobre un soporte $\mathcal{S}$. Entonces*

d es una respuesta aproximada de *c si y sólo si existe una secuencia de sustituciones ς, tales que* proy$(c, \varsigma(d)) \neq \emptyset$.

■

Ejemplo 6.15 *Supongamos que planteamos la consulta $c \in \mathcal{Q}$ cuyo* GCB *asociado es $\mathcal{G}$, ilustrado en la Fig. 6.11.*

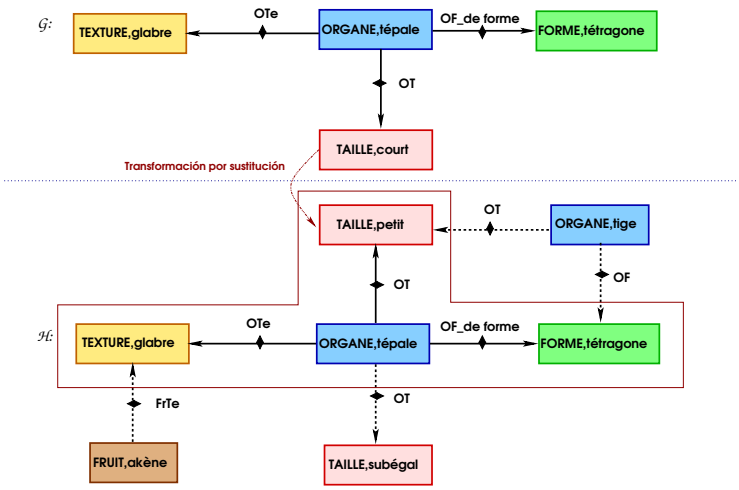


Figura 6.11: Respuesta aproximada

Si suponemos que nuestro documento $d \in \mathcal{D}$ es el GCB *$\mathcal{H}$ de la misma figura, observamos como podemos proyectar casi todos los componentes de $\mathcal{G}$ en $\mathcal{H}$, menos un nodo concepto. Se trata del concepto* [Taille,court] `([Tamaño,corto])`. *En cambio, el documento posee uno similar a éste. Es* [Taille,petit] `([Tamaño,pequeño])`. *Por lo tanto, el documento $\mathcal{H}$ proporciona una respuesta aproximada. En este sentido, simplemente será necesario realizar una sustitución del nodo* [Taille,court] `([Tamaño,corto])` *por* [Taille,petit] `([Tamaño,pequeño])`.

■

Intuitivamente, para calcular una respuesta aproximada, la estructura del GCB inicial d se ve ligeramente modificada. Dado que las respuestas exactas son un tipo particular de las aproximadas y que constituyen un fenómeno raro sin casi interés práctico, en adelante sólo hablaremos de respuestas aproximadas para referirnos a ambas categorías, las exactas y las aproximadas. Con el fin de ampliar el grado de flexibilidad asociados a las consultas, aumentaremos el umbral de las transformaciones estructurales permitidas, por ejemplo, incluyendo las uniones. Esto permite definir las *respuestas plausibles*.

Definición 6.12 *Sean $d \in \mathcal{D}$ y $c \in \mathcal{Q}$ dos* GCB*'s definidos sobre un soporte $\mathcal{S}$. Se dice que una* secuencia ς de sustituciones y uniones es aceptable *si y sólo si ς no contiene*

demasiadas uniones en relación al número de nodos en c. La proporción de uniones permitidas (μ_u) se establece por el usuario.

■

Definición 6.13 *Sean $d \in \mathcal{D}$ y $c \in \mathcal{Q}$ dos* GCB*'s definidos en un soporte $\mathcal{S}$. Se dice que d es una* respuesta plausible *a c si y sólo si existe una secuencia aceptable ς de sustituciones y uniones, tal que* $\text{proy}(c, \varsigma(d)) \neq \emptyset$.

■

Ejemplo 6.16 *Supongamos que planteamos la consulta $c \in \mathcal{Q}$ cuyo* GCB *asociado es $\mathcal{G}$, ilustrada en la Fig. 6.12. Si suponemos que nuestro documento $d \in \mathcal{D}$ es el* GCB *$\mathcal{H}$ de la misma figura, observamos como en dicho grafo incluyen todos los nodos, tanto concepto como relación, que aparecen en la consulta.*

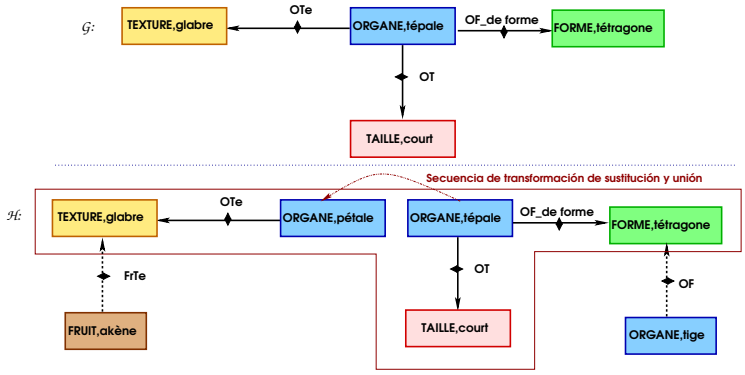


Figura 6.12: Respuesta plausible

Sin embargo, no ocurre lo mismo con las aristas. Así, [Forme,tétragone] (`[Forma, tetragonal]`) *y* [Taille,court](`[Tamaño,corto]`) *están relacionados con el nodo* [Organe,tépale] (`[Órgano,tépalo]`), *mientras que* [Texture,glabre] (`[Textura,glabro]`) *lo está con* [Organe,pétale] (`[Órgano,pétalo]`), *aunque se puede intuir que este documento puede ser interesante para el usuario que ha formulado la consulta c. En este sentido, la utilización de una secuencia de sustitución del nodo* [Organe,pétale] (`[Órgano,pétalo]`) *en* [Organe,tépale] (`[Órgano,tépalo]`) *y su posterior unión permite realizar la proyección pertinente de $\mathcal{G}$ en $\mathcal{H}$.*

■

Para completar la oferta relacionada con las consultas, incluimos finalmente las agregaciones de nodos. Aunque esto no permite cubrir totalmente el abanico de transformaciones para grafos, sí se centra en aquellas interrogaciones cuyo impacto es menor en lo que a la intención inicial expresada por el usuario se refiere. Se trata de las *respuestas parciales*.

Definición 6.14 *Sean $d \in \mathcal{D}$ y $c \in \mathcal{Q}$ dos* GCB*'s definidos sobre un soporte $\mathcal{S}$. Se dice que una* secuencia ς de sustituciones, uniones y agregaciones de nodos es aceptable *si y sólo si ς es aceptable para las uniones y no existen demasiados nodos añadidos en relación al número de nodos de c. La proporción de nodos agregados permitidos (μ_a) se establece por el usuario.*

■

Definición 6.15 *Sean $d \in \mathcal{D}$ y $c \in \mathcal{Q}$ dos* GCB*'s definidos sobre un soporte $\mathcal{S}$. Se dice que d es una* respuesta parcial *a c si y sólo si existe una secuencia aceptable ς de sustituciones, uniones y agregaciones de nodos, tal que* $\text{proy}(c, \varsigma(d)) \neq \emptyset$.

■

Ejemplo 6.17 *Supongamos que planteamos la consulta $c \in \mathcal{Q}$ cuyo* GCB *asociado es $\mathcal{G}$, ilustrada en la Fig. 6.13. Si suponemos que nuestro documento $d \in \mathcal{D}$ es el* GCB *$\mathcal{H}$ de la misma figura, observamos como no se pueden proyectar todos los elementos del grafo $\mathcal{G}$ en $\mathcal{H}$. Sin embargo, si antes aplicamos sobre $\mathcal{H}$ primero una transformación de sustitución del nodo* [Organe, pétale] `([Órgano,pétalo])` *por* [Organe,tépale] `([Órgano,tépalo])` *y luego agregamos los nodos concepto* [Taille,court] `([Tamaño,corto])` *y relación* OT*, así como sus correspondientes aristas, comprobamos como* $\text{proy}(\mathcal{G}, \varsigma(\mathcal{H})) \neq \emptyset$.

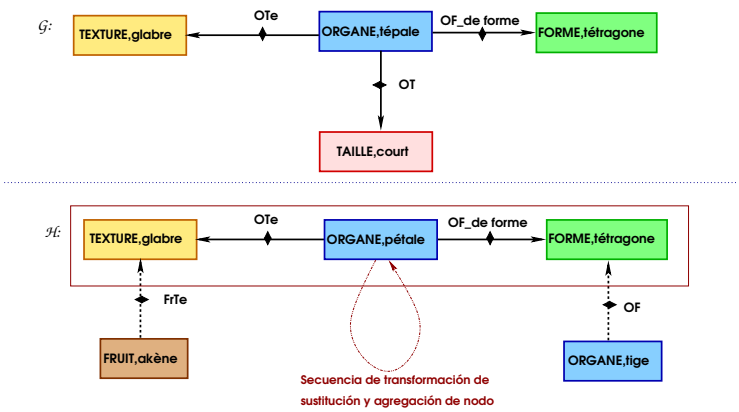


Figura 6.13: Respuesta parcial

■

6.3.2.3 | La función de ordenación

Una vez formalizado el problema de las respuestas a consultas, necesitamos integrar una estrategia de ordenación como último paso para completar el diseño de nuestra

arquitectura de RI conceptual. Con este propósito, la utilización de GCB's como términos de indexación nos permite situar de forma natural la pregunta en el dominio de las funciones basadas en subsunción y en instancias. En este punto, aunque los enfoques basados en CMAC's tienen el potencial suficiente para convertirse en un medio de clasificación poderoso, padecen en la práctica de carencia de eficiencia computacional, debido a su alto coste. Como alternativa, Genest [111] amplía la gama de relaciones conceptuales para conseguir técnicas más flexibles y menos ambiciosas, buscando un compromiso entre la eficiencia y el poder de discriminación. Por este motivo, el autor introduce las funciones de ordenación como simples órdenes parciales en el conjunto de transformaciones aplicadas a una consulta para alcanzar una proyección sobre la colección documental, es decir, para obtener una respuesta.

Definición 6.16 *Dado un soporte $\mathcal{S}$, sean $\mathcal{Q}$, $\mathcal{D} = \{d\}_{i \in I}$ los* GCB*'s asociados a una consulta y a una colección documental, y sea $\mathcal{R}_{\mathcal{Q}}^{\mathcal{D}}$ la colección de respuestas obtenidas mediante un conjunto $\mathcal{T}_{\mathcal{Q}}^{\mathcal{D}}$ de secuencias de transformaciones sobre grafos aplicadas en $\mathcal{Q}$ para obtener una proyección en algún $d_i,\ i \in I$. Se define una* función de ordenación asociada a $\mathcal{Q}$ y $\mathcal{D}$ *como la ordenación inducida naturalmente en $\mathcal{R}_{\mathcal{Q}}^{\mathcal{D}}$ mediante cualquier orden parcial de $\mathcal{T}_{\mathcal{Q}}^{\mathcal{D}}$.*

■

Este enfoque generaliza a los basados en CMAC's, al tiempo que nos permite flexibilizar las restricciones computacionales. En la práctica, nos centraremos concretamente en el orden parcial introducido por Genest en [111].

Definición 6.17 *Dado un soporte $\mathcal{S}$, sean $\mathcal{Q}$, $\mathcal{D} = \{d_i\}_{i \in I}$ los* GCB*'s asociados a una consulta y a una colección documental, y sea $\mathcal{R}_{\mathcal{Q}}^{\mathcal{D}}$ la colección de respuestas obtenidas mediante un conjunto $\mathcal{T}_{\mathcal{Q}}^{\mathcal{D}}$ de secuencias de transformaciones sobre grafos aplicadas en $\mathcal{Q}$ para obtener una proyección en algún $d_i,\ i \in I$. Se define el* orden parcial de Genest sobre los elementos $t,\ t' \in \mathcal{T}_{\mathcal{Q}}^{\mathcal{D}}$ *de la siguiente manera:*

$$t <_G t' \text{ si y sólo si } \begin{cases} t' & \text{asocia una respuesta aproximada } \textbf{OR} \\ t & \text{asocia una respuesta parcial } \textbf{OR} \\ t \text{ (resp. } t') & \text{asocia una respuesta parcial (resp. plausible) } \textbf{OR} \\ t,\ t' & \text{asocia el mismo tipo de respuesta } \textbf{AND } |t| > |t'| \end{cases}$$

mientras que

$$t =_G t' \text{ si y sólo si } t \textbf{ AND } t' \text{ asocian el mismo tipo de respuesta, } \textbf{AND } |t| = |t'|$$

■

Intuitivamente esto implica que cualquier respuesta aproximada es considerada más relevante que una plausible, y éstas, a su vez, son consideradas más relevantes que las

parciales. Si consideramos un mismo tipo de respuestas, la relevancia es inversamente proporcional al número de transformaciones individuales aplicadas[9]. Desde un punto de vista teórico, esto sigue siendo consistente con respecto a las consideraciones realizadas anteriormente sobre el impacto estructural en los GCB's debido a la aplicación de sustituciones, uniones o agregaciones. A pesar de su simplicidad, esta técnica ha demostrado aparentemente ser superior a las más recientes y sofisticadas [259], lo cual justifica su revisión y consideración formal.

Definido el entorno de trabajo basado en GCB's, vamos ahora con la introducción de las medidas de evaluación experimental de sistemas de RI.

6.4 | Medidas de evaluación

El modelo tradicional de evaluación experimental de sistemas de RI [64, 65] implica tres tareas complementarias: la recopilación de una colección documental, la definición de una serie de medidas de confianza para su evaluación y la elección adecuada de un conjunto de tópicos, es decir, de consultas.

A este respecto, es necesario tomar como punto de partida un fondo documental. Con respecto a las otras dos tareas, se trata de minimizar la carga de trabajo asociada a la creación de los JREL's así como a la selección de tópicos. Esto nos permitirá no tener que hacer frente a colecciones de prueba, que incluyen un número arbitrario de documentos en cualquier ámbito del conocimiento, algo difícilmente abordable a escala humana.

El objetivo aquí es tratar de discriminar la eficacia entre diferentes sistemas de RI, detectando cuales resultan ser más sensibles a la hora de identificar documentos relevantes. En un primer momento será necesario garantizar la estabilidad operativa del propio concepto de relevancia, ya introducido en la Definición 6.5. Sin embargo, lo cierto es que al parecer existen factores que influyen en la concretud de esta definición [277]. Es el caso de las discrepancias entre evaluadores o incluso contradicciones individuales [290] por parte de un mismo evaluador, factores que se ven reforzados por el hecho de que estamos hablando de una magnitud continua que se pretende clasificar mediante una secuencia de valores [297]. Con respecto a esto, asumimos que la influencia de estos factores de desestabilización es mínima, como ya se sugirió en un principio en [131], y que más tarde se corroboró experimentalmente en [334]. Del mismo modo, el desacuerdo en el número de documentos relevantes parece no tener un fuerte impacto en la clasificación de los sistemas [290], probablemente porque tener más documentos relevantes beneficia a la mayoría de los sistemas de manera uniforme.

Si centramos ahora nuestra atención en la elección de un conjunto de tópicos y en las clasificaciones devueltas por los entornos de RI, se pueden distinguir dos marcos genéricos de acuerdo con el estado del arte. Por un lado, el inspirado en la extensa experiencia

[9] esto es, sustituciones, uniones y agregaciones de nodos.

acumulada durante décadas en los eventos del TREC y caracterizado esencialmente por el uso preferente de juicios humanos[10], sin tener en cuenta en el proceso de la sencillez o complejidad del tópico. Se habla entonces de un marco *basado en la valoración de tipo humano*. Por el otro, un conjunto de técnicas inspiradas en dos supuestos razonables esbozados en [207] en relación al *«principio de facilidad y/o dificultad»* de determinadas consultas y el *«principio de lo bueno o malo»* que puede resultar ser un sistema de RI. A diferencia de la basada en la valoración de tipo humano, ésta formaliza la sencillez o complejidad de un tópico a partir de medidas basadas en JREL's como un factor importante que impacta en esta tarea. De un modo más detallado, el primer principio establece que deberíamos asignar un peso mayor (resp. menor) tanto si se comete un error en consultas sencillas (resp. difíciles), como si se contesta correctamente en las consideradas difíciles (resp. fáciles). El segundo asume que deberíamos ser capaces de realizar consultas complicadas a los buenos sistemas, mientras que los malos sólo debieran ser capaces de contestar a las sencillas. En adelante, nos referiremos a este marco como el *basado en una valoración tipo máquina*.

Como alternativa, en lo que ocupa exclusivamente la ordenación de sistemas de RI, se ha propuesto una tercera vía que prescinde por completo de uso de recursos basados en JREL's [347]. Se trata en este caso de evaluar el rendimiento de un motor de búsqueda utilizando una medida llamada *contador de referencia*, un tipo específico de puntuación que se calcula mediante el número de ocurrencias de los documentos más relevantes devueltos en los resultados de una colección de otros sistemas de recuperación.

6.4.1 | Sistemas de RI con ordenación usando JREL's

La utilización de JREL's es la base de la mayoría de las medidas de evaluación de los sistemas de RI, popularizadas entre la comunidad investigadora gracias a las conferencias del TREC. De este modo, podemos distinguir entre dos acercamientos según tengamos en cuenta o no el orden asociado a la clasificación de los resultados devueltos durante la recuperación, lo que actualmente es habitual en los motores de búsqueda.

6.4.1.1 | Medidas de evaluación basadas en conjuntos

Este tipo de medida estima la calidad de un conjunto no ordenado de documentos recuperados. Se trata de técnicas asociadas a la evaluación de un modelo de RI bidimensional [131]. Esto es, no se considera el orden asociado a las clasificaciones de los contextos y la evaluación sólo se centra en el carácter relevante o no de los documentos recuperados. En este sentido, se introducen una serie de medidas que detallamos a continuación.

Definición 6.18 *Sean σ un sistema de* RI*, donde $\mathcal{D} = \{d_i\}_{i \in I}$ es una colección*

[10] mediante mecanismos de JREL's o similares, como en el caso de PJREL's.

documental, y $\mathcal{Q} = \{c_j\}_{j \in J}$ *un conjunto finito de tópicos (consultas). Se define la* precisión *(resp. la* cobertura*)* de σ con respecto del tópico c_j para la colección documental $\mathcal{D}$ *como:*

$$\text{P}(\sigma, c_j, \mathcal{D}\ ) := \frac{|\text{rec}(\sigma, c_j, \mathcal{D}) \cap \text{rel}(c_j, \mathcal{D})\ |}{|\text{rec}(\sigma, c_j, \mathcal{D})|} \tag{6.26}$$

$$(\textit{resp.}\ \text{C}(\sigma, c_j, \mathcal{D}\ ) := \frac{|\text{rec}(\sigma, c_j, \mathcal{D}) \cap \text{rel}(c_j, \mathcal{D})\ |}{|\text{rel}(c_j, \mathcal{D})\ |}) \tag{6.27}$$

donde $\text{rec}(\sigma, c_j, \mathcal{D})$ *(resp.* $\text{rel}(\sigma, c_j, \mathcal{D})$*) es el conjunto de documentos de* $\mathcal{D}$ *recuperados por* σ *(resp. los documentos relevantes) para el tópico* $c_j \in \mathcal{Q}$*.*

■

Tanto la *precisión* como la *cobertura* fueron introducidas por Cleverton *et al.* en [63]. Intuitivamente, la precisión (resp. la cobertura) representa la proporción entre el número de documentos relevantes recuperados y el número de documentos recuperados en total (resp. documentos relevantes totales), es decir, un valor predictivo positivo de la tarea de búsqueda (resp. la sensibilidad). Por lo tanto, la precisión (resp. la cobertura) evalúa la exactitud (resp. la exhaustividad) de la búsqueda en función de los resultados. En particular, la precisión (resp. la cobertura) no se define cuando no se recuperan documentos (resp. cuando no hay documentos relevantes) en la colección y es mínima (resp. máxima) cuando todos ellos son devueltos por el buscador. En cualquier caso, se trata de conceptos complementarios calculados con respecto a toda la lista de documentos devueltos por el sistema, lo cual plantea algún problema a la hora de estimar la efectividad. Esto justifica la introducción por von Rijsbergen en [314] de la medida F_β como una manera de estimar la efectividad de la recuperación con respecto al usuario, que concede β veces tanta importancia a la cobertura como a la precisión.

Definición 6.19 *Sean* σ *un sistema de* RI*,* $\mathcal{D} = \{d_i\}_{i \in I}$ *una colección documental y* $\mathcal{Q} = \{c_j\}_{j \in J}$ *un conjunto finito de tópicos (consultas). Se define, por* $\beta \in \mathbb{R}^+ \cup \{0\}$*, la medida* F_β de σ con respecto al tópico c_j y la colección documental $\mathcal{D}$ *como:*

$$\text{F}_\beta(\sigma, c_j, \mathcal{D}) := \frac{(1 + \beta^2)\ .\ [\text{P}(\sigma, c_j, \mathcal{D})\ .\ \text{C}(\sigma, c_j, \mathcal{D})]}{\beta^2\ .\ \text{P}(\sigma, c_j, \mathcal{D}) + \text{C}(\sigma, c_j, \mathcal{D})} \tag{6.28}$$

En el caso particular de que $\beta = 1$*, se habla de* medida F.

■

La medida F_β permite hacer enfásis sobre los pesos asociados a la precisión con respecto a la cobertura, utilizando como valor de control a β. Así, cuando $\beta = 1$, se

obtiene la *media armónica* de ambas medidas, que en comparación con la aritmética requiere que los dos valores sean elevados para que a su vez también ella lo sea. En cambio, para valores $\beta < 1$ pesará más la precisión mientras que para valores $\beta > 1$ lo hará la cobertura. Por otro lado, ninguna de estas medidas considera la proporción de documentos no relevantes que se recuperan, situación a la que pretende dar respuesta la introducción del ratio de *fracaso o irrelevancia*[11].

Definición 6.20 *Sean σ un sistema de* RI, *$\mathcal{D} = \{d_i\}_{i \in I}$ una colección documental y $\mathcal{Q} = \{c_j\}_{j \in J}$ un conjunto finito de tópicos (consultas). Se define el* fracaso de σ con respecto al tópico c_j en la colección documental $\mathcal{D}$ *como:*

$$\text{FR}(\sigma, c_j, \mathcal{D}) := \frac{|\text{rec}(\sigma, c_j, \mathcal{D}) \cap \text{nrel}(c_j, \mathcal{D})\,|}{|\text{nrel}(c_j, \mathcal{D})\,|} \tag{6.29}$$

donde $\text{nrel}(c_j, \mathcal{D})$ *es el conjunto de documentos de $\mathcal{D}$ que no son relevantes a $c_j \in \mathcal{Q}$.*

■

De esta manera, el fracaso, que fue inicialmente introducido por Salton y McGill [273], se puede interpretar como la probabilidad de que un documento no relevante sea recuperado. Así, este valor devolverá 0 cuando no se recupere ningún documento como respuesta a una consulta.

6.4.1.2 | Medidas de evaluación basadas en ordenación

Este tipo de medida considera el orden en el que se presentan los documentos devueltos, una mejora sustancial en relación con las métricas anteriores, ya que estima la precisión en todos los niveles de cobertura. Como consecuencia, se pueden derivar dos mejoras prácticas. La primera hace referencia a la real contribución que implica disponer de información extra sobre el grado de relevancia asociado al sistema de recuperación con respecto a una consulta dada. La segunda permite estimar la eficiencia de un sistema de RI, incluso cuando sólo estamos interesados en calcularlo sobre resultados recuperados en los niveles más bajos. Es el caso típico de la recuperación Web, donde el usuario normalmente se desentiende de las respuestas que no se encuentren en las primeras páginas. Formalmente [268], estas mejoras se traducen en dos aspectos: la *estabilidad*[12] y la *sensibilidad*[13] de la tarea de evaluación.

[11] en terminología anglosajona *fall-out rate*.

[12] la estabilidad de una medida está relacionada con la capacidad que tiene de identificar sistemáticamente las diferencias entre los sistemas a partir de una muestra de tópicos [51].

[13] también llamada *ratio de cobertura*, se refiere a las medidas de evaluación del poder de discriminación de un sistema de RI, sobre una colección de prueba y una serie de ejecuciones realizadas y definidas a partir de la colección [336].

Una primera aproximación para conseguirlo consiste en determinar la precisión frente a la cobertura de cada uno de los documentos recuperados. Para ello, sincronizaremos ambas medidas sobre la base de los primeros k documentos devueltos.

Definición 6.21 *Sean σ un sistema de* RI, $\mathcal{D} = \{d_i\}_{i\in I}$ *una colección documental, y $\mathcal{Q} = \{c_j\}_{j\in J}$ un conjunto finito de tópicos (consultas). Se define la* precisión *(resp. la* cobertura*)* de los k documentos devueltos por σ con respecto a los tópicos c_j sobre $\mathcal{D}$, *denotada por* P@$k(\sigma, c_j, \mathcal{D})$ *(resp.* C@$k(\sigma, c_j, \mathcal{D})$*), como:*

$$\text{P@}k(\sigma, c_j, \mathcal{D}) := \frac{|\{\text{reco}(\sigma, c_j, \mathcal{D})_l\}_{l=1}^{k} \cap \text{rel}(c_j, \mathcal{D})\,|}{k} \tag{6.30}$$

$$(\textit{resp.}\ \text{C@}k(\sigma, c_j, \mathcal{D}) := \frac{|\{\text{reco}(\sigma, c_j, \mathcal{D})_l\}_{l=1}^{k} \cap \text{rel}(c_j, \mathcal{D})\,|}{|\text{rel}(c_j, \mathcal{D})\,|}) \tag{6.31}$$

donde reco$(\sigma, c_j, \mathcal{D})$ *es la lista, ordenada en base a su relevancia, de los documentos recuperados por σ para el tópico c_j.*

■

Llegados aquí, estamos en disposición de expresar la precisión en función de la cobertura, simplemente calculando ambas medidas en los puntos de sincronización. Como resultado obtenemos un grafo de la precisión/cobertura [198, 241].

Definición 6.22 *Sean σ un sistema de* RI, $\mathcal{D} = \{d_i\}_{i\in I}$ *una colección documental, y $\mathcal{Q} = \{c_j\}_{j\in J}$ un conjunto finito de tópicos (consultas). Se puede expresar la* precisión de σ sobre el tópico c_j para la colección documental $\mathcal{D}$ en función de la cobertura *como:*

$$\text{P}_{\text{C}}(\sigma, c_j, \mathcal{D}, c) := \text{P@}k(\sigma, c_j, \mathcal{D}),\ c = \text{C@}k(\sigma, c_j, \mathcal{D}) \tag{6.32}$$

■

Intuitivamente, la precisión se calcula en el mismo instante que la cobertura, justo en el momento en el que el motor de búsqueda devuelve el documento. Como resultado [199], este tipo de curvas tiene una particularidad y es que presenta la forma de diente de sierra ya que si el $(k+1)$-ésimo documento recuperado no es relevante entonces la cobertura será la misma para los k primeros, pero la precisión experimentará un descenso. Sin embargo, en el caso de que el documento sea relevante, entonces tanto la precisión como la cobertura se incrementarán, y la curva despuntará hacia la derecha. En este sentido, resulta útil eliminar estas sacudidas y la manera estándar de hacerlo es a través de la interpolación.

Definición 6.23 *Sean σ un sistema de* RI*, $\mathcal{D} = \{d_i\}_{i\in I}$ una colección documental y $\mathcal{Q} = \{c_j\}_{j\in J}$ un conjunto finito de tópicos (consultas). Se define la* precisión interpolada de σ sobre el tópico c_j en función de la cobertura *para la colección documental $\mathcal{D}$, como:*

$$\text{PI}_\text{C}(\sigma, c_j, \mathcal{D}, c) := \max_{c' \geq c} \text{P}_\text{C}(\sigma, c_j, \mathcal{D}, c') \tag{6.33}$$

■

De esta manera, la medida refiere a la precisión más alta encontrada para la solución del problema planteado. Por el otro lado, aunque hemos utilizado P@k como primer paso para introducir el grafo de precisión/cobertura, el concepto también posee interés en sí mismo. Así, una de las ventajas que se suele argumentar en su favor es que no requiere de la estimación del conjunto de documentos pertinentes. Sin embargo, por el mismo motivo no calcula correctamente la media y no podemos considerarlo como un criterio estable de evaluación [199]. Una alternativa para aliviar este problema es la *R-precisión* (resp. la *R-cobertura*) [273].

Definición 6.24 *Sean σ un sistema de* RI*, $\mathcal{D} = \{d_i\}_{i\in I}$ una colección documental y $\mathcal{Q} = \{c_j\}_{j\in J}$ un conjunto finito de tópicos (consultas). Se define la* R-precisión*, denotada por* P@$R(\sigma, c_j, \mathcal{D})$ *(resp.* R-cobertura *y denotada por* C@$R(\sigma, c_j, \mathcal{D})$*),* de σ sobre el tópico c_j para la colección documental $\mathcal{D}$ *como:*

$$R\text{-P}(\sigma, c_j, \mathcal{D}\ ) := \text{P@}R(\sigma, c_j, \mathcal{D}) \tag{6.34}$$

$$(\textit{resp. } R\text{-C}(\sigma, c_j, \mathcal{D}\ ) := \text{C@}R(\sigma, c_j, \mathcal{D})) \tag{6.35}$$

donde $R = |\text{rel}(c_j, \mathcal{D})|$.

■

Intuitivamente, si la colección documental incluye R documentos relevantes para una consulta dada, entonces R-P indicará la cantidad de relevantes una vez que los R mejores resultados hayan sido estudiados por el sistema. En resumen, se refiere a la mejor precisión sobre el grafo P_C, lo que justifica que también sea conocido como el *punto de equilibrio de* P_C, ya que la precisión y la cobertura coinciden en él.

En cualquier caso, ninguna de las métricas de relevancia graduada es tan ampliamente utilizada actualmente como la *precisión media* (PM), que proporciona una interpretación geométrica de los grafos de precisión/cobertura [272]. En efecto, calcula el área bajo la curva P_C, lo que implica estimar el valor medio de la cobertura para el intervalo $[0, 1]$.

Definición 6.25 *Sean σ un sistema de* RI*, $\mathcal{D} = \{d_i\}_{i\in I}$ una colección documental y $\mathcal{Q} = \{c_j\}_{j\in J}$ una colección de tópicos (consultas). Se define la* precisión media de σ con respecto al tópico c_j para la colección $\mathcal{D}$ *como:*

$$\text{PM}(\sigma, c_j, \mathcal{D}\ ) := \int_0^1 \text{P}_\text{C}(\sigma, c_j, \mathcal{D})\ d\text{C} \tag{6.36}$$

En la práctica, este valor se aproxima mediante una suma discreta sobre cada posición de la secuencia ordenada de documentos devueltos, tal como sigue:

$$\text{PM}(\sigma, c_j, \mathcal{D}\,) := \frac{1}{|\text{rel}(c_j, \mathcal{D})\,|} \sum_{k=1}^{|\text{reco}(\sigma, c_j, \mathcal{D})|} \delta(\text{reco}(\sigma, c_j, \mathcal{D})_k)\ .\ \text{P@}k(\sigma, c_j, \mathcal{D}) \quad (6.37)$$

donde

$$\delta(\text{reco}(\sigma, c_j, \mathcal{D})_k) := \begin{cases} 1 & \text{si reco}(\sigma, c_j, \mathcal{D})_k \in \text{rel}(c_j, \mathcal{D}\,) \\ 0 & \text{en cualquier otro caso} \end{cases}$$

■

En la práctica, PM y R-P están altamente correlacionados [302, 337] y muestran una estabilidad similar en términos de comparación de sistemas usando tópicos diferentes [37]. Aunque esto podría parecer algo aparentemente sorprendente[14], se puede demostrar formalmente [15] que si se asume un conjunto razonable de suposiciones, ambas medidas aproximan el área bajo la curva P_C, lo que explica el fenómeno. Además, podemos mejorar la estabilidad calculando el promedio de la PM a través de las consultas [127].

Definición 6.26 *Sean σ un sistema de* RI, *$\mathcal{D} = \{d_i\}_{i\in I}$ una colección documental y $\mathcal{Q} = \{c_j\}_{j\in J}$ un conjunto finito de tópicos (consultas). Se define el* promedio de la precisión media de σ sobre el conjunto de tópicos $\mathcal{Q}$ para una colección documental $\mathcal{D}$ *como:*

$$\text{PPM}(\sigma, \mathcal{Q}, \mathcal{D}\,) := \frac{\sum_{j\in J} \text{PM}(\sigma, c_j, \mathcal{D})}{|\mathcal{Q}|} \quad (6.38)$$

■

Mientras que PM aproxima al área bajo la curva P_C, PPM es aproximadamente el promedio de ese mismo área para un conjunto de consultas. De hecho, PPM es la medida de uso más frecuente en lo que a recuperación con ordenación se refiere, lo que provocó que se convirtiera en un estándar para la comunidad TREC. Considera aspectos orientados tanto a la cobertura como a la precisión, y es sensible a la ordenación devuelta por el sistema, proporcionando una medida de calidad a través de los niveles de cobertura sobre una única figura. Sin embargo, el PPM tiene el efecto de ponderar por igual cada una de las necesidades de información en el resultado final que devuelve, aunque existan muchos documentos relevantes para algunas consultas, mientras que existan muy pocos para otras. Esto significa que un conjunto de prueba debe ser lo suficientemente grande y variado

[14] el cómputo de la R-P considera un único punto de precisión mientras que la PM evalúa el área bajo toda la curva P_C.

para llegar a ser representativo de la eficacia del sistema sobre las diferentes consultas. Asumiendo estas condiciones, el PPM ha demostrado poseer una especial sensibilidad y estabilidad entre las medidas de evaluación [199]. Por lo demás, es necesaria la utilización de otro tipo de métricas cuando lo que interesa es destacar las mejoras en consultas de bajo rendimiento.

Definición 6.27 *Sean σ un sistema de* RI*, $\mathcal{D} = \{d_i\}_{i\in I}$ una colección documental y $\mathcal{Q} = \{c_j\}_{j\in J}$ un conjunto finito de tópicos (consultas). Se define el* promedio geométrico de la precisión media de σ sobre el conjunto de tópicos $\mathcal{Q}$ de la colección documental $\mathcal{D}$ *como:*

$$\text{PGPM}(\sigma, \mathcal{Q}, \mathcal{D}\ ) := \sqrt[J]{\prod_{j\in J} \text{PM}(\sigma, c_j, \mathcal{D})} \tag{6.39}$$

■

Tanto el PPM como el PGPM pueden verse como maneras diferentes de alcanzar una medida de calidad a través de la incorporación de diferentes observaciones individuales. Así, mientras la primera es la media aritmética de la PM, considerando un conjunto de tópicos, la segunda es la media geométrica. En este sentido, el PGPM es más representativo de la eficacia a través de un conjunto de consultas, y más robusto frente a situaciones en las que la presencia de unas pocas interrogaciones con buen rendimiento pueden sesgar la clasificación obtenida mediante el PPM. Concretamente, el PGPM fue introducido por Voorhees en [335].

En este punto, si quisiéramos resumir en una característica común las métricas descritas hasta el momento, tendríamos que decir que todas ellas vienen completamente determinadas por la ordenación de los documentos relevantes en el conjunto resultante. Por lo tanto, no hacen distinción entre los documentos que son explícitamente juzgados como no relevantes y aquéllos que se asume que no son relevantes por no haber sido juzgados, lo cual plantea un problema cuando se sabe que los JREL's proporcionados están lejos de ser completos, haciéndose aconsejable el atenuar esta situación.

Definición 6.28 *Sean σ un sistema de* RI*, $\mathcal{D} = \{d_i\}_{i\in I}$ una colección documental y $\mathcal{Q} = \{c_j\}_{j\in J}$ un conjunto finito de tópicos (consultas). Se define la* relación de preferencia binaria de σ sobre el tópico c_j en la colección documental $\mathcal{D}$ *como:*

$$\text{PREFB}(\sigma, c_j, \mathcal{D}\ ) := \frac{1}{\text{R}} \sum_{r\in \text{R}} [1 - \frac{|\text{nrel}(c_j, \mathcal{D}) \cap \{\text{reco}(\sigma, c_j, \mathcal{D})\}_{r+1}^{\text{R}}|}{\text{mín}\{\text{R}, |\text{nrel}(c_j, \mathcal{D})|\}}] \tag{6.40}$$

donde $\text{R} = |\text{rel}(c_j, \mathcal{D})|$*. Se puede extender de un modo natural esta definición al conjunto finito de tópicos $\mathcal{Q}$.*

■

La medida PREFB, introducida por Buckley *et al.* [38] puede pensarse como la inversa de la fracción de los documentos recuperados que son juzgados como no relevantes y que se sitúan en una posición anterior a los relevantes. De este modo, se calcula una relación de preferencia en función de si los documentos juzgados como relevantes se recuperan antes que los juzgados como irrelevantes, esto es, la medida está basada únicamente en las ordenaciones relativas de los documentos que han sido juzgados previamente. Hablamos de preferencias binarias porque la relación se define a partir de un JREL binario, de tal manera que, dada una consulta, se prefiere cualquier documento relevante frente a los que no lo son. En este sentido, PREFB y PPM están altamente correlacionados cuando se utilizan con JREL's completos. Sin embargo, cuando éstos son incompletos, aunque los sistemas de ordenación mediante la PREFB todavía se correlacionan mucho con los originales, no es el caso de los que ordenan mediante PPM.

Una última propuesta que ha conseguido una aceptación cada vez mayor, especialmente cuando se emplea asociada a sistemas de aprendizaje automático, es la *ganancia acumulativa* (GAA) [199]. Normalmente, la valoración inicial proporcionada por los sistemas de RI posee múltiples grados y, en consecuencia, la mejora debería ser evaluada separadamente en cada nivel de relevancia. En este sentido, los documentos considerados como más relevantes que aparezcan en peores puestos en la lista proporcionada por el sistema debieran ser penalizados, reduciendo el valor de su relevancia. Sea como sea, las medidas dependientes de la ordenación descritas hasta ahora son calculadas usando unas valoraciones dicotómicas acerca de la relevancia, colapsando éstas en dos para su evaluación.

Definición 6.29 *Sean σ un sistema de* RI*, $\mathcal{D} = \{d_i\}_{i\in I}$ una colección documental y $\mathcal{Q} = \{c_j\}_{j\in J}$ un conjunto finito de tópicos (consultas). Se define la* ganancia acumulativa reducida de σ sobre el tópico c_j en la colección documental $\mathcal{D}$ en la posición ordenada $r \in [1, R] \cap \mathbb{N}$ *como:*

$$\mathrm{GAAR}(\sigma, c_j, \mathcal{D})_r := \mathrm{G}(\sigma, c_j, \mathcal{D})_1 + \sum_{k=2}^{r} \frac{\mathrm{G}(\sigma, c_j, \mathcal{D})_k}{log_b(k)} \tag{6.41}$$

donde $R = |\mathrm{rel}(c_j, \mathcal{D})|$ y G *es la secuencia de valores relevantes asociados a la lista* $\mathrm{reco}(\sigma, c_j, \mathcal{D})$*. Se puede extender naturalmente esta definición al conjunto finito de tópicos $\mathcal{Q}$.*

■

En la práctica, la GAAR usa el nivel de relevancia como una medida de valor acumulado en la posición de ordenación asociada al documento, añadiendo esta ganancia progresivamente desde la primera posición a la última. Se asocia una función logarítmica reducida con el fin de aminorar poco a poco el valor del documento al mismo tiempo que se incrementa su posición en la ordenación, pero no demasiado bruscamente.

Normalmente, se usa un logaritmo en base dos, esto es, considerando $b = 2$ en la Ecuación 6.41.

Aunque el conjunto de documentos recuperados puede variar ampliamente entre diferentes sistemas, para comparar sus rendimientos, la versión normalizada de esta medida utiliza el mayor valor posible de GAAR para cada una de las posiciones.

Definición 6.30 *Sean $\sigma = \{\sigma_i\}_{i\in I}$ una colección de sistemas de* RI*, $\mathcal{D} = \{d_j\}_{j\in J}$ una colección documental, $\mathcal{Q} = \{c_k\}_{k\in K}$ un conjunto finito de tópicos (consultas) y $\{\text{GAAR}(\sigma, c_k, \mathcal{D})_l\}_{l\in L}$ la secuencia (conjunto ordenado) de valores de* GAAR *para el tópico c_k. Se define la* ganancia acumulativa reducida normalizada de σ_i sobre el tópico c_k de la colección documental $\mathcal{D}$ en la posición ordenada $r \in [1, R] \cap \mathbb{N},\ R = |rel(c_k, \mathcal{D})|$ *como:*

$$\text{GAARN}(\sigma_i, c_k, \mathcal{D})_r := \frac{\text{GAAR}(\sigma_i, c_k, \mathcal{D})_r}{\text{GARI}(\sigma_i, c_k, \mathcal{D})_r} \tag{6.42}$$

donde GARI *se denomina la* GAAR ideal*, y se define como el* GAAR *máximo alcanzable en el rango r. Ésta se puede calcular fácilmente a partir de las* GAAR*'s de una lista ordenada que sitúa todos los documentos con mejor clasificación por encima de todos los segundos y así sucesivamente. Se puede extender naturalmente esta definición al conjunto finito de tópicos $\mathcal{Q}$.*

■

Obviamente, en un algoritmo de ordenación perfecto asociado a un sistema de RI, los valores correspondientes para GAARN serán iguales a 1. Ambas métricas GAAR y GAARN fueron introducidas por Järvelin y Kekäläinen en [143]. Los resultados obtenidos indican una fuerte correlación entre la satisfacción de los usuarios, la GAA y la precisión; una correlación más moderada con la GAAR y una sorprendentemente posible correlación casi despreciable con la GAARN [8].

6.4.2 | Sistemas de RI con ordenación usando PJREL's

Introducida por Soboroff *et al.* en [290], esta técnica simplemente retoma el proceso oficial de evaluación del TREC [332], cambiando algún aspecto referido a la valoración del entrenamiento basado en asesoramiento humano. Más exactamente, se consideran los siguientes pasos, descritos por los autores:

1. Se selecciona un grupo de 50 consultas siguiendo la propuesta de un grupo de expertos de confianza, normalmente de la organización NIST[15].

[15] por *National Institute of Standards and Technology*.

2. Se lanzan para su evaluación un número de ejecuciones, asociadas a cada sistema de RI evaluado. Cada una de estas ejecuciones consta (como máximo) de los mejores 1.000'00 documentos recuperados para cada tópico. Por cada participante se crea un subconjunto con estas características que se etiqueta como *ejecución oficial*.

3. El grupo de expertos toma los n primeros documentos devueltos en cada consulta para cada una de las ejecuciones oficiales, eliminando las duplicidades, con el fin de crear un *fondo* para cada una de ellas.

4. Se selecciona aleatoriamente un conjunto de documentos para formar los PJREL's, utilizando un modelo para determinar la relevancia de los documentos que están en ese fondo.

5. A partir del conjunto de PJREL's, se evalúan todas las ejecuciones usando el paquete de evaluación del TREC[16].

Esto es, con respecto al TREC, Soboroff *et al.* tomaron en el tercer paso los valores $n = 10$ ó $n = 100$, mientras que el TREC considera únicamente el caso de que n sea igual a 10. A su vez, en el cuarto paso sustituyeron el papel de los expertos por una elección totalmente aleatoria. Finalmente, en los pasos cuarto y quinto, consideraron los PJREL's en vez de los JREL's. Obviamente, para estimar este tipo de clasificación podemos considerar todas las medidas previamente descritas para los entornos de evaluación basados en JREL's.

6.4.3 | Sistemas de RI con ordenación basada en la valoración de la máquina

Descrita por Mizzaro *et al.* [208], esta técnica toma como base la estimación de lo fácil o difícil que puede resultar un tópico, considerando que si el motor de búsqueda quiere tener un alto rendimiento deberá ser suficientemente eficaz en las consultas difíciles. Vamos a bautizar a esta propiedad asociada a un sistema de RI como su *autoridad*, y antes de formalizarla necesitaremos introducir algunos conceptos para la captura de las nociones de facilidad de la consulta y la eficacia del sistema. El punto de partida para esta metodología es la noción de PM, cuyo cálculo puede ser aplicado tanto a JREL's como a PJREL's.

Definición 6.31 *Sean $\sigma = \{\sigma_i\}_{i\in I}$ un conjunto de sistemas de* RI*, $\mathcal{D}$ una colección documental, y $\mathcal{Q} = \{c_j\}_{j\in J}$ un conjunto finito de tópicos (consultas). Se define la* media de la precisión media del conjunto de sistemas de RI σ aplicado a un tópico c_j para la colección $\mathcal{D}$*, como:*

$$\text{MPM}(\sigma, c_j, \mathcal{D}\ ) := \frac{\sum_{i\in I} \text{PM}(\sigma_i, c_j, \mathcal{D})}{|\sigma|} \tag{6.43}$$

■

[16] consultar `http://trec.nist.gov/trec_eval/`.

Intuitivamente, la MPM es un indicador de la facilidad asociada a la satisfacción de la consulta, entendiéndola como una magnitud directamente relacionada con el número de sistemas de RI que poseen un buen rendimiento para ese tópico. A partir de la base que ofrece esta medida, Mizzaro *et al.* [208] extienden el concepto de PM con el fin de obtener una directriz fiable para estimar el rendimiento de un sistema de RI sobre las distintas consultas. La idea pasa, en primer lugar, por normalizar la PM con el fin de eliminar cualquier influencia achacable a la facilidad de aquéllas por separado (resp. de la eficacia del sistema de manera individual), con el fin de obtener una medida fiable del rendimiento en un conjunto de sistemas de RI (resp. de lo fácil que resulte ser una consulta).

Definición 6.32 *Sean $\sigma = \{\sigma_i\}_{i\in I}$ un conjunto de sistemas de* RI*, $\mathcal{D}$ una colección documental, y $\mathcal{Q} = \{c_j\}_{j\in J}$ un conjunto finito de tópicos (consultas). Se define la* precisión media normalizada de σ_i aplicada al tópico c_j de acuerdo con la $\text{MPM}(\sigma, c_j, \mathcal{D})$*, como:*

$$\text{PMN}_{\text{MPM}}(\sigma_i, c_j, \mathcal{D}\ ) := \text{PM}(\sigma_i, c_j, \mathcal{D}) - \text{MPM}(\sigma, c_j, \mathcal{D}\ ) \tag{6.44}$$

■

De esta manera, la matriz de adyacencia $[\text{PMN}_{\text{MPM}}(\sigma_i, c_j, \mathcal{D})]_{(i,j)\in I\times J}$ puede ser interpretada como un grafo ponderado bipartito, donde el peso de los arcos $c_j \to \sigma_i$ corresponde a los valores de $\text{PMN}_{\text{MPM}}(\sigma_i, c_j, \mathcal{D})$, lo que refleja el desempeño individual de σ_i sobre el tópico c_j y la eliminación de las desviaciones debido a la facilidad de éste. La medida de PMN_{MPM} fue introducida por Wu y McClean en [346], y Mizzaro [207] calculó su media con el fin de buscar una mejor estabilidad.

Definición 6.33 *Sean $\sigma = \{\sigma_i\}_{i\in I}$ un conjunto de sistemas de* RI*, $\mathcal{D}$ una colección documental, y $\mathcal{Q} = \{c_j\}_{j\in J}$ un conjunto finito de tópicos (consultas). Se define el* promedio normalizado de la precisión media de σ_i sobre el conjunto de consultas $\mathcal{Q}$ para la colección documental $\mathcal{D}$*, como:*

$$\text{PNPM}(\sigma_i, \mathcal{Q}, \mathcal{D}\ ) := \frac{\sum_{j\in J} \text{PMN}_{\text{MPM}}(\sigma_i, c_j, \mathcal{D})}{|\mathcal{Q}|} \tag{6.45}$$

■

Sorprendentemente, el PNPM muestra un comportamiento algo distinto a los resultados del TREC, proporcionando una clasificación muy diferente en relación con PPM, aunque ambas medidas están relacionadas[17]. En la práctica, lo que generalmente

[17] la correlación tau de Kendall [162] es $0'87$ y la correlación lineal [48] es $0'92$.

se considera una versión mejorada de un sistema mediante la aplicación de criterios del TREC[18] a menudo resulta no serlo cuando se utiliza PNPM.

Una alternativa para aprovechar la información contenida en la matriz de adyacencia PMN_{MPM} pasa por analizarla sobre la base del algoritmo de HITS de Kleinberg [165] para obtener medidas de evaluación más sofisticadas teniendo en cuenta los conjuntos en su totalidad para ambos, sistemas de RI y consultas. La idea básica propuesta por Mizzaro *et al.* consiste en retomar los indicadores descritos por Kleinberg para la localización de información de alta calidad relacionada con las estructuras de enlace: la *conectividad* y la *autoridad*.

Definición 6.34 *Sean $\sigma = \{\sigma_i\}_{i\in I}$ un conjunto de sistemas de* RI*, $\mathcal{D}$ una colección documental, y $\mathcal{Q} = \{c_j\}_{j\in J}$ un conjunto finito de tópicos (consultas). Se define la* autoridad de un sistema de RI σ_i sobre el conjunto de consultas $\mathcal{Q}$ *(resp. la* conectividad del tópico c_j en el sistema de RI σ*)* para la colección $\mathcal{D}$*, como:*

$$\text{A}(\sigma_i, \mathcal{Q}, \mathcal{D}\,) := \sum_{j\in J} \text{T}(c_j, \sigma, \mathcal{D}\,) \cdot \text{PMN}_{\text{MPM}}(\sigma_i, c_j, \mathcal{D}\,) \tag{6.46}$$

$$(\textit{resp.}\ \text{T}(c_j, \sigma, \mathcal{D}\,) := \sum_{i\in I} \text{A}(\sigma_i, \mathcal{Q}, \mathcal{D}\,) \cdot \text{PMN}_{\text{MPM}}(\sigma_i, c_j, \mathcal{D})) \tag{6.47}$$

■

Intuitivamente, un sistema de RI posee una autoridad alta si es más eficiente sobre los tópicos con una también alta conectividad, es decir, cuando se trata de consultas difíciles. Esto proporciona un criterio de ordenación simple, ya que un sistema que quiere ser eficaz debería presentar unos valores altos en la autoridad asociada.

6.4.4 | Sistemas de RI con ordenación en base a contadores de referencia ponderados

Descrito por Wu *et al.* en [347], esta propuesta aplica una técnica de fusión de datos que compara los resultados obtenidos para un motor de búsqueda con las tomadas a partir de una colección de otros sistemas de RI distintos. Ello requiere la introducción previa de un cierto número de conceptos.

Definición 6.35 *Sean $\sigma = \{\sigma_i\}_{i\in I}$ un conjunto de sistemas de* RI*, $\mathcal{D} = \{d_j\}_{j\in J}$ una colección documental, y $\mathcal{Q} = \{c_k\}_{k\in K}$ un conjunto finito de tópicos (consultas). Denotamos por*

$$\text{CR}(\sigma_i, c_k, \mathcal{D}\,) := \sum_{j_i\in J_i} a(\text{reco}(\sigma_i, c_k, \mathcal{D})_{j_i}) \tag{6.48}$$

[18] es decir, una versión con un mayor PPM.

al contador de referencia de σ_i sobre el tópico c_k para la colección documental $\mathcal{D}$, *donde* $a(\text{reco}(\sigma_i, c_k, \mathcal{D})_{j_i})$ *es el número de apariciones de un documento* $\text{reco}(\sigma_i, c_k, \mathcal{D})_{j_i}$ *en la lista* $\{\text{reco}(\sigma_l, c_k, \mathcal{D})\}_{j_l \in J_l, l \neq i}$.

Dado $a(\text{reco}(\sigma_i, c_k, \mathcal{D})_{j_i})$, *bautizamos como* $\text{reco}(\sigma_i, c_k, \mathcal{D})_{j_i}$ *al* documento original *y a sus homólogos en* $\{\text{reco}(\sigma_l, c_k, \mathcal{D})\}_{j_l \in J_l, l \neq i}$ *como los* documentos de referencia *denotados por* $\gamma(\text{reco}(\sigma_i, c_k, \mathcal{D})_{j_i})$.

■

Intuitivamente, dada una consulta y un cierto número de los documentos originales devueltos en las mejores posiciones por un determinado sistema RI en una determinada colección, su CR es la suma de las referencias proporcionadas por los otros sistemas. Esto inspira un método sencillo de ordenación al margen de la consideración de los JREL's y al que Wu *et al.* denominaron *método básico*.

Definición 6.36 *Sean* $\sigma = \{\sigma_i\}_{i \in I}$ *un conjunto de sistemas de* RI, $\mathcal{D}$ *una colección documental, y* $\mathcal{Q} = \{c_j\}_{j \in J}$ *un conjunto finito de tópicos (consultas). Denotamos por*

$$\text{CRM}(\sigma_i, \mathcal{Q}, \mathcal{D}) := \frac{\sum_{j \in J} \text{CR}(\sigma_i, c_j, \mathcal{D})}{|\mathcal{Q}|} \tag{6.49}$$

al contador de referencia media de σ_i en el conjunto de tópicos $\mathcal{Q}$ para la colección documental $\mathcal{D}$.

■

Intuitivamente, dado un sistema de RI, se calculan sus CRM's como el valor medio de los valores individuales de CR en cada consulta, lo que proporciona una técnica de ordenación fiable para sistemas de RI. Entre las mejoras propuestas por los autores de este método básico se optó por considerar la posición de relevancia de ambos, los documentos originales y los de referencia. Esto hace necesario ampliar la noción de CR con el fin de integrarlos.

Definición 6.37 *Sean* $\sigma = \{\sigma_i\}_{i \in I}$ *un conjunto de sistemas de* RI, $\mathcal{D}$ *una colección documental,* $\mathcal{Q} = \{c_j\}_{j \in J}$ *un conjunto finito de tópicos (consultas), y* $\{\varrho_{j_i}\}_{j_i \in J_i}$ *las puntuaciones normalizadas*[19] *asociadas a* $\{\text{reco}(\sigma_i, c_j, \mathcal{D})\}_{j_i \in J_i}$. *Sea también* $\forall\ m \in [1, \text{NumDocsMax} = 1.000],\ k \in [1, 4]$*:*

$$\hat{a}(\text{reco}(\sigma_i, c_j, \mathcal{D})_{j_i}) := \sum_{\text{reco}_{(\sigma_k, c_j, \mathcal{D})_{k_l}} \in \gamma(\text{reco}_{(\sigma_i, c_j, \mathcal{D})_{j_i}})} \Delta - l \ \ (\text{resp. } \varrho_{k_l})$$

[19] asumimos, sin pérdida de generalización, que estas puntuaciones están en el intervalo $[0, 1]$.

y

$$\omega_{j_i} := \begin{cases} \zeta(200) - \zeta(m-1)\ , & \text{si} \quad j_i = 5m \\ \omega_{5m} - \frac{1}{m} + \frac{5}{j_i}\ , & \text{si} \quad j_i = 5m-k \end{cases}$$

siendo $\hat{a}(\text{reco}(\sigma_i, c_j, \mathcal{D})_{j_i})$ *y* ω_{j_i} *las funciones de peso asociadas a la relevancia de las posiciones de referencia y a los documentos originales, respectivamente, definiéndose la función auxiliar* ζ *como*

$$\zeta(m) := \begin{cases} 0\ , & \text{si } m = 0 \\ 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{m}\ , & \text{en cualquier otro caso} \end{cases}$$

donde NumDocsMax *es el tamaño máximo de la colección documental* $\mathcal{D}$ *y* Δ *es un valor constante, que los autores establecen empíricamente en sus experimentos a 1.501'00. Denotamos a la expresión*

$$\sum_{j_i \in J_i} \omega_{j_i}.\hat{a}(\text{reco}(\sigma_i, c_j, \mathcal{D})_{j_i}) \tag{6.50}$$

como $\text{CRP}_{\text{o}}(\sigma_i, c_j, \mathcal{D})$ *(resp.* $\text{CRP}_{\text{v}}(\sigma_i, c_j, \mathcal{D})$*), al* contador de referencia ponderado basado en la ordenación *(resp.* basado en la puntuación*)* de σ_i sobre el tópico c_j para la colección $\mathcal{D}$.

■

Siguiendo el mismo proceso que se aplicó para introducir CRM a partir de los CR, ahora podemos introducir naturalmente la *media de contadores de referencia ponderados*, MCRP_{o} (*resp.* MCRP_{p}) de CRP_{o} (*resp.* CRP_{p}), que ofrece dos medidas adicionales de ordenación.

Sin embargo, algunas de las elecciones en esta propuesta de ordenación son difíciles de justificar, ya que no se han argumentado razones convincentes para presentar la constante Δ, ni los (muy complejos) valores de ω_{j_i}. Como las fórmulas resultantes son poco claras y difíciles de entender, se propone modificar ligeramente el planteamiento original, razón por la cual no lo abordaremos en esta sección, sino que lo ilustraremos en detalle más adelante.

6.4.5 | Selección del conjunto de tópicos

El objetivo ahora es seleccionar un conjunto de consultas minimal con el fin de evaluar nuestro sistema de RI comparándolo con una colección de las ya existentes, tomando como referencia los diferentes niveles de dificultad en su resolución por parte del usuario. En este sentido, a nuestro conocimiento no se ha presentado ni documentado, hasta ahora, ninguna técnica específica para este fin concreto; por lo que este enfoque específico constituye otra de las contribuciones de esta tesis, razón por la cual no lo abordaremos en esta sección, sino que lo ilustraremos en detalle más adelante.

PARTE III

Trabajo desarrollado

CAPÍTULO VII

Nivel léxico

Aunque nuestra propuesta no requiere de ningún entorno específico de análisis léxico, el esquema elegido a efectos de implementación está integrado en una cadena de PLN para su aplicación en el ámbito de la RI, en este caso para el francés. Concretamente, y tomando como referencia la Fig. 7.1, vamos a detallar los recursos[1] y herramientas[2] que se han utilizado. Se observa como existen tres pilares fundamentales. Por un lado, se dispone de un recurso denominado LEFFF [266], que no es otro que el lexicón sobre el que vamos a apoyarnos en esta fase. Por otro, al trabajar con una lengua rica en formas flexionadas, tal y como ocurre con el francés, el español o el alemán, el análisis léxico resulta especialmente complejo. Por este motivo, antes de proceder a su realización, es necesario asegurar una correcta segmentación en frases, así como una adecuada detección y marcado de sus palabras. Es lo que se conoce por *preprocesamiento*. Aquí echaremos mano de la herramienta SXPIPE [264], disponiendo a su vez del lexicón para identificar correctamente cada uno de estos componentes en las frases. Finalmente, la aplicación FRMG LEXER realizará la función de analizador morfológico y será la responsable de llamar al preprocesador, a la vez que hará uso del lexicón.

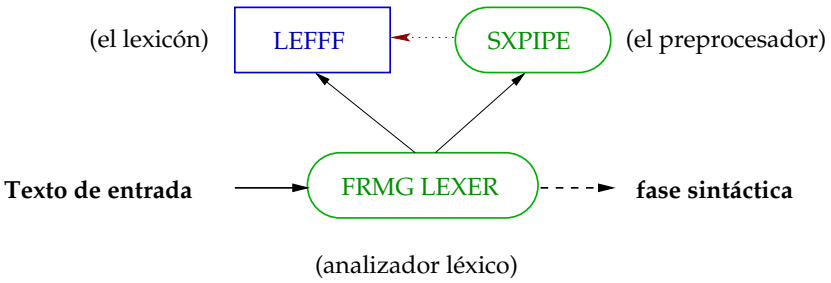


Figura 7.1: Esquema de la cadena utilizada a nivel léxico

[1] se representan en la figura mediante la forma cuadrada.
[2] se representan en la figura mediante la forma ovalada.

7.1 | Recurso léxico: el LEFFF

Todo análisis léxico pasa forzosamente por comprobar la pertenencia de cada una de sus palabras a un diccionario [245]. En este contexto, la estructuración y desarrollo de un recurso de este tipo resulta esencial. Por un lado, un lexicón requiere de una amplia cobertura, lo que implica una gran cantidad de entradas. Por el otro, por cada una de estas entradas es necesario disponer de información asociada adicional, tanto de tipo morfológica, como sintáctica. Facilitar estas tareas supone automatizar en lo posible procesos y asegurar su completud y corrección.

Ejemplo 7.1 *Consideremos las palabras francesas* «rapidement» (`«rápidamente»`) *y* «probablement» (`«probablemente»`). *Ambas presentan cierta similitud ya que evidencian un mismo modelo de derivación: una raíz al que se le añade el sufijo* «-ment» (`«-mente»`).

Sin embargo, cuando prestamos atención a su comportamiento en el contexto en el que se encuentran, las diferencias se hacen evidentes [211, 212, 213]:

- *La palabra* «rapidement» (`«rápidamente»`) *se usa:*
  - *En posición antepuesta y desligada del verbo. Generalmente tiene por función caracterizar el espacio de tiempo que transcurre en el acontecimiento descrito por la frase:* «rapidement, il observa l'exocarpe» (`«rapidamente, observó el exocarpio»`).
  - *En posición pospuesta con respecto al verbo. Si el verbo es compatible con la noción de velocidad, caracteriza el modo en el que se realiza la acción:* «l'exocarpe se forme rapidement» (`«el exocarpio se forma rápidamente»`). *Si es incompatible, caracteriza el espacio de tiempo transcurrido en el acontecimiento descrito por la frase:* «la pluie se mit a tomber rapidement» (`«empezó a llover rápidamente»`).
  - *En posición pospuesta al adverbio de negación «pas»:* «l'exocarpe ne se formera pas rapidement» (`«el exocarpio no se formará rápidamente»`).
- *La palabra* «probablement» (`«probablemente»`) *se usa:*
  - *En posición antepuesta y desligada del verbo. Generalmente tiene por función formular una duda sobre la frase:* «probablement, Jean a raison» (`«probablemente, Juan tiene razón»`).
  - *En posición pospuesta con respecto al verbo, cuando éste es compatible con la noción de veracidad:* «il se trompe probablement» (`«se equivoca probablemente»`).
  - *En posición antepuesta a la conjunción de subordinación «que» seguido de la frase:* «Probablement que Jean sera là» (`«Probablemente que Juan estará aquí»`).

- *En posición antepuesta al adverbio de negación «pas»:* «l'exocarpe ne se formera probablement pas» (`«el exocarpio probablemente no se formará»`).

Estudiando lo expuesto, se comprueba como en esta clase se incluyen unidades diversas por su uso y tipo de significado. De este modo, será necesario, además de proporcionar información morfológica, disponer de información de otra índole.

■

Estos objetivos son la base del trabajo desarrollado en torno al formalismo denominado *Alexina*[3] [66] y a la provisión de un lexicón morfológico y sintáctico de amplia cobertura, denominado *Lexicón Francés de Formas Flexionadas*[4] (LEFFF) [92, 266]. La arquitectura del LEFFF se basa en una jerarquía con herencia de propiedades, que lo hace más compacto y fácil de mantener. Además, permite una descripción de las entradas léxicas, independiente de los formalismos gramaticales en los que se use *a posteriori*. De un modo más específico, puede ser usado directamente en aplicaciones de PLN de alto nivel, especialmente en aquéllas que requieren un análisis sintáctico profundo. En este sentido, es independiente del idioma a tratar, lo que justifica su elección para nuestro trabajo. Se basa en dos niveles de representación motivados lingüísticamente, que separa la descripción en sí misma, del diccionario que usa. Así, el lexicón LEFFF se construye siguiendo dos fases a partir de las informaciones elementales factorizadas [263, 266], que pasamos a describir a continuación.

7.1.1 | Representación intensional

Una representación *intensional* es una representación comprimida o factorizada del contenido del lexicón. Cada entrada de este formato se corresponde a un lema acompañado de toda la información morfológica y sintáctica necesaria para crear la familia de formas asociada al lema.

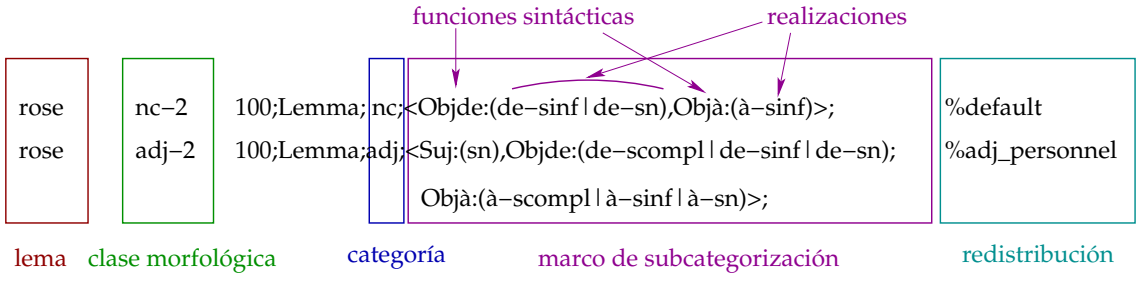


Figura 7.2: Ejemplo de entrada intensional en el LEFFF

[3] en terminología francesa, *Atelier pour les LEXiques INformatiques et leur Acquisition*.

[4] en terminología francesa, *Léxique Français de Forme Fléchies*. Se distribuye bajo licencia LGPL-LR. Ver el enlace `http://alpage.inria.fr/~sagot/lefff.html`.

Concretamente, posee la información que resumimos en la Fig.7.2, usando como entrada de ejemplo el lema en francés «*rose*» («`rosa`»). Una de esas informaciones es la clase morfológica, que indica el patrón seguido para crear sus formas flexionadas [265]. En el primer caso, pertenece a la clase *nc-2* y en la segunda a *adj-2*. Además de la clase morfológica, también posee una categoría léxica que muestra que en el primer caso hace referencia a un sustantivo, y en el otro a un adjetivo. Incluye igualmente un marco de subcategorización explícito señalando como usar el lema en una construcción sintáctica. Es decir, propone una lista de *funciones sintácticas* de los posibles argumentos que puede poseer el lema, así como cada una de las posibles *realizaciones*[5] que se les puede atribuir, esto es, de como se puede utilizar.

En este sentido, las funciones sintácticas que se suelen utilizar en el lexicón LEFFF son las siguientes. En primer lugar, *Suj* para sujeto. *Obj* se usará para complementos directos. En el caso de complementos indirectos, se dispone de *Objà* y *Objde* en función de si son introducidos por las preposiciones «*à*» o «*de*» respectivamente. Para indicar los complementos de lugar se usará *Loc* como en los casos de «*là*» («`ahí`») e «*ici*» («`aquí`»). Cuando éstos estén introducidos por una preposición o implícitamente incluidos en ella, la función sintáctica a emplear será *DLoc*, como en «*de là*» («`de ahí`») y «*d'ici*» («`de aquí`»). Para atributos[6] se establece *Att*, y finalmente *Obl* para casos oblicuos[7]. Además, las realizaciones que pueden usarse son de tres tipos:

- Pronombres clíticos «*cln*», «*cla*» y «*cld*» para los casos nominativo, acusativo y dativo.
- Sintagmas directos «*sn*», «*sinf*», «*scompl*», «*sa*» y «*qcompl*» para los sintagmas nominal, infinitivo, completivo, adjetival y preguntas indirectas.
- Sintagmas preposicionales que se construyen de la forma «*prep-real*», donde «*prep*» es una preposición y «*real*» una realización sintagmática directa.

[5] esto permite, por ejemplo, representar correctamente marcos de subcategorización donde dos funciones gramaticales idénticas pueden coexistir. Es el caso de dos complementos indirectos introducidos por la misma preposición. Un ejemplo sería «*La taille des feuilles a été divisée par deux par l'Évolution*» («`El tamaño de las hojas se ha dividido por dos por la Evolución`»).

[6] es la construcción de dos elementos gramaticales unidos, donde el segundo especifica al primero. Es lo que se llama epíteto. Por ejemplo, se pueden emplear adjetivos para cualificar a sustantivos, como en «*la feuille verte*» («`la hoja verde`»), pero también en el caso de verbos sustantivados, como en «*un étudiant brillant est celui qui a un savoir avantageux*» («`un estudiante brillante es aquél que tiene un saber beneficioso`»).

[7] es un caso gramatical que se emplea normalmente en un sustantivo o pronombre que no es el sujeto de la oración. Es lo que hace que los sintagmas adverbiales y circunstanciales sean ascendidos a la posición de objetos o sujetos, siendo entonces marcados como tal. Si pensamos en una frase como «*la fleur porte des étamines*» («`la flor sostiene estambres`»), se podría transformar en voz pasiva a «*les étamines sont portées par la fleur*» («`los estambres son sostenidos por la flor`»). El objeto directo inicial se transforma en el sujeto pasivo y en un agente opcional, es decir, se añade «*par la fleur*» («`por la flor`»).

Tomando como referencia la Fig.7.2, las realizaciones dispuestas entre paréntesis son opcionales [92]. Así, por ejemplo, en la frase «*Les fleurs de couleur rose*» («`Las flores de color rosa`») «*rose*» («`rosa`») tiene función de adjetivo, y posee un «*Suj*» y un «*Objde*» cuya realización es «*de-sn*», siendo ésta no obligatoria. Finalmente, existe un último elemento denominado *redistribución* que indica el tipo morfosintáctico utilizado. Por ejemplo, en el caso de tratar con entradas verbales, representa el tipo de voz o el tipo de verbo empleado. Éstos vienen introducidos por las macros « %» y pueden ser, por ejemplo, « *%default*», « *%active*», « *%passive*», « *%impersonal active*» o « *%infinitive*». Esto es, una vez construidas las entradas extensionales a partir del lema, se pueden realizar las transformaciones adecuadas sobre ellas de tal manera que sea posible operar en la estructura sintáctica de base[8].

7.1.2 | Representación extensional

Una representación *extensional* es aquélla que se genera automáticamente en una segunda fase a partir de la compilación del lexicón intensional. Por cada entrada intensional, la extensional asocia todas las posibles formas flexionadas con toda su información morfológica y sintáctica. Por ejemplo, cada entrada en el lexicón extensional constará de la etiqueta morfológica y del marco de subcategorización de la correspondiente redistribución.

La encargada de realizar dicha compilación es la herramienta denominada ALEXINA-TOOLS [92], que a su vez sirve para modelar y adquirir léxico [66, 262] usando el formalismo descrito. Más concretamente, tal y como muestra la Fig. 7.3, ésta recibe la representación intensional del lexicón y, a partir de ahí, la compila y construye todas las palabras pertenecientes a la familia de cada lema, usando para ello su clase morfológica.

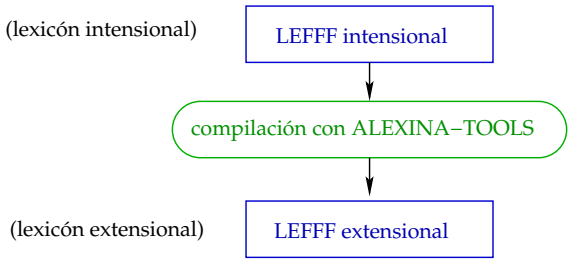


Figura 7.3: Proceso de compilación del LEFFF intensional en LEFFF extensional

Estas clases morfológicas están definidas en el formato descrito en [262], que cubre la mayor parte de las entradas del lexicón. Tan sólo los lemas que se flexionan de una forma especial, por ser irregulares, se describen de forma manual en un fichero adicional. De

[8] así, por ejemplo, al poner como redistribución « *%infinitive*», lo que se está indicando es que se puede omitir el sujeto en el marco de subcategorización.

este modo, el resultado obtenido, es decir, las entradas extensionales generadas a partir de una intensional, se ilustra en la Fig.7.4 que pasamos a describir.

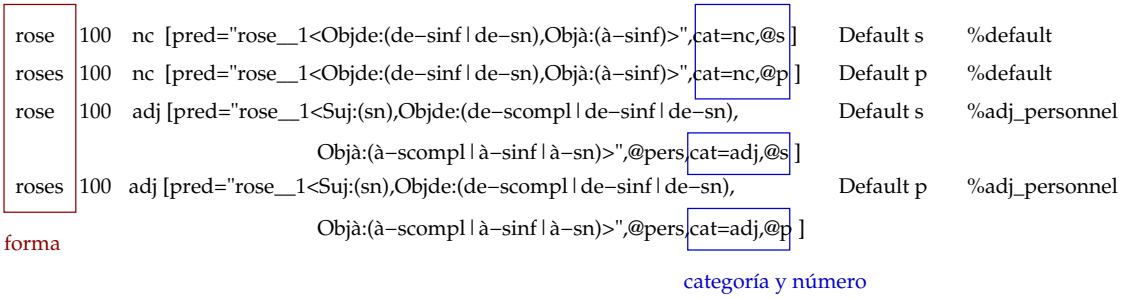


Figura 7.4: Ejemplo de entrada extensional en el LEFFF

Para las dos entradas intensionales de la Fig. 7.2, se han generado cuatro extensionales. Las dos primeras hacen referencia a la primera intensional del anterior ejemplo, donde «*rose*» («`rosa`») tiene por categoría léxica la de sustantivo. Así, en el primer caso, se trata de un sustantivo singular, denotado por *@s*, y en el segundo de uno plural, denotado por *@p*. Las dos siguientes hacen referencia a la intensional referida al adjetivo y, al igual que antes, también muestran cual es su singular y cual su plural.

7.1.3 | Construcción del lexicón LEFFF

La construcción de un lexicón con amplia cobertura es, en cualquier circunstancia, un trabajo difícil debido al gran número de entradas necesarias y a la complejidad de las informaciones que se deben asociar a cada una de ellas para asegurar su calidad. En este sentido, la arquitectura mostrada asegura una factorización importante de las informaciones que permiten generar el lexicón LEFFF. A pesar de esto, éstas han de ser obtenidas de algún modo y, seguidamente, completadas y/o corregidas. Más concretamente, el proceso considerado es el siguiente:

- Adquisición automática de las entradas morfológicas de categorías léxicas, aunque complementado por una validación manual. Permite añadir palabras que no se encuentran en recursos clásicos, como por ejemplo las derivadas[9] y las contemporáneas[10] [66].
- Aplicación de algoritmos de detección de errores de entradas morfológicas en resultados de análisis sintácticos [267], o corrección guiada mediante técnicas automáticas, como las estadísticas sobre *corpus* etiquetados [210].
- Aplicación de un corrector ortográfico, denominado SXSPELL [262, 264]. Éste detecta palabras desconocidas en el lexicón y propone posibles correcciones. Las

[9] es el caso de las palabras con prefijos.
[10] es el caso de las palabras técnicas.

técnicas empleadas se basan en reglas de reescritura que pueden ser dependientes del contexto. Tras su aplicación, se genera de un modo automático una lista de formas flexionadas desconocidas previamente en el lexicón que se está construyendo.

7.1.4 | Enriquecimiento del lexicón LEFFF

Parece razonable disponer de léxicos asociados al dominio de conocimiento. En el caso de la botánica, éste se desarrolló tomando como punto de partida el lexicón LEFFF, posteriormente enriquecido con el *corpus* de la «*Polynésie Française*»[11] [196] y después con el de la «*Flore du Cameroun*».

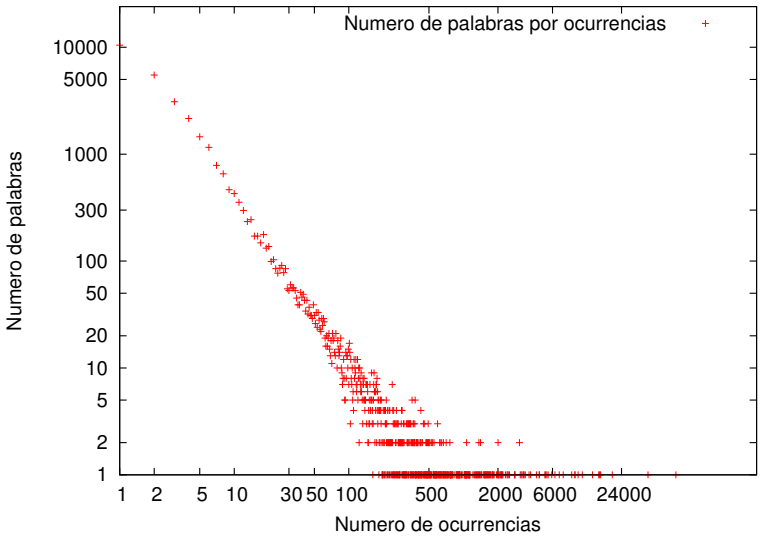


Figura 7.5: Frecuencia de aparición de palabras en el *corpus*

Los tests realizados pusieron en evidencia la existencia de palabras que no se habían extraído y de errores[12] en entradas del LEFFF enriquecido [267]. Así, las palabras correctamente escritas poseen frecuencias generalmente elevadas, mientras que las de las erróneas suelen ser inferiores o igual a dos. Además se evidenciaron otras particularidades como, por ejemplo, que estos textos disponen de gran cantidad de palabras en latín, inglés y dialectos del propio Camerún.

En este sentido, la Fig. 7.5 representa la cantidad de palabras existentes en el *corpus* en función de sus ocurrencias. A modo de ejemplo, la palabra «*nervures*» posee 2.160 ocurrencias, mientras que «*Feuilles*» tiene 2.448. De la misma manera, existen 10.467 y 5.506 palabras diferentes que aparecen una y dos veces respectivamente en todo el *corpus*. Éstas son, por ejemplo, para el primer caso, las palabras «*ellip'tque*», «*logitudinales*» y

[11] se trata de un trabajo anterior en el que se utilizaron las palabras claves recuperadas a partir de él.
[12] algunos de ellos se deben a erratas en los textos originales.

«*dVifrique*»; y para el segundo, «*l'endocarps*» y «*ros'ées*». Todas ellas poseen de un modo u otro errores tipográficos.

7.2 | **Preprocesamiento:** SXPIPE

El preprocesamiento se lleva a cabo mediante la herramienta SXPIPE. Consiste en una aplicación secuencial de diferentes módulos, centrados en la correcta identificación de palabras y frases que constituyen las unidades fundamentales sobre las que trabajarán las fases posteriores, tales como etiquetadores, analizadores sintácticos o sistemas de RI. Además también permiten tratar diversos fenómenos lingüísticos que ocurren en un idioma, como por ejemplo, el *reconocimiento de contracciones* o el REN, así como cubrir un determinado ámbito de aplicación.

Dicho esto, estamos en disposición de pasar a describir la arquitectura de SXPIPE descomponiéndola en cinco etapas, tal y como se observa en la Fig.7.6. El punto de partida son los documentos sobre los que se ha realizado un proceso de selección de las descripciones botánicas y que se han convertido a texto plano. En el Apéndice B, se detallan las transformaciones llevadas a cabo en la digitalización de los casi 40 volúmenes hasta la obtención de un fichero XML por cada familia, género o especie presente en los textos.

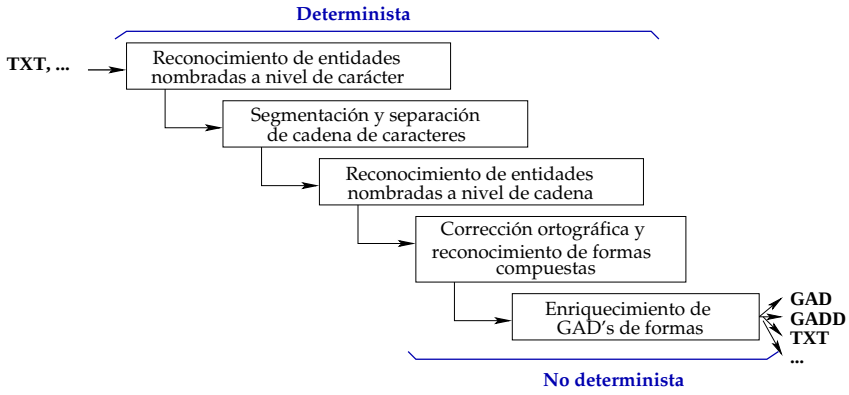


Figura 7.6: Arquitectura de SXPIPE

En este sentido, hay que destacar que las fases de *segmentación* y *separación de cadenas de caracteres*, y de *corrección ortográfica* y *reconocimiento de formas compuestas* modifican el formato del *corpus*. Concretamente, la fase de segmentación y separación convierten el texto en una secuencia de cadenas, y la de corrección ortográfica y reconocimiento de formas compuestas transforma el flujo de esta sucesión en un *grafo acíclico dirigido* (GAD) de formas, que detallaremos llegado el momento. Además, las fases de REN a nivel de carácter, y de cadena, así como de enriquecimiento de los GAD's de formas son modulares, es decir, están compuestas de varios módulos que se pueden activar o no. A continuación, vamos a explicar con más detalle cada una de ellas.

7.2.1 | REN a nivel de carácter

Los *corpus* pueden incluir secuencias de caracteres que no son analizables ni morfológica ni sintácticamente, y que debemos identificar [202]. Estas secuencias son generalmente entidades nombradas. Pero, entre ellas, algunas contienen caracteres que se identifican como signos de puntuación, como por ejemplo «.» o «,». Por este motivo, se requiere de una fase de reconocimiento cuya base descriptiva es un conjunto de gramáticas locales [264], cada una de las cuales asume la modelización de un pequeño conjunto de entidades nombradas. Estas gramáticas deben ser aplicadas antes de la segmentación y de la separación de cadenas de caracteres. Así, por ejemplo, se reconocen fenómenos tales como:

- *Direcciones url*, con detección de numerosos casos de error y numerosos formatos. Se representa por «_URL».
- *Fechas*, en diferentes formatos, así como intervalos de fechas, representadas por «_DATE_arto», «_DATE_artf» y «_DATE_year», que permiten diferenciar los comportamientos sintácticos.
- *Números de teléfono* en diversos formatos y representados por «_TEL».
- *Horarios* en diversos formatos y representados por «_HEURE».
- *Direcciones* en diversos formatos y representados por «_ADRESSE».

También, se ha tratado de representar fenómenos particulares que aparecían en el *corpus* $\mathscr{B}$, tales como las dimensiones y sus intervalos, representados por la etiqueta «_DIMENSION», pero también las cantidades y sus intervalos, representados por «_NUMBER».

7.2.2 | Segmentación y separación de cadenas de caracteres

La funcionalidad de la separación de cadenas de caracteres consiste en identificar y separar los diferentes componentes presentes en el texto, utilizando para ello delimitadores, tales como espacios o algún tipo de marca tipográfica, como los signos de puntuación. Por otro lado, la segmentación consiste en la descomposición del texto en frases, una tarea más compleja de lo que pudiera parecer *a priori*, justamente por los problemas que plantean estas marcas[13].

Por este motivo, una vez realizado el REN [264], se aplican un conjunto de expresiones regulares, extendiendo las ideas propuestas por Grefenstette y Tapanainen [119], con el fin de realizar una correcta separación de cadenas de caracteres. Así, lo primero que

[13]normalmente, se utilizan para indicar los finales de frase, aunque también son necesarias en abreviaturas, como en la palabra «*etc.*», acrónimos, fechas o dimensiones [287].

se hace es compactar los delimitadores, por ejemplo, eliminando los múltiples espacios en blanco existentes. Una vez hecho esto, considera cada una de las cadenas obtenidas individualmente, centrándose en aquéllas que se componen de un carácter «.». A su vez, se verifica su existencia en el lexicón LEFFF. Si existen se tratarán como abreviaturas. En caso contrario, se considerará que el carácter «.» es un signo de puntuación que delimita la frase, permitiendo realizar la segmentación.

7.2.3 | REN a nivel de cadenas

Una vez delimitadas las cadenas, se identifican aquéllas que no pueden ser analizadas debido a su ausencia en el lexicón LEFFF. A cada una de ellas se le asigna una etiqueta que proporciona información acerca de su modo de representación, tomando en consideración los datos suplementarios que se pueden extraer a partir del corrector ortográfico[14], que detallaremos en el siguiente apartado. Así, por ejemplo, si consideramos la cadena «*Linné*», la etiqueta que se establezca indicará que posee una inicial en mayúscula, que puede estar en francés, pero también en otro idioma extranjero.

Una vez identificadas dichas cadenas o secuencias de cadenas, se aplican un cierto número de expresiones regulares que actúan sobre la información presente en las etiquetas. De este modo, se reconocen las entidades nombradas, como por ejemplo los acrónimos identificados por «_NP_WITH_INITIALS», los nombres propios representados por «_NP»; o incluso secuencias en lengua extranjera con «_ETR». Concretamente, en el ámbito botánico, es necesario reconocer entidades como nombres propios, en el caso de los autores de los volúmenes de la colección o en el de los descubridores de un género o especie. En este sentido, se ha incluido un módulo capaz de reconocer nombres científicos, cuya etiqueta es «_SCIENTIFIC_NAMES».

7.2.4 | GAD's de formas

Las fases de corrección ortográfica y reconocimiento de formas compuestas pueden producir diferentes preprocesados susceptibles de conservarse en paralelo para su posterior análisis léxico, de modo que no se descarte ninguna interpretación. Para mostrarlos, es esencial usar una representación adecuada. En este sentido, la elegida consiste en el empleo de GAD's bajo una forma de expresiones regulares o de una lista de transiciones denominada GAD *desplegado*[15] (GADD).

Ejemplo 7.2 *Consideremos la frase francesa* «Feuilles à nervures denticulées» (`«Hojas con nervaduras dentadas»`). *El* GADD *asociado a ésta se puede ver en la Fig. 7.7.*

Cada uno de los nodos del GADD *representan los estados por los que transitan las formas. Del mismo modo, las aristas son las encargadas de representarlas y mostrar el orden en*

[14] no se realiza ninguna corrección, simplemente se obtiene información complementaria.
[15] en terminología anglosajona *unfolded* DAG, (UGAD).

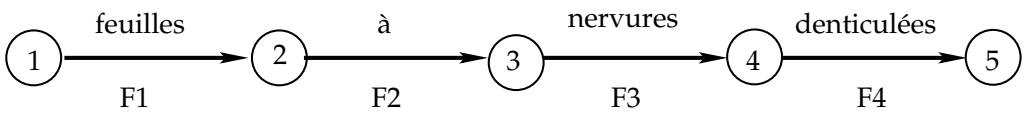


Figura 7.7: GADD asociado a la frase «*Feuilles à nervures denticulées*»

el que se van sucediendo. Así, por ejemplo, «feuilles» (`«hojas»`) *se encuentra entre el estado 1 y el 2, ocupando la primera posición, la cual se representa mediante* $F1$*. Lo mismo ocurre con las demás formas. Como se ve, en este ejemplo, no existe ningún tipo de ambigüedad.*

■

Ejemplo 7.3 *Consideremos la frase francesa* «Les carpelles du pistil» (`«Los carpelos del pistilo»`). *La cadena* «du» *es una fusión de varias formas, concretamente de la preposición* «de» *y del determinante* «le»*, formando el artículo definido. También se puede usar como un artículo indefinido. Por este motivo se representa como se indica en la Fig. 7.8, donde* «de» *y* «le»*, o* «du» *se encuentran en la tercera posición en la frase, es decir,* $F3$*, formando una* amalgama.

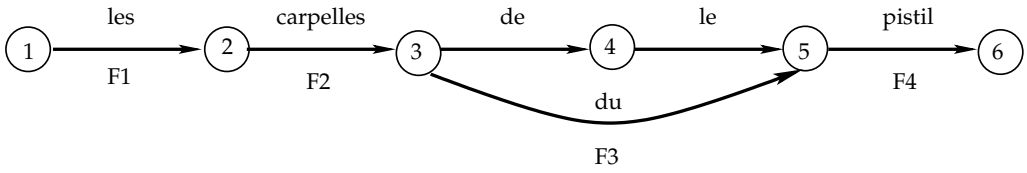


Figura 7.8: GADD asociado a la frase «*les carpelles du pistil*»

■

Ejemplo 7.4 *Basándonos en un ejemplo de [264], consideremos la frase* «Pomme de terre cuite» (`«Patatas cocidas»`). *En ella las cadenas* «Pommes» (`«manzana»`), «de» (`«de»`) *y* «terre» (`«tierra»`) *se pueden considerar de dos maneras diferentes. La primera de ellas consiste en una única forma compuesta* «pomme_de_terre» (`«patatas»`)*, en cambio la segunda consta de formas totalmente independientes. Ambas pueden observarse en la Fig. 7.9. Si se consideran como formas independientes, cada una ocupa una posición, pero si lo hacemos como una compuesta, ocupan las posiciones* $F1$*,* $F2$ *y* $F3$*.*

Lo mismo ocurre con las cadenas «terre» (`«tierra»`) *y* «cuite» (`«cocida»`)*. Si se considera que éstas dan lugar a formas independientes, entonces cada una ocupará una posición. En cambio si da lugar a la forma compuesta* «terre_cuite» (`«barro»`)*, ocupará las posiciones señaladas como* $F3$ *y* $F4$*.*

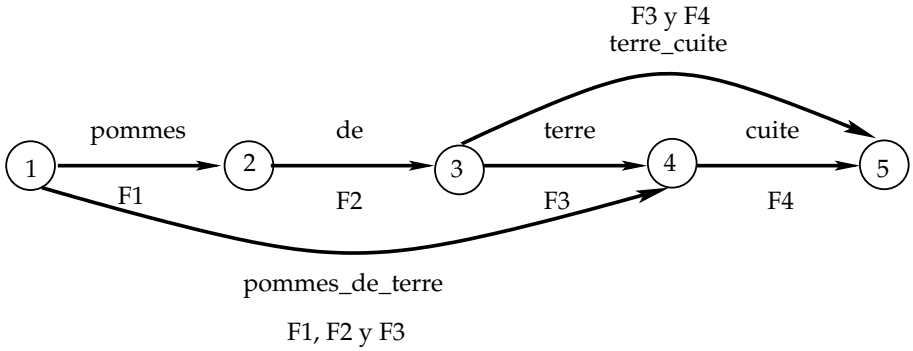


Figura 7.9: GADD asociado a la frase «*Pommes de terre cuite*»

■

Ejemplo 7.5 *Supongamos que queremos representar la frase francesa* «Stipules linéaires, 6 mm;» (`«Estípulas lineales, 6 mm;»`). *En ella las cadenas* «6» *y* «mm» *son reconocidos como una entidad nombrada dando lugar a una forma especial de tipo* «_DIMENSION». *En este sentido, las formas especiales están incluidas en una misma transición a pesar de ocupar posiciones diferentes. Observemos la Fig. 7.10. En ella se ve como* «6 mm» *ocupa las posiciones* $F4$ *y* $F5$.

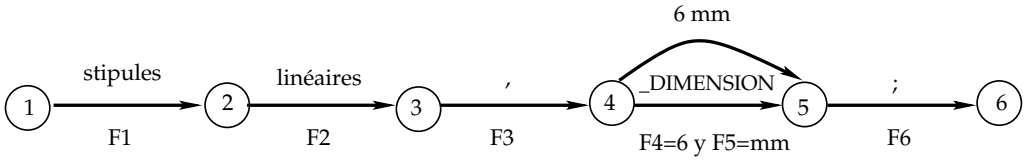


Figura 7.10: GADD asociado a la frase «*Stipules linéaires, 6 mm;*»

■

Para representar los GAD's se emplea una notación XML. De este modo, a partir de la identificación de cada una de las cadenas delimitadas por separadores y para todas las fases del proceso, aquéllas que conforman la secuencia de entrada se conservan entre *comentarios*[16], precediendo a la forma asociada. Cada una se representa mediante $\{< F \;\; id = "E_iF_j" > cadena < /F >\}$, donde i es el número de la frase y j es su posición en la misma. Al resultado de crear un GAD en formato XML lo vamos a denominar GAD-XML.

Ejemplo 7.6 *Consideremos el resultado de la aplicación de* SXPIPE *sobre la frase francesa* «Les carpelles du pistil» (`«Los carpelos del pistilo»`) *del Ejemplo 7.3, cuyo formato es el* GAD-XML *representado en la Fig. 7.11.*

[16]es decir, entre llaves y completados por la posición que ocupan en la secuencia de entrada inicial.

```
{<F id="E1F1">Les</F>} les {<F id="E1F2">carpelles </F>} carpelles
({<F id="E1F3">du</F>} du | {<F id="E1F3">du</F>} de__prep {<F id="E1F3">du</F>}
le__det) {<F id="E1F4">pistil</F>} pistil
```

Figura 7.11: GAD-XML para la frase «*Les carpelles du pistil*»

En la figura se ve como la primera cadena obtenida tras la separación «Les» *se encuentra en la primera posición* $(F1)$ *de la primera frase* $(E1)$*, identificándolo con la forma simple* «les»*. Pero además, en la posición* $F3$ *puede existir una cierta ambigüedad que se representa con la ayuda de paréntesis y separando las alternativas mediante el símbolo* «|»*. En este sentido, puede existir una forma simple* «du» *que hace referencia al artículo indefinido, o bien a una amalgama cuyas formas fusionadas son* «de_prep» *y* «le_det»*, representando al artículo definido.*

■

Ejemplo 7.7 *Consideremos el resultado de la aplicación de* SXPIPE *sobre la frase en francés* «Stipules linéaires, 6 mm;» (`«Estípulas lineales, 6 mm;»`) *del Ejemplo 7.5, cuyo formato es el* GAD-XML *de la Fig. 7.12. Este ejemplo es sencillo ya que no aparece ningún tipo de ambigüedad, pero lo consideramos de interés para facilitar la comprensión del tratamiento de* REN *llevado a cabo.*

```
{<F id="E1F1">Stipules</F>} stipules {<F id="E1F2">linéaires </F>} linéaires
{<F id="E1F3">,</F>} , {<F id="E1F4">6</F> <F id="E1F5">mm</F>} _DIMENSION
{<F id="E1F6">;</F>} ;
```

Figura 7.12: GAD-XML para la frase «*Stipules linéaires, 6 mm;*»

En dicha figura se ve como las cadenas cuarta y quinta, es decir, «6» *y* «mm» *son los elementos que componen una forma especial. La entidad nombrada que representan no es otra que una dimensión, de ahí que se represente por _DIMENSION.*

■

Ejemplo 7.8 *Consideremos ahora el resultado de la aplicación de* SXPIPE *sobre la frase en francés* «Pommes de terre cuite» (`«Patatas cocidas»`)*, cuyo formato es el* GAD-XML *de la Fig. 7.13 y que incluye ambigüedades léxicas.*

La primera interpretación se refiere a la posibilidad de la existencia de dos formas simples y de una compuesta, es decir, «pommes», «de» *y* «terre_cuite»*, cuyo significado sería* «`manzanas de barro`»*. La segunda se refiere a la existencia de una forma compuesta y de una simple, es decir,* «pommes_de_terre» *y* «cuite»*, que se traduciría por* «`patatas cocidas`»*. La última interpretación supone la existencia de cuatro formas simples, es decir,* «pommes», «de», «terre» *y* «cuite»*, cuyo significado sería* «`manzanas de tierra cocida`».

```
{<F id="E1F1">Pomme</F>} pomme {<F id="E1F2">de</F>} de
{<F id="E1F3">terre</F> <F id="E1F4">cuite</F>} terre_cuite |
({<F id="E1F1">Pomme</F> <F id="E1F2">de</F> <F id="E1F3">terre</F>} pomme_de_terre |
{<F id="E1F1">Pomme</F>} pomme {<F id="E1F2">de</F>} de
{<F id="E1F3">terre</F>} terre) {<F id="E1F4">cuite</F>} cuite
```

Figura 7.13: GAD-XML para la frase «*Pommes de terre cuite*»

■

Si se decidiera mostrar estos GAD's con formato XML en forma de transiciones, es decir, indicando el camino a seguir dentro del grafo, sería necesario plasmarlos mediante GADD's, tal y como vimos en las Figs. 7.7, 7.8, 7.9 y 7.10, usando el formato correspondiente. Al resultado de crear un GADD en formato XML lo denominaremos GADD-XML. Se trata de un GAD donde, además de mostrar la información asociada a la forma, también se muestra la asociada a la transición en cuestión, todo ello bajo un formato XML. Por tanto, cada transición estará constituida por su estado inicial, su estado final y la información asociada a la forma.

Ejemplo 7.9 *Supongamos que queremos desplegar el* GAD *del Ejemplo 7.4. El resultado sería el mostrado en la Fig. 7.14.*

```
##DAG BEGIN
1  {<F id="E1F1">Pomme</F>} "pomme"  2
1  {<F id="E1F1">Pomme</F> <F id="E1F2">de</F>
    <F id="E1F3">terre</F>} "pomme_de_terre"  4
2  {<F id="E1F2">de</F>} "de"  3
3  {<F id="E1F3">terre</F>} "terre"  4
3  {<F id="E1F3">terre</F> <F id="E1F4">cuite</F>} "terre_cuite"  5
4  {<F id="E1F4">cuite</F>} "cuite"  5
##DAG END
```

Figura 7.14: GADD-XML para la frase «*Pommes de terre cuite*»

Aquí se observa como la forma compuesta «pommes_de_terre» *va del estado 1 al estado final 4. Lo mismo ocurre si consideramos la forma compuesta* «terre_cuite» *que va del estado 3 al 5.*

■

Ejemplo 7.10 *Supongamos que queremos desplegar el* GAD *del Ejemplo 7.7. El resultado será el* GADD-XML *dispuesto en la Fig. 7.15.*

En esta ocasión se observa como la primera cadena separada mediante delimitadores va del estado inicial 1 al final 2. Si nos centramos en la transición referente a la forma «_DIMENSION», vemos como ésta está formada por «6» *y* «mm»*, dando lugar a una*

```
##DAG BEGIN
1  {<F id="E1F1">Stipules</F>} "stipules"  2
2  {<F id="E1F2">linéaires</F>} "linéaires"  3
3  {<F id="E1F3">,</F>} ","      4
4  {<F id="E1F4">6</F> <F id="E1F5">mm</F>} "_DIMENSION"  5
5  {<F id="E1F6">;</F>} ";"  6
##DAG END
```

Figura 7.15: GADD-XML para la frase «*Stipules linéaires, 6 mm;*»

forma especial denominada entidad nombrada*. Sin embargo sólo forma una transición, aquélla que va del estado 4 al 5.*

■

Ejemplo 7.11 *Supongamos que queremos desplegar el* GAD *del Ejemplo 7.6. El resultado será un* GADD-XML *como el que se puede ver en la Fig. 7.16. En esta figura se observa como la tercera cadena* «du» *va del estado inicial 3 (es el elemento que se encuentra en la primera columna) al estado final 5 (es el elemento de la última columna), donde la transición del estado 3 al 4 representa la preposición* «de»*, y la transición del estado 4 al 5 representa el determinante* «le».

```
##DAG BEGIN
1  {<F id="E1F1">Les</F>} "les"  2
2  {<F id="E1F2">carpelles</F>} "carpelles"  3
3  {<F id="E1F3">du</F>} "du"  5
3  {<F id="E1F3">du</F>} "de__prep"  4
4  {<F id="E1F3">du</F>} "le__det"  5
5  {<F id="E1F4">pistil</F>} "pistil"  6
##DAG END
```

Figura 7.16: GADD-XML para la frase «*Les carpelles du pistil*»

■

7.2.5 | Corrección ortográfica y reconocimiento de formas compuestas

La corrección ortográfica corre a cargo de SXSPELL [262, 264]. Se trata de proporcionar mecanismos que, además de advertir de la presencia de un error, ofrecen una lista de posibles correcciones [229]. Este tipo de tratamiento se hace necesario ya que el *corpus* $\mathscr{B}$ empleado tiene una tasa elevada de errores ortográficos, incluso producidos en una fase previa de *reconocimiento óptico de caracteres* (OCR[17]). Además, si no se corrigen, estas palabras se convierten en desconocidas para las herramientas que hacen

[17] son las siglas de *Optical Character Recognition*.

uso de las salidas del preprocesador SXPIPE, como por ejemplo FRMG LEXER. Un ejemplo concreto podría ser la palabra «*ieuille*».

La situación se complica cuando se quiere gestionar aquellas cadenas separadas por delimitadores que son el resultado de la acumulación de varias formas, y que incluyen uno o varios errores ortográficos. La experiencia muestra que la única manera factible de tratar estos problemas es la de hacerlo simultáneamente, preservando el no determinismo, mientras no se disponga de informaciones que permitan eliminarlo. La idea es guardar la o las cadenas de partida (posiblemente mal ortografiadas) entre comentarios y producir una o varias formas corregidas. De este modo por ejemplo, la frase «*ieuilles avecpoints*» («`hojas con puntos`») se convertirá utilizando el preprocesador SXPIPE en la estructura que se muestra en la Fig. 7.17, donde se observa como «*ieuilles*» se puede corregir de dos maneras. O bien puede ser el verbo «*vouloir*» («`desear`») con «*veuilles*» o la opción correcta «*feuilles*» («`hojas`»).

```
({<F id="E1F1">ieuilles</F>} veuilles | {<F id="E1F1">ieuilles</F>} feuilles)
{<F id="E1F2">avecpoints</F>} avec {<F id="E1F2">avecpoints</F>} points
```

Figura 7.17: GAD con correcciones ortográficas para la frase «*ieuilles avecpoints*».

En cambio, «*avecpoints*» no da lugar a dudas. Es un error ortográfico que hace referencia a dos formas: la primera «*avec*» («`con`») y la segunda «*points*» («`puntos`»). Globalmente, el proceso funciona como pasamos a describir:

- En un primer momento se descomponen y/o unen las cadenas separadas por delimitadores para generar formas simples o compuestas, eventualmente a corregir. Para ello, se simulan correcciones intentando comprobar si la concatenación de las cadenas reconocidas como expresiones en lengua extranjera se pueden convertir a una/s forma/s concreta/s. Es el caso, por ejemplo, cuando existe un espacio en medio de una forma, como en «*feui lles*», que se ha reconocido como «_ETR». Eliminando ese espacio, se convierte en «*feuilles*».
- Luego, se transmite el flujo de palabras producido anteriormente. Algunas de éstas son ya formas correctas. Otras, en cambio, son amalgamas o componentes de palabras compuestas. En este caso es necesario construir el GAD de formas, teniendo en cuenta todos estos fenómenos.

7.2.6 | Enriquecimiento de los GAD's

Una vez se han realizado todos los tratamientos de corrección ortográfica, puede ocurrir que existan determinadas formas en los GAD's que sigan siendo desconocidas en el lexicón. Esto quiere decir que no se ha aplicado ningún tipo de corrección cuyo coste se encuentre dentro de un intervalo especificado por el propio usuario. En este sentido,

el último módulo sustituirá en el GAD dicha forma por la etiqueta asociada a palabras desconocidas `_uw` o `_Uw`[18], en función de si se escribe totalmente en minúscula o si, por el contrario, posee algún tipo de mayúscula en su interior.

7.3 | Analizador léxico: FRMG LEXER

Dada la forma de una palabra, el análisis léxico nos permite identificar sus rasgos morfológicos [299] tales como género, número y persona; lematizar y etiquetar[19]. En este sentido se pueden considerar diferentes acercamientos: aquéllos que se basan en la utilización de léxicos [262, 307], los que aplican una fase superficial de reducción de la palabra a su raíz[20] [189, 237]; o aquéllos que se basan en un análisis morfológico más profundo revelando la estructura interna de las palabras [116]. En nuestro contexto, el acercamiento elegido se basa en la primera opción. Más concretamente, se dispone de una herramienta que hace uso del LEFFF, denominada FRMG LEXER[21], cuya función consiste en recoger y gestionar los diferentes recursos y herramientas, de tal forma que se identifique, lematice y etiquete cada una de las palabras presentes en el texto.

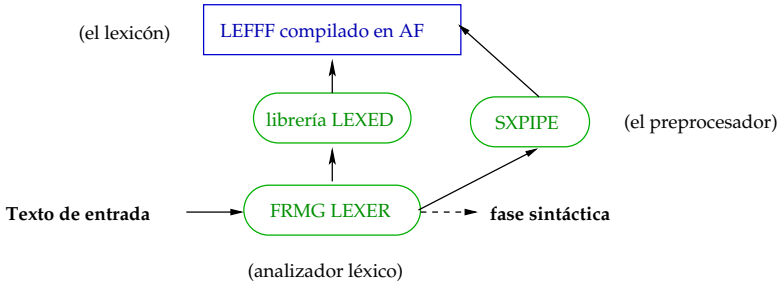


Figura 7.18: Funcionamiento de FRMG-LEXER

Para ello se utiliza la información morfosintáctica recogida en la representación extensional del lexicón LEFFF gestionada a través de una librería en C++. Ésta se distribuye para plataformas Unix bajo licencia GPL y se denomina LEXED[22]. Esta librería es la que provee las funcionalidades necesarias para ejecutar el análisis léxico y permite buscar una cadena, preprocesada por SXPIPE o no, en las entradas de la base de datos construida a partir del lexicón LEFFF, tal y como se observa en la Fig. 7.18. De hecho, esa base de datos no es más que una concatenación de todas las representaciones extensionales procedentes del LEFFF. La arquitectura en la que se centra LEXED está basado en AF's, siendo éste particularmente rápido a la hora de consultarlo, así como una buena alternativa a las tablas *hash* para grandes diccionarios.

[18] en terminología anglosajona corresponde a *unknown word*.
[19] consiste en asignar etiquetas a elementos que se pueden deducir de la morfología de la palabra.
[20] en terminología anglosajona se conoce como *stemming*.
[21] ver enlace `http://alpage.inria.fr/docs/alpchain-doc.pdf`.
[22] ver el enlace `http://www.labri.fr/perso/clement/lexed/`.

Si consideramos el caso en el que la cadena de entrada del analizador FRMG LEXER ha sido tratada por el preprocesador SXPIPE, ésta será combinada con la información recuperada del lexicón LEFFF, asignando a cada unidad léxica del GAD una o varias estructuras en el lexicón compilado. Así, una misma forma podría poseer diferentes entradas. Por ejemplo, podría dar lugar a varios lemas, categorías léxica o disponer de diferentes características sintácticas como distintos marcos de subcategorización o redistribuciones.

A continuación, vamos a proporcionar dos ejemplos para ilustrar las estructuras resultantes. El primero es un ejemplo de salida del analizador FRMG LEXER sin la realización previa de la fase de preprocesamiento, y el segundo ilustra este proceso tras la aplicación del preprocesador SXPIPE.

Ejemplo 7.12 *El siguiente ejemplo representa la salida del analizador* FRMG LEXER *sin la utilización del preprocesador* SXPIPE *para el preprocesamiento de la frase francesa* «Feuilles à nervures denticulées» (`«Hojas con nervaduras dentadas»`). FRMG LEXER *utiliza la librería* LEXED *para que éste le proporcione la información referente a cada una de las cadenas de caracteres separadas mediante delimitadores en el* LEFFF *compilado. Una vez obtenida la información necesaria, ésta es transformada en el formato mostrado en la Fig. 7.19.*

```
'C'(0, lemma{  lex  => feuilles,
    truelex  => 'Feuilles',
    lemma    => feuille,
    cat      => nc,
    top      => nc{gender => fem, number => pl},
    anchor   => tag_anchor{
                name => ht{arg0 => arg{kind => kind[prepvcomp,prepobj,(-)],
                        pcas => prep[de,(-)]}, arg1 => arg{kind => kind[prepvcomp,(-)],
                        pcas => prep['à',(-)]}, arg2 => arg{kind => (-),
                        pcas => (-)}, refl => (-)}, coanchors => [], equations => []}
  },1).
'C'(0, lemma{  lex  => feuilles,
    truelex  => 'Feuilles',
    lemma    => feuiller,
    cat      => v,
    top      => v{diathesis => active, mode => mode[indicative,subjonctive],
                number => sg, person => 2, tense => present},
    anchor   => tag_anchor{
                name => ht{arg0 => arg{kind => subj, pcas => (-)},
                        arg1 => arg{kind => kind[obj,(-)], pcas => (-)},
                        arg2 => arg{kind => (-), pcas => (-)},
                        diathesis => active, imp => '-', refl => (-)}, coanchors => [],
                         equations => []}
  },1).
'C'(1, lemma{  lex  => 'à',
    truelex  => 'à',
    lemma    => 'à',
    cat      => prep,
    top      => prep{pcas => prep[loc,'à']},
    anchor   => tag_anchor{
                name => ht{arg0 => arg{kind => kind[acomp,sadv,scomp,vcomp,obj],
```

```
                        pcas => (-)}, arg1 => arg{kind => (-), pcas => (-)},
                        arg2 => arg{kind => (-), pcas => (-)},
                        refl => (-)}, coanchors => [], equations => []}
  },2).
'C'(2, lemma{  lex  => 'nervures',
    truelex    => 'nervures',
    lemma      => 'nervure',
    cat        => nc,
    top        => nc{gender => fem, number => pl},
    anchor     => tag_anchor{
                name => ht{arg0 => arg{kind => kind[prepvcomp,prepobj,(-)],
                        pcas => prep[de,(-)]}, arg1 => arg{kind => kind[prepvcomp,(-)],
                        pcas => prep['à',(-)]}, arg2 => arg{kind => (-), pcas => (-)},
                        refl => (-)}, coanchors => [], equations => []}
  },3).
'C'(2, lemma{  lex  => 'nervures',
    truelex  => 'nervures',
    lemma    => 'nervurer',
    cat      => v,
    top      => v{diathesis => active, mode => mode[indicative,subjonctive],
                number => sg, person => 2, tense => present},
    anchor  => tag_anchor{
                name => ht{arg0 => arg{kind => subj, pcas => (-)},
                        arg1 => arg{kind => kind[obj,(-)], pcas => (-)},
                        arg2 => arg{kind => (-), pcas => (-)},
                        diathesis => active, imp => '-', refl => (-)}, coanchors => [],
                        equations => []}
  },3).
'C'(3, lemma{ lex  => '_uw',
    truelex  => 'denticulées',
    lemma    => '_ETR',
    cat      => etr,
    top      => etr,
    anchor  => tag_anchor{
                name => ht{arg0 => arg{kind => (-), pcas => (-)},
                        arg1 => arg{kind => (-), pcas => (-)},
                        arg2 => arg{kind => (-), pcas => (-)},
                        refl => (-)},
                        coanchors => [], equations => []}
  },4).
'C'(3, lemma{ lex  => '_uw',
    truelex  => 'denticulées',
    lemma    => 'uw',
    cat      => v,
    top      => v{diathesis => active, mode => infinitive},
    anchor  => tag_anchor{
                name => ht{arg0 => arg{function => suj, kind => kind[subj,(-)],
                                pcas => (-),real => cat['N2',prel,pri,'PP','S',...]},
                        arg1 => arg{function => obj, kind => obj, pcas => (-),
                                real => cat['N2',prel,pri,clr,'PP',antepro,(-)]},
                        arg2 => arg{kind => (-), pcas => (-)},
                        diathesis => active, refl => (-)},coanchors=>[],equations=>[]}
  },4).
'C'(3, lemma{ lex  => '_uw',
    truelex  => 'denticulées',
    lemma    => 'uw',
    cat      => v,
    top      => v{diathesis => active, mode => gerundive},
    anchor  => tag_anchor{
                name => ht{arg0 => arg{function => suj, kind => subj, pcas => (-),
```

```
                                real => cat['N2',prel,pri,'PP','S','CS',(-)]},
                        arg1 => arg{function => obj, kind => obj, pcas => (-),
                                real => cat['N2',prel,pri,clr,'PP',antepro,(-)]},
                        arg2 => arg{kind => (-), pcas => (-)},
                        diathesis => active, refl => (-)}, coanchors => [], equations => []}
  },4).

...

'C'(3, lemma{ lex  => '_uw',
    truelex  => 'denticulées',
    lemma    => 'uw',
    cat      => adv,
    top      => adv{},
    anchor  => tag_anchor{
                name => ht{arg0 => arg{kind => (-), pcas => (-)},
                        arg1 => arg{kind => (-), pcas => (-)},
                        arg2 => arg{kind => (-), pcas => (-)},
                        refl => (-)}, coanchors => [], equations => []}
  },4).
'C'(3, lemma{ lex  => '_uw',
    truelex  => 'denticulées',
    lemma    => 'uw',
    cat      => adj,
    top      => adj{},
    anchor  => tag_anchor{
                name => ht{arg0 => arg{function => suj, kind => kind[subj,(-)],
                                pcas => (-), real => cat['N2',prel,pri,'PP',(-)]},
                        arg1 => arg{function => objde, kind => kind[prepvcomp,(-)],
                                pcas => prep[de,'à',(-)]},
                        arg2 => arg{kind => (-), pcas => (-)},
                        refl => (-)}, coanchors => [], equations => []}
  },4).
'C'(3, lemma{ lex  => '_uw',
    truelex  => 'denticulées',
    lemma    => 'uw',
    cat      => nc,
    top      => nc{},
    anchor  => tag_anchor{
                name => ht{arg0 => arg{function => suj, kind => kind[subj,(-)],
                                pcas => (-), real => cat['N2',prel,pri,'PP',(-)]},
                        arg1 => arg{function => objde, kind => kind[prepobj,(-)],
                                pcas => prep[de,(-)]},
                        arg2 => arg{function => 'objà', kind => kind[prepvcomp,(-)],
                                pcas => prep[de,'à',(-)]},
                        refl => (-)}, coanchors => [], equations => []}
  },4).
```

Figura 7.19: Frase «*Feuilles à nervures denticulées*» representada por FRMG LEXER.

Estas estructuras muestran la salida ofrecida por el analizador léxico. En este sentido, la palabra «denticulées» (`«dentadas»`) *resulta desconocida* («uw»), *por lo que considera todas sus posibles etiquetas, es decir,* «etr», «v», «adv», «adj», «nc». *A este propósito, en la Fig. 7.19 hemos omitido considerar todas las salidas de esa palabra cuando se trata de un verbo, con el fin de no hacer más tediosa dicha representación.*

■

La salida del analizador FRMG LEXER de la Fig. 7.19 asigna a cada una de las cadenas separadas por delimitadores de una frase una estructura arbórea que representa su entrada

en el lexicón compilado, es decir, toda la información morfológica y morfosintáctica de la que se dispone [164]. Es lo que se conoce por *hiperetiquetas*[23]. Posteriormente, éstas serán utilizadas por el analizador sintáctico. En este punto, nos vamos a limitar a describir el formato generado por el analizador léxico.

Como se puede apreciar, las hiperetiquetas que se están describiendo en la Fig. 7.19 están compuestas de los siguientes elementos, procedentes íntegramente de las representaciones extensionales compiladas a partir del LEFFF:

- $i \in [0, n]$: Es la posición en la que comienza la palabra en la oración, considerando que la primera comienza en la posición 0.
- *lex*: Es la forma de la palabra. En el caso de que exista algún error ortográfico y la cadena haya sido preprocesada incluyendo corrección ortográfica, *lex* será la forma corregida.
- *truelex*: Es la palabra tal cual aparece en el texto de entrada. Ésta puede no coincidir con *lex* debido a, por ejemplo, una contracción o un error ortográfico.
- *lemma*: Es el lema de la forma en cuestión.
- *cat*: Almacena la categoría léxica de la palabra como, por ejemplo, «*adj*» para adjetivos, «*nc*» para sustantivos o «*prep*» para preposiciones.
- *top*: Recoge información más detallada acerca de la forma, en función de la categoría léxica. Así por ejemplo, en el caso de tratarse de un verbo, esa información constará de género, número, persona, modo[24], diátesis[25] y auxiliar requerido[26], entre otros. En cambio, si se tratase de un sustantivo sólo dispondrá de género y número.
- *anchor*: En este apartado se detalla la información sintáctica presente en la entrada extensional. Hace referencia a los posibles argumentos del marco de subcategorización, a su redistribución. Esta información es la que sirve de enlace o ancla entre el léxico y la sintaxis. Esto es, a partir de la información resultante en este apartado de cada palabra, se consigue enlazar con las estructuras sintácticas del analizador sintáctico.
- $i \in [1, n + 1]$: Es la posición en la que termina la palabra en la oración.

Ejemplo 7.13 *El siguiente ejemplo representa la salida del analizador* FRMG LEXER *después del preprocesamiento de la frase francesa* «Feuilles de 3-4cm» (`«Hojas de 3-4cm»`)*, formato mostrado en la Fig. 7.20.*

[23] en terminología anglosajona se denomina *hypertag*.

[24] hace referencia a los verbos, por ejemplo, el modo indicativo o subjuntivo.

[25] hace referencia a la *voz*, es decir, la voz activa o la voz pasiva.

[26] en las construcciones verbales, el verbo principal requiere de un auxiliar que puede ser «*ser*/estar» o «*haber*» en función del tipo de estructura.

```
'C'(0, lemma{  lex  => feuilles,
    truelex  => 'Feuilles',
    lemma    => feuille,
    cat      => nc,
    top      => nc{gender => fem, number => pl},
    anchor   => tag_anchor{
                name => ht{arg0 => arg{kind => kind[prepvcomp,prepobj,(-)],
                                  pcas => prep[de,(-)]},
                           arg1 => arg{kind => kind[prepvcomp,(-)], pcas => prep['à',(-)]},
                           arg2 => arg{kind => (-), pcas => (-)},
                           refl => (-)}, coanchors => [], equations => []}},1).
'C'(0, lemma{  lex  => feuilles,
    truelex  => 'Feuilles',
    lemma    => feuiller,
    cat      => v,
    top      => v{diathesis => active, mode => mode[indicative,subjonctive],
                number => sg, person => 2, tense => present},
    anchor   => tag_anchor{
                name => ht{arg0 => arg{function => suj, kind => subj, pcas => (-),
                                  real => cat[cln,'CS','S','N2',prel,pri,'PP',(-)]},
                           arg1 => arg{function => obj, kind => kind[obj,(-)], pcas => (-),
                                  real => cat[cla,'N2',prel,pri,clr,'PP',antepro,(-)]},
                           arg2 => arg{kind => (-), pcas => (-)}, diathesis => active,
                           imp => '-', refl => (-)}, coanchors => [],equations => []}
  },1).
'C'(1, lemma{ lex  => de,
    truelex  =>  'de',
    lemma  => un,
    cat  => det,
    top  => det{def => '-', det => (+), number => pl},
    anchor  => tag_anchor{name => _, coanchors => [], equations => []}
   },2).
'C'(1, lemma{ lex  => de,
    truelex => 'de',
    lemma   => de,
    cat     => prep,
    top     => prep{pcas => de},
    anchor  => tag_anchor{
                name => ht{arg0 => arg{function => obj,
                           kind => kind[acomp,sadv,scomp,vcomp,obj], pcas => (-)},
                           arg1 => arg{kind => (-), pcas => (-)},
                           arg2 => arg{kind => (-), pcas => (-)}, refl => (-)},
                           coanchors => [], equations => []}
   },2).
'C'(2, lemma{ lex  => '_DIMENSION',
    truelex => '3 - 4 cm',
    lemma   => '_DIMENSION',
    cat     => nc,
    top     => nc{},
    anchor  => tag_anchor{
                name => ht{arg0 => arg{function => suj, kind => kind[subj,(-)],
                                  pcas => (-), real => cat['N2',prel,pri,'PP',(-)]},
                           arg1 => arg{function => objde, kind => kind[prepvcomp,
                                  prepobj,(-)],pcas => prep[de,(-)]},
                           arg2 => arg{function => 'objà', kind => kind[prepvcomp,(-)],
                                  pcas => prep['à',(-)]},
```

```
  refl => (-)}, coanchors => [], equations => []}
   },3).
'C'(2, lemma{ lex  => '_DIMENSION',
    truelex => '3 - 4 cm',
    lemma   => '_DIMENSION',
    cat     => np,
    top     => np{number => pl},
    anchor  => tag_anchor{ name => _, coanchors => [], equations => []}
    },3).
```

Figura 7.20: Frase preprocesada «*Feuilles de 3-4cm*» representada por FRMG LEXER.

En estas estructuras, se pueden observar como después del preprocesamiento «3-4 cm» *se ha agrupado bajo una denominación propia del preprocesador* SXPIPE *llamada* _DIMENSION.

■

7.4 | Interfaz entre lexicón y sintaxis: LEFFF-FRMG

Cuando se trabaja con una cadena de PLN, debe de mantenerse cierta uniformidad en lo que a notación se refiere entre las diferentes herramientas que se utilizan, por lo que debe existir cierta dependencia entre ellas. Por ello, es aconsejable la utilización de un módulo que haga de interfaz entre el formato del propio lexicón y las herramientas que lo van a usar. En nuestro caso este papel lo desempeña LEFFF-FRMG, tal y como se ilustra en la Fig. 7.21.

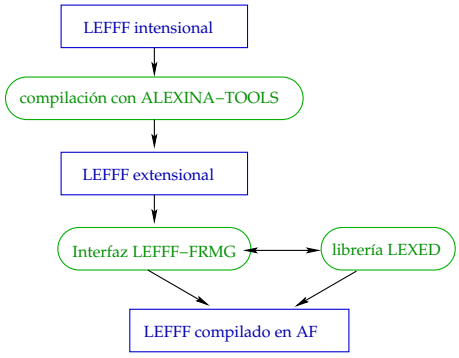


Figura 7.21: Proceso de obtención del AF a partir del LEFFF extensional

De este modo, el conjunto de representaciones extensionales se transforman en un lexicón compilado en un AF del francés, es decir, en un diccionario de fácil y rápida consulta capaz de proveer información morfosintáctica. Ésta será la base de información del analizador FRMG LEXER. Para obtener dicho lexicón, una vez el LEFFF se encuentre ya en forma extensional, es necesario someterlo a una segunda compilación para poder utilizarlo bajo el analizador sintáctico elegido. En nuestro caso, se trata de uno basado en *metagramáticas* del francés. Por ello es necesario el uso del interfaz LEFFF-FRMG.

CAPÍTULO VIII

El nivel sintáctico

Desde un punto de vista descriptivo, la opción elegida recae en las GA's [149, 152]. Se trata de un formalismo gramatical suavemente dependiente del contexto, que se caracteriza por una capacidad generativa superior a las GIC's e inferior a las GDC's, el cuál ha visto últimamente incrementado su interés en el modelado de la sintaxis en PLN por tres razones fundamentales. La primera, un *dominio de localidad extendido* (DLE) que permite definir dependencias sintácticas a cualquier nivel. La segunda, la posibilidad de considerar dependencias cruzadas. La tercera, la extensión natural del modelo independiente del contexto clásico, al pasar la unidad básica de reescritura del símbolo al árbol. A grandes rasgos, los GIC's son un entorno de reescritura de símbolos y sus estructuras elementales de derivación son las producciones. Por contra, las GA's permiten, además, la reescritura explícita y directa de árboles [50] y una complejidad computacional que permite su consideración práctica. Para más detalle consultar el Apéndice C.

De hecho, el analizador sintáctico empleado es un híbrido (GA/GIA) entre las GA'S y las *gramáticas de inserción de árboles* (GIA) [11], también detalladas en el Apéndice C, que utiliza la información proporcionada por la herramienta FRMG LEXER en combinación con el paquete LEFFF-FRMG, tal y como se muestra en la Fig. 8.1.

En este sentido, uno de los principales inconvenientes de las GA's es el que hace referencia a su diseño y mantenimiento. De hecho, una lengua puede necesitar varios miles de *árboles elementales* [201] para conseguir alcanzar una cobertura adecuada. Por lo que, si consideramos la posibilidad de crearlos manualmente, su generación resulta una tarea inabordable. A este respecto, el analizador sintáctico utilizado (FRMG PARSER) es el resultado de la compilación en GA por parte de la herramienta *DyALog* [326, 327] de la información disponible en la metagramática FRMG[1] [44]. Esta conversión se realiza mediante la aplicación denominada MGCOMP [305]. Una vez superado el análisis sintáctico, el resultado obtenido se podrá transformar al formato deseado entre un

[1]se traduce por *metagramática del francés* y en terminología francesa *métagrammaire du français*.

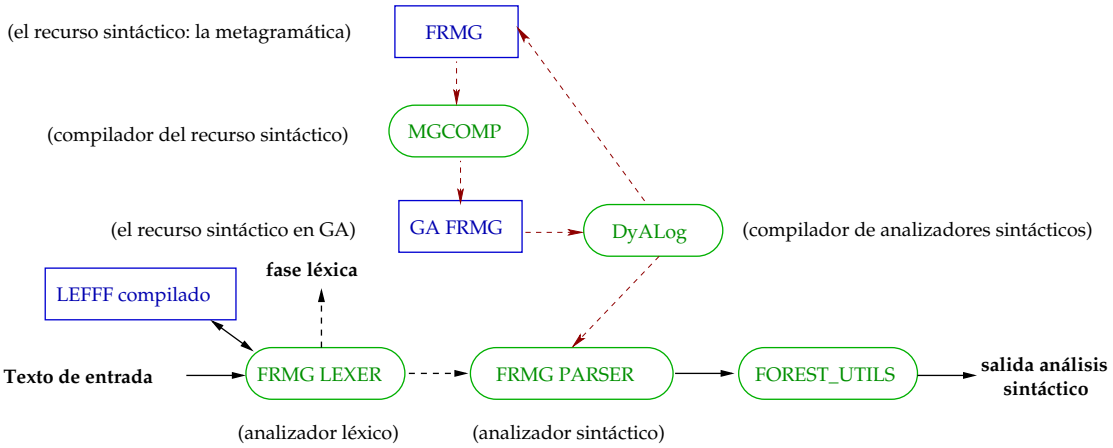


Figura 8.1: Esquema de la cadena utilizada a nivel sintáctico

conjunto de opciones, de tal forma que sea útil y comprensible para los posibles usuarios, mediante la utilización de la herramienta FOREST_UTILS [24].

8.1 | Recurso sintáctico: la metagramática FRMG

La noción original de *metagramática* (MG) se debe a Candito [44], siguiendo la propuesta de Schabes [319], aunque ha sufrido una notoria evolución desde entonces. En este sentido, surge como respuesta frente a los problemas de desarrollo y mantenimiento que presentan las grandes GA's [329, 330]. Para ello, introducen un nivel más abstracto de descripción aplicando restricciones elementales sobre los nodos, agrupados en clases relativamente sencillas, a su vez insertadas en una jerarquía de herencia múltiple. Cada elemento de esta jerarquía se declara como un conjunto de descripciones parciales de árboles [256]. Estas definiciones pueden también subespecificar algunas relaciones entre nodos, por lo que cada subclase de la jerarquía puede enriquecer las restricciones existentes sobre ellas [23].

Con el fin de construir estructuras de árboles prelexicalizadas [2] y de agrupar aquéllas que pertenecen a la misma familia, la MG usa además de las descripciones parciales, funciones sintácticas. Cada clase, por lo tanto, dependerá de una de estas tres dimensiones que estructuran la jerarquía mencionada:

- Dimensión 1 : Subcategorización inicial.
- Dimensión 2 : Redistribución de funciones sintácticas.
- Dimensión 3 : Realizaciones de funciones sintácticas.

donde la subcategorización se expresa como una lista de partes posibles del discurso, sobre la que se asocia una lista de funciones. Esta subcategorización inicial puede ser

modificada por una redistribución, haciendo que la herencia no sea monótona[2]. Los árboles elementales que compartan la misma subcategorización inicial sólo diferirán en el modo de realizar sus funciones sintácticas, y en sus redistribuciones.

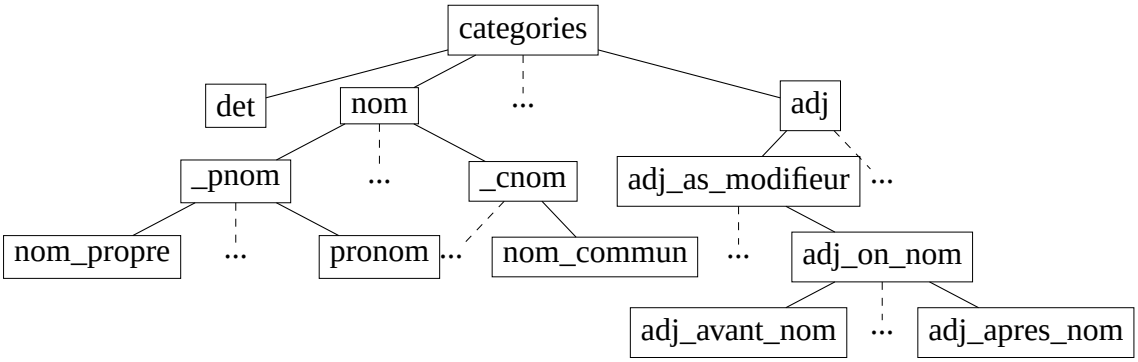


Figura 8.2: Herencia de clases en las categorías léxicas de FRMG

Cada clase en la jerarquía depende necesariamente de una de esas tres dimensiones. Concretamente, las informaciones se organizarán alrededor de *variables globales* que designarán un nodo del árbol, y a las que se le asociará una lista de posibles categorías del discurso, tal y como se observa en la Fig. 8.2, e incluso de funciones sintácticas. Tomando como base esta figura, podemos observar como las clases referentes a las categorías léxicas cuelgan de una clase genérica llamada *categories*. A partir de ahí, se introduce una primera restricción referente a la función léxica desempeñada, como en el caso de *det* para determinante, *nom* para sustantivo o *adj* para adjetivo. Tomando *adj*, vemos como también se pueden ir insertando más restricciones como la de *adj_as_modifieur* que no es más que adjetivos que hacen función de modificadores y que pueden ser aplicados por ejemplo a sustantivos, como en el caso de *adj_on_nom*. En estas situaciones, también es necesario tener en cuenta la localización del adjetivo con respecto al sustantivo dando lugar a, por ejemplo, *adj_avant_nom*. Es decir, aquéllos que se colocan antes del sustantivo. Obtenemos así las clases que forman parte de la MG aplicada al francés, denominada FRMG, y que se muestran en la Fig. 8.3.

De este modo, las MG's permiten una descripción sintáctica expandida con la ayuda de restricciones elementales agrupadas en clases. Pero además de esta característica, también poseen las siguientes [201]:

- *Restricciones topológicas*: Cada clase de la jerarquía contiene una descripción parcial de la estructura de los árboles GA's elementales. Para ello, se emplean las relaciones siguientes: $(=)$ igualdad[3], $(<)$ la precedencia[4], $(\gg)$ el dominio

[2]en cambio la topología de las descripciones parciales es monótona.

[3]dos identificadores de nodos que se relacionen mediante el operador de igualdad equivale a afirmar que ambos se refieren al mismo nodo.

[4]permite establecer el orden entre dos nodos. Cuando éstos son nodos hermanos se denomina precedencia inmediata, aunque en FRMG no se hace esta distinción.

```
1  class categories {
2      node Anchor : [type:anchor];
3      desc.@htcat = node(Anchor).cat;
4      node(Anchor).id = node(Anchor).cat;
5      desc([ht:@ht_fs]);
6  }
7  class det {
8     %%Determiner
9      <: categories;
10     node det : [cat: det ]; det=Anchor;
11     desc.ht = value([arg0: @emptyarg_fs,
12                      arg1: @emptyarg_fs,
13                      arg2: @emptyarg_fs]);
14 }
15 class noun {
16     %% Nouns
17     <: categories;
18     node N2 : [cat: N2,type: std, bot: [enum: -]];
19     desc.ht = value([arg0: [ pcas: -|de|à,
20                              kind: -|obj|scomp|vcomp,
21                              real: -|S],
22                      arg1: @emptyarg_fs,
23                      arg2: @emptyarg_fs]);
24 }
25 class _pnoun {
26     %% model class for proper nouns and pronouns
27     <: noun;
28     N2 >> N;
29     N >> Anchor;
30     - n::agreement; N = n::N;
31     - anchor::agreement; Anchor = anchor::N;
32     node N : [ cat: N, type: std ];
33     node(N2).bot.sat = value(+);
34 ...
35 }
36 class _cnoun {
37     %% Model for Common nouns
38     <: noun;
39     N2 >> N;
40     N2 >> det;
41     N >> Nc;
42     det < N;
43     Nc=Anchor;
44     node N : [cat: N];
45     node det : [cat: det, type: subst];
46     - nc::agreement; Nc = nc::N;
47     - n::agreement; N = n::N;
48     node(det).top.number = node(N2).bot.number;
49     node(det).top.gender = node(N2).bot.gender;
50     node(det).top.wh = node(N2).bot.wh;
51     node(Anchor).bot.person = value(3);
52  ...
53 }
54 ...
```

Figura 8.3: Ejemplo de clases representando categorías léxicas en FRMG

inmediato o directo[5] y $(\gg +)$ el dominio indirecto[6].

Ejemplo 8.1 *En la Fig. 8.3 se observa como una clase puede heredar las restricciones topológicas de una o más superclases. Normalmente esta herencia se representa mediante «<» y se encuentra en la primera línea de cada clase. Así, en la línea 9 se muestra como la clase* det *hereda de la clase* categories*. Lo mismo ocurre con la clase* «noun» *y* «_pnoun».

Pero además de la herencia de restricciones a nivel de clase, las líneas 28 y 29 declaran las relaciones topológicas que han de mantener los nodos implicados en el árbol descrito. Concretamente, el nodo N2 *tiene que dominar directamente al nodo* N $(N2 \gg N)$ *y éste dominar directamente a* Anchor[7] $(N \gg Anchor)$.

Si tomamos ahora las líneas 39 a la 42, vemos como se indica que el nodo N2 *tiene que dominar directamente al nodo* N $(N2 \gg N)$*, y a su vez al nodo* det $(N2 \gg det)$*. El nodo* N *domina directamente al nodo* Nc $(N \gg Nc)$ *y finalmente el nodo* det *tiene que preceder el nodo* N $(det < N)$.

■

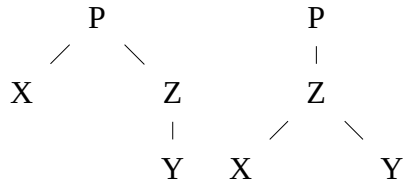
- *Descripción parcial de árbol y árboles* GA*'s minimales*: Los árboles descritos son a menudo *cuasi-árboles* [318]. Un *cuasi-árbol* es un árbol subespecificado, es decir, una descripción que permite construir un número infinito de árboles que no violan las restricciones topológicas. Vamos a ilustrarlo mediante un ejemplo tomado de [201].

Ejemplo 8.2 *El siguiente ejemplo muestra dos cuasi-árboles que proceden de una misma descripción. Si en ella se utilizan relaciones de dominancia indirectas como en el caso del ejemplo propuesto, existe la posibilidad de construir un número arbitrariamente grande de árboles a partir de ella. Así, podemos ver como los dos posibles árboles cumplen con las restricciones descritas.*

descripción:

$$(X < Y) \wedge (Z \gg +Y) \wedge (P \gg +X) \wedge (P \gg +Z)$$

cuasi-árboles:



■

[5] un nodo domina directamente a otro cuando el primero es padre del segundo.
[6] un nodo domina indirectamente a otro cuando es su ancestro.
[7] es el enlace o ancla entre el léxico y la sintaxis.

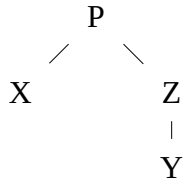
Para dar cauce al problema de la elevada cantidad de árboles que se pueden construir, se introduce el concepto de *árbol minimal*. Un árbol minimal no es más que un cuasi-árbol en el cual se han sustituido las relaciones indirectas por las directas, evitando el incremento de interpretaciones para una misma descripción topológica e impidiendo la inserción infinita de nodos entre dos *cuasi-nodos*, es decir, entre dos nodos subespecificados del cuasi-árbol.

Ejemplo 8.3 *Siguiendo con el Ejemplo 8.2, si sustituimos las relaciones de dominio indirecto por las de directo, impedimos que esa descripción topológica dé lugar a diferentes interpretaciones, ya que no se permite introducir más nodos entre ellos. Al cuasi-árbol generado es el que llamamos* árbol minimal.

descripción:

$$(X < Y) \wedge (Z \gg Y) \wedge (P \gg X) \wedge (P \gg Z)$$

cuasi-árboles:



■

- *Restricciones de unificación*: Otro conjunto de restricciones está determinado por las declaraciones o ecuaciones de *estructuras de rasgos*[8]. Se trata de que los nodos de los árboles elementales pueden estar decorados con un conjunto de pares atributo-valor denominado *rasgo*, de tal manera que el valor puede ser atómico o a su vez ser otro rasgo. De este modo, cada una de esas estructuras describen tanto al nodo como a sus relaciones con los demás nodos del mismo árbol, mientras que las operaciones de adjunción y sustitución se definen en términos de unificación de dichas estructuras.

  Así, normalmente se suele asociar a cada árbol elemental dos rasgos denominados *superior* (*top*) e *inferior* (*bot*). Intuitivamente, el rasgo superior es la relación que se establece con el nodo superior, es decir, con respecto al del superárbol, mientras que el inferior es el que se establece con respecto al del subárbol. De este modo, cuando se posee un nodo marcado para sustitución, no es necesaria la presencia del rasgo inferior.

  Ejemplo 8.4 *Si tomamos de nuevo la Fig. 8.3, las líneas 10, 18, 32, 44 y 45 declaran directamente las estructuras de rasgos para los nodos* det, N2 *y* N. *Además, de las líneas 48 a 52 se expresan restricciones de unificación a través de ecuaciones de rasgos. Concretamente, en estas condiciones se establece la*

[8] para más detalle, ver las GA*'s basadas en estructuras de rasgos* (GAER's).

concordancia de género y número entre el nodo superior det *y el nodo inferior* N2. *Lo mismo ocurre con las oraciones interrogativas* wh.

■

- *Guardas*: Una clase puede contener restricciones condicionales sobre ciertos nodos, denominadas *guardas*, que dependiendo de la existencia o no de un determinado nodo conllevará la validación o no de las ecuaciones de estructuras de rasgos descritas en su parte derecha. Éstas se expresan mediante ecuaciones indicando a que rasgo o atributo concreto nos estamos refiriendo. Se representa de la siguiente manera:

$$X \ => \ nodo(Y).estructura\,de\,rasgos \ = \ valor(v1), ...;$$

Esta ecuación indica que si el $nodo(Y)$ que modela la clase está acompañado del elemento X, entonces el valor de su estructura de rasgos viene dado por $valor(v1)$. Del mismo modo, también se puede querer expresar una negación sobre un valor atómico, representándose de la siguiente forma:

$$\sim X \ => \ nodo(Y).estructura\,de\,rasgos \ = \ valor(v2), ...;$$

Esta ecuación indica que si el $nodo(Y)$ no está acompañado del elemento X, expresado mediante $\sim X$, entonces el valor de su estructura de rasgos viene dado por $valor(v2)$.

Ejemplo 8.5 *Supongamos que tenemos las siguientes guardas:*

```
    det => node ( N2 ). bot . sat = value (+);

  ~ det => node ( N2 ). bot . wh = value (-);
```

La primera indica que si el sustantivo que modela la clase tiene un determinante (det)*, entonces el sintagma nominal que reúne a ambos* (N2) *está* saturado*, es decir, que su núcleo está acompañado por un determinante.*

En cambio, en el segundo caso se describe que si el sustantivo no está acompañado por un determinante (det)*, entonces no puede tratarse de un sintagma nominal dentro de una oración interrogativa.*

■

Además, también existe la posibilidad de usar disyunciones $(|)$, tal y como se puede observar entre las líneas 19 y 21 de la Fig. 8.3.

8.2 | Compilación de la metagramática en GA: MGCOMP

A partir de la jerarquía de tres dimensiones descrita manualmente, y gracias a las características comentadas anteriormente, una fase de compilación permitirá obtener de esas clases todos sus rasgos, incluso aquéllos adquiridos a través de la herencia, y usar las restricciones para derivar estructuras gramaticales aptas para las GA's. El compilador de MG's utilizado, denominado MGCOMP [305], generará automáticamente los árboles elementales asociados a las descripciones parciales, en un proceso que conlleva dos etapas principales [23]:

- En primer lugar, crea las clases con las estructuras de rasgos, tanto propias como heredadas, para traducirlas seguidamente en árboles elementales, especificando las relaciones de dominio y de precedencia de las descripciones parciales.
- En segundo lugar, cada clase creada por el compilador heredará la estructura de rasgos de una clase terminal de dimensión 1, después de una clase terminal de dimensión 2, y después de tantas clases terminales de dimensión 3 como funciones sintácticas existan.

De este modo, el resultado obtenido es un conjunto de árboles elementales y minimales denominado GA FRMG, donde las descripciones abstractas se hacen más modulares, a la vez que se favorece la factorización de conjuntos de restricciones comunes a varios fenómenos sintácticos[9].

8.3 | Compilación de analizadores sintácticos: DyALog

El sistema *DyALog* [326, 327] es una herramienta que integra un entorno de compilación y de ejecución de programas lógicos orientados a la construcción de analizadores sintácticos. Cubre diversos formalismos, entre ellos las GA's, las GDC's o las GIA's; lo que en particular permite construir analizadores híbridos GA/GIA [9] capaces de analizar una GA e identificar sus partes GIA. Es de señalar que la GA obtenida tras la compilación de FRMG, que hemos denominado GA FRMG, es casi enteramente GIA. Más concretamente, *DyALog* permite la compilación de la gramática GA FRMG en un analizador tabular basado en FRMG, con las siguientes características:

- *Representación de los pasos de cálculos de la estrategia de análisis,* donde se realiza un estudio previo de la gramática GA FRMG para determinar cuáles son los árboles que pueden ser compilados en árboles GIA, ya que éstos ofrecen una complejidad menor. Con el fin de reducir el número de árboles, realiza un proceso de factorización sobre los subárboles de la gramática. Esta factorización

[9] como por ejemplo las reglas de concordancia.

no cambia la naturaleza del formalismo, pero permite reducir exponencialmente su talla [329, 330].

- *Aplicación de un algoritmo de tabulación basado en subsunción para gestionar los objetos*, con el fin de evitar cálculos particulares si uno más general se ha realizado ya [181]. Tomando como punto de partida la Fig. 8.4, el funcionamiento del sistema se desarrolla alrededor de una *tabla de objetos*, en la que éstos se encuentran ya tabulados, y que es gestionado por una *agenda* encargada de indicar el orden en el que se van a tratar cada uno de ellos.

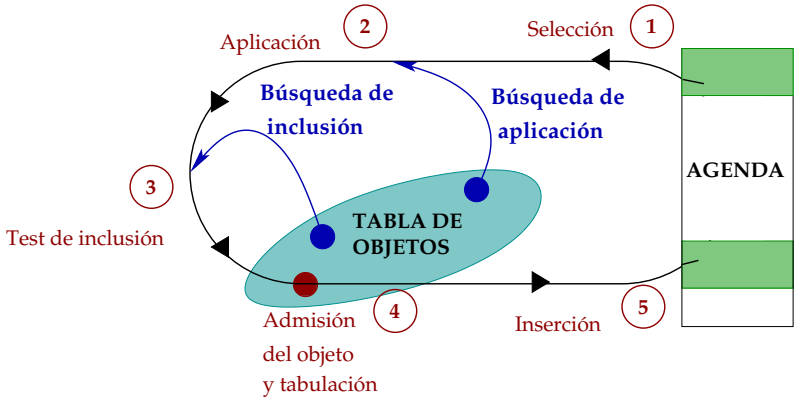


Figura 8.4: Modelo de ejecución de *DyALog*

Así, el proceso comienza seleccionando un objeto O_1 en la *agenda*, tomando prioritariamente los más generales. A continuación, se busca en la *tabla de objetos* aquéllos que pueden aplicarse sobre O_1. De este modo, cada objeto O_2 encontrado se aplica sobre O_1 de tal manera que genere un objeto O_3. Luego, cada objeto O_3 producido se somete a un *test de subsunción* que se descompone en dos fases bien diferenciadas:

  - El *test de subsunción débil* que elimina O_3 si es una instancia de un objeto ya tabulado.
  - El *test de subsunción fuerte* que elimina los objetos ya tabulados y que son instancias de O_3.

  Cada nuevo objeto producido O_3 que supere el test de subsunción débil se incluye tanto en la tabla como en la agenda. De este modo, los tests se pueden realizar no solamente sobre los objetos de la tabla sino también sobre los de la agenda. El proceso se repite hasta el agotamiento de ésta.

- *Compartición de cálculos*, usando técnicas de programación dinámica, permite devolver el conjunto de árboles de derivación producidos por un análisis en un formato compacto, evitando la multiplicación de cálculos y estructuras.

8.4 | Analizador sintáctico: FRMG PARSER

Se trata de un analizador sintáctico profundo y de gran cobertura para el francés. Una descripción gramatical de alto nivel en forma de GA sirve de punto de partida para la obtención de dicho analizador [149, 152]. Concretamente, se trata de GA FRMG. Así, esta gramática se compila mediante el sistema *DyALog* [326, 327] dando lugar a un analizador sintáctico denominado FRMG PARSER. Además, su salida toma la forma de un *bosque compartido de derivación* GA/GIA, que al manejar sentencias ambiguas permite factorizar la representación de los resultados del análisis.

Ejemplo 8.6 *Si analizamos sintácticamente la frase francesa* « Feuilles à nervures» (`«Hojas con nervaduras»`)*, obtenemos la siguiente salida mediante* FRMG PARSER.

```
Shared Forest
*ANSWER*{answer=> [L = [],N = 3,A = 0]}
  0 <-- [0]1
S{mode=> -, extraction=> -, sat=> +, xarg=> -, control=> -, tense=> -, person=> -,
gender=> -,number=> -}(0,4)
  1 <-- [start]2 [comp]3 [S]4 5
start (0,0)
  2 <-- 6
comp{number=> pl, gender=> fem, person=> 3, real=> N2}(0,3)
  3 <-- [N2]7 8
S{mode=> G__1, extraction=> H__1::extraction[-, adjx, cleft, topic, wh], inv=> I__1,
  sat=> -, xarg=> K__1, control=> -, tense=> L__1, neg=> M__1, person=> N__1,
gender=> O__1, wh=> P__1, number=> Q__1}(3,3) * S{mode=> G__1, extraction=>
H__1::extraction[-, adjx, cleft, topic, wh], inv=> I__1, sat=> +, xarg=> K__1,
control=> -, tense=> L__1, neg=> M__1, person=> N__1, gender=> O__1, wh=> P__1,
number=> Q__1}(3,3)
  4 <-- [Punct]9 10
verbose!struct(7 comp_sentence, 7 comp_sentence)
  5 <-- verbose!struct(start, ht{cat=> -})
  6 <-- N2{number=> pl, gender=> fem, sat=> -, hum=> -, time=> -, wh=> -, person=> 3,
enum=> -}(0,3)
  7 <-- [nc]11 [N2]12 13
verbose!struct(57 N2_as_comp N2:agreement, 57 N2_as_comp N2:agreement)
  8 <-- end (3,3)
  9 <-- 14
verbose!struct(25 empty_spunct shallow_auxiliary, 25 empty_spunct shallow_auxiliary)
  10 <--
verbose!anchor(feuilles, 0, 1, 59 n:agreement nc:agreement cnoun_leaf, nc{number=> pl,
gender=> fem, person=> 3, def=> +, hum=> -, time=> -}, [feuille,E1F1|Feuilles],
tag_anchor{name=> ht{anchor=> feuilles, arg0=> arg{kind=> -, pcas=> prep[-, de], real=>
cat[-, S], extracted=> arg[-, cleft, rel, topic, wh], function=> objde}, arg1=>
arg{kind=> -, pcas=> -, real=> -, extracted=> -, function=> objà}, arg2=>
arg{kind=> -, pcas=> -, real=> -, extracted=> -}, cat=> nc, refl=> -}, equations=> []})
   11 <-- N2{number=> Q__1, gender=> R__1, sat=> S__1, case=> V__1, person=> X__1,
enum=> Y__1}(1,3) * N2{number=> Q__1, gender=> R__1, sat=> S__1, case=> V__1,
person=> X__1, enum=> Y__1}(3,3)
```

```
  12 <-- [prep]15 [N2]16 17
verbose!struct(59 n:agreement nc:agreement cnoun_leaf, ht{anchor=> feuilles, arg0=>
arg{kind=> -, pcas=> prep[-, de], real=> -, extracted=> -, function=> objde}, arg1=>
arg{kind=> -, pcas=> -, real=> -, extracted=> -, function=> objà}, arg2=> arg{kind=> -,
pcas=> -, real=> -, extracted=> -}, cat=> nc, refl=> -})
  13 <-- verbose!struct(end, ht{cat=> -})
  14 <-- verbose!anchor(à, 1, 2, 42 prep_noun_modifier shallow_auxiliary, prep{pcas=>
prep[loc, à]}, [à,E1F2|à], tag_anchor{name=> ht{anchor=> à, arg0=> arg{kind=>
kind[acomp, obj, sadv, scomp, vcomp], pcas=> -, real=> cat[-, N, N2, S, adj, adv],
extracted=> -, function=> obj}, arg1=> arg{kind=> -, pcas=> -, real=> -, extracted=> -},
arg2=> arg{kind=> -, pcas=> -, real=> -, extracted=> -}, cat=> prep, refl=> -},
equations=> []})
  15 <-- N2{number=> pl, gender=> fem, sat=> -, hum=> -, time=> -, wh=> -, person=> 3,
enum=> -}(2,3)
  16 <-- [nc]18 19
verbose!struct(42 prep_noun_modifier shallow_auxiliary, ht{anchor=> à, arg0=>
arg{kind=> obj, pcas=> -, real=> N, extracted=> -, function=> obj}, arg1=> arg{kind=> -,
pcas=> -, real=> -, extracted=> -}, arg2=> arg{kind=> -, pcas=> -, real=> -,
extracted=> -}, cat=> prep, refl=> -})
  17 <-- verbose!anchor(nervures, 2, 3, 59 n:agreement nc:agreement cnoun_leaf,
nc{number=> pl, gender=> fem, person=> 3, def=> +, hum=> -, time=> -}, [nervure,E1F3|
nervures], tag_anchor{name=> ht{anchor=> nervures, arg0=> arg{kind=> -, pcas=>
prep[-, de], real=> cat[-, S], extracted=> arg[-, cleft, rel, topic, wh],
function=> objde}, arg1=> arg{kind=> -, pcas=> -, real=> -, extracted=> -,
function=> objà}, arg2=> arg{kind=> -, pcas=> -, real=> -, extracted=> -}, cat=> nc,
refl=> -}, equations=> []})
  18 <-- verbose!struct(59 n:agreement nc:agreement cnoun_leaf, ht{anchor=> nervures,
arg0=> arg{kind=> -, pcas=> prep[-, de], real=> -, extracted=> -, function=> objde},
arg1=> arg{kind=> -, pcas=> -, real=> -, extracted=> -, function=> objà}, arg2=>
arg{kind=> -, pcas=> -, real=> -, extracted=> -}, cat=> nc, refl=> -})
  19 <--
```

Figura 8.5: Ejemplo de bosque compartido de derivación

El bosque compartido de derivación se representa mediante reglas gramaticales [177], siendo su raíz el 0. A partir de aquí, cada una de estas reglas representan a su vez fragmentos de éste, formado en su parte izquierda por un símbolo no terminal que representa el nodo padre, y en su parte derecha los nodos descendientes. Así, la Fig. 8.6 muestra la regla en la que se indica como el nodo padre es el 0 y el nodo descendiente es el 1.

```
0 <-- [0]1
```

Figura 8.6: Primera regla del bosque compartido de derivación

A su vez, cada no terminal numérico puede estar etiquetado, como lo muestra la Fig. 8.7, donde «18» *lo está de* «[nc]». *De este modo, la etiqueta resulta útil para determinar el tipo de relación sintáctica existente entre los símbolos terminales de dos árboles involucrados en una operación. Por ejemplo, en el fragmento* `18 <- verbose!struct(59 n:agreement nc:agreement cnoun_leaf, ht{anchor=>nervures,...`, *se indica que la parte derecha de la regla ocupará el lugar del no terminal numérico 18, etiquetado con [nc] en la regla de parte izquierda 16. Desde el punto de vista arbóreo,*

se puede ver que ese nodo es sobre el que se realiza una operación de sustitución[10]*, cuyo símbolo terminal es el sustantivo* «nervures» (`«nervaduras»`). *En este sentido,*

```
16 <-- [nc]18 19
```

Figura 8.7: Ejemplo de etiqueta sobre un no terminal

además de los símbolos no terminales, los árboles parciales que constituyen el bosque compartido de derivación, también pueden poseer elementos terminales en la parte derecha de las reglas gramaticales. Estos símbolos son las anclas y se recogen en las etiquetas `verbose!anchor`. *Es el caso, por ejemplo, en la Fig. 8.8, donde el ancla es la palabra* «feuilles» (`«hojas»`)*, cuya posición se encuentra entre el* 0 *y el* 1*, y a su vez es el ancla de la estructura sintáctica* (`59 n:agreement nc:agreement cnoun_leaf ...`) *perteneciente a la* GA.

```
verbose!anchor(feuilles, 0, 1, 59 n:agreement nc:agreement cnoun_leaf, nc{number=> pl,
gender=> fem, person=> 3, def=> +, hum=> -, time=> -}, [feuille,E1F1|Feuilles],
tag_anchor{name=> ht{anchor=> feuilles, arg0=> arg{kind=> -, pcas=> prep[-, de], real=>
cat[-, S], extracted=> arg[-, cleft, rel, topic, wh], function=> objde}, arg1=>
arg{kind=> -, pcas=> -, real=> -, extracted=> -, function=> objà}, arg2=>
arg{kind=> -, pcas=> -, real=> -, extracted=> -}, cat=> nc, refl=> -}, equations=> []})
```

Figura 8.8: Elemento terminal recogido en la etiqueta `verbose!anchor`

■

8.5 | Representación del análisis sintáctico: FOREST_UTILS

Una vez analizada la frase, es necesario tratar el bosque compartido de derivación generado para obtener una salida bajo forma de dependencias. De ello se encarga la herramienta FOREST_UTILS. Concretamente, las anclas de los árboles relacionadas por una operación GA sobre un determinado nodo con una determinada etiqueta, generan una relación de dependencia en asociación con aquélla. En FRMG, éstas son generalmente elegidas para reflejar la función gramatical que desempeñan, como por ejemplo la de *sujeto* u *objeto*, aunque en este caso simplemente permiten indicar el tipo de sintagma a tratar.

Con el objetivo de que el análisis sintáctico generado pueda ser utilizado por aplicaciones de PLN de alto nivel, en FOREST_UTILS se encuentran implementados diversos módulos *Perl* [305] que permiten una primera conversión de estos bosques compartidos de derivación a diferentes formatos. Uno de los posibles es XML DEP [305]. Concretamente, éste trata de representar un *grafo de dependencias* mediante XML.

[10] si fuese una operación de adjunción, el nodo estaría precedido por el símbolo #.

Ejemplo 8.7 *Supongamos que hemos analizado sintácticamente la frase* «Feuilles à nervures denticulées» (`«Hojas con nervaduras dentadas»`) *del* corpus $\mathscr{B}$*, y que hemos obtenido el bosque compartido de derivación gracias a* FRMG PARSER*. Posteriormente si éste se trata con* FOREST_UTILS *utilizando el formato* XML DEP*, se obtiene una salida bajo forma de dependencias, tal como la que se muestra en la Fig. 8.9.*

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes"?>
<dependencies>
  <cluster left="0" right="0" id="E1c_0_0" tok="" lex=""/>
  <cluster left="0" right="1" id="E1c_0_1" tok="feuilles" lex="E1F1|Feuilles"/>
  <cluster left="1" right="2" id="E1c_1_2" tok="à" lex="E1F2|à"/>
  <cluster left="2" right="3" id="E1c_2_3" tok="nervures" lex="E1F3|nervures"/>
  <cluster left="3" right="3" id="E1c_3_3" tok="" lex=""/>
  <cluster left="3" right="4" id="E1c_3_4" tok="_uw" lex="E1F4|denticulées"/>
  <cluster left="4" right="4" id="E1c_4_4" tok="" lex=""/>
  <node cluster="E1c_0_0" tree="7 comp_sentence" form="" lemma="" xcat="S" cat="S"
    id="E1n012" deriv="E1d000011"/>
  <node cluster="E1c_0_0" tree="57 N2_as_comp N2:agreement" form="" lemma="" xcat="comp"
    cat="comp" id="E1n011" deriv="E1d000014"/>
  <node cluster="E1c_0_0" tree="start" form="" lemma="" xcat="start" cat="start"
    id="E1n008" deriv="E1d000010"/>
  <node cluster="E1c_0_1" tree="59 n:agreement nc:agreement cnoun_leaf" form="feuilles"
    lemma="feuille" xcat="N2" cat="nc" id="E1n007" deriv="E1d000000 E1d000007"/>
  <node cluster="E1c_1_2" tree="42 prep_noun_modifier shallow_auxiliary" form="à"
    lemma="à" xcat="N2" cat="prep" id="E1n006" deriv="E1d000001 E1d000008"/>
  <node cluster="E1c_2_3" tree="59 n:agreement nc:agreement cnoun_leaf" form="nervures"
    lemma="nervure" xcat="N2" cat="nc" id="E1n004" deriv="E1d000002 E1d000006 E1d000009"/>
  <node cluster="E1c_3_3" tree="127 S:agreement modifier_after_x participiale_on_noun
    shallow_auxiliary" form="" lemma="" xcat="N2" cat="N2" id="E1n002" deriv="E1d000003"/>
  <node cluster="E1c_3_4" tree="197 V1VMod:agreement arg0:caimp:agreement clsubj:agreement
    lsubj_alt:agreement clsubj_il:agreement clsubj_ilimp:agreement arg0:ilimp:agreement
    arg0:imp_subj_alt:agreement ante:clitic_sequence post:clitic_sequence clitics
    arg1:collect_real_arg arg2:collect_real_arg arg0:collect_real_subject
    arg1:real_group_comp arg2:real_group_comp ncpred:real_group_comp
    arg0:PP:true_subject
    arg0:cl:true_subject arg0:noun:true_subject arg0:post_PP:true_subject
    arg0:post_noun:true_subject arg0:post_s:true_subject arg0:post_v:true_subject
    arg0:s:true_subject arg0:v:true_subject v_with_subcat Infl:verb_agreement
    V:verb_agreement v:verb_agreement V1:verb_agreement_ancestor arg1:verb_argument_other
    arg2:verb_argument_other arg0:verb_argument_subject verb_canonical
    verb_categorization_active" form="_uw" lemma="uw" xcat="S" cat="v"
    id="E1n001" deriv="E1d000004"/>
  <node cluster="E1c_3_4" tree="124 adj_after_noun arg0:adj_argument arg1:adj_argument
    adj:agreement modifier_after_x adjP:node_agreement shallow_auxiliary" form="_uw"
    lemma="uw" xcat="N2" cat="adj" id="E1n003" deriv="E1d000005"/>
  <node cat="nc" cluster="E1c_3_4" tree="lexical" form="_uw" lemma="uw" id="E1n005"/>
  <node cluster="E1c_4_4" tree="25 empty_spunct shallow_auxiliary" form="" lemma=""
    xcat="S" cat="S" id="E1n010" deriv="E1d000012"/>
  <node cluster="E1c_4_4" tree="end" form="" lemma="" xcat="end" cat="end" id="E1n009"
    deriv="E1d000013"/>
  <edge source="E1n012" target="E1n011" label="comp" type="subst" id="E1e001">
   <deriv names="E1d000011" source_op="E1o1" target_op="E1o3" span="0 4"/>
  </edge>
  <edge source="E1n012" target="E1n008" label="start" type="subst" id="E1e002">
   <deriv names="E1d000011" source_op="E1o1" target_op="E1o2" span="0 0"/>
```

```
  </edge>
  <edge source="E1n011" target="E1n007" label="N2" type="subst" id="E1e003">
   <deriv names="E1d000014" source_op="E1o3" target_op="E1o7" span="0 4"/>
  </edge>
  <edge source="E1n007" target="E1n006" label="N2" type="adj" id="E1e004">
   <deriv names="E1d000007" source_op="E1o7" target_op="E1o12" span="1 3 3 3"/>
   <deriv names="E1d000000" source_op="E1o7" target_op="E1o15" span="1 4 4 4"/>
  </edge>
  <edge source="E1n006" target="E1n004" label="N2" type="subst" id="E1e005">
   <deriv names="E1d000008" source_op="E1o12" target_op="E1o18" span="2 3"/>
   <deriv names="E1d000001" source_op="E1o15" target_op="E1o24" span="2 4"/>
  </edge>
  <edge source="E1n004" target="E1n002" label="N2" type="adj" id="E1e006">
   <deriv names="E1d000002" source_op="E1o24" target_op="E1o13" span="3 4 4 4"/>
  </edge>
  <edge source="E1n007" target="E1n002" label="N2" type="adj" id="E1e007">
   <deriv names="E1d000007" source_op="E1o7" target_op="E1o13" span="3 4 4 4"/>
  </edge>
  <edge source="E1n004" target="E1n003" label="N2" type="adj" id="E1e008">
   <deriv names="E1d000002" source_op="E1o24" target_op="E1o13" span="3 4 4 4"/>
  </edge>
  <edge source="E1n007" target="E1n003" label="N2" type="adj" id="E1e009">
   <deriv names="E1d000007" source_op="E1o7" target_op="E1o13" span="3 4 4 4"/>
  </edge>
  <edge source="E1n002" target="E1n001" label="SubS" type="subst" id="E1e010">
   <deriv names="E1d000003" source_op="E1o13" target_op="E1o20" span="3 4"/>
  </edge>
  <edge source="E1n004" target="E1n005" label="Nc2" type="lexical" id="E1e011">
   <deriv names="E1d000006" source_op="E1o24" target_op="E1o29" span="3 4"/>
  </edge>
  <edge source="E1n012" target="E1n010" label="S" type="adj" id="E1e012">
   <deriv names="E1d000011" source_op="E1o1" target_op="E1o4" span="4 4 4 4"/>
  </edge>
  <edge source="E1n010" target="E1n009" label="Punct" type="subst" id="E1e013">
   <deriv names="E1d000012" source_op="E1o4" target_op="E1o9" span="4 4"/>
  </edge>
  <op cat="N2" span="0 3" id="E1o7" deriv="E1d000000">
   <narg type="top"> ... </narg>
  </op>
  <hipertag derivs="E1d000001" id="E1ht0006"> ... </hipertag>
</dependencies>
```

Figura 8.9: Salida en formato XML DEP de la frase «*Feuilles à nervures denticulées*»

■

La estructura, mostrada en la Fig. 8.9, representa un grafo de dependencias de la frase en cuestión. Como se puede apreciar, la información que en ella se describe hace referencia a, entre otros, sus componentes y al modo que éstos tienen de relacionarse entre sí. Concretamente, distinguimos los siguientes elementos en este formato:

- *Cluster* (o grupo): Representa una cadena delimitada por una posición concreta de la frase analizada. De este modo cada uno se compone a su vez de diversas etiquetas que representan informaciones como:
  - *left*: Delimita la posición de inicio de la cadena en la frase.
  - *right*: Delimita la posición de fin de la cadena en la frase.

- *id*: Es el identificador de la cadena en cuestión.
- *tok*: Contiene la cadena delimitada que se está representando en la frase, sobre el que, en caso de usarlo, se ha aplicado el preprocesador.
- *lex*: Contiene la cadena delimitada tal y como aparece en la frase, acompañado previamente, en el caso de usarlo, del identificador proporcionado por el preprocesador.

Ejemplo 8.8 *Si extraemos del Ejemplo 8.7 el fragmento de la Fig. 8.10, observamos como el grupo que comienza en la posición* 0 `(left)` *de la frase y que termina en la* 1 `(right)` *tiene por* `tok` *a* «feuilles», *e* `id` *es* «E1c_0_1».

```
<cluster left="0" right="1" id="E1c_0_1" tok="feuilles" lex="E1F1|Feuilles"/>
```

Figura 8.10: Ejemplo de `cluster`

Hay que destacar que `tok` *no tiene porque siempre estar representando a la forma de la palabra. Así, por ejemplo, después de pasar por la fase de preprocesamiento, las dimensiones se etiquetan como* «_DIMENSION», *como en el caso de la Fig. 8.11 con* «3-4 cm».

```
<cluster left="2" right="3" id="E1c_2_3" tok="_DIMENSION" lex="E1F3|3 E1F4|
- E1F5|4 E1F6|cm"/>
```

Figura 8.11: Otro ejemplo de `cluster`

■

- *Node* (o nodo): Representa cada una de las opciones de análisis léxico obtenidas para una cadena delimitada por una posición concreta de la frase. De este modo, cada nodo posee los siguientes atributos:
  - *cluster*: Es el identificador del grupo al que pertenece ese nodo.
  - *tree*: Se trata del árbol que cubre dicho nodo.
  - *xcat*: Se trata de la categoría maximal del árbol anterior.
  - *form*: Es la forma de la palabra después de realizar posibles correcciones ortográficas y de convertirla a minúsculas mediante SXPIPE.
  - *lemma*: Es la palabra en su forma canónica, tal y como aparece en el diccionario LEFFF.
  - *deriv*: Es el conjunto de identificadores de árboles compartidos de derivaciones que involucran al nodo, siendo éste el origen de un `edge` (o arco).
  - *cat*: Es la categoría léxica asignada a ese nodo.

- *id*: Es el identificador del nodo.

Ejemplo 8.9 *Si extraemos del Ejemplo 8.7 el fragmento de la Fig. 8.12, observamos como existe un nodo que está asociado al grupo* «E1c_0_1» *con forma* `(form)` «feuilles» *e identificador* `(id)` «E1n003», *y a su vez posee por lemma a* «feuille» *y categoría léxica* `(cat)` *a* «nc».

```
<node cluster="E1c_0_1" tree="59 n:agreement nc:agreement cnoun_leaf" form=
"feuilles" lemma="feuille" xcat="N2" cat="nc" id="E1n003" deriv="E1d000000"/>
```

Figura 8.12: Ejemplo de `node`

■

- *Edge* (o arco): Relaciona un nodo origen con uno destino. De este modo, cada arco posee los siguientes atributos:
  - *source*: Indica cuál es el identificador del nodo del que parte el arco.
  - *target*: Indica cuál es el identificador del nodo al que llega el arco.
  - *label*: Es la etiqueta del arco o dependencia sintáctica, y representa su función.
  - *type*: Indica el tipo de operación que se ha realizado en el árbol.
  - *id*: Es el identificador del arco.
  - *deriv*: Cada arco relaciona un nodo origen con uno destino, marcando las dependencias sintácticas entre ellos. Pero también, cada arco puede ser utilizado por un subconjunto de árboles de derivación compartidos en su nodo origen, particionándose entre varias derivaciones, cada una de ellas identificada por el atributo *names*. Estas derivaciones son operaciones GA realizadas durante el análisis, de modo que puede existir más de una entre dos nodos, ya que las estructuras sintácticas en la GA se pueden solapar.

Ejemplo 8.10 *Si extraemos del Ejemplo 8.7 el fragmento de la Fig. 8.13, observamos como el arco con identificador* `(id)` «E1e004», *tiene como nodo de partida* `(source)` «E1n003» *y de llegada* `(target)` *al* «E1n002». *Además, la función sintáctica del arco* `(label)` *es* «N2», *y el tipo de operación* `(type)` *que se realizó para su obtención ha sido* «adj», *es decir, una adjunción.*

```
<edge source="E1n003" target="E1n002" label="N2" type="adj" id="E1e004">
 <deriv names="E1d000000" source_op="E1o7" target_op="E1o12" span="1 3 3 3"/>
</edge>
```

Figura 8.13: Ejemplo de `edge`

■

- *Op* (u operaciones): Son las trazas de las operaciones realizadas dentro de los bosques compartidos de derivación. Cada una de ellas posee:
  - *id*: Es el identificador de la misma.
  - *cat*: Es la categoría sintáctica no terminal.
  - *span*: Es la unidad que mide la amplitud que se está considerando en el árbol, indicando de donde a donde se están tomando los nodos.
  - *deriv*: Una operación puede estar asociada a un conjunto de derivaciones que están rivalizando para construirla. De este modo, cada derivación está asociada a un nodo origen.
  - *narg*: Indica la estructura de rasgo que describe el nodo y su relación con los demás en el mismo árbol. En este caso, se indicaran los rasgos superiores, con la etiqueta *top* y los inferiores, con la etiqueta *bot*.
- *Hipertags* (o hiperetiquetas): Son las estructuras arbóreas proporcionadas por el analizador léxico, que indican información morfológica y morfosintáctica[11] de un nodo origen.

Parte de esta información se puede materializar en una vista gráfica con el fin de obtener un *grafo de dependencias* [329, 330]. Así, por ejemplo, tras aplicar todo el proceso de análisis a la frase del Ejemplo 8.7 «*Feuilles à nervures denticulées*» («`Hojas con nervaduras dentadas`»), su resultado se podría resumir en el grafo de la Fig. 8.14.

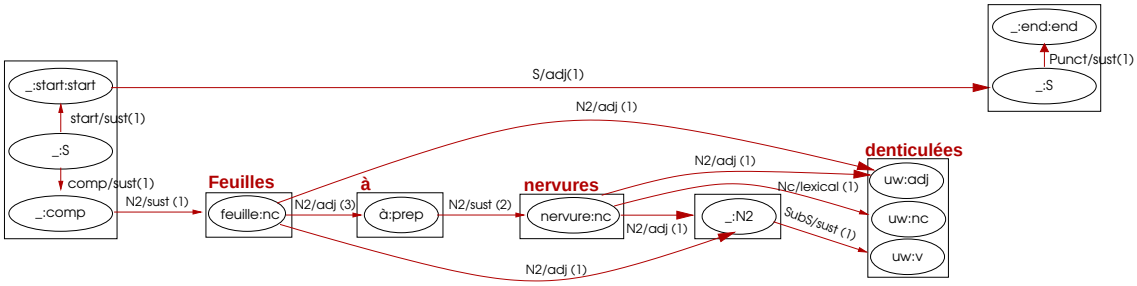


Figura 8.14: Grafo de dependencias

En él, se ha plasmado la información disponible a partir de la salida de FRMG PARSER utilizando el formato XML DEP, aunque hemos omitido alguna de la información relacionada con rasgos que describen a los nodos y arcos, con objeto de no hacerlos más tediosos.

Concretamente, el resultado se devuelve proporcionando cada *nodo*, representado mediante una elipse, con su etiqueta correspondiente que indica gráficamente la información recogida. Se trata del lema asociado y su categoría léxica.

[11] proporciona informaciones como la forma, categoría léxica, o los marcos de subcategorización.

Ejemplo 8.11 *Si se toma un pequeño fragmento de la Fig. 8.14, como el que se ilustra en la Fig. 8.15, se ve como el nodo* «feuille:nc» *tiene por lema a* «feuille» *y* «nc» *es su categoría léxica.*

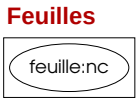


Figura 8.15: Nodo «*feuille:nc*» procedente de la Fig. 8.14

■

Además, cada nodo se encuentra incluido dentro de un *grupo* representado mediante un rectángulo. Esta estructura se refiere a una posición en la cadena de entrada e incluye a todos los posibles nodos asignados por el analizador a dicha posición, refiriéndose a la forma de la palabra considerada en cada caso. Así, una misma forma podría poseer diferentes categorías gramaticales, es decir, referirse a diferentes nodos tal y como se explica en el ejemplo 8.12.

Ejemplo 8.12 *Si se toma un pequeño fragmento de la Fig. 8.14, como el que se ilustra en la Fig. 8.12, se ve como la forma* «denticulées» (`«dentadas»`) *ocupa la cuarta posición en la frase. Además, su grupo incluye tres nodos:* «uw:adj», «uw:nc» *y* «uw:v», *donde* «uw» *indica que se trata de un lema desconocido, con las categorías léxicas adjetivo, sustantivo o verbo.*

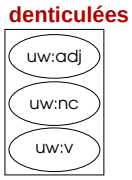


Figura 8.16: Grupo procedente de la Fig. 8.14

Si comparamos el resultado de la salida proporcionada por FRMG LEXER *en la figura del Ejemplo 7.12 y la de* FRMG PARSER *para el grupo representado en el Ejemplo 8.12, podemos observar como en el sintáctico se pueden llegar a descartar algunos de los nodos, por no estar involucrados en ninguna dependencia. Se trata de la de adverbio* (`adv`) *y de la de palabra extranjera* (`etr`).

■

Siguiendo con la Fig. 8.14, se observan dependencias binarias entre nodos, representadas por arcos dirigidos y etiquetados con la función sintáctica correspondiente y por la cantidad de derivaciones del nodo origen que apoya dicha dependencia. En función del tipo de operación que se realice sobre el árbol, la etiqueta del arco se representa de manera diferente. En particular, aquéllos que simbolizan operaciones de adjunción

se indican mediante *«/adj»*, y los que simbolizan operaciones de sustitución lo hacen mediante *«/sust»*. Finalmente, aquéllos referidos a la aparición de anclas lexicales, se caracterizan por *«/lexical»*. Estas anclas hacen referencia a aquéllas palabras que unifican correctamente con las hiperetiquetas de un determinado conjunto de árboles GA.

Ejemplo 8.13 *El fragmento de la Fig. 8.14 relativo a la dependencia que surge entre el nodo* «feuille:nc» *y* «à:prep» *posee una etiqueta* N2/adj *y está apoyada por tres derivaciones* (3)*, tal y como se muestra en la Fig. 8.17.*

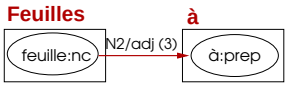


Figura 8.17: Dependencia con operación de adjunción entre *«feuille:nc»* y *«à:prep»*

Además, la etiqueta de la dependencia indica que la estructura sintáctica de la que depende el nodo «à:prep» *se ha insertado por adjunción en la estructura de la que depende* «feuille:nc».

■

Ejemplo 8.14 *Volviendo al fragmento de la Fig. 8.14, el nodo* «uw:nc» *tiene una función de ancla en la dependencia que la une con* «nervure:nc»*, tal y como se ilustra en la Fig. 8.18.*

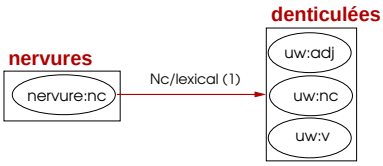


Figura 8.18: Dependencia con operación de anclaje entre *«nervure:nc»* y *«uw:nc»*

■

Ejemplo 8.15 *Retomando la Fig. 8.14, el fragmento relativo a la dependencia que surge entre el nodo* «à:prep» *y* «nervure:nc» *posee una etiqueta* N2/sust *y existen dos derivaciones que la apoyan* (2)*, tal y como se observa en la Fig. 8.19.*

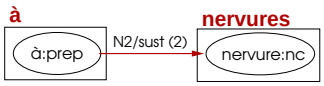


Figura 8.19: Dependencia con operación de sustitución entre *«à:prep»* y *«nervure:nc»*

Además, la etiqueta de la dependencia entre ambos nodos indica que la estructura sintáctica de la que depende «nervure:nc» *se ha insertado por sustitución en la estructura de la que depende* «à:prep».

■

Para facilitar el entendimiento de los grafos de dependencias, es necesario minimizarlos. Se trata de eliminar la información prescindible del grafo. En este sentido, es posible omitir todos aquellos nodos que hacen referencia a la raíz del árbol inicial, así como los referentes a los últimos grupos asociados a la finalización de la frase, se encuentre el signo de puntuación de manera explícito o no.

Ejemplo 8.16 *Si consideramos la Fig. 8.14, existe un elemento, representado mediante un grupo que se localiza a la izquierda del grafo, con tres nodos en su interior. El primero es* «_:start:start» *e indica que es la raíz de todos los árboles; el segundo es* «_:S» *y el tercero* «_:comp»*, tal y como se observa en la Fig. 8.20.*

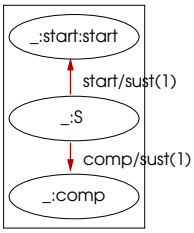


Figura 8.20: Grupo de inicio referente a la raíz del árbol

Del mismo modo, la estructura situada en el último lugar en el grafo, la cual hace referencia a la finalización de la frase, puede mostrarse de dos maneras diferentes, en función de si se explicita o no el signo de puntuación. En caso de no hacerlo, existe un único grupo con dos nodos, donde el primero es «_:S» *y el segundo es* «end:end»*, ilustrados en la Fig. 8.21.*

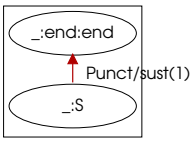


Figura 8.21: Grupos de finalización de frase sin explicitar el signo de puntuación

En caso de sí hacerlo, se representa mediante dos grupos relacionados cada uno con un nodo, tal y como se puede ver en la Fig. 8.22. Concretamente, el segundo es un ancla, ya que generalmente todas las frases finalizan con punto.

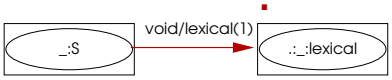


Figura 8.22: Grupos de finalización de frase explicitando el signo de puntuación

■

También existen estructuras que no se encuentran ni al principio ni al final del grafo, y que no poseen ni forma ni lema. Es lo que llamamos *puntos de anclaje* cuya principal misión es interconectar varios grupos entre sí. Su eliminación conlleva una serie de modificaciones sobre el grafo. En definitiva, se trata de relacionar directamente los nodos que enlazan con ellos, creando una única relación, lo que trasladaremos en la concatenación de las etiquetas de sus dependencias. En este sentido, el número de derivaciones de la nueva dependencia será el mínimo entre las de la primera y las de la segunda.

Ejemplo 8.17 *Si consideramos la Fig. 8.14, observamos como existe un grupo que no posee forma, situada ente los grupos cuyas formas son* «nervures» *y* «denticulées». *Se trata del punto de anclaje* «_:N2».

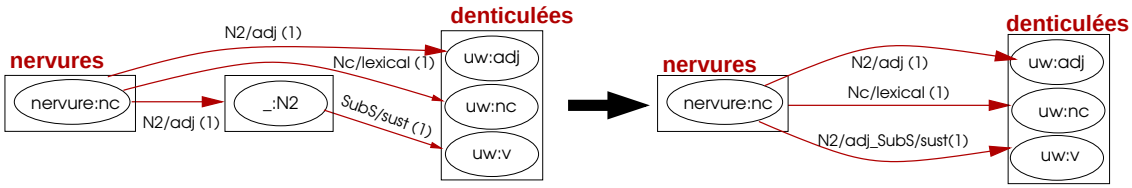


Figura 8.23: Punto de anclaje entre las formas *«nervure»* y *«denticulées»*

El resultado de unir ambos grupos y eliminar el tercero da lugar a una dependencia entre los nodos «nervure:nc» *y* «uw:v» *con la etiqueta* «N2/adj_Subs/sust». *El número resultante de derivaciones en la dependencia es* 1.

■

Aplicando las consideraciones descritas, conseguimos minimizar los grafos de dependencias hasta obtener lo que denominamos un *grafo inicial de dependencias* (GID). Así, a partir de la Fig. 8.14, obtenemos el GID de la Fig. 8.24.

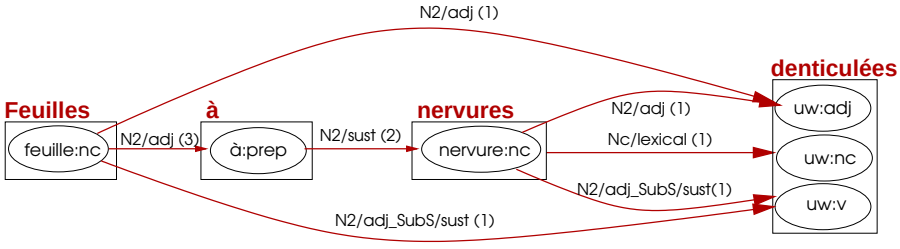


Figura 8.24: Grafo inicial de dependencias

Ejemplo 8.18 *Supongamos que partimos de la salida de* XML DEP *que se observa en la Fig. 8.25. Podemos ver como existe información acerca de la raíz del árbol, de*

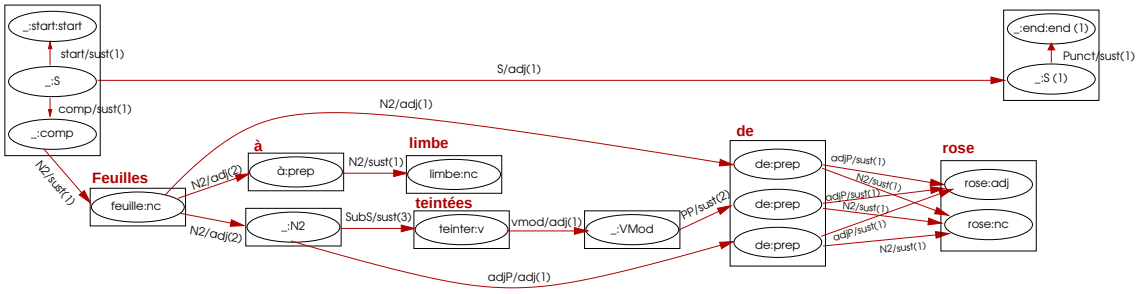


Figura 8.25: Ejemplo de grafo de dependencias

finalización de la frase, así como dos estructuras que no poseen ni forma, ni lema, ni categoría léxica. Se trata de los puntos de anclaje «_:N2» *y* «_:VMod».

Para eliminar éstos últimos del grafo, es necesario interconectar directamente entre sí los nodos involucrados en las dependencias asociadas. Es decir, el nodo origen del primer arco se une con el nodo destino del segundo, estableciendo como etiqueta de la nueva dependencia la concatenación de las dos primeras.

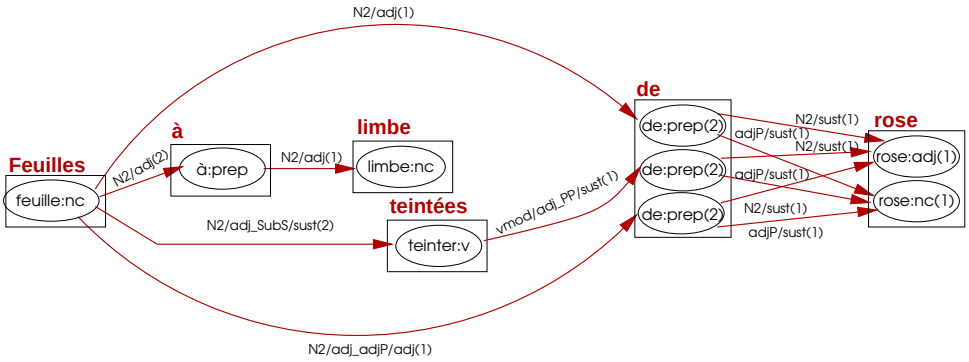


Figura 8.26: GID sin anclas vacías

Aplicando los cambios sobre el grafo, obtenemos el GID *de la Fig. 8.26. En ella observamos como las dependencias que iban del nodo* «feuille:nc» *al elemento* «_:N2», *y de él hasta* «teinter:v», *se ha transformado en una nueva dependencia, cuya etiqueta es* «N2/adj_subs/sust». *Además, el número de derivaciones es* 2, *ya que representa el mínimo entre el número de derivaciones que transitaban originalmente desde* «feuille:nc» *hacia el punto de anclaje, y el de ese punto hacia* «teinter:v».

Lo mismo ocurre con las dependencias que iban del nodo «feuille:nc» *a* «_:N2», *y de él al nodo* «de:prep», *que se han convertido en una nueva. Finalmente, las dependencias que iban del nodo* «teinter:v» *a* «_:VMod», *y de él al nodo* «de:prep», *también se han convertido en una nueva dependencia.*

■

8.6 | Almacenamiento y manejo de los GID's

Para facilitar la extracción de dependencias del *corpus* $\mathscr{B}$, se decidió crear una base de datos que poseyera toda la información derivable de los documentos en el análisis sintáctico tras la aplicación del formato XML DEP. El diagrama de entidad-relación mostrado en la Fig. 8.27, cuyas tablas asociadas son las mostradas en la Fig. 8.28, detalla el resultado.

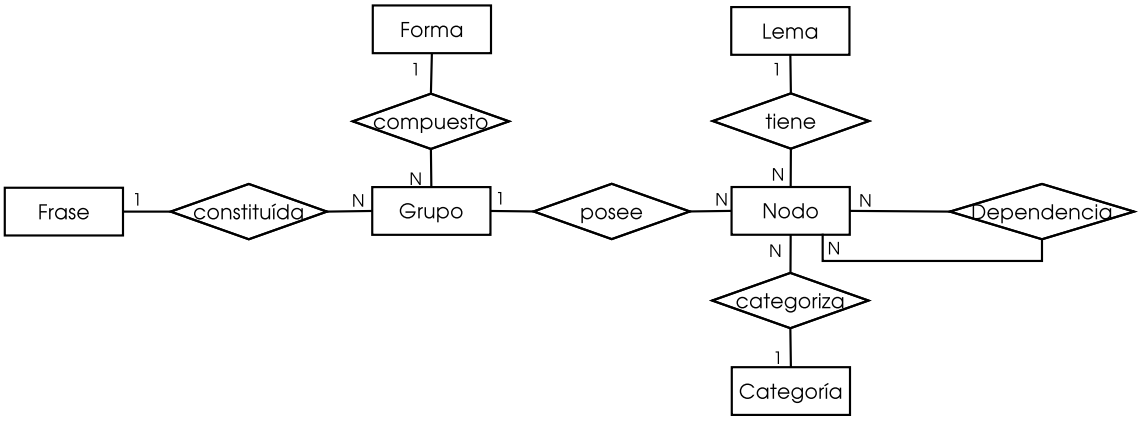


Figura 8.27: Base de datos creada

Podemos así observar las informaciones presentes en los GID's. Por ejemplo, la tabla *Frase* posee toda la información asociada a las oraciones del *corpus*, identificadas mediante el campo *id_frase*, mientras que las tablas *Forma* y *Categoría* hacen lo propio almacenando el conjunto de formas y categorías léxicas, respectivamente. En el caso de la tabla *Grupo*, ésta posee la información asociada a una determinada posición de la frase. Por este motivo, guarda constancia tanto del identificador de la frase como del identificador de la forma. Además, la localización de dicho grupo vendrá indicada gracias a su posición de comienzo por la izquierda, y la de fin por la derecha con los campos *a_left* y *a_right*. Del mismo modo, cada nodo se identificará mediante una categoría léxica, pero también se guardará un identificador del lema correspondiente, con el fin de saber cuales de ellos son conocidos y cuales no. Finalmente, la tabla *Dependencia*, es la encargada de representar las relaciones entre nodos, guardando los identificadores de los nodos de origen, los de fin, la etiqueta y el número de derivaciones que transitan por la dependencia a partir del nodo origen. Una vez introducidas las tablas, es fácil darse cuenta de que existe dos campos que no hemos mencionado. Es el caso de *ambigüedad* y *fallo* en la tabla *Frase*.

El primero hace referencia al grado de no determinismo de la frase en cuestión y representa información que no se encuentra reflejada en la salida de XML DEP, pero que sí podemos obtener mediante cálculos adicionales. Se estima, considerando el nivel de ambigüedad de los grupos que la conforman [328], mediante *la tasa de ambigüedad media*

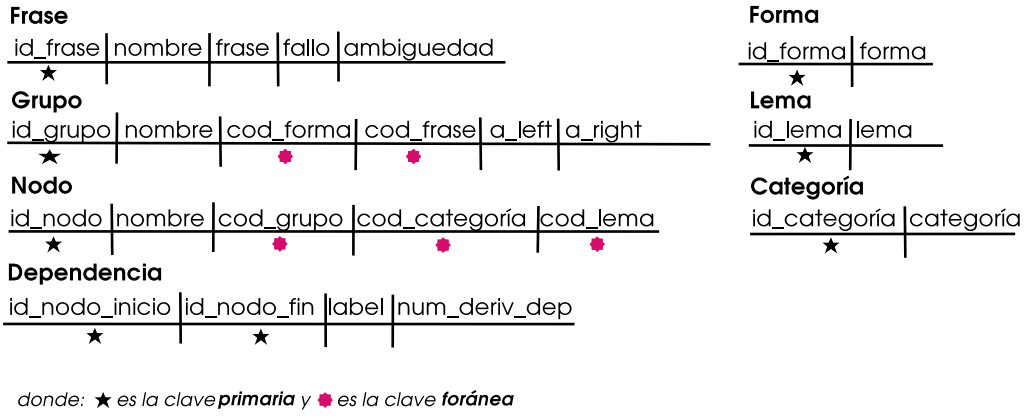


Figura 8.28: Tablas de la base de datos creada

por palabra, que se define como:

$$\alpha = \frac{1 + |\text{dependencias}_{s_i}|}{|\text{grupos}_{s_i}|} - 1 \ , \forall s_i \text{ tal que } 1 \leq i \leq n \tag{8.1}$$

donde s_i es una frase del *corpus*, $|\text{dependencias}_{s_i}|$ representa el número de dependencias existentes en s_i y $|\text{grupos}_{s_i}|$ el número de grupos para esa misma frase. En el caso de que no exista ambigüedad, existen tantos grupos como dependencias, por lo que $\alpha = 0$. En cambio, cuando existe ambigüedad, esta tasa va a indicar el número medio de dependencias que apuntan a mayores sobre un grupo destino. Así, una tasa de 1 indica que existen dos dependencias entrantes de media, dando lugar a 2^m análisis para una frase de longitud m.

Ejemplo 8.19 *Tomando el* GID *que se muestra en la Fig. 8.24, disponemos de un total de 4 grupos y 7 dependencias en total. En otras palabras, aplicando la medida α, toma el valor:*

$$\alpha = \frac{1+7}{4} - 1 = 1$$

■

El segundo hace referencia a un indicador que representa si el analizador ha proporcionado una salida adecuada o no. Así, para proporcionar más robustez y eficacia, se puede analizar imponiendo un plazo. Esto quiere decir que al finalizar éste, las respuestas se devuelven aunque los cálculos no se hayan terminado. Así, por ejemplo, en el caso de que transcurrido ese tiempo los cálculos realizados no se hayan terminado, y no llegase a devolver ningún tipo de análisis, aunque sean parciales, se provocará una desconexión por tiempo, lo que implicaría que no se devolviera ningún tipo de salida [329].

Teniendo en cuenta estos aspectos, a modo de resumen, podemos decir que hemos insertado unas 75.032 frases procedentes del *corpus* de botánica $\mathscr{B}$, de las cuales se

han podido analizar de un modo robusto unas 65.292. En relación a las restantes, o bien no proporcionaron una salida adecuada, o bien provocaron una desconexión. Muchas de las primeras son resultado del ruido introducido durante la fase de estructuración de los documentos, posterior al OCR. Por este motivo, el analizador sintáctico no ha sido capaz de establecer las dependencias adecuadas entre los nodos de los grupos.

Ejemplo 8.20 *Algunos ejemplos de frases no analizadas, es decir, con indicador de fallo, son:*

```
in Hooker , Syn .
- pl . pl . XXVI , 1 - 2 , p. 183 .
pl . pl . XXIX , 1 - 2 , p. 197 .
IFAN 28 : 179 , t. 33 , f. 4 - 5 ( 1953 ) , non Hook .
PL. XXVI , 4 - 5 , p. 183 .
Tardieu , Mém .
Bot . Bot. Fr. Fr. 55 : XLI ¡ 1908 ) .
- Calathea confería Benth 4 m , in Benth .
in Ledoux , Compt .
Bot . Bot . France g 3 : 202 , ig. 46 ) ;
in Wallich , PL Asiat .
Syntypes : Zenker 2250 , 2250 a , Cameroun .
in Hooker , Niger Fl. : 826 ( i 849 ) . ' ,
in herb . herb . Linné 674 , Amérique tropicale ( holo- , LINN ) .
in Dürand , Ind .
A signaler quç si le syntype Mann ¡gs [ non g¿s qpliß 1 m .
Afr . Afr . Exped. 1907 - 08 , 2 ;
p. 223 .
in C. Christensen , Ind .
Afr . Afr . Trop .
Lectotype : Vuillet 692 , Koulikoro , Mali ( P ! ) .
Bosser 12027 , la Réunion ; 3 , in litt ) .
- Rhaptopetalum scandens Pierre , ms . ms . in sched. sched. , P.
- Obermeyer , in Codd , De Winter & Rycroft , Fl .
- Buchenau , in Engt. , Pflanzenreich 16 ( IV.15 ) : 59 , tab . tab .
```

Como se puede constatar, todas estas frases no deberían de formar parte de las descripciones botánicas a analizar. En ellas se observa como algunas hacen referencia a la paginación del libro original, y otras forman parte del apartado de título. De este modo, se concluye que estas frases son simplemente ruido que se introduce en nuestro corpus.

■

A modo, de resumen, en las frases analizadas tenemos 1.200.303 grupos y 1.720.148 nodos, tal como se puede observar en la Fig. 8.29, donde cada bloque indica el tipo de elemento que se describe. En función de su posición, tenemos que distinguir aquéllos que se encuentran en el punto de partida de los árboles de análisis, representado en la figura como «`... en puntos de partida`», el de fin, identificados como «`... en puntos finales`», en los puntos de anclaje mostrados como «`... en anclas`» o en cualquier otra posición con «`... ninguna de las anteriores`». Así, de todos los grupos computados, unos 362.108 no poseen forma. De un modo más detallado, 64.949

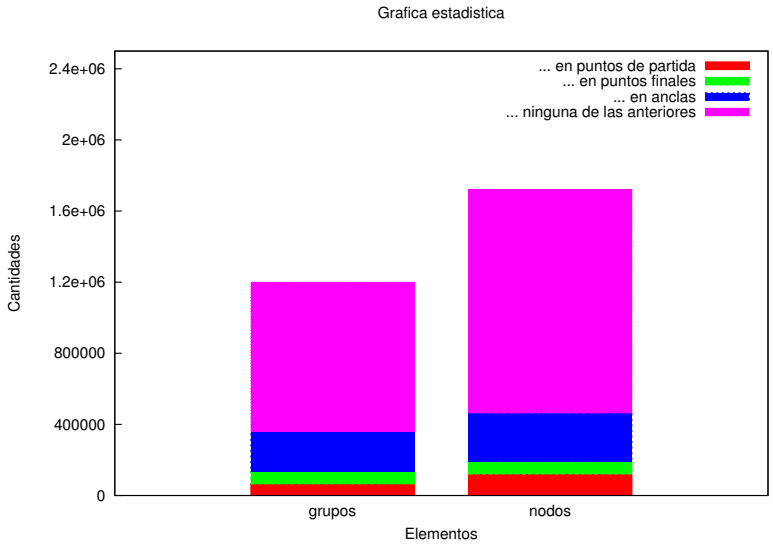


Figura 8.29: Gráfica acerca del origen de las agrupaciones y nodos

son grupos de punto de partida, 70.182 son de fin, y 226.957 son puntos de anclaje. En términos de nodos, existen 462.736 nodos que están incluidos dentro de los grupos descritos anteriormente. Dicho de otra forma, 120.303 están incluidos dentro de los de partida, 70.182 en los de finalización, y 272.251 son puntos de anclaje.

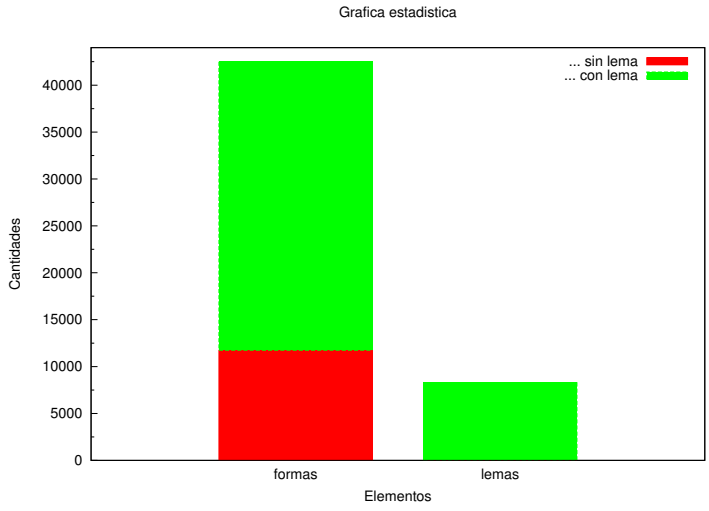


Figura 8.30: Cantidad de formas y lemas diferentes

Si eliminamos a ambos de la primera estimación se obtienen 838.195 grupos y 1.257.412 nodos, reflejados en la gráfica como porciones mayores. Esto quiere decir que cada frase posee de media aproximadamente 13 grupos. Y cada uno contiene de media 1,5 nodos. Otro dato que resulta interesante es el referente a la cantidad de palabras diferentes utilizadas. En este sentido, en nuestra base de datos hemos repertoriado unas 42.511

formas diferentes, de las cuales 11.743 no se han podido lematizar, generalmente debido a errores ortográficos que no se han solventado, dando lugar a palabras desconocidas, tal y como se observa en la Fig. 8.30. Además, entre las que sí se han podido lematizar, nos podemos encontrar con aproximadamente unos 8.282 lemas distintos. A continuación, en el Ejemplo 8.21 vamos a ilustrar algunas de las palabras que no fueron lematizadas debido tanto a errores ortográficos, como a su ausencia en el LEFFF.

Ejemplo 8.21 *Algunos ejemplos de palabras no lematizadas debidos a su ausencia en el* LEFFF*:*

```
verruqueux
loculicide
pollinaire
toruleux
cylindrico-conique
```

Algunos ejemplos de palabras no lematizadas debidos a errores ortográficos son :

```
quadripinnatiflde
sombre ^ luisant
pubescenïes
limbe©
iatteignant
sacciformeconique
elliptiquesoblong
ellip$'$tque
linéairetriangulaire
```

■

CAPÍTULO IX

Nivel semántico

Nuestro objetivo es la generación automática de GCB's directamente a partir de la colección documental, y con una intervención mínima por parte del usuario. En este sentido, fundamentaremos el nexo entre la sintaxis y la semántica en la *hipótesis distribucional de Harris* [130], que pone en relación el reparto sintáctico de las palabras con sus contenidos de información. Partimos pues de la suposición de que el significado de éstas y sus relaciones gramaticales están ligadas a las restricciones que se imponen sobre sus posibles combinaciones.

De un modo general, la gramática de un dominio concreto comparte su sintaxis con la de la lengua general y se distingue de ella por coocurrencias específicas en clases de palabras que le son propias. Así, la semántica de los términos se puede aproximar en base al estudio de sus contextos sintácticos, ya que sus posibles combinaciones vendrán dadas en función del ámbito de aplicación.

En esta línea, si el sentido de las palabras se deduce de las construcciones en las que aparecen [124], se desprende que si dos formas diferentes comparten contextos idénticos, sus sentidos son próximos. Esto nos permite abordar el tratamiento de un dominio específico, de tal manera que sea posible generalizar los patrones sintáctico-semánticos propios del sublenguaje a estudiar [25, 244].

La interpretación se sitúa en el origen de dos tipos de análisis distribucionales: el basado en un entorno gráfico y el basado en uno sintáctico. El primero considera que los contextos se sitúan entre las n palabras a la izquierda y derecha de una dada. El tratamiento estadístico de éstos permitirá resaltar aquéllos definidos como combinaciones de palabras dependientes de un dominio [224]. El segundo tipo de modelización se basa en el análisis sintáctico de relaciones de dependencia entre formas.

Ejemplo 9.1 *Supongamos que queremos comparar los dos tipos de análisis distribucionales tomando como base una frase extraída del* corpus *de ejemplo:*

«Sépales latéraux, oblongs, obovales, cochléiformes au sommet, avec un court apicule latéral dressé» (`«Sépalos laterales, oblongos, obóvalos, con forma de espiral en la parte superior, con un corto apículo lateral elevado»`).

Si nos centramos en el primer tipo de modelización y estudiamos los contextos de la palabra «cochléiforme» (`«en forma de espiral»`)*, suponiendo que estamos dispuestos a analizar las 3 palabras más a la izquierda y las 3 más a la derecha, se obtienen los que se observan en la Fig. 9.1.*

latéraux	oblongs	obovales	cochléiformes	à	sommet	avec
−3m	−2m	−1m	cochléiformes	+1m	+2m	+3m

Figura 9.1: Ejemplo de análisis basado en un contexto gráfico 3-gramas

En cambio, en el segundo tipo, si nos interesamos en las relaciones de tipo nombre-adjetivo, se obtienen las de la Fig. 9.2.

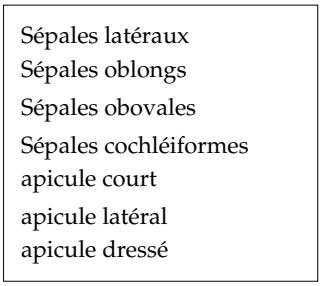


Figura 9.2: Ejemplo de dependencias sustantivo-adjetivo basado en análisis sintáctico

Comparando ambos, podemos constatar como en el primer acercamiento, «cochléiforme» *no posee como contexto a* «sépales» *en cambio en el segundo, sí.*

■

En nuestro caso, nos decidiremos por la segunda alternativa, donde ya disponemos de los GID's asociados al *corpus* $\mathscr{B}$ resultado del análisis sintáctico previo [327, 329, 330], los cuales nos permitirán identificar esos contextos. Para ello, no se considerarán todas las dependencias, sino que se realizará un paso previo de filtrado, inspirado en la metodología distribucional de Sager [261], restringiendo el estudio a los sintagmas nominales.

9.1 | Generación de dependencias gobernante/gobernado

Si observamos el GID de la Fig. 8.24, vemos como las unidades léxicas se vinculan dos a dos, obteniendo dependencias que relacionan formas entre sí del tipo [*Feuilles* $\rightarrow$ *à*] y [*à* $\rightarrow$ *nervures*] ([Hojas $\rightarrow$ con] y [con $\rightarrow$ nervaduras]). En este sentido, necesitamos que el análisis sintáctico se resuma en un *grafo de dependencias gobernante/gobernado* (GDGG) que compile las relaciones semánticas iniciales del texto analizado. Intuitivamente, se trata de que dichas relaciones binarias expresen el nexo entre un gobernante y un gobernado, o lo que es lo mismo, entre el núcleo y sus modificadores. De este modo, y siempre a partir de la Fig. 8.24, una posible dependencia sintáctica gobernante/gobernado sería [*Feuilles* $\overset{\text{à}}{\rightarrow}$ *nervures*] (Hojas $\overset{\text{à}}{\rightarrow}$ nervaduras).

Bajo esta perspectiva, el primer paso consiste en centrarnos en el GID para extraerlas. Dado que nuestra propuesta se basa en el estudio del régimen nominal, estamos interesados en destacar las que implican a sustantivos y adjetivos, pero también en otras que pasamos a enumerar:

- *Sustantivo-Adjetivo*: Se trata de un sustantivo modificado por un adjetivo en posición anterior y/o posterior. Este esquema captura las dependencias existentes entre los sintagmas adjetivales simples que anteceden y/o siguen al sustantivo, como en el caso de las enumeraciones. Son las más sencillas de extraer. Un ejemplo viene dado por la frase «*Sépales latéraux*» («Sépalos laterales»).

  Puede ocurrir que, en determinados casos, como en el de los adjetivos que provienen de participios, el sistema los haya etiquetado como verbos aunque éstos tengan una función de adjetivo. Por este motivo, es necesario también incluir las dependencias *Sustantivo-Verbo*.

- *Sustantivo-Preposición-Sustantivo*: Consiste en un sustantivo modificado por un único complemento nominal. El sustantivo puede estar a su vez modificado por sintagmas adjetivales, los cuales serán extraídos por los correspondientes patrones. Para ilustrarlo, tenemos la frase anterior «*feuille à nervure*» («hoja con nervadura»).

- *Adjetivo-Preposición-Adjetivo*: Radica en un adjetivo que es modificado por otro. El primero puede ser a su vez el modificador de un sustantivo. Al igual que en el caso anterior, este esquema extrae únicamente esta dependencia, permitiendo que sean los correspondientes patrones quienes se encarguen de las demás. Un ejemplo de este tipo de construcción es «*Pétales ovales à ovales-lancéolés*» («Pétalos óvalos a óvalos-lanceolados»).

  Tal y como comentábamos en el esquema *Sustantivo-Adjetivo*, puede ocurrir que determinados adjetivos estén etiquetados como verbos debido a su presencia como participio. Por este motivo, se han de incluir también los patrones siguientes

Adjetivo-Preposición-Verbo, *Verbo-Preposición-Adjetivo* y *Verbo-Preposición-Verbo*.

- *Sustantivo-Conjunción/Disyunción-Sustantivo*: Se trata de un sustantivo coordinado con otro mediante una conjunción/disyunción, como por ejemplo «*et*» («`y`») y «*ou*» («`ó`»). Este esquema cubre el caso de la existencia de la coordinación de dos sustantivos. Un ejemplo es «*Tige et feuilles*» («`Tallo y hojas`»).

- *Adjetivo-Conjunción/Disyunción-Adjetivo*: Consiste en un adjetivo coordinado con otro mediante una conjunción/disyunción, como por ejemplo «*et*» («`y`») y «*ou*» («`ó`»). Este esquema cubre el caso de la existencia de dos adjetivos coordinados. Un ejemplo es «*Tige verte et jaune*» («`Tallo verde y amarillo`»). Al igual que en los casos anteriores, también se han de incluir los siguientes patrones *Adjetivo-Conjunción-Verbo*, *Verbo-Conjunción-Adjetivo* y *Verbo-Conjunción-Verbo*.

- *Adjetivo-Adjetivo*: Radica en un adjetivo que modifica a otro adjetivo. Un ejemplo es «*Tige de couleur vert jaunâtre*» («`Tallo de color verde amarillento`»). También será necesario considerar los patrones *Adjetivo-Verbo*, *Verbo-Adjetivo* y *Verbo-Verbo*.

- *Sustantivo-Preposición-Adjetivo*: Se trata de un adjetivo que modifica a un sustantivo que no está presente en el texto, y éste a su vez modifica a otro. Esta construcción no es habitual en francés, nuestro lenguaje de ejemplo. Sin embargo, se trata de una estructura comúnmente empleada en lo que a botánica se refiere, lo que justifica su consideración. Un ejemplo es «*Tige de vert jaunâtre*» («`Tallo de verde amarillento`»). En este caso, se ha omitido el sustantivo «*couleur*» («`color`»). También se incluirá el patrón *Sustantivo-Preposición-Verbo*.

Estos esquemas son en realidad dependencias que tienen su lugar en la organización correspondiente al análisis del grupo nominal y son el resultado de la descomposición de un árbol sintáctico en dependencias binarias. De hecho, inspiradas por la estructura GID's, hemos desarrollado un método para extraerlas, descrito en la Tabla 9.1. Así, la función `Explora(Grafo,Esquema,Nodo de partida)` recorre el GID siguiendo el esquema pedido. El recorrido se para sobre un nodo N cuya categoría léxica es la permitida en base al esquema seguido. Llegados a este punto, la función `ExtraerLaDependencia(Nodo1,Nodo2)` tiene en cuenta sólo aquellas informaciones pasadas por parámetro que están unidas por alguna dependencia en el GID, siguiendo los esquemas mencionados anteriormente, guardando su posición y el camino realizado. Concretamente, esta función de exploración considera la especificidad de las etiquetas de las categorías léxicas y de las dependencias existentes entre nodos.

Sea `G`, el GID representado por una estructura de dependencias.
Sea `Tp`, una tabla de punteros sobre los `Nmax` nodos presentes en `G`.
Cada unidad léxica `N` de `G` se identifica por una constante arbitraria `id` que corresponde al elemento `Tp` de la tabla que apunta sobre ese nodo (`Tp[N → id].nodo=N`).

Para cada `X` de `0` a `Nmax` hacer{
    Para todos los `ESQUEMAS` hacer{
        `N = Explora(G,ESQUEMA,Tp[X].nodo);`
        Si (`N` no es `null`) entonces{
            `ExtraerLaDependencia(Tp[X].nodo,N);`
        }
    }
}

Tabla 9.1: Algoritmo de extracción de dependencias gobernante/gobernado

Ejemplo 9.2 *Retomando el* GID *de la Fig. 8.24 correspondiente a la frase* «Feuilles à nervures denticulées» (`«Hojas con nervaduras dentadas»`)*, extraeremos las dependencias gobernante/gobernado.*

La tabla de punteros recorrida está constituida de las siguientes entradas: {«feuille:nc», «à:prep», «nervure:nc», «uw:adj», «uw:nc», «uw:v»}. *Siguiendo el primer esquema, el nodo* «feuille:nc» *domina a* «uw:adj»*, mediante una dependencia etiquetada por* «N2/adj»*. De este modo, ésta será la primera dependencia gobernante/gobernado extraída, tal y como se observa en la Fig. 9.3 mediante las líneas discontinuas.*

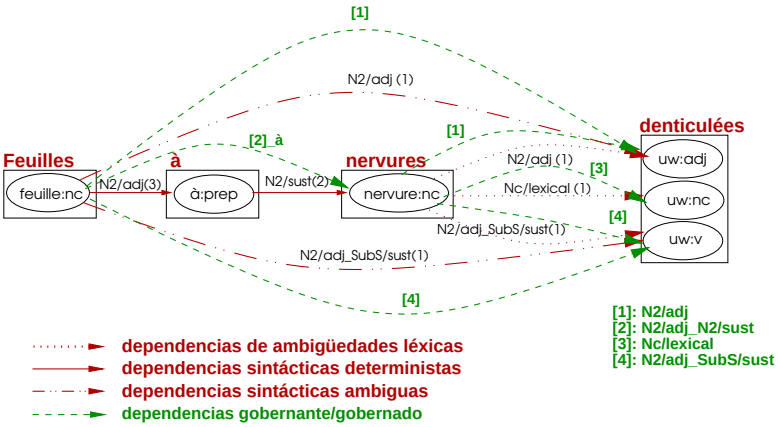


Figura 9.3: Ejemplo de dependencias gobernante/gobernado extraídas

Del mismo modo, y usando el segundo esquema, el nodo «feuille:nc» *domina a* «à:prep» *a través de la dependencia etiquetada por* «N2/adj»*, y éste a su vez domina al nodo*

«nervure:nc», *a través de la que se encuentra etiquetada por* «N2/sust». *Se creará entonces la dependencia entre los nodos* «feuille:nc» *y* «nervure:nc», *y su etiqueta será el resultado de la concatenación de las dos anteriores y de la preposición* «à» (`con`).

Si ahora tomásemos el GID *de la Fig. 8.26 correspondiente a la frase* «Feuilles à limbe teintées de rose» (`«Hojas con limbo teñidas de rosa»`), *las dependencias gobernante/gobernado serían las que se observan en la Fig. 9.4. Más en detalle, siguiendo*

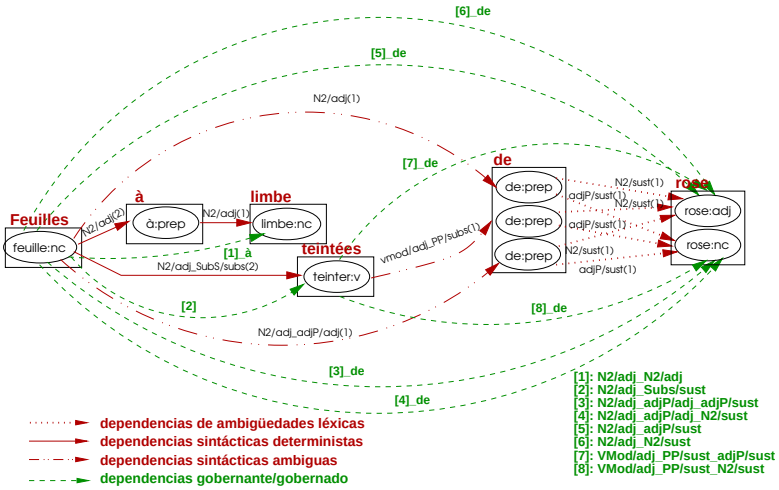


Figura 9.4: Otro ejemplo de dependencias gobernante/gobernado extraídas

el último esquema, el nodo «feuille:nc» *domina a* «de:prep» *a través de la dependencia etiquetada por* «N2/adj», *y éste a su vez domina al nodo* «rose:adj», *a través de la dependencia etiquetada por* «adjP/sust». *Se creará entonces una dependencia que una el primer nodo con el último, y cuya etiqueta será* «N2/adj_adjP/sust_de».

■

Estas nuevas dependencias constituyen el punto de partida para detectar conceptos relacionados y componen el GDGG.

9.2 | Adquisición de conocimiento

Construido el GDGG, ahora necesitamos denotar con fines descriptivos todas las posibles categorías léxicas para las ocurrencias de las formas que lo componen, introduciendo algunos detalles estructurales adicionales a fin de integrar, más adelante, datos semánticos.

Definición 9.1 *Sean* $\{s_i\}_{1\leq i\leq n}$ *la secuencia de frases de un* corpus $\mathscr{C}$ *y* $\Theta_{i,j}$, $1 \leq j \leq |s_i|$ *la ocurrencia de una* forma *en la* j*-ésima posición de la frase* s_i. *Se denota la asociación*

de una categoría léxica (a) y una clase semántica (b) con esa forma $\Theta_{i,j}$, por $\Theta_{i,j}^{a,b}$, y la denominamos término.

Del mismo modo, se introduce una notación utilizando una variable anónima, $\Theta_{i,j}^{a,\_}$, denominada token*, con el fin de designar al conjunto de términos sólo diferenciables por su clase semántica. En ese sentido, también se denota por $\Theta_{i,j}^{\_,\_}$ el conjunto de tokens referidos a la misma ocurrencia de una forma, denominada* agrupación*.*

Finalmente, se considera una notación mediante la utilización de variables libres, empleando para ello letras mayúsculas del final del abecedario, con el fin de enumerar rangos de valores. Así, por ejemplo, $\Theta_{i,j}^{a,X}$ se refiere al conjunto de términos en el token $\Theta_{i,j}^{a,\_}$, cuya clase semántica X sea aplicable en ese contexto. Además, esta notación puede ser extendida de un modo natural tanto a los tokens como a las agrupaciones.

■

Introducidos estos conceptos, identificaremos gráficamente las *agrupaciones* mediante rectángulos, los *tokens* mediante elipses y los *términos* mediante triángulos. De hecho, lo que se describe como agrupación y token posee cierta relación en la estructura creada de GDGG con los elementos grupo y nodo, respectivamente. Se trata de que, según esta definición, entendamos que una agrupación en una frase hace referencia a una posición en la cadena de entrada, la cual está en estrecha relación con la forma que simboliza, mientras que los tokens que los componen recogen la información léxica involucrada en las dependencias sintácticas extraídas en el GDGG. Finalmente, las clases semánticas asociadas a los términos serán introducidas más adelante.

Ejemplo 9.3 *Supongamos que partimos de la Fig. 9.3 del Ejemplo 9.2, considerando que dicha frase es la número 104 del* corpus *$\mathscr{B}$. En la Fig. 9.5 se observan cada una de las agrupaciones $\Theta_{104,j}^{\_,\_}$ presentes en ella, que a su vez hacen referencia a la forma $\Theta_{104,j}$. Además, cada token, representado por $\Theta_{i,j}^{a,\_}$, se caracteriza por poseer una categoría léxica.*

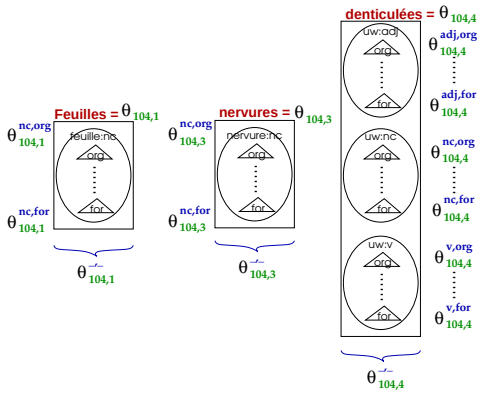


Figura 9.5: Notación léxica empleada para la frase «*Feuilles à nervures denticulées*»

De este modo, $\Theta^{-,-}_{104,1}$ *ilustra la forma* $\Theta_{104,1}$*, es decir, a* «Feuilles» (`«Hojas»`)*, y su único token se representa por* $\Theta^{nc,-}_{104,1}$*. En el caso de la agrupación con forma* «denticulées» (`«dentadas»`)*, el tercer token se representa mediante* $\Theta^{v,-}_{104,4}$*. Por último, la secuencia de tokens aceptados por* $\Theta^{X,-}_{104,4}$ *será* $\{\Theta^{nc,-}_{104,4}, \Theta^{adj,-}_{104,4}, \Theta^{v,-}_{104,4}\}$.

De igual manera, supongamos que partimos de la Fig. 9.4 del mismo ejemplo, considerando que dicha frase es la número 98 *del* corpus $\mathscr{B}$*. En la Fig. 9.6 se observan cada una de las agrupaciones* $\Theta^{-,-}_{98,j}$ *presentes en ella, así como sus tokens y términos.*

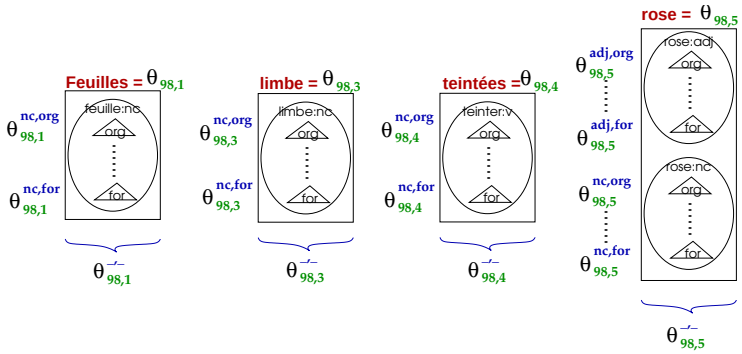


Figura 9.6: Notación léxica para la frase *«Feuilles à limbe teintées de rose»*

■

Ejemplo 9.4 *Retomemos la figura del Ejemplo 7.13, para la frase* «Feuilles de 3-4 cm» (`«Hojas de 3-4 cm»`)*, suponiendo que se trata de la frase* 15 *del* corpus $\mathscr{B}$*. Si aplicamos el análisis sintáctico sobre ella, y después extraemos las dependencias gobernante/gobernado, se obtiene el* GDGG *de la Fig. 9.7.*

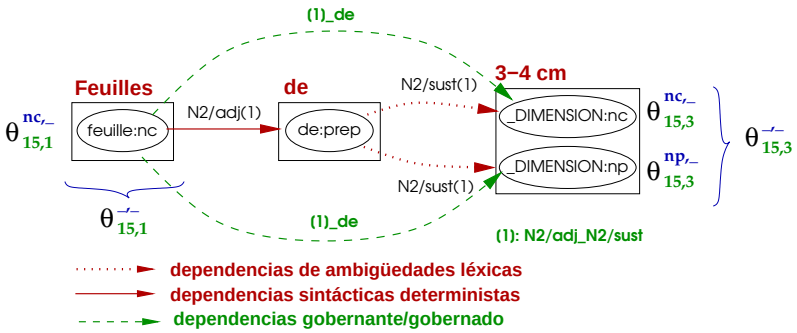


Figura 9.7: Notación léxica para la frase *«Feuilles de 3-4 cm»*

En ella podemos observar como cada $\Theta_{15,j}$ *hace referencia a una forma de la frase. Así, por ejemplo, la primera es* $\Theta_{15,1}$*. Por otro lado, cada una de ellas está asociada a una agrupación. Concretamente,* $\Theta^{-,-}_{15,3}$ *posee dos tokens. El primero es* $\Theta^{nc,-}_{15,3}$ *y representa la entrada asociada a la forma* «3-4cm» *con categoría léxica* «nc»*, mientras que la*

segunda es $\Theta^{np,\_}_{15,3}$ *con categoría léxica* «np»*, es decir, nombre propio. Hemos omitido la información relacionada con los términos simplemente con el propósito de no complicar más la figura.*

■

Volviendo a las figuras de los Ejemplos 9.2, 9.3 y 9.4, podemos observar el impacto que las ambigüedades, tanto de tipo léxico como sintáctico, generan en el número de posibles dependencias que han de pasar a la posterior fase de análisis semántico. En el primer caso, resulta clara su multiplicación en relación al número de tokens en una misma agrupación, esto es, al número de categorías léxicas asignables a una forma en una posición dada de una frase concreta del *corpus*. En este sentido, a menudo la etiquetación es una tarea no determinista y a veces incompleta, especialmente cuando se está tratando con un *corpus* enciclopédico con gran cantidad de palabras desconocidas. En esta situación, una forma de sugerir una etiqueta es a través del analizador sintáctico, guiándose por una estrategia predictiva basada en la gramática asociada.

Concretamente, este fenómeno lo podemos observar en la frase «*Feuilles à nervures denticulées*» («`Hojas con nervaduras dentadas`») de la Fig. 9.3, donde la forma «*denticulées*» («`dentadas`») está etiquetada con tres posibles categorías léxicas asociadas: verbo (`v`), adjetivo (`adj`) y nombre común (`nc`). Si lo comparamos con la Fig. 7.13, se ve como analizando sintácticamente se han descartado para esa agrupación dos posibles categorías léxicas: el adverbio (`adv`) y la palabra extranjera (`etr`). Lo mismo ocurre en el caso de la frase «*Feuilles à limbe teintées de rose*» («`Hojas con limbo teñidas de rosa`») de la Fig. 9.4, donde «*rose*» («`rosa`»), a pesar de no ser una palabra desconocida, puede hacer referencia a un sustantivo, refiriéndose a la planta, o por el contrario al adjetivo de color. Para evitar descartar interpretaciones útiles, deberíamos trasladar la resolución de estas ambigüedades, que no pueden ser resueltas a nivel lexical, a una fase posterior.

En el segundo caso, podemos observar un efecto análogo como resultado de la multiplicación de dependencias sobre los modificadores. Es el caso de *«denticulées»* («`dentadas`») como modificador bien de *«feuilles»* («`hojas`») o bien de *«nervures»* («`nervaduras`») en la Fig. 9.3. Debido a que ambos coinciden en género y número, existe el mismo número de arcos que los unen con la forma $\Theta_{104,4}$, o sea «*denticulées*» («`dentadas`»), para el caso de adjetivo y el de verbo. Se trata en este caso de un fenómeno conocido y ligado a la asociación de complementos preposicionales a un sintagma nominal, que aquí proporciona dos posibles interpretaciones: «`hojas con -nervaduras dentadas-`» o, alternativamente, «`-hojas dentadas- con nervaduras`». A causa de la ambigüedad léxica existente, no se tiene claro si «*denticulées*» («`dentadas`») resulta ser:

- Un modificador de una u otra palabra, considerando que realiza función de adjetivo, es decir, que la «`hoja`» o la «`nervadura`» tienen la propiedad de ser «`dentada`».

- Un sustantivo que complementa a otro sustantivo, como en el caso de «`nervaduras dentadas`». Este caso es muy común cuando se trata de nombres compuestos como por ejemplo en la frase «*Jardines con hierba luisa*», donde «*hierba luisa*» hace referencia a un arbusto de 3 a 7 m de altura con tallos leñosos en la parte superior.

- El participio correspondiente en frases con la ausencia del verbo atributivo «*être*» («`ser/estar`») para el caso de la voz pasiva. Así, la frase podría ser «*les feuilles/nervures sont denticulées*» («`las hojas/nervaduras son dentadas`»). Hay que destacar que en francés no existe el verbo «*denticuler*», aunque si bien es cierto, generalmente todos los participios de este idioma terminan en «*-é*» para la tercera forma del singular masculina y «*-és*» para el plural, «*-ée*» para la tercera forma del singular femenino y «*-ées*» para el plural.

Tomando ahora como ejemplo la frase francesa «*feuilles à limbe teintées de rose*» («`hojas con limbo teñidas de rosa`») de la Fig. 9.4, ésta se puede interpretar de varias maneras:

- La primera podría ser como «`-hojas de rosa- teñidas con limbo`», es decir, las hojas ya tienen el color rosa y a su vez se encuentran teñidas.

- La segunda como «`-hojas de rosa- teñidas con limbo`», es decir, las hojas son de la planta rosa.

- La tercera como «`hojas -teñidas de rosa- con limbo`», es decir, las hojas están teñidas del color rosa.

- La cuarta como «`hojas -teñidas de rosa- con limbo`», es decir, las hojas están teñidas por la planta rosa.

Realmente, existen otras dos interpretaciones más, aunque éstas coinciden con las dos primeras. Lo único en lo que varían es en el tipo de árbol que las ha creado. Es decir, se llega a las mismas interpretaciones, pero mediante dos árboles diferentes y, por lo tanto, con diferentes derivaciones.

Con respecto a esto, mientras las ambigüedades léxicas sólo dependen de la estructura del lenguaje, las sintácticas están fuertemente influenciadas por el formalismo gramatical elegido para describirlo, por la gramática particular considerada y por la falta de una cobertura gramatical completa. Existen incluso no pocas situaciones en las que las ambigüedades han de resolverse forzosamente a nivel semántico, toda vez que su origen puede no ser ni de naturaleza léxica ni sintáctica. Un ejemplo clásico es el uso de estructuras de coordinación relacionando entidades con una lista de adjetivos [258], como en la frase *«des sépales ovales-aigus, glabres ou éparsement hérissés»* (`«sépalos ovalados-agudos, glabros`

`o dispersamente espinosos»`), donde la propiedad *«hérissés»* (`«espinosos»`) se podría unir al adjetivo *«glabres»* (`«glabros»`) o a *«ovales-aigus»* (`«ovalados-agudos»`). En este caso, sólo hay una forma de resolver el problema, y pasa por conocer la naturaleza exacta de los órganos de las plantas, algo que nada tiene que ver ni con la morfología ni con la gramática del lenguaje.

Así, el fenómeno de la ambigüedad puede entenderse como una ilustración de la complejidad del lenguaje en sí mismo [240], siendo éste un problema fundamental a resolver en PLN. En estas condiciones, es difícil estimar el conjunto de esquemas sintácticos asociados al no determinismo, lo cual podría complicar un acercamiento analítico para resolver el problema. Afortunadamente, existe una condición topológica que resulta ser fácilmente detectable y que lo caracteriza completamente en grafos de dependencias, independientemente de su origen. De un modo más detallado, una ambigüedad se corresponde con una situación donde un token gobernado tiene más de un gobernante. Esto proporciona, a su vez, un mecanismo sencillo para solucionar la cuestión, a saber, se trata de filtrar las dependencias menos plausibles en favor de las que lo son más, asegurando de este modo que un token gobernado tenga únicamente un gobernante. Así, por ejemplo, volviendo al ejemplo, «*denticulées*» (`«dentadas»`) está gobernada por «*Feuilles*» (`«Hojas»`), pero también por «*nervures*» (`«nervaduras»`). El sistema debería dar prioridad a una de esas dependencias.

A este respecto, no se considera ninguna otra restricción topológica y, por lo tanto, un token gobernante puede tener más de un gobernado, como es el caso en la Fig. 9.4, donde la forma «*Feuilles*» (`«Hojas»`) gobierna a «*limbe*» (`«limbo»`) y «*teintées*» (`«tintadas»`). Además, un token puede ser gobernante y gobernado simultáneamente, como es el caso en la misma figura, donde la forma «*teintées*» (`«tintadas»`) está gobernando a «*rose*» (`«rosa»`), pero a su vez está siendo gobernado por «*Feuilles*» (`«Hojas»`).

Sin embargo, la materialización de esta idea no resulta ser tan sencilla. La mayoría de las ambigüedades pasan inadvertidas, ya que los humanos somos muy hábiles a la hora de resolverlas gracias a un amplio conocimiento del contexto y del mundo, mientras que los sistemas informáticos no tienen plena capacidad en ese terreno. Como consecuencia, a menudo no realizan un buen trabajo de desambiguación [309]. Por este motivo, es necesario recurrir a otro tipo de mecanismo. Lo que queremos es priorizar estas relaciones para extraer de forma efectiva la semántica del texto. Intuitivamente, el proceso consistirá en recopilar información a partir del *corpus* con el objetivo de detectar aquellas dependencias que resulten más plausibles. Técnicamente, la heurística propuesta se organiza en tres niveles de complejidad. Los dos primeros están concebidos para explotar la secuencia de estructuras resultantes de las fases previas de análisis léxico y sintáctico, clasificando en orden de prioridad las ambigüedades correspondientes. El tercer nivel determinará que información semántica está involucrada en cada una de las dependencias.

Para conseguir este objetivo, es necesario introducir una notación específica, ya que deberemos extrapolar nuestras estimaciones desde un contexto local hacia uno global. Así, los datos obtenidos inicialmente de las frases deben ser combinados y evaluados a lo largo de todo el *corpus* con el fin de extraer nuevas conclusiones susceptibles de ser de nuevo aplicadas en cada frase, para luego recomenzar iterativamente el proceso. Deberíamos entonces hablar de *términos, tokens* y *agrupaciones plausibles*, nociones que extenderán los conceptos del mismo nombre desde el nivel local a uno de *corpus*.

Definición 9.2 *Sean* $\{s_i\}_{1\leq i\leq n}$ *la secuencia de frases de un* corpus $\mathscr{C}$ *y* $\Theta_{i,j}$, $1 \leq j \leq |s_i|$ *la ocurrencia de una forma en la* j*-ésima posición de la frase* s_i*. Se denota la asociación de la categoría léxica (*a*) y la clase semántica (*b*) con esa forma* $\Theta_{i,j}$*, por* $\tilde{\Theta}_{i,j}^{a,b}$*, llamado* término plausible*.*

Esta notación puede ser extendida aquí explotando la utilización de las variables anónimas (resp. las variables libres) previamente introducidas para términos, tokens y agrupaciones en la Definición 9.1.

■

Será necesario igualmente proveernos de la notación para la gestión de dependencias gobernante/gobernado a nivel de frase (resp. de *corpus*). A este respecto, habremos de referirnos tanto a las transiciones entre tokens (resp. tokens plausibles) que constituyen la salida proporcionada en los GID's por el analizador sintáctico, como a los conjuntos de transiciones entre tokens de dos agrupaciones (resp. agrupaciones plausibles) diferentes. Finalmente, ya en la fase de categorización semántica consideraremos el tratamiento de transiciones entre términos (resp. términos plausibles).

Definición 9.3 *Sea* $s_i,\ 1 \leq i \leq n$ *la* i*-ésima frase de un* corpus $\mathscr{C}$ *y* τ *la secuencia de reglas gramaticales necesarias para generar el token* $\Theta_{i,k}^{c,\bar{-}}$ *a partir del token* $\Theta_{i,j}^{a,\bar{-}}$ *en el* GDGG*. Se denota la dependencia entre los tokens* $\Theta_{i,j}^{a,\bar{-}}$ *y* $\Theta_{i,k}^{c,\bar{-}}$*, etiquetada por* τ *como* $\delta^{\theta_{i,j}^{a,\bar{-}},\tau,\theta_{i,k}^{c,\bar{-}}}$.

La notación puede extenderse naturalmente a los términos, agrupaciones y estructuras plausibles mediante la utilización de la notación previamente introducida de las variables anónimas. Cuando una dependencia relaciona estructuras plausibles, se habla de dependencias plausibles*.*

■

Con el fin de facilitar la comprensión de las sucesivas secciones, la Tabla 9.2 recoge a modo de recordatorio toda aquélla notación previamente introducida para los elementos que intervienen en los GDGG's, tanto a nivel de frase como de *corpus*. Finalmente, la Tabla 9.3 representa aquella notación utilizada en los cálculos iterativos a realizar, para tratar la desambiguación y el aprendizaje, que introduciremos a medida que expliquemos nuestra propuesta.

Representación	**Explicación**
s_i	La i-ésima frase de un *corpus* $\mathscr{C}$, donde $1 \leqslant i \leqslant n$.
$\|L\|$	El cardinal del conjunto L.
$\mathscr{T}$	El conjunto de clases semánticas asociadas a $\mathscr{C}$.
$\mathscr{F}$	El conjunto de formas semánticas asociadas a $\mathscr{T}$. Del mismo modo, se expresa mediante $\mathscr{F}(b)$ al subconjunto de formas asociadas a $b \in \mathscr{T}$.
$\Theta_{i,j}$	La ocurrencia de la forma en la j-ésima posición de la frase s_i, donde $1 \leqslant j \leqslant \|s_i\|$.
$\Theta_{i,C}^{A,B}$	Las variables A y B pueden ser: - Instancias, representadas por letras minúsculas. - Variables anónimas, representadas por «_». - Variables cuantificables en un rango, que siempre se expresarán por una letra mayúscula del final del abecedario, con el fin de enumerar un rango. Sin embargo, C sólo va a poder ser una variable cuantificable en un rango o una instancia. En función de los valores de éstas, tendrá un significado u otro: $\Theta_{i,j}^{a,b}$: La asociación de una categoría léxica a y una clase semántica b a una forma $\Theta_{i,j}$ en s_i, denominado *término*. $\Theta_{i,j}^{a,\_}$: El conjunto de términos sólo diferenciables por su clase semántica, denominado *token*. $\Theta_{i,j}^{\_,\_}$: El conjunto de tokens referidos a la ocurrencia de una forma $\Theta_{i,j}$, denominada *agrupación*. $\Theta_{i,j}^{a,X}$: La secuencia de términos del token $\Theta_{i,j}^{a,\_}$, cuya clase semántica X es aplicable en ese contexto. $\Theta_{i,j}^{X,Y}$: La secuencia de términos de la agrupación $\Theta_{i,j}^{\_,\_}$, cuya categoría léxica X y clase semántica Y son aplicables en ese contexto. $\Theta_{i,j}^{X,\_}$: La secuencia de tokens de la agrupación $\Theta_{i,j}^{\_,\_}$, cuya categoría léxica X es aplicable en ese contexto. $\Theta_{i,X}^{\_,\_}$: La secuencia de agrupaciones de s_i, donde $X \in [1, \|s_i\|]$ es aplicable en ese contexto. $\Theta_{i,j}^{X,b}$: La secuencia de términos con clase semántica b de la agrupación $\Theta_{i,j}^{\_,\_}$, cuya categoría léxica X es aplicable en ese contexto.
$\delta^{\Theta_{i,C}^{A,B},G,\Theta_{i,F}^{D,E}}$	Son las dependencias entre dos elementos $\Theta_{i,C}^{A,B}$ y $\Theta_{i,F}^{D,E}$ en s_i. Así, se conoce a $\Theta_{i,C}^{A,B}$ como el *gobernante* y a $\Theta_{i,F}^{D,E}$ como el *gobernado*. Por su parte, G simboliza la etiqueta de la dependencia y puede ser: - Una instancia, representada en este caso por una letra griega.

	- Una variable cuantificable en un rango, y que siempre se representará por una letra mayúscula del final del abecedario. Sin embargo, G no podrá ser en ningún caso una variable anónima, ya que todas las dependencias deben poseerla con algún valor. En función de los valores, tendrá un significado u otro. Por ejemplo, $\delta^{\Theta^{a,\_}_{i,\_},\tau,\Theta^{b,\_}_{i,k}}$: La dependencia entre los tokens $\Theta^{a,\_}_{i,\_}$ y $\Theta^{b,\_}_{i,k}$, con etiqueta τ. $\delta^{\Theta^{a,b}_{i,j},\tau,\Theta^{c,d}_{i,k}}$: La dependencia entre los términos $\Theta^{a,b}_{i,j}$ y $\Theta^{c,d}_{i,k}$, con etiqueta τ. $\delta^{\Theta^{a,X}_{i,j},\tau,\Theta^{b,Y}_{i,k}}$: La secuencia de dependencias entre los términos $\Theta^{a,X}_{i,j}$ y $\Theta^{b,Y}_{i,k}$, con etiqueta τ. $\delta^{\Theta^{a,b}_{i,j},T,\Theta^{c,d}_{i,k}}$: La secuencia de dependencias entre los términos $\Theta^{a,b}_{i,j}$ y $\Theta^{c,d}_{i,k}$, cuya etiqueta T es aplicable en ese contexto. $\delta^{\Theta^{X,Y}_{i,Z},T,\Theta^{V,W}_{i,k}}$: La secuencia de dependencias entre los términos $\Theta^{X,Y}_{i,Z}$ y los existentes en la agrupación $\Theta^{\_,\_}_{i,k}$, cuya etiqueta T es aplicable en ese contexto.
$\widetilde{\Theta}^{A,B}_{i,C}$	Las variables A y B pueden ser: - Instancias, representadas por letras minúsculas. - Variables anónimas, representadas por «_». - Variables cuantificables en un rango, que siempre se expresarán por una letra mayúscula del final del abecedario, con el fin de enumerar un rango. Sin embargo, C sólo va a poder ser una variable cuantificable en un rango o una instancia. En función de los valores de éstas, tendrá un significado u otro: $\widetilde{\Theta}^{a,b}_{i,j}$: La asociación de una categoría léxica a y una clase semántica b a una forma $\Theta_{i,j}$ en $\mathscr{C}$, denominado *término plausible*. $\widetilde{\Theta}^{a,\_}_{i,j}$: El conjunto de términos plausibles en $\mathscr{C}$ sólo diferenciables por su clase semántica, denominado *token plausible*. $\widetilde{\Theta}^{\_,\_}_{i,j}$: El conjunto de tokens plausibles referidos a la ocurrencia de una forma$\Theta_{i,j}$ en $\mathscr{C}$, denominada *agrupación plausible*. $\widetilde{\Theta}^{a,X}_{i,j}$: La secuencia de términos plausibles del token $\widetilde{\Theta}^{a,\_}_{i,j}$, cuya clase semántica X sea aplicable en el contexto $\mathscr{C}$. $\widetilde{\Theta}^{X,Y}_{i,j}$: La secuencia de términos plausibles de la agrupación $\widetilde{\Theta}^{\_,\_}_{i,j}$, cuya categoría léxica X y clase semántica Y sean aplicables en el contexto de $\mathscr{C}$. $\widetilde{\Theta}^{X,\_}_{i,j}$: La secuencia de tokens plausibles de la agrupación $\widetilde{\Theta}^{\_,\_}_{i,j}$, cuya categoría léxica X sea aplicable en el contexto de $\mathscr{C}$.
$\delta^{\widetilde{\Theta}^{A,B}_{i,C},G,\widetilde{\Theta}^{D,E}_{i,F}}$)	Es la dependencia entre dos elementos $\widetilde{\Theta}^{A,B}_{i,C}$ y $\widetilde{\Theta}^{D,E}_{i,F}$ en $\mathscr{C}$, denominada *dependencia plausible*. Así, se conoce a $\widetilde{\Theta}^{A,B}_{i,C}$ como el *gobernante plausible*, y a $\widetilde{\Theta}^{D,E}_{i,F}$ como el *gobernado plausible*.

Tabla 9.2: Notación de los componentes del GDGG a nivel local y global

Representación	Explicación
$P(\Theta_{i,C}^{A,B})_{\text{local}(z)}$	La probabilidad de $\Theta_{i,C}^{A,B}$ en la frase s_i, durante la iteración z.
$W(\delta^{\Theta_{i,j}^{a,\_},\tau,\Theta_{i,k}^{b,\_}})$	El peso inicial de la dependencia $\delta^{\Theta_{i,j}^{a,\_},\tau,\Theta_{i,k}^{b,\_}}$ en la frase s_i.
$P(\delta^{\Theta_{i,C}^{A,B},G,\Theta_{i,F}^{D,E}})_{\text{local}(z)}$	La probabilidad de la dependencia $\delta^{\Theta_{i,C}^{A,B},G,\Theta_{i,F}^{D,E}}$ en la frase s_i, durante la iteración z.
$P(\widetilde{\Theta}_{i,C}^{A,B})_{\text{global}(z)}$	La probabilidad de $\widetilde{\Theta}_{i,C}^{A,B}$ en el *corpus* $\mathscr{C}$ durante la iteración z.
$P(\delta^{\widetilde{\Theta}_{i,C}^{A,B},G,\widetilde{\Theta}_{i,F}^{D,E}})_{\text{global}(z)}$	La probabilidad de la dependencia $\delta^{\widetilde{\Theta}_{i,C}^{A,B},G,\widetilde{\Theta}_{i,F}^{D,E}}$ en el *corpus* $\mathscr{C}$ durante la iteración z.

Tabla 9.3: Notación para la representación de los pesos de los diversos componentes

Planteada la notación, estamos en disposición de precisar la heurística propuesta de tres niveles de complejidad.

9.2.1 | Categorización de los tokens

El objetivo es calcular, para cada agrupación del texto, cual es el token más probable. Es decir, para cada frase del *corpus*, queremos determinar la categoría léxica de cada una de las ocurrencias de las formas que ahí figuren. El proceso, iterativo, se corresponde con las ecuaciones de la Tabla 9.4, que pasamos a comentar:

$$P(\Theta_{i,j}^{a,\_})_{\text{local(0)}} = \frac{1}{|\{\Theta_{i,j}^{X,\_}\}|} \tag{9.1}$$

$$P(\tilde{\Theta}_{i,j}^{a,\_})_{\text{global(n+1)}} = \frac{\sum_{\Theta_{k,l}=\Theta_{i,j}} P(\Theta_{k,l}^{a,\_})_{\text{local(n)}}}{\sum_{\Theta_{k,l}^{X,\_},\Theta_{k,l}=\Theta_{i,j}} P(\Theta_{k,l}^{X,\_})_{\text{local(n)}}} \tag{9.2}$$

$$P(\Theta_{i,j}^{a,\_})_{\text{local(n+1)}} = \frac{P(\tilde{\Theta}_{i,j}^{a,\_})_{\text{global(n+1)}}}{\sum_{\Theta_{k,l}^{X,\_},\Theta_{k,l}=\Theta_{i,j}} P(\tilde{\Theta}_{k,l}^{X,\_})_{\text{global(n+1)}}} \tag{9.3}$$

Tabla 9.4: Modelo para la categorización de tokens

(9.1). El proceso se inicia con el cálculo de la probabilidad local a nivel de frase, asociable a un token en una agrupación. Se trata de un simple *ratio* en razón al número de tokens que involucran a dicha agrupación. Obviamente, si sólo existe un token en la agrupación, su probabilidad será de 1.

(9.2). Define la probabilidad global en el *corpus* de un token plausible, en la iteración $n+1$ del proceso. Se calcula como una proporción de la probabilidad local asociada a tokens con la misma categoría léxica y forma que la del token considerado, en relación a la probabilidad cuando la categoría léxica es libre.

(9.3). Establece el valor de la probabilidad local asociable a un token en una agrupación, en la iteración $n+1$ del proceso. Para ello, se repercuten las probabilidades calculadas globalmente, distribuyéndolas proporcionalmente entre las globales de los tokens plausibles asociados a la agrupación.

El proceso iterativo continúa hasta la convergencia [267] sobre un punto fijo, o sobre un umbral prefijado de aproximación. Nos serviremos de un ejemplo para su ilustración.

Ejemplo 9.5 *Supongamos que queremos calcular la probabilidad de cada una de las ocurrencias* $\Theta^{a,-}_{104,j}$ *de la Fig. 9.5. Si nos centramos única y exclusivamente en el caso de la forma* «denticulées» (`«dentadas»`)*, la probabilidad local inicial del token con categoría léxica* adj *es la que viene expresada en la Ecuación 9.4.*

$$P(\Theta^{adj,-}_{104,4})_{\text{local(0)}} = \frac{1}{|\{\Theta^{X,-}_{104,4}\}|} = \frac{1}{|\{\Theta^{adj,-}_{104,4}, \Theta^{nc,-}_{104,4}, \Theta^{v,-}_{104,4}\}|} = \frac{1}{3} \tag{9.4}$$

Una vez calculado este valor local, estimaremos la probabilidad global de ese mismo token en el corpus $\mathscr{B}$*, para la primera iteración, tal y como se ilustra en la Ecuación 9.5. En ella,* $P(\widetilde{\Theta}^{adj,-}_{104,4})_{\text{global(1)}}$ *se expresa como un ratio entre el sumatorio de todas las probabilidades locales de los tokens cuya forma es* «denticulées» (`«dentadas»`) *y categoría léxica es* «adj»*, y el sumatorio de todas aquéllas probabilidades locales de tokens con misma forma, y entre los que se encuentra* $P(\Theta^{adj,-}_{104,4})_{\text{local(0)}}$.

$$P(\tilde{\Theta}^{adj,-}_{104,4})_{\text{global(1)}} = \frac{\sum_{\Theta_{k,l}=\text{denticulées}} P(\Theta^{adj,-}_{k,l})_{\text{local(0)}}}{\sum_{\Theta^{X,-}_{k,l},\Theta_{k,l}=\text{denticulées}} P(\Theta^{X,-}_{k,l})_{\text{local(0)}}} \tag{9.5}$$

Finalmente, se calculará la probabilidad local para la primera iteración de la ocurrencia $\Theta^{adj,-}_{104,4}$*, realizando una normalización con respecto a todas las posibles categorías léxicas que tiene en cuenta dicha agrupación, tal y como se muestra en la Ecuación 9.6. Una vez obtenido, este valor será utilizado para calcular las sucesivas iteraciones.*

$$P(\Theta^{adj,-}_{104,4})_{\text{local(1)}} = \frac{P(\tilde{\Theta}^{adj,-}_{104,4})_{\text{global(1)}}}{\sum_{\Theta^{X,-}_{k,l},\Theta_{k,l}=\text{denticulées}} P(\tilde{\Theta}^{X,-}_{k,l})_{\text{global(1)}}} = \frac{P(\tilde{\Theta}^{adj,-}_{104,4})_{\text{global(1)}}}{P(\tilde{\Theta}^{adj,-}_{104,4})_{\text{global(1)}} + \cdots + P(\tilde{\Theta}^{v,-}_{104,4})_{\text{global(1)}}} \tag{9.6}$$

En este sentido, la Fig. 9.8 ilustra el cálculo realizado para cada uno de los tokens de $\Theta^{\cdot,-}_{104,j}$ *utilizando tres columnas, una por cada uno de los pasos introducidos.*

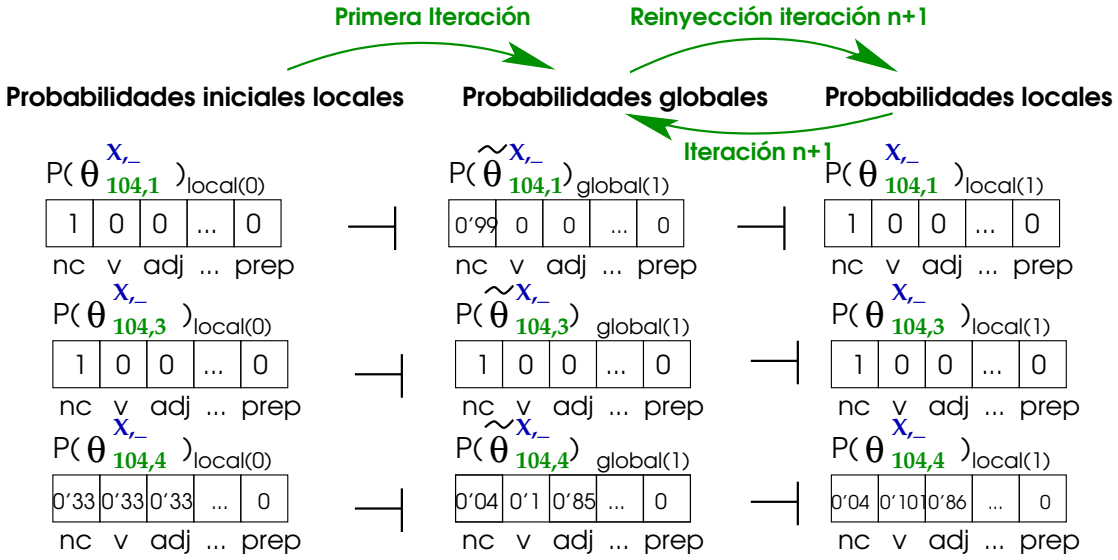


Figura 9.8: Cálculo de las probabilidades para la categorización de tokens

Un elemento de estas columnas es una lista de probabilidades de categorías léxicas *incluyendo todas las alternativas posibles para la correspondiente forma léxica. Más concretamente, la columna de la izquierda es la estimación de las probabilidades iniciales locales del token* $\Theta^{X,-}_{104,j}$*. La del centro se refiere al cálculo de su probabilidad global, y la columna de la derecha representa la reinyección de ella en la siguiente iteración. Como se puede observar, en el caso de* «Feuilles» (`«Hojas»`) *la probabilidad inicial es la misma que el resultado obtenido después de la primera iteración, debido a que sólo tiene una posible categoría léxica.*

■

9.2.2 | Categorización de las dependencias entre tokens

Se trata ahora de dar una medida objetiva de la viabilidad de las dependencias sintácticas generadas por el analizador sintáctico, entre los tokens previamente categorizados. Teniendo en cuenta que la caracterización topológica de la ambigüedad sintáctica significa la existencia de varios tokens gobernantes para un mismo gobernado, determinado éste buscaremos definir cual es su gobernante de entre los posibles propuestos por el analizador, con el fin de eliminar dicha ambigüedad. De nuevo consideraremos una estrategia iterativa, en este caso determinada por las ecuaciones de la Tabla 9.5, que describimos a continuación:

$$W(\delta^{\Theta^{a,\_}_{i,j},\tau,\Theta^{b,\_}_{i,k}}) = \frac{|S \overset{*}{\Rightarrow} \Theta^{a,\_}_{i,j} \overset{\tau}{\Rightarrow} \Theta^{b,\_}_{i,k}|}{\sum_{\delta^{\Theta^{Y,\_}_{i,X},T,\Theta^{Z,\_}_{i,k}}} |S \overset{*}{\Rightarrow} \Theta^{Y,\_}_{i,X} \overset{T}{\Rightarrow} \Theta^{Z,\_}_{i,k}|} \tag{9.7}$$

$$P(\delta^{\Theta^{a,\_}_{i,j},\tau,\Theta^{b,\_}_{i,k}})_{\text{local(0)}} = \frac{P(\Theta^{a,\_}_{i,j})_{\text{local}} \cdot\ P(\Theta^{b,\_}_{i,k})_{\text{local}} \cdot\ W(\delta^{\Theta^{a,\_}_{i,j},\tau,\Theta^{b,\_}_{i,k}})}{\sum_{\Theta^{Y,\_}_{i,X},\Theta^{Z,\_}_{i,k},\delta^{\Theta^{Y,\_}_{i,X},T,\Theta^{Z,\_}_{i,k}}} P(\Theta^{Y,\_}_{i,X})_{\text{local}} \cdot\ P(\Theta^{Z,\_}_{i,k})_{\text{local}} \cdot\ W(\delta^{\Theta^{Y,\_}_{i,X},T,\Theta^{Z,\_}_{i,k}})} \tag{9.8}$$

$$P(\delta^{\tilde{\Theta}^{a,\_}_{i,j},\tau,\tilde{\Theta}^{b,\_}_{i,k}})_{\text{global(n+1)}} = \frac{\sum_{\Theta_{l,m}=\Theta_{i,j},\Theta_{l,p}=\Theta_{i,k}} P(\delta^{\Theta^{a,\_}_{l,m},\tau,\Theta^{b,\_}_{l,p}})_{\text{local(n)}}}{\sum_{\delta^{\Theta^{Y,\_}_{l,X},T,\Theta^{Z,\_}_{l,p}},\Theta_{l,p}=\Theta_{i,k}} P(\delta^{\Theta^{Y,\_}_{l,X},T,\Theta^{Z,\_}_{l,p}})_{\text{local(n)}}} \tag{9.9}$$

$$P(\delta^{\Theta^{a,\_}_{i,j},\tau,\Theta^{b,\_}_{i,k}})_{\text{local(n+1)}} = \frac{P(\delta^{\tilde{\Theta}^{a,\_}_{i,j},\tau,\tilde{\Theta}^{b,\_}_{i,k}})_{\text{global(n+1)}}}{\sum_{\delta^{\tilde{\Theta}^{Y,\_}_{l,X},T,\tilde{\Theta}^{Z,\_}_{l,m}},\Theta_{l,m}=\Theta_{i,k}} P(\delta^{\tilde{\Theta}^{Y,\_}_{l,X},T,\tilde{\Theta}^{Z,\_}_{l,m}})_{\text{global(n+1)}}} \tag{9.10}$$

Tabla 9.5: Modelo para la categorización de las dependencias entre tokens

(9.7). Antes de iniciar el proceso iterativo, calcularemos para cada dependencia sintáctica un peso inicial en función de su etiqueta. Buscamos con ello dar protagonismo a aquellas dependencias compartidas por un mayor número de análisis, de entre las que comparten un mismo token gobernado. En el caso en que la dependencia gobernante/gobernado $\delta^{\theta^{a,\_}_{i,j},\tau,\theta^{b,\_}_{i,k}}$ se haya obtenido concatenando varias reglas gramaticales para generar el token $\Theta^{b,\_}_{i,k}$ a partir de uno $\Theta^{a,\_}_{i,j}$, pasando por otro $\Theta^{p,\_}_{i,l}$ en el análisis sintáctico, el número de derivaciones de dicha dependencia, es decir, $|S \overset{*}{\Rightarrow} \Theta^{a,\_}_{i,j} \overset{\tau}{\Rightarrow} \Theta^{b,\_}_{i,k}|$ será el mínimo entre el número de derivaciones de $\delta^{\theta^{a,\_}_{i,j},\tau',\theta^{p,\_}_{i,l}}$ y el de $\delta^{\theta^{p,\_}_{i,l},\tau'',\theta^{b,\_}_{i,k}}$, es decir

$$|S \overset{*}{\Rightarrow} \Theta^{a,\_}_{i,j} \overset{\tau}{\Rightarrow} \Theta^{b,\_}_{i,k}| = \min\{|S \overset{*}{\Rightarrow} \Theta^{a,\_}_{i,j} \overset{\tau'}{\Rightarrow} \Theta^{p,\_}_{i,l}|\ ,\ |S \overset{*}{\Rightarrow} \Theta^{p,\_}_{i,l} \overset{\tau''}{\Rightarrow} \Theta^{b,\_}_{i,k}|\}$$

(9.8). El proceso iterativo se inicia con el cálculo de la probabilidad local, a nivel de frase, asociable a una dependencia sintáctica. Dado que aquellas se caracterizan por sus tokens gobernante y gobernado, y por su etiqueta, haremos depender esta probabilidad de las locales de dichos tokens; y del peso asignado a la etiqueta asociada. Se calcula como una proporción de los valores citados para la dependencia sintáctica considerada, en relación al conjunto de las asociadas a la agrupación del token gobernado.

(9.9). Define la probabilidad global en el *corpus* de una dependencia plausible en la iteración $n + 1$ del proceso. Se calcula como una proporción de la probabilidad

local asociada a dependencias sintácticas coincidentes con la considerada (salvo en la frase que la localiza), en relación al conjunto de las locales asociadas a tokens gobernados también coincidentes con el considerado (salvo en la agrupación que lo localiza).

(9.10). Establece el valor de la probabilidad local de una dependencia en la iteración $n+1$ del proceso. Para ello repercutimos las probabilidades calculadas globalmente, distribuyéndolas proporcionalmente entre las globales de las dependencias sintácticas plausibles asociadas a tokens gobernados coincidentes con el considerado (salvo en la agrupación que lo localiza).

Como en el caso de la categorización léxica el proceso itera hasta la convergencia sobre un punto fijo o la aproximación a un umbral prefijado. Ilustremos estos cálculos mediante el Ejemplo 9.6.

Ejemplo 9.6 *Retomemos el Ejemplo 9.5. Supongamos que ahora queremos calcular las probabilidades de las dependencias, centrándonos sobre todo en aquéllas que apuntan sobre* $\Theta^{-,-}_{104,4}$. *El primer paso consistirá en estimar los pesos iniciales de las dependencias, tal y como se ilustran en las siguientes ecuaciones, de tal manera que la suma de todos los que apuntan sobre una misma agrupación sea* 1.

$$
\begin{array}{ll}
W(\delta^{\Theta^{nc,-}_{104,1},[2]\_\grave{a},\Theta^{nc,-}_{104,3}}) = \frac{2}{2} = 1 & W(\delta^{\Theta^{nc,-}_{104,1},[1],\Theta^{adj,-}_{104,4}}) = \frac{1}{5} = 0'2 \\
W(\delta^{\Theta^{nc,-}_{104,3},[1],\Theta^{adj,-}_{104,4}}) = \frac{1}{5} = 0'2 & W(\delta^{\Theta^{nc,-}_{104,3},[3],\Theta^{nc,-}_{104,4}}) = \frac{1}{5} = 0'2 \\
W(\delta^{\Theta^{nc,-}_{104,3},[4],\Theta^{v,-}_{104,4}}) = \frac{1}{5} = 0'2 & W(\delta^{\Theta^{nc,-}_{104,1},[4],\Theta^{v,-}_{104,4}}) = \frac{1}{5} = 0'2
\end{array}
\tag{9.11}
$$

Así, la probabilidad local inicial en el caso de la dependencia que une el token $\Theta^{nc,-}_{104,1}$ *con el token* $\Theta^{adj,-}_{104,4}$ *mediante la etiqueta* [1]*, es decir,* $\delta^{\Theta^{nc,-}_{104,1},[1],\Theta^{adj,-}_{104,4}}$*, se calcula en base a lo indicado en la Ecuación 9.6.*

$$
\begin{aligned}
&P(\delta^{\Theta^{nc,-}_{104,1},[1],\Theta^{adj,-}_{104,4}})_{\text{local(0)}} = \\
&= \frac{P(\Theta^{nc,-}_{104,1})_{\text{local}} \cdot\ P(\Theta^{adj,-}_{104,4})_{\text{local}} \cdot\ W(\delta^{\Theta^{nc,-}_{104,1},[1],\Theta^{adj,-}_{104,4}})}{\sum_{\Theta^{Y,-}_{104,X},\Theta^{Z,-}_{104,4},\delta^{\Theta^{Y,-}_{104,X},T,\Theta^{Z,-}_{104,4}}} P(\Theta^{Y,-}_{104,X})_{\text{local}} \cdot\ P(\Theta^{Z,-}_{104,4})_{\text{local}} \cdot\ W(\delta^{\Theta^{Y,-}_{104,X},T,\Theta^{Z,-}_{104,4}})}
\end{aligned}
\tag{9.12}
$$

donde $P(\Theta^{nc,-}_{104,1})_{\text{local}}$ *y* $P(\Theta^{adj,-}_{104,4})_{\text{local}}$ *representan las probabilidades de que la forma* «Feuilles» (`«Hojas»`) *sea un sustantivo en la agrupación* $\Theta^{-,-}_{104,1}$*, y* «denticulées» (`«dentadas»`) *sea un adjetivo en* $\Theta^{-,-}_{104,4}$*. Ambas se calcularon mediante la categorización de tokens en el Ejemplo 9.5, por lo que supongamos que sus resultados*

son los que mostramos a continuación

$$\begin{array}{lll} P(\Theta^{nc,\_}_{104,1})_{\text{local}} = 1; & P(\Theta^{nc,\_}_{104,3})_{\text{local}} = 1; & P(\Theta^{adj,\_}_{104,4})_{\text{local}} = 0\text{'}57; \\ P(\Theta^{v,\_}_{104,4})_{\text{local}} = 0\text{'}31 & P(\Theta^{nc,\_}_{104,4})_{\text{local}} = 0\text{'}12; & \end{array} \tag{9.13}$$

Por lo tanto, el valor de $P(\delta^{\Theta^{nc,\_}_{104,1},[1],\Theta^{adj,\_}_{104,4}})_{\text{local(0)}}$ *es*

$$P(\delta^{\Theta^{nc,\_}_{104,1},[1],\Theta^{adj,\_}_{104,4}})_{\text{local(0)}} = \frac{1 \cdot 0\text{'}57 \cdot 0\text{'}2}{1 \cdot 0\text{'}57 \cdot 0\text{'}2 + \cdots + 1 \cdot 0\text{'}31 \cdot 0\text{'}2} = \frac{0\text{'}114}{0\text{'}376} = 0\text{'}3031$$

Una vez calculada la probabilidad local inicial para cada una de las dependencias, tenemos que estimar a nivel global las dependencias plausibles para el corpus *en la primera iteración. Así, tenemos que* $P(\delta^{\tilde{\Theta}^{nc,\_}_{104,1},[1],\tilde{\Theta}^{adj,\_}_{104,4}})_{\text{global(1)}}$ *se expresará tal y como se muestra a continuación.*

$$P(\delta^{\tilde{\Theta}^{nc,\_}_{104,1},[1],\tilde{\Theta}^{adj,\_}_{104,4}})_{\text{global(1)}} = \frac{\sum_{\Theta_{l,m}=\text{Feuilles},\Theta_{l,p}=\text{denticulées}} P(\delta^{\Theta^{nc,\_}_{l,m},[1],\Theta^{adj,\_}_{l,p}})_{\text{local(0)}}}{\sum_{\delta^{\Theta^{Y,\_}_{l,\bar{X}},T,\Theta^{Z,\_}_{l,p}},\Theta_{l,p}=\text{denticulées}} P(\delta^{\Theta^{Y,\_}_{l,\bar{X}},T,\Theta^{Z,\_}_{l,p}})_{\text{local(0)}}} \tag{9.14}$$

Finalmente, calcularemos la probabilidad local para la siguiente iteración de la ocurrencia de $\delta^{\Theta^{nc,\_}_{104,1},[1],\Theta^{adj,\_}_{104,4}}$*, realizando una normalización con respecto a todas las posibles dependencias que tienen por destino la agrupación gobernada* $\Theta^{\_,\_}_{104,4}$*. Una vez deducido, se utilizará para calcular el valor global de las sucesivas iteraciones.*

$$P(\delta^{\Theta^{nc,\_}_{104,1},[1],\Theta^{adj,\_}_{104,4}})_{\text{local(1)}} = \frac{P(\delta^{\tilde{\Theta}^{nc,\_}_{104,1},[1],\tilde{\Theta}^{adj,\_}_{104,4}})_{\text{global(1)}}}{\sum_{\delta^{\tilde{\Theta}^{Y,\_}_{l,\bar{X}},T,\tilde{\Theta}^{Z,\_}_{l,m}},\Theta_{l,m}=\text{denticulées}} P(\delta^{\tilde{\Theta}^{Y,\_}_{l,\bar{X}},T,\tilde{\Theta}^{Z,\_}_{l,m}})_{\text{global(1)}}} \tag{9.15}$$

En este sentido, la Fig. 9.9 ilustra el cálculo realizado para cada una de las dependencias que tienen el token gobernado en $\Theta^{\_,\_}_{104,j}$*, donde* j *puede tener el valor* 3*, asociándolo a la forma* «nervures» (`«nervaduras»`)*, ó* 4*, cuya forma es* «denticulées» (`«dentadas»`)*. Para ambos casos, se utilizan tres columnas que introducen cada uno de los pasos que acabamos de describir. A su vez, cada una está dividida por filas que indican el número de tokens gobernados de la frase.*

Así, la primera columna identifica los valores locales iniciales de las dependencias que apuntan sobre un token gobernado, y la primera fila lo hace con aquéllas que lo hacen sobre la forma $\Theta_{104,3}$*, es decir,* $\delta^{\Theta^{Y,\_}_{104,X},T,\Theta^{Z,\_}_{104,3}}$*. En cambio, la segunda fila representa aquéllas que tienen por token gobernado a la forma* $\Theta_{104,4}$*, mediante* $\delta^{\Theta^{Y,\_}_{104,X},T,\Theta^{Z,\_}_{104,4}}$*. Cada una de estas filas a su vez posee una lista de probabilidades de dependencias, incluyendo todas las posibles alternativas para la correspondiente agrupación «*`X`*» y categoría léxica*

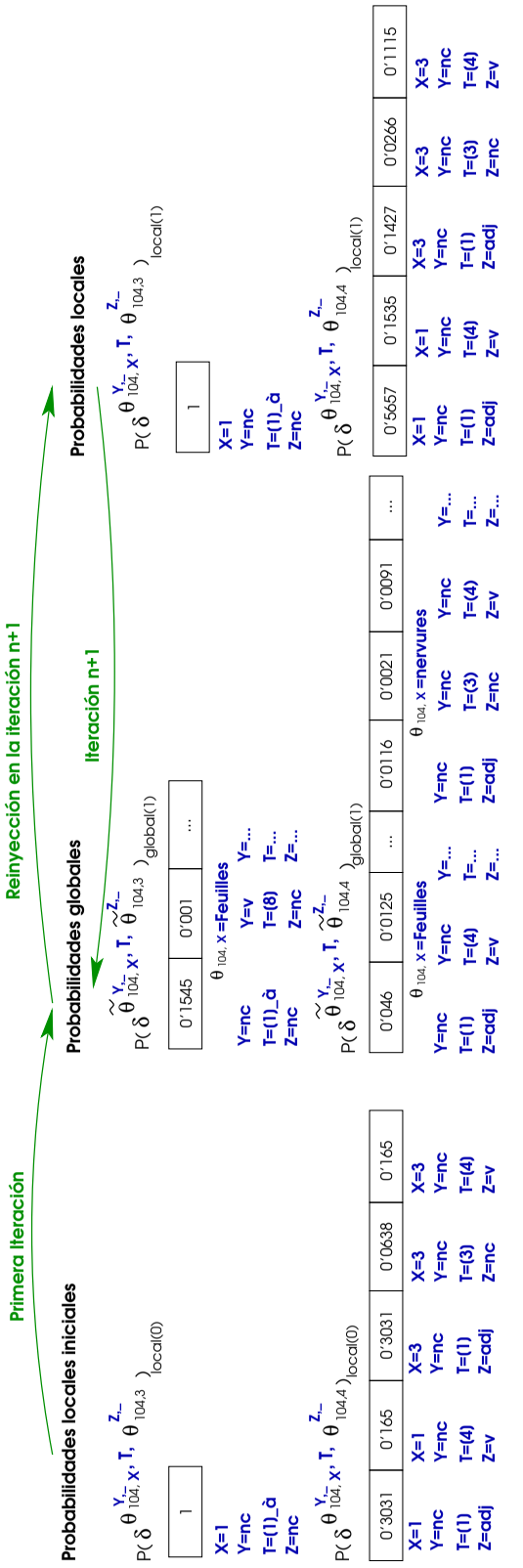


Figura 9.9: Cálculo de las probabilidades de las dependencias entre tokens

*«*Y*» del token gobernante, etiqueta de la dependencia «*T*», y categoría léxica «*Z*» del gobernado.*

La siguiente columna se refiere a los cálculos de dichas dependencias a nivel global, considerando que se estiman todas las posibles formas que apuntan mediante una dependencia sobre otra dada en el corpus $\mathscr{C}$*. Finalmente, la columna de la derecha lo hace después de reinyectar los valores calculados previamente en la siguiente iteración.*

■

9.2.3 | Categorización de las dependencias entre términos

El objetivo en este nivel es determinar las clases semánticas correctas de los tokens que participan en una misma dependencia sintáctica, con el fin de identificar las que unen términos de dos agrupaciones diferentes. Más exactamente, dado un término gobernado, buscamos definir cual es su gobernante a través de las dependencias sintácticas previamente categorizadas.

En este sentido, existen trabajos [118] que buscan agrupar a las palabras en base a un mismo eje semántico, estableciendo relaciones de coocurrencia entre sus contextos locales. De este modo, consiguen generar clases semánticas constituidas a partir de datos léxico-sintácticos, donde la subjetividad del lingüista no interviene directamente. Sin embargo, los resultados obtenidos no siempre son acordes y perfectamente interpretables usando una base puramente endógena. Pueden parecer semánticamente correctas a primera vista [30, 204], pero, a pesar de ello, resulta necesario ajustar sus límites y comprobar su coherencia. De hecho, a menudo es inevitable la utilización de alguna fuente externa de información.

Definición 9.4 *Sea* $s_i,\ 1 \leq i \leq n$ *la* i*-ésima frase de un* corpus $\mathscr{C}$*, y* $\mathscr{T}$ *(resp.* $\mathscr{F}$*) el conjunto de clases semánticas (resp. de formas semánticas) asociadas a* $\mathscr{C}$ *(resp. a* $\mathscr{T}$*) por medio de alguna técnica fiable. Se denota por* $\mathscr{F}(b)$ *al subconjunto de formas asociadas a* $b \in \mathscr{T}$*, y se dice que* $\Theta_{i,j}^{a,b},\ 1 \leq j \leq |s_i|$ *es un* término estable *si y sólo si* $b \in \mathscr{T}$ *y* $\Theta_{i,j} \in \mathscr{F}(b)$.

■

Intuitivamente, un término es estable cuando tenemos información fidedigna acerca de la correspondencia entre su categoría semántica y su forma. El origen de ésta puede ser el propio usuario o algún método considerado plenamente fiable. Nuestra propuesta considera ambos mecanismos [96]. Por un lado, el usuario define el conjunto de clases semánticas. En nuestro *corpus* de ejemplo botánico $\mathscr{B}$ éstas se organizan en entidades ($\mathscr{E}$) y propiedades ($\mathscr{P}$), de tal manera que dichas propiedades proporcionen información acerca de los atributos aplicables a las entidades; y complementados por un conjunto asociado de formas iniciales tales como las que se muestran en la Tabla 9.6. Estos valores

se toman a partir de los tokens, cuyo lema en el análisis sintáctico es conocido, y su elección viene determinada por su alta frecuencia de aparición.

Entidades	**Lemas** (en francés)
organe	fleur, staminode, tige, feuille, hypanthe, périanthe, rameau, ...
fruit	fruit, samare, drupe, capsule, akène
Propiedades	**Lemas** (en francés)
couleur	verdâtre, violacé, noirâtre, violet, jaunâtre, orange, roux, rose
forme	obconique, oblancéolé, oblong, bifolié, crateriforme, punctiforme, ...
taille	moyen, petit, double, épais, inégal, entier, longue
texture	hispide, bifide, globuleux, coriace, velutineux, gélatineux, barbu
position	antérieur, dessus, voisin, seul, latéral, transversal

Tabla 9.6: Conjunto $\mathscr{T}$ de clases semánticas (tipos) para el ejemplo de funcionamiento

Ejemplo 9.7 *Volviendo al Ejemplo 9.6, cada token se expresa mediante un conjunto de términos, representados en la Fig. 9.5 mediante triángulos. Concretamente, todos ellos contienen una abreviatura de la clase semántica en cuestión. Así, el primer término presente en* $\Theta_{104,1}^{nc,-}$ *se representa por* $\Theta_{104,1}^{nc,org}$*, donde* «org» *hace referencia a la clase semántica* «Organe» (`«Órgano»`)*. Del mismo modo, el último término presente en ese token se representa por* $\Theta_{104,1}^{nc,for}$*, donde* «for» *hace referencia a la clase semántica* «Forme» (`«Forma»`)*. Para tratar de facilitar la comprensión de la figura, se evitó representar todos los términos, de ahí la utilización de los puntos suspensivos.*

■

Marcador(francés)	**Posición**	**Clase**	**Marcador**(francés)	**Posición**	**Clase**
teinté	[2]	couleur	épaisseur	[1]	tamaño
texture	[2]	texture	atteindre	[1]	órgano/fruto
taille	[1]	Organe/Fruit	taille	[2]	Taille
teinte	[1]	Organe/Fruit	teinte	[2]	Couleur
couleur	[1]	Organe/Fruit	couleur	[2]	Couleur
texture	[1]	Organe/Fruit	texture	[2]	Texture
forme	[1]	Organe/Fruit	forme	[2]	Forme
position	[1]	Organe/Fruit	position	[2]	Position
altitude	[1]	Organe/Fruit	environ	[2]	Taille
tache	[1]	Organe/Fruit	tache	[2]	Couleur
longueur	[1]	Taille	formé	[2]	Organe/Fruit
composé	[1,2]	Organe/Fruit	dépassant	[2]	Taille
diamètre	[1]	Taille	contour	[2]	Forme/Texture
contour	[2]	Forme/Texture	bord	[2]	Forme

Tabla 9.7: Parte del fichero de colocaciones

Por otro lado, el sistema saca ventaja de las *colocaciones*, secuencias de palabras que coocurren con más frecuencia de lo esperado y en las cuales conservan su significado original, al contrario de lo que ocurre con las *locuciones*. La idea es filtrar los análisis

con el fin de localizar aquéllas que permitan asociar una forma a una clase semántica. Para la ocasión, las representamos como una tripleta de la forma *marcador/posición/clase semántica*. El marcador sirve para identificar la colocación, para la que la forma indicada por la posición pueda ser asociada a la clase semántica, tal y como se muestra en la Tabla 9.7, en el caso de nuestro *corpus* de ejemplo $\mathscr{B}$.

Presumiblemente estas colocaciones proporcionarán una información más fiable tanto en relación a las clases como a las dependencias, concentrando el vocabulario a su alrededor. De este modo, el resultado sirve para adquirir conceptos simples, permitiendo proporcionar más valores de las entidades y propiedades, y propagando alguno de ellos.

Ejemplo 9.8 *Supongamos que tenemos la frase* «teintées de rose» (`«teñidas de rosa»`). *La presencia del marcador* «teinté» (`«teñida»`) *pone en evidencia que* «rose» (`«rosa»`) *es una instancia de la clase semántica* «couleur» (`«color»`), *debido a que se localiza en la posición [2] de la dependencia, tal y como podemos observar en la Fig. 9.10.*

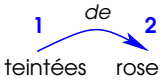


Figura 9.10: Un ejemplo de estructura con colocaciones

■

El proceso iterativo se corresponde con las ecuaciones de la Tabla 9.8 que ahora describimos:

$$W(\Theta_{i,\bar{j}}^{a,-}) > \frac{\left|\{\Theta_{i,j}^{a,X}\}_{X\in\mathscr{T},\Theta_{i,j}\in\mathscr{F}(X)}\right|}{\left|\{\Theta_{i,j}^{a,X}\}_{X\in\mathscr{T}}\right|} \subseteq (0,1] \tag{9.16}$$

$$W(\Theta_{i,j}^{a,b}) = \begin{cases} \frac{W(\Theta_{i,\bar{j}}^{a,-})}{\left|\{\Theta_{i,j}^{a,X}\}_{X\in\mathscr{T},\Theta_{i,j}\in\mathscr{F}(X)}\right|} & \text{si } \Theta_{i,j} \in \mathscr{F}(b) \\ \frac{1-W(\Theta_{i,\bar{j}}^{a,-})}{\left|\{\Theta_{i,j}^{a,X}\}_{X\in\mathscr{T},\Theta_{i,j}\notin\mathscr{F}(X)}\right|} & \text{en otro caso} \end{cases} \tag{9.17}$$

$$P(\delta^{\Theta_{i,j}^{a,b},\tau,\Theta_{i,k}^{c,d}})_{\text{local(0)}} = \frac{P(\delta^{\Theta_{i,\bar{j}}^{a,-},\tau,\Theta_{i,\bar{k}}^{c,-}})_{\text{local}} \cdot W(\Theta_{i,j}^{a,b}) \cdot W(\Theta_{i,k}^{c,d})}{\sum_{\Theta_{i,X}^{Y,Z},\Theta_{i,k}^{V,W},\delta^{\Theta_{i,\bar{X}}^{Y,-},T,\Theta_{i,\bar{k}}^{V,-}}} P(\delta^{\Theta_{i,\bar{X}}^{Y,-},T,\Theta_{i,\bar{k}}^{V,-}})_{\text{local}} \cdot W(\Theta_{i,X}^{Y,Z}) \cdot W(\Theta_{i,k}^{V,W})} \tag{9.18}$$

$$P(\delta^{\tilde{\Theta}^{a,b}_{i,j},\tau,\tilde{\Theta}^{c,d}_{i,k}})_{\text{global(n+1)}} = \frac{\sum_{\Theta_{l,m}=\Theta_{i,j},\Theta_{l,p}=\Theta_{i,k}} P(\delta^{\Theta^{a,b}_{l,m},\tau,\Theta^{c,d}_{l,p}})_{\text{local(n)}}}{\sum_{\delta^{\Theta^{Y,Z}_{l,X},T,\Theta^{V,W}_{l,p}},\Theta_{l,p}=\Theta_{i,k}} P(\delta^{\Theta^{Y,Z}_{l,X},T,\Theta^{V,W}_{l,p}})_{\text{local(n)}}} \tag{9.19}$$

$$P(\delta^{\Theta^{a,b}_{i,j},\tau,\Theta^{c,d}_{i,k}})_{\text{local(n+1)}} = \frac{P(\delta^{\tilde{\Theta}^{a,b}_{i,j},\tau,\tilde{\Theta}^{c,d}_{i,k}})_{\text{global(n+1)}}}{\sum_{\delta^{\tilde{\Theta}^{Y,Z}_{l,X},T,\tilde{\Theta}^{V,W}_{l,m}},\Theta_{l,m}=\Theta_{i,k}} P(\delta^{\tilde{\Theta}^{Y,Z}_{l,X},T,\tilde{\Theta}^{V,W}_{l,m}})_{\text{global(n+1)}}} \tag{9.20}$$

Tabla 9.8: Modelo para la categorización de las dependencias entre términos

(9.16). Antes de iniciar el proceso, asociaremos a cada token un peso que verifique la condición expuesta, y cuyo valor justificamos a continuación.

(9.17). Ahora vamos a distribuir equitativamente el peso calculado a partir de la Ecuación 9.16 entre los términos estables. Esto asegura que el peso que asociamos aquí a un término no estable en dicho token es inferior al asociado a los otros. Tratamos así de dar inicialmente preferencia a los términos estables.

(9.18). El proceso iterativo se inicia con el cálculo de la probabilidad local, a nivel de frase, asociable a una dependencia semántica. Dado que ésta queda perfectamente caracterizada por sus términos gobernante y gobernado junto con la dependencia sintáctica entre los tokens asociados a éstos, haremos depender este valor de los pesos asociados a dichos términos, así como de la probabilidad local correspondiente a la dependencia sintáctica. Se calcula como una proporción de los valores citados para la dependencia semántica considerada, en relación al conjunto de las asociadas a la agrupación del término gobernado.

(9.19). Define la probabilidad global en el *corpus* de una dependencia semántica plausible en la iteración $n+1$ del proceso. Se calcula como una proporción de la probabilidad local asociada a dependencias semánticas coincidentes con la considerada (salvo en la frase que la localiza), en relación al conjunto de las locales asociadas a términos gobernados también coincidentes con el considerado (salvo en la agrupación que lo localiza).

(9.20). Establece el valor de la probabilidad local asociable a una dependencia semántica en la iteración $n+1$ del proceso. Para ello repercutimos las probabilidades calculadas globalmente, distribuyéndolas proporcionalmente entre las globales de las dependencias semánticas plausibles asociadas a términos gobernados coincidentes con el considerado (salvo en la agrupación que lo localiza).

Como en el caso de la categorización de dependencias sintácticas, el proceso itera hasta la convergencia sobre un punto fijo o la aproximación a un umbral prefijado. En este sentido, la hipótesis de Harris, según la cual la similitud semántica puede detectarse a través del análisis del contexto lingüístico, se puede aplicar gracias a la utilización de las colocaciones. De esta manera, y usando los términos estables durante el proceso iterativo, conseguimos determinar las categorías semánticas asignables, mediante la aplicación de la desambigüación realizada a nivel de dependencias entre tokens. A la estructura resultante lo denominamos la *semántica del corpus* $\mathscr{C}$ con el que trabajamos.

Definición 9.5 *Sean* $\{s_i\}_{1\leq i\leq n}$ *una secuencia de frases de un* corpus $\mathscr{C}$*, y* $\mathscr{T}$ *(resp.* $\mathscr{F}$*) el conjunto de clases semánticas (resp. de formas) asociadas a* $\mathscr{C}$ *(resp. a* $\mathscr{T}$*) por medio de alguna técnica fiable. Se define la* semántica del corpus $\mathscr{C}$ *como*

$$\mathscr{S}_{\mathscr{C}} := \{\delta^{\Theta_{i,j}^{a,b},\tau,\Theta_{i,k}^{c,d}},\ P(\delta^{\Theta_{i,j}^{a,b},\tau,\Theta_{i,k}^{c,d}})_{\text{local}} = \text{máx}\{P(\delta^{\Theta_{i,j}^{X,Y},Z,\Theta_{i,k}^{V,W}})_{\text{local}}\}\}$$

donde máx *es la función maximal en* $\mathbb{N}$*, y* $\delta^{\Theta_{i,j}^{X,Y},Z,\Theta_{i,k}^{V,W}}$ *son las dependencias calculadas como resultado del proceso de adquisición de conocimiento previamente descrito.*

El concepto puede restringirse naturalmente para referirse a la semántica del documento $\mathscr{D}$ en $\mathscr{C}$ *por*

$$\mathscr{S}_{\mathscr{C}}^{\mathscr{D}} := \{\delta^{\Theta_{i,j}^{a,b},\tau,\Theta_{i,k}^{c,d}} \in \mathscr{S}_{\mathscr{C}},\ s_i \in \mathscr{D}\}$$

■

Intuitivamente, definimos la semántica del *corpus* como el conjunto de las dependencias más probables entre sus términos. Esto compila todas las relaciones sintácticas y semánticas consideradas como viables, entre las categorías léxicas en el texto estudiado. La semántica del *corpus* será el punto de partida para la generación de grafos conceptuales que nos sirven como representación del conocimiento formal para propósitos de RI.

Ejemplo 9.9 *Volviendo con el Ejemplo 9.6, vamos ahora a representar las dependencias de la Fig. 9.9, considerando esta vez que se realizan entre términos, representados por* $\delta^{\Theta_{i,j}^{a,b},\tau,\Theta_{i,k}^{c,d}}$*. La idea es que, dada una dependencia entre dos tokens, vamos a aplicar una redistribución de ésta entre los diferentes términos posibles, tal y como se ilustra en la Fig. 9.11.*

Por lo tanto, el primer paso debe consistir en determinar el peso $W(\Theta_{104,j}^{a,\_})$ *para cada token de la frase. En este caso, consideraremos que cada uno dispone de un conjunto de términos, cuyas clases semánticas son las que se encuentran disponibles en el conjunto* $\mathscr{T}$*, ilustrado en la Tabla 9.6. Así, cada token contiene un total de 7 términos.*

- $W(\Theta_{104,1}^{nc,\_}) > \frac{1}{7} = 0'1429$*. Existe una forma en la Tabla 9.6 que indica que dicho token va a contener un término estable.*

- $W(\Theta^{nc,-}_{104,3}) > \frac{0}{7} = 0$. *Como no existe ninguna forma que indique que dicho token pueda contener un término estable, su valor es 0.*
- $W(\Theta^{adj,-}_{104,4}) > \frac{0}{7} = 0$. *Idem al caso anterior.*
- $W(\Theta^{nc,-}_{104,4}) > \frac{0}{7} = 0$. *Idem al caso anterior.*
- $W(\Theta^{v,-}_{104,4}) > \frac{0}{7} = 0$. *Idem al caso anterior.*

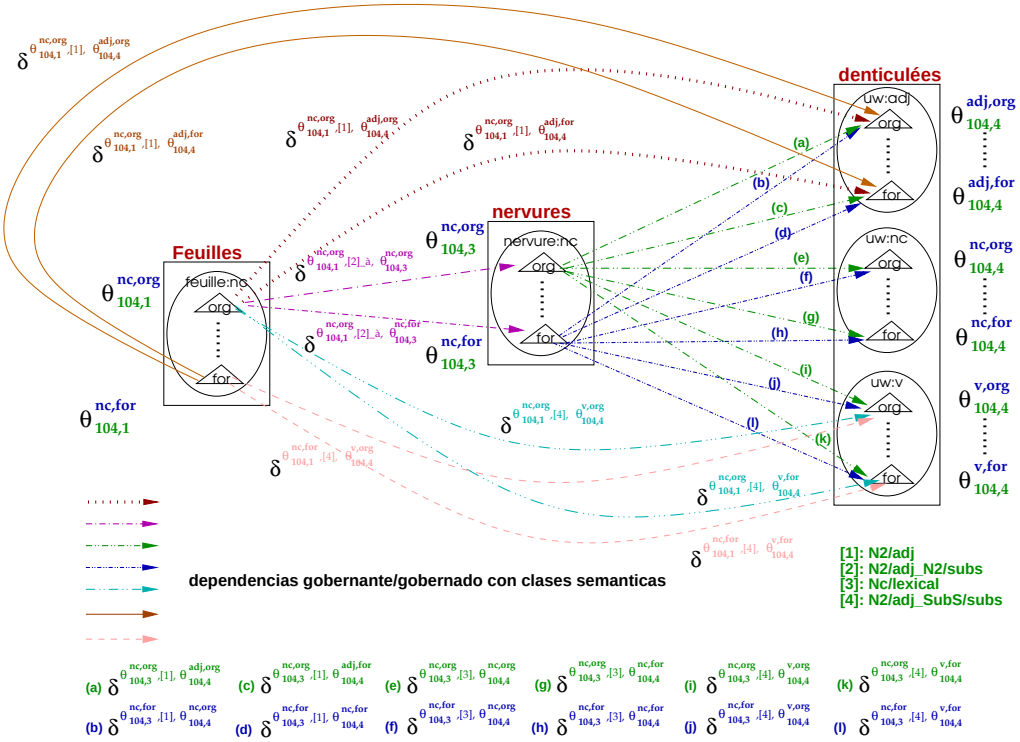


Figura 9.11: Notación de las ocurrencias de las dependencias entre términos

Una vez calculados, partiremos de la idea de que todos estos pesos poseen un valor igual a 0'7 *y lo distribuiremos equitativamente por cada uno de los términos. Empezaremos por los que componen* $\Theta^{nc,-}_{104,1}$.

- $W(\Theta^{nc,org}_{104,1}) = \frac{0'7}{1} = 0'7$. *Se divide entre un único elemento ya que este peso representa el término estable.*
- $W(\Theta^{nc,fru}_{104,1}) = \frac{1 - 0'7}{6} = 0'05$. *En este caso se divide entre seis, es decir, el número de términos restantes en el token.*
- $W(\Theta^{nc,cou}_{104,1}) = \frac{1 - 0'7}{6} = 0'05$. *Idem al caso anterior.*
- $W(\Theta^{nc,tai}_{104,1}) = \frac{1 - 0'7}{6} = 0'05$. *Idem al caso anterior.*
- $W(\Theta^{nc,tex}_{104,1}) = \frac{1 - 0'7}{6} = 0'05$. *Idem al caso anterior.*

- $W(\Theta^{nc,pos}_{104,1}) = \frac{1 - 0'7}{6} = 0'05$. *Idem al caso anterior.*
- $W(\Theta^{nc,for}_{104,1}) = \frac{1 - 0'7}{6} = 0'05$. *Idem al caso anterior.*

Del mismo modo, realizaremos el mismo proceso para los términos de los restantes tokens. En este sentido, sea cual sea la clase semántica asignada a $\Theta^{nc,\_}_{104,3}$*, sus pesos serán idénticos, es decir,*

$$W(\Theta^{nc,X}_{104,3}) = \frac{1 - 0'7}{7} = 0'043 \text{ siendo } X \in \mathscr{T},$$

De la misma manera, independientemente de cual sea la clase semántica de la agrupación $\Theta^{\_,\_}_{104,4}$*, sus pesos serán iguales.*

$$W(\Theta^{a,X}_{104,4}) = \frac{1 - 0'7}{7} = 0'043 \text{ siendo } X \in \mathscr{T},$$

Así, en la Fig. 9.12 vemos como quedarían para el caso de los términos presentes en las agrupaciones $\Theta^{\_,\_}_{104,1}$ *y* $\Theta^{\_,\_}_{104,3}$*.*

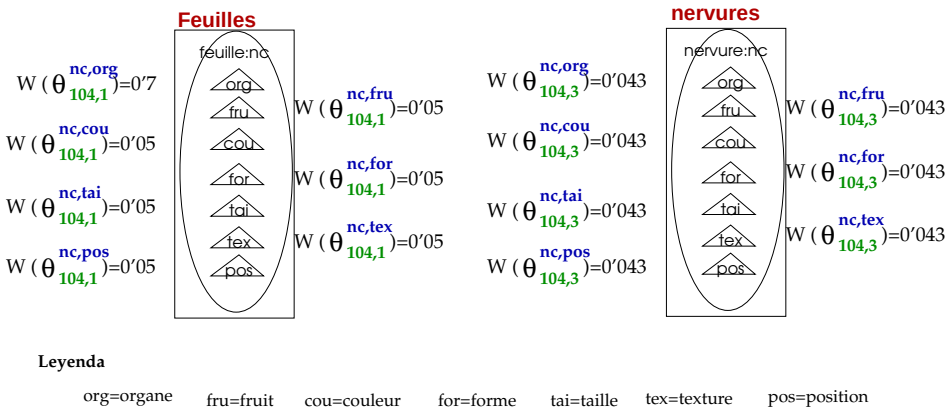


Figura 9.12: Lista de pesos semánticos

Ahora, calcularemos la probabilidad local inicial en el caso de la dependencia que une el término $\Theta^{nc,org}_{104,1}$ *con* $\Theta^{adj,org}_{104,4}$ *mediante la etiqueta* [1]*, es decir,* $\delta^{\Theta^{nc,org}_{104,1},[1],\Theta^{adj,org}_{104,4}}$*, tal como la Ecuación 9.21.*

$$P(\delta^{\Theta^{nc,org}_{104,1},[1],\Theta^{adj,org}_{104,4}})_{\text{local(0)}} = \\ = \frac{P(\delta^{\Theta^{nc,\_}_{104,1},[1],\Theta^{adj,\_}_{104,4}})_{\text{local}} \cdot W(\Theta^{nc,org}_{104,1}) \cdot W(\Theta^{adj,org}_{104,4})}{\sum_{\Theta^{Y,Z}_{104,X},\Theta^{V,W}_{104,4},\delta^{\Theta^{Y,\_}_{104,X},T,\Theta^{V,\_}_{104,4}}} P(\delta^{\Theta^{Y,\_}_{104,X},T,\Theta^{V,\_}_{104,4}})_{\text{local}} \cdot W(\Theta^{Y,Z}_{104,X}) \cdot W(\Theta^{V,W}_{104,4})} \tag{9.21}$$

donde $P(\delta^{\Theta^{nc,\_}_{104,1},[1],\Theta^{adj,\_}_{104,4}})_{\text{local}}$ *representa a la probabilidad local de la dependencia entre esos tokens, que ya se calculó mediante la categorización de dependencias entre tokens en el Ejemplo 9.6. Supongamos entonces que esos valores son los siguientes.*

$$
\begin{aligned}
&P(\delta^{\Theta^{nc,-}_{104,1},[1],\Theta^{adj,-}_{104,4}})_{\text{local}} = 0\text{'}5657; \quad P(\delta^{\Theta^{nc,-}_{104,1},[4],\Theta^{v,-}_{104,4}})_{\text{local}} = 0\text{'}1535;\\
&P(\delta^{\Theta^{nc,-}_{104,1},[4],\Theta^{v,-}_{104,4}})_{\text{local}} = 0\text{'}1535; \quad P(\delta^{\Theta^{nc,-}_{104,3},[1],\Theta^{adj,-}_{104,4}})_{\text{local}} = 0\text{'}1427;\\
&P(\delta^{\Theta^{nc,-}_{104,3},[3],\Theta^{nc,-}_{104,4}})_{\text{local}} = 0\text{'}0266; \quad P(\delta^{\Theta^{nc,-}_{104,3},[4],\Theta^{v,-}_{104,4}})_{\text{local}} = 0\text{'}1115
\end{aligned}
\tag{9.22}
$$

Por lo tanto, el valor de $P(\delta^{\Theta^{nc,org}_{104,1},[1],\Theta^{adj,org}_{104,4}})_{\text{local(0)}}$ *es*

$$
P(\delta^{\Theta^{nc,org}_{104,1},[1],\Theta^{adj,org}_{104,4}})_{\text{local(0)}} = \frac{0.5657 \cdot 0.7 \cdot 0.043}{0.5657 \cdot 0.7 \cdot 0.043 + \cdots + 0.1115 \cdot 0.043 \cdot 0.043} = \frac{0.017}{0.24191} = 0.00702
$$

Una vez calculada la probabilidad inicial para cada dependencia entre términos, estimaremos la probabilidad global de la dependencia plausible en el corpus *para la primera iteración. Así, tenemos que* $P(\delta^{\tilde{\Theta}^{nc,org}_{104,1},[1],\tilde{\Theta}^{adj,org}_{104,4}})_{\text{global(1)}}$ *se expresará tal y como se muestra a continuación.*

$$
P(\delta^{\tilde{\Theta}^{nc,org}_{104,1},[1],\tilde{\Theta}^{adj,org}_{104,4}})_{\text{global(1)}} = \frac{\sum_{\Theta_{l,m}=\text{Feuilles},\Theta_{l,p}=\text{denticulées}} P(\delta^{\Theta^{nc,org}_{l,m},[1],\Theta^{adj,org}_{l,p}})_{\text{local(0)}}}{\sum_{\delta^{\Theta^{Y,Z}_{l,X},T,\Theta^{V,W}_{l,p}},\Theta_{l,p}=\text{denticulées}} P(\delta^{\Theta^{Y,Z}_{l,X},T,\Theta^{V,W}_{l,p}})_{\text{local(0)}}} \tag{9.23}
$$

Finalmente, calcularemos la probabilidad local para la siguiente iteración de la ocurrencia de $\delta^{\Theta^{nc,org}_{104,1},[1],\Theta^{adj,org}_{104,4}}$*, realizando una normalización con respecto a todas las posibles dependencias que tienen por destino la agrupación gobernada* $\Theta^{-,-}_{104,4}$*. Una vez deducido, se utilizará para calcular el valor global de las sucesivas iteraciones.*

$$
P(\delta^{\Theta^{nc,org}_{104,1},[1],\Theta^{adj,org}_{104,4}})_{\text{local(1)}} = \frac{P(\delta^{\tilde{\Theta}^{nc,org}_{104,1},[1],\tilde{\Theta}^{adj,org}_{104,4}})_{\text{global(1)}}}{\sum_{\delta^{\tilde{\Theta}^{Y,Z}_{l,X},T,\tilde{\Theta}^{V,W}_{l,m}},\Theta_{l,m}=\text{denticulées}} P(\delta^{\tilde{\Theta}^{Y,Z}_{l,X},T,\tilde{\Theta}^{V,W}_{l,m}})_{\text{global(1)}}} \tag{9.24}
$$

En este sentido, la Fig. 9.13 ilustra el cálculo realizado para cada una de las dependencias entre términos. Para ello, se utilizan tres columnas que introducen cada uno de los pasos que acabamos de describir. A su vez, cada una está dividida por filas. Así, la primera columna identifica los valores locales iniciales de las dependencias que apuntan sobre un término gobernado, y la primera fila lo hace de aquéllas que apuntan sobre la forma $\Theta_{104,3}$*, es decir,* $\delta^{\Theta^{Y,Z}_{104,X},T,\Theta^{V,W}_{104,3}}$*. En cambio, la segunda fila representa aquéllas que tienen por término gobernado a la forma* $\Theta_{104,4}$*, mediante* $\delta^{\Theta^{Y,Z}_{104,X},T,\Theta^{V,W}_{104,4}}$*. Cada una de estas filas a su vez posee una lista de probabilidades de dependencias, incluyendo todas las posibles alternativas para la correspondiente agrupación «*X*», categoría léxica «*Y*» y clase semántica «*Z*» del término gobernante; la etiqueta de la dependencia «*T*»; la categoría léxica «*V*» y la clase semántica «*W*» del gobernado. La siguiente columna se refiere a los cálculos de dichas dependencias a nivel global, y la columna de la derecha lo hace después de reinyectar los valores calculados previamente en la siguiente iteración.*

■

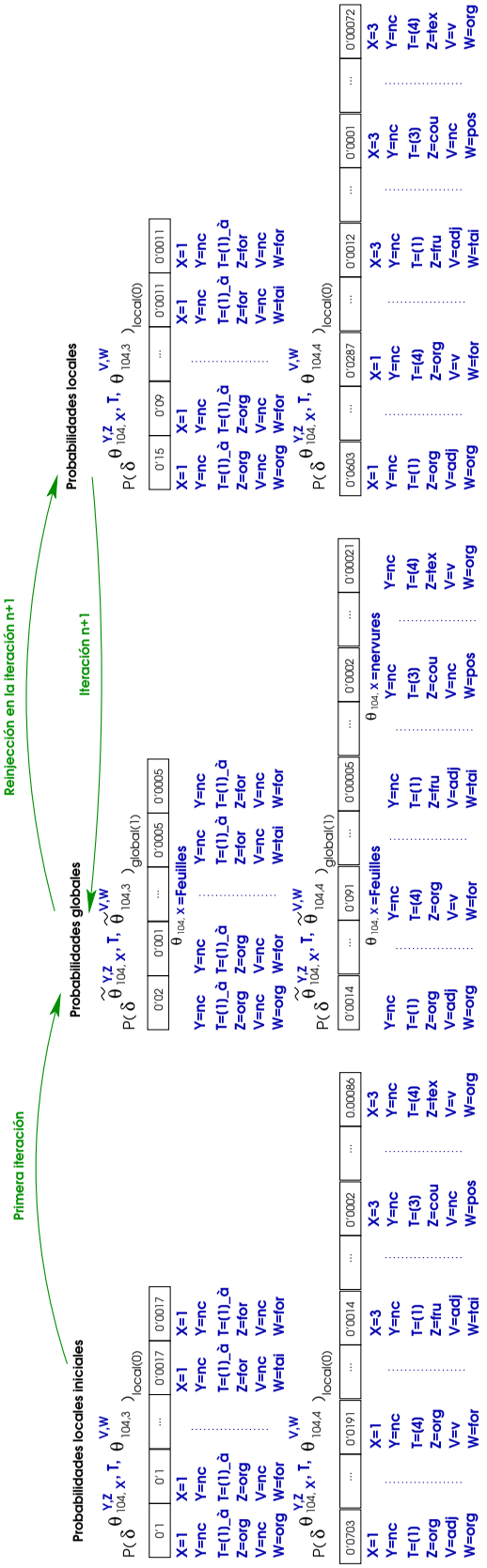


Figura 9.13: Cálculo de las probabilidades de las dependencias entre términos

9.3 | Representación del conocimiento: generación de grafos conceptuales

Una vez atribuidos los conceptos a los componentes de las dependencias gobernante/gobernado, estamos listos para estructurar los GCB's que vamos a utilizar en nuestras pruebas experimentales. Aunque la propuesta es independiente del ámbito de conocimiento considerado, es necesario centrar nuestro trabajo en la descripción botánica, tomando como referencia el *corpus* de ejemplo $\mathscr{B}$, con el fin de modelizar adecuadamente el soporte sobre el que se definirán los grafos.

En este sentido, retomamos el conjunto de clases semánticas (tipos) $\mathscr{T}$ mostrado en la Tabla 9.6 para el *corpus* $\mathscr{B}$, con el fin de introducir en él un orden parcial en la forma:

$$\forall\, t \in \mathscr{E} = \{\textit{fruit, organe}\},\ t \leq \varepsilon \leq \top$$

$$\forall\, t \in \mathscr{P} = \{\textit{couleur, forme, taille, texture, position}\},\ t \leq \rho \leq \top$$

donde ε (resp. ρ) es el mayor elemento que representa a las entidades $\mathscr{E}$ (resp. propiedades $\mathscr{P}$). De esta manera, introducimos nuestro soporte de ejemplo $\mathcal{S} = (\mathcal{T}_{\mathcal{C}_{\mathscr{B}}}, \mathcal{T}_{\mathcal{R}_{\mathscr{B}}}, \mathcal{I}_{\mathscr{B}})$ definiendo:

$$\mathcal{T}_{\mathcal{C}_{\mathscr{B}}} := \{\varepsilon, \rho\} \cup \mathscr{E} \cup \mathscr{P} \cup \{\top\}$$
$$\mathcal{T}_{\mathcal{R}_{\mathscr{B}}} := \{[b, \tau, d], [b, *, d],\ \exists\, \delta^{\Theta^{a,b}_{i,j}, \tau, \Theta^{c,d}_{i,k}} \in \mathscr{S}_{\mathscr{B}}\} \cup \{[\varepsilon, *, \varepsilon]\} \cup \{[\varepsilon, *, \rho]\} \cup \{[\rho, *, \rho] \cup \{[\top, *, \top]\}$$
$$\mathcal{I}_{\mathscr{B}} := \{\Theta^{a,-}_{i,j},\ \Theta^{c,-}_{i,k}\}_{\delta^{\Theta^{a,-}_{i,j}, -, \Theta^{c,-}_{i,k}}}$$

donde $\mathscr{S}_{\mathscr{B}}$ es la semántica asociada al *corpus* de ejemplo $\mathscr{B}$.

Intuitivamente, consideramos que el conjunto de conceptos $\mathcal{T}_{\mathcal{C}_{\mathscr{B}}}$ que manejaremos para el caso del *corpus* $\mathscr{B}$, se puede clasificar en entidades y propiedades, tal como se describe en la Tabla 9.6, y no se tiene en cuenta el orden seguido entre elementos similares y/o diferentes. Sólo se define una relación de subsunción entre las entidades individuales (resp. propiedades) y el correspondiente elemento genérico, $*$, tal y como se observa en la Fig. 9.14.

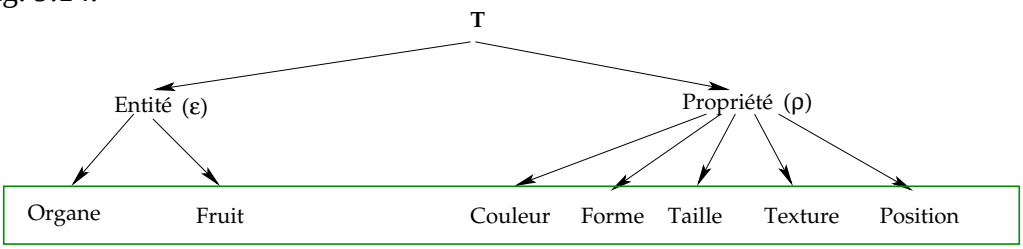


Figura 9.14: Conjunto de tipos primitivos de conceptos

Con respecto al conjunto de relaciones $\mathcal{T}_{\mathcal{R}_{\mathscr{B}}}$, se extraen directamente a partir de $\mathscr{S}_{\mathscr{B}}$ a través de la dinámica de transición, resumiéndose desde el punto de vista de las clases semánticas (tipos) de los términos que participan en ella. Como elementos adicionales, se añaden tripletas que representan cualquier posible transición en la semántica que

relacione conceptos genéricos. El orden parcial que consideramos en $\mathcal{T}_{\mathcal{C}_{\mathscr{B}}}$ es el inducido naturalmente por el ya definido en $\mathscr{T}$. Concretamente, la Fig. 9.15 muestra dicha jerarquía, aunque de un modo simplificado, evitando indicar la información asociada con la construcción del árbol sintáctico en la etiqueta τ de la dependencia.

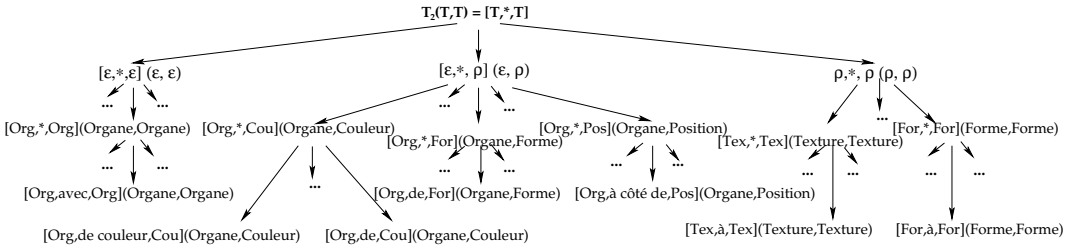


Figura 9.15: Algunos tipos de relaciones conceptuales

Así, el tipo relacional $[\rho, *, \rho]$ se puede especializar en otros diferentes. Por ejemplo, lo puede hacer en $[\textit{Tex}, *, \textit{Tex}]$, pero también en $[\textit{For}, *, \textit{For}]$. A su vez, cada uno de ellos puede especializarse en $[\textit{Tex}, \text{à}, \textit{Tex}]$ y $[\textit{For}, \text{à}, \textit{For}]$, donde *«à»* representa parte de la etiqueta τ de la dependencia. Finalmente, definimos los referentes individuales $\mathcal{I}_{\mathscr{B}}$ como un conjunto de formas del *corpus* $\mathscr{B}$, tal y como se observa en la Fig. 9.16.

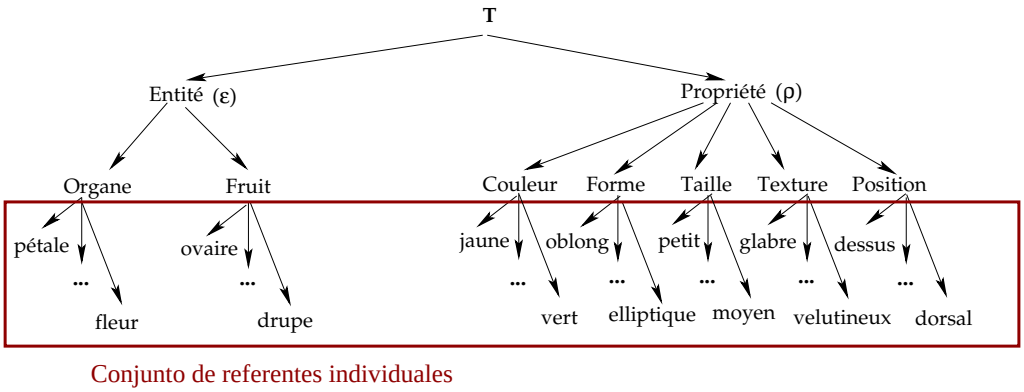


Figura 9.16: Conjunto de referentes individuales

Ahora estamos en disposición de presentar los GCB's que vamos a considerar sobre este soporte. Nuestro punto de partida es la semántica $\mathscr{S}_{\mathscr{D}_m}$ asociada a cada uno de los documentos que constituyen el *corpus*

$$\mathscr{B} = \bigcup_{m \in M} \mathscr{D}_m$$

donde M es el número de estos documentos:

$$\mathcal{C}_{\mathscr{D}_m} := \{\Theta^{a,b}_{i,j}, \Theta^{c,d}_{i,k}\}_{\delta^{\Theta^{a,b}_{i,j}, \_, \Theta^{c,d}_{i,k}} \in \mathscr{S}_{\mathscr{D}_m}} \qquad \mathcal{R}_{\mathscr{D}_m} := \{[b, \tau, d],\ \exists\ \delta^{\Theta^{\_,b}_{\_,\_}, \tau, \Theta^{\_,d}_{\_,\_}} \in \mathscr{S}_{\mathscr{D}_m}\}$$

$$\mathcal{A}_{\mathscr{D}_m} := \bigcup_{\delta^{\Theta^{a,b}_{i,j},\tau,\Theta^{c,d}_{i,k}} \in \mathscr{S}_{\mathscr{D}_m}} \{([b,\tau,d],1,\Theta^{a,b}_{i,j}),([b,\tau,d],2,\Theta^{c,d}_{i,k})\}$$

$$\mathcal{E}_{\mathscr{D}_m}(X) := \begin{cases} [b,\Theta^{a,\_}_{i,j}] & \text{si} \quad X = \Theta^{a,b}_{i,j} \in \mathcal{C}_{\mathscr{D}_m} \\ X & \text{si} \quad X \in \mathcal{R}_{\mathscr{D}_m} \\ 1 & \text{si} \quad X = (\_,1,\_) \in \mathcal{A}_{\mathscr{D}_m} \\ 2 & \text{si} \quad X = (\_,2,\_) \in \mathcal{A}_{\mathscr{D}_m} \end{cases}$$

Brevemente, un nodo conceptual en $\mathcal{C}_{\mathscr{D}_m}$ es cualquier término involucrado en la semántica $\mathscr{S}_{\mathscr{D}_m}$, mientras que los nodos relaciones en $\mathcal{R}_{\mathscr{D}_m}$ son elementos de $\mathcal{T}_{\mathcal{R}_{\mathscr{B}}}$ asociados a las transiciones en $\mathscr{S}_{\mathscr{D}_m}$. El multiconjunto de aristas $\mathcal{A}_{\mathscr{D}_m}$ contiene en este caso únicamente las relaciones binarias correspondientes a los términos gobernante (resp. gobernado) de la primera tripleta (resp. la segunda).

En cuanto a la función de etiquetado $\mathcal{E}_{\mathscr{D}_m}$, permite recuperar la clase semántica y el token asociado a un término dado representando un concepto, al tiempo que implementa la identidad en las relaciones, ya que en nuestro caso las construimos directamente a partir de la semántica del *corpus*. El valor de esta función sobre las aristas identifica las gobernantes (1) y las gobernadas (2).

Ejemplo 9.10 *Supongamos que tenemos la frase* «Feuilles à nervures denticulées» (`«Hojas con nervaduras dentadas»`)*, y que después de la fase de adquisición, y considerando las correspondencias comentadas, su representación en forma de* GCB *es la que se muestra en 9.17, donde* Org=Organe *y* For=Forme.

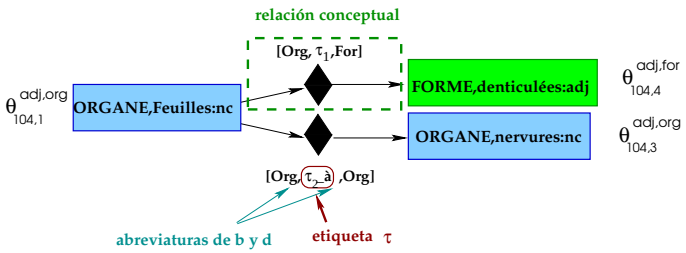


Figura 9.17: Ejemplo de GCB para «*Feuilles à nervures denticulées*»

En esta figura podemos observar que si aplicamos la función de etiquetado $\mathcal{E}_{\mathscr{D}_m}$ sobre el nodo conceptual $\Theta^{nc,org}_{104,1}$ obtenemos lo siguiente: $\mathcal{E}_{\mathscr{D}_m}(\Theta^{nc,org}_{104,1}) = [Org, \Theta^{nc,\_}_{104,1}]$. Del mismo modo, si lo aplicamos sobre el nodo conceptual $\Theta^{nc,org}_{104,3}$ obtenemos $\mathcal{E}_{\mathscr{D}_m}(\Theta^{nc,org}_{104,3}) = [Org, \Theta^{nc,\_}_{104,1}]$.

Si tomamos ahora las aristas que van del nodo concepto $\Theta^{nc,org}_{104,1}$ al nodo relación $[Org, \tau_2\_à, Org]$, y de éste al nodo $\Theta^{nc,org}_{104,3}$, sabemos gracias al sentido de la flecha cual es el nodo gobernante y cual el gobernado. Para más detalle, usando $\mathcal{E}_{\mathscr{D}_m}(([Org, \tau_2\_à, Org], 1, \Theta^{nc,org}_{104,1})) = 1$.

■

CAPÍTULO X

El marco de evaluación

Nuestro objetivo ahora es tratar de discriminar la eficacia entre los diferentes sistemas de RI[1] aplicando las medidas indicadas en la Sección 6.4. En este sentido, hemos propuesto una modificación para el caso de la evaluación utilizando la medida *contador de referencia* debido a lo difícil de su justificación. De la misma manera, otro objetivo consiste en elegir adecuadamente un conjunto de consultas minimal con el fin de evaluar nuestro sistema de RI comparándolo con una colección de las ya existentes, tomando como referencia los diferentes niveles de dificultad en su resolución por parte del usuario. Nuestra contribución en este punto se localiza en la novedad de la técnica empleada ya que a nuestro conocimiento no se ha presentado ni documentado, hasta ahora, ninguna para este fin concreto.

10.1 | Sistemas de RI con ordenación en base a contadores de referencia ponderados

Dado que las fórmulas resultantes indicadas en la Sección 6.4.4 son poco claras y difíciles de entender, y que algunas de las elecciones en esta propuesta de ordenación son difíciles de justificar, ya que no se han argumentado razones convincentes para presentar la constante Δ, ni los (muy complejos) valores de ω_{j_i}, se propone modificar ligeramente el planteamiento original.

Definición 10.1 *Sean* $\{\sigma_i\}_{i\in I}$ *un conjunto de sistemas de* RI, $\mathcal{D}$ *una colección documental,* $\mathcal{Q} = \{c_j\}_{j\in J}$ *un conjunto finito de tópicos (consultas), y* $\{\varrho_{j_i}\}_{j_i\in J_i}$ *las puntuaciones normalizadas*[2] *asociadas a* $\{\text{reco}(\sigma_i, c_j, \mathcal{D})\}_{j_i\in J_i}$. *Sea también* $\forall\ m, n \in$

[1] entre los que encontramos el nuestro.

[2] asumimos, sin pérdida de generalización, que estas puntuaciones están en el intervalo $[0, 1]$.

$[1, |\mathcal{D}|]$*:*

$$\hat{a}(\text{reco}(\sigma_i, c_j, \mathcal{D})_{j_i}) := \sum_{\text{reco}(\sigma_k, c_j, \mathcal{D})_{k_l} \in \gamma(\text{reco}(\sigma_i, c_j, \mathcal{D})_{j_i})} \omega_{k_l} \;\; \textit{(resp. } \varrho_{k_l}\textit{)}, \; \textit{donde } \omega_{k_l} := \begin{cases} 1 & \text{si } l = 1 \\ \frac{1}{log_b(l)} & \text{en cualquier otro caso} \end{cases}$$

y

$$\hat{\omega}_{j_i} := \begin{cases} 1 & \text{si } j_i = 1 \\ \frac{1}{log_b(j_i)} & \text{en cualquier otro caso} \end{cases}$$

siendo las funciones de peso asociadas a la relevancia de las posiciones de referencia y a los documentos originales, respectivamente. Denotamos a la expresión

$$\sum_{j_i \in J_i} \hat{\omega}_{j_i} . \hat{a}(\text{reco}(\sigma_i, c_j, \mathcal{D})_{j_i}) \tag{10.1}$$

como $\text{CRP}_{\text{OL}}(\sigma_i, c_j, \mathcal{D})$ *(resp.* $\text{CRP}_{\text{PL}}(\sigma_i, c_j, \mathcal{D})$*), denominado como el* contador de referencia ponderado basado en la ordenación logarítmica *(resp.* basado en la puntuación logarítmica*)* de σ_i sobre el tópico c_j para la colección $\mathcal{D}$.

■

Siguiendo el mismo proceso que se aplicó para introducir MCRP_{O} (resp. MCRP_{P}), ahora podemos introducir MCRP_{OL} (resp. MCRP_{PL}), lo cual proporciona las medidas de ordenación usando contadores de referencia ponderados que tendremos en cuenta en este trabajo. Tomaremos $b = 2$.

10.2 | Selección del conjunto de tópicos

Como visión general, consideramos una técnica de muestreo estratificado para seleccionar *un conjunto inicial de tópicos*, sobre el que más adelante aplicaremos una técnica de minimización para reducir su tamaño sin perder su poder de discriminación. Esto nos va a permitir simplificar en gran medida la tarea de pruebas que aquí es especialmente compleja por cuanto no sólo pretendemos estimar la eficiencia del sistema de RI, sino también identificar los factores que impactan en términos de imprecisión y de incompletud.

10.2.1 | El tamaño de la muestra inicial

Una cuestión fundamental consiste en determinar el tamaño del conjunto de consultas que deberíamos utilizar para evaluar la propuesta, para lo que tomamos como referencia la discusión que plantean al respecto Guiver *et al.* [121], a su vez referida a diversos trabajos anteriores. En este sentido, los autores ponen de manifiesto una clara evolución en el

estado del arte, atribuyendo las primeras estimaciones a Jones y a van Rijsbergen [296], que llegaron a la conclusión de que usando un número de 75 no era suficiente, 250 eran por lo general aceptable, e incluso 1.000 podían llegar a ser necesarios. Más tarde Zobel [356] apoya la idea de que un conjunto de 25 consultas ya permite realizar un trabajo razonable, mientras que Buckley y Voorhees [37] proporcionan la primera evidencia efectiva de que el número de tópicos necesarios para un buen experimento es de al menos 25, aunque 50 parece ser mejor. Más recientemente, en el contexto de las evaluaciones al estilo TREC, Webber *et al.* [340] afirman que se requieren de unas 150 consultas para distinguir de forma fiable entre sistemas de RI, aunque por lo general sólo se consideran 50 [332]. En nuestro caso, hemos seleccionado en un primer momento una muestra inicial de 150 tópicos.

10.2.2 | El proceso de muestreo

En primer lugar, clasificamos nuestro espacio muestral[3] (población) siguiendo dos criterios independientes, cada uno formando su propia partición, y que creemos puede estar correlacionada con la noción intuitiva de dificultad (durante la resolución) de las consultas. Esta última constituye la variable dependiente deseada para el muestreo, una elección basada en Mizzaro *et al.* [208] que sugiere que es un factor importante en los tópicos para discriminar eficazmente entre sistemas de RI. En la práctica, introducimos de manera concisa estos criterios mediante sus variables asociadas:

- La *especificidad del tópico*, entendiéndola como el nivel de detalle con el que el usuario la expresa. Consideramos tres niveles diferentes: alto, medio y bajo.
- El *tipo de respuesta* devuelto por un motor de búsqueda siguiendo un enfoque conceptual: aproximado, plausible y parcial. Asumimos aquí que una consulta pertenece a un determinado tipo cuando el conjunto de respuestas de esa clase dentro de las 10 primeras devueltas por el sistema[4] posee un mayor peso estimable que el correspondiente a los demás tipos. Por lo tanto, es necesario fijar la relación μ_u (resp. μ_a) que limita el número de uniones (resp. de agregaciones) asociadas a respuestas plausibles (resp. parciales), así como calcular formalmente dicho peso.

Estos criterios también nos van a permitir combinar ambos puntos de vista, el del usuario y el del sistema de RI. Con el fin de equilibrar la muestra que nos va a servir como conjunto inicial de tópicos, tendremos que minimizar (resp. maximizar) la variabilidad dentro de (resp. entre) las subpoblaciones (estratos) correspondientes a las diferentes particiones. Por lo tanto, distribuimos la muestra entre las tres subpoblaciones introducidas para cada

[3] formado por la totalidad de las posibles consultas a aplicar sobre nuestro *corpus* $\mathscr{B}$.

[4] lo que aproximadamente se corresponde con la primera página de resultados devueltos por un motor de búsqueda cualquiera, justo el límite por encima del cual el usuario deja de mostrar interés en la revisión de las respuestas [117].

uno de ellas[5], lo que proporciona homogeneidad en todos los niveles de la estratificación. Asimismo, los tópicos de un determinado estrato de una de las particiones se reparten equitativamente entre los estratos de la otra. De este modo, aseguraríamos que la probabilidad de que una de las consultas de la muestra tenga un tipo de respuesta y una especificidad dadas sea aproximadamente la misma, cualquiera que fuera la combinación considerada para estas variables. De esta manera, esperamos mejorar la precisión y la eficiencia de la estimación, sacar conclusiones sobre las subpoblaciones y permitir un mayor equilibrio estadístico en las pruebas sobre las diferencias entre las particiones. Para lograr este objetivo hemos puesto en práctica un cuidadoso proceso de selección.

En relación con la especificidad del tópico, partimos de una colección de tópicos propuestos por expertos y repartida en tres estratos, de tal manera que las consultas de uno se obtienen refinando el contenido de las del estrato anterior. El objetivo es integrar, en número similar, los tópicos con especificidad alta, media y baja. Más en detalle, consideramos una colección inicial de tópicos verificando:

$$\mathcal{Q} := \{\mathcal{Q}_i^{ea}\}_{i\in I} \cup \{\mathcal{Q}_i^{em}\}_{i\in I} \cup \{\mathcal{Q}_i^{eb}\}_{i\in I},\ \mathcal{Q}_i^{ea} \succ \mathcal{Q}_i^{em} \succ \mathcal{Q}_i^{eb},\ \forall i \in I$$

donde $\succeq$ es el orden parcial naturalmente inducido en el espacio muestral por la especificidad detectada por los expertos.

Con respecto al tipo de respuesta, en primer lugar tomamos el valor $\mu_u = 0'34$ (resp. $\mu_a = 0'18$) con el fin de moderar el número de respuestas plausibles devueltas (resp. parciales)[6], lo que equivale a aplicar un muestreo ajustado con la probabilidad adecuada.

Una vez que se ha hecho esto, es necesario introducir algún criterio para medir el peso de un determinado tipo de respuesta en un conjunto finito de éstas, repartiéndolo equilibradamente entre los tipos considerados. Aquí, asumimos que no sólo hemos de tener en cuenta el número de respuestas de determinado caso, sino también la posición de éstas en la ordenación. Por lo tanto, el tipo de respuesta que aparece más abajo en la lista resultante de la búsqueda debería ser penalizado a la vez que se reduce el grado del valor de relevancia. Ello nos sitúa en un contexto equiparable al considerado en la determinación de las medidas de evaluación basadas en ordenación de los sistemas de RI y, más concretamente, en el proceso de construcción de la medida GAARN, que nos servirá ahora de inspiración para introducir la noción de *peso acumulado descontado* asociada a un tipo de respuesta dada.

Definición 10.2 *Sea σ un sistema de* RI, *$\mathcal{D} = \{d_i\}_{i\in I}$ una colección documental y $\mathcal{Q} = \{c_j\}_{j\in J}$ un conjunto finito de tópicos (consultas). Se define el* peso acumulado descontado de σ sobre el tópico c_j para un tipo de respuesta ι y una colección documental

[5] esto implica que asociamos 50 consultas por estrato, el mismo número considerado por el protocolo clásico del TREC [332] para la evaluación de sistemas de RI.

[6] el número de respuestas plausibles y, especialmente, de parciales pueden incrementar artificialmente su número debido al hecho de que se generan aplicando mecanismos que pueden hacer crecer indefinidamente el tamaño de los GCB's asociados a las consultas, algo que no ocurre con las aproximadas.

$\mathcal{D}$ con tamaño de selección $p \in [1, |rec(\sigma, c_j, \mathcal{D})|]$ *como:*

$$\text{PAD}(\sigma, \iota, c_j, \mathcal{D})_p := \delta_\iota^{\text{tipo}(\text{reco}_{(\sigma, c_j, \mathcal{D})_1})} + \sum_{k=2}^{p} \frac{\text{tipo}(\text{reco}(\sigma, c_j, \mathcal{D}))_k}{log_b(k)} \tag{10.2}$$

donde tipo *devuelve el tipo de respuesta que le sirve de argumento, y* δ_i^j *es la función conocida como* delta de Kronecker*, el cual se define de la siguiente manera:*

$$\delta_i^j := \begin{cases} 1 & \text{si } i = j \\ 0 & \text{en cualquier otro caso} \end{cases} \tag{10.3}$$

■

En nuestro caso particular, tomamos $p = 10$, $b = 2$ y nuestra propuesta de RI conceptual como σ, lo cual implica que $\iota \in \{aproximada, plausible, parcial\}$. En la práctica, el equipo de expertos emplea la medida PAD para alcanzar una distribución uniforme para la muestra basada en el tipo de respuesta, teniendo en cuenta simultáneamente el criterio de especificidad previamente descrito. Como resultado, se consigue un conjunto inicial de tópicos que verifica todas las restricciones descritas anteriormente a partir de ambos puntos de vista: heterogeneidad entre estratos en las diferentes particiones y homogeneidad en todos los niveles de estratificación. En este sentido, las Figs. 10.1, 10.2 y 10.3 muestran estas subpoblaciones. Esto nos coloca en el punto de comienzo de la fase de minimización que introducimos en tres pasos.

10.2.3 | Selección de tópicos individuales para un sistema dado

El primer acercamiento para tratar con la selección de tópicos pasa por fijar una estrategia de estimación de la adecuación de una consulta individual para medir el rendimiento de un proceso de RI. En este sentido, y tomando como fuente de inspiración la experiencia del TREC, la medida PM mide la eficacia de un sistema σ sobre un tópico individual $c \in \mathcal{Q}$ para una colección documental $\mathcal{D}$, lo que aparentemente podría resolver la cuestión.

Sin embargo, situándonos en el marco de la valoración tipo máquina, no podemos concluir que σ presente un mejor rendimiento para el tópico c que en el tópico $\tilde{c}$ (resp. que σ considera más fácil a c que $\tilde{\sigma}$), en base al dato $\text{PM}(\sigma, c, \mathcal{D}) > \text{PM}(\sigma, \tilde{c}, \mathcal{D})$ (resp. $\text{PM}(\sigma, c, \mathcal{D}) > \text{PM}(\tilde{\sigma}, c, \mathcal{D})$). Simplemente c podría ser un tópico más sencillo[7] y $\tilde{c}$ uno difícil[8] (resp. σ podría ser un buen sistema[9] y $\tilde{\sigma}$ uno malo[10]). Esto nos lleva a volver

[7] esto es, una consulta sobre la cual todos o la mayoría de los sistemas de RI tienen un buen desempeño.

[8] es decir, una consulta sobre la que todos o la mayoría de los sistemas de RI tienen un desempeño deficiente.

[9] es decir, un sistema cuya efectividad se extienda a todos o a la mayoría de las consultas difíciles.

[10] es decir, un sistema cuya efectividad se limita a las consultas fáciles.

```
Quelque chose de pubescent.
Je cherche une plante avec un rachis d’une certaine texture.
Quelles sont les plantes avec un limbe de couleur?
Les plantes avec un limbe de couleur et fleur d’une certaine texture.
Je cherche quelque chose de relativement court.
Je cherche des graines avec des arilles d’une certaine forme.
Quelles sont les plantes qui ont une partie courte?
Je veux savoir celles qui ont une partie longue?
Elles doivent avoir quelque chose d’obtus.
Quelles sont celles qui ont un organe charnu?
La plante qui a des pétales linéaires et quelque chose frêle.
Je cherche un organe cylindrique.
Je cherche un fruit ovoïde.
Quelles sont les parties qui sont grêles ou acuminées?
Elles doivent avoir quelque chose d’une certaine forme.
Je cherche une plante qui a le pistil d’une certaine taille.
Quelles sont les plantes qui ont une partie d’une certaine taille?
Je cherche un fruit obtus.
Quelles sont celles qui ont un organe charnu ou un fruit obtus?
Je cherche celles qui ont un fruit avec les lobes ciliés.
Corolle avec les organes ciliés.
Quelles sont les parties qui ont des rhizomes?
Je cherche ceux qui ont une fronde d’une certaine couleur.
Je cherche une couleur grande.
Je cherche des fougères avec des rhizomes d’une certaine texture.
Je cherche une partie de la penne à une certaine position.
Je veux savoir celles qui ont des sépales latéraux d’une certaine couleur.
Je cherche une inflorescence vivace avec une certaine texture.
Je veux savoir quelles sont les fougères d’une certaine taille qui ont des lobes.
Elles doivent avoir des dents asymétriques ou de certaine forme.
Je cherche quelque chose d’étalée avec des lobes linéaires.
Fruit d’une certaine forme.
Quelles sont les plantes qui ont certaines parties avec un limbe pubescent?
Fougères terrestres avec quelque chose portant des écailles.
Je cherche des parties basales ou basilaires.
Je cherche des couleurs blanchâtres.
Sépales ou quelque chose d’autre jaune
La plantes qui a des anthères avec quelque chose long.
Quelles sont celles qui ont quelque chose d’alterne avec une partie acuminée?
Sore à indusie d’une certaine couleur et taille
Quelque chose sessile et sigmoïde.
La plante a un organe samaroïde ou linéaire.
Quelles sont celles qui ont un éperon d’une certaine forme ou spiciforme?
Les plantes qui ont les restes du rostelle de certaines formes.
Contrefort de certaine taille et forme
Je cherche des organes médians ou très larges.
Cette plante a des parties dentées ou acuminées.
Le limbe a quelques choses d’acuminés.
Je veux savoir quelles sont celles qui ont une nerville portant une veinule à une
position.
Je cherche quelque chose portant des écailles de certaines couleurs.
```

Figura 10.1: Subpoblación de tópicos con nivel de especificidad bajo

```
Plantes avec stipules.
Quelles sont les plantes qui ont des stipules persistantes?
Je cherche les plantes qui ont des bractées pubescentes.
Plantes avec des gousses longues de 14 cm.
Je cherche les plantes qui ont des graines noires.
Quelles sont les plantes qui ont des pétales onguiculés?
Plantes avec graine obovoïde.
La plante a des feuilles obtuses.
Limbe denté ou acuminé.
Je veux savoir quelles sont celles qui ont des graines avec arilles.
Quelles sont les plantes qui ont des pinnules sur le costae canaliculées?
Je veux savoir quelles sont les plantes qui ont un rhizome portant des écailles.
Quelles sont les plantes qui ont un pétiole long de 9 cm?
Le sépale dorsal est mince.
Je veux savoir quelles plantes ont des sépales latéraux.
Plantes avec des feuilles acuminées.
Plantes avec 1 inflorescence dense.
Quelles sont les plantes qui ont des bractées florales?
Je cherche des feuilles avec des folioles elliptiques.
Quelles sont les plantes qui ont le limbe des feuilles coriace?
Quelles sont celles avec des pétioles larges ou longs?
Je cherche des plantes avec les pétales et feuilles falciformes.
Une gousse samaroïde ou linéaire.
Plantes qui a un éperon cylindrique et spiciforme.
Le staminode ou la drupe est charnu.
Quelles sont les plantes qui ont un ovaire hirsute avec des ovules.
Je cherche un rameau avec des ombelles circulaires.
Quelles sont les plantes avec un calice et des glandes brillantes?
Je veux savoir quelles sont celles qui ont un calice avec des glandes et des périanthes
cupuliformes
La plante a un style falciforme ou glabre.
Quelles sont celles qui ont des pennes latérales ou des pennes inférieures?
Le reste du rostelle est trilobé.
Ces plantes ont les tubes du calice verts.
La plante qui a des anthères avec des déhiscences longues.
Tubercule unique.
Je cherche des gaines ou des nervures basales.
Sépales ou tépales jaunes.
Je veux savoir quelles sont celles qui ont des étamines avec des anthères connectives.
Anthères avec valves transversales.
Les plantes qui ont les aisselles des feuilles caduques.
Je veux savoir quelles sont les tubercules ellipsoïdes et uniques.
Cette plante a des contreforts ou les racines minces.
La plante a un tronc couvert d’écaille brune.
Un sore sur une nervure courte.
Je cherche un style à appendice uniflore.
Quelles sont les plante qui ont un limbe avec un lobe denté?
Le limbe a les lobes acuminés.
Une nerville portant une veinule circulaire.
Je cherche une plante qui a entre 12 - 14 ovules basales.
La plante a des racines portant des écailles foncées.
```

Figura 10.2: Subpoblación de tópicos con nivel de especificidad medio

```
Rachis grêle.
Plantes avec graine ovoïde.
Quelles sont les plantes qui ont les tiges relativement courtes?
Je veux savoir quelles sont les plantes qui ont les inflorescences relativement courtes.
Je cherche celles qui ont des gousses ligneuses très épaisses.
Plantes avec un fût étroit et cylindrique.
Quelles sont celles qui ont des feuilles oblongues ou oblongues-lancéolées?
Les plantes qui ont des feuilles obtuses ou arrondies.
Quelles sont les plantes qui ont un rachis grêle et pubescent?
Quelles sont les plantes qui ont des stipules velues et courtes?
Quelles sont celles qui ont une graine avec des arilles jaunes?
Quelles sont les plantes qui ont des graines noires avec des arilles jaunes?
On cherche celles qui ont une corolle blanc ou rose.
Quelle est celle qui a une graine obovoïde ou ovoïde?
Quelles sont les plantes qui ont les étamines externes avec des anthères de 4 mm?
Quelles sont les plantes qui ont des bractées florales membraneuses?
Quelles sont les plantes qui ont le labelle obtus ou ovale?
Quelles sont les plantes qui ont un labelle avec des nervures épaisses?
Quelles sont les plantes qui ont le pédicelle grêle et glabre?
La plante qui a des pétales minces et des sépales latéraux glabres.
La plante qui a des pétales linéaires et des bractées courtes.
Je cherche des feuilles alternes à nervures.
Quelles sont les plantes qui ont un labelle avec des nervures pubescentes?
Elles doivent avoir une gousse vive.
Je veux savoir quelles sont les plantes qui ont un rhizome portant une fleur en racème.
Je veux savoir quelles sont les plantes qui ont un arbrisseau portant des fleurs petites.
Quelles sont celles qui ont une corolle à lobes violets?
Fougères à rhizome petites.
Je cherche une plante avec limbe deltoïde et pétiole roussâtre.
Plantes qui ont un rhizome portant des écailles obtuses avec des frondes.
Je cherche celles avec un pétiole grisâtre et long de 9 cm.
Je veux savoir quelles sont celles qui ont des nervures espacées et bifurquées.
Je cherche des plantes avec des sépales latéraux linéaires.
Plante qui a le pétiole straminé.
Plantes avec des feuilles acuminées avec les nervures épaisses.
Elles doivent avoir des dents asymétriques.
Pennes dorsales alternes.
Je cherche celles qui ont un ovaire hirsute et des ovules hispides.
Je cherche des feuilles alternes avec des folioles elliptiques.
Tige étalée avec feuilles linéaires.
Quelles sont les plantes qui ont le limbe des feuilles sessiles coriace?
Fougères terrestres avec rhizome portant des écailles.
Quelles sont celles qui ont des sépales, des tépales ou des bractées jaunes?
Elles doivent avoir les anthères ou les valves longues avec des déhiscences.
Je veux celles qui ont le sore avec une indusie pâle et mince.
Une fronde qui a des pennes mucronés portant des sporanges.
Cette plante a une indusie entière, membraneuse et pâle
Quelles sont celles qui ont un limbe à lobe denté ou acuminé?
Ces plantes ont le foliole avec des lobes dentés ou acuminés.
Ces plantes ont les fleurs roses avec des pseudonervures ligneuses.
```

Figura 10.3: Subpoblación de tópicos con nivel de especificidad alto

nuestra atención al concepto de PMN_{MPM} donde, contrariamente a lo que ocurre con PM, la condición $\text{PMN}_{\text{MPM}}(\sigma, c, \mathcal{D}) > \text{PMN}_{\text{MPM}}(\sigma, \tilde{c}, \mathcal{D})$ nos permite inferir que un sistema de RI σ tiene un buen rendimiento en la consulta c y uno malo en $\tilde{c}$.

10.2.4 | Selección de un conjunto de tópicos para un sistema dado

Entre todas las técnicas inspiradas en el TREC y disponibles en el estado del arte para resolver esta cuestión, se optó por trabajar con la de Guiver *et al.* en [121]. El punto de partida es ahora la medida PPM, de hecho un indicador de la eficacia de un sistema de RI que nos orienta sobre su bondad, una vez que el conjunto de consultas ha sido fijado para una colección de documentos dada. La idea consiste en aplicar una búsqueda exhaustiva en todos los posibles subconjuntos de tópicos en una colección determinada. De esta forma, podemos centrarnos en la correlación más alta de estos valores de PPM con el del concepto de la colección, con el fin de estimar la bondad de la predicción sobre un subconjunto de consultas del rendimiento del sistema de RI.

Por otra parte, también podemos retomar aquí un razonamiento similar en el marco de la valoración tipo máquina, usando ahora valores PNPM en lugar de los PPM y teniendo en cuenta que estas dos métricas no siempre coinciden.

10.2.5 | Selección de un conjunto de tópicos para un conjunto de sistemas

A nuestro conocimiento, no se han presentado ni documentado propuestas, hasta ahora, a este respecto en el estado del arte. Nuestra estrategia se apoya tanto en el marco basado en la valoración de tipo humana como en el basado en la valoración tipo máquina, sobre la base de las técnicas presentadas anteriormente, lo mismo para la selección individual que para los conjuntos de consultas en sistemas de RI particulares. Sin embargo, aunque los pasos a aplicar para conseguirlo son los mismos, su naturaleza dependerá en cada momento del tipo de marco de trabajo elegido:

1. El primer paso consiste en generar, a partir de la muestra que sirve de conjunto inicial de tópicos, una colección de subconjuntos con distintas capacidades para medir el rendimiento del sistema en diferentes niveles, y que denominamos *colección de referencia de tópicos*. En el extremo superior (resp. en el inferior) de esta gradación, situaremos subconjuntos de consultas formadas exclusivamente por aquéllas consideradas difíciles (resp. fáciles) con el poder de discriminación más alto (resp. más bajo). Cualquier tópico no catalogado como difícil o fácil se considerará como medio. El tamaño de cada uno de estos subconjuntos será nuevamente de 50, siguiendo con la propuesta de Webber *et al.* [340].

   Se generan dos tipos de colecciones, dependiendo del marco que nos indique la estimación del nivel de sencillez de los tópicos. Por lo tanto, recurrimos a la opinión

de un experto en el dominio, en el caso de la estrategia basada en la valoración basada en tipo humano. Por el contrario, con respecto al criterio basado en máquina, se identifican las consultas difíciles (resp. las fáciles) con la mayor conectividad (resp. la menor conectividad) en el conjunto de sistemas de RI.

2. A continuación, se aplican a cada una de estas colecciones de referencia una estrategia de minimización con el fin de reducir su tamaño sin afectar perceptiblemente su poder de discriminación. El resultado constituirá dos conjuntos de *colecciones finales de tópicos*, uno especialmente orientado a una valoración basada en tipo humano y otro en tipo máquina, distinguiendo cada cual tres niveles de dificultad: alto, medio y bajo. Para calcular el primero, seguimos la técnica propuesta por Guiver *et al.* en [121] sobre la base de la medida de correlación PPM[11]. Dado que tanto la PPM y la PNPM se pueden calcular a partir de JREL's o PJREL's, finalmente obtenemos cuatro colecciones finales de tópicos. Dos de ellos consideran JREL's (resp. PJREL's) como base para calcular la PPM y la PNPM, uno usando una valoración basada en el tipo humano y otra aplicando una basada en el tipo máquina.

   La única cuestión pendiente ahora es determinar la composición de estos subconjuntos finales, un problema para el que los autores no proporcionan un criterio claro. En este sentido, hemos decidido escoger aquéllos candidatos cuya cardinalidad se encuentra en el intervalo $[1, 50)$, mientras alcance un nivel suficientemente alto de correlación PPM (resp. PNPM) con el correspondiente subconjunto de tópicos de referencia.

   En el caso de las consultas basadas en JREL's, tomamos un nivel de correlación PPM (resp. PNPM) con la correspondiente valoración basada en el tipo humano (resp. basada en tipo máquina) orientado a la colección de tópicos de referencia que sea superior o igual a $0'99999932$. Esto supone en una aproximación de tipo humano (resp. valoración tipo máquina) considerar una colección de subconjuntos finales con 12 tópicos (resp. con 10) para dificultades altas, 22 (resp. 15) para dificultades medias y 32 (resp. 8) para las bajas, que denominaremos *colección de tópicos tipo humano sobre* JREL*'s* (resp. *tipo máquina*), o brevemente, CTHJ (resp. CTMJ).

   En el caso de las consultas basadas en PJREL's, tomamos un nivel de correlación PPM (resp. PNPM) con la correspondiente valoración basada en el tipo humano (resp. basada en tipo máquina) orientado a la colección de tópicos de referencia que sea superior o igual a $0'9999990$. Esto supone en una aproximación de tipo humano (resp. valoración tipo máquina) considerar una colección de subconjuntos finales con 30 tópicos (resp. con 2) para dificultades altas, 29 (resp. 22) para dificultades medias y 24 (resp. 48) para las bajas, que denominaremos *colección de tópicos tipo humano sobre* PJREL*'s* (resp. *tipo máquina*), o brevemente, CTHPJ (resp. CTMPJ).

[11] ambos acercamientos se han descrito previamente cuando se introdujo la selección de un conjunto de tópicos para un sistema de RI individual

En este contexto, ninguna consulta de la muestra del conjunto inicial posee mayor probabilidad de ser incluido en el conjunto reducido final ni por su tipo de respuesta ni por su especificidad, sino que ello dependerá exclusivamente de su dificultad de resolución, determinada por cualquiera de los dos métodos antes descritos. Esto garantizará la objetividad y la validez de los resultados experimentales que se obtengan usando una muestra reducida. Sin embargo, parece razonable esperar que el protocolo que seguimos para mejorar la selección de consultas proporcione conclusiones sensiblemente diferentes en función del marco específico sobre el que se realice las pruebas. En efecto, los trabajos anteriores [207, 208] muestran que aunque un sistema de RI que quiera ser eficaz en el TREC tendrá que serlo sobre los tópicos fáciles, el sentido común indica que un motor de búsqueda eficaz debe demostrar su verdadero poder en los difíciles.

10.3 | El conjunto de sistemas de RI

Elegimos una muestra de cuatro plataformas de motores de búsqueda bien conocidas con el fin de servir como valores de referencia de comparación para estimar el rendimiento de la eficiencia de nuestra propuesta, que bautizamos como COGIR:

1. ZETTAIR (ver `http://www.seg.rmit.edu.au/zettair/`) es un motor de búsqueda de código abierto desarrollado por el *Search Engine Group* de la Universidad RMIT, desarrollado en C. Fue diseñado buscando simplicidad, así como velocidad y flexibilidad, y su principal característica es su capacidad de manejar grandes cantidades de texto. Este motor de búsqueda admite consultas de tipo booleano y como frases.

2. SOLR (ver `lucene.apache.org/solr/`) es una plataforma de búsqueda de código abierto del proyecto Apache Lucene. Sus características principales son que está escrito en JAVA y que se ejecuta como un servidor de búsqueda de texto independiente incluido dentro de un contenedor de servlets como es el caso de TOMCAT. Utiliza la librería de búsqueda de JAVA Lucene en su núcleo para la indexación completa de texto y posterior búsqueda. SOLR proporciona una búsqueda distribuida y la replicación de índices, impulsando la búsqueda y las características de navegación de muchos de los sitios Web más importantes.

3. TERRIER[12] (ver `http://ir.dcs.gla.ac.uk/terrier/`) es motor de búsqueda de código abierto altamente flexible, eficaz, y efectivo, fácilmente desplegable en colecciones de documentos a gran escala y desarrollado en la Universidad de Glasgow. Está escrito en JAVA y proporciona múltiples estrategias de indexación, como el de una sola pasada, de múltiples pasadas y de indexación a gran escala usando algoritmos de MapReduce.

[12] de TERabyte RetrIEveR

4. INDRI (ver `http://www.lemurproject.org/indri/`) es un motor de búsqueda de código abierto para gran escala, escrito en C++. Fue construido a partir del proyecto LEMUR (ver `http://www.lemurproject.org/`), el cual es un conjunto de herramientas diseñado para la investigación en el modelado de lenguaje y la RI. Este proyecto fue desarrollado gracias al trabajo cooperativo entre las Universidades de Massachusetts y de Carnegie Mellon.

Estos motores de búsqueda proporcionan un abanico representativo de los más populares en la actual oferta de buscadores, incluyendo tanto diferentes lenguajes de implementación como diferentes modelos de búsqueda.

PARTE IV

Trabajo experimental

CAPÍTULO XI

Resultados experimentales

Una vez formalizado el marco de evaluación, ya sólo queda introducir, visualizar e interpretar los resultados, teniendo en cuenta que la manera más sencilla de comparar los diferentes sistemas de RI es ordenándolos mediante valores decrecientes, de acuerdo con las diferentes métricas asociadas al rendimiento. A este respecto, vamos a seguir el mismo orden que el considerado anteriormente a la hora de introducirlas, en función de su tipo.

11.1 | Sistemas de RI con ordenación usando JREL's

En este nivel, hemos considerado el conjunto total de las diferentes métricas de rendimiento presentadas previamente (y en número de catorce) con el fin de experimentar con ellas, lo cual debería ser suficiente para detectar cualquier posible mal funcionamiento en nuestra propuesta, al tiempo que garantizamos la robustez de la evaluación. Así, los tests se realizaron sobre las dos colecciones de conjuntos de tópicos establecidas, CTHJ y CTMJ, buscando adecuar el criterio de la selección de tópicos al enfoque específico de ordenación, ambos basados en JREL's. Esto debería proporcionar fiabilidad al proceso.

11.1.1 | Usando una colección de conjuntos de tópicos basada en la valoración tipo humano

Tomamos aquí la CTHJ como colección de tópicos, que proporcionará una visión general del comportamiento de nuestra propuesta para hacer frente a la ordenación basada en JREL's sobre tópicos seleccionados mediante la valoración tipo humano. Esto debería constituir un protocolo de evaluación bien fundado.

11.1.1.1 | Medidas de evaluación basadas en conjuntos

Tratamos aquí con los resultados de las medidas P y C, que se muestran en las Figs. 11.1 y 11.2 respectivamente. Tal y como puede comprobarse, en cualquier caso los resultados indican una mayor precisión del modelo conceptual COGIR sobre los demás, a la vez que una mayor contención de la cobertura.

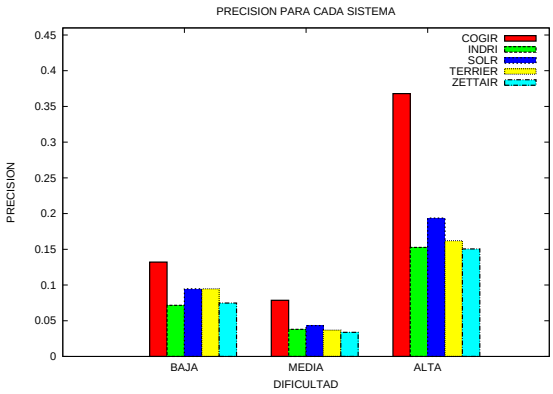


Figura 11.1: P sobre CTHJ usando JREL's

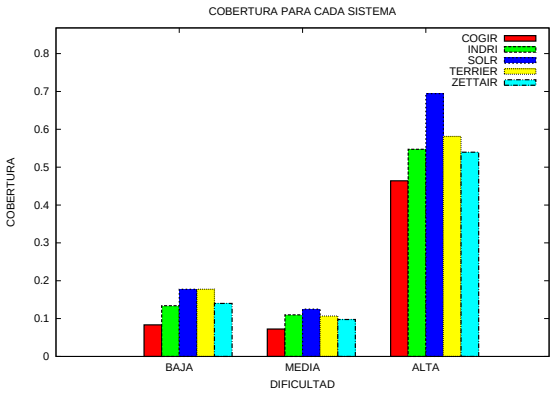


Figura 11.2: C sobre CTHJ usando JREL's

También se incluyen los tests para las métricas F y FR, a fin de tener en cuenta la proporción de documentos no relevantes que son recuperados. Los gráficos asociados se muestran en las Figs. 11.3 y 11.4, respectivamente. En este caso, los valores favorecen claramente al modelo conceptual frente a los otros para el conjunto de tópicos de mayor dificultad, esto es, en aquéllos con mayor poder de discriminación entre sistemas en lo que a evaluación se refiere. Sin embargo, los resultados son menos impactantes para los tópicos con menor poder de discriminación.

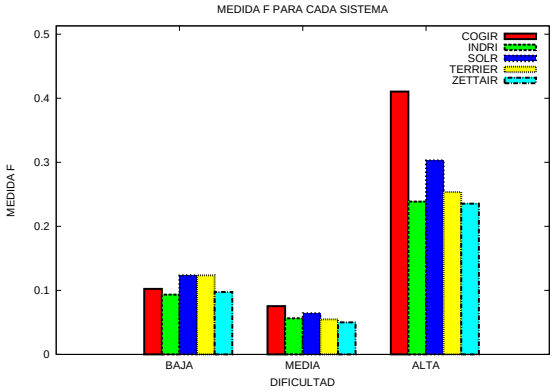


Figura 11.3: F sobre CTHJ usando JREL's

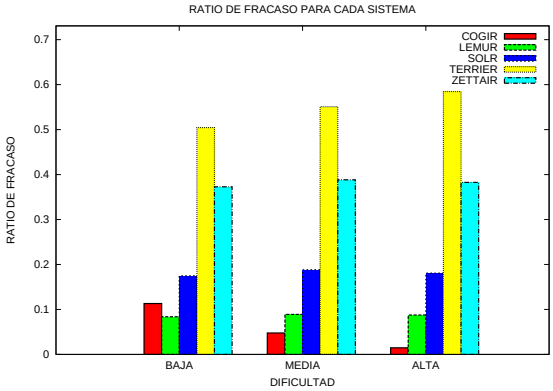


Figura 11.4: FR sobre CTHJ usando JREL's

11.1.1.2 | Medidas de evaluación basadas en ordenación

Tratamos aquí con los resultados de las medidas P@10 y C@10, que se muestran en las Figs. 11.5 y 11.6, respectivamente. Tal y como puede comprobarse, en cualquier

caso los resultados muestran una mayor precisión del modelo conceptual COGIR sobre los demás, a la vez que una mayor contención de la cobertura.

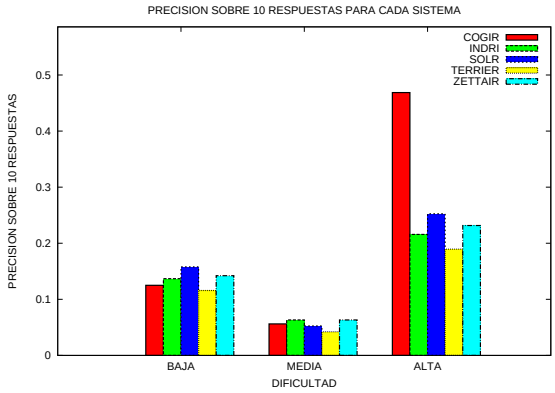


Figura 11.5: P@10 sobre CTHJ usando JREL's

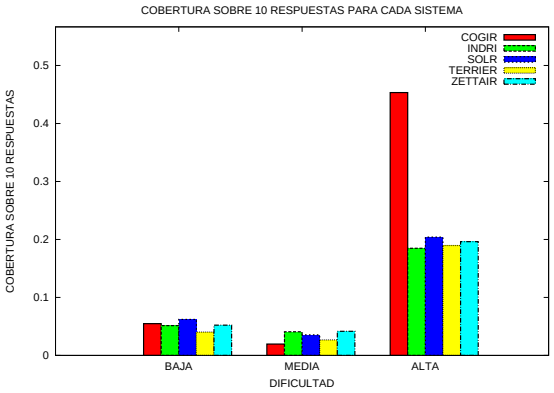


Figura 11.6: C@10 sobre CTHJ usando JREL's

Con el fin de estudiar la posible extensión de los resultados observados en la primera página al conjunto de respuestas obtenidas, calculamos PI_C para niveles 0 (resp. 0'10) de cobertura en la Fig. 11.7 (resp. en la Fig 11.8). De nuevo, como en los casos anteriores, vuelve a quedar patente el mejor comportamiento del modelo conceptual sobre los tópicos con mayor nivel de dificultad.

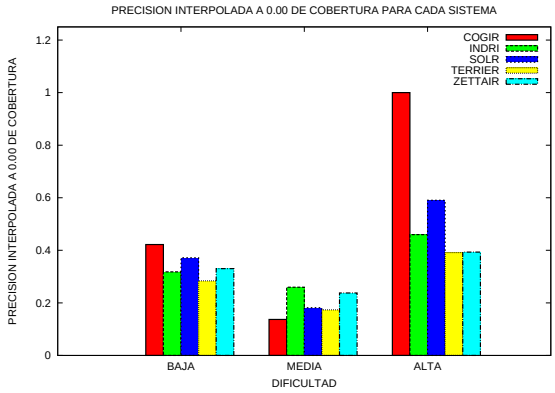


Figura 11.7$\text{PI}_{C=0'00}$ sobre CTHJ usando JREL's

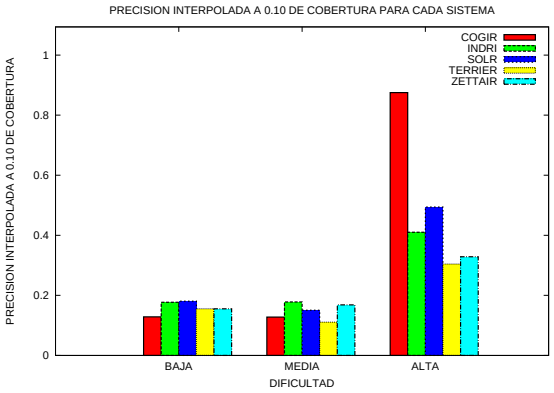


Figura 11.8$\text{PI}_{C=0'10}$ sobre CTHJ usando JREL's

Por su parte, las Figs. 11.9 y 11.10 vuelven a avalar la robustez del modelo conceptual sobre la base de las medidas R-P y PPM. Al tiempo, estos resultados destacan su rendimiento en el tratamiento de las consultas con mayor dificultad, manteniendo las prestaciones en relación al resto de entornos considerados en otro caso.

En cuanto a los valores obtenidos para PGPM y PREFB, éstos se muestran en las Figs. 11.11 y 11.12. Nuevamente vuelve a repetirse el comportamiento habitual, reflejándose un comportamiento similar en todos los sistemas cuando tratamos consultas

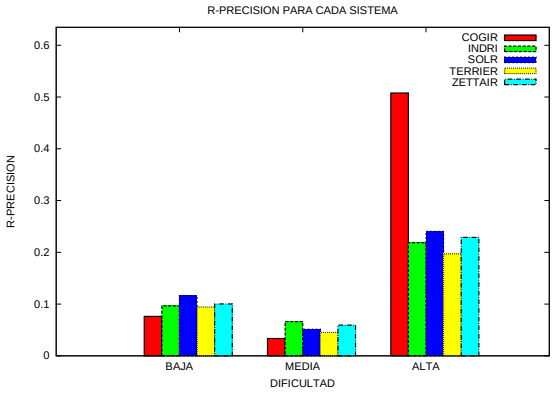


Figura 11.9: R-P sobre CTHJ usando JREL's

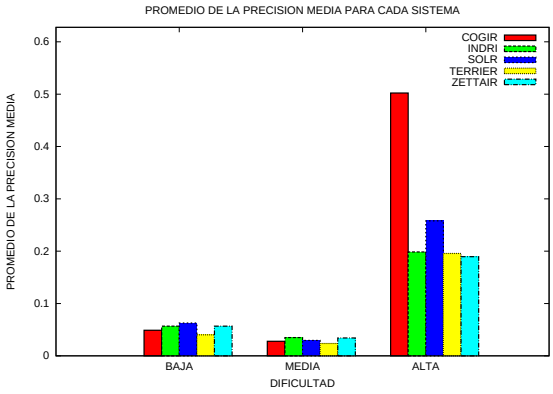


Figura 11.10: PPM sobre CTHJ usando JREL's

con un nivel de dificultad medio o bajo, siendo los resultados mucho mejores en otro caso para el modelo conceptual.

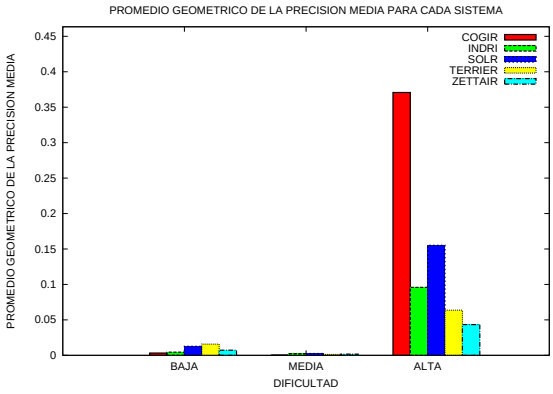


Figura 11.11: PGPM sobre CTHJ usando JREL's

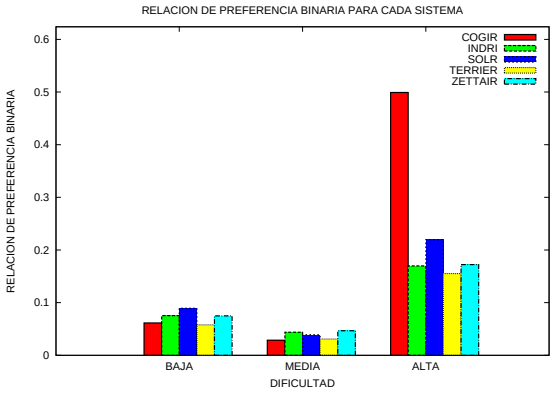


Figura 11.12ꝖPREFB sobre CTHJ usando JREL's

Finalmente, introducimos los valores para GAAR y GAARN en las Figs. 11.13 y 11.14, respectivamente. Al contrario de lo que ocurría en la totalidad de los casos anteriores, aquí los resultados son netamente superiores para el modelo conceptual en el caso de consultas de bajo nivel de dificultad, mientras que en el resto el comportamiento es similar al del conjunto de entornos comparados. Ello no es sorprendente, puesto que ya algunos autores [8] han advertido de los resultados posiblemente sorprendentes en lo que a la correlación con las medidas antes comentadas se refiere.

11.1.2 | Usando una colección de conjuntos de tópicos basada en la valoración tipo máquina

Ahora aplicamos el mismo conjunto de medidas anteriores sobre el conjunto de tópicos CTMJ. En este caso, el valor del experimento consiste en corroborar las conclusiones alcanzadas anteriormente.

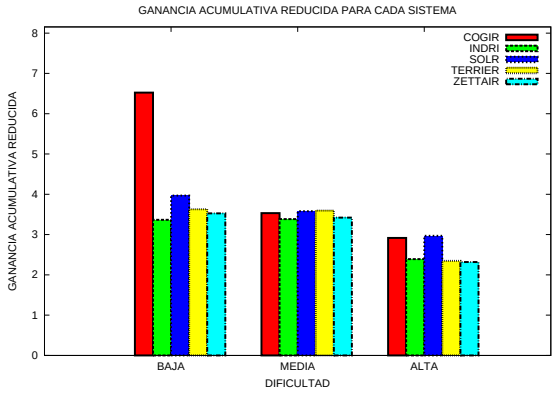


Figura 11.13:GAAR sobre CTHJ usando JREL's

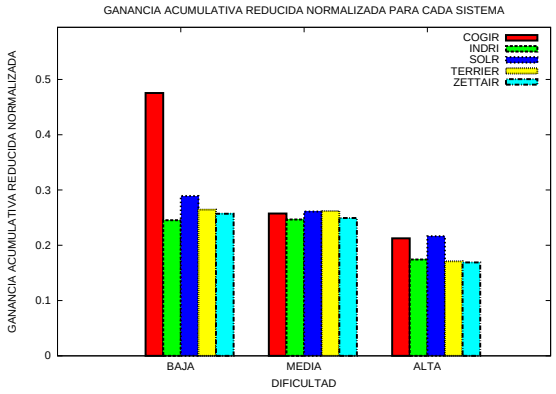


Figura 11.14: GAARN sobre CTHJ usando JREL's

11.1.2.1 | Medidas de evaluación basadas en conjuntos

Retomamos aquí el cálculo de las medidas P, C, F y FR, , cuyas gráficas son las que se observan en las Figs. 11.15, 11.16, 11.17 y 11.18, respectivamente. En todas ellas, el enfoque conceptual pone de manifiesto un empeoramiento en el funcionamiento sobre el conjunto de tópicos de dificultad alta, en comparación con los de tipo medio y bajo, aunque aún así logra los mejores resultados para las medidas P y F. Las otras dos medidas resultan estar entre los mejores.

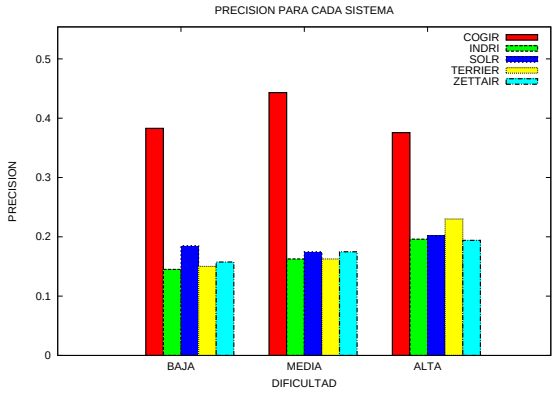


Figura 11.15: P sobre CTMJ usando JREL's

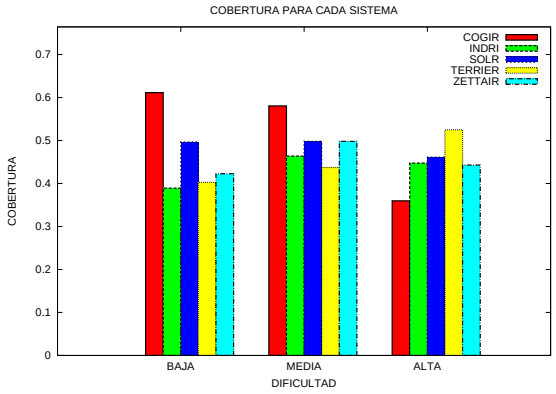


Figura 11.16: C sobre CTMJ usando JREL's

11.1.2.2 | Medidas de evaluación basadas en ordenación

Recalculamos la P@10, C@10, PI_C para los niveles 0 y 0'10 de cobertura, R-P, PPM, PGPM, PREFB, GAAR y GAARN en las Figs. 11.19, 11.20, 11.21, 11.22, 11.23, 11.24, 11.25, 11.26, 11.27 y 11.28, respectivamente. Las figuras muestran como COGIR consigue mejores resultados que los demás sistemas sobre todos los conjuntos de tópicos. Sin embargo, en contraposición a los obtenidos en el caso del CTHJ, proporciona un peor rendimiento sobre los tópicos de dificultad alta en comparación con los de tipo medio y bajo.

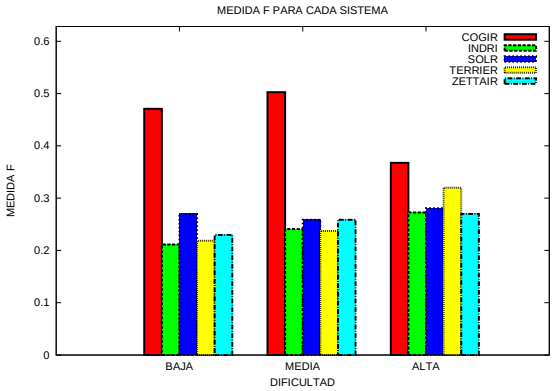


Figura 11.17: F sobre CTMJ usando JREL's

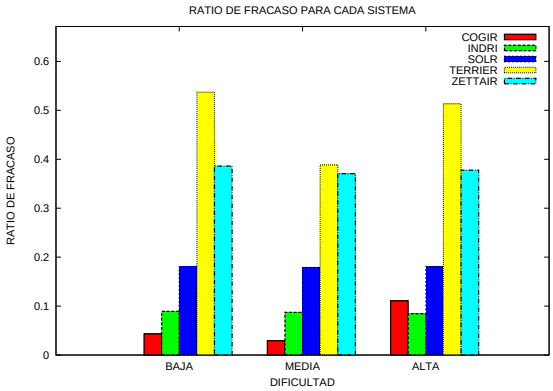


Figura 11.18: FR sobre CTMJ usando JREL's

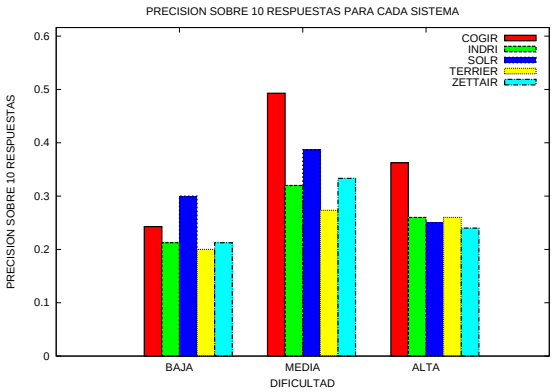


Figura 11.19:P@10 sobre CTMJ usando JREL's

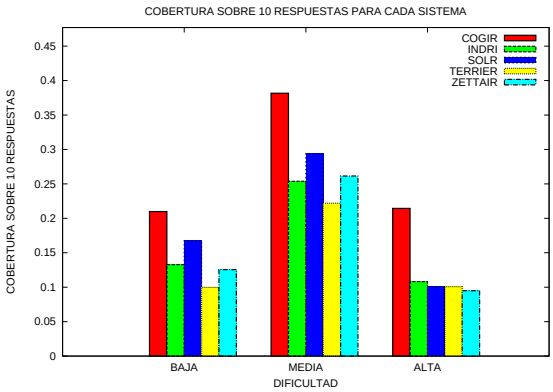


Figura 11.20:C@10 sobre CTMJ usando JREL's

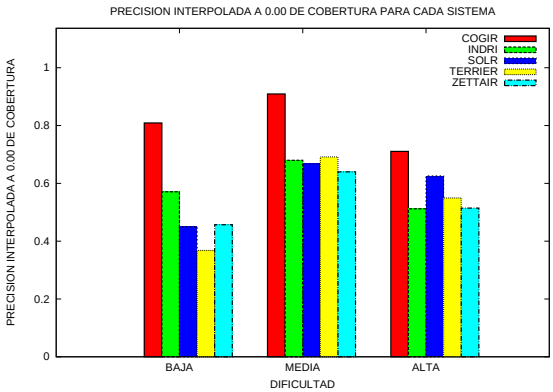


Figura 11.21: $PI_{C=0'00}$ sobre CTMJ usando JREL's

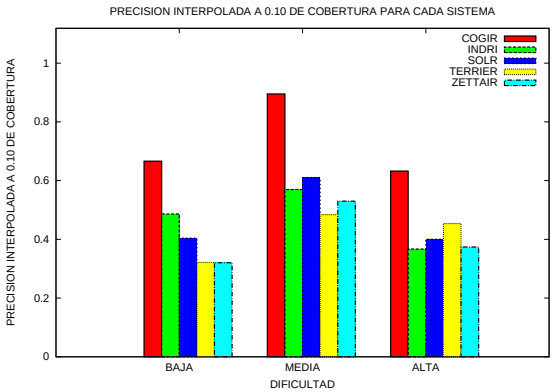


Figura 11.22: $PI_{C=0'10}$ sobre CTMJ usando JREL's

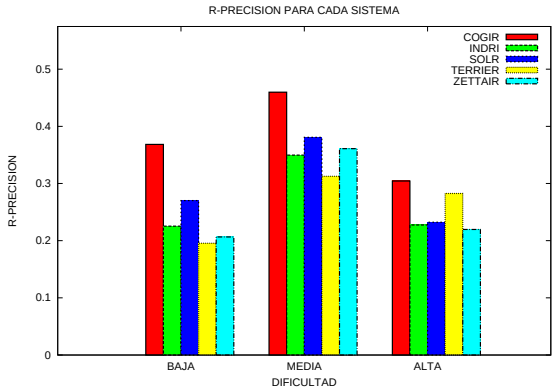


Figura 11.23: R-P sobre CTMJ usando JREL's

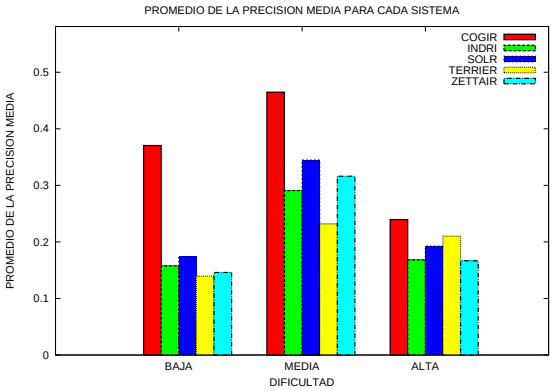


Figura 11.24: PPM sobre CTMJ usando JREL's

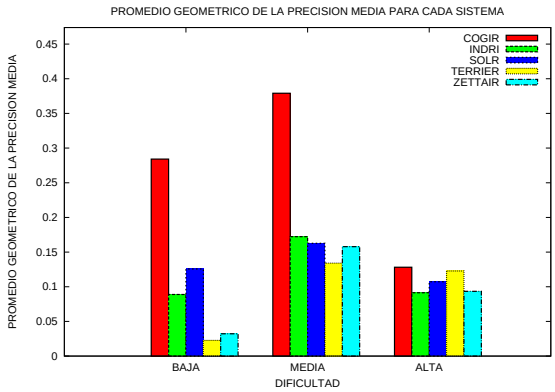


Figura 11.25:PGPM sobre CTMJ usando JREL's

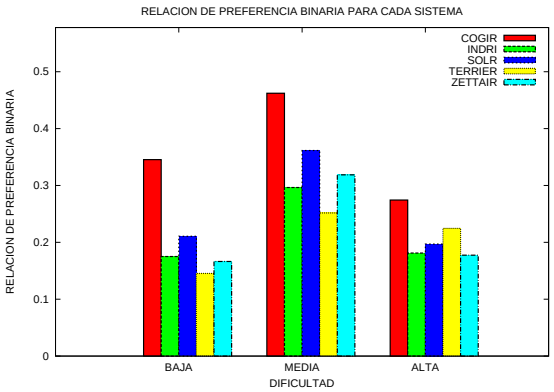


Figura 11.26PREFB sobre CTMJ usando JREL's

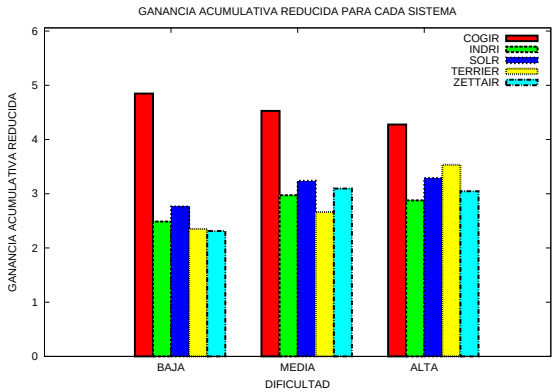


Figura 11.27:GAAR sobre CTMJ usando JREL's

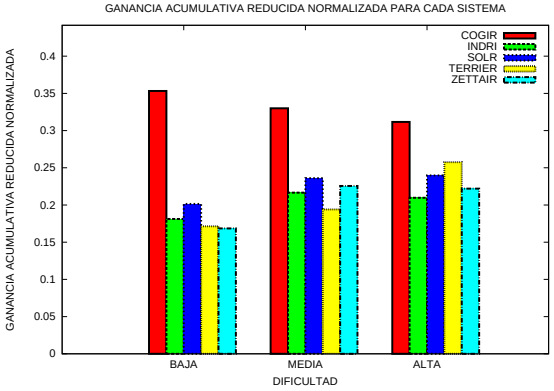


Figura 11.28: GAARN sobre CTMJ usando JREL's

11.2 | Sistemas de RI con ordenación usando PJREL's

Seguimos aquí el mismo protocolo aplicado a la ordenación orientada a JREL's, considerando el conjunto total de las diferentes métricas de rendimiento (y en número de catorce) usadas en los anteriores experimentos. La única diferencia es el par de conjuntos de tópicos que usaremos en adelante, remplazando el CTHJ (resp. CTMJ) por CTHPJ (resp. CTMPJ), buscando adecuar el criterio de la selección de tópicos al enfoque específico de ordenación, ambos basados en PJREL's.

11.2.1 | Usando una colección de conjuntos de tópicos basada en la valoración tipo humano

Tomamos aquí CTHPJ como colección de tópicos, que nos servirá para proporcionar una visión general de nuestra propuesta para hacer frente a la ordenación basada en PJREL's sobre los tópicos seleccionados usando la valoración tipo humano.

11.2.1.1 | Medidas de evaluación basadas en conjuntos

Tratamos aquí con los resultados de las medidas P y C, que se muestran en las Figs. 11.29 y 11.30 respectivamente. Los resultados obtenidos constituyen prácticamente un calco de los obtenidos para el caso de los JREL, otorgando nuevamente a COGIR los mejores resultados en cuanto a precisión, manteniendo la contención en la cobertura.

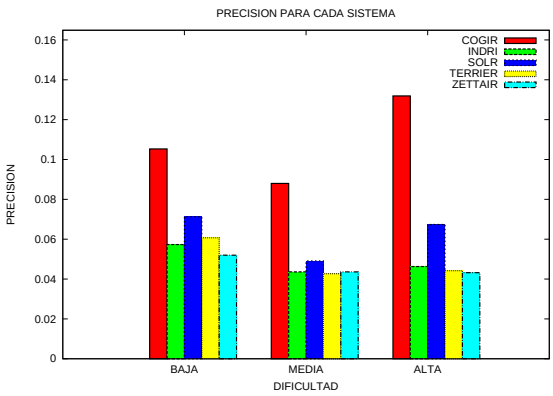


Figura 11.29: P sobre CTHPJ usando PJREL's

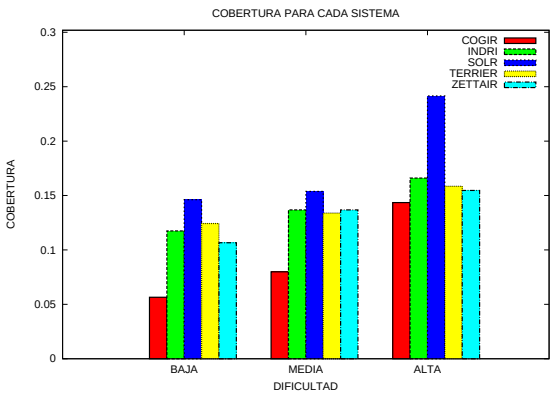


Figura 11.30: C sobre CTHPJ usando PJREL's

Como ya se hizo para los JREL's, también incluimos los resultados de las métricas F y FR en las Figs. 11.31 y 11.32, respectivamente. De nuevo, el modelo conceptual mejora sus resultados sobre el conjunto de tópicos de mayor dificultad, mientras que los resultados son menos impactantes sobre los tópicos con menor poder de discriminación.

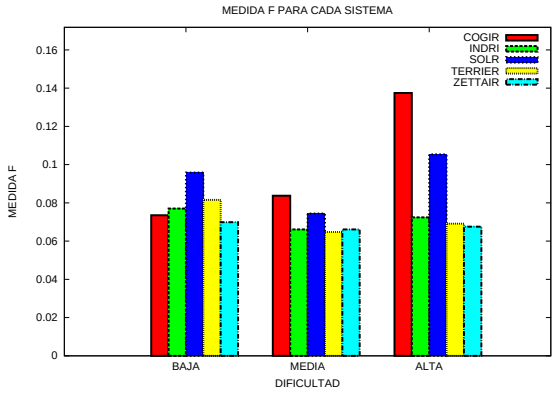


Figura 11.31: F sobre CTHPJ usando PJREL's

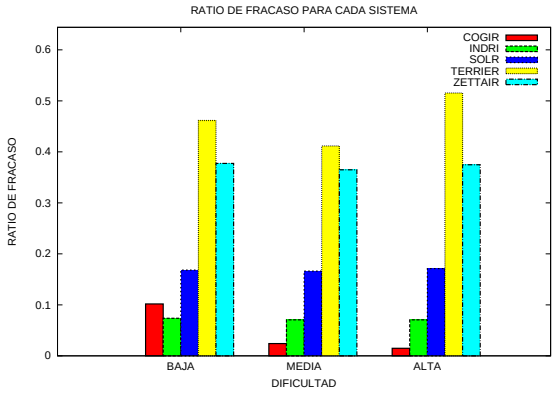


Figura 11.32: FR sobre CTHPJ usando PJREL's

11.2.1.2 | Medidas de evaluación basadas en ordenación

Calculamos la P@10, C@10, PI_C para niveles de cobertura 0 y 0'10, R-P, PPM, PGPM, PREFB, GAAR y GAARN en las Figs. 11.33, 11.34, 11.35, 11.36, 11.37, 11.38, 11.39, 11.40, 11.41 and 11.42; respectivamente. Los resultados obtenidos ilustran que COGIR mantiene estable su rendimiento con respecto a los demás motores de búsqueda en el tratamiento de tópicos con mayor dificultad. En este sentido, el uso de PJREL's tiende a favorecer los demás sistemas ya que todos ellos comparten el mismo modelo teórico, lo que provoca listas con resultados similares. Esto repercute en su beneficio, dado que los PJREL's se calculan a partir de dichas listas.

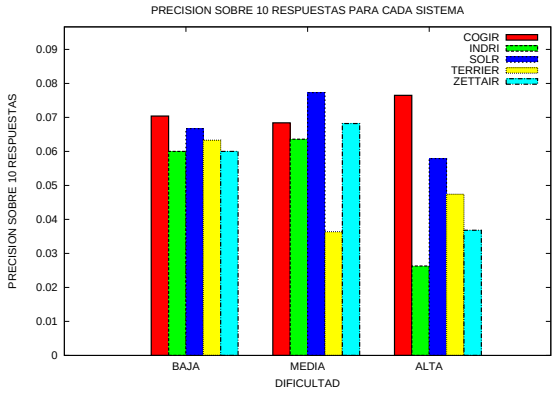


Figura 11.33: P@10 sobre CTHPJ usando PJREL'S

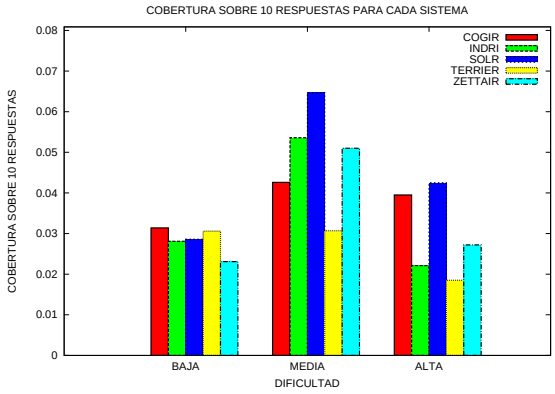


Figura 11.34: C@10 sobre CTHPJ usando PJREL'S

11.2.2 | Usando una colección de conjuntos de tópicos basada en la valoración tipo máquina

Ahora calculamos el mismo conjunto de medidas anteriores sobre el conjunto de tópicos CTMPJ. En este caso, el valor del experimento consiste en corroborar las conclusiones alcanzadas con anterioridad.

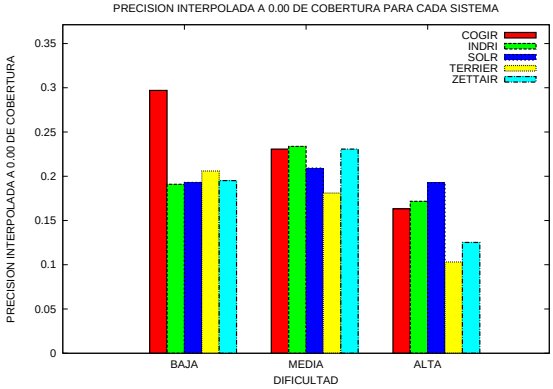


Figura 11.35: $\text{PI}_{C=0'00}$ sobre CTHPJ usando PJREL's

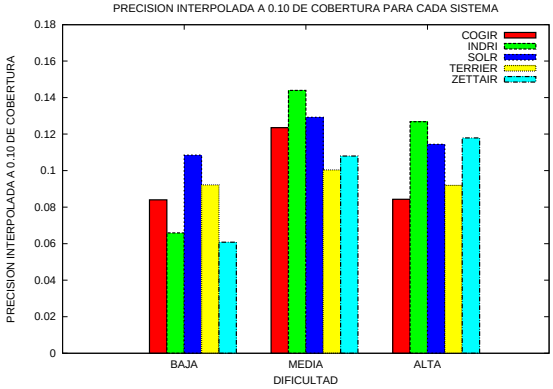


Figura 11.36: $\text{PI}_{C=0'10}$ sobre CTHPJ usando PJREL's

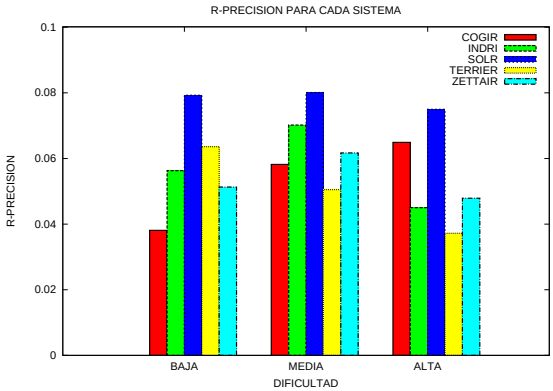


Figura 11.37: R-P sobre CTHPJ usando PJREL's

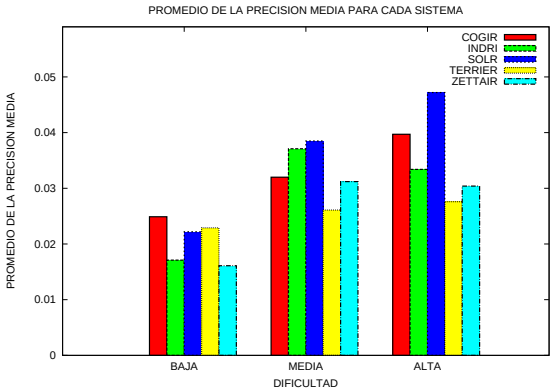


Figura 11.38: PPM sobre CTHPJ usando PJREL's

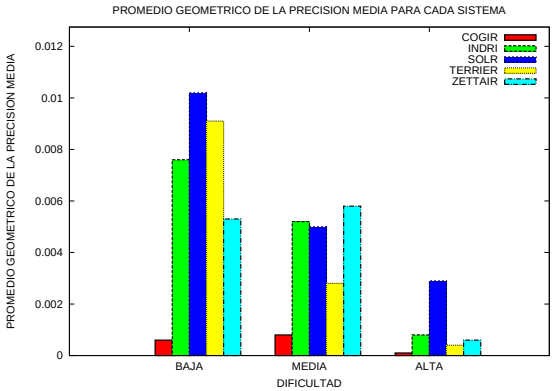


Figura 11.39: PGPM sobre CTHPJ usando PJREL's

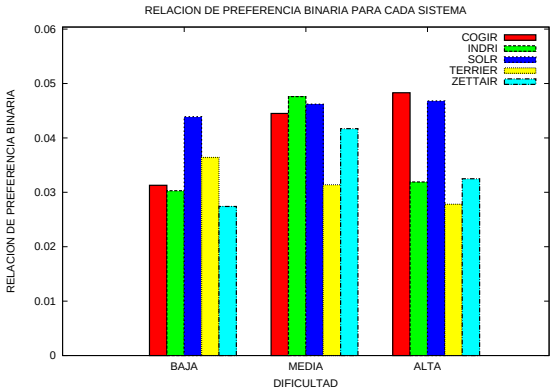


Figura 11.40: PREFB sobre CTHPJ usando PJREL's

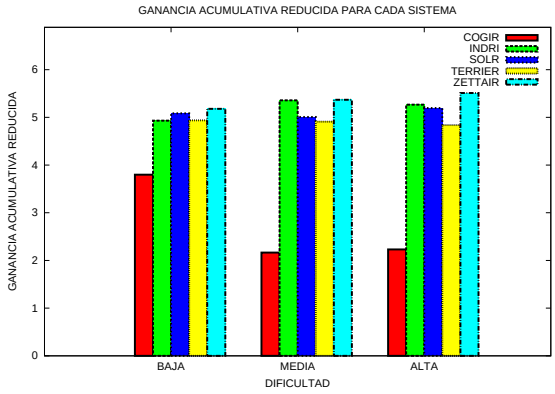


Figura 11.41: GAAR sobre CTHPJ usando PJREL'S

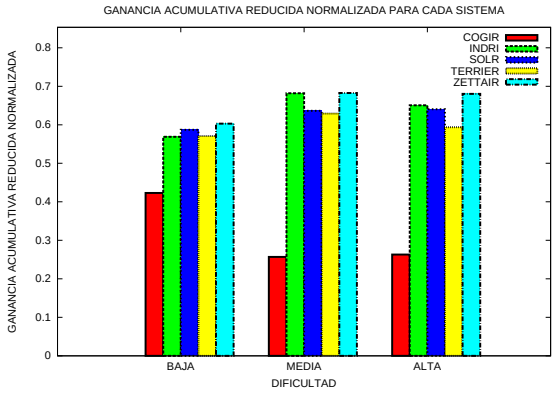


Figura 11.42:GAARN sobre CTHPJ usando PJREL'S

11.2.2.1 | Medidas de evaluación basadas en conjuntos

Los resultados obtenidos para las medidas P, C, F y FR se muestran en las gráficas de las Figs. 11.43, 11.44, 11.45 y 11.46 respectivamente. Proporcionan valores que claramente favorecen a los demás sistemas con respecto a COGIR sobre los tópicos de dificultad baja y media. Sin embargo, en el caso de los tópicos con mayor poder de discriminación, nuestra propuesta consigue mantener su posición con respecto a los ya comentados para el conjunto de tópicos CTHPJ.

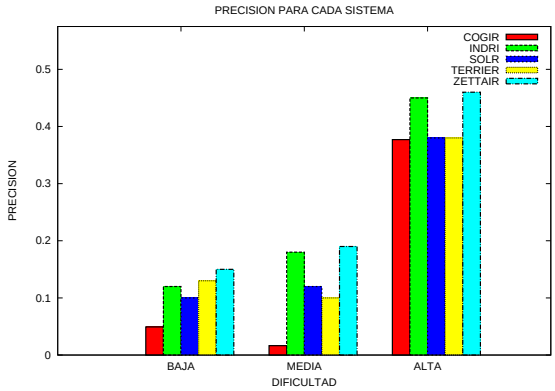


Figura 11.43: P sobre CTMPJ usando PJREL's

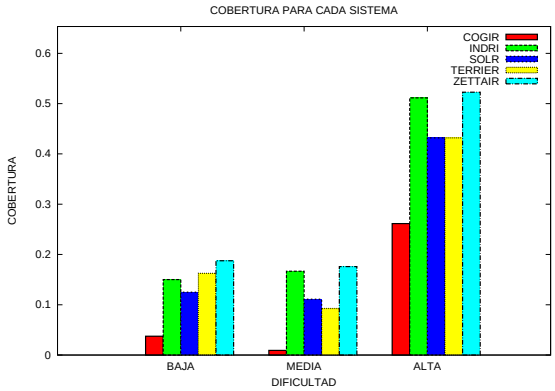


Figura 11.44: C sobre CTMPJ usando PJREL's

11.2.2.2 | Medidas de evaluación basadas en ordenación

Calculamos la P@10, C@10, PI_C para niveles de cobertura 0 y 0'10, R-P, PPM, PGPM, PREFB, GAAR y GAARN en las Figs. 11.47, 11.48, 11.49, 11.50, 11.51, 11.52, 11.53, 11.54, 11.55 y 11.56; respectivamente. Las pruebas sugieren que los resultados obtenidos sobre los tópicos de dificultad baja y media son peores en el caso del motor COGIR. Los obtenidos en el intervalo superior de dificultad se mantienen más o menos en la misma línea que para el conjunto CTHPJ, aquí también penalizado por el uso de PJREL. Al igual que para aquel conjunto de tópicos, el enfoque conceptual no supera a sus competidores.

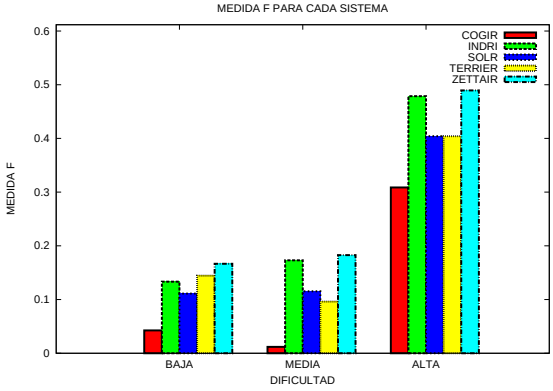


Figura 11.45: F sobre CTMPJ usando PJREL's

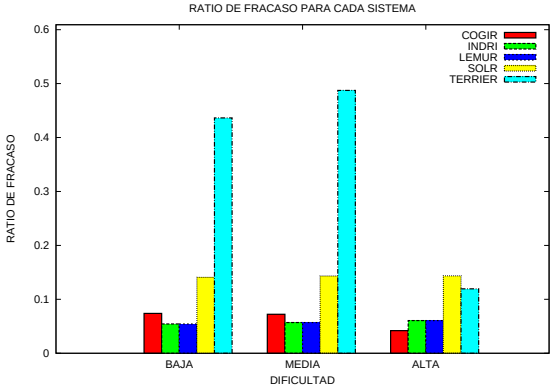


Figura 11.46: FR sobre CTMPJ usando PJREL's

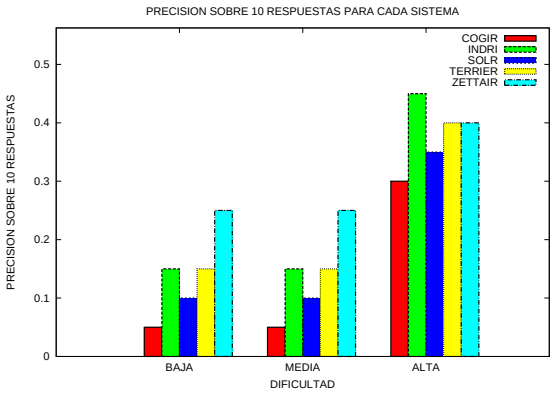


Figura 11.47: P@10 sobre CTMPJ usando PJREL's

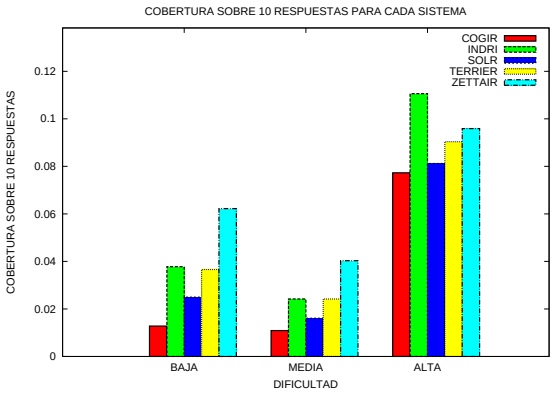


Figura 11.48: C@10 sobre CTMPJ usando PJREL's

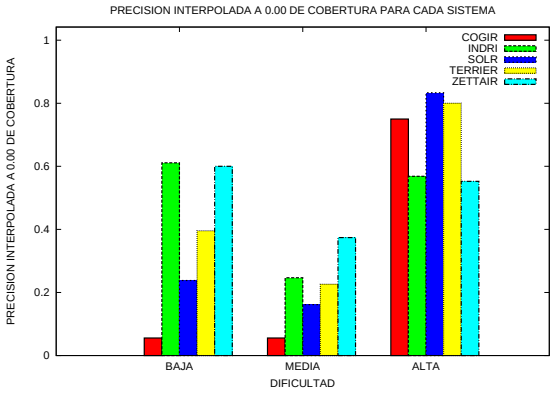


Figura 11.49: $PI_{C=0'00}$ sobre CTMPJ usando PJREL's

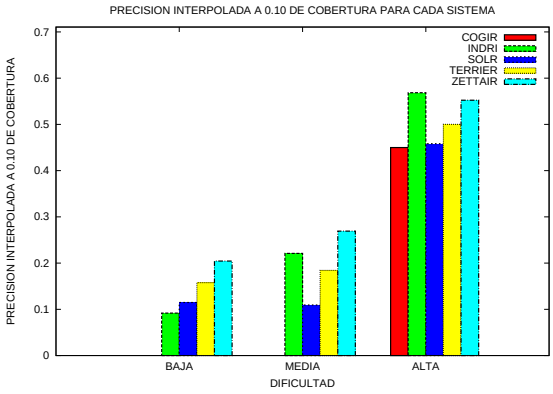


Figura 11.50: $PI_{C=0'10}$ sobre CTMPJ usando PJREL's

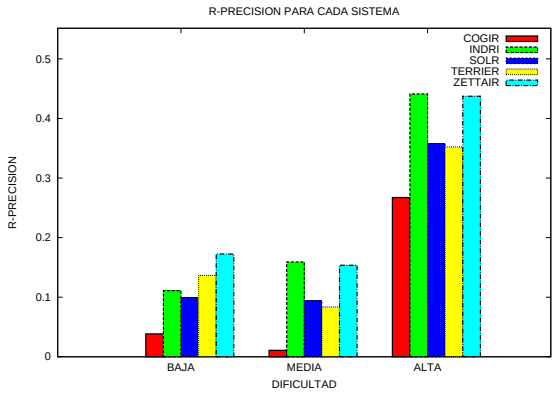


Figura 11.51: R-P CTMPJ usando PJREL's

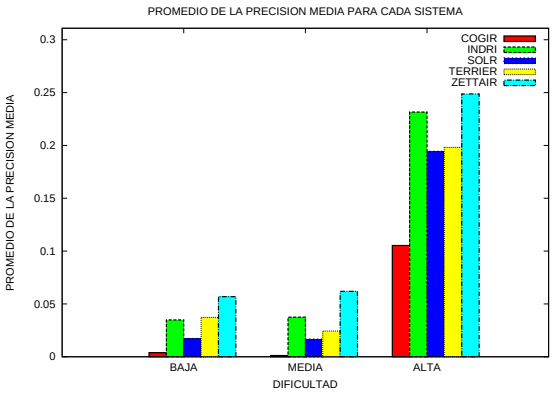


Figura 11.52: PPM CTMPJ usando PJREL's

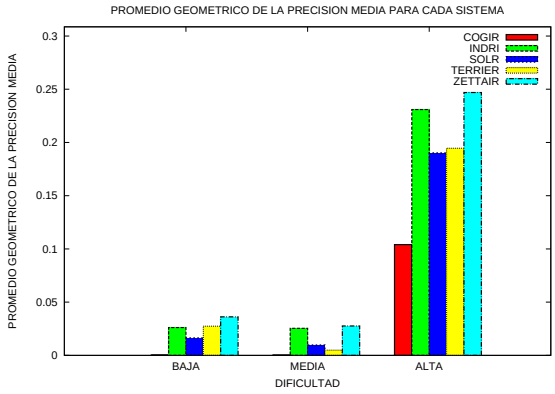


Figura 11.53: PGPM CTMPJ usando PJREL's

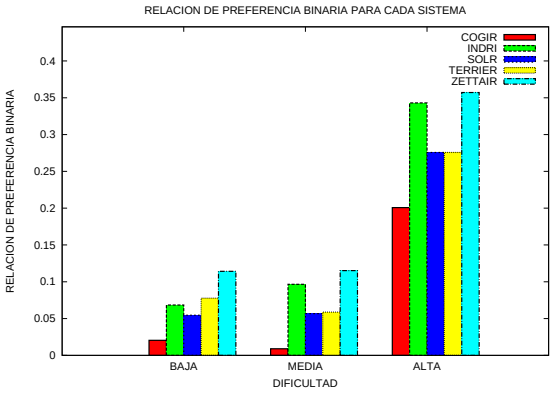


Figura 11.54: PREFB CTMPJ usando PJREL's

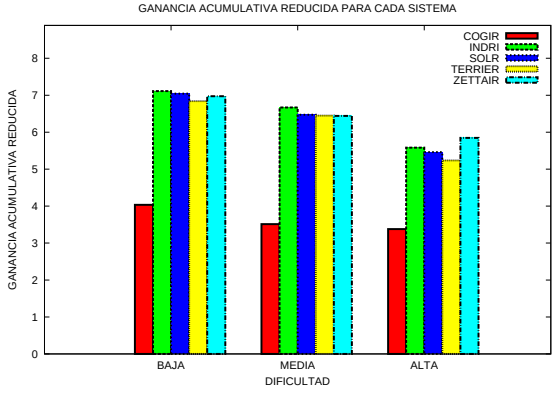


Figura 11.55: GAAR CTMPJ usando PJREL's

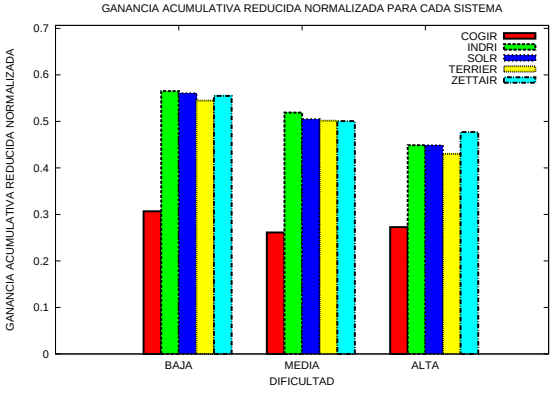


Figura 11.56: GAARN CTMPJ usando PJREL's

11.3 | Sistemas de RI con ordenación usando valoración tipo máquina

Como ya se ha dicho, el punto de partida de esta técnica de ordenación [208] es la medida PM, lo que implica que es necesario un cierto número de juicios de relevancia para iniciar el proceso. Teniendo en cuenta que previamente los hemos introducido como estrategias de enjuiciamiento, experimentamos en este nivel tanto con JREL's como con PJREL's.

11.3.1 | Calculando la PM a partir de JREL's

Como ya se había hecho para la clasificación basada en JREL's, en este punto podemos diferenciar dos series de tests, uno por cada conjunto de tópicos construido a partir de JREL's: CTHJ y CTMJ.

11.3.1.1 | Usando una colección de conjuntos de tópicos basada en la valoración tipo humano

En este punto, vamos a probar una ordenación usando una valoración tipo máquina sobre la colección de tópicos tipo humano CTHJ. Los resultados para la medida A se muestran en la Fig. 11.57, dando nuevamente una ventaja al motor de búsqueda conceptual sobre el resto, en especial en el caso de los tópicos con menor y mayor poder de discriminación. De hecho, aunque los peores resultados de COGIR se refieren a los tópicos de dificultad media, aún en este caso su rendimiento mejora el mostrado por cualquiera de los demás sistemas que, en general, muestran un mejor comportamiento justamente sobre ese conjunto de tópicos.

11.3.1.2 | Usando una colección de conjuntos de tópicos basada en la valoración tipo máquina

Probamos ahora una ordenación basada en la valoración tipo máquina sobre la colección de tópicos tipo máquina CTMJ. Los resultados para la medida A se muestran en la Fig. 11.58. Los resultados corroboran el comportamiento previamente observado sobre la colección de tópicos CTHJ.

11.3.2 | Calculando la PM a partir de PJREL's

Siguiendo el mismo protocolo descrito para la PM calculada a partir de JREL's, aquí consideramos dos series de pruebas, uno por cada conjunto de tópicos construido a partir de PJREL's: CTHPJ y CTMPJ.

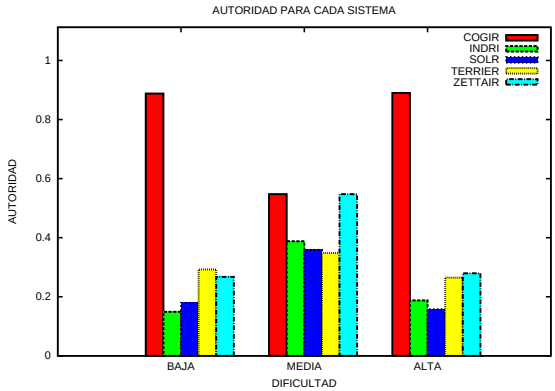


Figura 11.57: A sobre CTHJ usando JREL's

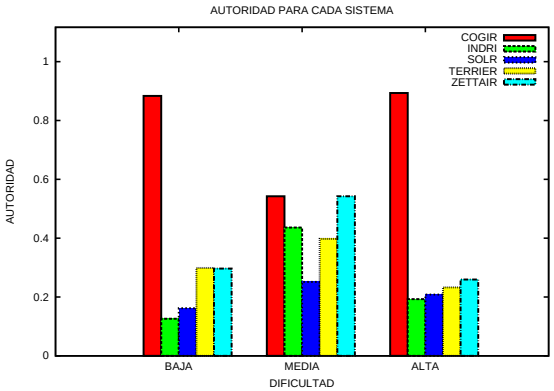


Figura 11.58: A sobre CTMJ usando JREL's

11.3.2.1 | Usando una colección de conjuntos de tópicos basada en la valoración tipo humano

Probamos ahora una ordenación basada en la valoración tipo máquina usando una colección de conjuntos de tópicos basada en la valoración tipo humano (CTHPJ). Los resultados para la medida A se muestran en la Fig. 11.59. Desde un punto de vista cualitativo, el rendimiento observable en relación a COGIR es análogo al previamente descrito en el caso en el que PM se calculaba a partir de JREL's.

11.3.2.2 | Usando una colección de conjuntos de tópicos basada en la valoración tipo máquina

El turno corresponde ahora a la ordenación basada en la valoración tipo máquina usando una colección de conjuntos de tópicos basada en la valoración tipo máquina (CTMPJ). Los resultados para la medida A se muestran en la Fig. 11.60. Aunque el mejor funcionamiento continúa correspondiendo a COGIR, contrariamente a las anteriores gráficas para el caso de la A, en este caso los peores resultados para el modelo conceptual se obtienen en el conjunto de tópicos de mayor dificultad.

11.4 | Sistemas de RI con ordenación usando la media de contadores de referencia ponderados

La última propuesta de ordenación que consideramos fue descrita por Wu *et al.* en [347] y se basa en el concepto de la media de contadores de referencia ponderados. Como ya hemos introducido, se pueden considerar aquí cuatro medidas: $MCRP_{O}$, $MCRP_{P}$, $MCRP_{OL}$ y $MCRP_{PL}$.

Dado que en este caso, la estrategia de ordenación no está relacionada con ninguna estrategia de enjuiciamiento en particular, vamos a considerar el conjunto completo de

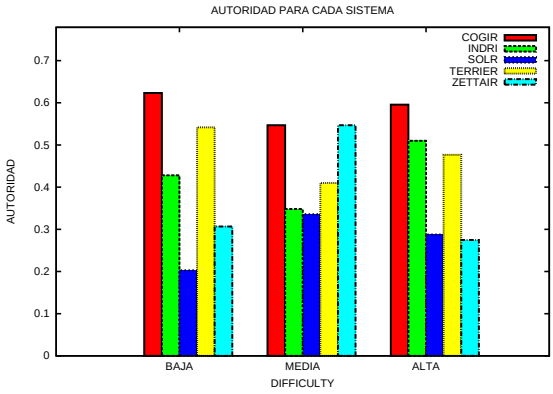


Figura 11.59: A sobre CTHPJ usando PJREL's

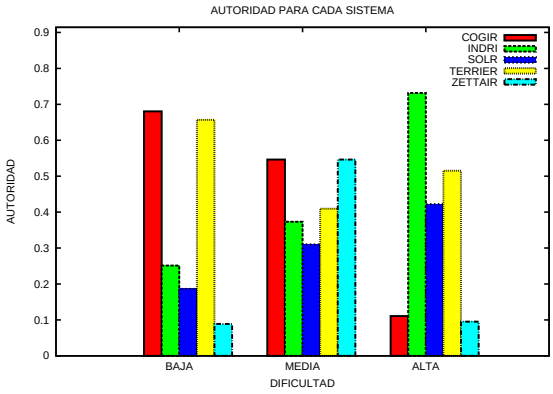


Figura 11.60: A sobre CTMPJ usando PJREL's

conjuntos de tópicos previamente introducidos con el fin de asegurar un procedimiento completo de prueba: CTHJ, CTMJ, CTHPJ y CTMPJ. Esto nos va a permitir considerar tanto la valoración de tipo humano como la de tipo máquina para seleccionar los tópicos, además de las técnicas basadas en JREL's y en PJREL's con el fin de reducir el tamaño de los conjuntos de tópicos. De esta manera, no vamos a favorecer a ninguna estrategia que pudiera ser usada para afinar algunos de los sistemas de RI que se están comparando, un aspecto importante a tener en cuenta cuando se considera un método de ordenación, cuyo punto de partida es el recuento de referencias cruzadas entre el conjunto de documentos devueltos por los motores de búsqueda.

11.4.1 | Usando la reducción de tópicos basados en JREL's

Experimentaremos primero con conjuntos de tópicos obtenidos a partir de técnicas de reducción de tópicos basados en JREL's, que incluyen tanto a las colecciones de conjuntos de tópicos de tipo humano como máquina.

11.4.1.1 | Usando una colección de conjuntos de tópicos basada en la valoración tipo humano

En este caso, los resultados se muestran para las métricas MCRP_{O}, MCRP_{P}, MCRP_{OL} y MCRP_{PL} sobre el conjunto de tópicos CTHJ en las Figs. 11.61, 11.62, 11.63 y 11.64, respectivamente. En estos casos, el enfoque conceptual aparentemente muestra el peor comportamiento posible, especialmente cuando se trata de tópicos de dificultad alta, si bien los resultados son un poco mejores para las medidas MCRP_{O} y MCRP_{OL}. Contrariamente a lo que uno pudiera pensar, tal comportamiento es no sólo congruente con las anteriores medidas sino perfectamente previsible.

Al aplicar técnicas relativistas, el sistema de RI objeto de test no podría en ningún caso mejorar las prestaciones del conjunto de los que le sirven de referencia comparativa. Es más, este tipo de metodologías puede llevar a situaciones estrepitósamente erróneas

cuando el conjunto de esos sistemas referentes muestra un rendimiento común pobre sobre un conjunto de tópicos, mientras que el sistema testeado ofrece una buena precisión. Es justamente el comportamiento que podemos observar en este caso sobre el conjunto de tópicos de mayor dificultad, que hemos visto favorecía al acercamiento conceptual en todas las métricas anteriores y que ahora, por el contrario, parecería mostrar un peor comportamiento.

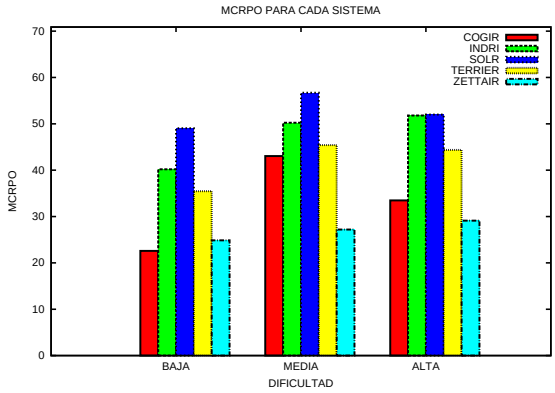


Figura 11.61: MCRP_{O} sobre CTHJ

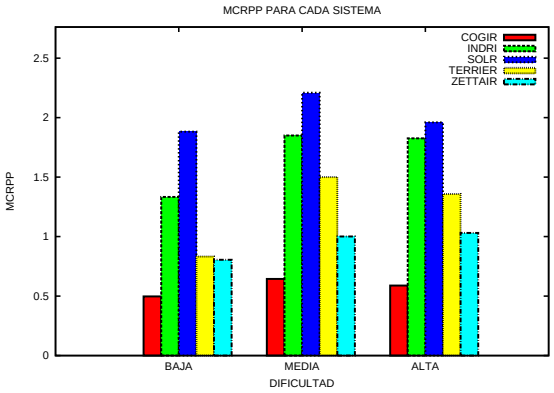


Figura 11.62: MCRP_{P} sobre CTHJ

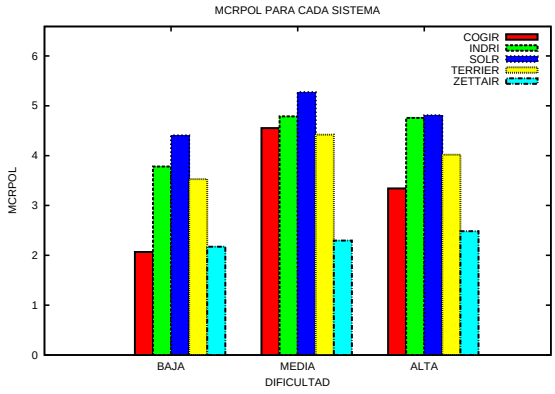


Figura 11.63: MCRP_{OL} sobre CTHJ

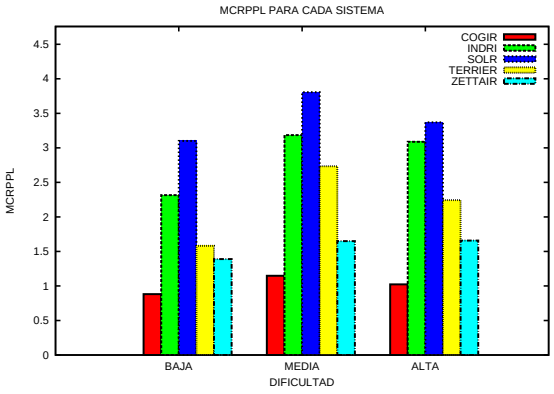


Figura 11.64: MCRP_{PL} sobre CTHJ

11.4.1.2 | Usando una colección de conjuntos de tópicos basada en la valoración tipo máquina

Los resultados se muestran ahora para las medidas MCRP_{O}, MCRP_{P}, MCRP_{OL} y MCRP_{PL} sobre el conjunto de tópicos CTMJ, en las Figs. 11.65, 11.66, 11.67 y 11.68 respectivamente. Podemos hacer extensivos exactamente los mismos comentarios previamente realizados con las pruebas sobre el conjunto de tópicos CTHJ, corroborando el razonamiento realizado más allá el tipo de valoración aplicado en la selección de tópicos.

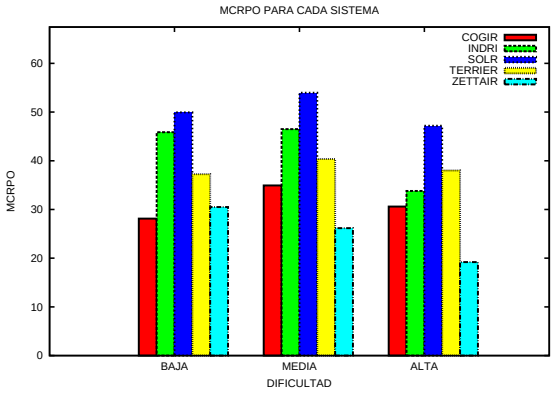


Figura 11.65: $MCRP_O$ sobre CTMJ

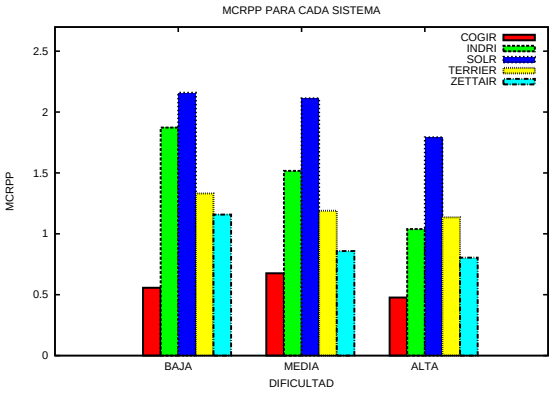


Figura 11.66: $MCRP_P$ sobre CTMJ

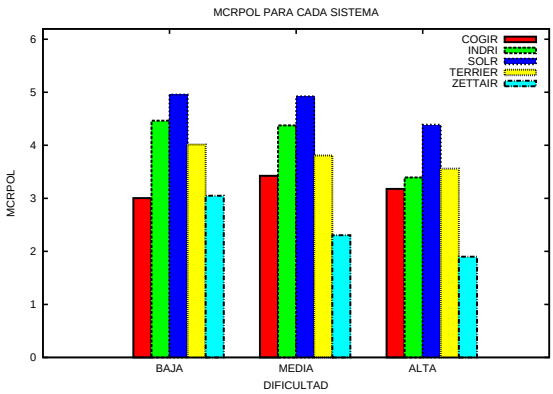


Figura 11.67: $MCRP_{OL}$ sobre CTMJ

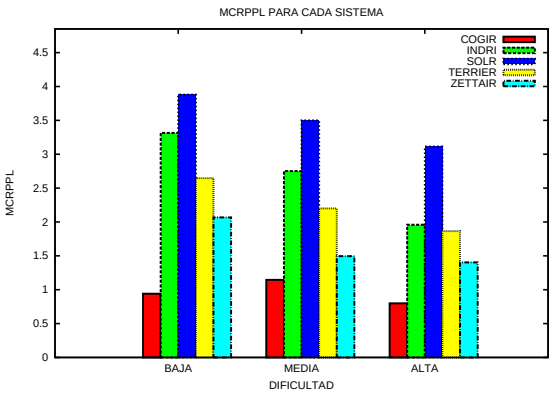


Figura 11.68: $MCRP_{PL}$ sobre CTMJ

11.4.2 | Usando la reducción de tópicos basados en PJREL's

Los experimentos están relacionados ahora con los conjuntos de tópicos obtenidos a partir de PJREL's basados en métodos de reducción, incluyendo colecciones de conjuntos de tópicos tanto de tipo humano como máquina.

11.4.2.1 | Usando una colección de conjuntos de tópicos basada en la valoración tipo humano

Como para el caso anterior de los JREL's, los resultados se muestran para las medidas $MCRP_O$, $MCRP_P$, $MCRP_{OL}$ y $MCRP_{PL}$ sobre el conjunto de tópicos CTHPJ, en las Figs. 11.69, 11.70, 11.71 y 11.72 respectivamente. Los resultados mostrados en las gráficas son cualitativamente equivalentes a los previamente comentados para la reducción de tópicos basados en JREL's, aunque existe una diferencia sustancial. Esto es, el modelo conceptual obtiene los mejores resultados para el conjunto de los tópicos difíciles, cuando para los casos anteriores conseguía los peores.

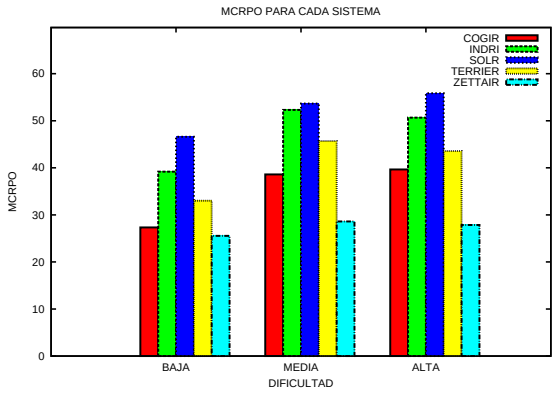


Figura 11.69: $MCRP_O$ sobre CTHPJ

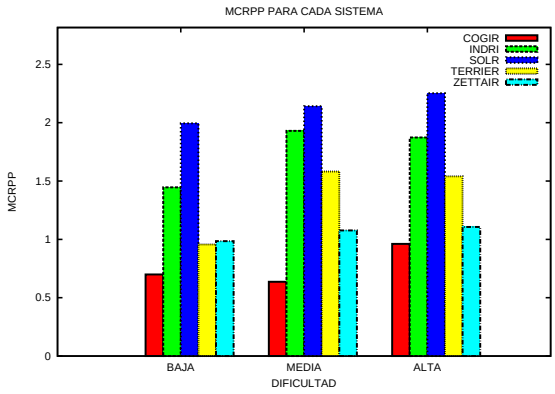


Figura 11.70: $MCRP_P$ sobre CTHPJ

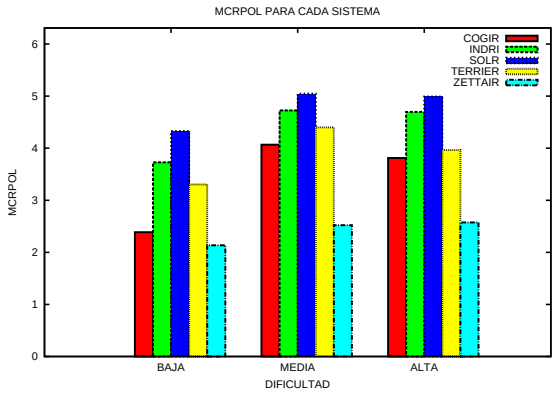


Figura 11.71: $MCRP_{OL}$ sobre CTHPJ

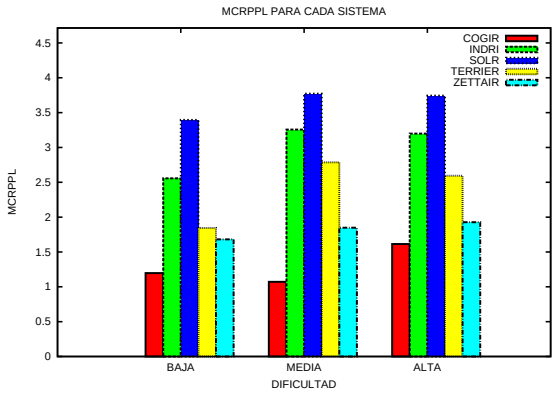


Figura 11.72: $MCRP_{PL}$ sobre CTHPJ

11.4.2.2 | Usando una colección de conjuntos de tópicos basada en la valoración tipo máquina

Los valores se muestran ahora para las medidas MCRP_{O}, MCRP_{P}, MCRP_{OL} y MCRP_{PL} sobre el conjunto de tópicos CTMPJ, en las Figs. 11.73, 11.74, 11.75 y 11.76, respectivamente. Los resultados experimentales son aquí cuantitativamente equivalentes a los comentados anteriormente, aunque sensiblemente diferentes desde un punto de vista cualitativo. En particular, al contrario de las pruebas anteriores, se obtienen los peores resultados para el enfoque conceptual en el caso de las medidas MCRP_{O} y MCRP_{OL}, considerando el conjunto de tópicos de dificultad baja. En relación con las métricas MCRP_{P} y MCRP_{PL}, los resultados son equivalentes a los obtenidos para el caso de la colección de conjuntos de tópicos basada en la valoración humana.

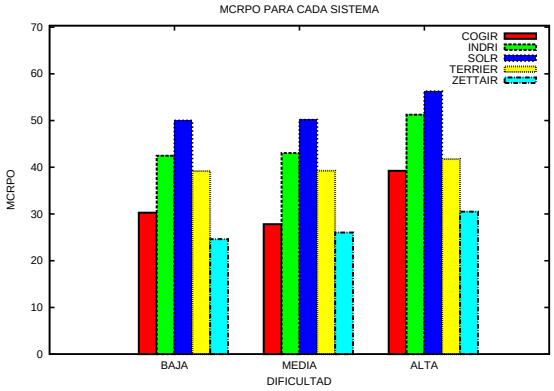


Figura 11.73: MCRP_{O} sobre CTMPJ

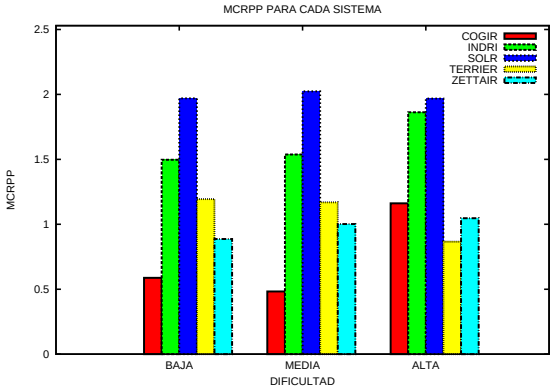


Figura 11.74: MCRP_{P} sobre CTMPJ

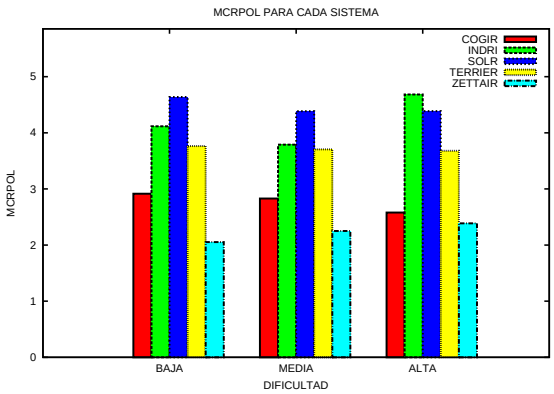


Figura 11.75: MCRP_{OL} sobre CTMPJ

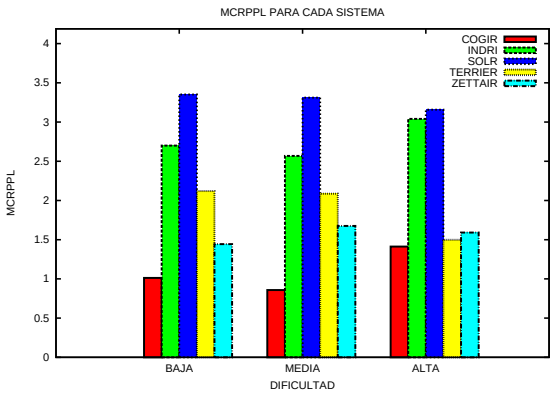


Figura 11.76: MCRP_{PL} sobre CTMPJ

CAPÍTULO XII

Conclusión

La conveniencia de la inclusión o no de conocimiento lingüístico específico en el diseño de motores de búsqueda, es una discusión que se remonta a los orígenes del propio ámbito de la RI. Habitualmente tres han sido las razones argumentadas para obviar esta cuestión: la complejidad algorítmica asociada, la escasez o incluso carencia de recursos lógicos y el aparentemente escaso rendimiento extra asociados a su consideración. Asumida la complejidad técnica de este tipo de estrategias, introducimos una metodología para la adquisición automática de la semántica del texto a partir de la información léxica y sintáctica resumidas en un grafo conceptual que no es sino el reflejo del conjunto de relaciones de dependencia previamente reconocidas. Ello nos permite no sólo disponer de una estructura formal que traslada fielmente el significado de un documento cualquiera, sino que también proporciona una base estructural idónea sobre la que sustentar un algoritmo de correspondencia de patrones aproximado capaz de estimar la proximidad semántica entre dos textos diferentes.

Pretendemos, además, arrojar alguna luz práctica en relación a lo que intuitivamente parece obvio, que una base semántica mejorada en el proceso de recuperación debería tener su reflejo en el rendimiento observado. Para ello, hemos definido un completo entorno de evaluación formal siguiendo lo que, a nuestro conocimiento, constituye una completa muestra de las técnicas actualmente disponibles. Ello nos ha permitido exprimir en profundidad las posibilidades de los acercamientos de RI conceptual, frente a la vocación más genérica de los motores de búsqueda clásicos.

Los resultados obtenidos parecen zanjar definitivamente la discusión por cuanto muestran un rendimiento que, en el peor de los casos, iguala al de los entornos basados en conjuntos de palabras independientemente de cual sea la base de implementación. Como única excepción observada, señalar los resultados de los tests basados en el uso de PJREL's, que favorecen naturalmente a las arquitecturas asociadas a los sistemas de RI que han sido utilizados como referencia para la generación de tales estructuras.

Además, se observa sistemáticamente un salto cualitativo importante cuando se trata de resolver consultas catalogadas como de dificultad creciente en su respuesta y que nosotros asociamos con tópicos con mayor poder de discriminación entre los sistemas comparados.

Intuitivamente este resultado coincide con lo esperable, puesto que la información semántica se revela determinante cuanto más complejo es el significado del texto a analizar, tanto en lo que se refiere a la colección documental a explorar como a la del tópico. Por el contrario, cuando la simplicidad de la interrogación o del propio contenido de los textos estudiados permite prescindir de relaciones semánticas complejas, todos los sistemas objeto de estudio presentan un rendimiento equiparable independientemente del tipo de arquitectura de indexación considerada.

PARTE V

Apéndices

APÉNDICE A

El recurso lingüístico: la «*Flore du Cameroun*»

Para describir el recurso lingüístico empleado: la «*Flore du Cameroun*», un conjunto de volúmenes de botánica, es necesario comenzar introduciendo la estructura particular de identificación dispuesta en niveles de complejidad, llamada *taxonomía botánica,* a la vez que resulta imprescindible considerar uno de los pilares fundamentales para la clasificación de plantas, conocido por *nomenclatura*. De esta manera, una vez explicados ambos conceptos (taxonomía y nomenclatura), será más sencillo entrar en los detalles de composición del recurso empleado.

A.1 | Taxonomías botánicas

En la tierra se conocen más de un millón de especies de animales y se superan las 300.000 de plantas, y los biólogos creen que pueda haber aún varios millones de especies diferentes. Para poner orden en este extenso conjunto de formas de vida, se han desarrollado estrategias para su clasificación denominadas *taxonomías*. Éstas no se limitan sólo a identificar y dar nombres a los organismos, sino que tratan de entender las relaciones existentes entre ellos. Un buen sistema de clasificación permite a los biólogos conocer con mayor detalle las características de un ser vivo en función del grupo al que pertenece.

El primer esfuerzo real para desarrollar un sistema taxonómico proviene de los antiguos griegos. Los filósofos Alcmeón de Cretona y Empédocles de Akragas fueron los pioneros. Luego les sucedió Aristóteles (384-322 a.C.), que intentó dividir a los organismos en dos grupos: animal y vegetal, introduciendo el término *especie* para referirse a «formas similares de vida», organizándolos en ocho grupos o categorías. Además, estructuró en una escala jerárquica 500 especies de animales, con el ser humano en la posición más alta, y continuando con cuadrúpedos, aves, serpientes, peces, insectos, moluscos y mohos. Hoy el término *especie* se interpreta como «un grupo de organismos

de una clase en particular». Las primeras obras botánicas importantes que se conservan se deben a Teofasto (372-287 a.C.), primer botánico que hizo una clasificación y dividió las plantas en base a la naturaleza de sus cotiledones (por un lado, las *monocotiledóneas*[1] y por otro las *dicotiledóneas*[2]). Más tarde, en el siglo I d.C., Dioscórides ordenó las especies botánicas en tres grupos, atendiendo a su utilidad: comestibles, medicinales y venenosas.

La Edad Media fue una época en la que se le dio más importancia a la descripción de las especies que a su propia ordenación. A partir de los siglos XVI y XVII, surgieron autores preocupados por recuperar clasificaciones antiguas. Los botánicos de entonces basaron sus estudios en ciertos rasgos morfológicos como el número de piezas florales y su disposición. En el siglo XVII, John Ray desarrolló un sistema de clasificación mejorado para organizar las plantas de semilla de acuerdo con su estructura, todavía vigente hoy en día. Ray diseñó una metodología que asociaba un nombre en latín a cada organismo, que consistía en una larga descripción científica del mismo.

Basándose en esas ideas, en 1735, el naturalista más importante de esta época Carl von Linné, conocido como Carolus Linnaeus, ideó un método completo y sistemático para asignar cada organismo a diferentes niveles jerárquicos, llamados *reinos*[3]. Concretamente, estableció tres: *Vegetabilia*, *Animalia* y un grupo *Mineralia* que pronto fue abandonado. Atendiendo a esta estructura, subdividió progresivamente cada uno en subcategorías, en base a sus características físicas compartidas. A Linneo se le considera el fundador de la taxonomía moderna, conocida como *Taxonomía de Linneo* [185, 186], y su método sigue vigente hoy en día aunque con variaciones.

Desde entonces, se han movido algunas formas de vida de un reino a otro. Pero hubo que esperar a que se descubrieran los microorganismos para que produjera una reorganización al distinguir entre seres unicelulares y pluricelulares, y dentro de éstos diferenciar los hongos de las plantas. Se conformaron así cuatro reinos: *Animalia*, *Plantae*, *Fungi* y *Protoctista*. Más tarde, tras el uso del microscopio electrónico, Whittaker[4] propuso una organización separando en dos el reino protoctista: el de *Monera* y el *Protoctista*. Esta propuesta permaneció vigente y arraigada mucho tiempo, pero hoy en día la forma de entender las relaciones entre los seres vivos ha cambiado. Linneo sólo pudo basarse en su clasificación a partir de estructuras externas, y se reflejaban las relaciones entre organismos según parecidos anatómicos. Cuando se acogió el concepto de evolución como mecanismo de diversidad biológica y formación de especies, se produjo una expansión en el número de niveles jerárquicos.

Actualmente, se consideran cinco reinos en tres *dominios*. Éstos son una jerarquía suprareinal, dada la necesidad de dividir los organismos teniendo en cuenta las grandes diferencias que presentan a nivel molecular. Así, la división se establece en base a aquellos

[1] con un cotiledón.

[2] con dos cotiledones.

[3] cada una de las grandes subdivisiones en que se consideran distribuidos los seres vivos, por razón de sus características comunes.

[4] un ecólogo vegetal, algólogo, botánico estadounidense, activo entre 1950 y 1980.

organismos que están compuestos por células procariotas[5] y eucariotas[6], dando lugar en el primer caso a dos dominios llamados *Archaea*, *Bacteria*, que incluye al reino antiguamente llamado Monera, y a uno denominado *Eukarya*. Dentro de éste último, se pueden distinguir los cuatro reinos *animalia*, *plantae*, *fungi* (hongos) y *protoctista* (comprende una colección de organismos, en su mayoría unicelulares, antes clasificados como «protozoos», «algas»), como se puede ver en la Fig. A.1.

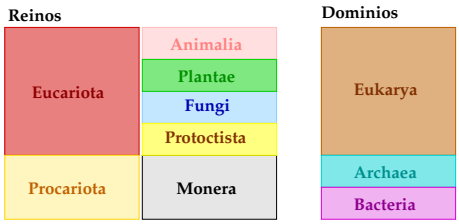


Figura A.1: División en reinos y dominios

Así mismo, existe una estructura de rangos para ellos, tal y como se muestra en la Tabla A.1, donde cada uno posee un ejemplo para cada subrango, en función de su posición en el listado. En la tabla se puede ver como los reinos se dividen en *filos* o *phylum*[7] para los animales, y en *divisiones*[8] para las plantas y otros organismos. A su vez, los filos o divisiones se dividen en *clases*[9], las clases en *órdenes*, los órdenes en *familias*, las familias en *géneros*[10] y los géneros en *especies*[11]. Entre las subdivisiones posteriores, han surgido entidades como *superclases*, *superórdenes*, *subórdenes* e *infraórdenes*, *superfamilias* y *subfamilias*, *tribus* y *subtribus*.

Si extendemos la taxonomía, se obtienen los rangos que aparecen en la Tabla A.2, donde una *tribu* es una categoría optativa empleada para organizar las familias que contienen muchos géneros. Una *especie* es una población o un conjunto de poblaciones donde los individuos pueden reproducirse entre ellos y engendrar una descendencia viable y fecunda en condiciones naturales. Una *subespecie* consiste en un grupo de individuos que se encuentran aislados (por cuestiones geográficas, ecológicas, etc) y que evolucionan de otro modo con respecto a la especie de referencia. Una *variedad* permite delimitar y agrupar de un modo más especializado un conjunto de individuos que difieren ligeramente unos de otros, en base a unos rasgos considerados como menores, como por ejemplo diferencias morfológicas (anatómicas), químicas u organolépticas (color,

[5] aquellas células que no tienen núcleo celular diferenciado, es decir, cuyo ADN se encuentra disperso en el citoplasma.

[6] aquellas células que tienen núcleo diferenciado dentro del citoplasma.

[7] agrupación de animales basada en su plan general de organización, referida básicamente a la disposición interna de sus tejidos, órganos y sistemas; a su simetría y el número de segmentos corporales y de extremidades que posee.

[8] agrupación de las plantas que se establecen siguiendo el orden de evolución.

[9] grupo taxonómico que comprende varios órdenes de plantas o de animales con características comunes.

[10] un grupo que reúne a varias especies emparentadas.

[11] la limitación de lo genérico en un ámbito morfológicamente concreto.

Rango	**Reino Animalia**	**Reino Plantae**
phylum/ división	Chordata, Mollusca, Echinodermata, Arthropoda, Nematoda, Cnidaria, Annelida, Porifera, ...	Magnoliophyta, Psilophyta Cycadophyta, Pinophyta, Gnetophyta, Bryophyta , ...
clase	Mammalia, Cephalopoda, Asteroidea, Remipedia, Enoplea, Scyphozoa, Clitellata, Demospongiae, ...	Magnoliopsida, Psilopsida, Cycadopsida, Pinopsida, Gnetopsida, Bryopsida, ...
orden	Primates, Sepiida, Forcipulatida, Nectiopoda, Enoplida, Semaeostomeae Arhynchobdellidae, Poecilosclerida, ...	Fabales, Psilotales, Cycadales, Pinales, Gnetales, Grimmiales, ...
familia	Hominidae, Sepiidae, Asteriidae, Godzilliidae, Tripyloididae, Cyaneidae, Gnathobdellae, Cladorhizidae, ...	Caesalpiniaceae, Psilotaceae, Zamiaceae, Pinaceae, Gnetaceae, Grimmiaceae, ...
género	Homo, Sepia, Urasterias, Godzillus, Tripyloides, Euphorbia, Hirudo, Asbestopluma, ...	Caesalpinia, Psilotum, Zamia, Pinus, Gnetum, Racomitrium, ...
especie	H. sapiens, S. orbignyana, U. linkii, G. robustus, T. gracilis, E. albipollinifera, H. medicinalis, A. hipogea, ...	C. coriaria, P. nudum, Z. skinneri, P. patula, G. leyboldii, R. crispulum, ...

Tabla A.1: Estructura de rangos en dominio *Eukarya*, con ejemplos para *animalia* y *plantae*

olor), ecológicos (hábitat), pero que poseen todas las características diagnosticadas en la definición de especie. Un *cultivar* es una variedad natural, pero cultivada en jardines, es decir, es una variante que ha sido seleccionada.

Rango	**Rango superior**
familia	
subfamilia	familia
tribu	familia
género	familia
especie	género
subespecie	especie
variedad	especie o subespecie
cultivar	especie

Tabla A.2: Tabla de la estructura de los rangos de taxones

A partir de aquí, es necesario reglar los nombre de los taxones en base a unas reglas escritas de nomenclatura, de tal manera, que sea fácil la catalogación de un organismo vivo.

A.2 | Nomenclatura de taxones

En biología, la nomenclatura es la subdisciplina de la taxonomía que se ocupa de reglar los nombres de los niveles de clasificación, denominados *taxones*. Así, la nomenclatura actúa una vez que los expertos deciden qué taxones hay y a qué categorías pertenecen. Pero para nombrarlos deben atenerse a una serie de reglas escritas en los *Códigos Internacionales de Nomenclatura* [227, 228].

Así, en estos códigos, los nombres científicos de taxones que estén ubicados en categorías taxonómicas superiores al de especie son *uninominales*[12], diferenciándose sólo en el sufijo que da cuenta de su posición en la jerarquía. Además estos nombres se escriben siempre con mayúsculas. La Tabla A.3 muestra esa nomenclatura [120]:

Reino	**División**	**Clase**	**Orden**	**Familia**	**Subfamilia**	**Tribu**	**Género**
Plantae	-phyta	-opsida	-ales	-aceae	-oideae	-eae, -ae	-ium, -cola, -oides, -um, -os, -ina, -a, -ides, -ella, -aster, -ula, -ensis, -us, -opsis, -is…

Tabla A.3: Tabla de nomenclaturas de taxones

Por debajo de la categoría de género, todos los nombres de taxones son llamados «*combinaciones*». La mayoría reciben también una terminación latina más o menos codificada en función de la disciplina. Se distinguen varias tipos de combinaciones:

- En el nivel de especie, las combinaciones son *específicas* y *binomiales*. Esto quiere decir que los nombres están compuestos por dos palabras, dónde la primera es el nombre del género y, la segunda el nombre que caracteriza a la especie, llamado *epíteto específico*». Por convención se escribe el nombre de género en mayúscula, y el del epíteto específico en minúscula. Por ejemplo la especie: *Afzelia pachyloba*.

- Por debajo de especie, las combinaciones son *infraespecíficas* y *trinomiales*. Esto quiere decir que se añade un tercer nombre siempre con minúscula detrás de los que se refieren al género y especie.

- En el caso de la variedad, se identifica escribiendo a continuación del nombre de la especie o subespecie la abreviatura «var.» seguida del nombre de la variedad en sí. Por ejemplo la variedad: *Afzelia bella var. bella*.

[12] quiere decir que son nombres compuestos por una sola palabra.

Para ilustrar en cierta medida la nomencaltura de taxones, tomaremos como ejemplo la familia de las *Caesalpiniaceae*, ilustrando primero los rangos superiores, tal como se puede ver en la Tabla A.4.

```
Plantae -> reino
  Magnoliophyta -> división
    Magnoliopsida -> clase
      Fabales -> orden
        Caesalpiniaceae o Leguminosae-> familia
          Caesalpinioideae -> subfamilia
            ...
            Cynometreae -> tribu
              ...
              Afzelia -> género
                Afzelia pachyloba -> especie
                Afzelia africana -> especie
                Afzelia bipindensis -> especie
                Afzelia bella -> especie
                  Afzelia bella var. bella -> variedad
                  ...
```

Tabla A.4: Ejemplo de nomenclatura par las *Caesalpiniaceae*

También es frecuente utilizar en los nombres una serie de signos y abreviaturas entre las que caben destacar los siguientes:

- **sp. / spp.**: especie / especies.
- **subsp. / subspp.**: subespecie / subespecies.
- **var. / varr.**: variedad / variedades.
- $\times$: híbrido.
- **fl.**: del latín *floruit*, «floreció», se pone junto a la abreviatura de autor, seguido de uno o varios años e indica que sólo se le conoce esa época activa como botánico (ej. Andrews fl. 1975).
- **aff.**: abreviatura de *affinis*, «semejante», y se utiliza para indicar en un trabajo que los ejemplares estudiados tienen la mayoría de los rasgos de un taxón, pero difieren en otros (ej. Sempervivum aff. tectorum).

A.3 | El corpus: *La «Flore du Cameroun»*

En botánica, el concepto de *flora* se refiere a un conjunto de especies vegetales que se pueden encontrar en una región geográfica, que son propias de un período geológico o que habitan en un ecosistema determinado. Éste atiende al número de especies, mientras que

la noción de *vegetación* hace referencia a la distribución de las mismas y a la importancia relativa, por número de individuos y tamaño, de cada una.

A las colecciones de documentos que recopilan este tipo de información también se les conoce por *floras*. Éstas poseen descripciones de taxones, a menudo restringidas a aquellos rasgos observados en el propio terreno de una misma zona geográfica. El *corpus* botánico sobre el que introducimos nuestra propuesta es el trabajo «*Flore du Cameroun*», que describe parte de la flora del África Occidental. Sobre él ha sido necesario aplicar todo un proceso previo hasta conseguir su adquisición electrónica digital completa[13]. Concretamente, esta colección ha sido publicada entre 1963 y 2001, y es fruto del trabajo de varios autores. Está compuesta aproximadamente de 40 volúmenes escritos en francés, donde cada uno consta de unas 300 páginas. Los tomos que lo forman son los siguientes:

- Vol. 1 (1963): Rutaceae, Zygophyllaceae, Balanitaceae. Autor: R. Letouzey. Editor: Association de Botanique Tropicale.
- Vol. 2 (1964): Sapotaceae. Autor: A. Aubreville. Editor: Association de Botanique Tropicale.
- Vol. 3 (1964). Autor: M.L. Tardieu-Blot. Editor: Association de Botanique Tropicale.
- Vol. 4 (1965): Scitaminales: Musaceae, Strelitziaceae, Zingiberaceae, Cannaceae, Marantaceae. Autor: J. Koechlin. Editor: Association de Botanique Tropicale.
- Vol. 5 (1966): Thymeleaceae: Onagraceae, Halorrhagaceae. Autores: G. Aymonin y A. Raynal. Editor: Association de Botanique Tropicale.
- Vol. 6 (1967): Cucurbitaceae. Autor: M. Keraudren. Editor: Association de Botanique Tropicale.
- Vol. 7 (1968): Les Botanistes au Cameroun. Autor: R. Letouzey. Editor: Association de Botanique Tropicale.
- Vol. 8 (1968): Ulmaceae, Urticaceae. Autor: R. Letouzey. Editor: Association de Botanique Tropicale.
- Vol. 9 (1970): Legumineuses (Cesalpinioideae). Autor: A. Aubreville. Editor: Association de Botanique Tropicale.
- Vol. 10 (1970): Ombellae (Ombelliferae, Araliaceae). Autor: H. Jaques-Felix. Editor: Association de Botanique Tropicale.
- Vol. 11 (1970): Ebenaceae, Ericaceae. Autores: R. Letouzey y F. White. Editor: Association de Botanique Tropicale.

[13] éste proceso se describe en el Apéndice B, y su resultado es el punto de partida de la investigación desarrollada en el marco de esta tesis.

- Vol. 12 (1972): Loganiaceae. Autor: A.M.J. Leeuwenberg. Editor: Association de Botanique Tropicale.
- Vol. 13 (1972): Vitaceae, Leeaceae. Autor: B. Descoings. Editor: Association de Botanique Tropicale.
- Vol. 14 (1972): Malpighiaceae, Linaceae, Lepidobotryaceae, Ctenolophonaceae, Humiriaceae, Erythroxylaceae, Ixonanthaceae, Santalaceae. Autores: F. Badre y A. Lawalree. Editor: Association de Botanique Tropicale.
- Vol. 15 (1973): Icacinaceae, Olacaceae, Opiliaceae, Octoknemaceae y y Pentadiplandraceae. Autor: J.F. Villiers. Editor: Association de Botanique Tropicale.
- Vol. 16 (1973): Sapindaceae. Autores: R. Fouilloy y N. Halle. Editor: Association de Botanique Tropicale.
- Vol. 17 (1974): Amaranthaceae. Autor: A. Cavaco. Editor: Association de Botanique Tropicale.
- Vol. 18 (1974): Lauraceae, Myristicaceae, Monimiaceae. Autor: R. Fouilloy. Editor: Association de Botanique Tropicale.
- Vol. 19 (1975): Celastraceae, Aquilifoliaceae, Salvadoraceae, Pandaceae, Avicenniaceae, Bixaceae, Cannabaceae, Bombacaceae. Autor: J.F. Villiers. Editor: Association de Botanique Tropicale.
- Vol. 20 (1978): Chrysobalanceae, Scytopetalaceae, Rosaceae. Autores: R. Letouzey y F. Whire. Editor: Association de Botanique Tropicale.
- Vol. 21 (1980): Crucifères, Dipsaceae. Autores: B. Jonsell, H. Poppendieck y A. Lawalrée. Editor: Herbier National.
- Vol. 22 (1981): Balsaminaceae, Xyridaceae. Autores: C. Grey-Wilson y J. Lewis. Editor: Herbier National.
- Vol. 23 (1982): Loranthaceae. Autor: S. Balle. Editor: Herbier National.
- Vol. 24 (1983): Melastomataceae. Autor: H. Jacques Felix. Editor: Herbier National.
- Vol. 25 (1983): Combretaceae. Editor: Herbier National.
- Vol. 26 (1984): Alismataceae, Flagellariaceae. Autor:J.-J. Symoens y J.-F. Villiers. Editor: Herbier National.
- Vol. 27 (1984): Gesneriaceae, Bignoniaceae. Autores: B.L. Burtt y A.H. Gentry. Editor: Herbier National.

- Vol. 28 (1985): Moraceae (incl.Cecropiaceae). Autores: C.C. Berg, M.E.E. Hijman y J.C.A. Weerdenburg. Editor: Herbier National.
- Vol. 29 (1986): Capparidaceae. Autor: L.E. Kers. Editor: Herbier National.
- Vol. 30 (1987): Amaryllidaceae, Hypoxidaceae, Podostemaceae, Tristichaceae. Autores: I. Nordal, J.I. Iversen y C. Cusset. Editor: Herbier National.
- Vol. 31 (1988): Araceae. Autor: C. Ntépé-Nyame. Editor: Herbier National.
- Vol. 32 (1990): Célastraceae (Hippocrateoideae). Autor: N. Halle. Editor: Herbier National.
- Vol. 33 (1991): Rhamnaceae, Balanophoraceae, Diptérocarpaceae. Autores: M.C. Johnston, B. Hansen, J.F. Villiers y R. Letouzey. Editor: Herbier National.
- Vol. 34 (1998): Orchidaceae I. Autores: L. Szlachetko y S. Olszewski. Editor: Herbier National.
- Vol. 35 (2001): Orchidaceae II. Autores: L. Szlachetko y S. Olszewski. Editor: Herbier National.
- Vol. 36 (2001): Orchidaceae III. Autores: L. Szlachetko y S. Olszewski. Editor: Herbier National.
- Vol. 37 (2001): Dichapetalaceae. Autor: F.J. Breteler. Editor: Herbier National.

En estos volúmenes se ha aplicado una extensión del conocimiento en niveles jerárquicos, tal y como hemos mencionado en las Secciones A.1 y A.2, pero sin llegar a hacer una descripción tan precisa de todas las categorías descritas. De hecho, hay que destacar que los textos en cuestión obvian los primeros rangos que se muestran en la Tabla A.1, teniendo generalmente como punto de partida a las familias.

A partir de aquí, cada tomo se organiza como una secuencia de secciones, donde cada una está dedicada a un taxón, normalmente género, y sigue un esquema de estructura sistemático, tal y como se observa en la Fig. A.2. Así, dicha organización suele incluir pequeños apartados relacionados con la nomenclatura, la ecología, la distribución geográfica, además de un texto libre describiendo la morfología de la planta en cuestión, enumerando aspectos como el color, textura o forma. Pero, a su vez, también puede describir aquéllos de rango inferior al dado utilizando subsecciones.

Concretamente, y basándonos en el fragmento de la Fig. A.2, se pueden hacer ciertas divisiones en la estructura del documento, atendiendo al título, a las referencias, a la descripción propiamente dicha, y a la clave dicotómica; tal y como se muestra en la Fig. A.3. A continuación, vamos a explicar en detalle cada uno de estos apartados.

A. AUBRÉVILLE. — LÉGUMINEUSES - CÉSALPINIOIDÉES

18. AFZELIA Smith

Trans. Linn. Soc. **4**: 221 (1798), *nom. cons.*; OLIVER, FTA **2**: 301 (1871); LÉONARD, Reinwardtia **1** (1) : 61 (1950); FCB **3**: 350, *fig. 27* (1952); KEAY, Kew Bull. **9**: 266 (1954).

Base des stipules intrapétiolaire, persistante, épaisse. Feuilles à folioles opposées. Pétiolules tordus. Fleurs en grappes ou en panicules. Bractéoles concaves, enveloppant les très jeunes boutons, mais rapidement caduques (sauf *Afzelia bracteata* d'Afrique occidentale). Réceptacle long ou très long. Sépales 4, imbriqués. Pétale 1 grand, ± longuement onguiculé; les autres rudimentaires ou nuls. Étamines fertiles 7(-8), presque libres, à longs filets exserts. Staminodes souvent 2, très petits. Stipe de l'ovaire soudé à la paroi du réceptacle. Nombreux ovules.

Fruits épais, oblongs, s'ouvrant en 2 fortes valves ligneuses, lisses, bosselées, sans nervures saillantes, à face interne garnie d'un tissu spongieux dans lequel sont logées les graines. Graines épaisses, munies d'un arille coloré basilaire.

ESPÈCE-TYPE : *A. africana* Smith ex Pers.

Genre paléotropical, comptant une quinzaine d'espèces surtout africaines. Dans les domaines camerouno-gabonais et congolais il est représenté par 2 espèces de grands arbres, connues commercialement sous le nom de Doussié : *A. bipindensis* (Doussié rouge), *A. pachyloba* (Doussié blanc), absentes du domaine libéro-ivorien.

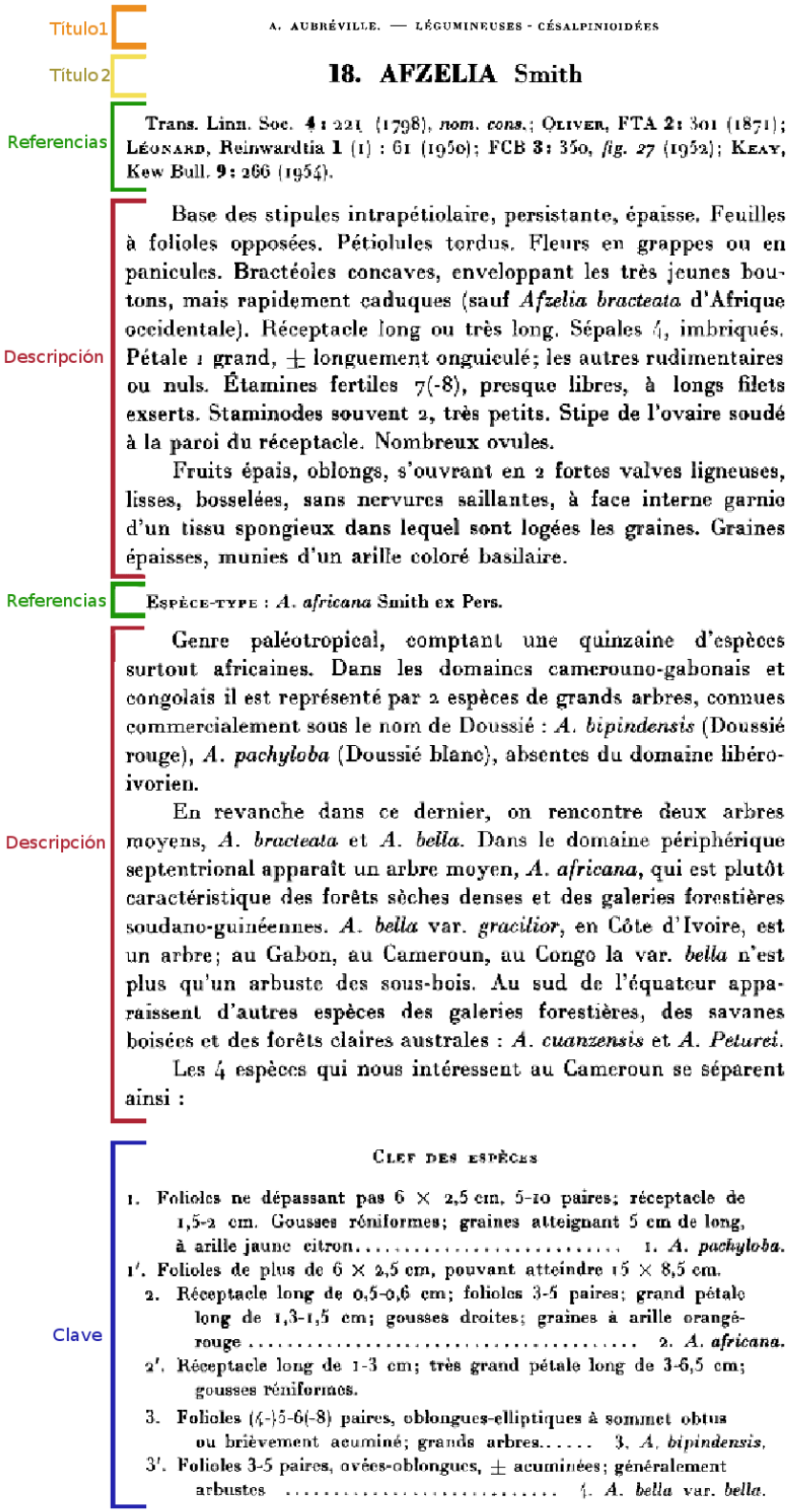
En revanche dans ce dernier, on rencontre deux arbres moyens, *A. bracteata* et *A. bella*. Dans le domaine périphérique septentrional apparaît un arbre moyen, *A. africana*, qui est plutôt caractéristique des forêts sèches denses et des galeries forestières soudano-guinéennes. *A. bella* var. *gracilior*, en Côte d'Ivoire, est un arbre; au Gabon, au Cameroun, au Congo la var. *bella* n'est plus qu'un arbuste des sous-bois. Au sud de l'équateur apparaissent d'autres espèces des galeries forestières, des savanes boisées et des forêts claires australes : *A. cuanzensis* et *A. Pelurei*.

Les 4 espèces qui nous intéressent au Cameroun se séparent ainsi :

CLEF DES ESPÈCES

1. Folioles ne dépassant pas 6 × 2,5 cm, 5-10 paires; réceptacle de 1,5-2 cm. Gousses réniformes; graines atteignant 5 cm de long, à arille jaune citron..... 1. *A. pachyloba*.
1′. Folioles de plus de 6 × 2,5 cm, pouvant atteindre 15 × 8,5 cm.
2. Réceptacle long de 0,5-0,6 cm; folioles 3-5 paires; grand pétale long de 1,3-1,5 cm; gousses droites; graines à arille orangé-rouge ..... 2. *A. africana*.
2′. Réceptacle long de 1-3 cm; très grand pétale long de 3-6,5 cm; gousses réniformes.
3. Folioles (4-)5-6(-8) paires, oblongues-elliptiques à sommet obtus ou brièvement acuminé; grands arbres..... 3. *A. bipindensis*.
3′. Folioles 3-5 paires, ovées-oblongues, ± acuminées; généralement arbustes ..... 4. *A. bella* var. *bella*.

Figura A.2: Fragmento del *corpus «Flore du Cameroun»*



A. AUBRÉVILLE. — LÉGUMINEUSES - CÉSALPINIOIDÉES

18. AFZELIA Smith

Trans. Linn. Soc. **4**: 221 (1798), *nom. cons.*; OLIVER, FTA **2**: 301 (1871); LÉONARD, Reinwardtia **1** (1) : 61 (1950); FCB **3**: 350, *fig. 27* (1952); KEAY, Kew Bull. **9**: 266 (1954).

Base des stipules intrapétiolaire, persistante, épaisse. Feuilles à folioles opposées. Pétiolules tordus. Fleurs en grappes ou en panicules. Bractéoles concaves, enveloppant les très jeunes boutons, mais rapidement caduques (sauf *Afzelia bracteata* d'Afrique occidentale). Réceptacle long ou très long. Sépales 4, imbriqués. Pétale 1 grand, ± longuement onguiculé; les autres rudimentaires ou nuls. Étamines fertiles 7(-8), presque libres, à longs filets exserts. Staminodes souvent 2, très petits. Stipe de l'ovaire soudé à la paroi du réceptacle. Nombreux ovules.

Fruits épais, oblongs, s'ouvrant en 2 fortes valves ligneuses, lisses, bosselées, sans nervures saillantes, à face interne garnie d'un tissu spongieux dans lequel sont logées les graines. Graines épaisses, munies d'un arille coloré basilaire.

ESPÈCE-TYPE : *A. africana* Smith ex Pers.

Genre paléotropical, comptant une quinzaine d'espèces surtout africaines. Dans les domaines camerouno-gabonais et congolais il est représenté par 2 espèces de grands arbres, connues commercialement sous le nom de Doussié : *A. bipindensis* (Doussié rouge), *A. pachyloba* (Doussié blanc), absentes du domaine libéro-ivorien.

En revanche dans ce dernier, on rencontre deux arbres moyens, *A. bracteata* et *A. bella*. Dans le domaine périphérique septentrional apparaît un arbre moyen, *A. africana*, qui est plutôt caractéristique des forêts sèches denses et des galeries forestières soudano-guinéennes. *A. bella* var. *gracilior*, en Côte d'Ivoire, est un arbre; au Gabon, au Cameroun, au Congo la var. *bella* n'est plus qu'un arbuste des sous-bois. Au sud de l'équateur apparaissent d'autres espèces des galeries forestières, des savanes boisées et des forêts claires australes : *A. cuanzensis* et *A. Peturei*.

Les 4 espèces qui nous intéressent au Cameroun se séparent ainsi :

CLEF DES ESPÈCES

1. Folioles ne dépassant pas 6 × 2,5 cm, 5-10 paires; réceptacle de 1,5-2 cm. Gousses réniformes; graines atteignant 5 cm de long, à arille jaune citron..... 1. *A. pachyloba*.
1′. Folioles de plus de 6 × 2,5 cm, pouvant atteindre 15 × 8,5 cm.
2. Réceptacle long de 0,5-0,6 cm; folioles 3-5 paires; grand pétale long de 1,3-1,5 cm; gousses droites; graines à arille orangé-rouge ..... 2. *A. africana*.
2′. Réceptacle long de 1-3 cm; très grand pétale long de 3-6,5 cm; gousses réniformes.
3. Folioles (4-)5-6(-8) paires, oblongues-elliptiques à sommet obtus ou brièvement acuminé; grands arbres..... 3. *A. bipindensis*.
3′. Folioles 3-5 paires, ovées-oblongues, ± acuminées; généralement arbustes ..... 4. *A. bella* var. *bella*.

Figura A.3: Fragmento de género de la «*Flore du Cameroun*»

A.3.1 | Título

Es necesario distinguir dos tipos de títulos. El primero suele indicar la familia a la que pertenece el taxón que se va a describir, tal y como se muestra en la Fig. A.4. En ese ejemplo, ese título aparece en la cabecera de las páginas impares de los volúmenes, dónde se indica en primera posición el autor del volumen; en segunda posición la familia que agrupa los taxones; y para finalizar el rango justo inferior al de familia que no es otro que el de subfamilia.

Figura A.4: Nombre de la familia de taxones del vol. 9 de la *«Flore du Cameroun»*

Por otro lado, el segundo título, será el especímen a detallar en función de la categoría que ocupe en la taxonomía. De este modo, se pueden distinguir diferentes tipos.

- En el caso en que se describa una *tribu*, el segundo título constará de un único campo. En la Fig. A.5, se muestra el nombre de la *tribu* en cuestión. Ésta siempre se escribirá con mayúsculas.

Figura A.5: Título en el caso de describir una tribu

- En el caso en que lo que se describa sea un *género*, el segundo título constará de varios campos. En las Figs. A.6 y A.7, se muestra en primera posición una numeración, indicando cuantos géneros lleva ya descritos para esa misma familia. A continuación, aparece el nombre que procede a describir el género, escrito en mayúsculas y en negrita y finalmente, se escribe la inicial, iniciales o apellido completo del autor o autores que por primera vez describieron la planta. Esta lista es oficial y no pueden usarse otras abreviaturas. Pueden añadirse las fechas en caso de considerarse oportuno, si bien no hay tradición de hacerlo. Y finalmente, pueden aparecer una serie de signos y abreviaturas.

Figura A.6: Título en el caso de descripción de géneros

Figura A.7: Título en el caso de descripción de géneros

- En el caso en que lo que se describa sea la *especie*, el segundo título constará también de varios campos. En la Fig. A.8, se muestra en primera posición la numeración, indicando cuantas especies lleva ya descritas para ese género. A continuación aparece el «nombre genérico», que es compartido por las especies del mismo género, escrito en negrita con la primera letra en mayúsculas y el resto en minúsculas. Despúes, el «epíteto específico» en negrita y minúsculas, lo que hace alusión a alguna característica o propiedad distintiva[14], al origen[15], al hábitat[16], homenajear a una personalidad de la ciencia o de la política, o atender a cualquier otro criterio. Prosigue, como sucedía en el caso de los géneros, con el nombre del autor que por primera vez la describió. Y, finalmente, pueden aparecer una serie de signos y abreviaturas.

Figura A.8: Título en el caso de descripción de especies

- Cuando es necesario trasladar una especie de un género a otro, se citará el nombre del primer autor entre paréntesis antes del autor que ha trasladado la especie. Así, por ejemplo, en la Fig. A.9 la especie descrita por *Harms* fue trasladada al género *Gilletiodendron* por *Vermoesen*, por lo que su nombre quedó como *Gilletiodendron mildbraedii (Harms) Vermoesen*.

Figura A.9: Título al trasladar una especie de un género a otro

[14] ésta puede atender al color *albus*, «blanco»; *cardinalis*, «rojo cardenal»; *viridis*, «verde»; *luteus*, «amarillo»; *purpureus*, «púrpura»; etc.

[15] ésta puede ser *africanus*, «africano»; *americanus*, «americano»; *alpinus*, «alpino»; *arabicus*, «arábigo»; *ibericus*, «ibérico»; etc.

[16] ésta puede ser *arenarius* , «que crece en la arena»; *campestris*, «de los campos»; *fluviatilis*, «de los ríos»; etc.

- A veces, tras el nombre científico, aparecen las partículas *ex* o *in* entre la abreviatura de dos autores, como por ejemplo en la Fig. A.10 con Welwitsch ex Bentham. En el primer caso, quiere decir que el primero sugirió el nombre y el segundo lo publicó válidamente, aunque éste le concedió la autoría del nombre al primero. En el segundo caso, el verdadero autor es el primero, pero lo hace en una obra o artículo de revista que corresponde al segundo, por lo que es conveniente que quede citado a modo de recordatorio.

2. Griffonia speciosa (Welwitsch ex Bentham) Taubert
Género Especie Autor

Figura A.10: Ejemplo de título con partícula ex

A.3.2 | Referencias

Con frecuencia un mismo taxón posee más de un nombre, lo que puede crear mucha confusión entre la comunidad científica. Todos ellos se rigen por el *principio de prioridad*, por el cual la denominación válida es la más antigua. Todas las demás que se atribuyan a ese taxón se consideran sinónimos. De este modo, éstos se indicaran en el apartado de referencias. Pero también se indicarán otros aspectos como los siguientes:

- **Una bibliografía**: Consta de al menos la referencia bibliográfica correspondiente al nombre dado en el título, llamada *diagnosis*. Ésta puede aparecer en varias obras que también se citan. Si los autores son diferentes del dado en el título, se especifican. Al final se dan las indicaciones de la página, la plancha o la figura en referenciada en la obra y una fecha entre parentesis. Ver la Fig. A.11.

Trans. Linn. Soe. 4: 221 (1798), nom. cons.; OLIVER, FTA 2: 301 (1871); LÉONARD, Reinwardtia 1 (I) : 61 (1950); FCB 3: 350, fig. 27 (1952); KEA
Bibliografía

Figura A.11: Bibliografía asociada a la especie *Afzelia pachyloba*

- **Una sinonimia**: Es posible que una familia sea conocida también bajo otro nombre, sin ser el que se da en el título. Normalmente, se considera como legítimo el más antiguo, pero también se señalan sus sinónimos así como los autores que los han empleado y las fechas correspondientes en un apartado llamado sinonimia. Ver la Fig. A.12

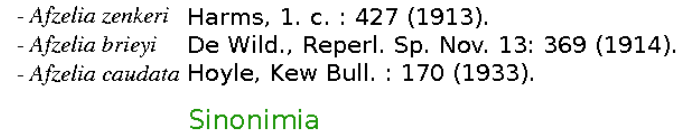


Figura A.12: Sinonimia asociada a la especie *Afzelia pachyloba*

- **Un tipo**: Es un ejemplar de una especie sobre el que se ha realizado la descripción y que, de ese modo, valida la publicación de un nombre científico. Es importante recordar que sólo los nombres de los taxones tienen tipos, como puede observarse en la Fig. A.13, la cual indica que se designa a un ejemplar de herbario (denominado por el apellido del colector y un número de colección: «Linneaeus 5284») con su correspondiente lugar («LINN» que corresponde con el Linn Botannical Gardens en Inglaterra).

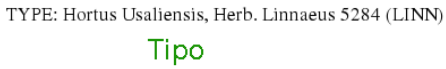


Figura A.13: Tipo situado en la descripción de la *Cassia absus Linné*

También se pueden encontrar *especie tipo* y *lectotipo*, en función de si se describen géneros o especies y subespecies respectivamente. En el primer caso, hace referencia a la especie más representativa del género, como en la Fig. A.14, en el que se presenta el nombre de la especie característica y los autores asociados a ella, acompañado en algunas ocasiones de un estatus de publicación y de los sinónimos de la especie tipo.

Figura A.14: Especie tipo situada en la descripción de la *Afzelia*

En el segundo caso, se refiere al espécimen o elemento seleccionado a partir del material original para servir como tipo nomenclatural cuando no fue asignado con la publicación o por pérdida del mismo. Por ejemplo, en la Fig. A.15.

A.3.3 | Descripción

A grandes rasgos, esta sección incluye una parte de enumeración de los aspectos morfológicos como el color, textura o forma. En este sentido, a nivel de familia, se suelen

Figura A.15: Lectotipo en la descripción de la *Zenkerella citrina Taubert*

describir las características comunes a los diferentes géneros, especies o subespecies. Por lo tanto la descripción suele ser bastante general y relativamente poco detallada. Cuanto más se baje en la clasificación, más detalladas serán las descripciones. Cada uno de los elementos que las componen se encuentra repertoriado en la Tabla A.5.

Elementos(francés)	**Elementos(español)**	**Aparición en rangos**
Feuilles	Hojas	todos
Inflorescences	Inflorescencias	todos
Fleurs	Flores	todos
Infrutescences	Infrutescencias	todos
Fruits	Frutos	todos
Matériel étudié	Material estudiado	Especie, subespecie, cultivar y variedad
Noms vernaculaire	Nombre común	Especie, subespecie, cultivar y variedad

Tabla A.5: Tabla de elementos que componen las descripciones

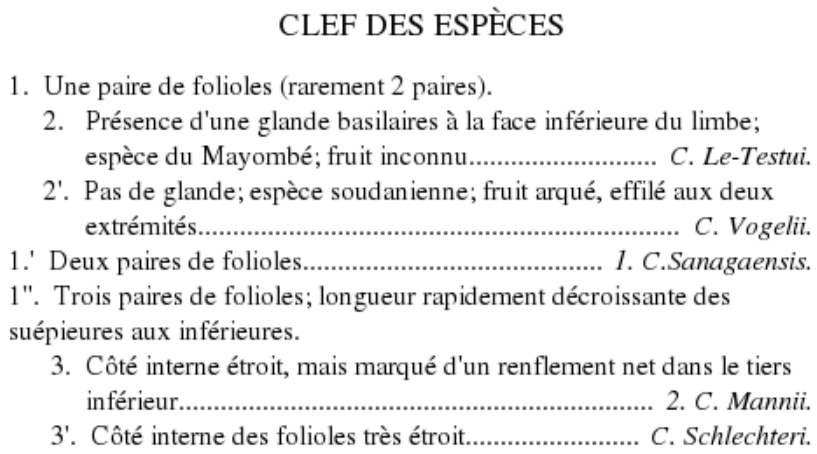
donde «*inflorescence*» («`inflorescencia`») es la disposición de las flores sobre las ramas o la extremidad del tallo, e «*infrutescence*» («`infrutescencia`») es el conjunto de frutos resultantes del desarrollo de una inflorescencia.

Por lo general, éstas contienen una frase separada por cada componente de las plantas. Esto implica la presencia de frases nominales, adjetivos y también adverbios para expresar frecuencia, intensidad y entidades nombradas para denotar dimensiones. Además, esta estructura proporciona un contexto para identificar adjetivos específicos a cada uno de ellos. Por ejemplo, los adjetivos como «*lancéolé*» («`lanceolado`») es adecuado para la descripción de las hojas, mientras que «*multiflore*» («`multiflora`») lo es para las inflorescencias. El primer caso significa que tiene cierta semejanza con una hoja de lanza por su forma, más larga que ancha, con un ápice puntiagudo y, el segundo, tiene un sentido de que produce muchas flores.

A.3.4 | Claves dicotómicas

Las *claves dicotómicas* están presentes después de la descripción si el rango presentado posee otros inferiores. Por tanto, se tienen claves de géneros incluidas dentro de las familias, claves de especies incluidas en los géneros y posibles claves de variedades, subespecies y cultivares dentro de las especies.

Estas claves sirven para determinar el nombre del rango inferior correspondiente a la



CLEF DES ESPÈCES

1. Une paire de folioles (rarement 2 paires).
   2. Présence d'une glande basilaires à la face inférieure du limbe; espèce du Mayombé; fruit inconnu..... *C. Le-Testui.*
   2'. Pas de glande; espèce soudanienne; fruit arqué, effilé aux deux extrémités..... *C. Vogelii.*

1.' Deux paires de folioles..... *1. C.Sanagaensis.*

1". Trois paires de folioles; longueur rapidement décroissante des suépieures aux inférieures.
   3. Côté interne étroit, mais marqué d'un renflement net dans le tiers inférieur..... *2. C. Mannii.*
   3'. Côté interne des folioles très étroit..... *C. Schlechteri.*

Figura A.16: Clave dicotómica de especies para el género *Cynometra*

planta que se trata de identificar, basándose en definiciones de caracteres morfológicos. Su funcionamiento es similar al de los árboles de decisión. La forma de organizar una clave es crear *dicotomías* (a veces *tricotomías* como en el caso de la Fig. A.16), o sea, pares de afirmaciones contrapuestas. En ella, se observa como los ítems $\mathbf{2}$ y $\mathbf{2}'$ son afirmaciones contrapuestas. Por un lado, en $\mathbf{2}$ existe una glándula basilar en la cara interna del limbo y por el otro, al contrario, en $\mathbf{2}'$ esa glándula es inexistente.

Las claves dicotómicas son herramientas útiles para clasificar organismos, y su empleo consiste siempre en tomar una y sólo una de las dos alternativas ofertada. La afirmación que se rechazó no se vuelve a contemplar en el desarrollo de la determinación. De este modo, por ejemplo en el caso de la Fig. A.16, si tomamos como buena la afirmación del ítem $\mathbf{1}''$, tendremos que decantarnos por una de las afirmaciones presentes en los ítems $\mathbf{3}$ y $\mathbf{3}'$. En el caso de $\mathbf{3}$, éste nos lleva a la descripción de la especie *Cynometra Mannii*, cuyo número indica que es la segunda especie que se va a describir para dicho género. En cambio, en el caso de $\mathbf{3}'$, la afirmación nos lleva a la especie *Cynometra Schlechteri*. En este caso, dicha especie no consta de número por lo que dicha especie no se describirá en el volumen en cuestión.

APÉNDICE B

Adquisición electrónica de documentos

La descripción de taxones constituye una información crucial para los científicos en el campo de la botánica, a menudo restringida a aquellos rasgos observados sobre el propio terreno. Concretamente, las floras publicadas antes de la era de la informática son documentos muy ricos en lo que a contenido se refiere, y aún plenamente válidos. Sin embargo, resulta difícil explotarlas adecuadamente y realizar consultas sobre ellas. Por ello, la gestión documental se está convirtiendo cada vez más en una necesidad imprescindible en cualquier ámbito, máxime dado el actual nivel de volúmen y necesidad de acceso alcanzados. La posibilidad de mantener disponible la información de una manera inmediata resulta fundamental.

A pesar de todo esto, sólo existen unas pocas investigaciones que se centren en la digitalización de este legado. Entre los proyectos que tratan de esforzarse en este sentido, se encuentra la digitalización de la «*Flora de Zambia*» («*Flora Zambesiaca*»), llevada a cabo por el herbario del *Real Jardín Botánico de Kew* («*Royal Botanic Garden Kew*»)[1], así como la «*Flora Ibérica*», realizada por el *Real Jardín Botánico de Madrid*[2]. Otros proyectos similares expuestos en la Web son *Flora de Australia*[3] y la *Flora de Norte América*[4]. Todos ellos usan información contenida en las floras únicamente para implementar un sistema de base de datos vía Web, en los que se permiten realizar búsquedas mediante palabras clave tales como nombres científicos, sinónimos o localizaciones geográficas [257]. Sin embargo, ninguna está interesada en descomponer y estudiar todas y cada unas de las descripciones que se encuentran en los textos.

En nuestro caso estamos interesados en ir más allá, por lo que necesitamos realizar diversos análisis sobre la flora en cuestión. Dado que la mayor parte estaba en formato

[1] disponible en `http://apps.kew.org/efloras/search.do`

[2] disponible en `http://www.floraiberica.es/index.php`.

[3] disponible en `http://www.environment.gov.au/biodiversity/abrs/online-resources/flora/main/`

[4] disponible en `http://www.fna.org/`

papel, una fase inicial para su posterior tratamiento computacional ha sido su traslación al formato electrónico [29]. Más concretamente, diferenciamos las siguientes etapas en esta fase:

- La digitalización de los documentos.
- La corrección de errores cometidos en el reconocimiento textual.
- La formalización del documento obtenido en una forma deseada.

Figura B.1: Adquisición electrónica de documentos

Una vez aplicadas cada una de estas tranformaciones [18] dispondremos de un soporte digitalizado que conformará el *corpus*. Antes de comenzar a detallar cada uno de los pasos, cabe señalar que una parte de la gestión documental aplicada a la «*Flore du Cameroun*», más concretamente aquélla realizada sobre el volumen 9 dedicada a las «*Caesalpiniacées*», está disponible en formato digital[5]. Ésta se presenta en formato PDF con una navegación basada en un índice en HTML. Pero además, la «*Flore du Cameroun*» también está disponible por géneros, a partir de la base de datos de «*l'Herbier National du Cameroun*» (base de Letouzey).

B.1 | La digitalización

Quizá una de las preguntas más habituales que uno se hace al considerar soluciones de *adquisición electrónica de documentos*[6] es cómo se puede digitalizar la enorme cantidad de documentos que uno dispone en formato papel de una forma rápida y eficiente. La parte central del problema hace referencia al reconocimiento de caracteres y a la estructuración del contenido textual. Se puede realizar esta etapa manualmente por operadores a bajo coste [28], o también usando herramientas automáticas, como por ejemplo OCR, siempre sujetas a un margen de error más o menos importante dependiendo de la calidad del *software* y del *hardware* usados [28].

[5] disponible en `http://www.orleans.ird.fr/UR_US/biodival/us84/francais/pages/flore_cameroun.html`.

[6] es un sistema utilizado para la búsqueda y almacenamiento de documentos electrónicos y/o imágenes de documentos soportados en papel.

En el marco de este trabajo se optó por la segunda vía. El OCR permite convertir, en forma de texto informático, los documentos que sólo existen en su origen en papel o en un soporte gráfico análogo. Esta operación exige, en general, la realización de tres fases bien diferenciadas. Para empezar, a partir de los documentos se realiza una fase de adquisición de la imagen mediante un escáner, obteniendo su imagen digital. A continuación, se aplica un preprocesamiento sobre ella con el fin de prepararla, eliminando los ruidos e informaciones redundantes y seleccionando las zonas a tratar. Finalmente, se realiza el reconocimiento de la forma de los caracteres.

A continuación, vamos a detallar con un poco más de precisión lo referente al primer y último aspecto, ya que lo relacionado con el preprocesamiento está hoy en día integrado en las herramientas de OCR y se consideran como suficientes en primeras aproximaciones.

B.1.1 | Adquisición de imágenes

Esta fase se hace por barrido óptico. El resultado se guarda en un fichero de puntos, llamados *píxeles*, cuyo tamaño dependerá de la resolución de la parte *hardware*. Los píxeles pueden tener como valores: 0 (apagado) ó 1 (activo) para imágenes binarias, 0 (blanco) hasta 255 (negro) para imágenes de escala de grises, y tres canales de valores de colores entre 0 y 255 para imágenes en color. La resolución se expresa en función del número de puntos por pulgada (ppp). Los valores más frecuentemente usados van de 100 a 400 ppp.

Por ejemplo, el tamaño de un píxel es de 200 ppp, es decir, 0,12 mm. Para un formato clásico A4 y una resolución de 300 ppp, el fichero imagen contendrá 2.520 x 3.564 píxeles. Es importante destacar que la imagen a este nivel es una simple estructura de líneas de píxeles que habrá que explotar para recuperar la información.

B.1.2 | Reconocimiento de caracteres

Un texto es una asociación de caracteres que pertenecen a un alfabeto, agrupada en palabras de un vocabulario dado. El OCR debe reconocer esos caracteres de un modo individual y luego validarlos. Esta tarea no es trivial, ya que un OCR además debe ser capaz de distinguir la forma de cada carácter, pero también de distinguirlos en cada uno de los estilos tipográficos e idiomas. Por ello un sistema de OCR se compone de varios módulos: la *segmentación*, el *reconocimiento* y la *verificación léxica*.

Si comenzamos por la segmentación, ésta permite aislar los elementos textuales, palabras y caracteres. Se basa en medidas de zonas blancas (interlineado[7] y distancias entre caracteres[8]) para hacer la separación. Debido a la gran cantidad de fuentes o tipos

[7] en terminología anglosajona, *interline-spacing*.
[8] en terminología anglosajona, *letter-spacing* o *tracking*.

de letras y la variedad de alineaciones, resulta casi imposible estabilizar los umbrales de separación entre letras.

El siguiente módulo es el dedicado al reconocimiento de caracteres, que permite pronunciarse sobre la identidad de éstos. Para ello es necesario una fase previa de parametrización, definiendo los datos, las medidas o los índices visuales sobre los que se va a apoyar el algoritmo de reconocimiento. Existen básicamente dos métodos: *la matriz de correspondencia* y *la extracción de características*. La primera es la más simple y consiste en comparar lo que el OCR reconoce como un carácter en una biblioteca de matrices de caracteres o plantillas. Cuando una imagen coincide con una de ellas, considerando un determinado nivel de similitud, éste la etiqueta con el correspondiente carácter ASCII. La segunda, la extracción de características, busca características generales, tales como espacios abiertos, formas cerradas, líneas diagonales o intersecciones de líneas. Este método se muestra más versátil que la matriz de correspondencia, que parece dar mejores resultados cuando el OCR se encuentra con un repertorio limitado de estilos de fuente. En otro caso, cuando los caracteres son menos predecibles, este método parece superior.

El proceso de reconocimiento termina con la generación de una lista de letras o de palabras posibles, eventualmente clasificadas por orden decreciente de verosimilitud. Comienza entonces una fase cuyo objetivo principal es mejorar la tasa de reconocimiento haciendo correcciones ortográficas o morfológicas con la ayuda de un diccionario de bigramas, trigramas o n-gramas[9] [287].

B.2 | Evaluación del sistema de OCR

Un empleo eficaz del OCR en la fase de adquisición electrónica de documentos requiere de una evaluación de sus prestaciones. Los tipos de errores más comunes suelen ser los de *segmentación*, *reconocimiento de caracteres* y *los de reconocimiento de palabras*.

B.2.1 | Errores de segmentación

La segmentación de un documento lleva a su descomposición en unidades estructurales tales como regiones textuales o gráficas. Una incorrecta ejecución de la segmentación puede llevar a diferentes errores [5, 6]:

- *Fusión horizontal de regiones textuales*: Lleva a confundir lineas adyacentes pertenecientes a columnas diferentes. Esto influye sobre el orden de lectura como

[9] para secuencias de caracteres, los trigramas que podrían generarse a partir de «hojas perennes» serían «hoj», «oja», «jas», «as », «s p», « pe», «per», «ere», «ren», «enn», «nne» «nes». Algunos sistemas procesan las cadenas de texto eliminando los espacios.

se ve en la Fig. B.2 donde la secuencia inicial: 1, 2, 3, 4 se transforma en 1, 3, 2, 4.

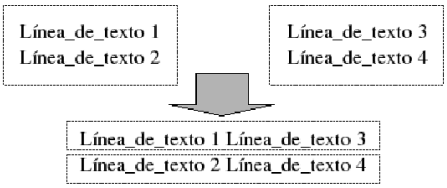


Figura B.2: Fusión horizontal de regiones textuales

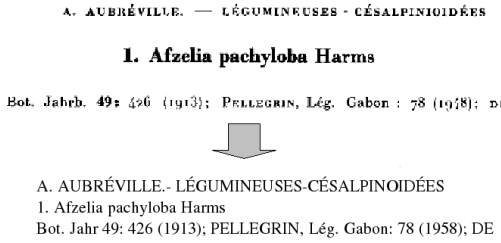


Figura B.3: Fusión vertical de regiones textuales en el título

- *Fusión vertical de regiones textuales*: Conduce a agrupar dos párrafos. No altera el orden de lectura, pero es necesario corregirla para su correcta clasificación como en la Fig. B.3 con la unión del título y de la bibliografía o como en la Fig. B.4 con las anotaciones de los pies de páginas pegados al texto.

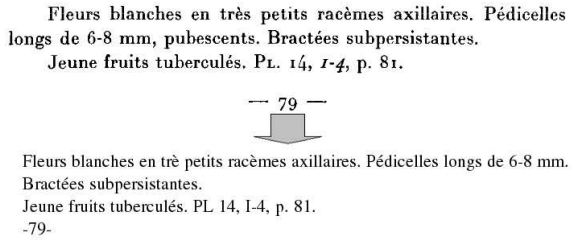


Figura B.4: Fusión vertical de regiones textuales en pies de páginas

- *Regiones no detectadas*: Indica la no detección de una región de texto, que podría llegar a ser asimilada con una gráfica o un ruido. Pero también, por ejemplo, podría provenir de una impresión defectuosa del documento. Es decir, que no todas sus hojas estén impresas con la misma intensidad de tinta, provocando que ciertas letras o palabras no sean detectadas. Es el caso de la palabra «*Arbres*» («`Árboles`») de la Fig. B.5. La intensidad de tinta con la que se encuentra escrita es inferior a las demás, por lo que no se detecta.

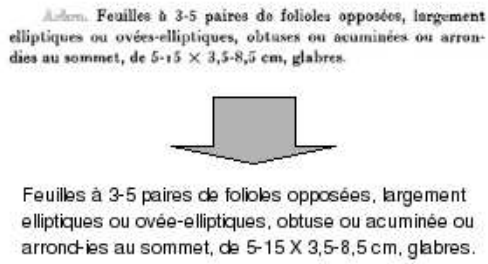


Figura B.5: Regiones no detectadas

- *Gráfica/ruido confundido con texto*: Indica que el OCR tuvo que interpretar una gráfica o ruido como texto. Es común en el tratamiento de fórmulas matemáticas.
- *Fusión horizontal con gráfica/ruido*: Conduce, como en el anterior caso, a la inserción de secuencias erróneas de caracteres en el texto. Podría ocurrir, por ejemplo, que si en el texto apareciese el símbolo ♢, el OCR lo considerara como la letra «O», y que lo añadiera al texto horizontalemente.
- *Fusión vertical con gráfica/ruido*: Idéntico al anterior, salvo que se produce verticalmente.

B.2.2 | Errores de reconocimiento de caracteres

Un OCR puede cometer, entre otros, cuatro tipos de errores de reconocimiento de caracteres:

- *Error de substitución*: Un carácter es remplazado por otro. Es frecuente cuando éstos son morfológicamente próximos (por ejemplo: «o, 0», «c,(», «n, h», «s,5», «à, a»). En la Tabla B.1 se pueden ver algunos ejemplos.

Palabra con error	**Palabra correcta**	**Substitución de letras**
1cgkrement	légérement	«1, l»; «c, e»
4nth2re	anthère	«4, a»; «2, è»
61evbc	élevée	«6, é»; «1 , l»; «b, é»
p6lalcs	pétales	«6, é»; «l, t»; «c, e»
gdndralement	généralement	«d, é»
infkrieure	inférieure	«k, é»
inflorescewce	inflorescence	«w, n»
frztit	fruit	«zt, u»

Tabla B.1: Tabla con errores de substitución de caracteres en el *corpus*

- *Error por omisión*: Un carácter se ignora o se considera como un ruido de la imagen. El sistema puede, de este modo, también rechazarlo o bien porque no lo conoce o porque no está seguro de lo que está reconociendo, tal como se puede observar en la Tabla B.2. En este caso concreto, el sistema puede proponer en algunos casos como carácter de reemplazo uno especial. Suele utilizarse el símbolo «∼», ya que no aparece con mucha frecuencia en los documentos.

Palabra con error	**Palabra correcta**
tigefertile	tige fertile
tiilob	trilobé
saccifor	sacciforme
rdcolt	récolte
ramifi	ramifié
profondeu	profondeur
paissie	épaissie
longueu	longueur

Tabla B.2: Tabla con errores por omisión de caracteres en el *corpus*

- *Error de acentuación*: Frecuente en el *corpus* usado como referencia en este trabajo. Por ejemplo, es muy común ver que la «é» está sustituida por alguno en la secuencia «e, t, è, 6, k, d, c, 2», en función de las palabras, tal como se observa en la Tabla B.3.

Palabra con error	**Palabra correcta**	**Letras acentuadas**
acumink	acuminé	«k, é»
4nth2re	anthère	«2, è»
61evbc	élevée	«6, é»; «b, é»
6largit	élargit	«6, é»
p6lalcs	pétales	«6, é»
p6dicelle	pédicelle	«6, é»
acunindes	acuminées	«d, é»
infkrieure	inférieure	«k, é»

Tabla B.3: Tabla con errores de acentuación de caracteres en el *corpus*

- *Error de desdoblamiento*: Consiste en añadir una letra, doblando un carácter por otros dos donde la morfología de sus formas son próximas. La situación se ilustra en la Tabla B.4. Un ejemplo sería la conversión de la letra «m» por «rn», de la «d» por «cl» o de la «w» por «vv».

Palabra con error	**Palabra correcta**	**Error cometido**
profoncl	profond	«cl, d»
sonmiet | sonnmet | sonunet	sommet	«nmi, mm» | «nn, m» | «nun, mm»
dentifornie	dentiforme	«ni, m»
distrbwion | distribititon	distribution	«dw, ibut» | «itit, uti»
flerrrs | fletirs	fleurs	«rr, u»/«ti, u»
seuleinent | seulenient	seulement	«in, m» | «ni, m»
inflorescerzce	inflorescence	«rz, n»
largeineiit | largeinenl	largement	«in, m»; «ii, n» | «in, m»; «l, t»
miiluscules	minuscules	«il, n»

Tabla B.4: Tabla con errores de desdoblamiento de caracteres en el *corpus*

B.2.3 | Errores de reconocimiento de palabras

Una mala interpretación habitual por parte del OCR es la amplitud de los espacios entre palabras es una fuente de errores frecuentes. Esta mala interpretación puede llevar o bien a la fusión de dos palabras, o bien a la escisión de una en varias. La causa principal de supresión corresponde a una adquisición electrónica defectuosa de la imagen de la palabra. A continuación, en la Tabla B.5, aparecen los errores más frecuentes de este tipo en nuestro *corpus* de referencia.

Palabra con error	**Palabra correcta**	**Error cometido**
ahsent	absent	«h, b»
aigola	angola	«i, m»
aigucs	aiguës	«c, ë»
aiifhere	anthère	«ii, n»; «f, t»; «e, è»
aiit8rieur | aiitkrieur	antérieur	«ii, n»; «8, é» | «ii, é»; «k, é»
aisdment	aisément	«d, é»
aiteignant	atteignant	«i, t»
sibaigus	subaigus	«i, u»
sfigmute	stigmate	«f, t»; «u, a»
skpales, skpalz	sépales	«k, é» | «z, e»
zqflorescence, zrflorescence	inflorescence	«z, i»; «q, n» | «r, n»
zrlflorescenee, zrlfloresceizce	inflorescence	«z, i»; «rl, n»; «c, e» | «iz, n»
zsiixine	zeuxine	«s, e»; «ii, u»
aiiique	afrique	«f, i»; «i, r»

Tabla B.5: Tabla con algunos errores del *corpus*

B.3 | Corrección de errores de OCR

Consideremos un fragmento de nuestro *corpus* $\mathscr{B}$. Se trata del documento 34 (Vol I) de la «*Flore du Cameroun*» que describe las «Orchidaceaes» página 2. El extracto del *corpus* tal y como aparece en el volumen se encuentra en la Fig. B.6. Más tarde, en la Fig. B.7, se muestra el mismo extracto después de realizar el tratamiento de OCR.

```
...Suite à d' autres difficultés survenues après cette date (problèmes financiers, docu-
ments égarés entre Paris et Yaoundé), la p.mtion de cette famille tant attendue fut une
fois de plus retardée. Entre temps, étaient publiées d'importantes révisions systémati-
ques de certains taxons d'Orchidées africaines, remettant en cause une grande partie des
résultats présentés dans le document prêt pour la Flore du Cameroun. Il fallait donc
le remettre à jour. Ce n'est qu'en 1995 qu'étaient enfin réunies les conditions optima-
les de la publication de ce travail. D'une part la subvention de la Banque Mondiale
attribuée au Royal Botanic Garden, Kew et au Laboratoire de Phanérogamie du Muséum dans
le cadre d'un G.E.F. (Global Environment Facility) intitulé :
<(Cameroun Biodiversity and Conservation Managenient Project : Botanical Surveys and
inventories)) initié par le premier partenaire, permettait entre autres choses, la pour-
suite de la Flore du Cameroun, oeuvre gigantesque fondée par A.AUBREVILLE en 1963 et
efficacement traitée depuis et jusqu'à sa mort par R.LETOUZEY, d'autre part, simultanément,
l'acceptation par D.ZLACHETKO & S.OLSZEWSK1, éminents spécialistes en Orchidologie, de
l'université de Gdansk, de reprendre tout le travail de W.SANFORD en le restructurant,
l' actualisant et l' étargissant à l' ensemble des taxons des régions avoisinantes,
susceptibles d'exister au Cameroun....
```

Figura B.6: Orchidaceaes, vol. 34, pág. 2

```
...Suite A d' aulres difficultés siirvenues après cette date (problèmes finianciers, docu-
ments égarés entre Paris et Yaoundé), la p.mtion de cette famille tant attendue ffit une
fois de plus retardée. Entre teinps, étaient publiées d'importantes rhisions systkmati-
ques de certains taxons d'Orchidées africaines. remetlant en cause une grande partie des
résultats présentés dans le document prêt pour la Flore du Canieroun. Il fallait donc
le remettre àjour. Ce n'est qu'en 1995 qu'étaient enfin réunies les conditions optima-
les B, la publication de ce travail. D'une pu? la subvention de la Banque Mondiale
attribuée au Royal Botanic Garden. Kew et au Laboratoire de Phaiiérogamie du Muséum daris
le cadre d'un G.E.F. (Global Environment Facility) intitulé :
<(Cameroun Biodiversity and Conservation Managenient Project : Botanical Surveys and
inventories)) initié par le premier partenaire. permettait entre autres choses, la pour-
suite de la Flore du Cameroun, oeuvre gigantesque fondée par A.AUBREVILLE en 1963 et
efEicacement aililnée depuis etjusqu'à sa mort par R.LETOUZEY, d'autre part. simultanément.
l'acceptation par D.ZLACHETKO & S.OLSZEWSK1, énlinents spécialistes en Orchidologie, de
l'université de Gdansk, de reprendre tout le travail de W.SANFORD en le restmcturaiit,
l' actualisait et l' étargissent a l' eilseinble des taxons des régions avoisinantes,
susceptibles d'exister au Cameroun....
```

Figura B.7: Orchidaceaes, vol. 34, pág. 2, tras OCR

La corrección automática de errores de reconocimiento de OCR es un trabajo arduo que comienza con la identificación de las palabras erróneas, una fase que puede incorporarse después de la fase de segmentación del texto. Con este propósito, los métodos lingüísticos para el REN descansan en la utilización o no de un diccionario, o sobre el análisis de su estructura interna, o sobre el análisis del contexto en el cual aparece [78].

B.4 | Formalización y estructura lógica

Un documento impreso se compone de dos elementos muy importantes: el contenido, es decir, las cadenas de caracteres asociados a su estructura lógica, y la presentación. En este sentido, la presentación se inscribe en el ámbito de la tipografía, y su acabado corresponde al editor, de acuerdo con las pautas que recibe de los grafistas. Concretamente, mediante el uso de hojas de estilo, somos capaces de interpretar de manera correcta el texto, ya que resulta primordial señalar correctamente sus diferentes niveles estructurales (títulos, texto normal, anotaciones, ...).

Sin embargo, cuando hacemos un tratamiento de OCR, toda la información referente a la presentación se pierde. Por lo tanto es necesario hacer todo un trabajo que permita recuperar parte de la estructura. Pero otro aspecto a tener en cuenta es que no todas las obras sobre las que han aplicado las técnicas de OCR usan los mismos estilos a la hora de presentar el trabajo. Algunas de ellas poseen en la parte superior, un título que hace referencia a la familia que están tratando, aunque otros no. Por lo tanto es necesario tratar cada uno de ellos de un modo diferente.

Por ejemplo, en la Fig.B.8 se ve como una vez pasado por el escaneo, existen líneas en el texto que hacen referencia a la paginación. Es el caso de «-*22*-».

D. SZLACHETKO & S. OLSZEWSKI

1.3. Disa nigerica Rolfe

Kew Bull. : 214 (1914). - Summerh., FWTA, ed. 1,2: 414 (1936) ; FWTA, ed. 2, 3: 200 (1968 à Geerinck, FI. Afr. Centr., Orchid. 1: 200 (1984)

Tubercules unique, ovoïde, de 1,7-2,2 x 0,7-1 cm. Tige stérile courte, avec 3 ou 4 feuilles de 10-15 x 0,5-1 cm, lancéolées, aiguës. Tige fertile de 15-35 cm de hauteur, dressée, délicate, glabre, feuillée sur toute sa hauteur. Feuilles 3-4 dont 1 ou 2 gaines básales, atteignant 6,5 cm de longueur et 1 cm de largeur, lancéolées, aiguës, dressées ou subdressées, lâchement aprimées sur la tige.

Inflorescence lâche, longue de 4-20 cm, composée de 15 à 25 fleurs. Fleurs petites, résupinées, lilas à violet foncé. Bractées florales longues de 6-12 mm, lancéolées, aiguës ou acuminées, plus ou moins aussi longues que l'ovaire. Ovaire atteignant 10 mm, dressé, tordu dans sa partie inférieure. Tépales grabres à nervures non ramifiées. Sépale dorsal de 5-7 x 3 mm. ovale à ovale-lancéolé au-dessus d'une partie básale rubanée, aigu, conique. Eperon de 5-7 mm, cylindrique à partir d'une base conique, arrondi au sommet, droit. Pétales de 3-4,3 x 1-2,1 mm, obliquement lancéolés-ovales au-dessus d'une partie básale rubanée, aigus, avec une courte carène surélevée près de la base. Sépales latéraux de 5-6,6 x 1,5-2,5 mm, obliquement oblongs-ovales à elliptiques, apiculés. Labelle long de 4-5 mm, oblong-ovale à oblong-lancéolé, plus large près de la base, subaigu, uninervé, horizontal. Anthère de 1,5-2 mm, horizontale. - Fig. 3, p.23 ; carte 3.

TYPE : Nelson 5, Nigeria (holo-K).

distribution : Nigeria, Cameroun, Zaïre. Alt. 1300-1850 m.
écologie : savannes ouvertes à herbes basses.

```
MATÉRIEL   CAMEROUNAIS   :

Daramola FHI 41189, Bangongo (région de Bamenda ?), (fl. mai).
De Wilde W. es. 2345.. 2479, Bangangté (fl. mai), P, WAG.
Menrillon CNAD 322, Dschang (fl. avr.), P.
Richards 5315, rés. for. Bafut-Ngemba près Bamenda (fl. mars), K.

Section Micranthae Lindley

Gen. Sp. orchid. Fig.: 347 (1838).

Sépale dorsal cochléiforme, non conique. Eperon généralement
pendant, étroitement cylindrique. Labelle pendant. Anthère dressée.

ESPÈCE-TYPE: Disa chrysoslachya Sw.

Trois espèces de celte section devraient se trouver au Cameroun.
Disa renziana Szlachetko

Fragm. FLor. Geobot. 39(2) : 545-546, flg. 2 (1994).

-22-

ORCHIDACEAE

PI. 3. - Disa hirqicornis Rchb. f. : A, fleur ; B, labelle ; C, sépale latéral ;
í). E, pétale ;
```

Figura B.8: Orchidaceaes, vol. 34, pág. 22, tras OCR y corrección de errores

Lo mismo ocurre con los autores de la obra en cuestión, como «*D. SZLACHETKO & S. OLSZEWSKI*» o con los títulos de las partes superiores de los libros, como en «*ORCHIDACEAE*».

Está claro que esa información no debería de estar ahí, en el documento final. Por lo tanto, después de aplicar una primera fase de corrección de errores ortográficos, lo siguiente fue tratar de hacer un tratamiento de las separaciones silábicas y la eliminación de la paginación y los títulos referentes a la obra y no a las descripciones. Se muestra un ejemplo en la Fig.B.9.

Sería conveniente además recuperar el formato de presentación de los documentos. El motivo es que cada uno de los libros hace una descripción clara y concreta de la familia a la que está dedicada. A su vez, estos libros describen los diversos géneros que la componen y las especies que la forman. En este sentido, y pensando en todo el PLN que se va a realizar a continuación, es necesario recuperar parte de dicha estructura. Esto es, ser capaces de saber cual es la información acerca de un género concreto, o de una especie. Pero también, por cada uno de ellos, destacar las partes de *referencias*, *descripción* y de *clave*, como hemos visto en la Fig.A.2.

Por lo tanto, es necesario aplicar alguna técnica que trate de recuperar en cierta medida esa información que se pierde. Una de las soluciones para diferenciar los elementos del texto es mediante la aplicación de un *protocolo de balizado*, concretamente el empleo

```
1.3. Disa nigerica Rolfe

Kew Bull. : 214 (1914). - Summerh., FWTA, ed. 1,2: 414 (1936) ; FWTA, ed. 2, 3:
200 (1968 à Geerinck, FI. Afr. Centr., Orchid. 1: 200 (1984)

Tubercules unique, ovoïde, de 1,7-2,2 x 0,7-1 cm. Tige stérile courte, avec 3
ou 4 feuilles de 10-15 x 0,5-1 cm, lancéolées, aiguës. Tige fertile de 15-35 cm
de hauteur, dressée, délicate, glabre, feuillée sur toute sa hauteur. Feuilles
3- 4 dont 1 ou 2 gaines básales, atteignant 6,5 cm de longueur et 1 cm de largeur,
lancéolées, aiguës, dressées ou subdressées, lâchement aprimées sur la tige.

Inflorescence lâche, longue de 4-20 cm, composée de 15 à 25 fleurs.
Fleurs petites, résupinées, lilas à violet foncé. Bractées florales longues de
6-12 mm, lancéolées, aiguës ou acuminées, plus ou moins aussi longues que l'ovaire.
Ovaire atteignant 10 mm, dressé, tordu dans sa partie inférieure. Tépales grabres à
nervures non ramifiées. Sépale dorsal de 5-7 x 3 mm. ovale à ovale-lancéolé
audessus d'une partie básale rubanée, aigu, conique. Eperon de 5-7 mm, cylindrique
à partir d'une base conique, arrondi au sommet, droit. Pétales de 3-4,3 x 1-2,1 mm,
obliquement lancéolés-ovales au-dessus d'une partie básale rubanée, aigus, avec une
courte carène surélevée près de la base. Sépales latéraux de 5-6,6 x 1,5-2,5 mm,
obliquement oblongs-ovales à elliptiques, apiculés. Labelle long de 4-5 mm,
oblong-ovale à oblong-lancéolé, plus large près de la base, subaigu, uninervé,
horizontal. Anthère de 1,5-2 mm, horizontale. - Fig. 3, p.23 ; carte 3.

TYPE  :  Nelson 5, Nigeria (holo-K).

distribution : Nigeria, Cameroun, Zaïre. Alt. 1300-1850 m.
écologie : savannes ouvertes à herbes basses.

MATÉRIEL   CAMEROUNAIS   :

Daramola FHI 41189, Bangongo (région de Bamenda ?), (fl. mai).
De Wilde W. es. 2345.. 2479, Bangangté (fl. mai), P, WAG.
Menrillon CNAD 322, Dschang (fl. avr.), P.
Richards 5315, rés. for. Bafut-Ngemba près Bamenda (fl. mars), K.

Section Micranthae Lindley

Gen. Sp. orchid. Fig.: 347 (1838).

Sépale dorsal cochléiforme, non conique. Eperon généralement pendant, étroitement
cylindrique. Labelle pendant. Anthère dressée.

ESPÈCE-TYPE: Disa chrysoslachya Sw.

Trois espèces de celte section devraient se trouver au Cameroun.
Disa renziana Szlachetko

Fragm. FLor. Geobot. 39(2) : 545-546, flg. 2 (1994).

PI. 3. - Disa hirqicornis Rchb. f. : A, fleur ; B, labelle ; C, sépale latéral;
í). E, pétale ;
```

Figura B.9: Orchidaceaes, vol. 34, pág. 22, tras separaciones silábicas, y eliminación de paginación y títulos

de XML. Es decir, la indicación del nivel lógico de todos los elementos del texto. Para establecer las *balizas*[10] se emplearon expresiones regulares que captan la presentación. El resultado es el que se ve en la Fig.B.10.

```
<species author="Rolfe" id="1.1.1.1.1.2.2" name="Disa nigerica">
  <type> Nelson 5, Nigeria (holo-K). </type>
  <distribution>Nigeria, Cameroun, Zaïre. Alt. 1300-1850 m. </distribution>
  <ecology>savannes ouvertes à herbes basses. </ecology>
  <material>
   <p>Daramola FHI 41189, Bangongo (région de Bamenda ?), (fl. mai).</p>
   <p>De Wilde W. es. 2345.. 2479, Bangangté (fl. mai), P, WAG.</p>
   <p>Menrillon CNAD 322, Dschang (fl. avr.), P.</p>
   <p>Richards 5315, rés. for. Bafut-Ngemba près Bamenda (fl. mars), K.</p>
  </material>
  <description>
   <p>Kew Bull. : 214 (1914). - Summerh., FWTA, ed. 1,2: 414 (1936) ; FWTA, ed. 2, 3:
   200 (1968 à Geerinck, FI. Afr. Centr., Orchid. 1: 200 (1984)</p>
   <p>Tubercules unique, ovoïde, de 1,7-2,2 x 0,7-1 cm. Tige stérile courte, avec 3
   ou 4 feuilles de 10-15 x 0,5-1 cm, lancéolées, aiguës. Tige fertile de 15-35 cm de
   hauteur, dressée, délicate, glabre, feuillée sur toute sa hauteur. Feuilles 3- 4
   dont 1 ou 2 gaines básales, atteignant 6,5 cm de longueur et 1 cm de largeur,
   lancéolées, aiguës, dressées ou subdressées, lâchement aprimées sur la tige.</p>
   <p>Inflorescence lâche, longue de 4-20 cm, composée de 15 à 25 fleurs.</p>
   <p>Fleurs petites, résupinées, lilas à violet foncé. Bractées florales longues
   de 6-12 mm, lancéolées, aiguës ou acuminées, plus ou moins aussi longues que
   l'ovaire.</p>
   <p>Ovaire atteignant 10 mm, dressé, tordu dans sa partie inférieure. Tépales
   grabres à nervures non ramifiées. Sépale dorsal de 5-7 x 3 mm. ovale à
   ovale-lancéolé audessus d'une partie básale rubanée, aigu, conique. Eperon
   de 5-7 mm, cylindrique à partir d'une base conique, arrondi au sommet, droit.
   Pétales de 3-4,3 x 1-2,1 mm, obliquement lancéolés-ovales au-dessus d'une partie
   básale rubanée, aigus, avec une courte carène surélevée près de la base. Sépales
   latéraux de 5-6,6 x 1,5-2,5 mm, obliquement oblongs-ovales à elliptiques,
   apiculés. Labelle long de 4-5 mm, oblong-ovale à oblong-lancéolé, plus large
   près de la base, subaigu, uninervé, horizontal. Anthère de 1,5-2 mm, horizontale.
   - Fig. 3, p.23 ; carte 3.</p>
  </description>
 </species>
</section>
<section author="Lindley" id="1.1.1.1.1.3" name="Micranthae">
 <biblio>
   <item>Gen. Sp. orchid. Fig.: 347 (1838)</item>
 </biblio>
 <type>Disa chrysoslachya Sw. </type>
```

Figura B.10: Orchidaceaes, vol. 34, pág. 22, tras aplicación de balizado XML

Las balizas utilizadas tienen la ventaja de poder ser interpretadas directamente, dando una idea de como era la estructura inicial del documento, haciendo necesario el laborioso trabajo de preparación de dichos documentos antes de su posterior tratamiento. En la Fig. B.10, por ejemplo, se observa como se distinguen entre etiquetas referentes a especie, tipo, distribución, ecología, material, descripción y demás. En este sentido, los tratamientos en PLN que señalábamos se pretenden aplicar a la parte de descripción del documento, ya que es la parte que ofrece mayor información acerca de los componentes

[10]una etiqueta o baliza es una marca con tipo que delimita una región en los lenguajes basados en XML.

de la planta en cuestión.

El trabajo descrito no es tan sencillo como pueda aparentar. Como señalábamos, cada libro puede tener un método de presentación diferente a los demás, por lo que no es siempre fácil establecer un mecanismo que sea capaz de asignar correctamente cada uno de los componentes a su correspondiente sección. Si observamos la Fig.B.10, la primera línea de la descripción asociada a la especie «*Disa nigerica*», se ve como esa información debería de estar entre unas balizas etiquetadas por «*referencia*».

APÉNDICE C

Análisis sintáctico suavemente dependiente del contexto

Las GA's [149, 152] son un formalismo gramatical *suavemente dependiente del contexto* inicialmente introducido por Joshi, Levy y Takahashi en [149]. Más tarde, Joshi refina ciertos aspectos en [152], estableciendo el concepto formal. En [150] puede encontrarse un estudio reciente de Joshi y Schabes acerca de sus características [11] y, en concreto, una descripción acerca de su capacidad generativa, que resulta ser superior a las GIC's e inferior a las GDC's,.

Definición C.1 *Formalmente, una* gramática de adjunción de árboles *(*GA*) se define como una quíntupla* $\mathscr{G} = (N, \Sigma, I, A, S)$*, donde:*

- N *es un conjunto finito de* símbolos no terminales*, o variables.*
- Σ *es un alfabeto finito de la gramática, o conjunto de* símbolos terminales*, verificando* $\Sigma \cap N = \emptyset$*.*
- I *es un conjunto finito de* árboles iniciales*, es decir, si* $\alpha \in I \Rightarrow Y(\alpha) \in \Sigma^*, \alpha(0) = S$*.*
- A *es un conjunto finito de* árboles auxiliares*, es decir,*

  $$si\ \beta \in A \Rightarrow \begin{array}{l} \beta(0) = X(X \in N) \\ Y(\beta) \in \Sigma^* \times \Sigma^+ \cup \Sigma^+ \times \Sigma^* \end{array}$$

- S *es el* símbolo inicial no terminal *de* N *denominado* axioma*, es decir,* $S \in N$*.*

Los árboles en $I \cup A$ *se denominan* árboles elementales*. Los* árboles iniciales *se caracterizan porque su raíz está etiquetada por el* axioma *de la gramática. Los nodos interiores de los árboles elementales son etiquetados con símbolos no terminales, y los nodos hoja con símbolos terminales o por la palabra vacía.*

Los árboles auxiliares *se comportan como los iniciales, excepto porque la etiqueta de su raíz puede ser un símbolo no terminal arbitrario y uno de sus nodos hoja, llamado* nodo pie*, será etiquetado con el mismo símbolo no terminal de la raíz.*

Se denomina espina *al camino que va desde el nodo raíz al nodo pie de un árbol auxiliar. La espina de un árbol auxiliar delimita dos regiones dentro del mismo. Así, denominamos* contexto izquierdo *(resp. derecho) de un árbol auxiliar a la región del árbol constituida por aquellos nodos que se encuentran a la izquierda (resp. a la* derecha*) de los nodos situados en la espina.*

Por convenio, se usará en este documento la letra α para referirse a los árboles iniciales, la letra β para los árboles auxiliares y la letra γ para árboles elementales.

■

De un modo más intuitivo, las GA's consisten en un conjunto de *árboles elementales*, divididos en *árboles iniciales y auxiliares*. Las GA's imponen una serie de restricciones sobre las etiquetas de los nodos en los árboles elementales, a saber:

- La raíz en los árboles iniciales estará etiquetada con el axioma, y las hojas estarán etiquetadas con terminales o con la palabra vacía ϵ.
- La raíz de los árboles auxiliares puede estar etiquetada con cualquier símbolo no terminal. Las hojas serán etiquetadas con terminales o con la palabra vacía, salvo un nodo cuya etiqueta coincide con la de su raíz. Habitualmente se decora ese nodo mediante el símbolo asterisco, $*$.
- Los demás nodos de los árboles elementales estarán etiquetados con símbolos no terminales.

Estos árboles constituyen la base del formalismo, y sobre ella se definen operaciones de combinación de diferentes árboles elementales mediante una de *sustitución*, y una de *adjunción*, como se explica a continuación. Además, el lenguaje definido por una GA será el conjunto de cadenas $w \in \Sigma^*$, tal que w constituye la *frontera*[1] de un *árbol derivado* a partir de un árbol inicial.

Ejemplo C.1 *Sea la* GA *$\mathcal{G}$ = ({a,b},{S,T},S,{α_1, α_2},{β_1, β_2}), cuyos árboles se describen en la Fig. C.1.*

[1] es la secuencia de los nodos que constituyen las hojas de un árbol.

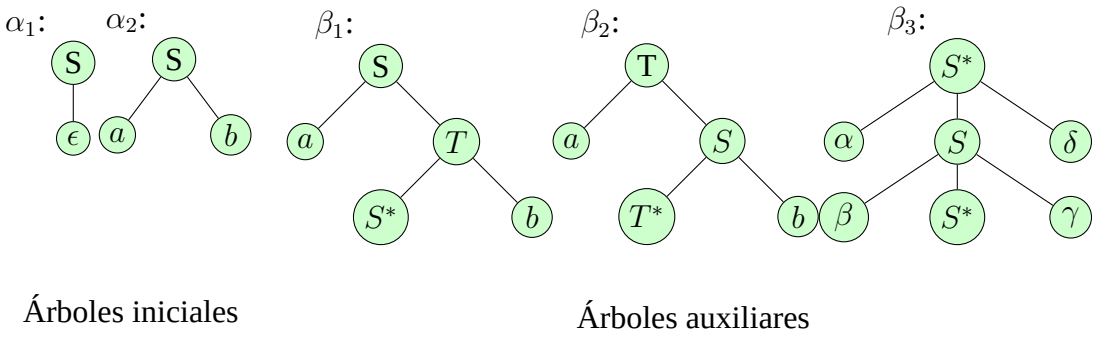


Figura C.1: Árboles iniciales y auxiliares en una GA

■

C.1 | La operación de adjunción

En el formalismo GA, se define una operación básica de composición llamada *adjunción*. Los árboles construidos mediante la composición de otros árboles se denominan *árboles derivados*, y se corresponden con las mencionadas estructuras derivadas, tal como se puede ver en la Fig. C.2.

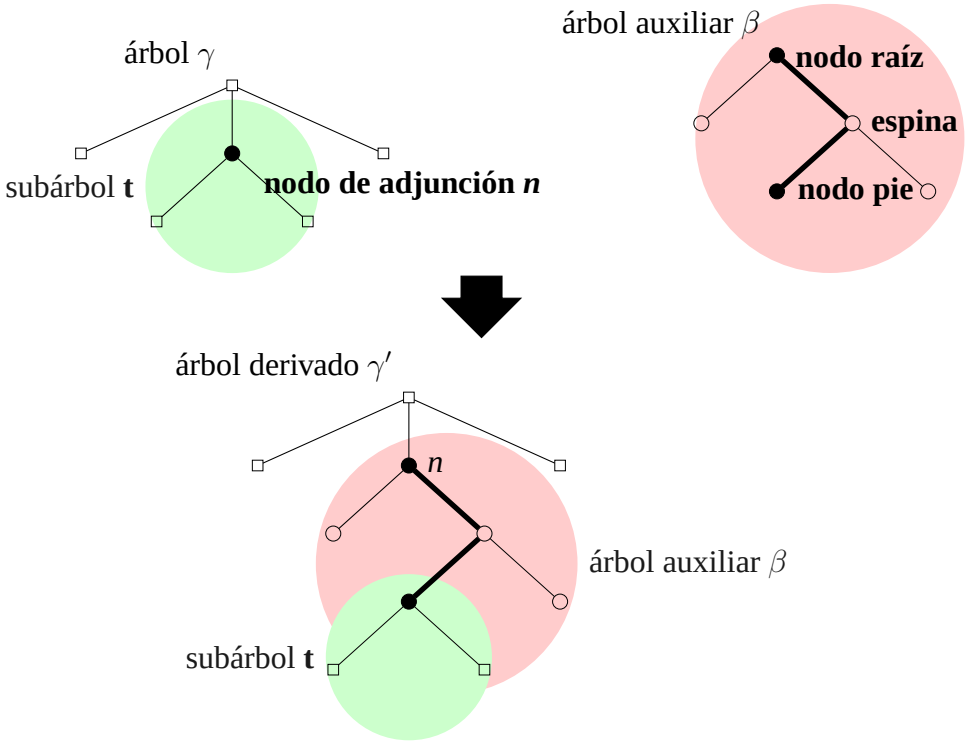


Figura C.2: Operación de adjunción

La operación de adjunción construye un nuevo árbol γ, llamado *árbol derivado*, combinando un árbol auxiliar β y otro árbol γ que puede ser un árbol inicial, auxiliar o

derivado de adjunciones realizadas previamente. Dados γ un árbol que contiene un nodo[2] n, cuya etiqueta es X, y β un árbol cuya raíz está etiquetada con X, el árbol resultante de adjuntar β en el nodo n de γ se obtiene de la siguiente forma:

1. Se poda el subárbol de γ dominado por el nodo de adjunción n, dejando una copia del nodo n. Denominaremos a este subárbol t.
2. El árbol auxiliar β se pega a la copia sobre el nodo de adjunción n, identificando su nodo raíz con n, de tal forma que la raíz del árbol auxiliar se identifica con dicha copia.
3. El subárbol t se pega sobre el nodo pie del árbol auxiliar β, identificando la raíz de t con el nodo pie de β.

Aunque la adjunción sólo depende de las etiquetas de los nodos, se puede especificar para cada nodo un conjunto de restricciones que permiten indicar con más precisión los árboles auxiliares sobre los que se pueden realizar la operación. Éstas se denominan *restricciones de adjunción* y pueden ser los tipos siguientes:

- *Restricciones de adjunción selectiva* (SA), que especifican el subconjunto de árboles auxiliares que pueden participar en una operación de adjunción. Esto es, no es obligatorio realizar una adjunción.
- *Restricciones de adjunción nula* (NA), que impiden la realización de adjunciones.
- *Restricciones de adjunción obligatoria* (OA), que especifica un subconjunto de árboles auxiliares, uno de los cuales ha de ser utilizado obligatoriamente en una operación de adjunción.

Ejemplo C.2 *Un ejemplo de* GA *con restricciones de adjunción que genera el lenguaje* $a^n b^m c^p$*, es el siguiente, mostrado en la Fig. C.3.*

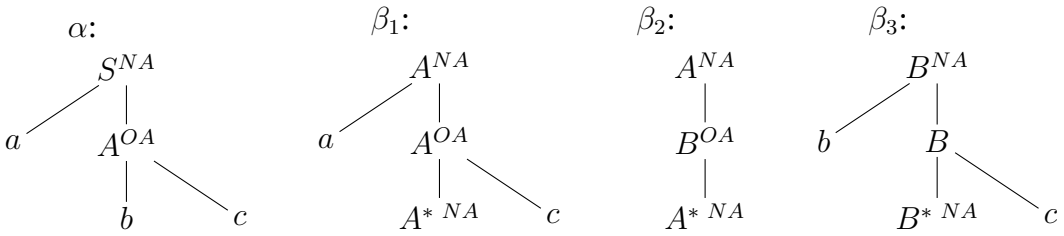


Figura C.3: GA con restricciones que genera el lenguaje $a^n b^m c^p$

■

[2]denominado *nodo de adjunción*.

C.2 | La operación de sustitución

Además de la adjunción ya descrita, las GA's incorporan igualmente la operación de *sustitución* [2], que en este tipo de gramáticas es análoga a la aplicada en las GIC's, aunque en este caso se realiza entre árboles en vez de producciones. Antes de proceder a su definición tendremos en cuenta que:

- Esta operación permite que existan símbolos no terminales en la frontera de los árboles elementales, los cuales se marcan con $\downarrow$ y se denominan *nodos de sustitución*.
- Se permitirá que la raíz de los árboles iniciales esté etiquetada con el axioma o con cualquier otro símbolo no terminal.

Definición C.2 *Decimos que* $\alpha \in I$ *puede ser* sustituido en el nodo marcado para sustitución con dirección p de un árbol $\gamma \in \tau_V$*, sii se cumple que* $\alpha(0) = \gamma(p)$.

■

Gráficamente, la operación de sustitución consiste en colgar un nuevo ejemplar de un árbol inicial dentro de un nodo marcado para sustitución de otro, siempre que la etiqueta no terminal del árbol inicial coincida con la etiqueta del nodo sustitución, tal como se muestra en la Fig. C.4.

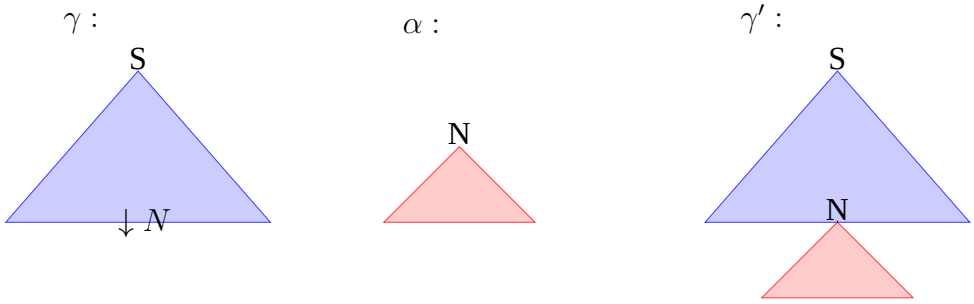


Figura C.4: Operación de sustitución

De forma análoga a como ocurría con la adjunción, podemos encontrarnos con un número indeterminado de árboles iniciales que pueden ser sustituidos en un nodo no terminal N, marcado con $\downarrow$, de la frontera de un árbol. En concreto, denotaremos mediante $Sus(\gamma, p)$ todos aquellos árboles iniciales que puedan ser sustituidos, en el nodo N del árbol γ, por un árbol cuya raíz esté etiquetada con el mismo símbolo que p.

Ejemplo C.3 *El siguiente ejemplo muestra una nueva versión del Ejemplo C.3, en la cual se ha modificado la forma del árbol* β_2 *y se ha añadido un árbol inicial* α_2*, tal y como se observa en la Fig. C.5. Los nodos marcados con* $\downarrow$ *son nodos de sustitución.*

En consecuencia, el árbol α_2 puede ser sustituido en los nodos etiquetados por $C\downarrow$ de los árboles α_1, β_1, β_2 y β_3, teniendo en cuenta que ese nodo posee una restricción local de adjunción nula. Podemos ver esta gramática como un lexicón en el que el terminal a *determina las estructuras sintácticas definidas por los árboles α_1 , β_1 y β_2, el terminal* b *determina la estructura definida por β_3, y el terminal* c *la estructura definida por α_2.*

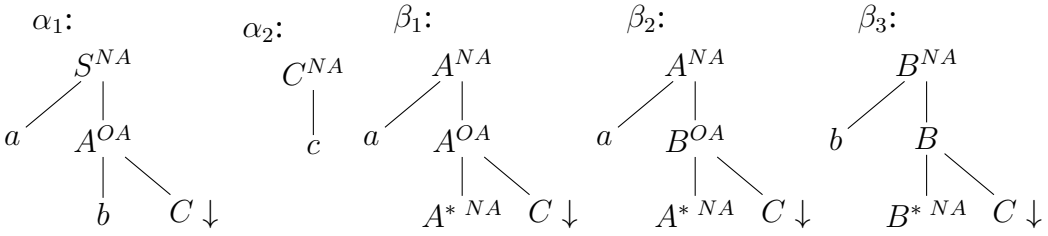


Figura C.5: GA con nodos de sustitución con restricción local de adjunción nula

■

Los nodos marcados para sustitución pueden presentar una restricción local de adjunción nula, como se muestra en el Ejemplo C.3. Se puede hablar también, en un sentido amplio, de restricciones locales respecto a la operación de sustitución aunque éstas serán implícitas. Todo nodo no marcado para sustitución presentará una restricción vacía mientras que los nodos marcados para tal fin presentan una restricción de sustitución obligatoria constituida por todos los árboles iniciales susceptibles de participar en la operación [83].

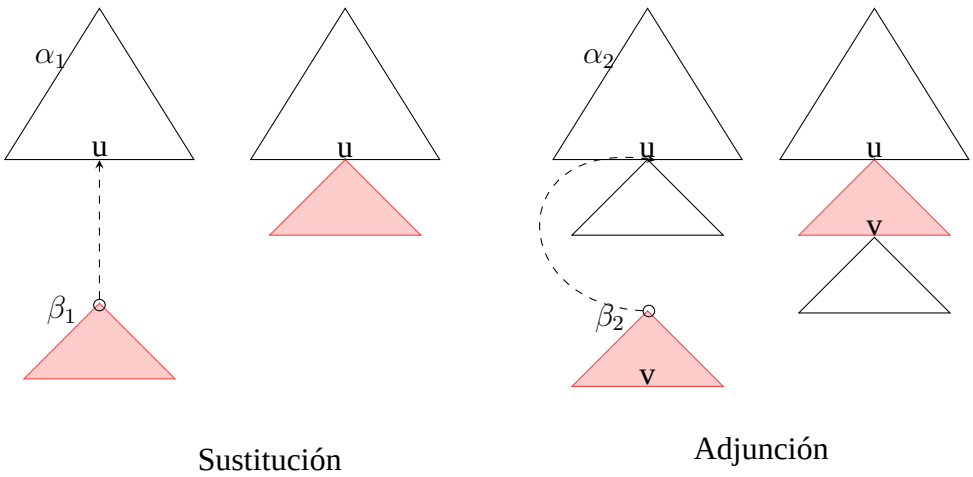


Figura C.6: Combinación de operaciones en GA's

En definitiva, los árboles en las GA's pueden ser combinados usando las operaciones de adjunción y sustitución, tal como se puede observar en la Fig. C.6. Así, la sustitución combina dos árboles, identificando un nodo hoja no terminal u de α_1 con el nodo raíz de β_1

(Fig. C.6-sustitución), mientras que la adjunción identifica un nodo central u del árbol α_2 con el nodo raíz del árbol β_2. En este último caso, el subárbol de α_2 que está encabezado por u se elimina de α_2 y se inserta justo debajo del nodo hoja v de β_2 (Fig. C.6-adjunción).

La operación de sustitución no incrementa la capacidad generativa del formalismo, ya que puede ser simulada mediante una adjunción, pero se considera habitualmente cuando se trabaja con GA's lexicalizadas [13, 154], que introduciremos más tarde. En este caso, algunos nodos hoja[3] de los árboles elementales pueden estar etiquetados por símbolos no terminales. Un árbol inicial α puede ser sustituido en un nodo N, hecho denotado por $\alpha \in Sus(N)$, si su raíz está etiquetada por el mismo no terminal que etiqueta a N. Como restricciones a estas operaciones, no se permite la adjunción sobre nodos marcados para sustitución y en tales nodos sólo pueden ser sustituidos *árboles derivados* de árboles iniciales.

C.3 | Los árboles de derivación

Los *árboles derivados* son obtenidos después de efectuar operaciones de adjunción. A diferencia de las GIC's, en las que el árbol derivado contiene toda la información necesaria para determinar qué operaciones se han realizado sobre qué nodos a lo largo de una derivación, los árboles derivados de las GA's no aportan suficientes datos acerca de cómo se construyen [255, 320], ya que no es posible determinar en que orden se han realizado las adjunciones.

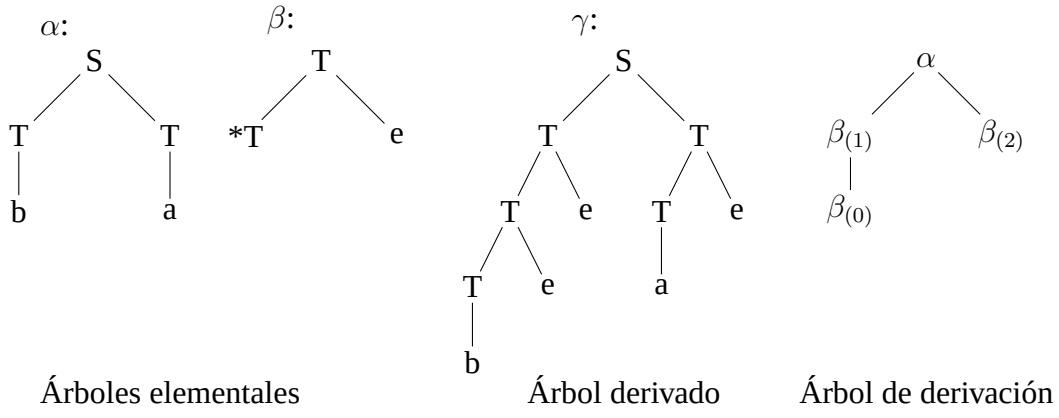


Figura C.7: Árbol de derivación

Para resolver este problema, se introduce una nueva clase de árboles, llamados *árboles de derivación*. En éstos se mostrará de modo inequívoco como se ha construido el árbol derivado, es decir, el orden de adjunciones indicando el nodo en el que tuvo lugar y el

[3] denominados *nodos de sustitución*.

árbol auxiliar involucrado, normalmente, usando direcciones de Gorn[4]. Así, en la parte derecha de la Fig. C.7, se muestra el árbol de derivación correspondiente al análisis de la cadena «*beeae*» según la gramática especificada en la parte izquierda, considerando que:

- La raíz del árbol estará etiquetada con el nombre del árbol inicial.
- Los demás nodos se etiquetarán con nombres de árboles auxiliares.
- Si un árbol auxiliar β ha sido adjuntado en la dirección p de un árbol elemental γ, entonces el nodo etiquetado con γ en el árbol de derivación dominará al nodo etiquetado con β. En este caso, el nodo β estará decorado con la dirección p de γ. Para ilustrarlo con más detalle, se muestra en el Ejemplo C.4.
- No está permitida la adjunción de dos árboles auxiliares en un mismo nodo. Por ello, el orden de las operaciones de adjunción efectuadas sobre un mismo árbol elemental es irrelevante.

Ejemplo C.4 *Partiendo de la gramática del Ejemplo C.1, se va a proceder a realizar la adjunción de β_1 sobre el nodo raíz de α_1, obteniendo γ. Lo mismo ocurre si realizamos una adjunción de β_2 sobre el nodo intermedio T (2) de γ, dando lugar a γ'. Los árboles de derivación están descritos a continuación, ilustrado en la Fig. C.8:*

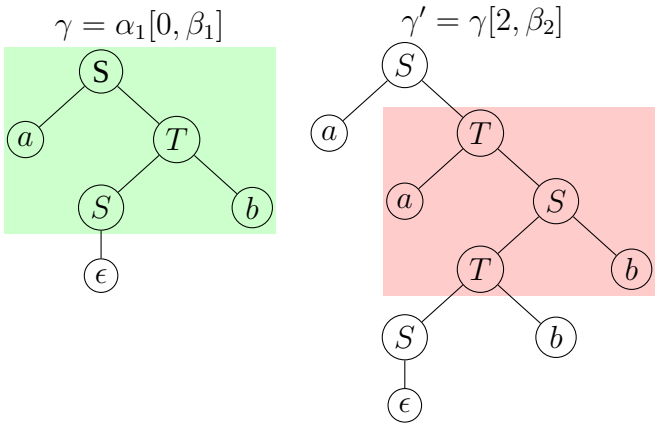


Figura C.8: Obtención de las operaciones de adjunción mediante derivación

■

C.4 | Variantes de las GA's

Antes de entrar a explicar cuales son las ventajas de las GA's sobre las GIC's, consideramos que es necesario dar una visión de ciertas variantes de las GA's como son

[4]en el direccionamiento de Gorn, se usa el 0 para referirse a la raíz y *n* para referirse al *n-ésimo* hijo del nodo raíz.

las *gramáticas de adjunción de árboles lexicalizadas* (GAL), las GAER's y las GIA's.

C.4.1 | Gramáticas lexicalizadas

Cuando se habla de la lexicalización de una gramática, se busca que, tanto las reglas sintácticas como los símbolos terminales, es decir, los items léxicos, no vayan por separado. En este sentido, podemos definir las GA's lexicalizadas como sigue:

Definición C.3 *Se dice que una gramática está* lexicalizada *(*GAL*) [1, 279] si posee dos características:*

- *Un conjunto finito de árboles elementales, cada uno asociado a un elemento léxico, es decir, un símbolo terminal, denominado* ancla*.*
- *Un conjunto finito de operaciones que permita la composición de las estructuras elementales.*

■

Pero además de estas dos características hay que poner como restricción que las operaciones conduzcan a un número finito de resultados. En este sentido, el hecho de que tanto las árboles elementales como el conjunto de operaciones y de resultados sean finitos, garantiza que las GAL's sean finitamente ambiguas. Es decir, dada una frase de longitud finita, ésta puede ser analizada mediante un número finito de árboles elementales. De este modo también se deduce que el reconocimiento de una oración es un problema decidible [83]. De hecho los árboles elementales no pueden estar constituidos únicamente por nodos marcados para la sustitución, ya que no se incluiría al menos un elemento léxico en su frontera. De este modo, todos aquellos nodos que se encuentran en la frontera del árbol elemental y que no sean anclas, se irán completando durante el análisis por los demás símbolos terminales, es decir, los items léxicos [50].

En definitiva, una GA se dice que está lexicalizada si cada uno de los árboles elementales posee al menos un nodo frontera etiquetado con un símbolo terminal. Para facilitar la descripción de los árboles elementales, en las GAL's, se permite que cualquier no terminal etiquete la raíz de un árbol inicial, cuando inicialmente las raíces de los árboles iniciales tenían que estar etiquetados por el axioma de la gramática. Una consecuencia directa de esto es que se permite que un árbol inicial se pegue en un nodo de sustitución de la frontera de otro árbol elemental, con la condición de que el no terminal que etiqueta dicho nodo de sustitución coincida con la etiqueta de la raíz. En este sentido, una GAL puede interpretarse como un lexicón donde cada lema está asociado a un conjunto de árboles elementales [2] en la que dicha palabra actúa como ancla.

Concretamente, en las GAL's diseñadas para reconocer LN's, es frecuente que exista un determinado terminal en un árbol elemental que juegue un papel más destacado que

los demás. Un ejemplo podría ser, el verbo. En este sentido, esto nos puede traer un serie de ventajas adicionales, tales como:

- Para reducir el tamaño de la gramática, en vez de utilizar como anclas a símbolos terminales en los árboles elementales, se podría utilizar a *símbolos preterminales*, es decir, colecciones de símbolos terminales. Para distinguirlos, serán decorados con el símbolo $\diamond$.
- La organización de la gramática puede realizarse a través de una colección de árboles que comparten ancla y donde se van a reflejar distintos entornos sintácticos. En este sentido, se dice que se trabaja con *familias de árboles semi-lexicalizados*, ya que las anclas no son terminales concretos sino conjuntos de ellos, y contienen todas los posibles árboles para una misma familia. Es el caso de, por ejemplo, si consideramos el conjunto de árboles asociados a los verbos transitivos.

Siguiendo con esta idea, vamos a mostrar a partir del ejemplo C.5 como se representa un árbol asociado a la familia de los verbos transitivos en voz activa y pasiva. Concretamente, el ejemplo C.6 muestra cómo usar esos árboles en un análisis concreto obteniendo los árboles derivados.

Ejemplo C.5 *En este ejemplo, se muestran dos árboles elementales anclados con un* símbolo preterminal *a través de una forma verbal, cuyos argumentos en la voz activa son un sujeto NP y un objeto directo NP [2]. El primer árbol muestra todos los argumentos en su posición natural, en cambio el segundo muestra como el sujeto de la voz activa pasa a ser el complemento agente en el segundo, precedida de un ancla* «par» *(*«`por`»*). El objeto directo del primero pasa a ser el sujeto del segundo. Podemos verlo en la Fig. C.9.*

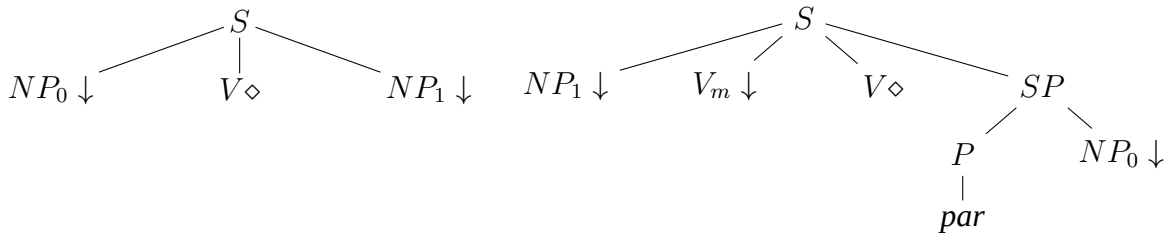


Figura C.9: GAL para frase activa y pasiva usando un ancla

■

Ejemplo C.6 *Usando el ejemplo C.5, vamos a ilustrarlo mediante la frase* «La feuille possède une nervure» *(*«`La hoja posee una nervadura`»*) en voz activa, que pasándola a voz pasiva se convierte en* «Une nervure est possédée par la feuille» *(*«`Una nervadura es poseída por la hoja`»*), ilustrado en la Fig. C.10.*

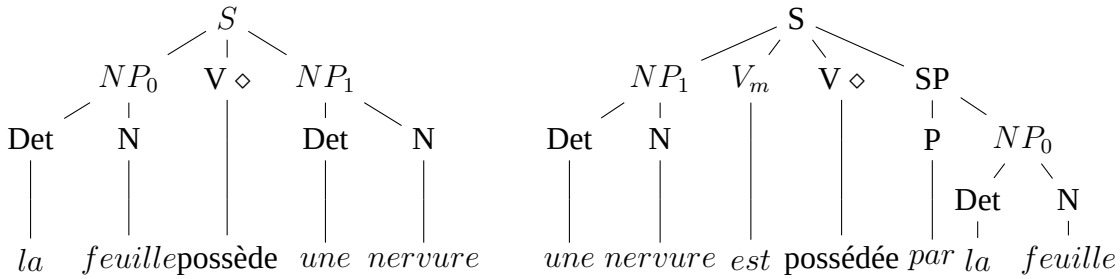


Figura C.10: GAL para frase activa y pasiva con la forma verbal *possède* como ancla

■

C.4.2 | Gramáticas basadas en estructuras de rasgos

Una de las aportaciones más importantes en lo que a lingüística computacional se refiere es la descripción declarativa de fenómenos lingüísticos mediante *estructuras de rasgos*. En este sentido, las restricciones más comunes que aparecen cuando se habla de las GIC's hacen referencia sobre todo a los fenómenos de concordancia y subcategorización. Las gramáticas que se basan en estructuras de rasgos logran tratar ambos casos [160].

Así, un *rasgo* no es más que un conjunto de pares atributo-valor, donde el valor puede ser atómico o a su vez otro rasgo, y el atributo es el que lleva el nombre que lo identifica. Por ejemplo, *«número=plural»* es un rasgo (*atributo=número, valor=plural*). En este sentido, las GA*'s basadas en estructuras de rasgos* (GAER's) son una variante de las GA's. En ellas, los nodos de los árboles elementales pueden estar decorados con dichas estructuras, describiendo el nodo y su relación con los demás nodos del mismo árbol. Por este motivo, estas gramáticas se caracterizan por hacer complejas descripciones formales mediante su uso y por utilizar una operación general para la combinación y comprobación de la información gramatical, conocida como *unificación* [286]. La unificación hace referencia a la composición de los rasgos mediante la operación que denotamos [159] por $\cup$.

Para que dicha operación se produzca, las estructuras deben tener información compatible, pues en caso contrario no unificarían. La compatibilidad tiene que ver con la naturaleza de los rasgos y sus valores. Los rasgos que sólo aparecen en una de las estructuras unificadas se incorporaran a la estructura resultado de la unificación, logrando combinar tanto la información común como la diferente.

Ejemplo C.7 *Supongamos que tenemos un rasgo denominado* superior (top) *que recoge la información acerca de las restricciones que debe mantener un nodo con su ancestro. Así, la estructura de rasgos* superior (top) *asociada a un nodo indicará que los nodos que*

lo dominan tendrán como categoría la de sintagma nominal, y además, su género será el de singular y en tercera persona.

$$\left| \begin{array}{c} superior :< cat >= SN \\ < genero >= SG \\ < persona >= 3 \end{array} \right|$$

■

En el caso de las GAER's, las operaciones de adjunción y sustitución se definen en términos de la unificación de estructuras de rasgos, por lo que las restricciones de adjunción pueden ser modeladas a través del éxito o del fracaso de la unificación entre las estructuras de rasgos de los nodos. Para comprenderlo mejor, vamos a ilustrarlo mediante unos ejemplos.

Ejemplo C.8 *Supongamos que tenemos el árbol de la parte izquierda S que posee un nodo en la frontera X con el rasgo t_r. Supongamos que tenemos otro árbol cuyo nodo raíz X posee los rasgos t_p y b_p. Al realizar la sustitución en el árbol S es necesario realizar una fase de unificación entre los rasgos con mismo nombre de atributo tal como $t_r \cup t_p$ así como con b_p, permitiendo indicar la viabilidad de dicha sustitución. Es lo que se observa en el árbol que se encuentra más a la derecha en la Fig. C.11.*

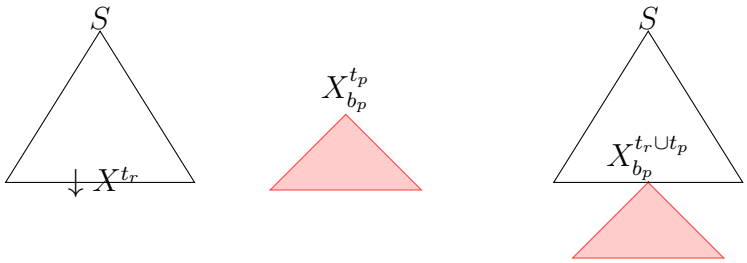


Figura C.11: Árbol representando unificación de rasgos

■

Ejemplo C.9 *Siguiendo con la descripción del Ejemplo C.7 y teniendo en cuenta el Ejemplo C.8, supongamos en un primer momento que el rasgo t_r del nodo X de S es* <género >= SG, *y que el rasgo b_p del segundo árbol es* <persona >= 3. *Si hacemos la unificación entre ellos se está generando la unión de dos estructuras que dan como resultado el deseado. Por el contrario, en el segundo caso, si t_r es* <género >= SG *y t_p es* <género >= PL *no se consigue la unificación ya que se está proporcionando valores diferentes para un mismo rasgo.*

$$| < genero >= SG| \cup | < persona >= 3| = \left| \begin{array}{l} < genero >= SG \\ < persona >= 3 \end{array} \right|$$

$$| < genero >= SG| \cup | < genero >= PL| = FAIL$$

■

De esta manera los rasgos se pueden emplear para tratar de eliminar determinadas restricciones locales de adjunción o sustitución, impidiendo la unificación en el caso de que no se satisfagan las condiciones exigidas por las restricciones locales.

C.4.3 | Gramáticas de inserción de árboles

Las *gramáticas de inserción de árboles* (GIA's) son una variante de las GA's que introducen una restricción sobre los árboles auxiliares. En este sentido, las GIA's se definen de forma análoga a las GA's, con la salvedad de que sólo permiten la inserción de un árbol auxiliar a la izquierda o a la derecha del nodo de adjunción. Esta condición implica concretamente que los árboles auxiliares tengan su espina como frontera izquierda o derecha. Así, la operación de adjunción es bastante restringida. No permiten [49]:

- La inserción de árboles auxiliares que no posean un nodo frontera que esté situado a la izquierda o a la derecha del nodo pie;
- La adjunción de un árbol auxiliar izquierdo (o derecho) en la espina de un árbol auxiliar derecho (o izquierdo);
- La adjunción en los nodos raíz y pie de los árboles auxiliares.

El mayor interés de las GIA's proviene del hecho de que son analizables, como las GIC's, con una complejidad $\mathcal{O}(n^3)$ cuando las GA's tiene una complejidad $\mathcal{O}(n^6)$, donde n denota la longitud de la cadena de entrada. Es más, la mayor parte de las GA's son esencialmente GIA's, siendo posible la construcción de analizadores sintácticos híbridos GA/GIA [13].

C.5 | Ventajas de las GA's sobre las GIC's

Las propiedades más importantes de las GA's son las siguientes [11]:

- Las GIC's están incluidas en las GA's, aunque las GA's pueden asignar a las cadenas de un LIC una estructura que es imposible de generar utilizando GIC's [152].

Ejemplo C.10 *La gramática* $\mathcal{G} = (\{a, b, c, d\},\ \{S, A, B\},\ S,\ \{\alpha\},\ \{\beta_1,\ \beta_2,\ \beta_3\})$*, representada en la Fig. C.12, tomada de [11], genera el lenguaje* $L(\mathcal{G}) = \{a^n b^m c^n d^m / n, m > 0\}$*; que es independiente del contexto. En este sentido, se obtiene el árbol derivado* γ *aplicando en orden las operaciones que se muestran en la derivación de la derecha. De este modo se obtiene la cadena* «abbbcddd» *ilustrada en la Fig. C.13. Este árbol* γ *no puede ser generado por una* GIC.

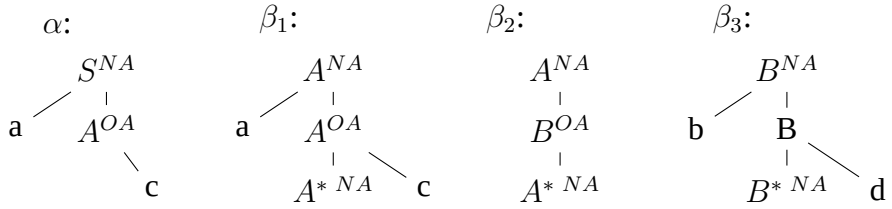


Figura C.12: Una GA para $a^n b^m c^n d^m$

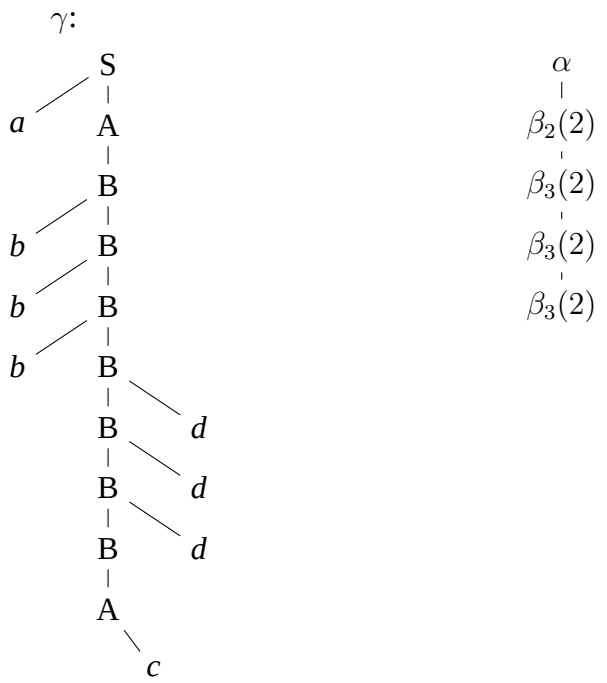


Figura C.13: Árbol derivado para «$abbbcddd$» y el árbol de derivación

■

- Las GA*'s pueden ser analizadas en tiempo polinomial* [12]. En lo que respecta a la complejidad temporal requerida en su tratamiento, ésta es $\mathcal{O}(n^9)$ en el peor de los casos para un texto de longitud n, pero se reduce a $\mathcal{O}(n^6)$ si se verifica la *propiedad del prefijo válido* (PPV).

Definición C.4 *Formalmente, un analizador sintáctico satisface la* PPV *si al leer la subcadena* $a_1 \cdots a_k$ *de la cadena de entrada* $a_1 \cdots a_k a_{k+1} \cdots a_n$*, se garantiza que hay una cadena* $b_1 \cdots b_m$*, donde* b_i *no tiene porque formar parte de la cadena de entrada, tal que* $a_1 \cdots a_k b_1 \cdots b_m$ *es una cadena válida del lenguaje.*

■

Intuitivamente, aquellos analizadores sintácticos que satisfacen la PPV se caracterizan por garantizar que, en tanto que lean una cadena de entrada de izquierda a derecha, las subcadenas leídas son *prefijos válidos* del lenguaje, es decir, tan pronto como es posible en la lectura de la cadena de entrada de izquierda a derecha, se posibilita su corrección por simples inserciones de sufijos.

Es además importante señalar que la complejidad en el peor de los casos sólo se alcanza en el tratamiento de ambigüedades sintácticas. En consecuencia, podemos sacar partido del fenómeno conocido comúnmente como *determinismo local* [177], lo que en la práctica permite mejorar la eficiencia computacional, ya que los programadores suelen diseñar gramáticas que son lo suficientemente próximas de las deterministas.

- Las GA's permiten igualmente capturar dependencias anidadas, usadas en construcciones de los LN's como la *replicación* [341], y ciertas clases de dependencias cruzadas, usadas en la *concordancia* [110], tal como se puede ver en la Fig. C.14.

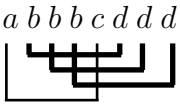


Figura C.14: Relaciones cruzadas en la cadena «$abbbcddd$»

- Las GA's poseen la *propiedad del crecimiento constante* [11]. Esta propiedad hace referencia al hecho de que si las cadenas de un lenguaje se disponen en orden de longitud creciente, la longitud de dos cadenas situadas en posiciones consecutivas no pueden diferir sustancialmente. De hecho, la longitud de cualquier cadena deberá poder obtenerse como una combinación de un conjunto finito de longitudes fijas. En este sentido, esta propiedad hace referencia a que las frases de un lenguaje se pueden construir a partir de un conjunto finito de construcciones de tamaño acotado mediante el uso de operaciones lineales.

- Las GA's poseen un DLE [153] más amplio que las GIC's. Esto se refiere al hecho de que los árboles elementales que conforman la gramática pueden abarcar extensiones de las sentencias más amplias que las correspondientes producciones que se usan en las GIC's. De este modo permiten la localización de dependencias a larga distancia

dentro de una misma estructura elemental, incluso sobre árboles que poseen varios niveles[5].

En definitiva y a diferencia de otros formalismos gramaticales, las GA's permiten establecer dependencias entre los nodos de los árboles que están más separados porque los elementos básicos del formalismo son árboles. Así, la relaciones entre un constituyente y su gobernante puede definirse localmente en las GA's, mientras que en la mayoría de los demás formalismos ello resulta bastante más complejo.

Desde un punto de vista lingüístico, un DLE como el comentado para las GA's permite capturar dependencias lejanas entre las categorías, otorgando una mayor capacidad generativa, algo especialmente interesante en el contexto del PLN.

Ejemplo C.11 *Vamos a ilustrar la propiedad* DLE *tomando como ejemplo un fragmento de gramática, inspirada de [150], de hecho una* GIC*, definida por las siguientes producciones:*

$$
\begin{array}{lcl}
S & \rightarrow & SN \;\; SV \\
SV & \rightarrow & SV \;\; Adv \\
SV & \rightarrow & V \;\; SN \\
SN & \rightarrow & Det \;\; N \\
Det & \rightarrow & \textit{les} \mid \textit{des} \\
N & \rightarrow & \textit{arbres} \\
N & \rightarrow & \textit{feuilles} \\
V & \rightarrow & \textit{possèdent} \\
Adv & \rightarrow & \textit{temporairement}
\end{array}
$$

Si quisiéramos especificar la dependencia entre «possèdent» *(*«`poseen`»*),* «arbres» *(*«`árboles`»*) y* «feuilles» *(*«`hojas`»*) en una única dependencia, se tendrían que usar la primera y tercera producción, dando lugar a una única producción que sería $S \rightarrow SN\ V\ SN$, lo que llevaría a la eliminación de SV en la gramática.*

En este sentido, al tomar producciones independientes del contexto como especificaciones de la DLE*, no se puede expresar localmente la dependencia entre el verbo y sus argumentos y mantener el sintagma SV. Sin embargo, en las* GA*'s, podemos especificar la dependencia entre V, el SN y el complemento, conservando el SV en la gramática, como se muestra a continuación en la Fig. C.15.*

[5] concretamente, ello permite, por ejemplo, que en las GAL's podamos establecer relaciones de coocurrencia de larga distancia entre el ancla y los nodos que poseen restricciones.

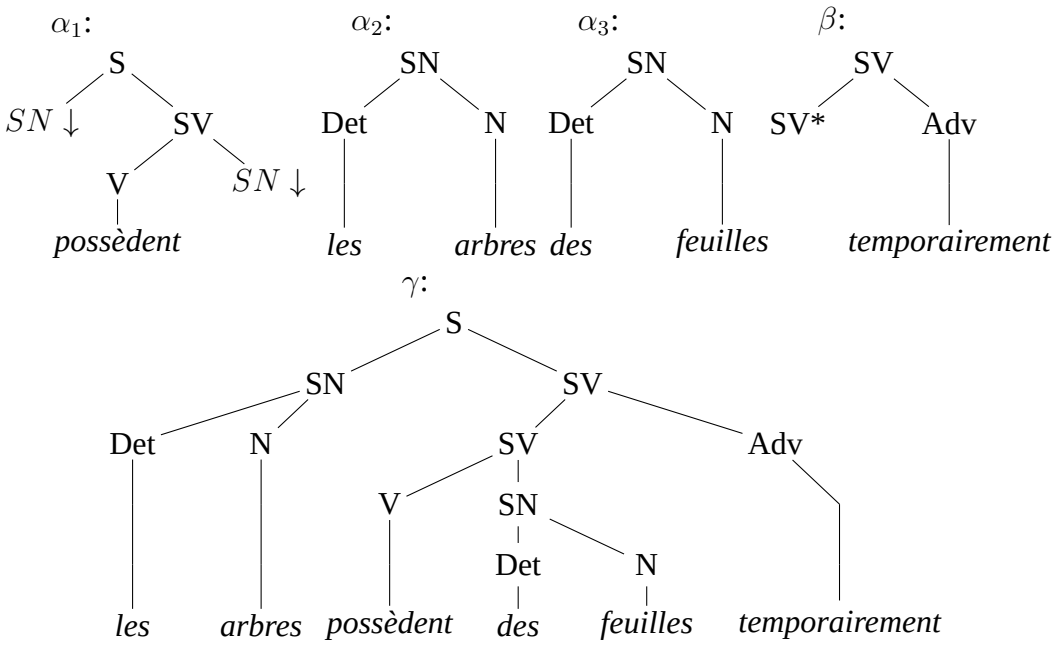


Figura C.15: Dominio de localidad extendido de las GA's

■

- La *factorización de la recursión en el* DLE. Los árboles son los DLE's sobre los que se van a establecer dependencias como pueden ser la concordancia o la subcategorización. Mediante la inserción de árboles auxiliares dentro de otros, usando la operación de adjunción, se permitirá que las dependencias creadas puedan ser de larga distancia, aunque se hayan especificadas localmente en un sólo árbol [11, 172].

APÉNDICE D

Las redes semánticas y los marcos

La representación del conocimiento es un problema central en IA. De entre las preguntas claves a las que nos podemos enfrentar en su tratamiento, la elección del formalismo de representación, el método y la forma de acceso a los conocimientos son esenciales. La lógica, aunque constituye una buena representación del conocimiento, no aporta mucho cuando tenemos que describir la estructura compleja del mundo y escoger un diseño de implantación. Para ello es muy útil agrupar las propiedades de los objetos en unidades de «descripción». Esto permite al sistema focalizar su atención en un objeto completo, sin considerar el resto de hechos que conoce, lo cual es importante para evitar la explosión combinatoria. Además, no sólo los objetos son unidades con estructura, sino que también lo son los acontecimientos y las secuencias típicas de acontecimientos o escenarios. Se trata pues de agrupar varias fórmulas lógicas en estructuras más amplias: objetos estructurados tales como *redes semánticas* o *marcos*. Los objetos estructurados, o esquemas, son organizaciones agrupadas de experiencias típicas adquiridas y que suponemos operativas en ocasiones futuras.

D.1 | Redes semánticas

En su sentido más amplio, una red se compone de un conjunto de nodos unidos entre sí por cierto tipo de enlace. En el caso que nos ocupa, se trata de un modelo teórico llamado *redes asociativas* donde cada nodo representa un concepto, o incluso una proposición, y los enlaces se corresponden a las relaciones que se establecen entre estos conceptos. Estas relaciones pueden referirse a causalidad, pertenencia e inclusión; pero también a categorías gramaticales como son un sujeto o un objeto. Concretamente, las redes asociativas destinadas a comprender el LN, se conocen como *redes semánticas* [82]. Actualmente, las redes asociativas sirven para representar, además de las reglas semánticas, asociaciones físicas o causales entre objetos.

Como ya se dijo en el Capítulo 5, una *red semántica* es una estructura de representación del conocimiento lingüístico, donde a las relaciones entre los diversos elementos semánticos se les da un aspecto de grafos cuyos nodos pueden representar objetos, entidades, atributos, eventos o estados; y donde los arcos representan sus relaciones. En particular, las redes semánticas pueden agruparse en dos tipos: los *sistemas asertivos* y los *taxonómicos* en función de si permiten realizar afirmaciones particulares o bien relacionar los conceptos mediante jerarquías. En este sentido, y con el fin de ilustrarlos vamos a detallar para el primer tipo los denominados *modelos de memoria semántica* o *grafos relacionales* [239], y los *grafos de dependencias conceptuales* de Schank [280, 281], mientras que las *jerarquías de conceptos* [36] harán lo propio con las segundas.

D.1.1 | Modelos de memoria semántica o grafos relacionales de Quillian

El primer modelo de representación formalizado fue desarrollado por Quillian [239], que basándose en los trabajos de Selz [289], trató de construir un modelo computacional cuya fundamentación pretendía ser la propia mente humana, con el fin de llegar a tratar el LN. El modelo desarrollado consistía en representar el significado de los términos de modo similar a como lo hacen los diccionarios. En este sentido, esta representación consta de un conjunto de enlaces que unen entre sí los términos, de ahí que se le conozca como *grafo relacional*.

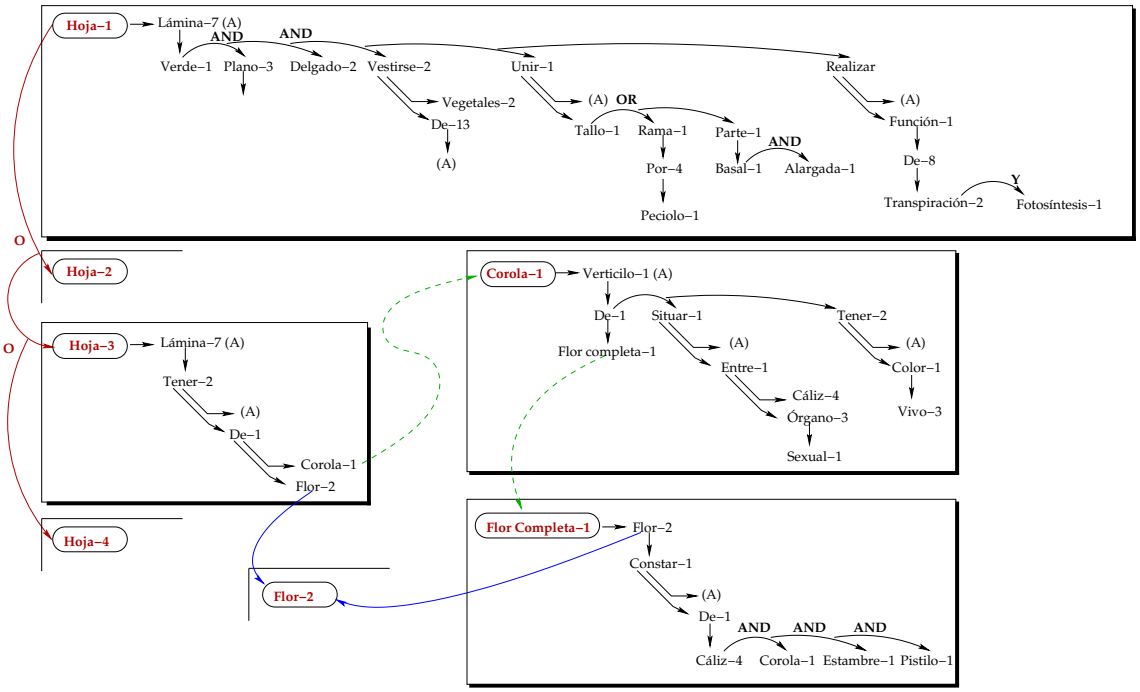


Figura D.1: Red semántica de Quillian para el plano de definición de *hoja* y *corola*

Tomando como punto de partida la Fig. D.1, vamos a ilustrar este tipo de red semántica, en el que se muestra dos de los diferentes sentidos de la palabra *hoja*, así como el de la palabra *corola*. Para ello, vamos a suponer las siguientes definiciones extraídas del diccionario de la RAE:

- *Hoja-1*: Cada una de las láminas verdes, planas y delgadas, de que se visten los vegetales, unidas al tallo o a las ramas por el peciolo o, por una parte basal alargada, en las que se realizan las funciones de transpiración y fotosíntesis.
- *Hoja-3*: Cada una de las láminas que tiene la corola de una flor.
- *Corola-1*: Verticilo de las flores completas, situado entre el cáliz y los órganos sexuales, y que tiene vivos colores

En este sentido, el conocimiento se organiza en *planos*, donde cada uno representa el grafo asociado a la acepción de una palabra. Además, se observa que los nodos encerrados en óvalos corresponden a los encabezamientos de las definiciones, es decir, *hoja* seguido de la acepción que ocupa en el diccionario, que puede ser *1,2,....* A estos nodos se les denomina *nodos-tipo*. Por ejemplo *Hoja-1* hace referencia a la primera acepción. De este modo se evitan ambigüedades en las definiciones, pues, por ejemplo *Hoja-1* hace referencia a la de los vegetales y *Hoja-5* al de los libros y cuadernos. Así, las palabras que aparecen en la propia definición, se les denomina *nodos-réplica* y estos a su vez serán nodos-tipo de su propia definición. Si observamos la definición de *Hoja-3*, éste posee un nodo-réplica *Corola-1* que a su vez es un nodo-tipo. Una vez definidos los nodos, es necesario indicar cuales son los tipos de relaciones que aparecen:

- *Subclase*. Une un nodo-tipo con la clase a la que pertenece. Por ejemplo, *Hoja-3* está unido con la clase *Lámina-7* y *Corola-1* con la clase *Verticilo-1*.
- *Disyunción*. Se usa mediante la etiqueta «OR», uniendo nodos entre sí. Por ejemplo, el enlace que une *Hoja-1* con *Hoja-2* y con *Hoja-3*, uniendo con las posibles interpretaciones de la palabra *Hoja*.
- *Conjunción*. Se usa mediante la etiqueta «AND», y también une nodos entre sí. Por ejemplo, el enlace que une *Verde-1* con *Plano-3* y con *Delgado-2*, une los dos nodo-réplicas con la subclase.
- *Propiedad*. Se usa para unir tres nodos, tal como se muestra en la Fig. D.2, donde *A* es la relación que se establece entre el sujeto, es decir, *B*, y el objeto, es decir, *C*. Por ejemplo, en la definición de *Hoja-1*, se unen el nodo-réplica *Realizar*, con el sujeto *A* y el objeto *Función-1*. En este caso, la variable *A* indica el concepto que aparece en el mismo plano de la definición, es decir, hace referencia a *Lámina-7*.
- *Referencia al tipo*. Estas referencias van siempre desde el nodo-réplica hasta el nodo-tipo, dándose siempre en planos diferentes. Por ejemplo, en la definición de

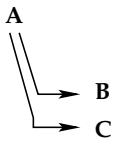


Figura D.2: Enlace de tipo «propiedad»

Hoja-3 aparece el término *Corola-1* que a través de su enlace lleva a su definición, en el plano adecuado, es decir, en la acepción correcta. En la Fig. D.1 se observa a través de la flecha punteada.

El programa creado por Quillian [239] usaba esta base de conocimiento con el fin de localizar relaciones entre pares de palabras. Dadas dos palabras, busca los grafos asociados a cada una de ellas. Puede ocurrir que exista en ambos grafos un nodo de concepto común, denominado *nodo intersección*. El camino a esos nodos intersección representa la relación entre los conceptos de esas palabras. Por ejemplo, en la Fig. D.1, el nodo intersección de los grafos asociados a las palabras *Hoja-3* y *Corola-1* es el nodo-réplica *Flor-2*, por lo que el camino que los une entre sí corresponde a la relación entre los significados de ambos conceptos.

Debido a la existencia de numerosos términos polisémicos, Quillian señaló la conveniencia de pasar de una representación de palabras a una representación de conceptos, sin depender de ningún idioma en particular. Esta idea dio lugar a una solución propuesta por Schank [280, 281], denominada *grafos de dependencias conceptuales*.

D.1.2 | Grafos de dependencias conceptuales de Schank

A diferencia de Quillian [239], Schank [280, 281] estaba interesado específicamente en la comprensión del LN, de ahí que sus perspectivas fueran diferentes y que quisiera representar los conceptos que se asocian a las palabras. Además, otra diferencia con Quillian era que las representaciones que creaba Schank trataban de ser independientes del idioma que se estuviera usando, lo que, en ese momento, no ocurría con Quillian.

Concretamente, este método consiste en representar cualquier frase mediante primitivas, que pueden ser de distintos tipos:

- *Categorías conceptuales*. Seis en total, indicando si es un objeto físico (PP), una acción (ACT), el atributo de un objeto (PA), el atributo de una acción (AA), tiempo (T) y localización (L).

- *Reglas sintácticas*. Dieciséis en total, determinando los diferentes tipos de relación que pueden existir entre los elementos de una frase. Entre otros se encuentran las relaciones *sujeto - verbo* (⟺), *objeto - verbo* ($\overset{o}{\longleftarrow}$), *posesión o parte-de* (⟵),

dirección(

), *recepción* (

), *causalidad* (), donde las flechas indican la dirección de las dependencias.

- *Acciones primitivas*. Indican el conjunto de acciones básicas que componen otras complejas. Es el caso:
  - PTRANS: Para transferir físicamente un objeto, es decir, cambiarlo de lugar, por ejemplo, «ir».
  - ATRANS: Para transferir una relación abstracta, como posesión o control, por ejemplo, «dar».
  - MTRANS: Para transferir información mentalmente, por ejemplo, «decir, contar, comunicar».
  - PROPEL: Es la aplicación de una fuerza física a un objeto, por ejemplo, «empujar».
  - MOVEL: El movimiento de una parte del cuerpo por su propietario, por ejemplo, «dar patadas».
  - GRASP: El acto por el que un actor coge un objeto, por ejemplo, «coger».
  - INGEST: Ingestión de un objeto por un ser animado, por ejemplo, «comer, ingerir».
  - CONC: La conceptualización o pensamiento de una idea por un actor.
  - EXPEL: Es la expulsión desde un cuerpo animado al exterior, por ejemplo, «llorar».
  - MBUILD: Es la construcción de una información a partir de una que existía, por ejemplo, «decidir».
  - ATTEND: Es la acción de dirigir un órgano de los sentidos hacia un objeto o estímulo, por ejemplo, «escuchar, mirar».
  - SPEAK: Es la acción de producir sonidos, por ejemplo, «hablar».

Estas primitivas se usan para definir *relaciones de dependencia conceptual* que describen el sentido de las estructuras semánticas. Estas relaciones de dependencia conceptual son las reglas de sintaxis y constituyen una auténtica guía para el establecimiento de las relaciones semánticas significativas. De este modo cada frase se descompone en elementos simples que pretenden ser independientes del idioma, utilizando estas relaciones luego para construir la representación interna de una frase. Para ilustrar estos conceptos, la Fig. D.3 muestra estas relaciones como un primer nivel de la construcción de la teoría, pero a partir a ellas se pueden obtener otras más complejas.

Esta teoría ofrece un número importante de beneficios. Al proporcionar una interpretación de la semántica del LN, reduce problemas de ambigüedad, limitándose a no proporcionar una forma canónica para el significado de las frases. Esto quiere decir

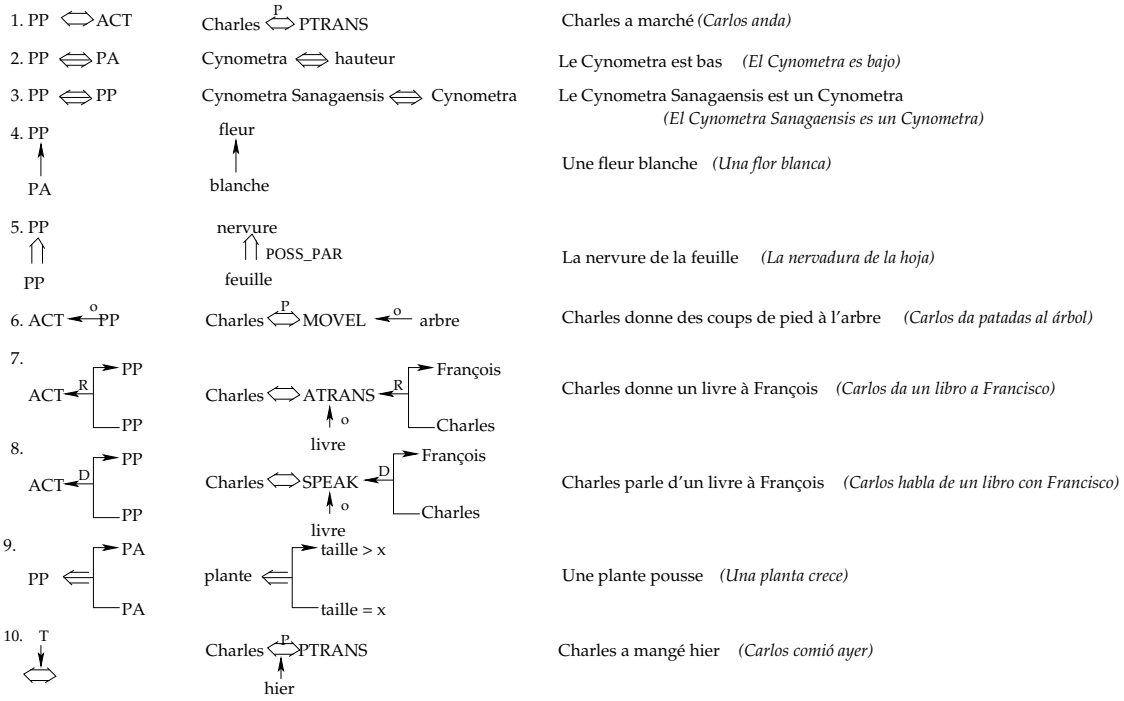


Figura D.3: Dependencias conceptuales básicas y uso más complejo

que sólo las frases con el mismo sentido se representarán sintácticamente de un mismo modo.

Otras ventajas tienen relación con la utilización de un conjunto limitado de primitivas. Éstas determinan unívocamente la representación del conocimiento, evitando una explosión combinatoria en el número de representaciones asociadas a cada frase. Al tiempo, al ser un método determinista y finito, se puede construir un intérprete capaz de realizar inferencias. Sin limitar el número de elementos y relaciones esto sería extremadamente difícil.

Sin embargo, el hecho de que este tipo de representación requiera una descripción demasiado detallada de las acciones representa una dificultad añadida, hasta el punto de que la descomposición puede resultar en extremo laboriosa. En este sentido, algunos autores, como podría ser Sowa [295], afirman que es más útil trabajar con distintos niveles de detalle y no con un conjunto cerrado de primitivas, de tal manera que se pueda explicitar los elementos cuando sea necesario. Por ejemplo, Schank [280, 281] sólo distingue entre seis tipos de categorías conceptuales. Concretamente, si nos centramos en la de objeto físico, Schank nos dirá que es de tipo PP, pero no podríamos hacer una distinción entre objetos móviles y objetos inmóviles, o incluso entre un objeto y un ser vivo. Por este motivo surgen otros tipos de representaciones tales como las que vamos a ver a continuación.

D.1.3 | Jerarquía de conceptos

Sin duda el tipo de red semántica por excelencia es el de *redes* ES-UN, de hecho, muchas veces se mencionan como sinónimo de red semántica. Esta red es una jerarquía taxonómica, es decir, es un entorno de clasificación compuesto por una jerarquía de clases anidadas, cuya espina dorsal está constituida por un sistema de enlaces de herencia entre los objetos o conceptos de representación, conocidos como nodos. Concretamente, este tipo de redes son el resultado de la observación de que gran parte del conocimiento humano está basado en la adscripción de un subconjunto de elementos como parte de otro más general. Las taxonomías clásicas naturales[1] son un buen ejemplo. De hecho, si quisiéramos representar en forma de LPO lo siguiente: «*un vitacola*[2] *es un Afzelia Africana, un Afzelia Africana es un Caesalpinioideae, un Caesalpinioideae es una Fabaceae, una Fabaceae es un vegetal*», quedaría del siguiente modo:

$$\forall x,\ (vitacola(x) \Rightarrow Afzelia\ Africana(x))$$
$$\forall x,\ (Afzelia\ Africana(x) \Rightarrow Caesalpinioideae(x))$$
$$\forall x,\ (Caesalpinioideae(x) \Rightarrow Fabaceae(x))$$
$$\forall x,\ (Fabaceae(x) \Rightarrow Vegetal(x))$$

Los nodos de las estructuras taxonómicas se han usado en multitud de representaciones [36], pero un hecho fundamental es la interpretación genérica o específica que se puede dar a los nodos, es decir, si éstos representan un único individuo o varios. Los nodos situados en lo más bajo de la jerarquía denotan individuos concretos o instancias, mientras que los nodos superiores denotan clases de individuos. En este sentido, un arco trazado desde un nodo *A* hacia un nodo *B* especifica que *A* es más general. Se trata de un GAD [166]. Un nodo puede tener varios ascendientes y descendientes, pero el descendiente de un nodo no se puede convertir en su ascendiente mediante un ciclo.

Concretamente, si observamos la Fig. D.4, se muestra como el concepto superior, el que engloba a todos los demás, se representa por «T». Los arcos representan relaciones de orden parcial. Es el caso del arco que va de *Vivo* a *Vegetal*, donde *Vegetal* $\subseteq$ *Vivo*. Es decir, *Vivo* es un concepto más general que *Vegetal*. Como se ha dicho, en este grafo se pueden obtener bucles y no ciclos, como ocurre con *Objeto físico*, *No vivo*, *Vivo*, *Móvil* y *Animal*.

El interés de agrupar los conceptos en una red jerárquica tiene como finalidad poder realizar un tipo de inferencia que permita que un concepto herede las propiedades de sus antepasados. Concretamente, la inferencia mediante herencia de propiedades consiste en aplicar una cadena de silogismos extraídos de la lógica clásica: *«Si X es un vegetal, los*

[1] son aquéllas que agrupan los seres vivos en función de determinadas características comunes y hereditarias. Para saber más acerca de las taxonomías botánicas, consultar el apéndice A.

[2] es el nombre común de la especie Afzelia Africana.

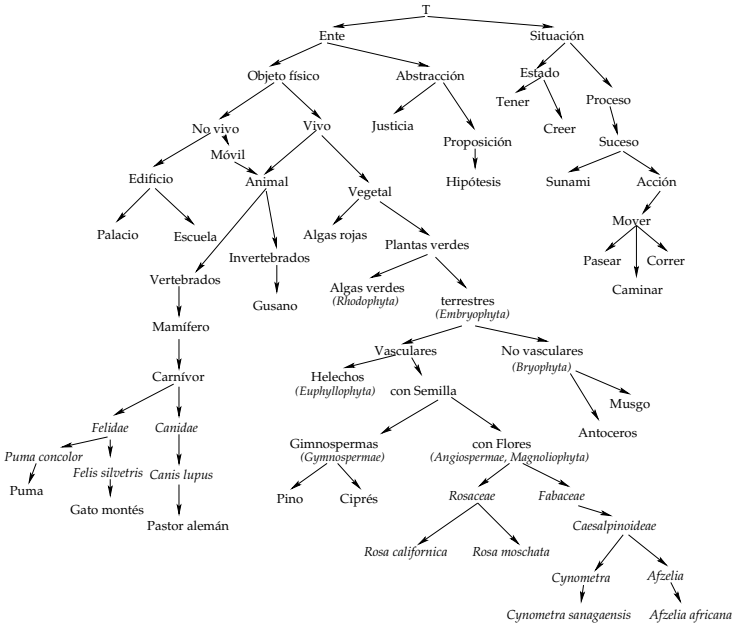


Figura D.4: Jerarquía de conceptos

vegetales son seres vivos, y los seres vivos son objetos físicos, entonces X es objeto físico». Este tipo de jerarquía permite a una categoría concreta añadir propiedades específicas a la misma, heredando las demás de las categorías superiores. Si nos centráramos única y exclusivamente en la clase *Caesalpinioideae*, las propiedades podrían ser las que se observan en la Fig. D.5, que serán heredadas por las categorías inferiores que dependan de la clase. Pero a su vez la clase *Caesalpinioideae* podría haber heredado propiedades de categorías superiores tales como *Fabaceae*.

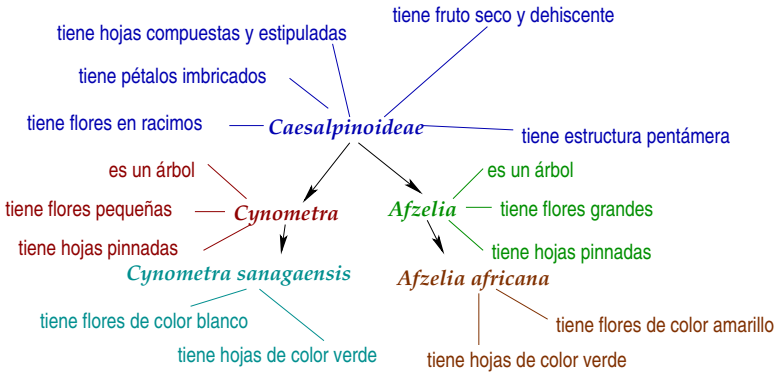


Figura D.5: Propiedades en jerarquía de conceptos

Concretamente, existen dos tipos de herencia en redes jerárquicas. La *herencia estricta* es aquélla en la que todos los conceptos descendientes de una clase poseen sus mismas propiedades. La *herencia por defecto* supone que los descendientes de una clase poseen sus mismas propiedades mientras no se indique lo contrario. Esta última se verá con

más detenimiento en la siguiente sección. De hecho, la posibilidad de trabajar con dos tipos de herencia plantea un problema al trabajar con grafos dirigidos acíclicos, ya que el hecho de que un nodo pueda tener distintos padres hace que puedan surgir contradicciones entre los diferentes valores por defecto heredados. De ahí surge la necesidad de establecer mecanismos para resolver estos conflictos [304, 308].

D.2 | Marcos

Los marcos fueron propuestos inicialmente por Minsky [206], considerando la resolución de los problemas humanos como el proceso de rellenar huecos de descripciones de la mente y usándolos para representar dicho conocimiento mediante el rellenado de esos espacios vacíos [289]. En este sentido, fueron propuestos para superar las limitaciones de la lógica a la hora de abordar problemas como la visión artificial [126], la comprensión del LN [82] o el razonamiento basado en el sentido común [82]. Los marcos son, de hecho, una evolución de las redes semánticas donde el nodo es sustituido por una estructura de datos que representa una situación estereotipada a partir de sus elementos más significativos.

Concretamente, los marcos se introducen en [247] como una colección de *ranuras o casillas* donde se almacena la información respecto a su uso y a lo que se espera que ocurra a continuación. Cada casilla contiene la información sobre un atributo particular del objeto que se modela o una operación del marco. En muchos aspectos, un marco se podría identificar con los objetos estructurados de los lenguajes imperativos.

En este sentido, las casillas asocian información, que puede ser de tipos diferentes, y que denominamos *facetas*. Las facetas son un modo de proporcionar conocimiento extendido acerca de un atributo. Cada una puede contener un valor por defecto o un puntero a otro marco, llamado *submarco* del propio marco; un conjunto de reglas o un procedimiento con el que se obtendrá el valor de la misma, tal y como podemos ilustrar a partir de la Fig. D.6. A continuación, haremos referencia a cada uno de los componentes de los marcos, refiriéndonos a ellos mediante ejemplos.

- *Nombre del marco*. Un ejemplo de nombre de clase de la Fig. D.6, sería *Género Cynometra*.

- *Relaciones de un marco con otro*: En la Fig. D.6, el marco *Recolectada 1* es un ejemplar de la clase *Especie Afzelia Africana*, el cual a su vez pertenece a la clase *Género Afzelia*.

- *Valor de la casilla*. El valor de una casilla puede ser simbólico, numérico o booleano. Por ejemplo en el marco de la Fig. D.6, la casilla *Prefloración* de la clase *Subfamilia Caesalpinioideae* tiene un valor simbólico *imbricada* y la casilla

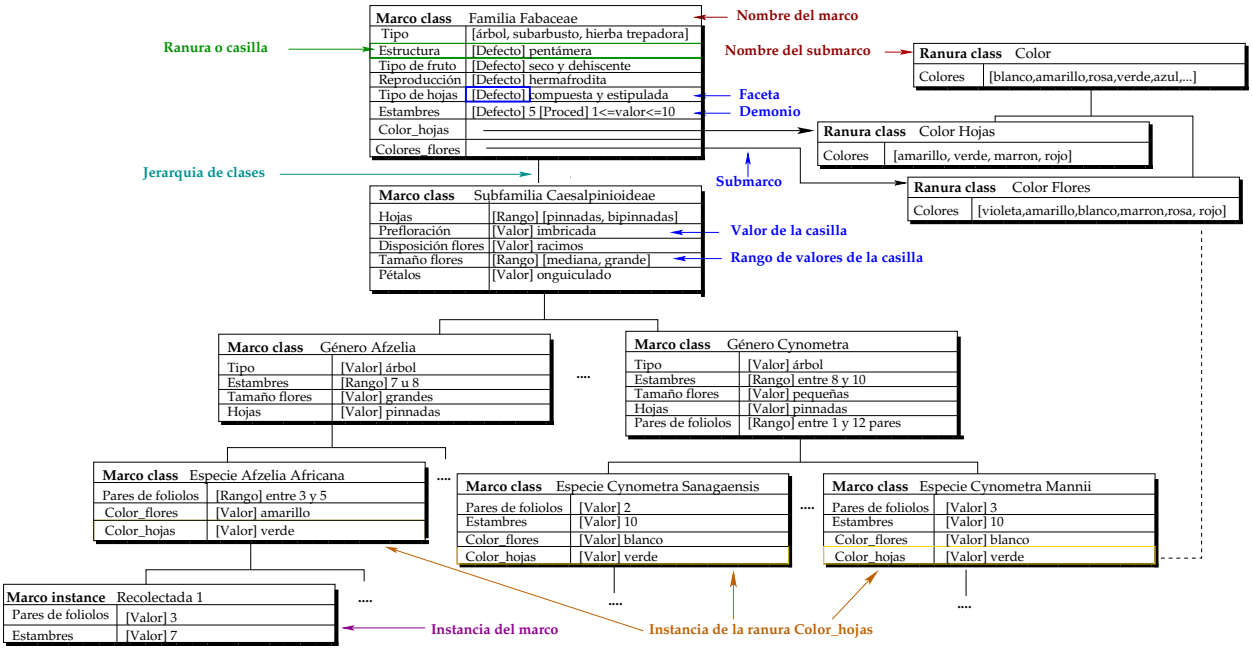


Figura D.6: Ejemplo de sistema de marcos simplificado

Pares de foliolos de la instancia *Recolectada 1* un valor numérico *3*. Estos valores se pueden asignar cuando se crea el marco, o más tarde.

- *Valor por defecto de la casilla*. El valor por defecto se toma cuando no hay evidencias de lo contrario. Por ejemplo, un marco *Familia Fabaceae* tiene una *estructura pentámera* como valor por defecto en su correspondiente casilla. Las clases que heredan de ésta, si no se indica lo contrario, también tendrán una estructura *pentámera*.

- *Rango de los valores de la casilla*. El rango va a determinar si un objeto en particular encaja con los requerimientos estereotipados definidos por el marco. Por ejemplo, las *hojas* de la *subfamilia Caesalpinioideae* podría ser considerado entre el rango de valores *pinada y bipinada*.

- *Información procedimental*. Una casilla puede tener asignado un procedimiento, el cuál se ejecutará si el valor de la casilla cambió o si, en cambio, se necesita para comprobar algún otro valor de otra casilla. A estos procedimientos anexados a la casilla se les denominan *demonios*. Por ejemplo, la casilla *Estambres* del marco *familia Fabaceae*, en la Fig. D.6, tiene por defecto un valor *5*, pero también posee un demonio que se activa cuando, en los marcos heredados, ese valor cambia. El demonio será el encargado de ejecutar un procedimiento que en este caso verifique que el nuevo valor que se le asigne se encuentre entre *1 y 10*.

La colección de marcos interconectados entre sí forma un *sistema de marcos*, es decir, una red de estructuras de datos y relaciones [225], donde los marcos de los niveles

superiores[3] dan una visión más general de la información manejada por el sistema. Los marcos de los niveles inferiores poseen muchas casillas que deben rellenarse mediante instancias específicas o datos [42]. Por ejemplo, en la Fig. D.6, el marco *Especie Cynometra Mannii* es el nivel inferior antes de definir una instancia concreta, y posee, además de las casillas de este marco[4], todas aquéllas cuyo valor se haya definido en niveles superiores. Es el caso de la casilla *hojas* del marco *Género Cynometra* cuyo valor es *pinada*. El marco *Especie Cynometra Mannii* heredará este valor. A diferencia de las redes semánticas, se pueden definir casillas sin valor en las clases, como ocurre en el marco *Familia Fabaceae* de la Fig. D.6 donde *tipo* no tiene un valor concreto. Este valor se rellena en las subclases o incluso en las instancias.

Además, una casilla puede tener asignado un objeto de valor suficiente, como en el caso del marco *Familia Fabaceae* de la Fig. D.6 para la casilla *estructura*. Pero también puede especificar varios, debiendo satisfacer cada una de sus asignaciones. En este sentido, estas asignaciones pueden delimitar submarcos de cierto tipo mediante la utilización de punteros. Es lo que ocurre con la casilla *color hojas* y *color flores* del marco *Familia Fabaceae*, cada uno apunta a un submarco que posee el mismo nombre y que hereda del submarco *colores*. Otras condiciones más complejas pueden especificar relaciones entre los objetos asignados a diferentes campos.

Una vez que se ha establecido la colección de marcos y se han interconectado entre sí, estamos en disposición de crear los objetos concretos que hacen referencia a esas situaciones estereotipadas. Concretamente, existen instancias de marcos que asignan ejemplares a las clases y marcos de clase que describen clases completas. La relación «ES-UN», abreviatura de «es miembro de la clase», asigna instancias a las clases de las que son miembros. Por ejemplo, la instancia *Recolectada 1* «ES-UN» *Especie Afzelia Africana*, es decir, *Recolectada 1* es un miembro de la clase. Otra relación es «TIPO-DE», que vincula clases entre sí. Esto implica que si una superclase tiene una relación, entonces el ejemplar la hereda. Es el caso del marco *Género Cynometra* que posee una relación «TIPO-DE» con *Subfamilia Caesalpinioideae*. De este modo, hereda los atributos de éste último siempre que no se redefinan en *Género Cynometra*. Lo mismo ocurre entre *Género Cynometra* y *Subfamilia Caesalpinioideae*.

Una vez explicada la sintaxis de los marcos, y partiendo de la Fig. D.6, se puede interpretar cuales son las características asociadas a cada concepto y las relaciones que se establecen entre ellos en nuestro ejemplo de trabajo. Por ejemplo, sabemos que la *Familia Fabaceae* es un tipo de *árbol, subarbusto o hierba trepadora* y que generalmente su tipo de *reproducción* es *hermafrodita*; que el *Género Cynometra* es una *Caesalpinioideae* cuya cantidad de *estambres* se sitúa entre *8 y 10*; que la *especie Cynometra Sanagaensis* es de tipo *género Cynometra*, que generalmente tienen *flores de tamaño pequeño*. La planta *Recolectada 1* es una *especie Afzelia Africana* con *7 estambres*, *3 pares de foliolos* y cuyas flores son de color *amarillo*.

[3] por ejemplo, en la Fig. D.6, el marco *Familia Fabaceae*.
[4] es decir, *pares de foliolos* y *estambres.*

De todo lo anterior podemos deducir que una base de conocimiento basada en marcos es una colección organizada jerárquicamente, según un número de criterios estrictos y otros principios más o menos imprecisos tales como el de similitud. A nivel práctico, los marcos poseen mayores posibilidades que las redes semánticas, en particular, en lo referente a:

- *Precisión*. Se precisan los objetos, las relaciones entre objetos y sus propiedades; en ausencia de evidencia contraria se usan valores por omisión. Es decir todos las propiedades especificadas en categorías superiores tienen especificado un valor. Y esos valores serán los que se tomen, si no se especifica lo contrario, en las categorías inferiores.

- *Sobrecontrol*. Para cada nodo hijo, el enlace con el nodo padre es un enlace de herencia. El nodo hijo hereda todos las casillas de su padre a menos que se especifique lo contrario. Por ejemplo, la *subfamilia Caesalpinioideae* hereda de la *familia Fabaceae* el *tipo de hojas* que tiene, es decir, *compuesta y estipulada*. Pero a su vez, el *género Cynometra* lo hereda de la *subfamilia Caesalpinioideae*.

- *La herencia por defecto es no monotónica*. Debido al sobrecontrol, no hay posibilidad de negar la herencia por defecto de propiedades en un contexto o situación determinada. Esta es una gran diferencia con las redes semánticas, donde la herencia es siempre monotónica. Por ejemplo, la *especie Afzelia Africana* al ser un marco que hereda del *género Afzelia* y no tener definido un valor para la propiedad *estambres*, por herencia de propiedades por defecto, esta propiedad toma el valor especificado en el marco *familia Fabaceae*. Esto es, la cantidad de estambres será de *5*. Por el contrario, la instancia *Recolectada-1*, a pesar de ser también una instancia de la *especie Afzelia Africana*, no hereda esta propiedad por defecto, pues tiene definido que su cantidad de estambres es de *7*.

- *Activación dinámica de procesos*. Se pueden adjuntar procedimientos a un marco o a alguno de sus componentes y ser llamados y ejecutados automáticamente tras la comprobación de cambio de alguna propiedad o valor. Es el caso de la *familia Fabaceae*, donde se activa dinámicamente un proceso para comprobar que la cantidad de estambres de sus categorías inferiores se encuentran entre 1 y 10.

- *Modularidad*. La base de conocimiento está organizada en componentes claramente diferenciados. Los nodos pueden ser de dos tipos: *nodos de clase*[5], como por ejemplo, la *especie Cynometra Sanagaensis*, y *nodos de instancia*[6], como por ejemplo la instancia *Recolectada-1*. Todos los nodos internos, no terminales, han de ser nodos de clase.

[5]hacen referencia a conceptos por especificar.
[6]hacen referencia a objetos concretos.

El potencial de estas estructuras se manifiesta en los procesos de razonamiento que son capaces de llevar a cabo. Así, éstos aplican dos mecanismos básicos: el reconocimiento de patrones y la herencia. El reconocimiento de patrones se centra en encontrar el lugar más apropiado para un nuevo marco dentro de la jerarquía completa. Esto requiere que el mecanismo de reconocimiento sea capaz de recibir información sobre la situación existente y lleve a cabo una búsqueda de aquél más adecuado de entre todos los contenidos en la base de conocimiento. En este sentido, al contrario que las reglas o las representaciones lógicas, los marcos son unidades de almacenamiento suficientemente grandes como para imponer una estructura en el análisis de una situación.

Pero además de este potencial, los marcos aportan un tipo de razonamiento que no se consigue a través de la lógica. Se trata del razonamiento por defecto y hace referencia a cierto tipo de deducciones usando valores heredados. Posiblemente, estas deducciones se deban eliminar cuando se tenga más información. Esto ocurriría por ejemplo, sobre la base de la Fig. D.6 si se quisieran crear instancias directamente del marco *Familia Fabaceae*, donde por defecto el número de *estambres* es de *5*. Cuando se hable de la *Especie Afzelia Africana* el razonamiento deductivo se habrá obtenido usando el valor por defecto, puesto que en este caso, el número de *estambres* oscilará entre *7 y 8*.

A modo de resumen y centrándonos en nuestro contexto botánico, cuando lo que se pretende es adquirir conocimiento de un modo automático sobre un dominio específico, procurando que el usuario no tenga que inmiscuirse en su realización, se busca un modo de representación que a su vez no necesite de especificaciones previas. Concretamente, en el caso de los marcos esto no ocurre ya que se tienen que definir situaciones estereotipadas. En este sentido, se parte de la base de que esas situaciones son conocidas, por lo que entra en total contradicción con el tipo de sistema que se está planteando.

En nuestro caso particular, realmente no existe un impedimento para crear las clases asociadas a la jerarquía de los marcos, sin embargo resulta difícil describir instancias concretas asociadas a ellas debido a la complejidad que esto supondría. Hay que recordar que nuestro *corpus* trata de describir conjuntos de plantas y no individuos concretos. De hecho, el caso expuesto en la Fig. D.6, es decir, la instancia *Recolectada 1* es una descripción concreta de una *Afzelia Africana*, que no se encuentra descrita en el *corpus*, por lo que se deberían crear tantas instancias como posibilidades hubiera. Siguiendo con el ejemplo, sería necesario crear tantas instancias como *pares de foliolos* se permitiesen, es decir, (*entre 3 y 5*), pero también tantos como posibilidades de *estambres* hubiese (heredada de *Afzelia*, *7 u 8*). En definitiva, sería extremadamente complejo representar el conocimiento mediante esta técnica.

Bibliografía

[1] Anne Abeillé. Parsing french with tree adjoining grammar: some linguistic accounts, 1988.

[2] Anne Abeillé. *Une grammaire lexicalisée d'Arbres adjoints pour le Français: Application à l'analyse automatique*. PhD thesis, Université Paris 7, Paris, France, 1991.

[3] Steven Abney. Partial parsing via finite-state cascades. *Nat. Lang. Eng.*, 2:337–344, December 1996.

[4] Steven Abney and Steven P. Abney. Parsing by chunks. In *Principle-Based Parsing*, pages 257–278. Kluwer Academic Publishers, 1991.

[5] S. Agne, A. Dengel, and B. Klein. Evaluating see - a benchmarking system for document page segmentation. In *ICDAR '03: Proceedings of the Seventh International Conference on Document Analysis and Recognition*, page 634, Washington, DC, USA, 2003. IEEE Computer Society.

[6] S. Agne, M. Rogger, and J. Rohrschneider. Benchmarking of document page segmentation. In Daniel P. Lopresti; Jiangying Zhou, editor, *Document and Recognition and Retrieval VII. January 26-27, San Jose,, CA, United States*, volume 3967 of *Proceedings of SPIE*, pages 165–171. SPIE- International Society for Optical Engineering, 2000.

[7] Alfred V. Aho and Jeffrey D. Ullman. *The theory of parsing, translation, and compiling*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1972.

[8] A. Al-Maskari, M. Sanderson, and P. Clough. The relationship between IR effectiveness measures and user satisfaction. In *Proc. of the 30th Int. Conf. on Research and Development in Information Retrieval*, SIGIR'07, pages 773–774, New York, NY, USA, 2007. ACM.

[9] Miguel A. Alonso and Víctor J. Díaz. Variants of mixed parsing of tag and tig. In *Proceedings of TALN'03*, pages 41–65, Dourdan, France, 2003.

[10] Miguel A. Alonso, Jesús Vilares Ferro, and Victor M. Darriba. On the usefulness of extracting syntactic dependencies for text indexing. In *Proceedings of the 13th Irish International Conference on Artificial Intelligence and Cognitive Science*, AICS '02, pages 3–11, London, UK, 2002. Springer-Verlag.

[11] Miguel A. Alonso Pardo. *Interpretación tabular de autómatas para lenguajes de adjunción de árboles*. PhD thesis, Departamento de Computación, Universidade da Coruña, A Coruña, Spain, September 2000.

[12] Miguel A. Alonso Pardo, David Cabrero Souto, Manuel Vilares, and Éric Villemonte de La Clergerie. Tabular algorithms for TAG parsing. In *Proc. of EACL'99*, 1999.

[13] Miguel A. Alonso Pardo, Vicente Carrillo, and Víctor J. Díaz. Análisis sintáctico combinado de gramáticas de adjunción de árboles y de gramáticas de inserción de árboles. *Procesamiento del Lenguaje Natural*, 29:65–72, 2002.

[14] G. Amati and C.J. Van Rijsbergen. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transanctions on Information Systems*, 20(4):357–389, October 2002.

[15] J.A. Aslam, E. Yilmaz, and V Pavlu. A geometric interpretation of R-precision and its correlation with average precision. In *Proc. of the 28th Int. Conf. on Research and Development in Information Retrieval*, SIGIR'05, pages 573–574, New York, NY, USA, 2005. ACM.

[16] J. Attenberg and T. Suel. Cleaning search results using term distance features. In *Proc. of the 4th Int. Workshop on Adversarial Information Retrieval on the Web*, AIRWeb '08, pages 21–24, New York, NY, USA, 2008. ACM.

[17] T. Galen Ault and Y. Yang. Information filtering in trec-9 and tdt-3: A comparative analysis. *Information Retrieval*, 5:159–187, April 2002.

[18] Edwige Fangseu Badjio. Traitement de corpus botaniques. Dea, DEA CHM, Université du Mans, September 2002.

[19] R. Baeza-Yates, F. Saint-Jean, and C. Castillo. Web structure, dynamics and page quality. In *Proc. of 9th Int. Symposium on String Processing and Information Retrieval*, volume 2476 of *SPIRE'02*, pages 117–132, Lisbon, Portugal, 2002. Springer.

[20] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley, May 1999.

[21] Jean-François Baget and Marie-Laure Mugnier. Extensions of simple conceptual graphs: the complexity of rules and constraints. *J. Artif. Int. Res.*, 16(1):425–465, 2002.

[22] S. Bani-Ahmad and G. Ozsoyoglu. On popularity quality: growth and decay phases of publication popularities. In *Proc. of the 6th Int. Conf. on Innovations in Information Technology*, IIT'09, pages 231–235, Piscataway, NJ, USA, 2009. IEEE Press.

[23] Nicolas Barrier. Une métagrammaire pour les adjectifs du français. In *Proc. of TALN'06 (poster)*, pages 351–357, 2006.

[24] François Barthélemy, Pierre Boullier, Philippe Deschamp, Linda Kaouane, Abdelaziz Khajour, and Éric Villemonte de La Clergerie. Tools and resources for tree adjoining grammars. In *Proceedings of ACL'01 workshop on Sharing Tools and Resources*, pages 63–70, Toulouse, France, July 2001.

[25] Roberto Basili, Maria Teresa Pazienza, and Paola Velardi. Acquisition of selectional patterns in sublanguages. *Machine Translation*, 8(3):175–201, 1993.

[26] T. Bayes. An essay towards solving a problem in the doctrine of chances. *Phil. Trans. of the Royal Soc. of London*, 53:370–418, 1763.

[27] Mustapha Baziz. *Indexation contextuelle guidée par ontologie pour la recherche d'information*. PhD thesis, Institut de Recherche en Informatique de Toulouse, December 2005.

[28] A. Belaïd and H. Cecotti. Reconnaissance de caractères : évaluation des performances. In Rémy Mullot, editor, *Les documents écrits: de la numérisation à l'indexation par le contenu Traité IC2, série informatique et systèmes d'information*, Traité IC2, série informatique et systèmes d'information. HERMES, 2006. J.: Computer Applications.

[29] A. Belaïd, L. Pierron, Laurent Najman, and D. Reyren. *Bibliothèques numériques*, chapter La numérisation de documents: le point de vue de l'informaticien face à l'industriel, pages 53–98. ADBS editions, 2000.

[30] P. A. Bensch and Walter J. Savitch. An occurrence-based model of word categorization. *Ann. Math. Artif. Intell.*, 14(1), 1995.

[31] Patrick Blackburn, Maarten de Rijke, and Yde Venema. *Modal Logic*. Cambridge University Press, 2001.

[32] D. Bollegala, N. Noman, and H. Iba. Rankde: learning a ranking function for information retrieval using differential evolution. In *Proc. of the 13th Annual Conf. on Genetic and Evolutionary Computation*, GECCO'11, pages 1771–1778, New York, NY, USA, 2011. ACM.

[33] Didier Bourigault. Surface grammatical analysis for the extraction of terminological noun phrases. In *Proceedings of the 14th conference on*

Computational linguistics - Volume 3, COLING '92, pages 977–981, Stroudsburg, PA, USA, 1992. Association for Computational Linguistics.

[34] Didier Bourigault. LEXTER, a terminology extraction software for knowledge acquisition from texts. *In Proceedings of the 9th knowledge acquisition for knowledge based system workshop (KAW'95)*, 1995.

[35] Didier Bourigault. LEXTER, a natural language tool for terminology extraction. *In proceeding of the seventh EURALEX international congress*, pages 771–779, 1996.

[36] R. J. Brachman. What is-a is and isn't: An analysis of taxonomic links in semantic networks. *Computer*, 16(10):30–36, 1983.

[37] C. Buckley and E.M. Voorhees. Evaluating evaluation measure stability. In *Proc. of the 23rd Int. Conf. on Research and Development in Information Retrieval*, SIGIR'00, pages 33–40, New York, NY, USA, 2000. ACM.

[38] C. Buckley and E.M. Voorhees. Retrieval evaluation with incomplete information. In *Proc. of the 27th Int. Conf. on Research and Development in Information Retrieval*, SIGIR'04, pages 25–32, New York, NY, USA, 2004. ACM.

[39] Chris Buckley, Darrin Dimmick, Ian Soboroff, and Ellen M. Voorhees. Bias and the limits of pooling. In *In* Proc. of the 29th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, *SIGIR'06*, pages 619–620, 2006.

[40] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *Proc. of the 22nd Int. Conf. on Machine learning*, ICML'05, pages 89–96, New York, NY, USA, 2005. ACM.

[41] C.J.C. Burges, R. Ragno, and Q. Viet Le. Learning to rank with nonsmooth cost functions. In B. Schölkopf, J.C. Platt, and T. Hoffman, editors, *Proc. of the 20th Annual Conf. on Neural Information Processing Systems*, volume 19, pages 193–200. MIT Press, 2006.

[42] C. Burkert. Lexical semantics and terminological knowledge representation. In *Computational lexical semantics*, pages 165–184, Cambridge, 1995. Cambridge University Press.

[43] David Cabrero. *Análisis eficaz de gramáticas de cláusulas definidas*. PhD thesis, Departamento de Computación, Universidade da Coruña, A Coruña, Spain, Sep 2002.

[44] M. Candito. *Organisation modulaire et paramétrable de grammaires électroniques lexicalisées*. PhD thesis, Université Paris 7, January 1999.

[45] Y. Cao, J. Xu, T.-Y. Liu, H. Li, Y. Huang, and H.-W. Hon. Adapting ranking SVM to document retrieval. In *Proc. of the 29th Annual Int. Conf. on Research and Development in Information Retrieval*, SIGIR'06, pages 186–193, New York, NY, USA, 2006. ACM.

[46] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li. Learning to rank: from pairwise approach to listwise approach. In *Proc. of the 24th Int. Conf. on Machine learning*, ICML'07, pages 129–136, New York, NY, USA, 2007. ACM.

[47] D. Carmel, H. Roitman, and E. Yom-Tov. On the relationship between novelty and popularity of user-generated content. In *Proc. of the 19th Int. Conf. on Information and knowledge Management*, CIKM'10, pages 1509–1512, New York, NY, USA, 2010. ACM.

[48] D. Carmel and E. Yom-Tov. *Estimating the Query Difficulty for Information Retrieval*. Synthesis Lectures on Information Concepts, Retrieval, and Services. Morgan & Claypool Publishers, 2010.

[49] V. Carrillo, V. J. Díaz, and M. A. Alonso. Algoritmos de análisis para gramáticas de inserción de árboles. *Procesamiento del Lenguaje Natural*, 29:89–96, 2002.

[50] Vicente Carrillo Montero, Víctor Jesús Díaz Madrigal, and Miguel Toro Bonilla. Un recorrido por los formalismos gramaticales lexicalizados basados en reescritura de Árboles. In *Novatica: Lengua y Tecnologías de la Información, vol 133*, pages 22–25, 1998.

[51] B. Carterette and P.N. Bennett. Evaluation measures for preference judgments. In *Proc. of the 31st Int. Conf. on Research and Development in Information Retrieval*, SIGIR'08, pages 685–686, New York, NY, USA, 2008. ACM.

[52] B. Carterette, V. Pavlu, E. Kanoulas, J.A. Aslam, and J. Allan. Evaluation over thousands of queries. In *Proc. of the 31st Int. Conf. on Research and Development in Information Retrieval*, SIGIR'08, pages 651–658, New York, NY, USA, 2008. ACM.

[53] C. Castillo and B.D. Davison. Adversarial web search. *Foundations and Trends in Information Retrieval*, 4(5):377–486, May 2011.

[54] C. Castillo, D. Donato, and A. Gionis. Estimating number of citations using author reputation. In *Proc. of 14th Int. Symposium on String Processing and Information Retrieval*, SPIRE'07, pages 107–117, Berlin, Heidelberg, 2007. Springer-Verlag.

[55] Michel Chein and Marie laure Mugnier. Conceptual graphs: fundamental notions. *Revue d'Intelligence Artificielle*, 6:365–406, 1992.

[56] Michel Chein and Marie-Laure Mugnier. *Graph-based Knowledge Representation: Computational Foundations of Conceptual Graphs*. Springer, London, 2008.

[57] Jean-Pierre Chevallet, Joo-Hwee Lim, and Diem Thi Hoang Le. Domain knowledge conceptual inter-media indexing: application to multilingual multimedia medical reports. In *CIKM*, pages 495–504, 2007.

[58] Bong-Hyun Cho, Changki Lee, and Gary Geunbae Lee. Exploring term dependences in probabilistic information retrieval model. *Inf. Process. Manage.*, 39:505–519, July 2003.

[59] J. Cho and S. Roy. Impact of search engines on page popularity. In *Proc. of the 13th Int. Conf. on World Wide Web*, WWW'04, pages 20–29, New York, NY, USA, 2004. ACM.

[60] J. Cho, S. Roy, and R.E. Adams. Page quality: in search of an unbiased web ranking. In *Proc. of the 24th Int. Conf. on Management of Data*, SIGMOD'05, pages 551–562, New York, NY, USA, 2005. ACM.

[61] Noam Chomsky. *Aspects of the theory of syntax*. Massachusetts Institute of Technology (Cambridge, Mass.). Research Laboratory of Electronics. Special technical report ; 11. Mass. Inst. of Techn. Pr, Cambridge Mass., 1969. 10, 251 S.

[62] Kenneth Ward Church and Patrick Hanks. Word association norms, mutual information, and lexicography. *Comput. Linguist.*, 16:22–29, March 1990.

[63] C. Cleverdon, J. Mills, and E.M. Keen. An inquiry in testing of information retrieval systems. 1966.

[64] C.W. Cleverdon. The significance of the Cranfield tests on index languages. In *Proc. of the 14th Int. Conf. on Research and Development in Information Retrieval*, SIGIR'91, pages 3–12, New York, NY, USA, 1991. ACM.

[65] Cyril Cleverdon. *The Cranfield tests on index language devices*, pages 47–59. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1997.

[66] Lionel Clément, Benoît Sagot, and Bernard Lang. Morphology based automatic acquisition of large-coverage lexica. In *proc. of LREC'04*, pages 1841–1844, May 2004.

[67] E.F. Codd. A relational model of data for large shared data banks. *Communications of the ACM*, 26:64–69, January 1983.

[68] W.W. Cohen, A. Borgida, and H. Hirsh. Computing least common subsumers in description logics. In *Proc. of the Tenth Int. Conf. on Artificial intelligence*, AAAI'92, pages 754–760. AAAI Press, 1992.

[69] W.W. Cohen and Y. Singer. Context-sensitive learning methods for text categorization. *ACM Transactions on Information Systems*, 17(2):141–173, 1999.

[70] Olivier Corby. Web, graphs and semantics. In *Proceedings of the 16th international conference on Conceptual Structures: Knowledge Visualization and Reasoning*, ICCS '08, pages 43–61, Berlin, Heidelberg, 2008. Springer-Verlag.

[71] Rene Cori and Daniel Lasca. *Mathematical Logic: A Course With Exercises-Propositional Calculus, Boolean Algebras, Predicate Calculus*. Oxford University Press, 2000.

[72] G. V. Cormack, C. L. A. Clarke, C. R. Palmer, and D. I. E. Kisman. Fast automatic passage ranking (multitext experiments for trec-8). In *In Voorhees and Harman [21*, pages 735–742, 1999.

[73] Michael A. Covington. *Natural Language Processing for Prolog Programmers*. Prentice-Hall, Englewood Cliffs, NJ, 1994.

[74] Carlos A. Cuadra and Robert B. Katter. Opening the Black Box of Relevance. *Journal of Documentation*, 23(4):291–303, 1993.

[75] R. Cummins and C. O'Riordan. Term-weighting in information retrieval using genetic programming: A three stage process. In *Proc. of the 17th European Conf. on Artificial Intelligence*, ECAI'06, pages 793–794, Amsterdam, The Netherlands, The Netherlands, 2006. IOS Press.

[76] K. Curran, C. Murphy, and S. Annesley. Intelligent information retrieval. *Int. Journal of Advanced Media and Communication*, 1(2):139–147, 2006.

[77] Bourigault D. and Fabre C. Approche linguistique pour l'analyse syntaxique de corpus, 2000.

[78] B. Daille and E. Morin. Reconnaissance automatique des noms propres de la langue écrite: les récentes réalisations. In *Traitement automatique des langues, vol. 41, no 3 (196 p.) (1 p.3/4)*, pages 601–621. Association pour le traitement automatique des langues, Paris, FRANCE (1993) (Revue), 2000.

[79] Béatrice Daille, Éric Gaussier, and Jean-Marc Langé. Towards automatic extraction of monolingual and bilingual terminology. In *COLING*, pages 515–524, 1994.

[80] Sophie David and Pierre Plante. De la nécessité d'une approche morpho-syntaxique en analyse de textes. *Intelligence Artificielle et Sciences Cognitives au Québec*, 2(3):140–155, September 1990.

[81] H.M. de Almeida, M.A. Gonçalves, M. Cristo, and P. Calado. A combined component approach for finding collection-adapted ranking functions based on genetic programming. In *Proc. of the 30th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, SIGIR'07, pages 399–406, New York, NY, USA, 2007. ACM.

[82] Ana Esperanza Delgado García, Francisco Javier Díez Vegas, Jesús González Boticario, and José Mira Mira. *Aspectos básicos de la inteligencia artificial*, volume 1. Sanz y Torres, 1995.

[83] Victor Jesús Diaz Madrigal. *Gramáticas de adjunción de árboles: Un enfoque deductivo en el análisis sintáctico*. PhD thesis, Departamento de Lenguajes y Sistemas Informáticos de Sevilla, Sevilla, Spain, June 2000.

[84] L.R. Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302, July 1945.

[85] Sandor Dominich. *The Modern Algebra of Information Retrieval (The Information Retrieval Series)*. Springer, 1 edition, April 2008.

[86] A. Dong, Y. Chang, Z. Zheng, G. Mishne, J. Bai, R. Zhang, K. Buchner, C. Liao, and F. Diaz. Towards recency ranking in web search. In *Proc. of the Third ACM Int. Conf. on Web Search and Data Mining*, WSDM'10, pages 11–20, New York, NY, USA, 2010. ACM.

[87] Y. Duan, L. Jiang, T. Qin, M. Zhou, and H.-Y. Shum. An empirical study on learning to rank of tweets. In *Proc. of the 23rd Int. Conf. on Computational Linguistics*, COLING'10, pages 295–303, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[88] Jay Earley. An efficient context-free parsing algorithm. *Commun. ACM*, 13(2):94–102, 1970.

[89] Miles Efron. Using multiple query aspects to build test collections without human relevance judgments. In *ECIR '09: Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval*, pages 276–287. Springer-Verlag, 2009.

[90] D. Ellis. The physical and cognitive paradigms in information retrieval research. *Journal of Documentation*, 48:45–64, 1992.

[91] J.L. Elsas and S.T. Dumais. Leveraging temporal dynamics of document content in relevance ranking. In *Proc. of the Third ACM Int. Conf. on Web Search and Data Mining*, WSDM '10, pages 1–10, New York, NY, USA, 2010. ACM.

[92] Danlos Laurence et Sagot Benoît. Constructions pronominales dans dicovalence et le lexique-grammaire – intégration dans le lefff. In *27th conference on Lexis and Grammar*, Aquila, Italia, October 2008.

[93] J.L. Fagan. Automatic phrase indexing for document retrieval: An examination of syntactic and non-syntactic methods. In *Proc. of the 10th Int. Conf. on Research and Development in Information Retrieval*, SIGIR'87, pages 91–101. ACM, 1987.

[94] W. Fan, M.D. Gordon, and P. Pathak. A generic ranking function discovery framework by genetic programming for information retrieval. *Information Processing & Management*, 40:587–602, May 2004.

[95] W. Fan, M.D. Gordon, and P. Pathak. Genetic programming-based discovery of ranking functions for effective web search. *Journal of Management Information Systems*, 21:37–56, April 2005.

[96] D. Faure and C. Nedellec. A corpus-based conceptual clustering method for verb frames and ontology. In P. Velardi, editor, *Proceedings of the LREC Workshop on Adapting lexical and corpus resources to sublanguages and applications*, pages 5–12, 1998.

[97] Christiane Fellbaum, editor. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA, 1998.

[98] Milagros Fernández, Eric Villemonte de la Clergerie, and Manuel Vilares Ferro. Mining conceptual graphs for knowledge acquisition. In Fotis Lazarinis, Efthimis N. Efthimiadis, Jesús Vilares, and John Tait, editors, *CIKM-iNEWS*, pages 25–32. ACM, 2008.

[99] Manuel Vilares Ferro, Victor M. Darriba, and Jesús Vilares Ferro. Parsing incomplete sentences revisited. In *CICLing*, pages 102–111, 2004.

[100] F. Fonseca, M. Egenhofer, C. Davis, and G. Câmara. Semantic granularity in ontology-driven geographic information systems. *Annals of Mathematics and Artificial Intelligence*, 36:121–151, September 2002.

[101] James C. French, Allison L. Powell, Fredric C. Gey, and Natalia Perelman. Exploiting A controlled vocabulary to improve collection selection and retrieval effectiveness. In *CIKM*, pages 199–206. ACM, 2001.

[102] Y. Freund, R. Iyer, R.E. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. *The Journal of Machine Learning Research*, 4:933–969, December 2003.

[103] N. Fuhr and C. Buckley. A probabilistic learning approach for document indexing. *ACM Transanctions on Information Systems*, 9:223–248, July 1991.

[104] Norbert Fuhr. Probabilistic models in information retrieval. *The Computer Journal*, 35:243–255, 1992.

[105] Michael Fuller, Marcin Kaszkiel, Sam Kimberley, Corinna Ng, Ross Wilkinson, Mingfang Wu, and Justin Zobel. The rmit/csiro ad hoc, q&a, web, interactive, and speech experiments at trec 8. In *TREC*, 1999.

[106] Antony Galton. Temporal logic. In *Stanford Encyclopedia of Philosophy*. 2008.

[107] José Miguel Gambra and Manuel Oriol. *Lógica Aristotélica*. Dykinson, Madrid, 2008.

[108] J. Gao and J.-Y. Nie. A study of statistical models for query translation: finding a good unit of translation. In *Proc. of the 29th Int. Conf. on Research and Development in Information Retrieval*, SIGIR'06, pages 194–201, New York, NY, USA, 2006. ACM.

[109] Jianfeng Gao, Jian-Yun Nie, Guangyuan Wu, and Guihong Cao. Dependence language model for information retrieval. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '04, pages 170–177, New York, NY, USA, 2004. ACM.

[110] G. Gazdar. Applicability of indexed grammars to natural languages. In U. Reyle and C. Rohrer, editors, *Natural Language Parsing and Linguistic Theories*, pages 69–94. Reidel, Dordrecht, 1988.

[111] David Genest. *Extension du modèle des graphes conceptuels pour la recherche d'informations*. PhD thesis, Université Montpellier II, 2000.

[112] David Genest and Michel Chein. A content-search information retrieval process based on conceptual graphs. *Knowl. Inf. Syst*, 8(3):292–309, 2005.

[113] P. Ghodsnia, A.M.Z. Bidoki, and N. Yazdani. A punishment/reward based approach to ranking. In *Proc. of the 2nd Int. Conf. on Scalable information systems*, InfoScale'07, pages 58:1–58:4, ICST, Brussels, Belgium, Belgium, 2007. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[114] J.A. Goldsmith, D. Higgins, and S. Soglasnova. Automatic language-specific stemming in information retrieval. In *Revised Papers from the Workshop of Cross-Language Evaluation Forum on Cross-Language Information Retrieval and Evaluation*, CLEF'00, pages 273–284, London, UK, 2001. Springer-Verlag.

[115] M. Gordon and P. Pathak. Finding information on the world wide web: the retrieval effectiveness of search engines. *Information Processing & Management*, 35:141–180, March 1999.

[116] Jorge Graña Gil, Miguel Angel Alonso Pardo, and Alberto Valderruten Vidal. Análisis léxico no determinista: Etiquetación eficiente del lenguaje natural. Technical Report 16, Departamento de Computación, Facultade de Informática, Universidade da Coruña, Campus de Elviña s/n, 15071 La Coruña, Spain, 1994.

[117] L.A. Granka, T. Joachims, and G. Gay. Eye-tracking analysis of user behavior in WWW search. In *Proc. of the 27th Int. Conf. on Research and Development in Information Retrieval*, SIGIR'04, pages 478–479, New York, NY, USA, 2004. ACM.

[118] Gregory Grefenstette. Corpus-derived first, second and third-order word affinities. In *In Proceedings of Euralex*, pages 279–290, 1994.

[119] Gregory Grefenstette and Pasi Tapanainen. What is a word, what is a sentence? problems of tokenization. In *3rd Conference on Computational Lexicography and Text Research*, pages 79–87, Budapest, Hungary, 1994.

[120] W. Greuter, J. McNeill, F. R. Barrie, H.-M. Burdet, V. Demoulin, T. S. Filgueras, D. H. Nicolson, P. C. Silva, J. E. Skog, P. Trehane, N. J. Turland, and D. L. Hawksworth. *International Code of Botanical Nomenclature (St Louis Code)*. Number 138 in Regnum Vegetabile. Koeltz Scientific Books, Königstein, 2000. Adopted by the Sixteenth International Botanical Congress St Louis, Missouri.

[121] J. Guiver, S. Mizzaro, and S. Robertson. A few good topics: Experiments in topic set reduction for retrieval evaluation. *ACM Transactions on Information Systems*, 27:21:1–21:26, November 2009.

[122] Carlos Muñoz Gutiérrez. Introducción a la lógica, 2006.

[123] Antonio-José Gómez Flechoso. *Inducción de conocimiento con incertidumbre en bases de datos relacionales borrosas*. PhD thesis, Escuela Técnica de Superior de Ingenieros de Telecomunicación. Universidad Politécnica de Madrid, Madrid, Spain, 1998.

[124] Benoît Habert and Adeline Nazarenko. La syntaxe comme marche-pied de l'acquisition des connaissances : bilan critique d'une expérience. In *Journées sur l'acquisition des connaissances*, pages 137–142, Sète, mai 1996. AFIA.

[125] Sanda M. Harabagiu, George A. Miller, and Dan I. Moldovan. WordNet 2 – a morphologically and semantically enhanced resource. In *Proc. SIGLEX 1999*, 1999.

[126] Robert M. Haralick and Linda G. Shapiro. *Computer and Robot Vision*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1992.

[127] D. Harman. Overview of the second text retrieval conference (trec-2). In *Proc. of the workshop on Human Language Technology*, HLT'94, pages 351–357, Stroudsburg, PA, USA, 1994. Association for Computational Linguistics.

[128] Sofia N. Galicia Haro and Alexander Gelbukh. *Investigaciones en análisis sintáctico para el español*. Instituto Politécnico Nacional, 2007.

[129] Zellig Harris. Distributional structure. *Word*, 10(23):146–162, 1954.

[130] Zellig Harris. *Mathematical Structures of Language*. John Wiley and Son, New York, 1968.

[131] S.P. Harter. Variations in relevance assessments and the measurement of retrieval effectiveness. *Journal of the American Society for Information Science*, 47:37–49, January 1996.

[132] Taher H. Haveliwala. Topic-sensitive pagerank. In *Proc. of the 11th Int. Conf. on World Wide Web*, WWW'02, pages 517–526, New York, NY, USA, 2002. ACM.

[133] Ben He and Iadh Ounis. Term frequency normalisation tuning for bm25 and dfr model. In *In Proceedings of ECIR 2005*, pages 200–214. Springer, 2005.

[134] Sandra Heiler. Semantic interoperability. *ACM Computing Surveys*, 27(2):271–273, 1995.

[135] R. Herbrich, T. Graepel, and K. Obermayer. Support vector learning for ordinal regression. In *In Proc. of Int. Conf. on Artificial Neural Networks*, ICANN'99, pages 97–102, 1999.

[136] R. Herbrich, T. Graepel, and K. Obermayer. Large margin rank boundaries for ordinal regression. In P.J. Bartlett, B. Schölkopf, D. Schuurmans, and A.J. Smola, editors, *Advances in Large Margin Classifiers*, pages 115–132. MIT Press, 2000.

[137] John E. Hopcroft and Jeffrey D. Ullman. *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley, 1979.

[138] J.I. Hualde, A. Olarrea, and A.M. Escobar. *Introducción a la lingüistica hispánica*. Cambridge University Press, 2002.

[139] Nancy Ide and Jean Veronis. Introduction to the special issue on word sense disambiguation: The state of the art. *Computational Linguistics*, 24:1–40, 1998.

[140] Paul Jaccard. Étude comparative de la distribution florale dans une portion des alpes et des jura. *Bulletin del la Société Vaudoise des Sciences Naturelles*, 37:547–579, 1901.

[141] Peter Jackson and Isabelle Moulinier. *Natural Language Processing for Online Applications: Text Retrieval, Extraction, and Categorization (Natural Language Processing, 5)*. John Benjamins Publishing Co, June 2002.

[142] Christian Jacquemin, Judith Klavans, and Evelyne Tzoukermann. Expansion of multi-word terms for indexing and retrieval using morphology and syntax. In *ACL*, pages 24–31, 1997.

[143] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. volume 20, pages 422–446, New York, NY, USA, October 2002. ACM.

[144] E. T. Jaynes. *Probability Theory: The Logic of Science (Vol 1)*. Cambridge University Press, April 2003.

[145] W. Jin and R.K. Srihari. Graph-based text representation and knowledge discovery. In *Proc. of the Symposium on Applied Computing*, SAC'07, pages 807–811, New York, NY, USA, 2007. ACM.

[146] T. Joachims. Optimizing search engines using clickthrough data. In *Proc. of the Eighth ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, KDD'02, pages 133–142, New York, NY, USA, 2002. ACM.

[147] K.S. Jones. *What is the role of* NLP *in text retrieval ?*, pages 1–24. Text, Speech and Language Technology Book Series. Kluwer Academic Publishers, 1999.

[148] T. Jones, D. Hawking, P. Thomas, and R. Sankaranarayana. Relative effect of spam and irrelevant documents on user interaction with search engines. In *Proc. of the 20th Int. Conf. on Information and Knowledge Management*, CIKM'11, pages 2113–2116, New York, NY, USA, 2011. ACM.

[149] Aravind Joshi, L.S. Levy, and M. Takahashi. Tree adjunct grammars. *Journal of Computer and System Sciences*, 10(1):136–163, 1975.

[150] Aravind Joshi and Yves Schabes. *Tree-Adjoining Grammars*, chapter Handbook of Fromal Languages, Vol.3: Beyond Words, chapter 2, pages 69–123. Springer-Verlag, Berlin / Heidelberg / New York, 1997.

[151] Aravind K. Joshi. Tree adjoining grammars: how much context-sensitivity is required to provide reasonable structural descriptions? In David R. Dowty, Lauri Karttunen, and Arnold Zwicky, editors, *Natural Language Parsing*, pages 206–250. Cambridge University Press, Cambridge, 1985.

[152] Aravind K Joshi. An introduction to tree adjoining grammar. In A Manaster-Ramer, editor, *Mathematics of Language*. John Benjamins, Amsterdam, 1987.

[153] Aravind K. Joshi. Domains of locality. *Data Knowledge Engineering*, 50(3):277–289, 2004.

[154] Aravind K. Joshi and Yves Schabes. Tree-adjoining grammars and lexicalized grammars. In *Tree Automata and Languages*, pages 409–432. 1992.

[155] Daniel Jurafsky and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition (International Edition).* Prentice Hall, February 2000.

[156] J.S. Justeson and S.M. Katz. Technical terminology: some linguistic properties and an algorithm for identification in text. *Natural Language Engineering*, 1:9–27, 1995.

[157] F. Lepage J.Y. Nie. *Toward a broader model for information retrieval*, chapter nformation Retrieval, Uncertainty and Logics, pages 17–38. eds. M. Lalmas, F. Crestani, C.J. van Rijsbergen, Kluwer Academic Publishers, 1998.

[158] R. Karp. Reducibility among combinatorial problems. In R. Miller and J. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972.

[159] R. T. Kasper and W. C. Rounds. A logical semantics for feature structures. In *Proc. of the 24th ACL*, pages 257–266, New York, 1986.

[160] M Kay. *Parsing in functional unification grammar*, pages 125–138. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1986.

[161] M. Keen. *Evaluation Parameters*, chapter 5. Prentice-Hall, Inc., 1971.

[162] M.G. Kendall. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93, 1938.

[163] Kimmo Kettunen, Eija Airio, and Kalervo Järvelin. Restricted inflectional form generation in management of morphological keyword variation. *Inf. Retr.*, 10:415–444, October 2007.

[164] Alexandra Kinyon. Hypertags. In *Proceedings of the 18th conference on Computational linguistics - Volume 1*, COLING '00, pages 446–452, Stroudsburg, PA, USA, 2000. Association for Computational Linguistics.

[165] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46:604–632, September 1999.

[166] Donald E. Knuth. *The art of computer programming*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1997.

[167] Phokion G. Kolaitis and Moshe Y. Vardi. Conjunctive-query containment and constraint satisfaction. In *Proceedings of the seventeenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*, PODS '98, pages 205–213, New York, NY, USA, 1998. ACM.

[168] T.G. Kolda and D.P. O'Leary. A semidiscrete matrix decomposition for latent semantic indexing information retrieval. *ACM Transactions on Information Systems*, 16(4):322–346, 1998.

[169] Kimmo Koskenniemi. Two-level model for morphological analysis. In *IJCAI-83*, pages 683–685, Karlsruhe, Germany, 1983.

[170] Kimmo Koskenniemi. Two-level morphology: a general computational model for word-form recognition and production. Technical Report 11, Department of General Linguistics, University of Helsinki, 1983.

[171] C.H.A. Koster and J.G. Beney. Phrase-based document categorization revisited. In *Proc. of the 2nd Int. Workshop on Patent Information Retrieval*, PaIR'09, pages 49–56, New York, NY, USA, 2009. ACM.

[172] Anthony S. Kroch. Unbounded dependencies and subjacency in a tree adjoining grammar. In Alexis Manaster-Ramer, editor, *Proceedings of the First Conference on the Mathematics of Language*, pages 143–172. Benjamins, Amsterdam, 1986.

[173] A. Kulkarni, J. Teevan, K.M. Svore, and S.T. Dumais. Understanding temporal query dynamics. In *Proc. of the Fourth ACM Int. Conf. on Web Search and Data Mining*, WSDM '11, pages 167–176, New York, NY, USA, 2011. ACM.

[174] J.-W. Kuo, P.-J. Cheng, and H.-M. Wang. Learning to rank from bayesian decision inference. In *Proc. of the 18th Int. Conf. on Information and Knowledge Management*, CIKM'09, pages 827–836, New York, NY, USA, 2009. ACM.

[175] R. Küsters and R. Molitor. Structural Subsumption and Least Common Subsumers in a Description Logic with Existential and Number Restrictions. *Studia Logica*, 81:227–259, 2005.

[176] Y. Lan, T.-Y. Liu, Z. Ma, and H. Li. Generalization analysis of listwise learning-to-rank algorithms. In *Proc. of the 26th Annual Int. Conf. on Machine Learning*, ICML'09, pages 577–584, New York, NY, USA, 2009. ACM.

[177] Bernard Lang. Deterministic techniques for efficient non-deterministic parsers. In *ICALP*, pages 255–269, 1974.

[178] Jean-Louis Laurière. Représentation et utilisation des connaissances-première partie: Les systèmes experts. In *Technique et Science Informatiques*, volume 1, pages 25–42, 1982.

[179] Ludovic Lebart and André Salem. *Statistique Textuelle*. Dunod, Paris, 1994.

[180] C. Lee and G.G. Lee. Probabilistic information retrieval model for a dependency structured indexing system. *Information Processing & Management*, 41(2):161–175, 2005.

[181] P. Lefèbvre and Eric Villemonte de la Clergerie. How to build quickly an efficient implementation of the domain prop with dyalog. In *LPE*, pages 33–38, 1993.

[182] Fritz Lehmann. Semantic networks. *Computers & Mathematics with Applications*, 23(2-5):1 – 50, 1992.

[183] M. Li, H. Li, and Z.-H. Zhou. Semi-supervised document retrieval. *Information Processing & Management*, 45:341–355, May 2009.

[184] P. Li, C.J.C. Burges, and Q. Wu. Mcrank: Learning to rank using multiple classification and gradient boosting. In J.C. Platt, D. Koller, Y. Singer, and S.T. Roweis, editors, *Proc. of Advances in Neural Information Processing Systems*, volume 20 of *NIPS'07*, pages 897–904. MIT Press, 2007.

[185] Carl von Linné and Salvii. Laurentii. *Caroli Linnaei...Systema naturae per regna tria naturae*, volume v.1. Holmiae :Impensis Direct. Laurentii Salvii,, 1758-1759. http://www.biodiversitylibrary.org/bibliography/542.

[186] Carl von Linné and Salvii. Laurentii. *Caroli Linnaei...Systema naturae per regna tria naturae*, volume v.2. Holmiae :Impensis Direct. Laurentii Salvii,, 1758-1759. http://www.biodiversitylibrary.org/bibliography/542.

[187] C. Liu, H. Wang, S. Mc Clean, J. Liu, and S. Wu. Syntactic information retrieval. In *Proc. of the Int. Conf. on Granular Computing*, GRC'07, page 703, Washington, DC, USA, 2007. IEEE Computer Society.

[188] Carlos M. Lorenzetti. *Caracterización Formal y Análisis Empírico de Mecanismos Incrementales de Búsqueda basados en Contexto*. PhD thesis, Universidad Nacional del Sur, Bahía Blanca, Argentina, Marzo 2011.

[189] Julie B. Lovins. Development of a stemming algorithm. *Mechanical Translation and Computational Linguistics*, 11:22–31, 1968.

[190] George F. Luger. *Artificial intelligence: Structures and strategies for complex problem solving*. Addison-Wesley, England, 2005.

[191] H. P. (Hans Peter) Luhn and Claire K Schultz. *H.P. Luhn : pioneer of information science : selected works / Edited by Claire K. Schultz*. New York, : Spartan Books ; London : Macmillan, 1968.

[192] H.P. Luhn. The automatic creation of literature abstracts. *IBM Journal*, 2:159–165, 1958.

[193] L. Maisonnasse, E. Gaussier, and J.-P. Chevallet. Revisiting the dependence language model for information retrieval. In *Proc. of the 30th Int. Conf. on Research and Development in Information Retrieval*, SIGIR'07, pages 695–696, New York, NY, USA, 2007. ACM.

[194] S. Maiti, D.P. Mandal, and P. Mitra. Tackling content spamming with a term weighting scheme. In *Proc. of the Joint Workshop on Multilingual OCR and Analytics for Noisy Unstructured Text Data*, MOCR-AND'11, pages 6:1–6:5, New York, NY, USA, 2011. ACM.

[195] Bill Z. Manaris and Brian M. Slator. Interactive natural language processing: Building on success. *Computer*, 29(7):28–32, 1996.

[196] P. Manchon. Structuration de documents. Stage X, DIX – École Polytechnique, July 2003.

[197] D. Manjula, G. Aghila, and T. V. Geetha. Document knowledge representation using description logics for information extraction and querying. In *Proc. of the Int. Conf. on Information Technology: Computers and Communications*, ITCC'03, page 189, Washington, DC, USA, 2003. IEEE Computer Society.

[198] C.D. Manning and H. Schütze. *Foundations of statistical natural language processing*. MIT Press, Cambridge, MA, USA, 1999.

[199] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schtze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

[200] M. E. Maron and J. L. Kuhns. On relevance, probabilistic indexing and information retrieval. *J. ACM*, 7:216–244, July 1960.

[201] Julien Martin. Mieux comprendre les méta-grammaires. Master's thesis, Université Paris 6, September 2006.

[202] Diana Maynard, Valentin Tablan, Cristian Ursu, Hamish Cunningham, and Yorick Wilks. Named entity recognition from diverse text types. In *In Recent Advances in Natural Language Processing 2001 Conference, Tzigov Chark*, 2001.

[203] Diana McCarthy. Word sense disambiguation: The case for combinations of knowledge sources, by mark stevenson. clsi, 2003. isbn: 1-57586-390-1. *Nat. Lang. Eng.*, 10(2):196–200, 2004.

[204] John G. McMahon and Francis J. Smith. Improving statistical language model performance with automatically generated word hierarchies. *Comput. Linguist.*, 22:217–247, June 1996.

[205] G. A. Miller. WordNet: a Lexical Database for English. *Communications of the ACM*, 38(11):39–41, 1995.

[206] Marvin Minsky. A framework for representing knowledge. Technical report, Cambridge, MA, USA, 1974.

[207] S. Mizzaro. The good, the bad, the difficult, and the easy: something wrong with information retrieval evaluation ? In *Proc. of the 30th European Conf. on Information Retrieval*, ECIR'08, pages 642–646, Berlin, Heidelberg, 2008. Springer-Verlag.

[208] S. Mizzaro and S. Robertson. Hits hits TREC: exploring IR evaluation results with network analysis. In *Proc. of the 30th Int. Conf. on Research and Development in Information Retrieval*, SIGIR '07, pages 479–486, New York, NY, USA, 2007. ACM.

[209] Dan I. Moldovan and Rada Mihalcea. Using wordnet and lexical operators to improve internet searches. *IEEE Internet Computing*, 4:34–43, January 2000.

[210] Miguel A. Molinero, Benoît Sagot, and Lionel Nicolas. A morphological and syntactic wide-coverage lexicon for spanish: The leffe. In *Proceedings of the International Conference RANLP-2009*, pages 264–269, Borovets, Bulgaria, September 2009. Association for Computational Linguistics.

[211] C. Molinier, I. Choi-Jonin, M. Bras, A. Dagnac, and M. Rouquier. *Questions de classification en linguistique: méthodes et descriptions:*. Sciences Pour La Communication. Peter Lang, 2005.

[212] C. Molinier and F. Levrier. *Grammaire des adverbes: description des formes en -ment*. Langue & cultures. Droz, 2000.

[213] Christian Molinier. Une classification des adverbes en -ment. *Langue française*, 88(1):28–40, 1990.

[214] Richard Montague. The proper treatment of quantification in ordinary English. In K. J. J. Hintikka, J. Moravcsic, and P. Suppes, editors, *Approaches to Natural Language*, pages 221–242. Reidel, Dordrecht, 1973.

[215] M. Montes y Gómez. *Minería de texto empleando la semejanza entre estructuras semánticas*. PhD thesis, Instituto Politécnico Nacional, México D.F., México, 2005.

[216] M. Montes y Gómez, A. López-López, and A. Gelbukh. Information retrieval with conceptual graph matching. In *Proc. of 11th Int. Conf. on Database and Expert Systems Applications*, number 1873 in Lecture Notes in Computer Science, pages 312–321. Springer-Verlag, 2000.

[217] Antonio Moreno Sandoval. *Lingüística Computacional. Introducción a los modelos simbólicos, estadísticos y biológicos.* Sintesis, Madrid, 1998.

[218] J. Mothe and L. Tanguy. Linguistic analysis of users'queries: Towards an adaptive information retrieval system. In *Proc. of the Third Int. Conf. on Signal-Image Technologies and Internet-Based System*, SITIS'07, pages 77–84, Washington, DC, USA, 2007. IEEE Computer Society.

[219] A. Mowshowitz and A. Kawaguchi. Bias on the web. *Communications of the ACM*, 45:56–60, September 2002.

[220] Marie-Laure Mugnier and Michel Leclère. On querying simple conceptual graphs with negation. *Data Knowl. Eng.*, 60(3):468–493, 2007.

[221] J. Myhill. *Linear Bounded Automata*. Us Dept. of Commerce Office of Tech. Services Ots. 1960.

[222] R. Nallapati. Discriminative models for information retrieval. In *Proc. of the 27th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, SIGIR'04, pages 64–71, New York, NY, USA, 2004. ACM.

[223] Fiammetta Namer, Robert Baud, Anita Burgun, Stéfan J. Darmoni, Natalia Grabar, Eric Jarrousse, Franck Le Duff, Patrick Ruch, Benoît Thirion, and Pierre Zweigenbaum. UMLF : construction d'un lexique médical francophone unifié. In *Journée Francophone d'informatique médicale*, Tunis Tunisie, 09 2003.

[224] E. Naulleau. *Apprentissage et filtrage syntaxico-semántique de syntagmes nominaux pertinents pour la recherche documentaire*. PhD thesis, Université Paris XIII, Paris, France, 1998.

[225] Michael Negnevitsky. *Artificial intelligence: a guide to intelligent systems (2º Edition)*. Pearson Education, 2005.

[226] A. Ntoulas, M. Najork, M. Manasse, and D. Fetterly. Detecting spam web pages through content analysis. In *Proc. of the 15th Int. Conf. on World Wide Web*, WWW'06, pages 83–92, New York, NY, USA, 2006. ACM.

[227] International Code of Botanical Nomenclature. *International code of botanical nomenclature : adopted by the Seventeenth International Botanical Congress, Vienna, Austria*. International Code of Botanical Nomenclature (Vienna Code), Regnum Vegetabile 146. A.R.G. Gantner Verlag, Königstein, 2005.

[228] International Commission on Zoological Nomenclature. *International Code of Zoological Nomenclature*. ICZN, Natural History Museum, London, 1999.

[229] J. Otero Pombo. *Análisis léxico robusto*. PhD thesis, Universidad de Vigo, Ourense, España, Junio 2009.

[230] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November 1999. Previous number = SIDL-WP-1999-0120.

[231] Chris D. Paice. A thesaural model of information retrieval. *Inf. Process. Manage.*, 27(5):433–447, 1991.

[232] Jie Peng, Craig Macdonald, Ben He, Vassilis Plachouras, and Iadh Ounis. Incorporating term dependency in the dfr framework. In *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 843–844, New York, NY, USA, 2007. ACM.

[233] Jose Perez Carballo and Tomek Strzalkowski. Natural language information retrieval: progress report. *Inf. Process. Manage.*, 36(1):155–178, 2000.

[234] J. Perron. Adepte-nomino, un outil de veille terminologique. *Terminologies nouvelles*, 15(2):32–47, 1996.

[235] J. Pitkow, H. Schütze, T. Cass, R. Cooley, D. Turnbull, A. Edmonds, E. Adar, and T. Breuel. Personalized search. *Communications of the ACM*, 45:50–55, September 2002.

[236] J.M.. Ponte and W.B. Croft. A language modeling approach to information retrieval. In *Proc. of the 21st Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, SIGIR'98, pages 275–281, New York, NY, USA, 1998. ACM.

[237] M. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[238] T. Qin, X.-D. Zhang, M.-F. Tsai, D.-S. Wang, T.-Y. Liu, and H. Li. Query-level loss functions for information retrieval. *Information Processing & Management*, 44:838–855, March 2008.

[239] M. R. Quillian. Word concepts: a theory and simulation of some basic semantic capabilities. *Behavioral Science*, 12(5):410–430, September 1967.

[240] C. Quiroga-Clare. Language ambiguity: A curse and a blessing. *Translation Journal*, 7(1), 2003.

[241] V. Raghavan, P. Bollmann, and G.S. Jung. A critical investigation of recall and precision as measures of retrieval system performance. *ACM Transactions on Information Systems*, 7:205–229, July 1989.

[242] N. Rescher. *Many-Valued Logic*. New York: McGraw-Hill, 1969.

[243] Philip Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *IJCAI'95: Proceedings of the 14th international joint conference on Artificial intelligence*, pages 448–453, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc.

[244] Philip Stuart Resnik. *Selection And Information: A Class-based Approach to Lexical Relationships*. PhD thesis, University of Pennsylvania, Philadelphia, USA, 1993.

[245] Dominique Revuz. *Dictionnaires et lexiques: méthodes et algorithmes*. PhD thesis, Institut Blaise Pascal, Paris, France, 1991. LITP 91.44.

[246] Francisco J. Ribadas, Manuel Vilares Ferro, and Jesús Vilares Ferro. Semantic similarity between sentences through approximate tree matching. In *IbPRIA (2)*, pages 638–646, 2005.

[247] Elaine Rich and Kevm Knight. *Inteligencia Artificial (2º Edición)*. Mc Craw Hill, Great Britain, 1994.

[248] S. E. Robertson. The Probability Ranking Principle in IR. *Journal of Documentation*, 33(4):294–304, 1977.

[249] S. E. Robertson and Sparck K. Jones. Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27(3):129–146, 1976.

[250] S. E. Robertson and S. Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 232–241, New York, NY, USA, 1994. Springer-Verlag New York, Inc.

[251] S. E. Robertson, S. Walker, and M. M. Hancock-Beaulieu. Large test collection experiments on an operational, interactive system: Okapi at trec. *Inf. Process. Manage.*, 31(3):345–360, 1995.

[252] S.E. Robertson, M.E. Maron, and W.S. Cooper. Probability of relevance: A unification of two competing models for document retrieval. *American Information Technology: Research and Development*, 1:1–21, 1982.

[253] Stephen E. Robertson and Karen Sparck Jones. Relevance weighting of search terms. pages 143–160, 1988.

[254] V. Rocio and G. P. Lopes. *Cascaded Partial Parsing (Análise sintáctica parcial em cascata)*, pages 235–251. Edições Colibri, p. marrafa e m. a. mota edition, 1999. ISBN=ISBN 972-772-090-0, URL=http://http://www.univ-ab.pt/ vjr/papers/Apl98.ps.

[255] James Rogers. A unified notion of derived and derivation structures in tag. In *Proc. of the Fifth Meeting on Mathematics of Language*, pages 95–104, Schloss Dagstuhl, Saarbruecken, Germany, April 1997.

[256] James Rogers and K. Vijay-shanker. Obtaining trees from their descriptions: An application to tree-adjoining grammars. *Computational Intelligence*, 10:401–421, 1994.

[257] François Role, Milagros Fernandez Gavilanes, and Éric Villemonte de la Clergerie. Large-scale knowledge acquisition from botanical texts. In *Proc. of NLDB'07*, 2007.

[258] Guillaume Rousse and Éric Villemonte de La Clergerie. Analyse automatique de documents botaniques: le projet Biotim. In *proc. of TIA'05*, pages 95–104, Rouen, France, April 2005.

[259] Catherine Roussey. *Une méthode d'indexation sémantique adaptée aux corpus multilingues*. Thèse de doctorat en informatique, INSA de Lyon, December 2001.

[260] Stuart Russell and Peter Norving. *Inteligencia Artificial. Un Enfoque Moderno. Segunda edición*. Pearson Educación, S. A. Madrid, 2004.

[261] Naomi. Sager, Carol Friedman, and Margaret S. Lyman. *Medical Language Processing: Computer Management of Narrative Data*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1987.

[262] B. Sagot. *Analyse automatique du français: lexiques, formalismes, analyseurs*. PhD thesis, Université Paris VII, Paris, France, 2006.

[263] B. Sagot. The lefff, a freely available and large-coverage morphological and syntactic lexicon for french. In *Proceedings of LREC'10*, Valetta, Malta, 2010.

[264] B. Sagot and P. Boullier. Sxpipe 2: architecture pour le traitement pré-syntaxique de corpus bruts. *Traitement Automatique des Langues*, 2(49):155–188, 2008.

[265] B Sagot and K. Fort. Améliorer un lexique syntaxique à l'aide des tables du lexique-grammaire - adverbes en -ment. In *26th conference on Lexis and Grammar*, Bonifacio, France, October 2007.

[266] Benoît Sagot, Lionel Clément, Éric Villemonte de La Clergerie, and Pierre Boullier. Vers un méta-lexique pour le français : architecture, acquisition, utilisation. Journée d'étude de l'ATALA sur l'Interface lexique-grammaire et lexiques syntaxiques et sémantiques, March 2005.

[267] Benoît Sagot and Éric Villemonte de La Clergerie. Error mining in parsing results. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 329–336, Sydney, Australia, July 2006. Association for Computational Linguistics.

[268] T. Sakai. On the reliability of information retrieval metrics based on graded relevance. *Information Processing & Management*, 43:531–548, March 2007.

[269] G. Salton. *The SMART Retrieval System—Experiments in Automatic Document Processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1971.

[270] G. Salton, C. Buckley, and C.T. Yu. An evaluation of term dependence models in information retrieval. In *Proc. of the 5th Int. Conf. on Research and Development in Information Retrieval*, SIGIR'82, pages 151–173, New York, NY, USA, 1982. Springer-Verlag New York, Inc.

[271] Gerard Salton. *Automatic text processing: the transformation, analysis, and retrieval of information by computer*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1989.

[272] Gerard Salton. Developments in automatic text retrieval. *Science*, 253:974–979, 1991.

[273] Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA, 1986.

[274] Gerard Salton, Anita Wong, and Chung-Shu Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975. The paper where vector space model for IR was introduced.

[275] Mark Sanderson. Word sense disambiguation and information retrieval. In *SIGIR-94*, pages 142–151, Dublin, Ireland, 1994. ACM.

[276] Mark Sanderson and Justin Zobel. Information retrieval system evaluation: effort, sensitivity, and reliability. In *SIGIR*, pages 162–169, 2005.

[277] T. Saracevic. Relevance: A review of the literature and a framework for thinking on the notion in information science. part II: nature and manifestations of relevance. *Journal of the American Society for Information Science and Technology*, 58:1915–1933, November 2007.

[278] Tefko Saracevic and Paul Kantor. A study of information seeking and retrieving, iii: Searchers, searches, overlap. *Journal of the American Society for Information Science and Technology*, pages 39–177, 1988.

[279] Yves Schabes. *Mathematical and computational aspects of lexicalized grammars*. PhD thesis, Philadelphia, PA, USA, 1990. Supervisor-Joshi, Aravind K.

[280] Roger C. Schank. *Conceptual Information Processing*. Elsevier Science Inc., New York, NY, USA, 1975.

[281] Roger C. Schank, Janet L. Kolodner, and Gerald DeJong. Conceptual information retrieval. In *SIGIR '80: Proceedings of the 3rd annual ACM conference on Research and development in information retrieval*, pages 94–116, Kent, UK, UK, 1981. Butterworth & Co.

[282] Tefko Saracevic School and Tefko Saracevic. Information science. *Journal of the American Society for Information Science*, 50:1051–1063, 1999.

[283] J. Seo and J. Jeon. High precision retrieval using relevance-flow graph. In *Proc. of the 32nd Int. Conf. on Research and Development in Information Retrieval*, SIGIR'09, pages 694–695, New York, NY, USA, 2009. ACM.

[284] Florian Seydoux. *Exploitation de connaissances sémantiques externes dans les représentations vectorielles en recherche documentaire*. PhD thesis, Lausanne, 2006.

[285] K. Shaban. *A semantic graph model for text representation and matching in document mining*. PhD thesis, Waterloo, Ont., Canada, 2006.

[286] Stuart M. Shieber. *An Introduction to Unification-Based Approaches to Grammar*, volume 4 of *CSLI Lecture Notes Series*. Center for the Study of Language and Information, Stanford, CA, 1986. Spanish translation: *Introducción a los*

Formalismos Grammaticales de Unificación, Editorial Teide, Barcelona, 1989. French translation: Formalismes Syntaxiques pour le Traitement Automatique du Langage Naturel, Philip Miller and Thérèse Torris, editors, Hermeès, Paris, 1990.

[287] Advaith Siddharthan. Christopher d. manning and hinrich schütze. foundations of statistical natural language processing. mit press, 2000. isbn 0-262-13360-1. 620 pp. *Nat. Lang. Eng.*, 8(1):91–92, 2002.

[288] T.J. Siddiqui. Intelligent techniques for effective information retrieval: a conceptual graph based approach. *SIGIR Forum*, 40(2):73–74, 2006.

[289] H. A. Simon. *Otto Selz and information-processing psychology*, chapter Otto Selz: His contribution to psychology, pages 147–164. Mouton De Gruyter; First Edition edition, 1981.

[290] I. Soboroff, C. Nicholas, and P. Cahan. Ranking retrieval systems without relevance judgments. In *Proc. of the 24th Int. Conf. on Research and Development in Information Retrieval*, SIGIR'01, pages 66–73, New York, NY, USA, 2001. ACM.

[291] F. Song and W.B. Croft. A general language model for information retrieval. In *Proc. of the 8th Int. Conf. on Information and Knowledge Management*, CIKM'99, pages 316–321, New York, NY, USA, 1999. ACM.

[292] David Cabrero Souto, Jesus Vilares Ferro, and Manuel Vilares Ferro. Dynamic programming of partial parses. 2001.

[293] John F. Sowa. Conceptual graphs for a data base interface. *IBM Journal of Research and Development*, 20:336–357, July 1976.

[294] John F. Sowa. Semantics of conceptual graphs. In *Proceedings of the 17th annual meeting on Association for Computational Linguistics*, ACL '79, pages 39–44, Stroudsburg, PA, USA, 1979. Association for Computational Linguistics.

[295] John F. Sowa. *Conceptual Structures: Information Processing in Mind and Machine*. Systems Programming Series. Addison-Wesley, July 1983.

[296] Karen Sparck Jones and C J Van Rijsbergen. Information retrieval test collections. *Journal of Documentation*, 32(1):59–75, 1976.

[297] A. Spink and H. Greisdorf. Regions and levels: measuring and mapping users'relevance judgments. *Journal of the American Society for Information Science and Technology*, 52:161–173, January 2001.

[298] Anselm Spoerri. Using the structure of overlap between search results to rank retrieval systems without relevance judgments. *Inf. Process. Manage.*, 43(4):1059–1070, 2007.

[299] Richard Sproat. *Handbook of Natural Language Processing*, chapter Lexical Analysis, 3, pages 37–57. Marcel Dekker, Inc., New York and Basel, 2000.

[300] T. Strzalkowski. Natural language information retrieval. *Information Processing & Management*, 31(3):397–417, 1995.

[301] A.-J. Su, Y.C. Hu, A. Kuzmanovic, and C.-K. Koh. How to improve your google ranking: Myths and reality. In *Proc. of the Int. Conf. on Web Intelligence and Intelligent Agent Technology*, volume 1 of *WI-IAT'10*, pages 50–57, Washington, DC, USA, 2010. IEEE Computer Society.

[302] J. Tague-Sutcliffe and J. Blustein. A statistical analysis of the TREC-3 data. In *Overview of the Third Text REtrieval Conference*, TREC-3, pages 385–398, 1994.

[303] T.T. Tanimoto. Internal report: Ibm technical report series. Technical report, IBM, November 1957.

[304] R. H. Thomason and D. S. Touretzky. Inheritance theory and networks with roles. In J. F. Sowa, editor, *Principles of Semantic Networks: Explorations in the Representation of Knowledge*, pages 231–266. Kaufmann, San Mateo, 1991.

[305] François Thomasset and Éric Villemonte de La Clergerie. Comment obtenir plus des méta-grammaires. In *Proceedings of TALN'05*, Dourdan, France, June 2005. ATALA.

[306] U. S. Tiwary and Tanveer Siddiqui. *Natural Language Processing and Information Retrieval*. Oxford University Press, Inc., New York, NY, USA, 2008.

[307] Elsa Tolone and Benoit Sagot. Using lexicon-grammar tables for french verbs in a large-coverage parser. In *Proceedings of the 4th conference on Human language technology: challenges for computer science and linguistics*, LTC'09, pages 183–191, Berlin, Heidelberg, 2011. Springer-Verlag.

[308] David S. Touretzky. *The Mathematics of Inheritance Systems*. Morgan Kaufmann, 1986.

[309] E.G. Traugott. *The Ubiquity of metaphor: metaphor in language and thought*, chapter Conventional and dead metaphors revisited, pages 17–56. Amsterdam studies in the theory and history of linguistic science: Current issues in linguistic theory. J. Benjamins, 1985.

[310] E. Trillas, C. Alsina, and J.M. Terricabras. *Introducción a la lógica borrosa*. Ariel Matemática. Ariel, 1995.

[311] A. Trotman. Learning to rank. *Information Retrieval*, 8:359–381, May 2005.

[312] M.-F. Tsai, T.-Y. Liu, T. Qin, H.-H. Chen, and W.-Y. Ma. Frank: a ranking method with fidelity loss. In *Proc. of the 30th Annual Int. Conf. on Research and Development in Information Retrieval*, SIGIR'07, pages 383–390, New York, NY, USA, 2007. ACM.

[313] Alan Turing. Intelligent machinery. *Machine Intelligence*, 5, 1969.

[314] C. J. van Rijsbergen. *Information Retrieval*. Butterworth, 1979.

[315] C. J. van Rijsbergen. Another look at the logical uncertainty principle. *Inf. Retr.*, 2:17–26, February 2000.

[316] V.N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, 1998.

[317] José Luis Vicedo. *Recuperación de Información de Alta Precisión: Los Sistemas de Búsqueda de Respuestas*, volume 2 of *Colección de Monografías de la SEPLN*. CEE Limencop, 2003. PhD Thesis.

[318] K. Vijay-Shanker. Using descriptions of trees in a tree adjoining grammar. *Comput. Linguist.*, 18(4):481–517, 1992.

[319] K. Vijay-Shanker and Yves Schabes. Structure sharing in lexicalized tree-adjoining grammars. In *COLING*, pages 205–211, 1992.

[320] K. Vijay-Shanker and David J. Weir. The use of shared forests in tree adjoining grammar parsing. In *Proceedings of the 6th Conference of the European Chapter of ACL*, pages 384–393, 1993.

[321] K. Vijay-Shanker and David J. Weir. The equivalence of four extensions of context-free grammars. *Mathematical Systems Theory*, 27(6):511–546, 1994.

[322] M. Vilares, V. M. Darriba, J. Vilares, and F. J. Ribadas. Análisis sintáctico de sentencias incompletas. *Procesamiento del Lenguaje Natural*, 30:107–113, 2003.

[323] M. Vilares, V. M. Darriba, J. Vilares, and F. J. Ribadas. A formal frame for robust parsing. *Theor. Comput. Sci.*, 328:171–186, November 2004.

[324] Jesús Vilares Ferro. *Aplicaciones del procesamiento del lenguaje natural en la recuperación de información en español*. PhD thesis, Departamento de Computación, Universidad de A Coruña, A Coruña, Spain, 2005.

[325] É. Villemonte de La Clergerie. *Automates à Piles et Programmation Dynamique. DyALog : Une application à la programmation en Logique*. PhD thesis, Université Paris 7, 1993.

[326] É. Villemonte de La Clergerie. Construire des analyseurs avec DyALog. In *Proc. of TALN'02*, June 2002.

[327] É. Villemonte de La Clergerie. DyALog: a tabular logic programming based environment for NLP. In *Proceedings of 2nd International Workshop on Constraint Solving and Language Processing (CSLP'05)*, Barcelona, Spain, October 2005.

[328] É. Villemonte de La Clergerie. Convertir des dérivations tag en dépendances. In *Proceedings of TALN'10*, Dourdan, France, July 2010.

[329] É. Villemonte de La Clergerie, B. Sagot, L. Nicolas, and M.-L. Guénot. Frmg: évolutions d'un analyseur syntaxique tag du français. In *Proceedings of TALN'09*. ATALA, 2009.

[330] É. Villemonte de La Clergerie, B. Sagot, L. Nicolas, and M.-L. Guénot. Frmg: évolutions d'un analyseur syntaxique tag du français. In *11 Conférence internationale sur les technologies d'analyse syntaxique (IWPT'09)*, Paris, France, 2009.

[331] Ellen M. Voorhees. Natural language processing and information retrieval. In Maria Teresa Pazienza, editor, *SCIE: Information Extraction: Towards Scalable, Adaptable Systems*, volume 1714 of *Lecture Notes in Computer Science*, pages 32–48. Springer, 1999.

[332] Ellen M. Voorhees. Trec: Continuing information retrieval's tradition of experimentation. *Commun. ACM*, 50:51–54, November 2007.

[333] Ellen M. Voorhees and Donna Harman. Overview of the sixth text retrieval conference (trec-6). In *TREC*, pages 1–24, 1997.

[334] E.M. Voorhees. Variations in relevance judgments and the measurement of retrieval effectiveness. *Information Processing & Management*, 36:697–716, September 2000.

[335] E.M. Voorhees. Overview of the TREC 2004 robust retrieval track. In *In Proc. of the Thirteenth Text REtrieval Conference*, TREC-13, page 13, 2004.

[336] E.M. Voorhees and C. Buckley. The effect of topic set size on retrieval experiment error. In *Proc. of the 25th Int. Conf. on Research and Development in Information Retrieval*, SIGIR'02, pages 316–323, New York, NY, USA, 2002. ACM.

[337] E.M. Voorhees and D. Harman. Overview of the seventh text retrieval conference trec-7. In *Proc. of the Seventh Text REtrieval Conference (TREC-7*, pages 1–24, 1998.

[338] Piek Vossen, editor. *EuroWordNet: a multilingual database with lexical semantic networks*. Kluwer Academic Publishers, Norwell, MA, USA, 1998.

[339] Warren Weaver. Translation. In W.N. Locke and D.A. Booth, editors, *Machine Translation of Languages: Fourteen Essays*. MIT Press, Cambridge, MA, 1955.

[340] W. Webber, A. Moffat, and J. Zobel. Statistical power in retrieval experimentation. In *Proc. of the 17th Int. Conf. on Information and Knowledge Management*, CIKM '08, pages 571–580, New York, NY, USA, 2008. ACM.

[341] Eric Wehrli. *L' analyse syntaxique des langues naturelles*. Masson, Paris, 1997.

[342] Douglas B. West. *Introduction to Graph Theory*. Prentice Hall, 2 edition, September 2000.

[343] Terry Winograd. *Language As a Cognitive Process: Syntax*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1982.

[344] William A. Woods. *Semantics For a Question-Answering System*. Outstanding Dissertations in the Computer Sciences. Garland Publishing, New York, 1967.

[345] William A. Woods. Conceptual indexing: A better way to organize knowledge. Technical report, Mountain View, CA, USA, 1997.

[346] S. Wu and S.I. McClean. Evaluation of system measures for incomplete relevance judgment in IR. In *Proc. of the 7th Int. Conf. on Flexible Query Answering Systems*, pages 245–256, 2006.

[347] Shengli Wu and Fabio Crestani. Methods for ranking information retrieval systems without relevance judgments. In *Proc. of the 2003 ACM Symposium on Applied computing*, SAC'03, pages 811–816, New York, NY, USA, 2003. ACM.

[348] F. Xia, T.-Y. Liu, J. Wang, W. Zhang, and H. Li. Listwise approach to learning to rank: theory and algorithm. In *Proc. of the 25th Int. Conf. on Machine learning*, ICML'08, pages 1192–1199, New York, NY, USA, 2008. ACM.

[349] J. Xu and H. Li. Adarank: a boosting algorithm for information retrieval. In *Proc. of the 30th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, SIGIR'07, pages 391–398, New York, NY, USA, 2007. ACM.

[350] J. Xu, T.-Y. Liu, M. Lu, H. Li, and W.-Y. Ma. Directly optimizing evaluation measures in learning to rank. In *Proc. of the 31st Annual Int. Conf. on Research and Development in Information Retrieval*, SIGIR'08, pages 107–114, New York, NY, USA, 2008. ACM.

[351] X. Yan, R.Y.K. Lau, D. Song, X. Li, and J. Ma. Toward a semantic granularity model for domain-specific information retrieval. *ACM Transanctions on Information Systems*, 29:15:1–15:46, July 2011.

[352] Y. Yue, T. Finley, F. Radlinski, and T. Joachims. A support vector method for optimizing average precision. In *Proc. of the 30th Annual Int. Conf. on Research and Development in Information Retrieval*, SIGIR'07, pages 271–278, New York, NY, USA, 2007. ACM.

[353] G. Yule and N.B. Rafecas. *El lenguaje*. Lingüística (Akal). Ediciones Akal, 2007.

[354] Zhaohui Z., Hongyuan Z., Tong Z., Olivier C., Keke C., and Gordon S. A general boosting method and its application to learning ranking functions for web search. In J.C. Platt, D. Koller, Y. Singer, and S.T. Roweis, editors, *Proc. of Advances in Neural Information Processing Systems*, volume 20 of *NIPS'07*, pages 1697–1704. MIT Press, 2007.

[355] Jinglei Zhao and Yeogirl Yun. A proximity language model for information retrieval. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '09, pages 291–298, New York, NY, USA, 2009. ACM.

[356] J. Zobel. How reliable are the results of large-scale information retrieval experiments? In *Proc. of the 21st Int. Conf. on Research and Development in Information Retrieval*, SIGIR'98, pages 307–314, New York, NY, USA, 1998. ACM.

[357] J. Zobel and A. Moffat. Exploring the similarity space. *SIGIR Forum*, 32:18–34, April 1998.

[358] J Łukasiewicz. On three-valued logic (in polish). *Ruch Filozoficzny*, 5:170–171, 1920.

Índice alfabético