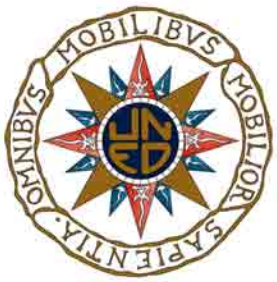
UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA
Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

ORGANIZACIÓN DE RESULTADOS DE BÚSQUEDA MEDIANTE ANÁLISIS FORMAL DE CONCEPTOS

TESIS DOCTORAL

Juan Manuel Cigarrán Recuero
Licenciado en Ciencias Físicas
2008

UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA
Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos



ORGANIZACIÓN DE RESULTADOS DE BÚSQUEDA MEDIANTE ANÁLISIS FORMAL DE CONCEPTOS

Juan Manuel Cigarrán Recuero
Licenciado en Ciencias Físicas por la Universidad Complutense de Madrid

Directores:

Julio Antonio Gonzalo Arroyo

Profesor Titular de Universidad del Departamento de Lenguajes y Sistemas Informáticos
de la Universidad Nacional de Educación a Distancia

Anselmo Peñas Padilla

Profesor Contratado Doctor del Departamento de Lenguajes y Sistemas Informáticos
de la Universidad Nacional de Educación a Distancia

A Ruth, Andrea, Angela y Daniela. Os quiero

En memoria de Carlota. Porque al final no pudo ser

Agradecimientos

Han sido muchos los años que he dedicado a este trabajo que hoy se cierra y también han sido muchas las personas con las que me he cruzado a lo largo de este camino. Todas ellas han influido positivamente en el resultado final, aportando la dosis exacta de inteligencia, discusión, comprensión simpatía y cariño para abordar un trabajo de esta envergadura. Así mismo, me han hecho valorar todo el trabajo que, aunque no pueda resultar visible, se esconde detrás de un investigador. Si bien es cierto que, en muchas ocasiones, la gran cantidad de horas invertidas para demostrar una idea chocan constantemente con un muro infranqueable, la satisfacción de terminar encontrando el camino merece todo el esfuerzo invertido.

De todas estas personas, en primer lugar quiero agradecer la paciencia que han tenido conmigo las que más quiero. Mi mujer Ruth y mis hijas Andrea, Angela y Daniela han sido mi principal apoyo y me han dado toda la fuerza y el equilibrio necesarios para terminar esta Tesis. Aparecieron en mi vida cuando ya había iniciado este camino y estoy seguro que no habría sido capaz de terminarlo sin ellas. A todas ellas quiero dedicar de manera íntegra este trabajo ya que han sacrificado un trocito de su vida conmigo para que yo pudiera dedicarle tiempo a este trabajo. De esto último me di cuenta al ver que mi hija Andrea con sólo cinco años ya pintaba unos retículos estupendos. Os juro que no volveré a nombrar esta palabra en casa.

De igual modo, el apoyo de mi familia, y en concreto de mis padres, ha resultado fundamental, siendo sus consejos y ánimos imprescindibles para superar esos momentos complicados en los que uno no sabe muy bien hacia donde se dirige. La inquietud por los ordenadores y la justificación del porqué estoy hoy aquí se la debo a ellos y al ZX Spectrum que me regalaron esas Navidades de los años 80.

En el plano profesional son muchas las personas que me han apoyado todo este tiempo, sin su ayuda este trabajo nunca habría finalizado. El proceso de desarrollo de esta Tesis ha sido complicado y ha pasado por un gran número de etapas de las que me gustaría destacar especialmente dos. De la primera, me gustaría agradecer de manera muy especial a Alfredo Fernández-Valmayor y Baltasar Fernández Manjón (y por extensión a todo el Departamento de Sistemas Informáticos y Programación de la UCM) sus consejos y apoyo como directores de este trabajo durante sus primeros años. Ellos me descubrieron el mundo de la Recuperación de Información cuando Internet no dejaba de ser una mera anécdota y me introdujeron en las técnicas que han hecho posible la materialización de este trabajo. En una segunda etapa, el apoyo de mis directores actuales Julio Gonzalo Arroyo y Anselmo Peñas Padilla ha sido decisivo para acotar el problema a resolver, permitiéndome focalizar

todo el conocimiento adquirido en los primeros años a la mejora de una tarea de recuperación de información que, de acuerdo a las hipótesis de partida, podría resultar inviable. Ellos me han aportado el conocimiento necesario para entender y plantear correctamente las medidas de evaluación propuestas en este trabajo, así como para el desarrollo de todas las propuestas para la extracción y selección presentadas. Su profesionalidad, objetividad y disposición me han facilitado mucho el camino en aquellos momentos en los que no sabía por dónde continuar. Julio, Anselmo: Finalmente me habéis hecho ver la luz al final del túnel! Gracias.

También me gustaría agradecer de manera especial el apoyo prestado por la Directora del Departamento de Lenguajes y Sistemas Informáticos Felisa Verdejo. En primer lugar por sus consejos, aportaciones, críticas y sugerencias a lo largo del desarrollo de este trabajo, así como por facilitarme en todo momento la tarea de redacción de esta Tesis. En segundo lugar, por confiar en mí y permitirme trabajar junto a un equipo de profesionales tan competente que, en ningún momento, ha dudado en aportar nuevas ideas al trabajo, sugerir mejoras o soluciones y que, en definitiva, me han aguantado a lo largo de todos estos años (que no es poco).

A Fernando López, Covadonga Rodrigo, Enrique Amigó, Ignacio Mayorga, Victor Peinado, Javier Artiles, Miguel Rodriguez, Raquel Martinez, Tim Read, Valentin Sama, Alvaro Rodrigo, Yolanda Calero, Emilio Lorenzo, Carlos Celorrio, Javier Vélez, Lourdes Araujo, José Luis Delgado, David Fernández, Jesús Herrera, Victor Fresno y nuestra inestimable secretaria Fátima Gil. Muchas gracias!

A todas las personas que he nombrado y a todas aquellas que, aun habiéndose quedado en el tintero no son por ello menos importantes. Muchas gracias. Al final lo he conseguido!

Resumen

En este trabajo se presenta una aproximación para la organización de resultados de búsqueda mediante Análisis Formal de Conceptos (AFC), aplicable a escenarios de Recuperación de Información (RI) tales como la búsqueda web. Este trabajo aborda cuatro aspectos principales:

- La definición de un modelo de clustering basado en AFC. La aplicación de esta teoría presenta las ventajas de permitir herencia múltiple sobre los clusters obtenidos y de realizar la descripción de los mismos de manera automática. Además, incluimos la noción de *nodo de información* con el fin de obtener estructuras de clustering que consideren una aproximación basada en un *Universo Abierto* para los documentos agrupados

- La definición de una metodología orientada a la aplicación del modelo sobre escenarios de RI reales. Para cada uno de los procesos involucrados en la construcción del clustering proponemos una serie de alternativas. Debemos destacar el uso de n-gramas para la descripción de los clusters, el uso de un *algoritmo balanceado* en el proceso de selección de descriptores para minimizar la población de documentos en el cluster raíz o la aplicación de Latent Semantic Indexing (LSI) para detectar relaciones descriptor-documento no explícitas.

- La definición de paradigmas para la visualización e interacción sobre las estructuras de clustering. El uso de retículos para representar la información agrupada complica el proceso de visualización ya que los paradigmas habitualmente utilizados para representar estructuras de clustering no resultan adecuados. En este trabajo se presentan dos aproximaciones al problema de la visualización cuya característica principal es la de explotar la estructura intrínseca de los retículos obtenidos. *La visualización basada en retículos* toma como punto de partida los diagramas de Hasse pero reduce el número de clusters visualizados por el usuario en cada momento. De este modo, únicamente se muestran aquellos clusters que, por ser más próximos al cluster que se está inspeccionando, podrían resultar útiles para refinarlo o generalizarlo. En segundo lugar, se propone una *visualización basada en directorios web* que, aprovechando un paradigma sobradamente conocido por los usuarios, permite mapear la estructura de un retículo de manera sencilla.

- La definición de un conjunto de medidas de evaluación orientadas a evaluar automáticamente la calidad, en una tarea de RI, de un sistema de clustering basado en retículos. Estas medidas están basadas en los conceptos de *área de navegación mínima* (MBA) y de *coste cognitivo*.

El primero representa el conjunto mínimo de clusters y enlaces que el usuario debe recorrer para acceder a toda la información relevante recuperada. El coste cognitivo permite introducir en las medidas de evaluación una estimación del esfuerzo que el usuario debe realizar para considerar como relevante un determinado item (un cluster o un documento) en base a su descripción. El *factor de destilación* (DF) únicamente tiene en cuenta el coste cognitivo asociado a la exploración de los documentos y, por lo tanto, no refleja el esfuerzo asociado a explorar el clustering. Esta medida informa acerca del grado de mejora de precisión del retículo con respecto a la lista ordenada de documentos. La *calidad del clustering* (CQ), sin embargo, extiende la medida anterior al incluir el coste cognitivo asociado a considerar las descripciones de los clusters, lo que permite evaluar de manera mucho más precisa la influencia de la estructura de clustering en el proceso de RI.

Finalmente, y con el objeto de demostrar la viabilidad de nuestra propuesta, así como de las medidas de evaluación, hemos desarrollado cuatro prototipos de los que tres de ellos han sido evaluados automáticamente. Los tres prototipos realizan el proceso de clustering sobre la colección de noticias EFE94. Esta colección ha sido utilizada en distintas campañas CLEF (Cross Language Evaluation Forum) y dispone de un amplio conjunto consultas con juicios de relevancia asignados manualmente por expertos. Debemos destacar que, en todos los experimentos realizados, los resultados obtenidos generaron estructuras de clustering cuyos valores de calidad (que mejoraban notablemente la precisión inicial de la lista de documentos recuperada) justificaban la realización del proceso de clustering.

El primero de los prototipos describe los clusters utilizando unigramas y lleva a cabo su selección mediante las aproximaciones $tf - idf$ y terminológica. Debemos destacar como resultado relevante que el hecho de aumentar el número de descriptores no mejora proporcionalmente la calidad del clustering.

En el segundo prototipo experimenta con sintagmas terminológicos para describir los clusters y utiliza un algoritmo balanceado como estrategia de selección. Como resultados destacables debemos remarcar la mejora sustancial del algoritmo de selección balanceado frente al algoritmo terminológico (204.3 % para la medida DF), así como capacidad para concentrar una gran cantidad de documentos no relevantes en el cluster raíz.

El tercer prototipo utiliza n-gramas para describir los clusters y aplica la estrategia de selección balanceada. Los resultados obtenidos demuestran que la aplicación de LSI mejora la calidad del clustering, aunque no de manera significativa (un 9.5 % con respecto a la misma aproximación sin considerar LSI). El uso de snippets, por otro lado, disminuye levemente la calidad de las estructuras de clustering generadas, aunque su capacidad para agrupar correctamente la información relevante (mejora la precisión con respecto a la lista inicial de documentos en un factor $3{,}48$) nos permite concluir que es una aproximación adecuada. De hecho, este resultado permite mostrar la validez de toda nuestra propuesta para ser implementada en sistemas on-line que obtengan los resultados de búsqueda de sistemas comerciales, generando de manera efectiva las estructuras de clustering. El sistema Jbraindead, presentado como prototipo final de este trabajo, lo demuestra.

Abstract

This work introduces a new approach to the organization of search results using Formal Concept Analysis (FCA). This approach is applicable to Information Retrieval (IR) scenarios such as web search. To define a complete framework to hold the proposal, this work solves four main aspects:

- The definition of a clustering model based on FCA. This theory has several advantages such as multiple inheritance for clusters and also their automatic description. We also define the notion of *information node* to get clustering structures that consider an *Open Universe* view for the clustered documents.

- The definition of a methodology to apply the model into real IR scenarios. We define several alternatives for each of the phases involved into the clustering process. We have to remark the use of n-grams for cluster description, the use of a *balanced algorithm* to select the best descriptors which minimize the number of documents in the root cluster or the application of *Latent Semantic Indexing* (LSI) to find non explicit descriptor-document relationships.

- The definition of paradigms to visualize and interact with the clustering. Lattices are complex structures which do not fit well in the visualization paradigms commonly applied in other hierarchical clustering approaches. In this work we introduce two new visualization approaches that exploit the intrinsic structure of a lattice. *Lattice based visualization* uses the notion of Hasse's diagrams but it reduces the number of clusters that the user is currently viewing. Only those clusters closer to the currently explored cluster will be shown. Secondly, we propose a *web directory based visualization* approach. It uses a well known visualization paradigm to map the lattice into.

- The definition of evaluation measures. They automatically evaluate the clustering quality form the point of view of a IR task. These measures are based on the definitions of Minimal Browsing Area (MBA) and cognitive cost. The first one represents the minimal set of clusters that the user has to traverse in order to find all the relevant information. The cognitive cost estimates the user's effort to consider an specific item (a cluster or a document) as relevant. The Distillation Factor (DF) only takes into account the cognitive effort associated to the documents but it does not reflects the cost associated to explore the clusters. DF can be understood as the precision improvement of the lattice with respect to the retrieved document list. On the other hand, the Clustering Quality (CQ) extends DF considering the cognitive cost associated

to explore the cluster's description. This allows to evaluate more precisely the improvement of the clustering structure in a IR process.

Finally, to demonstrate the viability of our proposal and the evaluation measures, we have developed four prototypes. We have evaluated three of them. They make the clustering process using the EFE94 news corpus. This corpus has been used in different CLEF (Cross Language Evaluation Forum) campaigns and it has a wide set of queries with relevance judgments manually created by experts. We have to remark that all the experiments gave good evaluation results which improved the precision of the initial retrieved document set.

First prototype describes the clusters using unigrams and the selection process is based on $tf - idf$ and terminological approaches. We evaluate the selection process and the number of selected descriptors in the final quality of the clustering. As a relevant result, to increase the number of descriptors does not proportionally increases the clustering quality.

Second prototype uses noun phrases as cluster's descriptors and the selection process is based on the balanced algorithm. In this case we evaluate the selection process and the importance of the root cluster in the final quality of the clustering. We also compare an approach based on unigrams versus noun phrases. Relevant results are a good behavior of the balanced algorithm, improving a $204{,}3\,\%$ the DF value with respect the terminological algorithm, and its quality to concentrate non relevant documents in the root cluster.

Finally, the third prototype experiments with noun phrases to describe the clusters and the balanced algorithm. Experiments are focused on the evaluation of formal context enrichment by using LSI and the increment of clustered documents. Finally we also experiment with the use of snippets instead of full text documents to perform the extraction process. Results demonstrate that the use of LSI improves the quality of the clustering but this improvement is not very significant ($9{,}5\,\%$ with respect the same approach without LSI). The use of snippets slightly reduce the mean precision of the clustering structures but without losing their ability to separate the relevant documents (in fact, this approach improves in a $3{,}48$ factor the precision with respect to the ranked list). This result allows to demonstrate the validity of our approach to be implemented in on-line systems which retrieve search results directly from commercial search engines. JBraindead system, that is introduced as final prototype, demonstrates this point.

Índice general

Índice de figuras

Índice de cuadros

Capítulo 1

Introducción

La gran mayoría del conocimiento humano se encuentra accesible en forma textual siendo el desarrollo de técnicas para almacenar, organizar y buscar este tipo de información casi tan antiguas como el propio lenguaje escrito. La aparición de las primeras computadoras permitió dar los primeros pasos hacia su digitalización, surgiendo la necesidad de aplicar estas técnicas en el ámbito computacional. La constante mejora en el hardware utilizado, así como la aparición de Internet y su rápida difusión social, han hecho que actualmente la cantidad de información accesible en formato electrónico (textual o multimedia) haya crecido de forma exponencial, haciendo imprescindible el desarrollo de técnicas eficientes para su búsqueda, acceso, filtrado y organización. Las Tecnologías de la Información (TI) pretenden dar solución a estos nuevos retos, poniendo a disposición de los usuarios sistemas que permitan manejar la gran cantidad de información disponible de acuerdo a necesidades de búsqueda concretas.

Esta Tesis Doctoral se sitúa en el escenario descrito, centrándose en una propuesta para la organización automática de resultados de búsqueda basada en el Análisis Formal de Conceptos (AFC) y siendo su fin último la mejora del acceso a la información relevante para una tarea de recuperación de información. Con el fin de hacer nuestra propuesta extensible e integrable en cualquier dominio y sistema de recuperación de información propondremos una metodología donde cada uno de los procesos involucrados en la organización de la información recuperada será exhaustivamente descrito. Finalmente, abordaremos el problema de la evaluación presentando un conjunto de medidas orientadas a la tarea que se resuelve.

El objetivo de este primer capítulo es motivar al lector presentando la problemática asociada a este trabajo, sus objetivos, y la metodología que aplicaremos. En primer lugar justificaremos en qué casos una aproximación orientada a la organización de la información es válida, explicando las ventajas frente a una aproximación clásica de recuperación de información. Así mismo, motivaremos las ventajas de utilizar una aproximación basada en AFC frente a otras propuestas de clustering de documentos. A continuación, plantearemos los objetivos principales de esta Tesis Doctoral y la metodología de trabajo que hemos aplicado para conseguirlos. Finalmente, describiremos la estructura del documento.

1.1. Motivación

Los sistemas de Recuperación de Información (RI) tienen como objetivo esencial facilitar el acceso a la información, proporcionando al usuario aquellos documentos que más se adecuan a unas necesidades de búsqueda concretas. Un sistema de recuperación de información procesa el contenido textual de los documentos para obtener una representación interna que es utilizada posteriormente para proporcionar los documentos más adecuados a las consultas formuladas por los usuarios. Sin embargo, y aunque el desarrollo que éste área ha experimentado en los últimos años no tiene precedentes, cada vez resulta más evidente que el paradigma de acceso a la información aplicado en estos sistemas no siempre satisface las necesidades de información de un usuario final.

Esto es debido a que un sistema clásico de RI presenta el conjunto de documentos recuperados como una lista ordenada de acuerdo a un ranking de relevancia. Esta aproximación, que puede encontrarse en prácticamente todos los motores de búsqueda web [59, 139, 91, 8, 69], está basada en un paradigma de exploración secuencial. Comenzando por los documentos que se encuentran en las posiciones más altas del ranking (i.e que son los más relevantes de acuerdo a los criterios del sistema) el usuario debe descender por la lista hasta haber satisfecho sus necesidades de información.

Esta estrategia de acceso a la información es adecuada en escenarios donde el usuario tenga una necesidad de información muy precisa, que pueda expresar claramente y cuya respuesta pueda encontrar en unos pocos documentos (e.g. la búsqueda de un artículo científico por su título, la búsqueda de un driver concreto para un sistema operativo, etc.). Sin embargo, existen otras situaciones donde este paradigma de acceso a la información no permite al usuario encontrar la información relevante en los primeros puestos del ranking, haciendo necesario que éste tenga que inspeccionar, analizar y filtrar manualmente una buena parte del conjunto de documentos recuperado.

De todos los posibles escenarios en los cuales esta estrategia de acceso a la información no proporciona los resultados óptimos, en este trabajo nos centraremos en los siguientes:

- *Escenario 1: Necesidad de búsqueda imprecisa*. Existen ocasiones en las que el usuario no sabe exactamente qué es lo que está buscando o cómo expresar su necesidad de búsqueda de forma adecuada y precisa. En estas situaciones, lo más habitual es que el usuario realice una consulta genérica, esperando que el sistema le proporcione información suficiente como para poder acceder a información relevante, identificar categorías en la información buscada o reformular su consulta de manera mucho más precisa. Sin embargo, un sistema de RI tradicional le proporcionará una extensa lista de documentos descritos por su título y un pequeño fragmento de texto (i.e. snippet) que éste deberá inspeccionar y filtrar manualmente para determinar qué información es relevante a sus necesidades de búsqueda.

  Un ejemplo de este escenario podría ser el siguiente. Supongamos que un usuario desea buscar en Internet información relacionada con un cantante famoso, pero sin una necesidad concreta de información. En otras palabras, el usuario desea obtener información que le permita acceder, por ejemplo, a datos biográficos, foros de interés, anuncios de próximos conciertos, sitios web de venta de entradas, páginas web creadas por usuarios anónimos acerca del cantante, páginas de descargas de conciertos y música, etc. El uso de un motor de RI clásico le proporcionará en primera instancia los diez documentos más relevantes, siendo necesaria

una exploración exhaustiva del conjunto de resultados para poder acceder a la información buscada. En este sentido, un sistema capaz de sugerir al usuario un conjunto de categorías relacionadas con el cantante buscado y de agrupar los documentos relacionados de acuerdo a éstas sería deseable.

- *Escenario 2: Uso de términos ambiguos*. En otras ocasiones, la consulta realizada puede incluir términos ambiguos. Consultas de este tipo hacen que un sistema de RI clásico recupere documentos no sólo relacionados con el objetivo de búsqueda, sino también con otros dominios de conocimiento. Además, dependiendo del algoritmo de ranking utilizado para ordenar la lista, los documentos buscados por el usuario podrían no aparecer en las primeras posiciones del mismo. Frente a esta situación el usuario dispondría de dos alternativas para acceder a la información relevante: a) explorar la lista proporcionada por el sistema hasta dar por satisfechas sus necesidades de información, o; b) desistir de la estrategia de búsqueda iniciada y tratar de ajustar la consulta con términos más específicos y orientados a sus necesidades concretas de búsqueda (e.g. expandiendo la consulta inicial con términos relacionados). La primera de las alternativas no tiene porqué producir una mejora sustancial ya que el número de documentos relevantes puede ser muy pequeño y encontrarse diseminado a lo largo de toda la lista de documentos recuperados. Esto, en cualquier caso, no justifica la inversión de tiempo y esfuerzo para recorrer y filtrar manualmente la lista. Respecto a la segunda de las alternativas, y aunque inicialmente pudiera resultar la más adecuada, ésta supone un conocimiento explícito del proceso de búsqueda necesario para la selección de los nuevos términos que no siempre se encuentra al alcance de todos los usuarios. Además, una reformulación incorrecta de la consulta podría descartar información relevante sin que el usuario llegue a percatarse de ello. En esta línea, muchos motores de búsqueda web proporcionan la posibilidad de expandir automáticamente la consulta en base a la selección de un documento como relevante. Aún así, este tipo de aproximaciones, cuya mejora en la eficiencia del proceso de recuperación está demostrada, son difíciles de comprender por los usuarios habituales de este tipo de sistemas.

  Para ejemplificar este escenario supongamos que un usuario desea encontrar en Internet páginas web relacionadas con los *jaguares*. Realizando esta consulta en un motor de búsqueda web éste obtendrá una gran cantidad de documentos de los cuales sólo unos pocos estarán relacionados con el objetivo de búsqueda. Esto es debido a que el término utilizado para realizar la consulta no es sólo aplicable en el contexto animal, sino que también puede ser aplicado al dominio de los *automóviles*, *consolas de videojuegos* o el de la *música* entre otros. En este escenario, el motor de búsqueda no es capaz de discriminar el dominio sobre el cual se desea realizar la búsqueda, mostrando al usuario un conjunto de resultados heterogéneo y ordenado de acuerdo a los criterios de ranking del buscador. Nuevamente, una situación de este tipo hace necesario que el usuario explore y filtre una gran cantidad de información para obtener los resultados relevantes. Un sistema capaz de identificar los diferentes dominios, describirlos con un conjunto de términos clave y agrupar los documentos relevantes sería deseable.

- *Escenario 3: Necesidad de compilar información*. Finalmente, en aquellos casos en los que el usuario pretenda encontrar documentos relevantes para realizar una tarea de extracción y

compilación de información (proceso conocido como *síntesis de información* [4]), el uso de un sistema de RI clásico tampoco resultará de gran ayuda. El hecho de obtener un conjunto de documentos extenso supone la clasificación y filtrado manual de los mismos con el fin de identificar documentos similares, detectar relaciones entre los documentos o seleccionar documentos originales y con contenido adecuado para realizar el proceso de síntesis.

Por ejemplo, supongamos que un usuario desea escribir un informe sobre las posibilidades de los sistemas de calefacción por biomasa. Para ello necesita una descripción del concepto de biomasa, de su impacto económico y una comparativa con otro tipo de energías. Al igual que ocurría en los escenarios anteriores, un motor RI clásico proporcionará una larga lista de páginas web relacionadas con la consulta que deberán ser exploradas, clasificadas y relacionadas manualmente por el usuario para poder realizar la tarea de síntesis.

Los tres escenarios propuestos tienen en común la necesidad de explorar, describir y organizar manualmente los resultados de búsqueda recuperados. Esta situación convierte el proceso de acceso a la información en una tarea lenta y tediosa donde la cantidad de información finalmente útil para el usuario no justifica el tiempo y esfuerzo invertidos.

Como solución al problema expuesto, y en el extremo opuesto a los sistemas RI, se encuentran los sistemas de Organización de Información (OI). Este tipo de sistemas utilizan atributos más o menos complejos de la colección para organizarla, poniendo a disposición del usuario paradigmas de visualización adecuados que hagan más sencilla su exploración. La principal diferencia entre un sistema de RI y de OI radica en el rol que desempeñan. El rol de un sistema de RI es activo dado que es el propio sistema el que impone al usuario una estrategia de exploración concreta (i.e. un ranking de documentos) para acceder a la información recuperada, esperando de éste que siga el orden propuesto para encontrar la información relevante. Los sistemas de OI son más relajados y su rol puede considerarse pasivo. El sistema organiza la información y es el usuario el que, aplicando sus propios criterios de exploración, decide qué documentos consultar y en qué orden.

Existen un gran número de modelos orientados a la organización de información [34, 3, 39, 82] capaces de representar y visualizar las relaciones existentes entre los documentos, sus términos e incluso las consultas asociadas a los mismos, haciendo uso de técnicas de agrupación de documentos que reciben el nombre genérico de *técnicas de clustering* o *técnicas de agrupación*[1]. Las investigaciones realizadas en este campo han experimentado un notable aumento en los últimos años, siendo revisiones como las de [70, 135, 61], y más recientemente la de Weili Wu [128], un excelente punto de partida para conocer el estado del arte, tanto desde el punto de vista de las técnicas y los métodos como de sus aplicaciones en sistemas de acceso a la información plenamente funcionales.

Basados en la *Hipótesis de Clustering* [119]:

'Closely associated documents tend to be relevant to the same requests'

[1] Dado que el término *clustering* se utiliza de manera frecuente sin traducir, en este trabajo utilizaremos indistintamente tanto el término en inglés como su correspondiente traducción al castellano.

Croft [33] y, más recientemente Hearst y Pedersen [67], propusieron aplicar técnicas de clustering no sólo sobre una colección completa sino, como alternativa, sobre un subconjunto de ésta obtenido como resultado de un proceso previo de recuperación. De este modo se pretende paliar el alto coste computacional que suponen este tipo de técnicas, aplicando el proceso únicamente a una pequeña parte de la colección completa. Como ventaja añadida, el proceso de generación del clustering se convierte en un proceso dinámico y adaptable a las necesidades específicas del usuario que tiene en cuenta la consulta de éste y el subconjunto de documentos recuperado.

Tanto Croft como Hearst se limitaron a hacer la propuesta, aunque no estudiaron empíricamente su impacto en la mejora de la precisión para este tipo de sistemas y, de hecho, Voorhes [125] no pudo encontrar evidencias concluyentes que apoyaran la propuesta inicial de Croft. De todos modos, y a pesar de que en la actualidad no existan evidencias experimentales, numerosos estudios y sistemas [67, 93, 77, 122, 113, 32, 90] apuntan al uso de técnicas de clustering como una aproximación plausible a la organización de los resultados proporcionados por un sistema de recuperación de información. Consideramos que una aproximación de este tipo resulta especialmente adecuada en los escenarios descritos en los párrafos anteriores, y sobre los cuales la utilidad de los sistemas de RI clásicos no proporciona resultados óptimos.

El trabajo presentado en esta Tesis Doctoral se basa en los planteamientos de Hearst y Pedersen. Consideramos que la construcción de un sistema de acceso a la información eficiente y flexible debería ser capaz de combinar de manera adecuada aproximaciones basadas en recuperación y organización de la información, aprovechando las virtudes de cada una de ellas. Tal y como hemos expuesto, en la mayor parte de las ocasiones el usuario no hace uso de un motor de búsqueda con el fin de acceder a un documento específico, sino que pretende compilar conocimiento a partir del conjunto de documentos recuperados para realizar una tarea más compleja. De hecho, en [106] se afirma que alrededor del 60 % de las búsquedas en Internet son informativas, sugiriendo que este tipo de combinaciones son las más adecuadas para satisfacer las necesidades de información del usuario. Este dato estadístico, unido a la tendencia actual del sector comercial [122, 113, 32, 90] avalarían nuestra propuesta.

Considerando la tarea de acceder a la información relevante recuperada por un motor de búsqueda, esta Tesis Doctoral presenta un modelo para la organización de los resultados de búsqueda cuyo objetivo es el de *reducir la cantidad de información no relevante que el usuario debe considerar para completar la tarea, minimizando el coste cognitivo asociado al proceso de exploración de la estructura generada*. De acuerdo a este planteamiento, nuestro trabajo no sólo pretende obtener un clustering eficiente (donde los documentos relevantes estén correctamente agrupados), sino también generar estructuras adecuadas que proporcionen suficiente información como para adquirir un conocimiento explícito de la información organizada sin necesidad de acceder al contenido de los documentos concretos.

Aunque nuestra propuesta combina una aproximación basada en RI y en OI, en este trabajo nos centraremos en los aspectos relacionados con el proceso de clustering. Esto es debido a que, actualmente, las técnicas de RI se encuentran lo suficientemente desarrolladas como para recuperar de forma precisa una gran cantidad de información, no siendo relevante nuestra aportación en este área. En el área del clustering, sin embargo, nuestro trabajo propondrá alternativas que podrían

clasificarse dentro del conjunto de técnicas de clustering jerárquico aunque, como justificaremos en los siguientes capítulos, este punto debe ser matizado. En este sentido, y en comparación con otras aproximaciones (experimentales y comerciales presentadas en la sección 2.4.1) desarrolladas en paralelo al trabajo aquí presentado, nuestra aportación en éste área propone alternativas en el proceso de clustering orientadas a mejorar los siguientes aspectos:

1. *El uso de herencia simple para construir el clustering*. Habitualmente un sistema de clustering jerárquico realiza una partición disjunta del espacio de documentos, haciendo que exista únicamente una secuencia de especialización para acceder a un documento específico. Una aproximación de este tipo considera herencia simple a la hora de agrupar la información, lo que implica que cada documento únicamente puede ser considerado en un contexto concreto. Una solución de este tipo resulta adecuada cuando el proceso de clustering se aplica sobre dominios disjuntos donde no existan documentos que puedan pertenecer a más de uno de ellos. Sin embargo, este no es el caso más habitual, siendo muy frecuentes las situaciones en las que un documento puede asociarse a más de un cluster (consultar sección 5.2.2). Las limitaciones de una clasificación de este tipo ya resultan evidentes en los sistemas de ficheros tradicionales donde, en muchas ocasiones, el usuario debe elegir entre un conjunto de carpetas dónde almacenar un fichero concreto que, por sus características específicas, podría guardarse en más de una de ellas. Por ejemplo, un documento que contenga la *presentación de un proyecto* podría almacenarse en una carpeta *proyectos* o, también, en una carpeta *presentaciones*. Como alternativa, considerar herencia múltiple a la hora de generar el clustering resolvería este problema, permitiendo que un mismo documento pueda pertenecer a varios contextos y que, por lo tanto, existan varias secuencias de especialización para acceder a éste.

2. *La consideración de un universo cerrado para generar el clustering*. Las aproximaciones de clustering jerárquico habitualmente optan por incluir en cada cluster, no sólo los documentos descritos de manera completa por el conjunto de descriptores asociados al mismo, sino también todos aquellos documentos pertenecientes a los clusters que lo especializan (consultar 5.2.1). En estos casos, se considera el espacio de información como un *universo cerrado* (i.e. similar al de las taxonomías de biología) donde si un elemento pertenece a un cluster específico, éste también pertenecerá a alguno de sus clusters descendientes. Consideramos que una aproximación de este tipo no resulta adecuada en los casos en los que el usuario desea acceder únicamente a los documentos descritos completamente por los descriptores del cluster. En estos casos, éste tendrá que filtrar manualmente los documentos del cluster con el fin de descartar aquellos que son especializados en los clusters descendientes. Como alternativa, una aproximación basada en un *universo abierto*, donde cada cluster contenga únicamente aquellos documentos descritos de manera completa por los descriptores del cluster resultaría más adecuada. Esta aproximación es utilizada por los directorios web (como ODP o Yahoo!), donde cada categoría únicamente contiene los documentos descritos por los términos que identifican la categoría, siendo necesario seleccionar una subcategoría para acceder a un conjunto de documentos más específico. Un ejemplo típico podría ser la categoría genérica Física, desde la cual sólo tendríamos acceso a páginas web generales relacionadas con la Física (como por ejemplo enciclopedias on-line). Si quisiéramos acceder a documentos

más específicos relacionados con la Astronomía, deberíamos seleccionar esta subcategoría en el directorio.

3. *La descripción de los clusters a posteriori*. La mayor parte de las técnicas de clustering se caracterizan por describir los clusters una vez que éstos han sido generados, lo que implica realizar un análisis exhaustivo de cada cluster a posteriori. Una aproximación de este tipo no sólo aumenta el coste computacional del proceso (es necesario extraer independientemente los descriptores de cada cluster), sino que condiciona el conjunto de descriptores generados al clustering obtenido. Consideramos que un análisis a priori de la información recuperada para determinar cuales son los descriptores más relevantes y utilizarlos tanto para guiar el proceso de clustering como para describir sus resultados finales constituye una aproximación novedosa cuyos resultados desarrollaremos en este trabajo.

4. *Visualización del clustering*. Finalmente, la mayor parte de los sistemas de clustering jerárquico aplican un paradigma de visualización y navegación basado en árboles (consultar sección 2.4.1). Una aproximación de este tipo es adecuada cuando el proceso de clustering considera herencia simple sin embargo, el hecho de considerar herencia múltiple, hace que este paradigma no sea el más adecuado. Esto es debido a que, en estos casos, el usuario percibe un espacio de información disjunto, lo que dificulta la tarea de conceptualzar e interiorizar las relaciones entre los documentos agrupados.

El trabajo presentado pretende aportar alternativas que tengan en consideración los aspectos expuestos. Para ello, aplicaremos la teoría del Análisis Formal de Conceptos (AFC) para generar estructuras de clustering adecuadas. Los fundamentos matemáticos de esta teoría [53], que serán expuestos en detalle en el capítulo 3, permiten disponer de un marco teórico sólido y formalmente demostrado en el que apoyar nuestra propuesta y cuyas principales ventajas como técnica de clustering son las siguientes:

1. *Consideración de herencia múltiple para construir el clustering*. El AFC puede entenderse como una teoría que permite organizar y clasificar automáticamente la información sobre un retículo, lo que implica que cada elemento de esta estructura puede tener más de un antecesor. Por su naturaleza, los retículos favorecen la implementación de un clustering basado en herencia múltiple de una manera sencilla y elegante, que además es soportada por la propia teoría. Una aproximación basada en clustering jerárquico que considere herencia múltiple debería duplicar en más de un cluster aquellos documentos con varios antecesores. En contraste, el hecho de utilizar un retículo evita esta duplicidad de información y permite trabajar con un único cluster que estará relacionado con el conjunto de antecesores. Finalmente, una estructura de este tipo resulta adecuada desde el punto de vista de la navegación y la conceptualización de la información dado que facilita la identificación de aspectos comunes a dominios disjuntos, así como la recuperación de decisiones de exploración incorrectas o inadecuadas.

2. *Consideración de un universo abierto a la hora de realizar el clustering*. Los *conceptos formales* son los actores principales en la teoría del AFC y se definen como pares formados por una extensión (conjunto de objetos) y una intensión (conjunto de atributos) que la describe

(consultar capítulos 3 y 5). En nuestra propuesta, los conceptos formales serán los clusters, definidos por un conjunto de documentos (extensión) y descritos por un conjunto de descriptores (intensión). Tal y como se plantea la definición de concepto formal, éste representa la información de acuerdo a un universo cerrado, ya que es posible que alguno de los elementos de la extensión pueda especializarse en conceptos formales más específicos. Por esta razón, en el modelo propuesto (capítulo 5) deberemos redefinir alguno de los conceptos de la teoría del AFC con el fin de considerar una aproximación basada en un universo abierto, tal y como justificamos anteriormente.

3. *Descripción automática de los clusters*. La teoría del AFC parte de un contexto formal como base para generar el conjunto de conceptos formales. La creación de este contexto requiere la identificación previa de un conjunto de objetos y atributos que, en nuestro caso serán los documentos y un conjunto de descriptores relevantes asociados al conjunto de documentos recuperado. Partiendo de esta información, las técnicas de AFC permitirán generar de manera automática la descripción intensional de todos los clusters, que se obtendrá mediante combinaciones del conjunto de descriptores seleccionado al comienzo del proceso. De este modo, y en contraste con las aproximaciones de clustering jerárquico, la aplicación de AFC no hace necesario describir de manera explícita cada cluster, sino que las descripciones serán generadas automáticamente como parte del proceso de clustering.

No obstante, la aplicación de AFC para resolver el problema de clustering propuesto no es una tarea sencilla e implica abordar una serie de retos para integrar la tarea de RI en el contexto descrito.

- *Problema de la selección de descriptores*. Tal y como describiremos en el capítulo 4, la mayor parte de las aplicaciones del AFC en el área de la RI han basado la construcción del retículo en un conjunto de descriptores generados manualmente o en el uso de tesauros. Nuestra propuesta pretende extraer dicha información a partir de texto libre, lo que supone el desarrollo de técnicas que, no sólo permitan extraer un conjunto de descriptores relevantes, sino también la selección de aquellos más adecuados para la construcción del retículo (desde el punto de vista informativo y desde el punto de vista computacional). Estas técnicas serán descritas y exploradas en las secciones 6.3 y 6.4.

- *Problema de la visualización*. El hecho de utilizar un retículo como estructura subyacente para la organización de la información supone una mejora considerable en el grado de relación entre el conjunto de documentos recuperado. Aún así, este tipo de estructuras pueden llegar a ser muy complejas para ser mostradas directamente al usuario. En este trabajo propondremos alternativas (sección 6.7) para su visualización y exploración que, sin reducir su capacidad informativa, faciliten al usuario su comprensión e interacción.

- *Problema de la evaluación*. Finalmente, el hecho de utilizar un retículo como estructura de clustering, así como aplicarlo a una tarea de RI hace que medidas clasicas, habitualmente utilizadas para evaluar sistemas de clustering no puedan aplicarse. A esto debemos añadir que, hasta la fecha, el área del AFC no ha propuesto ningún marco de evaluación que permita

abordar este problema. En este trabajo (capítulo 7) presentaremos un conjunto de medidas que permitan resolverlo.

Como adelanto a alguno de los resultados obtenidos en este trabajo, y con el fin de que el lector pueda hacerse una idea de las principales diferencias entre nuestra propuesta y otras aproximaciones de clustering existentes, la figura 1.1 muestra el resultado de la consulta Madonna en el sistema JBraindead (que será presentado y discutido en detalle en la sección 9.4) desarrollado como último de los prototipos de esta Tesis Doctoral. El hecho de utilizar un retículo supone una mejora en el modo en el que la información se encuentra relacionada y en el modo en el que el usuario percibe dichas relaciones. En concreto, en el ejemplo presentado, podemos observar como el sistema ha detectado, entre otras, las categorías *Madonna's Bibliography*, *Madonna's Pictures* y *Madonna's Lyrics*, siendo la propia estructura del retículo la encargada de definir clusters mucho más específicos a partir de la combinación de los tres descriptores presentados (herencia múltiple). De este modo, el usuario puede identificar un cluster que combina los descriptores *Madonna's Bibliography* y *Madonna's Pictures*, pudiendo generalizar su búsqueda hacia dos clusters disjuntos que contengan información relacionada sólo con uno de ellos (universo abierto). En una aproximación basada en jerarquías, este modo de agrupar la información no sería factible y sería necesario mostrar al usuario dos caminos distintos hacia dos nodos independientes que representarían el mismo cluster.

Finalmente, el ejemplo ilustra la problemática de evaluación a la que nos enfrentamos. Esto es debido a que, al existir numerosos caminos a través de los cuales se puede acceder a la información relevante, resulta complejo decidir cual de ellos es el óptimo desde el punto de vista de una tarea de recuperación de información.

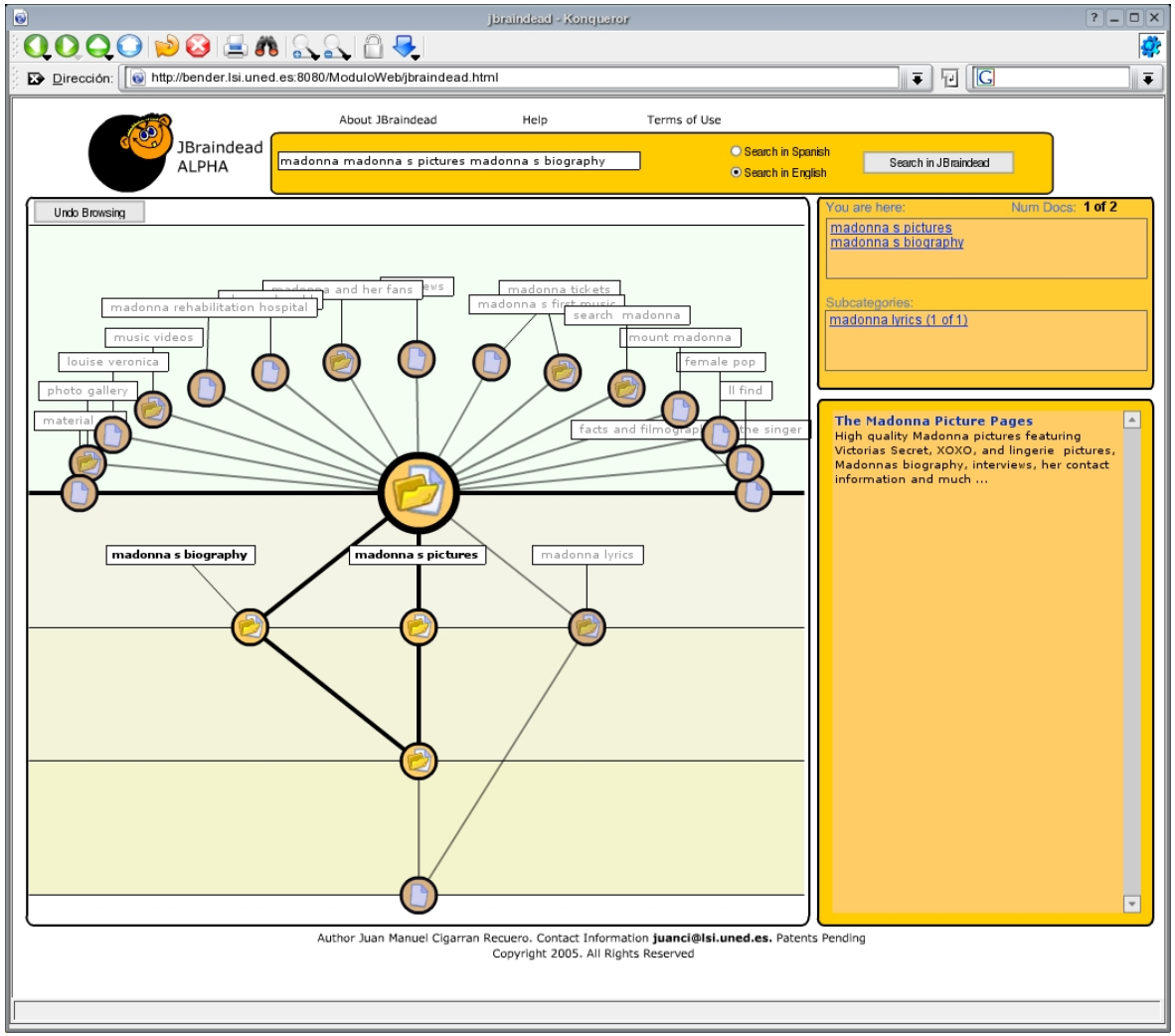


Figura 1.1: Clustering obtenido con el sistema JBraindead para la consulta *'Madonna'*

1.2. Objetivos

El objetivo principal de esta Tesis Doctoral se centra en estudiar cómo aplicar la teoría del AFC para mejorar la organización de los resultados de búsqueda para una tarea de recuperación de información genérica, como puede ser la búsqueda Web.

Para ello, en este trabajo abordaremos la solución a tres problemas que, a nuestro entender, resultan fundamentales a la hora de definir un marco completo para conseguir el objetivo marcado:

- En primer lugar definiremos el modo en el AFC puede ser aplicado a una tarea de recuperación de información basada en clustering de documentos. Para ello, partiremos de los fundamentos matemáticos de la teoría del AFC y reformularemos algunas de sus definiciones con el objeto de crear un modelo genérico que permita la obtención de estructuras de clustering basadas en retículos (capítulo 5).
- Dado que la aplicación del modelo descrito requiere de un conjunto de datos de entrada sobre los cuales operar, así como definir el modo en el que los resultados serán visualizados, en segundo lugar presentaremos una metodología que describa estos procesos. De este modo, en este trabajo resolveremos dos problemas principales (capítulo 6):
  - Nos centraremos en describir los métodos para extraer la información necesaria para la construcción del retículo. Dirigiremos nuestros esfuerzos a la propuesta, diseño y aplicación de técnicas orientadas a la extracción y selección de los descriptores necesarios para la construcción del contexto formal inicial. De igual modo, y para mejorar la calidad final de las estructuras de clustering generadas, experimentaremos con técnicas que permitan enriquecer las relaciones descriptor-documento tales como *Latent Semantic Indexing* (LSI).
  - Así mismo, dedicaremos parte del trabajo a proponer alternativas a la visualización de los retículos orientadas a optimizar su potencial informativo de cara al usuario final.
- Finalmente, y dado que consideramos que un trabajo de este tipo debe ser respaldado por un marco de evaluación, abordaremos la definición de un conjunto de medidas específicamente orientadas a la tarea propuesta. Aunque la evaluación de este tipo de sistemas puede hacerse con usuarios reales (mucho más costosa en tiempo y dinero) o de un modo automático, en este trabajo optaremos por proponer un marco de evaluación automático. Una propuesta de este tipo permite una comparación entre sistemas que facilita la toma de decisiones a la hora de implementar diferentes alternativas que afecten a los resultados del proceso de clustering (capítulo 7).

1.3. Metodología

La metodología de trabajo propuesta para la consecución del objetivo principal, así como para la resolución de los tres problemas principales planteados, es la siguiente:

1. Definición de un modelo de clustering basado en Análisis Formal de Conceptos:

   a) Estudio del estado del arte en técnicas de clustering orientadas a tareas de recuperación de información, prestando especial atención a aquellos métodos o aproximaciones en las cuales los clusters sean organizados de acuerdo a una jerarquía.

   b) Estudio de la teoría matemática del Análisis Formal de Conceptos y de sus aplicaciones previas a tareas de recuperación de información y clustering de documentos.

   c) Propuesta de un modelo de clustering basado en Análisis Formal de Conceptos con aportaciones novedosas respecto a las aproximaciones tradicionales de clustering, así como respecto a las aproximaciones propuestas en el área del AFC.

2. Definición de una metodología capaz de agrupar el conjunto de procesos asociados a la construcción de un sistema de clustering basado en nuestro modelo:

   a) Estudio y propuesta de diferentes alternativas para la extracción de información a partir del conjunto de documentos recuperados. En este sentido, se pretenden evaluar aproximaciones que consideren diferentes tipos de descriptores para el proceso de caracterización de los documentos. En concreto, se experimentará con técnicas para la extracción de unigramas, sintagmas terminológicos y n-gramas de longitud variable.

   b) Estudio y propuesta de diferentes alternativas para la selección del conjunto de descriptores óptimo para caracterizar el conjunto de documentos recuperado. De este modo, se pretende proponer un abanico de aproximaciones orientadas a la selección de los mejores descriptores para la construcción de retículos suficientemente simples e informativos.

   c) Estudio y propuesta de diferentes alternativas para la mejora y enriquecimiento de la información proporcionada al modelo para la construcción del retículo. En este sentido se explora el uso de *Latent Semantic Indexing* para extraer información semántica no explícita del conjunto de documentos recuperados con el fin de obtener nuevas relaciones descriptor-documento que permitan mejorar la caracterización final de los documentos.

   d) Estudio y propuesta de diferentes alternativas orientadas a mejorar el proceso de visualización e interacción del usuario con las estructuras de clustering generadas.

3. Definición de un conjunto de métricas de evaluación:

   a) Estudio del estado del arte relativo a evaluación de sistemas de clustering en tareas de recuperación de información, justificando la necesidad de desarrollar un conjunto de métricas especialmente orientadas a nuestra propuesta.

   b) Propuesta y justificación de un conjunto de métricas orientadas a nuestro modelo y a la tarea de recuperación de información que pretendemos resolver en este trabajo.

4. Evaluación y experimentos sobre sistemas reales. Con el fin de demostrar la viabilidad de las propuestas que desarrollaremos en este trabajo, se hace necesario un apartado específicamente orientado a probar tanto el modelo de clustering propuesto como las diferentes aproximaciones presentadas para la construcción de sistemas basados en él. En este sentido, proponemos la realización de las siguientes tareas:

   a) Desarrollo de diferentes prototipos que, basados en el modelo de clustering propuesto, implementen alguna de las aproximaciones de extracción y selección de descriptores presentadas en este trabajo.

   b) Evaluación de los prototipos desarrollados utilizando las medidas de evaluación propuestas. Dado que en este trabajo no presentamos una metodología de evaluación con usuarios reales, únicamente se propone una evaluación automática que nos permita extraer conclusiones acerca de la viabilidad de cada uno de los prototipos, dejando la obtención de los valores con usuarios reales y su discusión respecto a los obtenidos automáticamente para un trabajo futuro.

1.4. Estructura del Trabajo

El trabajo está estructurado en tres partes:

- *Preliminares.* En esta parte de la Tesis Doctoral se abordan los trabajos preliminares en el área del clustering de documentos y de la teoría del Análisis Formal de Conceptos. Su objetivo es presentar al lector una visión teórico-práctica de las principales aportaciones en ambos campos, haciendo especial hincapié en aquellas relacionadas con el área de la recuperación de información desde el punto de vista de los modelos, su visualización y los métodos presentados para evaluarlos.

- *Propuesta.* Se trata de la parte principal del trabajo y en ella exponemos nuestra aportación a la solución del problema planteado. En concreto, esta parte se encuentra dividida del siguiente modo:

  - *Propuesta del modelo.* En este capítulo se presenta el modelo de clustering desarrollado para esta Tesis Doctoral, así como la arquitectura que permitiría integrarlo en un sistema a gran escala, orientado a extraer la información necesaria para que éste pueda producir estructuras lo suficientemente simples y adecuadas a la tarea propuesta.
  - *Propuesta de un marco de evaluación.* En este capítulo se presenta el conjunto de métricas desarrolladas para evaluar los resultados obtenidos por cualquier sistema basado en el modelo propuesto.

- *Evaluación y Experimentos.* En esta última parte del trabajo se presenta el conjunto de prototipos desarrollados sobre , así como una cuidadosa evaluación realizada sobre éstos aplicando las medidas propuestas. Este conjunto de prototipos es plenamente funcional e implementa las

diferentes aproximaciones propuestas para llevar a cabo los diferentes procesos de extracción y selección de información necesarios para llevar a cabo la construcción del clustering.

- *Conclusiones y Trabajo Futuro.* Finalmente realizaremos una recapitulación de todo el trabajo presentado, destacando las aportaciones del mismo, así como el conjunto de trabajos futuros y lineas de investigación abiertas que pueden ser cubiertas en los próximos años.

Parte I

PRELIMINARES

Capítulo 2

Clustering

En este capítulo presentaremos, en líneas generales, los principales aspectos relacionados con las técnicas de clustering, así como los trabajos más relevantes realizados en este área. En concreto, focalizaremos nuestra atención en describir en qué consiste el clustering desde un punto de vista general para, a continuación, justificar su aplicación mediante la *Hipótesis de Clustering* a tareas de recuperación de información, describiendo alguno de los métodos utilizados para construir un clustering jerárquico. En la segunda parte de este capítulo presentaremos alguna de las aproximaciones propuestas para evaluar estructuras de clustering, así como para llevar a cabo su visualización.

2.1. Introducción

El término *clustering* o agrupación hace referencia a la tarea de particionar un espacio de información no etiquetada en un conjunto de grupos, clases o *clusters*. Dicha partición se lleva a cabo a partir de propiedades intrínsecas del espacio de información por agrupar que, habitualmente, son extraídas mediante un proceso estadístico. En su trabajo *Cluster Analysis* [44], Everitt presentó las siguientes definiciones de cluster:

- 'A cluster is a set of entities which are alike, and entities from different clusters are not alike.'
- 'A cluster is an aggregation of points in the test space such that the distance between any two points in the cluster is less than the distance between any point in the cluster and any point not in it.'
- 'Clusters may be described as connected regions of a multi-dimensional space containing a relative high density of points, separated from other such regions by a region containing a relatively low density of points.'

De las tres definiciones presentadas, las dos últimas asumen que los objetos por agrupar pueden estar representados por puntos dentro de un espacio de medida, lo que implica que abordar una definición

operacional del clustering sea algo extremadamente complicado [70]. Las principales razones de esta afirmación se fundamentan en que:

1. Cualquier conjunto de objetos puede ser agrupado en clusters de acuerdo a diferentes criterios de proximidad. Esto hace que sean posibles diferentes configuraciones de clustering que, en última instancia, pueden ser igualmente válidas para resolver la tarea requerida.
2. Una estructura de clustering no tiene porqué permanecer estática, sino que puede adaptarse a la llegada de nuevos datos al espacio de información o la variación de los ya existentes.
3. El número de clusters generados a partir de un conjunto de datos está estrechamente ligado al grado de resolución con el que se desea llevar a cabo la agrupación con el fin de facilitar su posterior acceso y análisis.

Comparadas con los procedimientos de agrupación de información manuales, las técnicas de clustering presentan las siguientes ventajas:

1. Un proceso de clustering se caracteriza por aplicar de forma precisa y objetiva los criterios de proximidad definidos. Por el contrario, un proceso de organización de la información manual siempre depende de los criterios de proximidad que posean los individuos encargados de llevarlo a cabo. Estos criterios introducen una componente de subjetividad en el proceso que suele derivar en la obtención de diferentes conjuntos de clusters para un mismo espacio de información.
2. Los procesos de clustering son mucho más rápidos en comparación con los procedimientos de agrupación manuales. Desde este punto de vista, resulta mucho más útil aplicar el conocimiento de los expertos a la comprobación y validación de los resultados obtenidos con este tipo de técnicas que a la generación completa de las agrupaciones.

Actualmente, los procesos de clustering se aplican en multitud de áreas, siendo las más importantes las de recuperación de información, clasificación de documentos, minería de reglas de asociación, reconocimiento de patrones, segmentación de imágenes, análisis de transacciones, etc.

2.2. Clustering de documentos

En este marco podemos situar un tipo de clustering específico denominado *clustering de documentos*, cuyo principal objetivo es el de obtener una clasificación o agrupación de documentos a partir de un conjunto de documentos inicial con el fin de mejorar la precisión de los sistemas de recuperación de información. La obtención de esta clasificación se lleva a cabo particionando el espacio de información en un número finito de grupos o clusters en base a las características intrínsecas del mismo, que habitualmente están representadas por vectores de pesos asociados a cada documento.

Con respecto a la estructura de los clusters obtenidos, es posible clasificar los algoritmos de clustering en dos grandes grupos:

1. *Métodos de Partición.* Aplican el proceso de clustering un única vez sobre el conjunto de documentos objetivo obteniendo, como resultado, un conjunto de clusters (habitualmente disjuntos), que realizan una clasificación de la información sobre un único nivel. Este tipo de clustering está recomendado cuando se desea realizar una única partición del espacio de información en clases y no se requiere dotar al sistema de la posibilidad de realizar refinamientos (subcategorización) del conjunto de clusters obtenido.

2. *Métodos Jerárquicos.* En contraste con el clustering de partición, el clustering jerárquico se caracteriza por aplicar el proceso de clustering iterativamente de modo que los clusters generados en cada paso son sucesivamente refinados generando, como consecuencia, una jerarquía de generalización-especialización.

   Cualquier aproximación de clustering jerárquico puede dirigir el proceso de clustering desde dos perspectivas distintas:

   - *Aproximación divisiva.* Considera inicialmente que todos los documentos del corpus forman parte de un único cluster realizando, en un proceso iterativo, su división en clusters mucho más pequeños y específicos.
   - *Aproximación aglomerativa.* Considera inicialmente que cada documento del corpus conforma un cluster individual, de modo que el proceso de clustering se orienta a agruparlos en clusters de mayor tamaño. En cada paso, el par de clusters más próximos serán combinados en un único cluster. Como consecuencia, la primera iteración en este tipo de aproximaciones supone la comparación entre pares de documentos con el fin de obtener un conjunto de clusters inicial, siendo necesario realizar, en etapas posteriores del proceso, la comparación entre clusters con más de un documento. Esto supone disponer de criterios adecuados que permitan decidir acerca de la similitud o proximidad entre clusters independientemente del número de documentos contenidos en ellos, siendo esta la característica principal que diferencia las distintas aproximaciones aglomerativas [135].

   En ambas aproximaciones se suelen aplicar criterios basados en valores umbral para decidir acerca de la combinación (aglomerativa) o la separación (divisiva) de los clusters, disponiendo de criterios de parada orientados a detener el proceso de clustering (por ejemplo, el número de clusters obtenidos en una iteración concreta). Debemos remarcar que los algoritmos de clustering jerárquico son muy sensibles a este tipo de parámetros, siendo crítico realizar su ajuste experimental sobre cada corpus con el fin de obtener estructuras adecuadas [142].

Cualquier proceso de clustering necesita de un método que proporcione la similitud entre cualquier par de objetos, entendiendo por objetos documentos o los propios clusters. Por lo general, determinar la similitud entre un par de objetos requiere realizar tres pasos principales:

- Seleccionar aquellas variables que servirán para caracterizar los objetos. Existen dos aproximaciones bien conocidas:

  - *Clustering basado en citas.* Implican caracterizar los documentos en base al conjunto de citas o referencias que éstos reciben. De este modo, el grado de similitud entre dos documentos puede obtenerse de manera sencilla a partir del número de referencias compartidas [112]. La validez de esta aproximación está restringida a los casos en los que existe un mecanismo de citas sistemático, como en la literatura científica o el hipertexto.
  - *Clustering basado en términos.* Implican caracterizar los documentos en base a un conjunto de términos obtenidos manual o automáticamente a partir de términos de indexación, palabras clave o tesauros que describen el contenido de éstos.

- Seleccionar un esquema de pesado aplicable al conjunto de variables seleccionadas.
- Seleccionar un coeficiente o medida de similitud que determine la proximidad o parecido entre los vectores de atributos de los documentos que se están comparando. Existen distintos tipos de medidas para determinar este grado de proximidad entre cualquier par de objetos. Sneath y Sokal describen cuatro clases principales: coeficientes basados en distancias (*distance coefficients*), coeficientes basados en asociación (*association coefficients*), coeficientes probabilísticos (*probabilistic coefficients*) y coeficientes de correlación (*correlation coefficients*) [114]. Salton y McGill proponen otros tipos de coeficientes [109]. La comparación experimental de los diferentes coeficientes de similitud [134, 1] sugiere que la elección de éstos está directamente relacionado con los resultados de clustering obtenidos, no existiendo un consenso extendido acerca de cual de los coeficientes resulta ser más adecuado y generalizable.

2.3. Métodos de construcción de un clustering jerárquico

Podemos distinguir entre tres tipos principales de estrategias orientadas a la construcción de estructuras de clustering jerárquico:

- *Estrategia completa o estática.* Su principal característica es la necesidad de disponer a priori del conjunto de documentos completo sobre el cual se va a realizar la agrupación, asumiendo que éste no va a variar durante el proceso de clustering. Suelen estar basados en una medida de similitud interdocumento que es aplicada, en un proceso previo de inicialización, sobre la totalidad de los documentos que se desean agrupar (obteniendo una matriz de similitudes interdocumento). Dado que el proceso considera el conjunto completo de documentos para la realización de todo el proceso de clustering, el coste computacional y de almacenamiento de este tipo de aproximaciones suele ser muy elevado.

  Entre sus ventajas respecto a otras estrategias de clustering debemos destacar la posibilidad de acceder directamente al conjunto completo de similitudes interdocumento y que se acomoda a los tres criterios deseables para un clustering propuestos por van Rijsbergen [119, 108]:

  1. El clustering no se ve drásticamente afectado por la adición de nuevos documentos al corpus.

2. Pequeños errores en la descripción de los documentos conducen a pequeños errores sobre el clustering obtenido.
3. El método es independiente del orden inicial en el que son procesados los documentos de referencia.

Esencialmente estos métodos tratan de descubrir la estructura subyacente en la información en lugar de imponer una estructura adecuada sobre éstos [119].

- *Estrategia heurística*. También denominada *ad hoc* [119], se caracteriza por hacer uso de heurísticas para mejorar la eficiencia del proceso de clustering. En este sentido, este tipo de estrategias o bien no generan o bien no acceden al conjunto completo de similitudes interdocumento del corpus completo sobre el cual se lleva a cabo el proceso de clustering. El hecho de aplicar esta simplificación en el proceso supone incumplir los criterios propuestos por van Rijsbergen [119], en particular el relacionado con la independencia del orden en el que son considerados los documentos a agrupar con respecto a los resultados obtenidos. En este sentido, las estrategias heurísticas suelen ser no deterministas, generando distintas agrupaciones en función del orden en el que son considerados los documentos para la realización del proceso. En contraste, las ventajas obtenidas al sacrificar estos criterios redundan en una significativa disminución del coste computacional asociado al proceso que, en ocasiones, puede llegar a considerarse lineal.

- *Estrategia incremental.* Su principal característica consiste en que este tipo de estrategias asumen que los documentos del corpus sobre los cuales se realiza el proceso de clustering se obtienen a lo largo del propio proceso, lo que supone no disponer a priori de la colección completa sobre la cual se va a realizar el clustering [142, 108]. En contraste con las estrategias completas, estas aproximaciones son mucho más flexibles y adecuadas para la realización de un clustering incremental sobre un corpus dinámico que soporte de manera frecuente la adición y eliminación de documentos. Este tipo de estrategias consideran cada nuevo documento como un elemento a añadir sobre el clustering generado hasta el momento, existiendo la posibilidad de añadirlo a alguno de los clusters ya generados o, por el contrario, utilizarlo como semilla para la generación de un nuevo cluster. Tal y como expondremos en la sección 2.3.3, el proceso puede implicar la reestructuración completa del clustering o, por el contrario, la adición de dicho elemento sin modificar la estructura previa obtenida hasta el momento.

2.3.1. Métodos estáticos o completos

Los principales métodos estáticos o completos son los siguientes:

- *Método del enlace único (single-link).* También denominado método del vecino más cercano (*nearest neighbour*). De todos los métodos completos de clustering jerárquico aglomerativo es el más conocido. Se caracteriza porque los clusters se obtienen a partir de la similitud entre los dos documentos más próximos pertenecientes a clusters distintos. Los clusters obtenidos de esta manera tienen la propiedad de que cualquier miembro del cluster se encuentra mucho más

próximo a, por lo menos, uno de los miembros del mismo cluster que a cualquier miembro de cualquier otro cluster. Una característica de este método es la formación de clusters con un bajo grado de cohesión y cuyas fronteras no se encuentran delimitadas claramente. Este fenómeno recibe el nombre de encadenamiento (*chaining*)

Una formulación alternativa del método del enlace único asume la disponibilidad de una matriz de similitudes interdocumento. De este modo, diremos que dados dos documentos y un umbral de similitud, ambos se encuentran en un mismo cluster si y sólo si existe una cadena de documentos intermedios tales que sus correspondientes similitudes se encuentren por encima del umbral de similitud determinado. De este modo, el número de documentos contenidos en un cluster aumentará conforme disminuya el umbral de similitud prefijado. Existen diferentes implementaciones del método del enlace único, de las que podemos destacar el algoritmo SLINK [110] como uno de los que obtiene mejores resultados de eficiencia con $\Theta(N^2)$ para el coste temporal y $\Theta(N)$ para el coste de almacenamiento.

- *Método del enlace completo (complete-link).* Es el caso opuesto al método del enlace único, debido a que la medida de similitud intercluster se basa en el par de documentos menos similares o próximos. De esta manera, la similitud de cada miembro de un cluster con respecto al documento más próximo de dicho cluster será mayor que su similitud con respecto al documento menos próximo de cualquier otro cluster. Esta definición de pertenencia a un cluster resulta ser mucho más estricta que la presentada en el caso del método del enlace único y permite obtener clusters mucho más cohesivos y definidos. De este modo, y aunque este método tiene un coste computacional mucho más elevado, las ventajas obtenidas al aplicarlo derivan en la generación de clusters tales que todo par de documentos pertenecientes a un mismo cluster se encuentran por encima de un valor umbral de similitud, por lo que cualquier documento cuya similitud con cualquier otro documento sea menor que la indicada en este valor umbral, no pertenecerá a ningún cluster. En contraste, un clustering basado en la aproximación del enlace único permite aplicar un valor umbral de similitud únicamente entre aquellos documentos pertenecientes a cada uno de los enlaces obtenidos en alguna de las etapas del proceso, lo que no garantiza que cualquier par de documentos contenidos en un mismo cluster, pero que no pertenezcan a la cadena de enlaces, estén por encima de este valor límite de similitud. Como consecuencia, esta situación permite la existencia de pares de documentos con similitudes muy bajas dentro de en un mismo cluster, lo que hace que esta aproximación no sea especialmente adecuada en aplicaciones de recuperación de información [135, 108]

- *Método del grupo promedio (group average).* Se considera un método híbrido que combina las aproximaciones enlace único y enlace completo que acabamos de presentar. Se caracteriza porque cada miembro de un cluster tiene una similitud promedio, con respecto al resto de los componentes del cluster, mayor que la que tiene con respecto a todos los miembros de cualquier otro cluster [135].

- *Método Ward.* Se caracteriza por fusionar aquellos pares de clusters de cuya unión resulte el menor incremento en la suma de las distancias Euclídeas de cada documento, dentro del

cluster fusionado, con respecto a su centroide.

2.3.2. Métodos heurísticos

De todos los métodos heurísticos podemos destacar el *algoritmo de Rocchio*[105], que fue desarrollado en el contexto del proyecto SMART. Este algoritmo se caracteriza por realizar los siguientes procesos para obtener el clustering sobre un corpus de documentos:

1. En una etapa inicial se aplica un test de densidad sobre cada uno de los documentos que todavía no han sido agrupados con el fin de identificar semillas de clusters. Estas semillas se caracterizan por ser documentos situados en regiones densas del espacio de documentos [108]. Todos aquellos documentos suficientemente similares a una semilla (cuyo valor de similitud supere un cierto valor límite) formarán un cluster. En esta aproximación los clusters pueden superponerse, lo que significa que un documento podría pertenecer a más de un cluster.
2. En una segunda etapa los clusters obtenidos son ajustados de acuerdo a ciertos criterios tales como el máximo y mínimo número de documentos por cluster, el número de clusters obtenido, el grado de superposición entre clusters, etc.
3. En una última etapa, aquellos documentos muy distantes de las semillas de los clusters o que ocupen regiones muy poco densas, continuarán no agrupados o serán procesados en una etapa de clustering posterior.

De igual modo, los métodos *Buckshot* y *Fractionation*, propuestos en [36], pueden ser considerados heurísticos y lineales, siendo utilizados como base para la navegación interactiva Scatter/Gather [36, 66, 67] sobre un clustering de documentos. Su característica principal es la de aplicar un método completo de clustering a una pequeña parte del corpus que se desea agrupar. De este modo, y aunque el coste de una aproximación completa resulta ser elevado, el hecho de reducir el tamaño de la muestra sobre la que se aplica permite reducir el coste computacional del proceso. El resultado es la rápida generación de un conjunto de clusters de referencia que hacen las veces de semillas para realizar el proceso de clustering (basado en una similitud documento-centroide) sobre la colección completa.

En concreto, el método Buckshot aplica un proceso de clustering a un conjunto de muestras aleatorias del corpus completo de tamaño $\sqrt{kN}$, siendo el coste computacional del proceso $\Theta(kN)$. Los centroides del conjunto de clusters obtenido son utilizados como semillas para realizar el clustering sobre el corpus completo.

En contraste, el método Fractionation divide el corpus original en un conjunto de $\frac{N}{m}$ *buckets*, de un tamaño fijo m (con $m > k$). El objetivo de este método es el de reducir cada uno de los buckets originales a un conjunto de mr clusters, donde r es un factor de reducción predeterminado y menor que uno. De este modo, después de una primera fase de clustering, se dispondrá de mr clusters por cada uno de los buckets generados. El método trata cada uno de los clusters como un documento individual (denominado *documento virtual*) de tamaño $\frac{1}{r}$ con respecto a los documentos originales.

Dado que existen $\frac{N}{m}$ buckets, el número de documentos a procesar será de $mr\frac{N}{m} = Nr$ documentos virtuales.

Para procesar el conjunto de documentos virtuales se opera de igual modo que en el caso de los documentos originales, obteniendo como resultado de esta segunda etapa $\frac{Nr}{m}mr = Nr^2$ documentos virtuales. El proceso se repite hasta que, después de j etapas, existan $Nr^j < k$ documentos virtuales o clusters. Una última etapa aglomerativa producirá k clusters, cuyos centroides pasarán a ser las semillas para realizar el clustering sobre el corpus completo.

La agrupación de los N documentos a partir de las k semillas obtenidas se realiza asignando a cada documento aquel centroide más próximo, pudiendo refinarse el proceso en sucesivas iteraciones (recalculando el centroide cada vez que se añada un nuevo documento). En [36] se propuso otro tipo de refinamientos como, por ejemplo, la definición de un *perfil* de cluster como un vector de términos igual a la suma de los vectores de términos que representan los documentos del cluster. De este modo, la asignación de documentos a los clusters supondría su comparación con sus correspondientes perfiles en lugar de su comparación con los centroides. Así mismo, definieron algoritmos de separación y unión de clusters (*split and join algorithms*) con el fin de mejorar la agrupación finalmente obtenida. Por lo general, la elección del tipo de refinamiento a aplicar, así como del número de iteraciones asociadas al proceso implican un compromiso entre la velocidad y la precisión en su ejecución.

Tanto Fractionation como Buckshots particionan el corpus, lo que significa que no permiten solapamiento entre los clusters generados [142]. Además, debido a que Buckshot obtiene los clusters iniciales a partir de un subconjunto de documentos aleatorio obtenido del corpus inicial, puede ser considerado un método no determinista. Finalmente, la aplicación de estos métodos puede considerarse no adecuada cuando se pretende acceder a clusters muy específicos de poco tamaño, debido principalmente a que el método puede no representarlos correctamente en el subconjunto aleatorio inicial [142].

La motivación principal de los métodos presentados es la de facilitar al usuario la navegación a través de una gran colección de documentos de manera interactiva, lo que implica la necesidad de un proceso de clustering rápido. En este sentido, disponer de la colección completa a priori permitiría aplicar una estrategia híbrida orientada a obtener una jerarquía de clusters off-line, de modo que los nodos de la jerarquía pudieran considerarse documentos virtuales cuya agrupación se llevará a cabo on-line. Obviamente, es posible aplicar cualquier método completo para obtener esta jerarquía de clusters off-line, aunque debemos remarcar que sobre colecciones muy extensas el coste computacional sería muy elevado.

Es por esta razón por la que en [35] se propone utilizar los métodos $\Theta(kN)$ incluso para la realización de la etapa off-line. Aunque estos métodos producen un único nivel de clusters (clustering de partición), es posible aplicarlos iterativamente sobre cada uno de los conjuntos de k clusters obtenidos con el fin de generar una estructura jerárquica. En este sentido, la aplicación de los métodos de clustering propuestos produciría un clustering jerárquico divisivo. Dado que cada iteración tendría un coste $\Theta(kN)$, la jerarquía completa podría obtenerse en un tiempo $\Theta(kNlogN)$.

Una vez obtenida la jerarquía de clusters precalculada, Cutting propuso la realización de un clustering en tiempo constante de forma interactiva basándose en la idea de agrupar un número fijo

de documentos virtuales $M >> k$. El proceso de obtención del conjunto de documentos virtuales (denominados *meta-documentos* en esta aproximación) se lleva a cabo a partir del cluster raíz o de un cluster seleccionado inicialmente por el usuario de manera interactiva en la estructura precalculada. Este cluster inicial es reemplazado por sus respectivos clusters hijos, que serán considerados documentos virtuales. El proceso se repite iterativamente, sustituyendo aquellos clusters con mayor número de hijos por éstos, hasta conseguir un número M de clusters. Una vez hecho esto, se realiza la agrupación de los M documentos virtuales en k clusters aplicando cualquiera de los métodos descritos.

No obstante, y como principal problema asociado a este método de clustering interactivo, debemos remarcar el hecho de que el proceso de clustering está íntimamente ligado a la estructura inicialmente generada. En este sentido, en [111] Silverstein pretendió solventar este problema presentando un método de clustering alternativo denominado *cuasiconstante*.

En este método también se asume un clustering jerárquico precalculado que cubre un corpus de documentos completo de tamaño N. Su principal diferencia con respecto a la aproximación propuesta por Cutting radica en que, mientras que el primero tomaba como punto de partida un cluster sobre esta jerarquía, Silverstein propone el uso de un subconjunto de documentos obtenidos como parte de un proceso inicial de recuperación de información para seleccionar el conjunto de clusters de la jerarquía que los contienen. En este sentido, esta propuesta pretende mapear el subconjunto de documentos sobre la jerarquía precalculada con el fin de mostrar al usuario los resultados de búsqueda adaptados a la estructura subyacente del corpus de documentos y, además, acelerar el proceso de clustering, que se realizará sobre M documentos virtuales.

La expansión del conjunto de documentos virtuales para obtener el valor M objetivo utiliza un test para comprobar el número de documentos recuperados contenidos en cada cluster seleccionado. Cuando un nodo es reemplazado por sus hijos se aplica este test para identificar al peor de ellos (que será aquel que contenga la peor proporción de documentos recuperados), que será reemplazado por alguno de sus hijos. El procedimiento para llevar a cabo el reemplazo se realiza del siguiente modo. Aquellos nodos hijo que no contengan documento alguno recuperado serán descartados directamente, mientras que en aquellos nodos que contengan uno o un número muy pequeño de documentos recuperados se procede a generar un nuevo nodo independiente únicamente con estos documentos. Una vez obtenidos los M documentos virtuales, se procede a realizar su agrupación en k clusters, al igual que propone Cutting. Finalmente, se procede a revisar los clusters generados, eliminando de ellos aquellos documentos que no pertenezcan al subconjunto de documentos inicialmente recuperado.

Mientras que el método de Cutting es constante en el tiempo (en la etapa de interacción con el usuario), el método propuesto por Silverstein implica la utilización de una función que permita identificar los documentos recuperados sobre la jerarquía precalculada. En este sentido, y aunque el coste de esta operación es de $\Theta(|S|logN)$ (donde S es el conjunto de documentos recuperados obtenido interactivamente a partir de la consulta del usuario), podemos considerar el coste del proceso completo como *cuasiconstante* ya que el tiempo invertido en el cálculo de la tabla de identificación de los documentos es insignificante con respecto al proceso de cálculo del clustering.

2.3.3. Métodos incrementales

La característica principal de los métodos incrementales de clustering es que no necesitan disponer, a priori, de la matriz de similitudes interdocumento para la realización del proceso, pudiendo realizar el proceso de clustering sobre un flujo de documentos del que se irán obteniendo, de manera incremental, los pares de similitudes interdocumento. Aún en el caso de disponer de la colección completa a priori, estos métodos procesan el documento i como si no conocieran de la existencia de los $N - i$ documentos restantes.

Desde el punto de vista del proceso, los métodos incrementales se pueden dividir en:

- *Métodos de una única pasada puros.* Se caracterizan por producir, para el elemento i procesado, una estructura de clustering adecuada y adaptada al nuevo documento sin necesidad de rehacer el clustering construido hasta el momento. Por lo general, estos métodos son dependientes del orden en el que son procesados los documentos y, por lo tanto, violan uno de los criterios propuestos por van Rijsbergen [119].

  Una aproximación de clustering incremental en una única pasada podría plantearse del siguiente manera:

  1. Dado un nuevo documento, si este está suficientemente próximo (dados un cierto valor umbral y una medida de similitud) a alguno de los documentos anteriores, estos serán combinados en un mismo cluster. De igual modo, si un documento es suficientemente similar a un cluster formado por dos o más documentos ya procesados, éste será añadido al cluster.
  2. Un documento será añadido a todos aquellos clusters suficientemente similares.
  3. Con el objeto de simplificar el proceso de cálculo de la similitud entre un documento y un cluster, se aplica una medida de similitud documento-centroide.
  4. El hecho de añadir un nuevo documento a un cluster ya existente supone recalcular el centroide con el objeto de reflejar los cambios aportados al cluster por el nuevo documento.
  5. Una vez completado el proceso para los N documentos se dispone de un conjunto de clusters cuyas características dependen directamente del orden en el que éstos han sido procesados. Como consecuencia, es posible obtener un gran número de clusters, clusters de gran tamaño o con un amplio grado de solapamiento, lo que implica fijar a priori parámetros tales como el tamaño máximo de los clusters, máximo grado de solapamiento, etc.
  6. Finalmente, es posible ajustar los resultados finales aplicando técnicas de separación o de unión sobre el conjunto de clusters generados.

- *Métodos de una única pasada no puros.* Se caracterizan por producir, para el elemento i procesado, el mejor conjunto de clusters, reestructurando el clustering previamente obtenido, lo que implica volver a procesar, para cada nuevo documento, los j documentos anteriores.

Así mismo, también es posible dividir los métodos incrementales en:

- *Métodos incrementales puros.* Permiten únicamente aplicar una estrategia incremental, no siendo posible combinar esta con una aproximación completa o estática.
- *Métodos incrementales no puros.* Pueden llevarse a cabo en modo incremental y no incremental. En este sentido, por ejemplo, el método del enlace único (presentado en la sección 2.3.1) suele aplicarse de manera completa sobre una colección de N documentos. Sin embargo, existen algoritmos que permiten ampliar el clustering obtenido de manera incremental con el fin de incluir nuevos documentos sin necesidad de recalcular la estructura de clustering completa. La única condición es que esta actualización respete la regla del enlace único, enlazando el nuevo documento con el documento D_i (con $i < N$) con respecto al cual éste sea más similar e incluyéndolo en todos los clusters que contengan a D_i. En el caso en que el documento presente la misma similitud con respecto a varios documentos distintos (lo que supone un valor de similitud bajo entre ellos y, como consecuencia, su pertenencia a clusters distintos en los niveles más altos de la jerarquía) implica enlazar los clusters a los cuales pertenecen en un único cluster. De este modo, el método del enlace único podría plantearse como un algoritmo incremental aplicable a partir del segundo documento del corpus.

  Las razones para decidir en este tipo de casos por un algoritmo incremental o un algoritmo completo se basan en la eficiencia requerida por ambas implementaciones, por la calidad de las estructuras de clustering generadas o por el modo en el que la información se encuentra inicialmente disponible para su procesamiento.

Zamir et al. [142] desarrollaron un método de clustering incremental novedoso denominado *Suffix Tree Clustering* (STC) motivado, principalmente, por el problema que surge al realizar consultas web mediante un motor de búsqueda. Como resultado de este proceso se obtiene una lista de documentos, habitualmente muy extensa, de los cuales sólo unos pocos resultan ser relevantes a las necesidades del usuario que, además, pueden aparecer en las últimas posiciones de la lista. El trabajo de Zamir et al. se orientó a mejorar la tarea de recuperación de información mediante la realización de un clustering de documentos aplicado sobre el subconjunto de documentos recuperados, con la particularidad de que cada cluster estuviera etiquetado con aquellas frases que mejor describieran el conjunto de documentos agrupados.

Aunque STC está motivado por las dificultades asociadas a un escenario de recuperación de información sobre Internet, su aplicación es adecuada sobre cualquier otro tipo de escenario caracterizado por disponer de un gran número de documentos, una precisión asociada al proceso de recuperación baja y la necesidad de realizar este proceso de manera interactiva.

Las características principales del método STC son las siguientes:

- STC es un método lineal. Lo que supone un coste computacional relativamente pequeño al aplicarlo sobre conjuntos de documentos muy grandes. En comparación con los métodos de coste constante, debemos destacar que estos últimos requieren del acceso a un clustering precalculado off-line sobre la colección completa, lo cual no siempre es posible.

- STC es un método incremental. Lo que supone la obtención de clusters completamente definidos sobre el subconjunto de documentos procesado en cada momento, no siendo necesario disponer del conjunto completo para comenzar el proceso de clustering.
- STC no es heurístico. Lo que supone que la estructura de clustering producida es independiente del orden en el que son agrupados los documentos.
- STC no requiere parámetros iniciales tales como tales como el tamaño máximo de los clusters o el número de clusters generados. Así mismo, no es necesario especificar criterios de parada o de limpieza de los clusters. Únicamente es necesario indicar el grado de solapamiento permitido para los clusters y el número de clusters a reconsiderar en un proceso de clustering posterior.
- STC permite solapamiento entre clusters. En este sentido, su objetivo es el de agrupar documentos de acuerdo a temas, lo que implica que un documento pueda tratar diferentes temas.
- STC utiliza cadenas de palabras como descriptores de los clusters. Estas cadenas son denominadas *phrases*, y su frecuencia de documento es utilizada como medida de similitud entre documentos en el proceso inicial de clustering. De este modo, si dos documentos comparten como poco una frase, serán combinados en un mismo cluster base. Esta aproximación al proceso de clustering resulta muy adecuada para aplicar árboles de sufijos con el objeto de indexar la colección de documentos en función de las frases que forman parte de éstos, en lugar de una matriz de similitudes N^2. Este índice crece linealmente con el tamaño del texto y puede ser actualizado y accedido en tiempo lineal.
- STC realiza una segunda etapa de clustering utilizando una regla de clustering distinta. Esta segunda regla combina los clusters obtenidos en la primera etapa de acuerdo a la proporción de documentos que éstos comparten. Dado que STC permite solapamiento entre clusters, la definición de un criterio de solapamiento permite fusionar aquellos clusters que compartan un excesivo número de documentos.
- Aunque STC no es un método de clustering jerárquico, éste puede ejecutarse iterativamente sobre los conjuntos de clusters obtenidos con el fin de generar una estructura de este tipo.

2.4. Clustering en RI

La aplicación del clustering de documentos en tareas de recuperación de información no es algo nuevo [119, 74, 13, 135, 83, 99, 11, 73, 54], siendo la *Hipótesis de Clustering* (*Cluster Hypothesis*) [119] la base sobre la cual se han apoyado todas las aproximaciones propuestas. Esta hipótesis fue formulada por van Rijsbergen [119], y afirma:

> 'Closely associated documents tend to be relevant to the same requests'

De este modo, un sistema capaz de organizar la información agrupando documentos similares debería mejorar el proceso de acceso a la información, permitiendo al usuario recuperar el conjunto de documentos relevantes a una consulta concreta mediante la selección de aquel o aquellos clusters (grupos de documentos) que contengan algún documento relacionado con su necesidad de información. Tal y como fue expuesta esta hipótesis, el proceso de clustering se plantea como una tarea que debe ser realizada con anterioridad a cualquier proceso de acceso a la información, siendo la estructura de clustering generada la base para dirigir dichos procesos.

No existe una demostración genérica que permita afirmar que esta hipótesis se satisfaga en cualquier caso, siendo posible demostrar su validez de manera empírica únicamente sobre colecciones concretas. Además, la aplicación de un algoritmo de clustering concreto puede no generar clusters suficientes, un conjunto de clusters excesivamente solapado o un conjunto de clusters que no se corresponda con los temas relevantes que el usuario está buscando. Aún así, siempre es posible asumir la partición de un conjunto de documentos en dos clases principales disjuntas (documentos relevantes y documentos no relevantes).

La *Hipótesis de Clustering* resulta especialmente efectiva cuando se aplica sobre estructuras de clustering jerárquico. En esta situación, los clusters situados en las posiciones inferiores de la jerarquía agruparán documentos muy similares y, como consecuencia, documentos con alta probabilidad de ser relevantes a un mismo tipo de consulta. Así mismo, debemos resaltar que la combinación de clustering con técnicas de visualización e interacción adecuadas hace de éste una herramienta muy efectiva para navegar sobre grandes corpus de documentos, permitiendo la rápida localización de temas concretos y documentos relacionados con estos [135].

2.4.1. Clustering a priori versus clustering a posteriori

Las aproximaciones más destacables para la aplicación de clustering sobre páginas web pueden enfocarse de dos maneras principales:

- *Clustering a priori.* Consistente en realizar el proceso de clustering sobre el corpus completo de páginas web indexadas. Los principales inconvenientes de este tipo de aproximación son:
  - La disponibilidad por adelantado del corpus completo, lo que en la mayor parte de las ocasiones no es posible.
  - La realización de un proceso de clustering que implica un tiempo de proceso muy elevado.
  - La obtención de una estructura de clustering estática sobre la cual va a navegar el usuario, lo que supone la adaptación del usuario a una estructura rígida que le obliga a dirigir su proceso de acceso a la información a través de caminos que, por ser muy generales, no se adapten a sus criterios de búsqueda [107, 62].
  - La dificultad para realizar la actualización del clustering generado.
- *Clustering a posteriori.* Consistente en realizar el proceso de clustering únicamente sobre un subconjunto de documentos recuperados previamente por un motor de búsqueda, facilitando

el acceso al usuario a la información relevante recuperada. Una aproximación de este tipo resulta ser mucho más flexible y adecuada en tareas de recuperación de información debido a que, en última instancia, es la estructura de clustering la que se adapta a las necesidades del usuario, proporcionándole clusters descritos con términos relacionados con su búsqueda. En contraste con la aproximación anteriormente descrita, en este caso el proceso de clustering se lleva a cabo considerando únicamente una pequeña parte de los documentos recuperados (snippets), que son extraídos automáticamente por el motor de búsqueda de acuerdo a los términos utilizados por el usuario para realizar su consulta.

2.4.2. Clustering de resultados de búsqueda web

Las aproximaciones a la aplicación del clustering de documentos en este contexto son muy variadas y debemos destacar su aumento en los últimos años. El hecho de disponer de las herramientas de programación adecuadas (proporcionadas por la gran mayoría de los buscadores comerciales) ha permitido la realización de prototipos que, en lugar de interactuar sobre colecciones de dominio específico, permiten aplicar las técnicas de clustering sobre un escenario tan heterogéneo como la Web. En este sentido [47] clasifica las distintas propuestas de clustering sobre páginas web en las siguientes categorías:

- *Basadas en términos y clustering de partición*. La propuesta Scatter/Gather [66, 67] fue una de las primeras propuestas de clustering sobre los resultados proporcionados por un motor de búsqueda y puede considerarse dentro de esta categoría de clustering, aunque no llegó a ser probada sobre un motor de recuperación sobre la web. WebCat [55] utiliza el método de las k-medias para obtener un clustering de partición. Retriever [71] utiliza lógica borrosa para la generación de los clusters. El sistema presentado en [127] expande el conjunto de snippets [1] recuperados con los enlaces entrantes y salientes a cada uno de ellos para mejorar la precisión. No obstante, y debido a que los motores de búsqueda no facilitan esta información directamente, esta aproximación supone disponer de una copia local de esta información con el fin de hacer el proceso de extracción de enlaces eficiente. Los métodos estándar de clustering tales como k-medias o el método del enlace único podrían englobarse también dentro de esta categoría debido a que utilizan los términos de los documentos como representantes para la realización del clustering. Dentro de este grupo, el único sistema actualmente accesible vía web es WebCat [55].

- *Basadas en sintagmas y clustering de partición*. Grouper [143] fue el primer sistema público que abordó el problema del clustering de documentos sobre páginas web (en concreto sobre sus snippets). Se caracterizó por utilizar sintagmas de longitud variable para describir cada uno de los clusters, obtenidos como n-gramas contiguos a partir de los snippets aplicando árboles de sufijos. Lingo [95] utiliza SVD (*Sigular Value Decomposition*) sobre una matriz de términos-documentos para encontrar etiquetas significativas. El problema asociado con esta

[1] fragmentos de texto utilizados para representar los documentos recuperados y que se caracterizan por contener alguno de los términos utilizados en la consulta

aproximación es el coste asociado al cálculo de SVD cuando es aplicado a un gran número de snippets. Recientemente, Microsoft [144] propuso un sistema que extraía sintagmas de longitud variable. Aunque el clustering es de partición, el cálculo de los sintagmas requiere de una fase de entrenamiento que hace compleja su adaptación a la web. De todos los sistemas y aproximaciones presentadas, únicamente se encuentra disponible Carrot2 [129], que es una implementación basada en código abierto del sistema Grouper.

- *Basadas en términos y clustering jerárquico*. FIHC [51] aplica un análisis basado en el Problema de Conjuntos de Objetos Frecuentes (*Frequent Itemset Problem*) orientado a la construcción de una jerarquía de carpetas. Credo [18, 19, 32] utiliza un retículo de conceptos basado en términos y se trata del único sistema de este tipo disponible on-line. Sin embargo, su paradigma de visualización está basado en una jerarquía de carpetas y únicamente considera términos como descriptores de los clusters obtenidos.

- *Basadas en sintagmas y clustering jerárquico*. A este grupo pertenecen todas aquellas aproximaciones y sistemas que simulan el comportamiento del motor de clustering comercial Vivisimo [122], siendo el sistema Lexical Affinities Clustering [88] el primero en proponer esta aproximación. Este sistema mejoraba la precisión sobre la cobertura utilizando una representación basada en snippets obtenida a partir de pares de términos (no necesariamente contiguos) unidos mediante una afinidad léxica. En [143], Etzioni propuso una extensión del sistema Grouper basada en clustering jerárquico a partir del grado de solapamiento de los clusters. SHOC [145] utiliza árboles de sufijos para la extracción de n-gramas contiguos que organiza sobre una estructura jerárquica aplicando SVD. Highlight [136] aplica análisis léxico sobre un marco probabilístico para la construcción de la jerarquía, aunque los autores no proporcionan ninguna evaluación. CIIRarchies [79] extrae sintagmas a partir de los snippets utilizando un modelo de lenguaje precalculado y construye la jerarquía aplicando un algoritmo iterativo. Los autores reconocen que la jerarquía obtenida es, a menudo, poco compacta, muy profunda y contiene términos sin sentido o términos repetidos. Recientemente, IBM [75] ha propuesto un sistema que construye la estructura jerárquica minimizando una función objetivo. En este sentido, SnakeT [46, 47] realiza el proceso de clustering de manera similar.

2.4.3. Estrategias de exploración del clustering

El proceso de búsqueda y acceso a la información sobre un clustering jerárquico a partir de una consulta concreta se puede llevar a cabo mediante un proceso de refinamiento de arriba a abajo (*top-down*) o de abajo a arriba (*bottom-up*).

- *Proceso de búsqueda arriba-abajo.* Se lleva a cabo comparando, mediante la aplicación de una medida concreta de similitud, la representación de la consulta inicial con cada uno de los representantes o centroides de los clusters de más alto nivel. El resultado de este proceso de comparación permite obtener un ranking donde los clusters más relacionados con la consulta aparecerán en las posiciones más altas del mismo. En un proceso iterativo, y hasta llegar al último nivel de la jerarquía, se repite el proceso de comparación únicamente sobre el conjunto

de clusters que especializan los clusters de más alto nivel previamente seleccionados. Al final de este proceso, se extraen los documentos pertenecientes al conjunto de clusters de nivel más específico que han sido seleccionados y se ordenan de acuerdo a su relevancia con la consulta original [34].

Un proceso de búsqueda de este tipo puede no ser capaz de acceder a la totalidad de la información relevante. Esto es debido a que, en un escenario caracterizado por clusters genéricos de gran tamaño y poco acoplados, los centroides obtenidos pueden no representar de manera exacta el contenido de todos los documentos incluidos en el cluster. En este sentido, el proceso de comparación de la consulta con ellos puede derivar el proceso de búsqueda hacia zonas del clustering que no se correspondan exactamente con las necesidades de búsqueda del usuario. Como consecuencia, este tipo de procedimientos únicamente son adecuados cuando el método de clustering pueda asegurar que los clusters más genéricos van a ser lo suficientemente pequeños y cohesivos como para que no se produzca este efecto.

- *Proceso de búsqueda abajo-arriba.* Su funcionamiento es inverso al expuesto para el tipo de búsqueda que acabamos de presentar, es decir, inicia el proceso de búsqueda en la parte inferior de la jerarquía. La principal ventaja con respecto a la aproximación anterior se basa en el acceso inicial a clusters con un menor tamaño y con un mayor grado de cohesión.

  El problema de esta aproximación es cómo seleccionar el cluster de más bajo nivel sobre el cual iniciar la búsqueda. Una posible solución pasaría por realizar una búsqueda convencional, basada en la consulta, que devuelva un documento relevante, de modo que el cluster inicialmente seleccionado sea el que contiene este documento. Una segunda alternativa podría basarse en realizar, igualmente, un proceso de búsqueda convencional pero, en este caso, realizando la comparación no con documentos individuales, sino con el conjunto de centroides representantes del conjunto de clusters de más bajo nivel obtenidos en el proceso de clustering. En este caso, el cluster más próximo a la consulta original será seleccionado como cluster inicial para realizar el proceso de búsqueda. Al igual que en el proceso de búsqueda abajo-arriba, el procedimiento se repetirá iterativamente hasta acceder al cluster que contenga el conjunto de documentos relevantes que el usuario desea recuperar [135].

En el caso de realizar el proceso de búsqueda basado en la comparación con los centroides de los clusters, el clustering jerárquico resulta ser una técnica altamente eficiente, permitiendo realizar las búsquedas rápidamente. Esto es debido a que el proceso de búsqueda implica la realización de comparaciones únicamente con los representantes de los clusters y no con todos los documentos contenidos en cada uno de ellos. No obstante, en el caso de disponer del corpus de documentos totalmente indexado, un proceso de recuperación basado en el modelo del espacio vectorial o en el modelo booleano puede llegar a ser tan eficiente como una búsqueda basada en clusters. Esto es debido a que en este tipo de escenarios únicamente se lleva a cabo el proceso de búsqueda sobre aquellos documentos que contengan algún término de la consulta siendo crucial, por tanto, que ésta se encuentre bien formulada.

2.5. Evaluación

Podemos distinguir diferentes aproximaciones a la evaluación de sistemas de clustering en función de su alcance:

- *Medidas orientadas a una evaluación de propósito general.* Muchas de las medidas definidas en el área del clustering pueden considerarse de propósito general, siendo su objetivo final evaluar la calidad del mismo en función únicamente de su capacidad para agrupar correctamente la información sobre un conjunto de clases previamente definidas. En este grupo podemos incluir medidas clásicas tales como la pureza, la pureza inversa, la medida-F, la cohesión interna, etc [128]. Su característica principal es que no están orientadas a una tarea concreta y, por lo tanto, no resultan adecuadas sobre el escenario planteado en este trabajo.

- *Medidas orientadas a evaluar una tarea de recuperación interactiva.* Conforme aumentó el tamaño de las colecciones y el proceso de recuperación se realizó sobre escenarios distribuidos (como la Web), la aplicación del clustering de documentos a la recuperación interactiva creció en importancia, planteando la necesidad de desarrollar nuevas metodologías capaces de evaluar los resultados de un proceso de recuperación sobre este tipo de escenarios. En este sentido, resulta relevante considerar en el proceso de evaluación no sólo la interacción del usuario con la estructura de clustering, sino también la complejidad de la propia estructura generada.

  Uno de los experimentos más conocidos es el planteado por Zamir et al. [142] para comparar el método STC (consultar sección 2.3.3) con varios métodos de clustering heurísticos y completos. Los resultados de sus experimentos indicaron que un usuario, en el 80 % de las ocasiones, era capaz de seleccionar el cluster que contenía la mayor proporción de documentos relevantes basándose en las etiquetas asociadas a los clusters o en los resúmenes proporcionados para describirlos. Para obtener estos resultados se generaron 10 consultas para las cuales se llevó a cabo un proceso de recuperación sobre Internet. Los documentos obtenidos fueron juzgados manualmente. Sobre los subconjuntos de documentos recuperados se procedió a realizar un clustering utilizando diferentes aproximaciones y ajustando sus parámetros de modo que únicamente fueran generados 10 clusters por cada aproximación. Sobre cada conjunto de clusters generado para cada método se seleccionaron los mejores clusters (aquellos que contenían un mayor número de documentos relevantes) hasta cubrir el 10 % de los documentos recuperados (lo que implicaba la posibilidad de partir el último cluster con el fin de seleccionar únicamente un subconjunto de documentos).

  El subconjunto de documentos extraídos de los clusters más relevantes fue ordenado de acuerdo a un ranking, calculando su precisión promediada sobre las 10 consultas (debido a que STC permite el solapamiento de los clusters, los documentos duplicados fueron descartados). La ordenación del conjunto de documentos se realizó teniendo en cuenta los clusters, es decir, colocando en primer lugar aquellos documentos contenidos en el primero de los clusters seleccionados, sin que Zamir et al. especificaran el modo en el que fueron ordenados los documentos contenidos dentro de un mismo cluster.

Los resultados obtenidos mostraron que STC mejoró notablemente los resultados respecto al resto de los métodos de clustering aplicados, siendo el método completo el que obtuvo los mejores resultados después de STC. Aún así, Zamir expuso estos resultados como preliminares debido principalmente a que habían sido obtenidos a partir de colecciones no estándar y de pequeño tamaño (200 documentos de los cuales había en promedio 40 documentos relevantes), los juicios de relevancia fueron generados por investigadores en lugar de por expertos y, finalmente, a que no se realizó el estudio considerando usuarios reales y consultas interactivas.

Evaluaciones como las de Hearst y Pedersen [66, 67] o Leuski [81, 80], aún estando orientadas a la tarea, no tenían en cuenta la estructura sobre la cual se realizaba el proceso de recuperación de información. En el primer caso, se evaluaba la capacidad del sistema Scatter-Gather para agrupar la información relevante seleccionando únicamente el cluster con más cantidad de documentos relevantes que finalmente era comparado con la lista ordenada original. De igual modo, la segunda de las propuestas cuantificaba el grado de separación entre la información relevante y no relevante sin tener en cuenta las propiedades jerárquicas y de navegación del clustering obtenido. Con este fin, [81, 80] introducía un factor de separación S, sobre el cual se redefinían las medidas clásicas de precisión y cobertura como la proporción de documentos relevantes agrupados y la proporción de documentos relevantes respectivamente.

Finalmente, aproximaciones más recientes como las de Lawrie [78, 79] o Kumammuru [75] consideran la estructura jerárquica del clustering obtenido y tienen como objetivo cuantificar el esfuerzo realizado por el usuario para acceder a la información relevante midiendo el tiempo necesario para realizar dicha tarea. [78, 79] organizó las jerarquías estimando el tiempo necesario para encontrar todos los documentos relevantes calculando el número de nodos que debían atravesarse y el número de nodos que debían ser inspeccionados. En este caso, la medida de evaluación fue utilizada para comparar la estructura de varias jerarquías construidas utilizando diferentes aproximaciones de clustering. El algoritmo de evaluación se basa en la construcción de los caminos óptimos a cada uno de los documentos relevantes, para cada uno de los cuales se lleva a cabo el cálculo del tiempo empleado en recorrerlo. El resultado de esta evaluación es el promedio de los tiempos obtenidos. En este caso, Lawrie no hizo diferencias entre el coste cognitivo de examinar la descripción de un cluster y el coste cognitivo de examinar la descripción de un documento. Por otra parte, Kumammuru [75], presenta una estrategia similar que compara las mejoras de recuperación de información sobre diferentes estrategias de clustering comparadas con una lista ordenada devuelta por un motor de búsqueda tradicional. En este caso, sí se consideran diferencias entre el coste cognitivo para examinar los descriptores de los clusters y el coste cognitivo para examinar la descripción de los documentos contenidos en los clusters pero, de nuevo, el algoritmo de evaluación opera sobre cada uno de los documentos relevantes de forma separada y da como resultado el promedio de los resultados.

En resumen, hasta la fecha no hemos encontrado ninguna medida de evaluación que tenga en cuenta el coste cognitivo asociado a la exploración de un clustering considerándolo como un todo, y por ello en este trabajo hemos decidido elaborar una medida específica (consultar

capítulo 7) que subsane las deficiencias encontradas en las medidas estudiadas.

2.6. Visualización

La mayor parte de los sistemas de recuperación de información devuelven los documentos recuperados en forma de una lista ordenada de documentos, donde cada documento habitualmente es representado por su título y un pequeño resumen (*snippet*) que suele estar relacionado con la consulta realizada por el usuario. Una aproximación de este tipo suele ser adecuada cuando se busca un documento concreto, pero muy limitada en los demás casos.

2.6.1. Sobre resultados de búsqueda

Con el fin de solventar este problema, Veerasamy et al. [120, 121] propusieron mostrar el conjunto de documentos recuperados como una matriz, donde las filas se correspondían con las palabras clave de la consulta, las columnas con los documentos recuperados ordenados de acuerdo al ranking obtenido por el motor de búsqueda y los elementos de la matriz consistían en barras verticales que representaban el peso de cada uno de los términos de la consulta en un documento concreto. De este modo, el usuario podría visualizar rápidamente aquellos términos de la consulta mejor representados en los documentos más relevantes, lo que le permitiría modificar de manera adecuada su consulta.

Con el fin de evaluar la propuesta de visualización, Veerasamy definió varias medidas para controlar la efectividad de la interacción del usuario. En concreto, definió la *precisión interactiva* como la proporción de documentos que son juzgados como relevantes por el usuario y por los expertos encargados de evaluar los resultados, la *cobertura interactiva* como el radio de documentos juzgados como relevantes por el usuario frente a los documentos juzgados como relevantes por los expertos. Finalmente, definió la *exactitud* (*accuracy*) como el número de juicios de relevancia correctos menos el número de juicios de relevancia incorrectos. En este contexto, el término corrección hace referencia a la existencia de consenso con los expertos, tanto sobre los documentos relevantes como sobre los documentos no relevantes. En comparación con las medidas estándar de precisión y cobertura de recuperación de información, las propuestas por Veerasamy tienen en cuenta los juicios de relevancia de los usuarios en lugar de los juicios de relevancia de un sistema de recuperación.

En [121] se describe de manera detallada el experimento realizado para medir la efectividad en la tarea de recuperación utilizando la técnica de visualización expuesta. Se utilizó una porción del corpus TREC sobre diez consultas pertenecientes a la misma competición. Así mismo, se utilizó INQUERY 2.1p3 como motor de búsqueda. Dado que consideraron que la tarea de reconocer documentos relevantes es mucho más compleja que la tarea de reconocer documentos irrelevantes, proporcionaron a los usuarios dos grupos de documentos por cada consulta. El primero de ellos (denominado de alta precisión) contenía los primeros 60 documentos recuperados por el motor, mientras que el segundo (denominado de baja precisión) contenía los documentos recuperados entre las posiciones 90 y 150 del ranking. Así mismo, ambos conjuntos fueron divididos en dos subconjuntos tales que el primero contenía los documentos con ranking par, mientras que el segundo contenía los documentos con ranking impar. Los conjuntos de documentos con ranking par fueron presentados al

usuario mediante la herramienta de visualización, mientras que los conjuntos de documentos con ranking impar fueron presentados al usuario directamente. Dada una consulta concreta, el usuario debía juzgar la relevancia de los documentos contenidos en cada uno de los cuatro conjuntos.

Los resultados obtenidos en el experimento mostraron que los usuarios pueden identificar la relevancia de un documento de forma más precisa con la herramienta de visualización que sin ella. Además, la mejora de precisión obtenida al aplicar la herramienta de visualización fue la misma sobre los conjuntos de documentos de alta y de baja precisión, disminuyendo el tiempo necesario por los usuarios para juzgar la relevancia de los documentos (siendo este efecto mucho más pronunciado en los conjuntos de baja precisión que en los de alta precisión). Finalmente, el experimento mejoró significativamente la cobertura interactiva y mínimamente la precisión interactiva, siendo la mejora en exactitud mucho más elevada en el caso del conjunto de documentos de baja precisión que en el caso de documentos de alta precisión, independientemente de haber utilizado o no la herramienta de visualización. Estos datos demostraron una habilidad mucho mayor de los usuarios para identificar documentos no relevantes que para identificar documentos relevantes, no pudiendo ser mejorado este resultado por la herramienta de visualización.

Debemos destacar la relación de resultados entre las tres medidas propuestas. La mejora en la cobertura interactiva supone que la herramienta de visualización ayuda a los usuarios a reconocer correctamente una gran proporción de los documentos relevantes, mientras que la leve mejora en la precisión interactiva supone que la mejora en el número de documentos relevantes identificados es contrarrestado por un incremento proporcional de documentos identificados erróneamente como relevantes. Por esta razón, una mejora en el parámetro de exactitud significa que la herramienta de visualización ayuda sustancialmente a los usuarios a clasificar correctamente los documentos como no relevantes.

Un paradigma de visualización alternativo al de Veerasamy, denominado *TileBar*, fue propuesto por Hearst [65]. En esta aproximación cada fila se corresponde con un documento recuperado que es representado por una serie de segmentos adyacentes no solapados que reciben el nombre de *tiles*. Un tile puede entenderse como un segmento multipárrafo del documento que trata de algún tema concreto. El orden en el que se muestran los tiles para cada documento es el orden natural dentro del mismo, siendo sus zonas frontera aquellas partes del documento donde se produce un cambio de temática. Cada tile se encuentra sombreado en función de la frecuencia de aparición en el mismo de los términos de la consulta. De este modo, el usuario no sólo conoce la relevancia de un documento concreto con respecto a su consulta sino que, además, puede identificar de manera sencilla las diferentes temáticas tratadas en cada documento y su relación con los términos de la consulta, pudiendo identificar puntos de relevancia dentro de los propios documentos.

Esta aproximación, además, permite visualizar un conjunto de documentos en función de varias consultas al mismo tiempo, lo que supone dividir cada tile en función del número de consultas con el fin de mostrar al usuario la influencia independiente de cada una de ellas.

Tanto Veerasamy como Hearst proporcionaron al usuario aproximaciones adecuadas para la visualización de conjuntos de documentos, permitiéndole estudiar las propiedades individuales de cada uno de ellos o comparar entre sí los documentos del conjunto. No obstante, otra aproximación para facilitar al usuario el acceso a una colección de documentos es el clustering. La aplicación de apro-

ximaciones de este tipo permiten al usuario inspeccionar una estructura mucho sencilla, agrupada en conjuntos de documentos descritos por palabras clave o frases, que una lista de documentos que, en muchas ocasiones, puede resultar muy larga.

2.6.2. Sobre un clustering

La realización de un proceso de clustering sobre un conjunto de documentos recuperados muy extenso supone considerar ciertos requerimientos. En primer lugar no es posible realizar el preprocesamiento de la colección ya que ésta no es accesible o implica un coste computacional muy elevado. En segundo lugar, el proceso de clustering debe realizarse rápidamente, sin incrementar de manera notable el tiempo invertido en llevar a cabo el proceso de recuperación. En tercer lugar, las etiquetas o descriptores asociados a cada cluster deberían facilitar al usuario la tarea de acceder a la información relevante. Finalmente, la selección de los clusters adecuados debería mejorar la precisión de la búsqueda en comparación con la exploración completa de la lista de documentos original.

Con el término navegación hacemos referencia a la idea de una búsqueda donde el usuario explora un espacio de información sin una visión clara y concreta de la totalidad de la información disponible o de su organización. Puede que el usuario no tenga un objetivo concreto y preciso en mente o que, incluso, persiga varios objetivos. En el caso de tener un objetivo poco preciso, es posible que no sepa expresarlo mediante una consulta que contenga los términos adecuados. Este paradigma de acceso a la información implica una reformulación y modificación constante de los objetivos de acuerdo a la información o a las categorías que va descubriendo y que considera de su interés.

Aunque el método de navegación depende, en primer lugar, del tipo de indexación aplicada sobre la colección (manual o automática), en esta sección nos centraremos en aquellas técnicas de navegación aplicadas sobre colecciones indexadas automáticamente.

El paradigma *espacio de navegación* (*browsing space*) permite al usuario visualizar un espacio vectorial y moverse por él libremente (o lo que es lo mismo, manipular el espacio). La dificultad más obvia asociada a este paradigma es que el número de dimensiones en un espacio vectorial típico en recuperación de información suele ser muy elevado. Incluso aplicando técnicas para la reducción de dimensiones sobre espacios vectoriales tales como *Latent Semantic Indexing* [38] no es posible alcanzar un valor razonable como para que éste pueda ser correctamente visualizado e interpretado por un usuario. La solución a este problema de exceso de información pasa por representar únicamente un número de dimensiones clave (relacionadas con términos relevantes de la consulta o de los documentos) que faciliten la navegación al usuario. En el caso de seleccionar más de tres dimensiones, será necesario mapear las dimensiones extra sobre características visuales distintas a las dimensiones espaciales con el fin de que sean fácilmente identificables por el usuario (color, tamaño, textura, grado de opacidad, etc.)

La solución a este problema consiste, por lo tanto, en seleccionar un número de dimensiones clave (términos importantes de la consulta o de los documentos) que faciliten la navegación al usuario. En el caso de seleccionar más de tres dimensiones, será necesario mapear las dimensiones extra como características visuales diferentes a las dimensiones espaciales con el fin de que el usuario pueda identificarlas (color, tamaño, textura, grado de opacidad, etc.) [42]

Una aproximación completamente diferente a la navegación por una colección de documentos es el método Scatter/Gather [36, 66, 67]. Esta aproximación está basada en una metáfora similar a consultar la tabla de contenidos de un libro (para obtener una visión de aquellas informaciones que se encuentran disponibles) y consultar su índice (para acceder a la página o sección relacionada con un tema específico). La *tabla de contenidos* es generada mediante un proceso de clustering realizado sobre el conjunto de documentos, y está descrita por las etiquetas o resúmenes asociados a cada cluster. De este modo, los documentos agrupados conjuntamente tratarán de temas comunes, siendo su etiqueta el elemento que lo identifica. Esta fase recibe el nombre de *scatter* debido a que los documentos, inicialmente contenidos en un corpus concreto, son divididos en múltiples clusters. En un proceso de inspección y selección posterior, el usuario accede al conjunto de clusters más adecuado a sus necesidades, agrupando sus documentos en un subconjunto, en un proceso que recibe el nombre de *gather*. Posteriormente, los documentos seleccionados volverán a ser agrupados, en un nuevo proceso de *scatter*, con el fin de realizar un clustering mucho más fino sobre la información considerada como relevante por el usuario. Esta nueva tabla de contenidos será nuevamente inspeccionada por el usuario, repitiendo el proceso hasta que éste haya dado por satisfechas sus necesidades de búsqueda. En cualquier momento de este proceso, el usuario puede optar por un proceso de búsqueda alternativa (basada en palabras clave o búsqueda booleana) que le permitirá seleccionar documentos concretos sobre un cluster de interés para el usuario, correspondiéndose esta forma de acceder a la información con el proceso de consultar un índice dentro de un libro. En cualquier momento del proceso, el usuario puede volver hacia niveles superiores y seleccionar diferentes temas a explorar, iniciando una nueva secuencia de scatter/gather.

Existen varias aproximaciones para generar descriptores a partir de un cluster. En [36] se utilizó el concepto de *resumen del cluster* (*cluster digest*), definido como los m documentos y los w términos más próximos al centroide del cluster (denominados perfil del cluster). Dado que este método requiere la realización de un clustering on-line de los clusters seleccionados interactivamente, se hace necesario el uso de un método y de un algoritmo de clustering eficiente, siendo Buckshot y Fractionation (seccion 2.3.2) los algoritmos elegidos en esos experimentos.

Capítulo 3

Análisis Formal de Conceptos

En este capítulo pretendemos introducir los fundamentos matemáticos más importantes derivados de la teoría del Análisis Formal de Conceptos y que serán la piedra angular sobre la que desarrollaremos el modelo de clustering propuesto en esta Tesis Doctoral. Introduciremos la noción de contexto formal para, a continuación, definir los conceptos formales. Desde el punto de vista de nuestro trabajo, éstos últimos serán utilizados para representar los clusters de documentos. El uso de AFC supone la aplicación de un conjunto de operadores definidos en la teoría y que resultan fundamentales para obtener el *Teorema Fundamental del Análisis Formal de Conceptos*. Este teorema será la base para generar las estructuras capaces de organizar el conjunto de conceptos formales obtenido. Finalmente, y dado que los aspectos relacionados con la visualización e interacción sobre un retículo no son triviales, en la última parte de este capítulo presentaremos los paradigmas de visualización más comúnmente utilizados para facilitar la interpretación, comprensión e interacción del usuario sobre el espacio de información agrupado.

3.1. Fundamentos Matemáticos

El AFC es una aproximación matemática relativamente reciente para la formalización del conocimiento conceptual [130, 132]. Es una teoría de formación de conceptos derivada de teorías de retículos y conjuntos ordenados que proporciona un modelo matemático para el análisis de jerarquías conceptuales. Desde el punto de vista computacional y dentro de nuestro contexto, el AFC puede entenderse como una teoría que permite la estructuración y clasificación automática de la información orientada a la construcción de sistemas de OI para el análisis y recuperación de información. Entre otras aplicaciones, el AFC puede utilizarse para la detección de patrones, regularidades y excepciones que, haciendo visible y accesible la estructura conceptual de la información, lo convierten en una herramienta muy adecuada para clasificar y recuperar información sobre colecciones de documentos de cualquier tipo.

3.1.1. Conceptos y Contextos Formales

La idea básica en la que se fundamenta el AFC es la noción de *concepto formal*, alrededor del cual es posible estructurar los datos pertenecientes a un dominio concreto. Los conceptos formales son abstracciones del pensamiento humano, obtenidos formalmente, que permiten una interpretación significativa y comprensible de la información. Es necesario resaltar que, aunque los conceptos pueden llegar a ser directamente comprensibles, se trata de entidades obtenidas matemáticamente y no deben confundirse con conceptos mentales [53]. De acuerdo a la teoría del AFC, un concepto viene definido por dos partes bien diferenciadas:

- Su *extensión*, que hace referencia al conjunto de objetos (entidades o instancias) pertenecientes al concepto.

- Su *intensión* o *comprensión*, que hace referencia a todos los atributos (propiedades o características) que comparten todos los objetos considerados.

Si un objeto y un atributo pertenecen a un mismo concepto, entonces se puede afirmar que el objeto 'tiene' ese atributo concreto, es decir, dentro de un concepto la extensión se relaciona con la intensión mediante una relación de incidencia entre los objetos y los atributos, que son recíprocamente dependientes.

Debido a que un concepto puede tener un gran número de instancias y a que estas pueden ser un conjunto prácticamente ilimitado de propiedades o atributos compartidos, habitualmente se trabaja sobre un contexto específico dentro del cual están limitados tanto el conjunto de objetos como el de atributos. El modelo matemático que representa la relación entre los objetos y los atributos se denomina *contexto formal* y se define como una terna $K := (G, M, I)$ formada por dos conjuntos G y M y una relación binaria de incidencia $I \subseteq G \times M$ entre G y M. Los elementos g de G representan los objetos o entidades del contexto, mientras que los elementos m de M representan los atributos o características que los objetos pueden tener asociados. La relación gIm afirma que 'el objeto g tiene el atributo m' o, de forma equivalente, que 'el atributo m se aplica sobre el objeto g'.

El cuadro 3.1 muestra el contexto formal correspondiente al conjunto de planetas del Sistema Solar descritos por un conjunto de atributos[1]. El conjunto de atributos seleccionado para describir los planetas (objetos del contexto formal) son los siguientes: su tamaño (que podrá ser pequeño, mediano o grande), su distancia al Sol (que podrá ser cercana o lejana), y, finalmente, si posee luna propia o no. De este modo, el contexto formal $K := (G, M, I)$ vendría descrito por el conjunto de planetas, el conjunto de características utilizadas para realizar la descripción del dominio que acabamos de describir y el conjunto de relaciones entre objetos y atributos que estarían representadas en el cuadro mediante un aspa.

[1] El ejemplo ha sido tomado del texto clásico *Introduction to Lattices and Order* [37]

	TP	TM	TG	DC	DL	LS	LN
Mercurio	×			×			×
Venus	×			×			×
Tierra	×			×		×	
Marte	×			×		×	
Júpiter			×		×	×	
Saturno			×		×	×	
Urano		×			×	×	
Neptuno		×			×	×	
Plutón	×				×	×	

Cuadro 3.1: Contexto formal correspondiente a los planetas del Sistema Solar descritos por un conjunto de atributos. Las siglas se corresponden con TP=Tamaño pequeño, TM=Tamaño mediano, TG=Tamaño grande, DC=Distancia al sol cercana, DL=Distancia al sol lejana, LS=Posee luna, LN=No posee luna

Conceptos Formales

Siendo $K := (G, M, I)$ un contexto formal y A un subconjunto de G, se define A' como el conjunto de todos lo atributos del conjunto M que se aplican sobre todos y cada uno de los objetos de A. De igual modo, dado un subconjunto B del conjunto de atributos M, B' denota el conjunto de objetos pertenecientes a G sobre los que se aplican todos los atributos de B. La definición de estos conjuntos puede formalizarse del siguiente modo:

$$A \longmapsto A' = \{m \in M \mid gIm \text{ para todo } g \in A\} \tag{3.1}$$

$$B \longmapsto B' = \{g \in G \mid gIm \text{ para todo } m \in B\} \tag{3.2}$$

Un concepto formal viene representado por un par extensión-intensión, aunque no todos los posibles pares obtenidos a partir de un contexto formal definen conceptos formales. Siendo A un subconjunto del conjunto de objetos G y B un subconjunto del conjunto de atributos M, el par (A, B) define un concepto formal sobre el contexto K, y se representa mediante $c(A, B)$, si y solo si:

$$c(A, B) \Longleftrightarrow (A' = B) \wedge (B' = A) \tag{3.3}$$

Es decir, si B contiene todos y cada uno de los atributos pertenecientes a M que se aplican sobre todos los objetos de A y, por otro lado, A contiene todos y cada uno de los objetos de G que tienen todos los atributos de B.

Siendo el par (A, B) un concepto formal, los conjuntos A y B definen la extensión y la intensión del concepto respectivamente y además cumplen las igualdades $A'' = A$ y $B'' = B$.

En general, la definición de concepto formal impone restricciones importantes, haciendo que el número real de conceptos correspondientes a un contexto concreto sea bastante pequeño en compa-

ración con todos los pares extensión-intensión posibles. Además, podemos afirmar que este número crece de manera lineal con el número de objetos dentro del contexto.
Volviendo al ejemplo presentado en el cuadro 3.1, el par:

$$((Pluton, Jupiter, Saturno, Urano, Neptuno), (DL, LS)) \tag{3.4}$$

sería un concepto formal dado que cumple:

$$(Pluton, Jupiter, Saturno, Urano, Neptuno)' = (DL, LS) \tag{3.5}$$

y

$$(DL, LS)' = (Pluton, Jupiter, Saturno, Urano, Neptuno) \tag{3.6}$$

Sin embargo, el par:

$$((Tierra, Saturno), (DL, LS)) \tag{3.7}$$

no sería un concepto formal dado que:

$$(Tierra, Saturno)' = \emptyset \tag{3.8}$$

que no coincide con la intensión (DL, LS).

Conceptos Objeto y Conceptos Atributo

Existen un tipo de conceptos derivados de un contexto formal y especialmente importantes debido a que pueden ser generados a partir de un único objeto o un único atributo del contexto. Este tipo de conceptos reciben el nombre de *conceptos objeto* y *conceptos atributo* respectivamente. Se denomina concepto objeto al concepto generado por un objeto $g \in G$, mientras que al concepto generado por un atributo $m \in M$ se le denomina concepto atributo. La formalización de este tipo de conceptos viene dada por:

$$\gamma(g) \stackrel{def}{=} \left(\{g\}'', \{g\}'\right) \equiv \left(g'', g'\right) \tag{3.9}$$

$$\mu(m) \stackrel{def}{=} \left(\{m\}', \{m\}''\right) \equiv \left(m', m''\right) \tag{3.10}$$

Los conceptos (g'', g') son los conceptos más específicos que incluyen al objeto g en su extensión, mientras que los conceptos (m', m'') son los conceptos más genéricos que incluyen el atributo m en su intensión.
Nuevamente, retomando el ejemplo presentado en el cuadro 3.1, del conjunto de todos los conceptos formales posibles únicamente unos pocos cumplirán las propiedades necesarias para ser considerados concepto objeto o concepto atributo. En concreto, el concepto presentado anteriormente:

$$((Pluton, Jupiter, Saturno, Urano, Neptuno), (DL, LS)) \tag{3.11}$$

puede ser considerado un concepto atributo $\mu(DL)$, lo que significa que no existe un concepto formal más genérico que contenga dicho atributo en su intensión. Sin embargo, no puede ser considerado concepto objeto de cualquiera de los objetos de su extensión dado que existen conceptos mucho más específicos que lo especializan y que, por lo tanto, no cumplen las definiciones dadas en la ecuación 3.9:

$$\gamma(Pluton) = ((Pluton), (TP, DL, LS)) \tag{3.12}$$

$$\gamma(Jupiter) = ((Jupiter, Saturno), (TG, DL, LS) \tag{3.13}$$

$$\gamma(Saturno) = ((Jupiter, Saturno), (TG, DL, LS) \tag{3.14}$$

$$\gamma(Urano) = ((Urano, Neptuno), (TM, DL, LS) \tag{3.15}$$

$$\gamma(Neptuno) = ((Urano, Neptuno), (TM, DL, LS) \tag{3.16}$$

3.1.2. Orden Conceptual y Retículos de Conceptos

El conjunto de todos los conceptos obtenidos a partir de un contexto $K := (G, M, I)$ se denota por $\beta(G, M, I)$ o $\beta(K)$. Sobre este conjunto se define una relación binaria $\leq$ del siguiente modo. Siendo $c_1 = (A_1, B_1)$ y $c_2 = (A_2, B_2)$ dos conceptos pertenecientes al contexto K, se dice que $c_1 \leq c_2$, es decir, c_1 es un *subconcepto* de c_2, si y solo si $A_1 \subseteq A_2$ (o de igual modo si $B_1 \supseteq B_2$). En este caso, también se puede decir que c_2 es un superconcepto de c_1.

La relación binaria propuesta cumple las propiedades reflexiva, transitiva y antisimétrica, y se corresponde con la idea de que un concepto siempre tiene una extensión más pequeña y una intensión más grande que cualquiera de sus superconceptos.

La relación de subtipo-supertipo definida sobre el conjunto $\beta(G, M, I)$ se denomina *generalización especialización* y, debido a que cumple las propiedades reflexiva, transitiva y antisimétrica, define una relación de orden sobre β siendo, por tanto, $\underline{\beta}(\beta(G, I, M), \leq)$ un conjunto ordenado [53].

De acuerdo al ejemplo presentado en el cuadro 3.1, una posible relación de generalización especialización entre alguno de los conceptos obtenidos sería $c_1 \geq c_2$ donde:

$$c_1 = ((Pluton, Jupiter, Saturno, Urano, Neptuno), (DL, LS)) \tag{3.17}$$

y

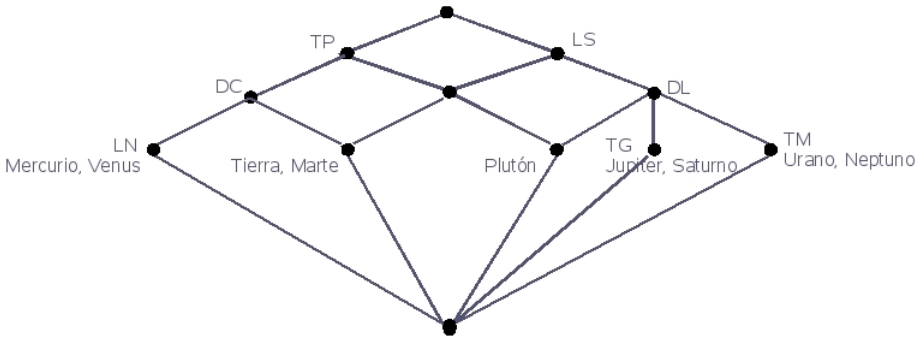


Figura 3.1: Diagrama de Hasse correspondiente al contexto formal presentado en el cuadro 3.1

$$c_2 = ((Pluton), (TP, DL, LS)) \tag{3.18}$$

Los elementos de un conjunto ordenado se pueden comparar. Dados dos elementos p y q pertenecientes a un conjunto ordenado P, se afirma que p y q son comparables si $p \leq q$ o $q \leq p$, en cualquier otro caso se dice que p y q son no comparables.

Dentro de un conjunto ordenado $(P, \leq)$ se puede definir una relación de vecindad $\prec$ entre sus elementos. Dados dos elementos p y q pertenecientes a un conjunto ordenado $(P, \leq)$, se dice que p es el *vecino inferior* de q o, de igual modo, que q es el *vecino superior* de p si y sólo si se cumple: a) $p \leq q$ y $p \neq q$ y; b) para todo elemento r perteneciente a P si $p \leq r \leq q$ entonces $r = p$ o $r = q$. Si $(P, \leq)$ es un conjunto ordenado finito, la relación de orden $\leq$ se encuentra directamente determinada por la relación de vecindad que se acaba de definir [53].

Volviendo al ejemplo anterior, $c_1 \succ c_2$, dado que no existe ningún otro concepto formal intermedio r tal que $c_1 \geq r \geq c_2$.

El conjunto ordenado $(P, \leq)$ que acabamos de definir puede representarse mediante un *diagrama de líneas* o *diagrama de Hasse*. En un diagrama de este tipo, los elementos de P se representan mediante pequeños círculos de tal forma que, cuando un elemento p es vecino inferior de otro elemento q (cuando $p \prec q$), el elemento q se representa por encima del elemento p y ambos quedan unidos por una línea que no pasa por ningún otro elemento del conjunto. A partir de un diagrama de este tipo es posible deducir cualquier relación $v \leq w$ existente entre los elementos del conjunto siguiendo la sucesión de líneas ascendentes que van desde el nodo que representa al elemento v hasta el nodo que representa al elemento w.

En el caso concreto del conjunto $\underline{\beta}(\beta(G, I, M), \leq)$, cada nodo representa un concepto y se etiqueta con el conjunto de objetos y de atributos que lo definen. Los objetos se sitúan en la parte inferior derecha del nodo, mientras que los atributos se disponen en su parte superior derecha. Por convenio, los nombres de los objetos se escribirán sólo en los nodos correspondientes a los conceptos objeto generados a partir de cada uno de los objetos y los nombres de los atributos se escribirán en la parte superior derecha de los nodos que representen conceptos atributo generados a partir de cada uno de los atributos. Todos los nodos que se encuentren por encima de un nodo que contenga al objeto

$g \in G$ también contendrán al objeto g en su extensión. De igual forma, todos los nodos que se encuentren por debajo de un nodo que contenga al atributo $m \in M$ también contendrán al atributo m en su intensión. La forma de leer la extensión de un concepto dentro del diagrama es tomando como extensión del concepto todos los objetos que definan al propio nodo y a cualquier otro nodo que se encuentre en el camino descendente desde el nodo hacia la parte inferior del retículo. De igual forma, la intensión de un concepto la podremos obtener tomando los atributos del propio nodo y los de todos los nodos que se encuentren en el camino ascendente desde el nodo hacia la parte superior del retículo. La figura 3.1 presenta el diagrama de Hasse correspondiente al contexto formal presentado en el cuadro 3.1.

Siendo N un subconjunto de $(P, \leq)$, se dice que un elemento q perteneciente a P es un límite superior del subconjunto N si $n \leq q$ para cualquier elemento n de N. Análogamente, se dice que el elemento p perteneciente a P es un límite inferior del subconjunto N si $p \leq n$ para cualquier elemento n de N.

Cuando este existe, se define el *supremo* (o menor de los límites superiores) del subconjunto N como el límite superior q_0 de N tal que $q_0 \leq q$ para cualquier q que sea límite superior de N. De igual modo, se define el *ínfimo* (o mayor de los límites inferiores) del subconjunto N como el límite inferior p_0 de N que cumple $p_0 \leq p$ para cualquier p que sea límite inferior de N. El ínfimo y el supremo de un subconjunto N de P se denotan por Inf N y Sup N.

3.1.3. Teorema Fundamental del AFC

Los conceptos obtenidos a partir de un contexto (G, M, I) y la relación de orden aplicada sobre este conjunto de conceptos definen una jerarquía conceptual representada por $\underline{\beta}(\beta(G, I, M), \leq)$. A continuación se presenta el teorema fundamental del análisis formal de conceptos, cuyo objetivo es establecer la correspondencia entre las teorías de conjuntos ordenados y la teoría general de retículos.

Se dice que un conjunto ordenado $(P, \leq)$ es un retículo cuando para cualquier par de elementos pertenecientes a P existen el supremo y el ínfimo. El hecho de que P sea un retículo implica la existencia de un supremo y un ínfimo para cualquier subconjunto finito y no vacío de P. Si P es además finito, también está definido el Sup $\emptyset$ y el Inf $\emptyset$ [53]. En la teoría de retículos, al supremo de un conjunto de elementos se le denomina *join* y al ínfimo se le denomina *meet*[2].

El teorema fundamental del análisis formal de conceptos afirma que el conjunto $\underline{\beta}$ de todos los conceptos pertenecientes a un contexto K es siempre un retículo completo [53]. Sobre este retículo se pueden formalizar los meets y los joins entre conceptos del siguiente modo:

$$\bigwedge_{t \in T} (A_t, B_t) = \left(\bigcap_{t \in T} A_t, \left(\bigcup_{t \in T} B_t \right)'' \right) \tag{3.19}$$

$$\bigvee_{t \in T} (A_t, B_t) = \left(\left(\bigcup_{t \in T} A_t \right)'', \bigcap_{t \in T} B_t \right) \tag{3.20}$$

[2] A lo largo de este trabajo utilizaremos los términos *meet* y *join* sin traducir al castellano

El meet de un conjunto de conceptos es una especialización de dichos conceptos y se corresponde con el mayor concepto que es subconcepto de los conceptos de partida. El conjunto de objetos de este subconcepto lo forman los objetos comunes a los conceptos de los que se parte, mientras que el conjunto de atributos está formado por los atributos comunes al conjunto de objetos del subconcepto obtenido.

El join de un conjunto de conceptos es la generalización de dichos conceptos y se corresponde con el menor concepto posible que es superconcepto de los conceptos de partida. El conjunto de atributos de este superconcepto se encuentra formado por los atributos comunes a los conceptos de los que se parte y el conjunto de objetos lo forman los objetos comunes al conjunto de atributos del superconcepto obtenido.

3.2. Paradigmas de Visualización

Todas las aplicaciones basadas en AFC requieren definir una forma de explorar el retículo obtenido. La obtención de visualizaciones de estas estructuras que sean útiles al usuario es una tarea compleja. Esto es debido a que habitualmente se presentan conflictos entre el tamaño, la distribución y la legibilidad del área mostrada al usuario. Además, la multitud de cruces entre los diferentes enlaces entre los conceptos dificulta notablemente su comprensión por parte del usuario. Finalmente, debemos considerar como un aspecto crítico la velocidad a la que se generan estas visualizaciones, de modo que minimizar el número de cruces generados entre los enlaces implicará un aumento del coste computacional asociado al proceso.

Con el fin de reducir el tamaño de los retículos que deben ser visualizados por el usuario, habitualmente se utilizan técnicas incrementales o de escala que permiten abordar este problema sobre contextos muy grandes. La aproximación más común se basa en permitir al usuario examinar un subconjunto de conceptos y enlaces en función de una tarea determinada y en las interacciones anteriores realizadas sobre el sistema. En este escenario, el sistema debería de ser capaz de mostrar u ocultar partes del retículo vía especificación directa o manipulación interactiva de los conceptos o de las relaciones entre estos.

Las tres aproximaciones principales para llevar a cabo la visualización de retículos obtenidos mediante AFC son las siguientes:

- *Carpetas jerárquicas.* Están orientadas a reducir el retículo de conceptos a una estructura muy simple que pueda ser mostrada de manera sencilla en un entorno visual. Dado que un retículo de conceptos puede ser representado de manera sencilla mediante un árbol, esta resulta ser la opción más adecuada. Para su construcción se aprovechan las relaciones de subsunción existentes entre los conceptos. El elemento top del retículo será considerado la raíz del árbol y cada secuencia de conceptos sobre el retículo estará asociado con un camino dentro del árbol.

  Dado que un concepto formal se caracteriza porque puede heredar sus atributos de más de un concepto padre, al aplicar una representación basada en árboles es posible la existencia de diferentes caminos (ramas dentro del árbol) que conduzcan a un mismo concepto formal.

La principal ventaja de este tipo de representación está asociada a la familiaridad del usuario con este tipo de estructuras, lo que simplifica notablemente su proceso de aprendizaje a la hora de enfrentarse a un sistema FCA que lo implemente. Además, pueden ser dibujadas de manera eficiente y prácticamente no ocupan espacio en pantalla. No obstante, esta aproximación también presenta desventajas, la principal está relacionada con la excesiva duplicación de información debida a la herencia múltiple que acabamos de comentar.

La forma más habitual de realizar este tipo de representaciones es en formato texto (asociando una etiqueta a cada nodo del árbol) [30, 89, 32], aunque existen aproximaciones recientes mucho más visuales y adecuadas para representar grandes árboles tales como los *árboles cónicos* (*cone trees*) [92], los *mapas árbol* (*tree maps*) [72], las *pirámides de información* (*information pyramids*) [6] o los *árboles radiales* [117]. Como contrapartida, estas aproximaciones, aunque novedosas, no resultan tan familiares al usuario, reduciendo notablemente la usabilidad de los sistemas que las implementan.

- *Diagramas anidados.* En algunas ocasiones resulta útil separar una representación compleja en múltiples visualizaciones más comprensibles individualmente. Esta es la idea que se esconde detrás de esta aproximación, que puede resumirse en los siguientes cuatro pasos:

  1. Particionar el conjunto de atributos que describen un contexto concreto en dos subconjuntos (aunque esta partición puede llevarse a cabo sobre más de dos subconjuntos de atributos).
  2. Encontrar los retículos de conceptos L_1 y L_2 de los subcontextos correspondientes a los subconjuntos de atributos identificados en el paso anterior.
  3. Realizar una copia de L_2 en cada nodo de L_1.
  4. Marcar con círculos rellenos los elementos de cada copia de L_2 tales que pertenezcan al retículo completo.

  En otras palabras, L_1 es utilizado como un marco externo sobre el que se inserta L_2, cuyas copias hacen las veces de estructura interna. Como resultado, los conceptos y los enlaces del retículo completo no se representan en un diagrama anidado. En su lugar, éstos pueden ser derivados combinando la información asociada a los diferentes niveles de anidamiento.

  Esta técnica resulta ser adecuada sobre contextos multivaluados, debido a que resulta mucho más sencillo identificar los subcontextos potencialmente útiles. La metáfora de las escalas conceptuales ha sido eficientemente implementada en los sistemas Anaconda y Toscana [123, 124, 10], cuya funcionalidad es la de definir un marco de trabajo para la creación y combinación de escalas conceptuales mediante diagramas anidados. Las escalas conceptuales se definen en un primer paso en Anaconda para ser posteriormente analizadas utilizando Toscana.

  El mecanismo básico de búsqueda ofrecido por el sistema está basado en operaciones de anidamiento y zoom cuyas funcionalidades son las siguientes: El anidamiento permite insertar el diagrama de una nueva escala en los nodos de la escala actual, mientras que el zoom permite

al usuario seleccionar un concepto concreto de la escala externa y mostrarlo mediante un diagrama aislado.

Una de las ventajas de los diagramas anidados es que el tamaño de cada uno de los diagramas locales nunca puede exceder del número de combinaciones posibles entre los valores de los atributos presentes en el subcontexto correspondiente, independientemente del número de objetos. De este modo, es posible dibujar los retículos completos de cada subcontexto, incluso en el caso de tratar con grandes bases de datos o corpus de documentos, siempre y cuando los subcontextos sean suficientemente pequeños.

Un refinamiento adicional, denominado *escala local* (*local scaling*), consiste en omitir todos los conceptos anidados que sean irrelevantes a la faceta correspondiente del retículo externo.

La idea de los diagramas anidados fue desarrollada e implementada principalmente por Rudolf Wille y su grupo de investigación [131, 123, 124, 133], siendo interesantes las aportaciones dirigidas a la eliminación de información visualmente redundante y a la mejora en la eficiencia. En este último aspecto, Cole y Eklund describen en [26] un procedimiento eficiente para la construcción de subcontextos de interés a partir de un contexto univaluado relativamente extenso del dominio médico. Los diagramas anidados pueden construirse de manera incremental conforme se añaden nuevos atributos a los subcontextos [63].

- *Vistas foco+contexto.* Las técnicas basadas en esta aproximación consideran que el retículo se construye a partir del conjunto completo de atributos, aunque no se muestre en su totalidad, y no todas sus partes sean construídas del mismo modo.

  En esta aproximación, se suele dar una mayor importancia a una parte del espacio de visualización donde supuestamente se encuentran situados los intereses del usuario. En otras palabras, se asume que existe un foco actual de interés dentro del retículo, que se corresponde con un concepto concreto y que puede ser identificado de diferentes maneras dependiendo del tipo de aplicación o sistema sobre el cual se aplique este tipo de visualización.

  La principal característica de esta aproximación es la integración natural del foco con el contexto próximo. En contraste con otras aproximaciones basadas en múltiples vistas, la transición del foco hacia el contexto es contínua, permitiendo una ampliación variable de la información de acuerdo a las necesidades del usuario.

  En esencia, la información contenida en el retículo se muestra al usuario de acuerdo a diferentes niveles de detalle dependiendo de la distancia al foco. La información próxima al foco se incrementa, mientras que la información más distante se reduce. La metáfora asociada a esta aproximación se corresponde con el efecto obtenido al mirar a través de una lente *ojo de pez*.

  La metáfora del ojo de pez fue adoptada por primera vez para la visualización de retículos de conceptos en el prototipo ULYSES [14, 15].

Capítulo 4

Recuperación de Información con AFC

En este capítulo presentaremos las principales aportaciones realizadas por el área del AFC en tareas de recuperación de información. Centraremos nuestra atención en las aportaciones para la expansión de consultas basada en retículos, así como en el uso de AFC para generar rankings de documentos. En la segunda parte del capítulo abordaremos los aspectos relacionados con la evaluación de sistemas basados en AFC para realizar tareas de RI, así como en los paradigmas de visualización más utilizados. Finalmente, haremos una recapitulación donde expondremos cuales son las aportaciones de nuestra propuesta frente a las aproximaciones presentadas.

4.1. Introducción

La aplicación de AFC a tareas de recuperación de información es un área en creciente expansión que ha llegado a producir, incluso, algunas aplicaciones comerciales. Sin embargo, la mayor parte de las investigaciones desarrolladas se basan en el uso tesauros o esquemas de clasificación preestablecidos, lo que convierte su aplicación en una tarea manual o semiautomática.

El uso de retículos en recuperación de información se ha focalizado principalmente en resolver dos aspectos:

- La modificación de las consultas realizadas por el usuario. AFC puede aplicarse para modificar una consulta inicial con el fin de facilitar el acceso a un conjunto de documentos relevantes. Este tipo de mecanismos favorecen la recuperación del usuario de situaciones en las que el conjunto de documentos recuperado es muy extenso o, en contraste, muy pequeño como para satisfacer plenamente sus necesidades de búsqueda.
- La ordenación o el ranking de documentos en respuesta a una consulta concreta. La aplicación de AFC permite calcular la distancia entre las consultas realizadas por el usuario y los documentos contenidos en el corpus.

En los siguientes apartados describimos cada una de estas aproximaciones exponiendo, para cada una de ellas, los trabajos más relevantes realizados.

4.2. Modificación de consultas

La estrategia de modificación de consultas permite al usuario resolver aquellas situaciones donde el conjunto de documentos obtenidos a partir de una consulta concreta es demasiado grande o demasiado pequeño, siendo los retículos de conceptos adecuados para este tipo de tareas. Esto es debido a que una consulta puede mapearse de manera sencilla en este tipo de estructuras, permitiendo al usuario modificar su consulta gradualmente. El uso de tesauros en estas tareas mejora los resultados al aplicar AFC.

Las primeras aproximaciones a la modificación de consultas sobre retículos puros (i.e. no retículos de conceptos formales) se deben a [115] y, más recientemente, a [116], estando orientadas a la formalización booleana de un retículo a partir de una consulta realizada por el usuario. Su principal desventaja es que el número de refinamientos propuestos en ocasiones es muy elevado (incluso para un número limitado de términos), no siendo muchos de ellos semánticamente útiles para el usuario. Así mismo, Godin [57] fue uno de los primeros en aplicar AFC al área de la RI. El sistema propuesto definía un contexto formal donde los objetos eran un conjunto de documentos y los atributos un conjunto de términos. De este modo, cada concepto formal relacionaba un conjunto de documentos (extensión) con un conjunto de términos que lo describían (intensión).

Las razones que motivan el uso y adecuación de los retículos y la teoría del AFC en este tipo de tareas son las siguientes:

- Cada nodo del retículo puede entenderse como una consulta formada por la conjunción de los términos de su intensión con respecto al conjunto de documentos recuperados en dicho nodo.
- Aquellos enlaces que, partiendo de un nodo concreto, ascienden o descienden hacia otros nodos representan los refinamientos o generalizaciones conjuntivas mínimas de la consulta con respecto al corpus indexado, no siendo posible la adición o eliminación de aquellos términos que den lugar a conceptos intermedios.

El retículo obtenido mediante un proceso de este tipo permite sugerir al usuario aquellos términos que podrían ser eliminados o añadidos a la intensión del concepto actual con el fin de realizar una generalización o un refinamiento mínimo de la consulta. Estas propiedades pueden explotarse para refinar una consulta realizada por el usuario sobre un corpus de documentos concreto.

Tomando la consulta formulada a partir de un subconjunto de términos de $A \subseteq M$ (donde M es el conjunto de atributos considerado para construir el retículo a partir del corpus de documentos), ésta se puede mapear en el retículo de conceptos con el fin de encontrar el concepto formal correspondiente a dicha consulta (denominado *concepto consulta* o *query concept*). Este concepto se define como aquel cuya intensión es igual a la consulta realizada, en el caso de existir alguno, o aquel concepto más general con una intensión mayor que A. Estará caracterizado por el meet del conjunto de conceptos atributo asociados a cada término de la consulta ($\bigwedge_{a \in A} \mu(a)$). De acuerdo a esta definición, consultas muy específicas podrían hacer que el concepto consulta coincidiera con el concepto bottom del retículo (que contendrá todos los términos indexados y, probablemente, ningún documento). En este caso particular resulta mucho más eficiente mostrar como concepto consulta

un conjunto de conceptos próximo a la consulta realizada que el concepto que realmente se corresponde con ésta [118]. Con el fin de facilitar la interacción con el usuario, habitualmente se presenta una interfaz gráfica donde figura el concepto consulta y el conjunto de conceptos vecinos (que debe seleccionar para realizar la generalización o el refinamiento de la consulta) utilizando cualquiera de las aproximaciones de visualización propuestas en 3.2 [84].

Dado que la construcción de un retículo sobre un corpus de documentos completo tiene un alto coste computacional, existen diferentes aproximaciones que no hacen necesaria su construcción completa. Una de ellas consistiría en determinar el concepto consulta a partir del contexto formal generado a partir del corpus de documentos completo, construyendo entonces una pequeña porción del retículo centrada alrededor de dicho concepto. La extensión del concepto estará formada por todos los documentos que contengan los términos de la consulta y la intensión la determinarán todos los términos comunes al conjunto de documentos obtenido. Una vez obtenido el concepto consulta, resulta muy sencillo obtener el conjunto de vecinos aplicando el algoritmo de los vecinos más cercanos que describiremos en la sección 6.6.

La aproximación propuesta puede generalizarse sobre escenarios basados en consultas booleanas (consultas expresadas mediante operadores booleanos). En este caso, la intensión del concepto consulta contendrá todos los todos los documentos que satisfagan la consulta booleana, mientras que la intensión se determinará del mismo modo que se determina la intensión sobre consultas no booleanas. Se han propuesto métodos heurísticos para realizar refinamientos de consultas booleanas basados, por ejemplo, en el debilitamiento de las clausulas *AND* mediante la eliminación de alguno de sus términos [23], o en conjuntos de reglas para modificar los constituyentes de una expresión booleana cuando el usuario se encuentra frente a una situación adversa [12]. El sistema Refiner [16] ilustra esta situación.

La principal ventaja de un sistema de este tipo es que permite una reformulación de la consulta guiada por el contenido del propio corpus, lo que supone proporcionar al usuario un conjunto de alternativas controlado y finito.

4.2.1. Mejora de la navegación mediante el uso de tesauros

En un retículo de conceptos, la proximidad de dos o más nodos es independiente del significado de los términos que los describen. De este modo, dos consultas semánticamente próximas pueden estar muy distantes en el retículo de conceptos obtenido a partir del corpus. La aplicación de un tesauro sobre los términos contenidos en los documentos permite generar un retículo de conceptos que aproxime aquellos términos semánticamente similares. De este modo, la relación de orden aplicada sobre los conceptos pertenecientes a los retículos no será independiente de la relación de orden existente entre los términos. Aquellos términos más generales serán indexados en conceptos más generales, lo que facilita al usuario la localización de aquellos nodos que relacionen consultas semánticamente similares. En [52, 141] se propuso la utilización de un tesauro en tareas de recuperación interactivas con el objeto de mejorar la cobertura sin perder precisión en el proceso.

Además, un tesauro puede utilizarse para especificar directamente una consulta más general o específica seleccionando, a partir del concepto consulta, una especialización o generalización en base

a las relaciones de proximidad contenidas en el tesauro. Una navegación de este tipo permite realizar movimientos mucho más distantes dentro del retículo (basadas en la semántica de los términos), en comparación con la navegación básica implementada por este tipo de aproximaciones. Galois Browser [15], es el sistema más representativo. Se caracteriza por presentar, para el concepto actualmente seleccionado, diferentes opciones de manipulación. En concreto, permite la especialización de la consulta mediante la selección de términos en una ventana de vocabulario y el refinamiento de la consulta mediante la selección de términos en una ventana de tesauro.

4.2.2. Extracción automática de términos de indexación

La última de las aproximaciones orientada a la modificación de consultas se basa en la extracción automática de términos de indexación para describir los documentos que serán organizados sobre el retículo resultante. El proceso habitual para llevar a cabo esta extracción se basa en los pasos siguientes (que describiremos en detalle en el capítulo 5, adaptados a nuestro trabajo):

- *Segmentación del texto.* Consiste en extraer del corpus de documentos el conjunto de términos contenidos en éste, ignorando los signos de puntuación.

- *Normalización de los términos.* Cada término puede reducirse a una forma normal basada en el truncamiento del término o en su lematización. Esta tarea se realiza haciendo uso de un extractor de raíces o de un lematizador creado a partir de una base de conocimiento léxico.

- *Eliminación de palabras vacías.* Consiste en eliminar del texto aquellos términos que no aportan contenido semático.

- *Pesado de los términos.* Este paso resulta crítico para llevar a cabo la selección de los términos utilizados para construir el retículo de conceptos. La aproximación más habitual consiste en determinar, para cada término, una medida de su importancia dentro de cada documento. El objetivo es identificar aquellos términos que caracterizan el documento frente al resto de documentos del corpus. Existen una gran variedad de aproximaciones para llevar a cabo este pesado, siendo el esquema $tf - idf$ el más utilizado. En este esquema el peso tiene relación directa con el número de ocurrencias del término en el documento (*term frequency* o *tf*) y una relación inversa con la frecuencia de aparición del término dentro del corpus (*inverse document frequency* o *idf*). En algunas ocasiones se realiza una normalización de este peso considerando la longitud de los documentos. En este sentido, dadas las mismas ocurrencias de un conjunto de términos sobre dos documentos, el documento más largo será menos importante que el documento más corto. El marco de pesado presentado ha sido implementado dentro de la mayoría de los modelos de RI, de los cuales el más conocido es el modelo del espacio vectorial [109]. Sobre documentos semiestructurados o documentos web, el proceso de pesado puede complementarse con otras técnicas que permitan aprovechar las fuentes de conocimiento adicionales tales como la metainformación contenida en los documentos, los enlaces entrantes y salientes, los textos ancla o la estructura de su URL.

- *Selección de términos.* Este último paso no se hace necesario en escenarios orientados a una tarea habitual de recuperación. Sin embargo, resulta crítico cuando se desea llevar a cabo una tarea de este tipo aplicando AFC. Esto es debido a que la construcción del retículo depende de un contexto formal obtenido a partir de un conjunto de términos previamente seleccionados. En este sentido, el conjunto de términos seleccionados está directamente relacionado con la efectividad del proceso de recuperación, siendo uno de los pasos más complejos a la hora de generar un retículo útil desde el punto de vista de una tarea de RI. El proceso de selección suele llevarse a cabo aplicando heurísticas que permitan seleccionar aquellos términos cuyo peso sea adecuado.

La aplicación de esta aproximación sobre diversos tipos de sistemas de RI basados en retículos de conceptos se ha llevado a cabo en [14, 15, 17, 101]. Así mismo, también ha sido utilizada sobre tesauros de dominio específico [25, 26] o WordNet [68].

4.3. Ordenación de documentos

Los retículos de conceptos no sólo pueden ser utilizados para llevar a cabo un refinamiento de consultas interactivo, sino que también pueden aplicarse a la ordenación o ranking automático de documentos. En este caso, la estructura del retículo es utilizada para calcular una distancia conceptual entre la consulta y cada uno de los documentos, favoreciendo la recuperación de documentos relevantes que no coincidan exactamente con los términos de la consulta.

4.3.1. El problema del vocabulario

Cuando un usuario realiza una consulta sobre un sistema de RI, éste debe transformar la conceptualización de sus requerimientos en términos adecuados para realizar su consulta. Esta situación hace que no siempre coincidan los términos utilizados con los términos utilizados por los autores para describir los mismos conceptos en los documentos. Este hecho, que habitualmente se traduce en una baja precisión de los sistemas de RI, está relacionado con aspectos tales como la polisemia o la sinonimia de los términos utilizados en la consulta, siendo especialmente apreciable en consultas cortas aplicadas sobre corpus excesivamente grandes y heterogéneos (como por ejemplo la Web). La dificultad de detectar coocurrencias de términos sobre las consultas y los documentos recuperados no permite abordar con garantías de éxito el problema de la sinonimia y la aplicación de un proceso de recuperación sobre un corpus de documentos heterogéneo incrementa el problema de la polisemia.

El problema del vocabulario ha sido abordado desde diferentes perspectivas tales como la realimentación por relevancia (*relevance feedback*) [64], el uso de tesauros de propósito general [126], la realimentación local (*local feedback*) [137, 138] o la aplicación de tesauros y redes léxico-semánticas sobre colecciones específicas [31]. Una solución habitual a este problema explota las relaciones de contenido existentes entre los documentos del corpus a la hora de decidir qué documentos serán recuperados en respuesta a una consulta concreta. La aproximación más conocida consiste en una organización de los documentos basada en clusters [119, 67].

Una ordenación de los documentos basada en clustering jerárquico toma como entrada una matriz de distancias interdocumento (basadas en una función de similitud) e iterativamente une los pares de clusters más próximos, mediante cualquiera de las estrategias de clustering presentadas en la sección 2.1, hasta obtener un único cluster. Una vez construida la jerarquía, la aplicación de una estrategia de búsqueda y el uso de una función de similitud consulta-cluster permite obtener un ranking de los clusters respecto a una consulta específica. El resultado de una ordenación de clusters basada en este método es un conjunto parcialmente ordenado de documentos. Debido a que los documentos contenidos en un cluster son considerados igualmente similares a la consulta realizada, es posible llevar a cabo su ordenación comparándolos de manera individual con la consulta realizada.

La aplicación de este tipo de técnicas resuelve, en parte, el problema del vocabulario dado que estos métodos tienen en cuenta tanto la similitud interdocumento con la similitud entre la consulta realizada y cada uno de los documentos considerados individualmente. En contraste, este tipo de técnicas presentan una serie de inconvenientes tales como su dependencia de parámetros libres, la imposibilidad de realizar clasificaciones múltiples o clasificaciones cruzadas y depender de estrategias esencialmente heurísticas.

4.3.2. Ordenación basada en retículos de conceptos

La concordancia entre una consulta y un documento específico puede describirse en términos de una secuencia de operaciones capaces de transformar la consulta en el documento. En particular, las operaciones básicas para transformar el vector de términos de la consulta en el vector de términos de un documento puede entenderse como la adición y eliminación de términos, mientras que la longitud de la secuencia que hace posible dicha transformación puede entenderse como una medida de la similitud entre ambos vectores.

Aunque esta idea se podría implementar de manera directa contando el número de términos no compartidos entre el vector consulta y el vector documento, esta aproximación no resuelve el problema del vocabulario que acabamos de presentar. En contraste, una aproximación basada en retículos de conceptos presenta una serie de propiedades que pueden facilitar el proceso de ordenación final de los documentos:

- Los conceptos formales permiten caracterizar de manera directa y natural el conjunto de consultas válidas sobre la colección.
- Cualquier consulta puede ser mapeada en un concepto consulta, tal y como se expuso en la sección 4.2.
- La relación de orden definida sobre el conjunto de conceptos obtenido proporciona información acerca del modo en el puede transformarse una consulta.
- El conjunto de conceptos contenido en un retículo es suficientemente grande como para representar de manera adecuada los documentos.

Al igual que cada consulta puede relacionarse con un concepto consulta, cualquier documento d de un corpus puede relacionarse con un *concepto documento* (*document concept*), definido como

(d'', d'). De este modo, el problema de obtener la secuencia de transformaciones de una consulta en un documento concreto podría obtenerse a partir de una búsqueda en anchura sobre el espacio parcialmente ordenado de consultas posibles. El estado inicial sería el concepto consulta, mientras que los estados siguientes se obtendrían aplicando la *relación vecino más próximo* (*nearest neighbour relation*), denotada por $\succ\prec$ y definida del siguiente modo:

$$x \succ\prec y \text{ si } x \prec y \vee x \succ y \tag{4.1}$$

El proceso terminará cuando se alcance el concepto documento, determinando la posición del documento dentro del ranking a partir del camino más corto entre el concepto consulta y el concepto documento. En la práctica resulta útil eliminar del retículo obtenido los conceptos top y bottom. Esto es debido a que no pueden considerarse estrictamente consultas relacionadas con documentos concretos, excepto en el caso en el que dispongan de un conjunto de términos o de documentos no vacío respectivamente.

Los métodos para llevar a cabo la visualización de este tipo de aproximaciones pueden utilizar cualquiera de las propuestas presentadas en la sección 3.2, aunque resulta especialmente adecuado aplicar una aproximación foco+contexto centrada alrededor del concepto consulta.

La aproximación de ordenación basada en retículos de conceptos fue originalmente propuesta por Carpineto y Romano en [16].

4.4. Evaluación de sistemas RI basados en AFC

Hasta la fecha, el área del AFC aplicada a una tarea de RI no ha propuesto una metodología concreta orientada a la evaluación automática de la efectividad del proceso de recuperación, existiendo únicamente evaluaciones puntuales realizadas con usuario reales.

Las dos primeras aplicaciones sobre las cuales se realizaron evaluaciones con usuarios reales fueron [58, 15]. En [58], se llevó a cabo una comparación, mediante una tarea de recuperación con usuarios, entre la navegación sobre un retículo de conceptos y una navegación jerárquica sobre los resultados obtenidos en un proceso de recuperación booleano. En el experimento, la cobertura obtenida utilizando retículos y recuperación booleana resultó ser superior a la obtenida mediante la navegación directa sobre la jerarquía. La colección de documentos consistió en 113 descripciones de cortos de animación descritos por una media de 6.53 términos asignados manualmente. En [15] se propuso un sistema de clustering basado en retículos que incorporaba conocimiento externo mediante la utilización de un tesauro en el proceso de construcción del retículo. La evaluación del sistema se orientó a comparar la navegación realizada por los usuarios cuando estos se enfrentaban al sistema según éste considerara o no éste conocimiento. El experimento se realizó sobre un corpus de 1555 documentos de Inteligencia Artificial extraídos de una colección de ingeniería informática (INSPEC). La navegación basada en retículos con conocimiento explícito produjo una mejora del 30 % sobre la cobertura, demostrando que la incorporación de relaciones específicas entre los términos de indexación mejoraba notablemente los retículos construídos con respecto a aquellos que no consideraban dichas relaciones.

Uno de los dominios de aplicación sobre los cuales se han producido un mayor número de sistemas basados en AFC es el dominio médico. En [25] se indexaron un total de 9000 expedientes médicos utilizando SNOMED (nomenclatura sistematizada para medicina), demostrando la viabilidad de aplicar AFC para mejorar el proceso de recuperación y navegación entorno a toda esta información. Esta aproximación tuvo continuación en [27, 24, 26], donde los documentos fueron indexados automáticamente utilizando UMLS (Unified Medical Language System), tesauros médicos e introduciendo las nociones de escala conceptual y contextos purificados con el fin de presentar una visualización mejorada y escalable del conocimiento. Desafortunadamente, ninguno de estos trabajos muestra evaluaciones empíricas o cuantitativas que permitan concluir la adecuación de estos sistemas en tareas de recuperación y acceso a la información.

El AFC ha sido combinado con tesauros de facetas con fines de RI, asociando esta aproximación a la noción de escala conceptual. En este caso, los diferentes aspectos relacionados con la descripción de un artículo (por ejemplo su tema o su nivel de dificultad) disponen de distintos descriptores sobre las facetas del tesauro. El sistema de recuperación de información FaIR [101] es un ejemplo de este tipo de aproximaciones. Se aplica a una colección on-line de 5000 documentos FAQ (Frequently Asked Questions) sobre informática. De igual modo, y aplicado sobre el dominio informático, Aran [45] es un sistema de ayuda que utiliza AFC para facilitar el acceso y recuperación de las páginas de ayuda *man* de Unix. Una característica destacable de este sistema es que no utiliza un tesauro sino que, en su lugar, los índices son obtenidos a partir del texto libre asociado a cada comando Unix en su descripción corta. Al igual que en el caso de las aplicaciones desarrolladas sobre el dominio médico, ninguna de estas aplicaciones fue evaluada cuantitativamente.

Un ejemplo muy sugerente acerca de las posibilidades del AFC para la organización y acceso al conocimiento es el sistema HIERMAIL [43, 30], que proporciona una ontología estructurada y un sistema de RI orientado a la búsqueda y descubrimiento de información en correos electrónicos. Aunque no existe una evaluación empírica de la utilidad de este sistema (quizás porque no resulta trivial diseñar una procedimiento de evaluación adecuado sobre tareas de administración y organización del conocimiento), una evidencia indirecta de su valor es su aplicación al desarrollo de una herramienta comercial denominada Mail-Sleuth [89].

Más recientemente, [29] combina la extracción de información sobre documentos web con AFC en un sistema de acceso a la información sobre el dominio de los anuncios clasificados. En lugar de proceder a extraer el conjunto de términos directamente de los documentos, el sistema utiliza plantillas para la extracción de los datos especialmente orientadas al dominio de los anuncios que permiten mejorar la calidad de los datos de entrada utilizados para construir el retículo.

Finalmente, dos trabajos próximos al presentado en esta Tesis Doctoral son [68, 19, 32]. En [68] Hotho y Stumme realizan un clustering de la colección de noticias Reuters-21578 combinando una técnica de clustering estándar, que es aplicada a la colección completa, con AFC, que es aplicado de manera individual sobre cada uno de los clusters generados. Una de las características principales de este sistema es que utiliza una base de conocimiento léxico (WordNet) en lugar de un tesauro de dominio específico tanto para realizar el proceso inicial de clustering como para el paso de aplicación de AFC. En la práctica, esto significa que la entrada utilizada para aplicar AFC está mucho más próxima a la utilización de términos libres para la indexación que en cualquiera de las aproxi-

maciones anteriormente presentadas. Hotho y Stumme aplican AFC a pequeños subconjuntos de la colección que han sido previamente agrupados mediante una técnica de clustering estándar, tomando como índices para el proceso de AFC aquellos términos del centroide del cluster con los valores más altos. La combinación de WordNet con los vectores centroides de los clusters es una alternativa interesante a la propuesta presentada en este trabajo.

En contraste, Credo [19, 32] es un sistema que organiza los resultados proporcionados por un motor de búsqueda de acuerdo a un esquema basado en AFC. Aunque en una primera aproximación pudiera parecer similar a la propuesta que realizamos en este trabajo, debemos decir que Credo se diferencia de los prototipos que presentaremos en el capítulo 8, tanto en el modo en el que el usuario navega a través de la información recuperada (paradigmas de visualización) como en el modo en el que los atributos utilizados para construir el contexto formal, y como consecuencia el retículo resultante, son extraídos y seleccionados. En Credo, la extracción de los términos utilizados para caracterizar el contexto formal se lleva a cabo seleccionando aquellos términos que mejor describen el contenido de los documentos, aunque no se especifica exactamente el modo en el que este pesado se lleva a cabo. Debemos destacar que, de acuerdo con nuestra experiencia en la utilización del sistema, la tipología de descriptor utilizada para describir los documentos está basada principalmente en unigramas, siendo bastante excepcional encontrar n-gramas de longitud mayor que uno describiendo alguno de los conceptos formales. Así mismo, no existe una evaluación formal del mismo que permita comparar su adecuación a la tarea de recuperación con respecto a sistemas similares. La figura 4.1 muestra los resultados del sistema Credo para la consulta *jaguar*.

4.5. Visualización de sistemas RI basados en AFC

Gran parte de las investigaciones realizadas en el área de la RI se han centrado en proponer una integración efectiva del modelo de recuperación con paradigmas de navegación adecuados, orientados a facilitar al usuario un acceso eficiente a la información recuperada.

Una opción habitual consiste en mantener diferentes métodos de búsqueda en paralelo [87, 56], donde el usuario pueda seleccionar en cualquier momento la estrategia a utilizar de manera independiente. Otra posible aproximación, un poco más estricta, se orienta a integrar más de una estrategia en cascada, de modo que se anteponga el proceso de navegación al de consulta [98], realizando el proceso de consulta antes que el de navegación [86], o permitiendo que ambos coexistan de manera simultánea en un mismo espacio de búsqueda [2, 60]. Este tipo de aproximaciones hace necesario que los sistemas mantengan diferentes estructuras de datos capaces de soportar distintos tipos de operaciones, lo que puede originar problemas de consistencia.

Las ventajas de utilizar retículos de conceptos sobre estrategias de búsqueda híbridas han sido descritas en un gran número de artículos [58, 15, 48, 30, 48], pudiendo resumirse en una mayor flexibilidad, una efectividad superior en tareas de recuperación y en la posibilidad de realizar tareas de minería de datos. Uno de los dominios de aplicación más interesantes es la administración de sistemas de ficheros, donde Ferre y Ridoux [48] describen un modelo de sistemas de ficheros conceptual donde los conceptos representan tanto directorios de ficheros como consultas que expresan propiedades de los ficheros.

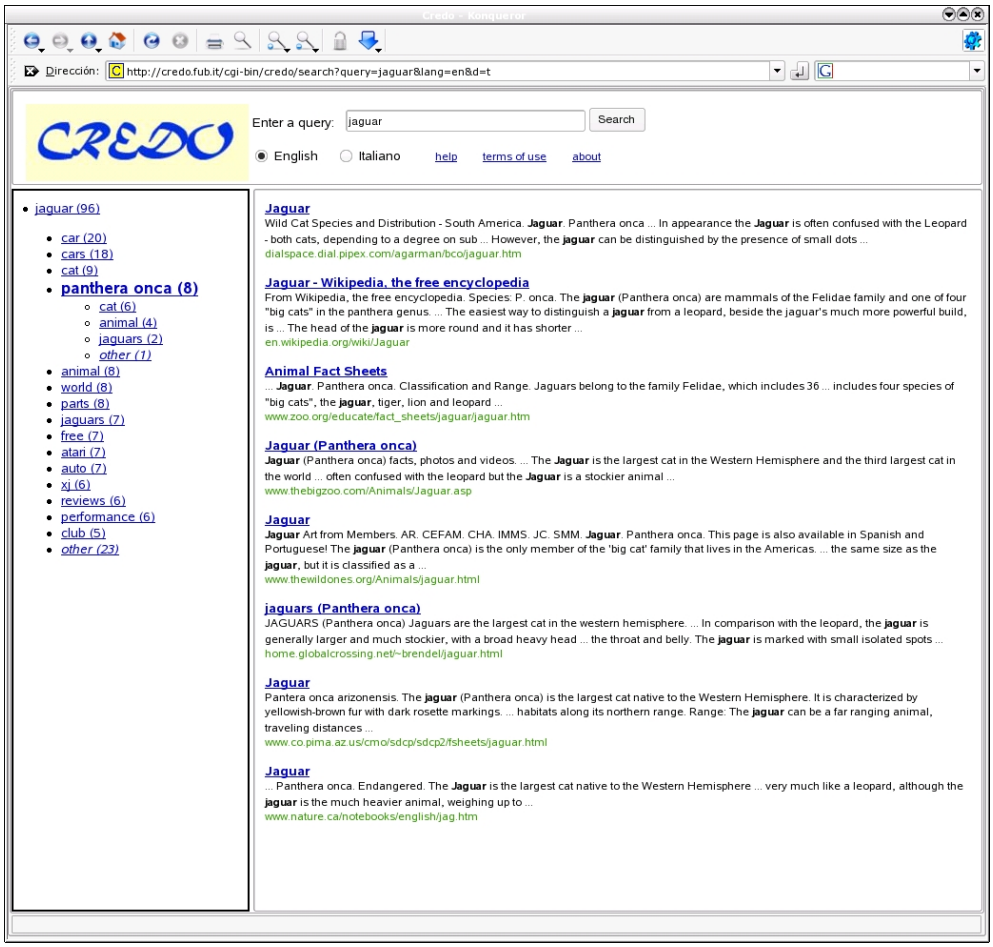


Figura 4.1: Sistema Credo. Resultados proporcionados para la consulta *jaguar*

Cuando la información puede ser clasificada con respecto a diferentes ejes, puede resultar conveniente para el usuario añadir nuevos atributos de manera incremental, tomando decisiones acerca de como es representada la información por el sistema de acuerdo a la selección actual de atributos. Esta aproximación puede implementarse de manera efectiva utilizando las técnicas de anidamiento y zoom descritas en la sección 3.2, pudiendo ser combinadas con otro tipo de estrategias de recuperación.

La utilización de una combinación de vistas resulta ser más apropiado para realizar minería sobre la información contenida en dominios semiestructurados. Una de las más interesantes aplicaciones desarrolladas es un sistema para realizar búsquedas sobre colecciones de mails [24, 30] que permite al usuario combinar de manera efectiva la información contenida en los diferentes campos de los mensajes. El análisis de información extraída de web en [28] se encuentra en esta misma línea de investigación.

4.6. Discusión

En este capítulo hemos presentado las principales aportaciones del área del AFC al desarrollo de sistemas de RI, siendo destacables las aproximaciones para la modificación de consultas y para la ordenación automática de documentos. Así mismo, se han presentado las principales propuestas para la evaluación de este tipo de sistemas, así como las estrategias para llevar a cabo su visualización. Los principales problemas sobre los que esta Tesis Doctoral pretende realizar propuestas innovadoras son:

- *Construcción de los retículos a partir del corpus completo de documentos.* La mayor parte de las aplicaciones del AFC a una tarea de RI requieren de la construcción de un retículo genérico sobre el corpus completo sobre el cual se lleva a cabo el proceso. Esto realizar su construcción a priori con el fin de poder mapear de forma adecuada tanto las consultas como los documentos obtenidos a partir de un proceso de recuperación. En nuestra propuesta, sin embargo, no planteamos trabajar con una estructura de clustering previamente calculada, sino que optamos por calcular el retículo correspondiente a un subconjunto de documentos obtenidos a partir de un proceso previo de recuperación. De esta manera, se obtienen estructuras mucho más adaptadas a la necesidad de búsqueda del usuario y menos complejas. Así mismo, no consideramos la aplicación directa de la teoría, sino que propondremos un modelo que la modifica y adapta con el fin de proporcionar al usuario información mucho más intuitiva, haciendo abordables las tareas de visualización y evaluación.

- *Utilización de tesauros o palabras clave asignadas manualmente*. Una gran parte de las aplicaciones del AFC a una tarea de RI se caracterizan por describir el contexto formal a partir de información obtenida manual o semiautomáticamente o mediante el uso de información externa (por ejemplo tesauros). La propuesta que realizamos en este trabajo, sin embargo, se orienta a construir la estructura de clustering a partir de descriptores que serán obtenidos de manera automática a partir del conjunto de documentos recuperado. Nuestra propuesta pretende ser generalizable a cualquier tipo de escenario de recuperación, sin que sea necesario el

acceso a fuentes de conocimiento externo o al uso de estrategias manuales o semiautomáticas para llevar a cabo la caracterización de los documentos a organizar.

- *Evaluaciones con usuario reales*. La totalidad de las aplicaciones del AFC a tareas de RI estudiadas se caracterizan por no disponer de una metodología de evaluación automática, siendo las aproximaciones con usuarios reales las más habituales para todos los sistemas propuestos. Debemos destacar que los aspectos relacionados con la evaluación de sistemas basados en AFC no han sido investigados en profundidad, siendo esta una carencia importante dentro del área. Esto es debido principalmente a la complejidad que los retículos presentan a la hora de definir medidas adecuadas que permitan cuantificar su adecuación a una tarea concreta. En el capítulo 7 presentaremos un conjunto de medidas que constituyen una aportación novedosa y única al área, pudiendo ser consideradas como la primera aportación propuesta para la evaluación automática de un retículo basado en AFC orientada a tareas de RI.

- *Diferentes aproximaciones para llevar a cabo la visualización e interacción con el retículo*. El uso de AFC para organizar información facilita la integración de diferentes estrategias de navegación e interacción en un mismo espacio de búsqueda. En este sentido, nuestra propuesta pretende experimentar con una aproximación híbrida proponiendo diversas estrategias que podrán ser utilizadas simultáneamente para la navegación y exploración de las estructuras generadas. Además, y en contraste con la aproximación más próxima a nuestra propuesta (el sistema Credo), presentaremos una estrategia de visualización y navegación basada en retículos (frente a la navegación basada en árboles de carpetas implementada en Credo) cuyo objetivo final será el de, sin reducir la capacidad informativa de los retículos, facilitar la navegación por este tipo de estructuras reduciendo la cantidad de información directamente accesible para el usuario.

Parte II

PROPUESTA

Capítulo 5

Modelo de Clustering Basado en AFC

Tal y como expusimos al presentar los objetivos de esta Tesis Doctoral, nuestro trabajo se enmarca dentro de la utilización de técnicas de clustering de documentos para mejorar el acceso a un conjunto de documentos obtenidos previamente mediante un proceso de búsqueda. En este capítulo presentamos el modelo basado en la teoría del AFC que utilizaremos para generar estructuras de clustering de calidad.

Comenzaremos presentando un conjunto de consideraciones fundamentales sobre las características deseables que, bajo nuestro punto de vista, debe presentar un clustering de documentos para resultar adecuado a una tarea de recuperación de información. A continuación presentaremos las restricciones fundamentales sobre las cuales se apoya el modelo propuesto y que resultan críticas tanto para su definición como para desarrollar las medidas de evaluación que serán presentadas en el capítulo 7. Finalmente, presentaremos el modelo formal basado en AFC propuesto en esta Tesis Doctoral.

5.1. Consideraciones fundamentales

Independientemente de los descriptores utilizados para caracterizar o describir los clusters generados, en la aplicación de un proceso de clustering a tareas de recuperación de información podemos dividir el espacio de información en dos clases bien diferenciadas: a) documentos relevantes a las necesidades de búsqueda del usuario, y; b) documentos no relevantes a las necesidades de búsqueda del usuario[1]. El primer objetivo de la tarea en este tipo de sistemas será, por lo tanto, el de facilitar el acceso a la mayor cantidad de información relevante minimizando el número de documentos no relevantes inspeccionados.

Con el objetivo de justificar las decisiones tomadas a la hora de presentar nuestro modelo, así como las diferentes aproximaciones necesarias para materializarlo en sistemas reales y plenamente funcionales, consideramos imprescindible dar una visión general que nos aproxime a las características

[1]Las figuras utilizadas en este trabajo para representar clusters de documentos utilizarán la siguiente nomenclatura: a) el símbolo (+) para representar que un documento contenido en un cluster es relevante a las necesidades del usuario, y; b) el símbolo (-) en el caso en el que dicho documento sea irrelevante.

Figura 5.1: Capacidad de un clustering para aislar y agrupar correctamente la información relevante

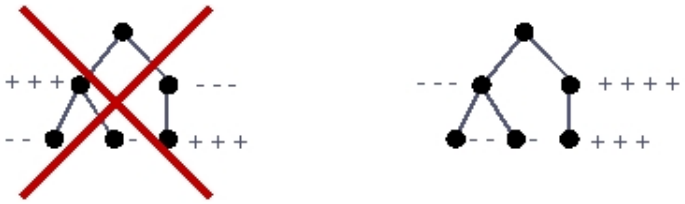


Figura 5.2: Capacidad de un clustering para compactar y relacionar clusters con información relevante

deseables de un clustering orientado a resolver la tarea propuesta:

1. *El contenido de los clusters*. Resulta obvio que un proceso de clustering que genere un conjunto de clusters donde la información relevante esté claramente diferenciada de la información no relevante es mucho más deseable que un proceso de clustering donde la información interesante para el usuario se encuentre arbitrariamente mezclada con documentos no relevantes. Por esta razón, cualquier aproximación que pretenda aplicar técnicas de agrupamiento para la mejora de tareas de recuperación de información debe tener como principal objetivo generar una estructura de clustering capaz de aislar lo máximo posible las dos clases descritas. La Figura 5.1 ilustra esta idea sobre un clustering jerárquico presentando dos agrupaciones posibles. Como puede observarse, la primera de ellas se caracteriza por mezclar en cada cluster información considerada relevante con información irrelevante, mientras que la segunda materializa correctamente la separación entre ambas clases. Obviamente, en este ejemplo la situación deseable de acuerdo a la tarea propuesta sería la segunda de las opciones.

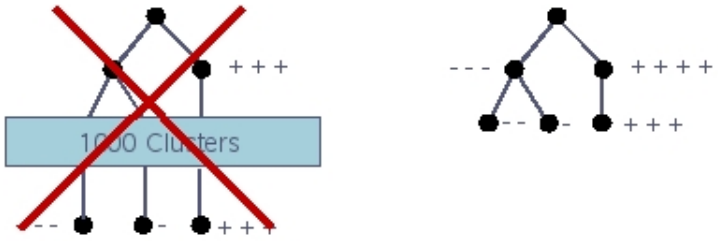


Figura 5.3: Capacidad de un clustering para generar un número razonable de clusters

2. *La estructura de clustering generada*. Es deseable que cualquier proceso de clustering aplicado a una tarea de recuperación de información genere una estructura donde los clusters con información relevante se encuentren directamente interconectados entre si. Una estructura donde la información relevante se encuentre perfectamente agrupada pero que requiera atravesar un gran número de clusters con información no relevante no es deseable desde el punto de vista de la tarea propuesta. La figura 5.2 muestra esta situación, el primer clustering muestra como la información relevante se encuentra perfectamente agrupada en clusters pero, sin embargo, estos no están interconectados, siendo necesario atravesar varios clusters con información no relevante para poder acceder a la información que busca el usuario. En contraste, el segundo clustering de la figura presenta la misma información relevante, agrupada en el mismo número de clusters, pero interconectada de una forma mucho más óptima.

3. *El número de clusters*. Resulta claro que un proceso de clustering que aísle e interconecte de manera correcta clusters con información relevante es la situación ideal. Sin embargo, consideramos que el número de clusters generados en este proceso también es un factor decisivo a tener en cuenta. Un clustering que organice correctamente la información pero que genere un gran número de clusters no es deseable. Esto obligaría a inspeccionar, un número de nodos muy elevado, lo que ralentiza el proceso de exploración. La figura 5.3 muestra esta situación, el primer clustering permite acceder a toda la información relevante de manera directa pero atravesando un gran número de clusters, lo que supone una inversión en tiempo de exploración por parte del usuario que en absoluto justifica el proceso de clustering realizado. En contraste, la segunda ilustración es mucho más adecuada debido a que permite al usuario acceder a la información relevante atravesando un número mucho menor de clusters.

4. *La descripción aplicada a cada uno de los clusters*. Finalmente, debemos destacar que un buen proceso de clustering no sólo se caracteriza por realizar una separación equilibrada de la información relevante y la no relevante, sino que resulta imprescindible asociar a cada uno de los clusters una descripción adecuada con el fin último de que el usuario sea capaz de predecir con exactitud la relevancia de la información que éstos contienen. Un proceso de clustering donde las descripciones aplicadas a los clusters no sean suficientemente informativas podría dirigir al usuario hacia zonas del clustering no relacionadas con el tipo de información que éste está buscando y, como consecuencia, no resultar útil para mejorar la eficiencia de la tarea de recuperación.

5.2. Restricciones asumidas por el modelo

El modelo de clustering que presentaremos en la siguiente sección, toma como punto de partida dos restricciones fundamentales que afectan a las decisiones tomadas en este trabajo respecto a la formalización, visualización y evaluación de nuestra propuesta. La consideración de ambas restricciones para la formalización del modelo suponen un avance respecto a otras propuestas de clustering en escenarios similares y merecen especial atención para la posterior comprensión del mismo.

En concreto, el modelo de clustering que será propuesto en la siguiente sección se basa en dos

	d_1	d_2	d_3	d_4
Física	×	×	×	×
Física Nuclear		×	×	
Astrofísica				×

Cuadro 5.1: Descriptores asociados a un conjunto de documentos d_1, d_2, d_3, d_4

Figura 5.4: Restricciones de clustering a un 'universo cerrado' y a un 'universo abierto' respectivamente considerando la información del cuadro 5.1

restricciones que denominaremos *Restricción del clustering a un 'Universo Abierto'* y la *Restricción del clustering a considerar herencia múltiple* que se describen a continuación.

5.2.1. Restricción del clustering a un 'Universo Abierto'

Esta restricción presupone que, dada una agrupación de documentos, cada uno de sus clusters contiene únicamente aquellos documentos descritos de forma completa por los descriptores del mismo. Esto significa que aquellos documentos que puedan ser especializados en cualquier cluster descendiente de uno concreto deberán ser considerados, tanto a efectos de evaluación como de visualización y navegación, componentes del cluster más específico que los contenga. La idea de esta restricción está basada en el paradigma de navegación utilizado habitualmente en directorios web tales como *Open Directory Project (ODP)* o *Yahoo! Directory* [94, 140]. En este tipo de directorios un documento concreto (página web) nunca aparece referenciado en distintos niveles de la jerarquía de categorías generada, sino que figura en un único cluster considerado como el que mejor describe las características principales del documento.

El cuadro 5.1 muestra un ejemplo donde se agrupa un conjunto de documentos representados por los descriptores *'Física'*, *'Física Nuclear'* y *'Astrofísica'*. La figura 5.4 muestra dos posibles construcciones del clustering asociado al conjunto de documentos propuesto. La primera de ellas no tiene en cuenta la restricción propuesta, situando los documentos más especializados (d_2, d_3, d_4) en el cluster caracterizado por el descriptor genérico *'Física'* y también en sus correspondientes clusters específicos descritos por *'Física Nuclear'* y *'Astrofísica'*. De acuerdo a esta organización, un usuario que comenzara su exploración en el cluster descrito por *'Física'* tendría un acceso directo al conjunto completo de los documentos recuperados, no existiendo diferencia alguna entre la exploración del clustering generado y la inspección completa de la lista ordenada original. En contraste, el segundo clustering presentado en la figura 5.4 sí respeta la restricción propuesta y, por lo tanto, sitúa los documentos más especializados (d_2, d_3, d_4) en aquellos clusters de la agrupación

	d_1	d_2	d_3	d_4	d_5	d_6
Física	×	×	×	×		×
Física Nuclear		×	×			
Astrofísica				×		
Chistes					×	×
Chistes de Física						×

Cuadro 5.2: Descriptores asociados a un conjunto de documentos $d_1, d_2, d_3, d_4, d_5, d_6$

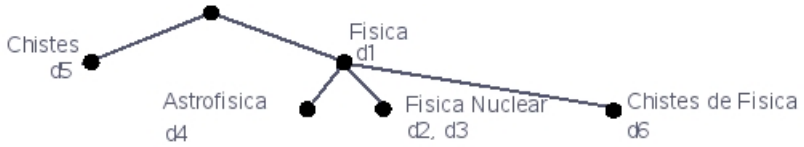


Figura 5.5: Aproximación clustering basada en la restricción de herencia simple

que describen de forma más completa el contenido de dichos documentos (los clusters *'Física Nuclear'* y *'Astrofísica'*). En este escenario, un usuario que comenzara su exploración en el cluster raíz, únicamente tendría acceso al documento d_1, pudiendo utilizar los descriptores *'Física Nuclear'* y *'Astrofísica'* (pertenecientes a los clusters descendientes) para especializar su búsqueda en el caso de que así fuera necesario. Esta segunda aproximación no sólo reduce drásticamente el número de documentos a inspeccionar en los nodos genéricos del clustering, sino que también mejora la comprensión que el usuario tiene de la agrupación obtenida debido a que el contenido de los clusters se encuentra relacionado de forma mucho más precisa con el conjunto de atributos utilizado para describirlos. Debemos destacar que los sistemas de clustering jerárquico disponibles en la web no adoptan esta convención.

5.2.2. Restricción del clustering a considerar herencia múltiple

Si un documento trata acerca de diferentes temas debería ser posible acceder a él desde diferentes partes de la agrupación generada, aunque éstas difieran mucho en su temática. Por ejemplo, una *presentación de proyecto* debería encontrarse bajo la categoría *proyectos* y, también, bajo la categoría *presentaciones*. Esta filosofía a la hora de concebir la colocación de los documentos dentro del

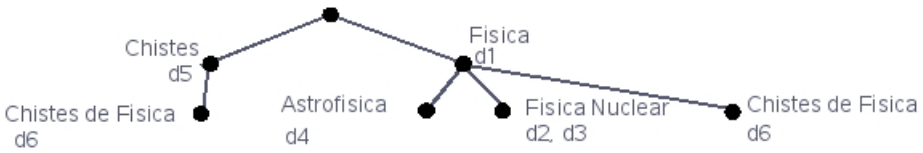


Figura 5.6: Aproximación de clustering basada en la restricción de herencia múltiple considerando una jerarquía como estructura subyacente

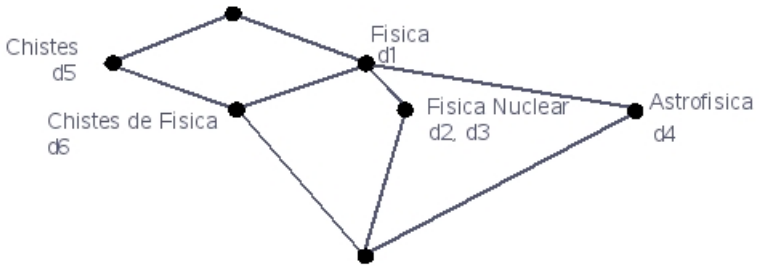


Figura 5.7: Aproximación de clustering basada en la restricción de herencia múltiple considerando un retículo como estructura subyacente

clustering es fundamental para proporcionar al usuario una visión clara de aquellos documentos que comparten características comunes y que permiten relacionar zonas del clustering aparentemente disjuntas.

Las aproximaciones de clustering jerárquico tradicionales, suelen considerar herencia simple para cada uno de los clusters generados. Esto significa que cuando un documento presenta las características descritas (puede aparecer indistintamente en más de un cluster dentro de la agrupación) se suele optar por asignarlo a un único cluster (aquel que mejor describa la temática general del documento a agrupar). Desde nuestro punto de vista, esta aproximación no se adapta al esquema mental que el usuario aplica para interiorizar y relacionar los conceptos relativos al conjunto de documentos a organizar, presenta una visión de la información excesivamente compartimentada y, en algunas ocasiones, incompleta. Además, una vez que se ha tomado una decisión de exploración el usuario únicamente puede especializarla o generalizarla sin posibilidad de llevar a cabo una navegación horizontal que le permitan pasar directamente a zonas del clustering que versen sobre una temática distinta pero que se encuentren relacionadas de alguna manera con el cluster que actualmente se está explorando.

Una alternativa a esta forma de concebir el clustering pasa por adoptar una visión más abierta (considerando herencia múltiple) respecto a la asignación de los documentos sobre el conjunto de clusters generado. De esta manera, un mismo documento podrá aparecer en diferentes partes del clustering si así fuera necesario, permitiendo al usuario acceder a él a través de caminos de exploración diferentes. Algunos sistemas de clustering jerárquico más modernos aplican este segundo planteamiento aunque, en estos casos, el problema de la estricta compartimentación de la información continua presente. Esto es debido a que el tipo de estructura sobre la cual apoyan la agrupación generada es una jerarquía. En este tipo de estructuras, la única manera de permitir el acceso a un documento concreto desde zonas disjuntas del clustering es repetirlo en varios clusters, lo que no permite que el usuario pueda realizar exploraciones horizontales que favorezca la conexión de las partes disjuntas de forma sencilla e intuitiva. Además, debemos destacar que el hecho de repetir documentos en diferentes partes del clustering dificulta enormemente el diseño de métricas capaces de evaluar de manera eficiente sistemas basados en este tipo de aproximaciones para el tipo de tarea propuesta.

El trabajo presentado en esta Tesis Doctoral basa el modelo de clustering propuesto en la teoría del AFC, que resuelve de manera elegante la problemática planteada y respeta de manera directa

la restricción de herencia múltiple propuesta. Esto es debido al uso de retículos, en lugar de estructuras jerárquicas, para llevar a cabo el proceso de organización de la información. La principal característica de estas estructuras matemáticas es la de permitir que cada uno de sus nodos[2] puedan tener más de un antecesor, lo que permite la existencia de clusters con documentos que compartan características heredadas de zonas disjuntas del clustering.

El cuadro 5.2 muestra un conjunto de documentos representados por los descriptores *'Física'*, *'Física Nuclear'*, *'Astrofísica'*, *'Chistes'* y *'Chistes de Física'*. Como puede observarse en el ejemplo, resulta claro diferenciar dos dominios aparentemente disjuntos en la información recuperada. En concreto, podemos identificar un dominio general acerca de temas relacionados con la *'Física'*, y al que pertenecerían los documentos d_1, d_2, d_3 y d_4, y otro dominio, también genérico, que contendría información acerca de documentos relacionados con *'Chistes'*, donde estaría ubicado el documento d_5. Sin embargo, observando con atención el cuadro 5.2 se puede ver como el documento d_6 trata temas relacionados con la *'Física'* y, además, contiene chistes relacionados con esta materia. De acuerdo a los planteamientos expuestos en este punto, un modelo de clustering podría optar por las siguientes aproximaciones: a) una aproximación basada en la restricción de herencia simple, donde habría que decidir en qué cluster situar el documento d_6; b) una aproximación basada en la restricción de herencia múltiple, pero apoyada en una estructura jerárquica subyacente, donde el documento d_6 se asignaría a dos clusters diferentes accesibles desde los clusters genéricos descritos por *'Física'* y *'Chistes'* respectivamente, y; c) una aproximación basada en la restricción de herencia múltiple, pero apoyada en un retículo como estructura subyacente, que generaría un único cluster accesible simultáneamente desde los clusters genéricos descritos por *'Física'* y *'Chistes'*.

La primera de las aproximaciones se presenta en la figura 5.5. Al basar el clustering en una restricción de herencia simple se ha optado por asignar el documento d_6 a un cluster descrito por *'Chistes de Física'* y descendiente del cluster genérico *'Física'* (hubiera sido igualmente válida la asignación de dicho cluster como descendiente del cluster genérico descrito por *'Chistes'*). Un usuario que tuviera que navegar por la información utilizando esta aproximación accedería al documento d_6 únicamente en el caso de haber dirigido su exploración hacia la categoría *'Física'*, no siendo posible para éste el acceder de forma directa a documentos relacionados con *'Chistes'* en general desde el cluster donde se encuentra. De la misma manera, el acceso del usuario a la zona del clustering relacionada con *'Chistes'* no le permitiría descubrir que el sistema ha recuperado un documento que trata de un tipo específico de chistes relacionados con el mundo de la física y que, por lo tanto, también podría ser de su interés.

La aproximación presentada en la figura 5.6 se corresponde con la segunda propuesta de clustering. En este caso, el hecho de basar el clustering en una restricción de herencia múltiple pero utilizando una estructura jerárquica implica repetir el documento d_6 en dos zonas diferentes del clustering. Para ello se crean dos clusters independientes, que en este caso están etiquetados con el mismo descriptor (*'Chistes de Física'*), que son colocados como descendientes de los clusters genéricos *'Física'* y *'Chistes'*. De acuerdo a esta construcción el usuario podría acceder sin problemas al documento d_6, independientemente del camino que inicialmente eligiera para explorar la información. Sin embargo, es interesante destacar que en el caso en el que el usuario se encontrara inspeccio-

[2] también denominados conceptos formales o clusters en nuestro trabajo

nando el documento d_6 en el cluster que especializa la categoría de *Chistes* y deseara continuar su inspección por más documentos relacionados con la física, no podría atravesar la barrera que separa ambas zonas disjuntas, siendo necesario acceder a clusters superiores (en este caso el nodo raíz del clustering) para poder redirigir su exploración hacia esa zona.

Finalmente, la figura 5.7 muestra la última de las aproximaciones, en la cual se basa nuestro modelo. Utilizar un retículo como estructura subyacente para organizar la información supone definir un único cluster, etiquetado por el descriptor *'Chistes de Física'*, conectado directamente con los clusters más generales *'Física'* y *'Chistes'* y que será el que contenga el documento d_6. Con esta organización, un usuario que comenzara su navegación en el cluster raíz podría acceder al cluster *Chistes de Física* independientemente de las decisiones de navegación inicialmente tomadas. Además, el hecho de utilizar un retículo permitiría a éste acceder intuitivamente a cualquiera de las zonas disjuntas del retículo desde dicho cluster, es decir, a clusters más generales acerca de *'Chistes'* o de *'Física'* sin necesidad de deshacer parte del camino ya realizado. Como puede observarse en este último ejemplo, una aproximación de este tipo, no sólo facilita la tarea de exploración al usuario sino que, también, le permite obtener directamente, conforme se realiza el proceso de navegación, una idea aproximada de la estructura conceptual de la información recuperada.

5.3. Propuesta del modelo

El objetivo de esta sección es presentar el modelo formal aplicado en esta Tesis Doctoral para realizar el clustering sobre la información recuperada por un motor de búsqueda tradicional. La base de este modelo es la teoría del AFC, descrita en la sección 3. Tal y como fue presentado, el AFC es una teoría matemática que facilita la estructuración y clasificación de la información de forma automática. En esta sección describiremos con detalle el modo en el que aplicamos esta técnica para organizar un conjunto de documentos, prestando especial atención a algunos aspectos de la teoría que resultan fundamentales para hacer cumplir las hipótesis propuestas anteriormente.

Partiremos del supuesto inicial según el cual damos por hecho que un motor de búsqueda ha recuperado un conjunto $Docs = \{doc_1, doc_2, \ldots, doc_n\}$ de n documentos a partir de una consulta q realizada por el usuario. También supondremos que hemos extraído (manual o automáticamente) un conjunto $Desc = \{desc_1, desc_2, \ldots, desc_m\}$ de m descriptores o atributos para describir el contenido del conjunto $Docs$ (las técnicas para la extracción, selección y asignación de dichos descriptores serán presentadas en detalle en el capítulo 6).

Según la teoría del AFC es posible definir un contexto formal K a partir de ambos conjuntos como una terna $K \equiv (Docs, Desc, I)$ donde $Docs$ representa el conjunto de documentos recuperado, $Descs$ es el conjunto de descriptores obtenido a partir del conjunto de documentos $Docs$ e I es una relación binaria $I \subseteq Docs \times Decs$ tal que $(doc_i, desc_j) \in I$ significa que el documento doc_i está descrito por $desc_j$.

Tomando como base el contexto formal K, definimos el conjunto $\beta(K) \equiv \beta(Docs, Desc, I)$, como el conjunto de todos los conceptos formales correspondientes a dicho contexto. Los elementos pertenecientes a $\beta(K) = \{c_1, c_2, \ldots, c_k\}$ se caracterizan por las siguientes propiedades:

1. Cualquier concepto formal c se define por un par (A, B) donde A y B reciben el nombre de extensión e intensión respectivamente y se definen como $A \subseteq Docs$ y $B \subseteq Desc$.
2. De todos los pares (A, B), únicamente serán considerados conceptos formales aquellos que cumplan $(A' = B) \wedge (B' = A)$.

Tal y como fue expuesto en el capítulo 3, es posible definir una relación binaria de orden sobre el conjunto $\beta(K)$, representada como $\underline{\beta}(\beta(K), \leq)$, de modo que siendo $c_1 = (A_1, B_1)$ y $c_2 = (A_2, B_2)$ dos conceptos formales pertenecientes a $\beta(K)$, se dice que $c_1 \leq c_2$, es decir, c_1 es un *subconcepto* de c_2, si y solo si $A_1 \subseteq A_2$ (o de igual modo si $B_1 \supseteq B_2$).

La relación binaria propuesta cumple las propiedades reflexiva, transitiva y antisimétrica y, por tanto, convierte a $\underline{\beta}(\beta(K), \leq)$ en una relación de orden. Esta relación define una jerarquía conceptual cuya interpretación de acuerdo a la teoría general de retículos viene dada por el *Teorema Fundamental del Análisis Formal de Conceptos* (presentado en el capítulo 3), y que define las operaciones de supremo e ínfimo necesarias para demostrar que $\underline{\beta}(\beta(K), \leq)$ es un retículo

Tal y como exponíamos anteriormente, el hecho de que $\underline{\beta}(\beta(K), \leq)$ sea un retículo significa que el modelo propuesto trabajará sobre una hipótesis de un 'universo abierto', con todas las implicaciones que ello conlleva. En nuestro caso concreto, identificaremos el conjunto de conceptos formales obtenido con el conjunto de clusters propuesto y las relaciones derivadas del orden definido sobre $\beta(K)$ con la organización final del clustering.

Con el fin de respetar la hipótesis de la descripción completa como base del modelo propuesto, no resulta adecuado trabajar con toda la información contenida en la extensión de un concepto formal. Esto es debido a que la extensión de un concepto contiene el conjunto de objetos (los documentos de nuestro modelo) pertenecientes a dicho concepto. De acuerdo a la hipótesis de la descripción completa, es deseable que cada cluster (concepto formal) muestre únicamente aquellos documentos descritos de manera completa por los descriptores asignados a dicho cluster. Por lo tanto, todo documento perteneciente a la extensión de un concepto que pueda ser especializado en cualquiera de sus subconceptos no debería ser considerado como componente de dicho cluster.

Con el fin de materializar esta idea, de entre todos los conceptos pertenecientes a $\beta(K)$, el AFC distingue aquellos que contienen algún objeto de su extensión completamente especializado por todos los descriptores de su intensión. Este tipo especial de conceptos formales reciben el nombre de *conceptos objeto* y se definen de la siguiente manera. Dado un $doc_i \in Docs$ decimos que el concepto formal c_j es su correspondiente concepto objeto si se cumple:

$$\gamma(doc_i) \stackrel{def}{=} \left(\{doc_i\}'', \{doc_i\}'\right) \equiv \left(doc_i'', doc_i'\right) = c_j \tag{5.1}$$

De este modo, para que nuestro modelo respete la hipótesis de la descripción completa únicamente deberían considerarse para cada cluster aquellos documentos para los cuales éste sea su concepto objeto. Con este fin, nuestro modelo hace uso de *funciones de transformación* para crear una correspondencia entre el retículo de conceptos obtenido aplicando AFC sobre la información recuperada y una representación alternativa del mismo que tiene en cuenta las restricciones planteadas. Para ello definiremos el concepto de *nodos de información* (*information nodes*) que formalizamos en la definición 1 y que serán utilizados para representar el contenido de final de los clusters.

Documento	Descriptores
d_1	F
d_2	F, FN
d_3	F, FN
d_4	F, A
d_5	C
d_6	F,C,CF

Cuadro 5.3: Ejemplo de un conjunto de documentos y sus descriptores asociados

Definicion 1 *Dado un concepto formal $c = (A, B)$ definimos su correspondiente nodo de información ni como un par (AI, BI) tal que $AI \subseteq A$ esta formado por aquellos elementos de A para los cuales c es su correspondiente concepto objeto y $BI = B$. Formalmente:*

$$ni \equiv (AI, BI) \xrightarrow{def} \begin{cases} AI \subseteq A, \textit{donde } \forall \alpha \in AI \cdot \gamma(\alpha) = c \\ BI = B, \end{cases} \tag{5.2}$$

La definición 2 introduce las funciones de transformación necesarias obtener un nodo de información a partir de su correspondiente concepto formal.

Definicion 2 *Dado un concepto formal $c = (A, B)$ y su correspondiente nodo de información $ni = (AI, BI)$, se definen las correspondencias $A \xrightarrow{f(x)} AI$ y $B \xrightarrow{g(x)} BI$ operadas por las funciones $f(x)$ y $g(x)$ sobre cada uno de los elementos de A y B respectivamente del siguiente modo:*

$$f(x) = \begin{cases} x, \textit{si } \gamma(x) = c \\ \emptyset, \textit{si } \gamma(x) \neq c, \end{cases} \tag{5.3}$$

$$g(x) = x \tag{5.4}$$

La idea que hay detrás de un nodo de información es la de mostrar al usuario en cada cluster únicamente aquellos documentos descritos en su totalidad por la intensión de su correspondiente concepto formal. Las funciones de transformación, por lo tanto, tienen como misión la de convertir los elementos de la extensión y la intensión de cada concepto formal en sus correspondientes conjuntos AI y BI.

El siguiente ejemplo pretende ilustrar paso a paso la aplicación del modelo propuesto. Partamos de la base de que un motor de búsqueda ha recuperado un conjunto de documentos $Docs = \{d_1, d_2, d_3, d_4, d_5, d_6\}$ sobre el cual se han extraído (manual o automáticamente) un conjunto de descriptores $Desc = \{$'Física', 'Física Nuclear', 'Astrofísica', 'Chistes', 'Chistes de Física'$\}$. Su-

K	**F**	**FN**	**A**	**C**	**CF**
d_1	×				
d_2	×	×			
d_3	×	×			
d_4	×		×		
d_5				×	
d_6	×			×	×

Cuadro 5.4: Contexto formal K correspondiente al conjunto de documentos presentado en el cuadro 5.3

$\beta(K)$	**Extensión**	**Intensión**
c_1	$\{d_1, d_2, d_3, d_4, d_5, d_6\}$	$\{\emptyset\}$
c_2	$\{d_5, d_6\}$	$\{C\}$
c_3	$\{d_6\}$	$\{C, CF\}$
c_4	$\{d_1, d_2, d_3, d_4, d_6\}$	$\{F\}$
c_5	$\{d_2, d_3\}$	$\{F, FN\}$
c_6	$\{d_4\}$	$\{F, A\}$
c_7	$\{\emptyset\}$	$\{F, FN, A, C, CH\}$

Cuadro 5.5: Conjunto de conceptos formales obtenidos del contexto K presentado en el cuadro 5.4

pongamos que, además, cada documento viene descrito por un subconjunto de descriptores, tal y como se muestra en el cuadro 5.3[3].

De acuerdo a nuestro modelo, el contexto formal K asociado a los conjuntos propuestos estaría representado en el cuadro 5.4 y el conjunto de clusters generado, $\beta(K)$, se correspondería con el cuadro 5.5. La relación de orden asociada a $\beta(K)$ se representa en la figura 5.8 mediante un diagrama de Hasse.

Como puede observarse en el cuadro 5.5 y en la figura 5.8, el conjunto de clusters generado no respeta la hipótesis de la descripción completa debido a que conceptos como c_2 o c_4 incluyen en su extensión documentos que pueden se especializados en subconceptos derivados de éstos ($\{d_6\}$ y $\{d_2, d_3, d_4, d_6\}$ respectivamente). Es por esta razón por la cual en el cuadro 5.6 mostramos la correspondencia entre cada uno de los conceptos formales y sus respectivos nodos de información de acuerdo a las definiciones 1 y 2. Nótese como, en este caso, los nodos de información n_2 y n_3 han reducido los elementos pertenecientes a su extensión para mostrar únicamente aquellos documentos completamente descritos por los descriptores asociados a cada uno de los clusters y, por lo tanto, se ajusta a la hipótesis de la descripción completa propuesta.

[3]Con el fin de poder presentar la información tanto del contexto formal como del conjunto de conceptos obtenido utilizaremos las siguientes abreviaturas para el conjunto de descriptores. F=Física, FN=Física Nuclear, A=Astrofísica, C=Chistes y CF=Chistes de Física

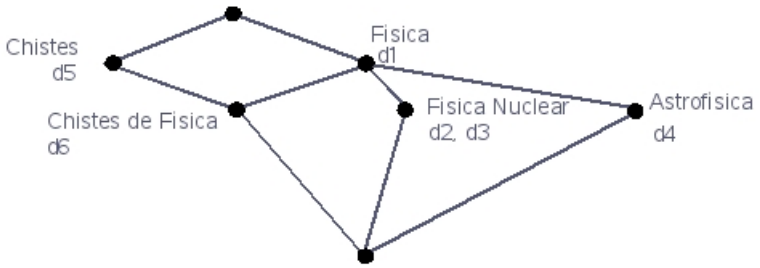


Figura 5.8: Retículo de conceptos correspondiente al conjunto $\beta(K)$ del cuadro 5.5

Concepto Formal	Nodo de Información
$c_1 = (\{d_1, d_2, d_3, d_4, d_5, d_6\}, \{\emptyset\})$	$ni_1 = (\{\emptyset\}, \{\emptyset\})$
$c_2 = (\{d_5, d_6\}, \{C\})$	$ni_2 = (\{d_5\}, \{C\})$
$c_3 = (\{d_6\}, \{C, CF\})$	$ni_3 = (\{d_6\}, \{C, CF\})$
$c_4 = (\{d_1, d_2, d_3, d_4, d_6\}, \{F\})$	$ni_4 = (\{d_1\}, \{F\})$
$c_5 = (\{d_2, d_3\}, \{F, FN\})$	$ni_5 = (\{d_2, d_3\}, \{F, FN\})$
$c_6 = (\{d_4\}, \{F, A\})$	$ni_6 = (\{d_4\}, \{F, A\})$
$c_7 = (\{\emptyset\}, \{F, FN, A, C, CH\})$	$ni_7 = (\{\emptyset\}, \{F, FN, A, C, CH\})$

Cuadro 5.6: Conjunto de nodos de información correspondientes al conjunto de conceptos formales $\beta(K)$ del cuadro 5.5

5.4. Recapitulación

En este capítulo hemos presentado el modelo formal basado en la teoría del AFC que utilizaremos en esta Tesis Doctoral para la obtención de un clustering de documentos. Está basado en dos restricciones principales que consideramos innovadoras con respecto al planteamiento del clustering realizado por otras aproximaciones. En concreto, la consideración de un 'universo abierto' y de herencia múltiple sobre el conjunto de clusters generados permite obtener una vista de la información agrupada mucho más informativa y completa que la proporcionada por las aproximaciones clásicas de clustering jerárquico.

Dado que el planteamiento de la teoría del AFC no se adecua de manera directa a nuestras propuestas, el modelo se basa en la definición de *nodo de información* (presentado en esta Tesis Doctoral) que permite adaptar los fundamentos de matemáticos del AFC al conjunto de restricciones que hemos impuesto.

Debemos destacar como la aplicación del modelo hace necesario partir de un contexto formal previamente construido con el fin de llevar a cabo la construcción automática del clustering. La construcción de este contexto no resulta trivial dado que una elección incorrecta de los descriptores utilizados para representar los documentos que se desean agrupar puede conducir a la obtención de estructuras de clustering excesivamente complejas y poco usables para el usuario. En el capítulo 6 presentaremos una metodología orientada a la construcción de este contexto que describirá, paso a

paso, el conjunto de procesos involucrados en su obtención, así como el conjunto de técnicas asociadas a cada uno de ellos orientadas a la generación de estructuras de clustering adecuadas de cara al usuario y a la tarea de recuperación de información.

Capítulo 6

Aplicación del Modelo a la Organización de Resultados de Búsqueda

En el capítulo anterior hemos presentado el modelo desarrollado en este trabajo para realizar un clustering de documentos. Tal y como hemos expuesto, la aplicación del modelo propuesto requiere partir de un contexto formal donde el conjunto de documentos recuperado se encuentre caracterizado por un conjunto de descriptores que deben ser previamente extraídos y seleccionados.

En este capítulo presentamos una metodología orientada a facilitar el diseño completo de un sistema de clustering de documentos basado en el modelo propuesto en este trabajo. Para ello, describiremos el conjunto de procesos involucrados en un sistema de este tipo, presentando diferentes alternativas para la extracción y selección de descriptores, así como para la construcción del contexto formal que finalmente será utilizado como entrada del modelo para construir el clustering de documentos. Finalmente, presentaremos dos paradigmas de visualización basados en los retículos obtenidos cuyo objetivo es el de facilitar al usuario su navegación y exploración.

6.1. Arquitectura propuesta

El objetivo del modelo presentado en el capítulo 5 es el de facilitar la construcción de agrupaciones o clusterings de documentos de forma automática y sencilla. Sin embargo, la obtención de los elementos necesarios para aplicar el modelo (Los documentos, los descriptores y su relación de incidencia) no es una tarea directa. Esto hace necesario realizar con previamente una serie de procesos que permitan obtener de manera automática los elementos que definirán el contexto formal necesario para la generación del clustering. El objetivo de esta sección es el de integrar el modelo de clustering propuesto dentro de una arquitectura que cubra todas las facetas relacionadas con el proceso de clustering. En concreto, en la arquitectura de un sistema de recuperación de información basado en clustering de documentos deberían considerarse los siguientes procesos:

- *Proceso de recuperación de información*. De las diferentes aproximaciones de clustering de documentos para la tarea presentada, en esta Tesis Doctoral hemos optado por aplicar el pro-

ceso de clustering sobre un conjunto de documentos previamente recuperado por un motor de búsqueda tradicional. En este sentido, en nuestro trabajo optamos por una estrategia de análisis local en lugar de una estrategia de análisis global, lo que supone disponer de mecanismos que permitan llevar a cabo el proceso de recuperación inicial en función de las necesidades del usuario. Dado que nuestra propuesta pretende ser generalizable, en nuestro trabajo dejamos abierta la realización de este proceso al propio sistema de clustering o, por el contrario, también permitimos que éste pueda ser realizado por un sistema externo, capaz de proporcionar al sistema de clustering la información suficiente para llevar a cabo el conjunto de procesos que describiremos a continuación. En el caso de optar por realizar el proceso de recuperación dentro del propio sistema de clustering, la sección 6.2 presenta sus características deseables.

- *Proceso de extracción de descriptores*. Una vez obtenido el conjunto $Docs$ de documentos relevantes recuperados, se hace necesario extraer el conjunto completo de descriptores candidatos a caracterizar los documentos que se van a agrupar. En este trabajo se presentan distintas aproximaciones para la extracción de descriptores cuyas principales diferencias radican en el tipo de descriptor extraído. En concreto, las aproximaciones para la extracción de descriptores que presentamos se basan en la extracción de unigramas, sintagmas terminológicos y n-gramas como unidades básicas para la descripción y caracterización de los documentos recuperados. La sección 6.3 presenta en detalle cada una de estas alternativas.

- *Proceso de selección de descriptores*. Una vez extraído el conjunto de descriptores candidatos para describir la información recuperada, es necesario seleccionar, de entre todos ellos, aquellos más adecuados que los describan de una manera completa y significativa. El conjunto de descriptores seleccionado será el que defina el conjunto $Desc$ presentado en el capítulo 5. En esta Tesis Doctoral se presentan diversas técnicas para llevar a cabo este proceso de selección, teniendo en cuenta sobre todo su impacto en el número de clusters generado así como en su potencial para separar de forma correcta la información relevante y no relevante. La sección 6.4 presenta las diferentes alternativas desarrolladas en este trabajo.

- *Aplicación del modelo propuesto*. Conocidos los conjuntos $Docs$ y $Desc$ ya es posible aplicar el modelo de clustering propuesto en el capítulo 5. Para ello debemos realizar dos procesos bien diferenciados:

  - *Proceso de asignación de descriptores*. En primer lugar, y debido a que el contexto formal K no se encuentra todavía definido, es necesario realizar un proceso de asignación de descriptores cuyo objetivo será el de obtener la relación de incidencia I capaz de determinar qué documentos contienen qué descriptores o, en otras palabras, de caracterizar los documentos a partir del conjunto de descriptores seleccionados. Nuestro trabajo presenta dos aproximaciones para obtener dicha relación que se diferencian principalmente en el modo en el que los descriptores son asignados a los documentos. En una primera propuesta se realiza una asignación directa de descriptores, mientras que en la segunda se realiza un enriquecimiento de las relaciones iniciales mediante la aplicación de *Latent Semantic Indexing* (LSI). El fin último de esta segunda aproximación es el de

aumentar el número de documentos afectados por un descriptor concreto con el objetivo de mejorar la calidad de las estructuras de clustering finalmente obtenidas. Ambas aproximaciones serán presentadas y discutidas en la sección 6.5 de esta Tesis Doctoral.

- *Proceso de construcción del retículo*. Finalmente, y una vez definido completamente el contexto K, se procede a la generación automática del conjunto $\beta(K)$ de conceptos formales que, aplicando las transformaciones y definiciones descritas en el modelo, dará lugar al clustering del conjunto de documentos recuperados. La sección 6.6 aborda las técnicas utilizadas para llevar a cabo su construcción.

- *Proceso de Visualización del Clustering*. Finalmente, y una vez obtenida la estructura de clustering, se hace necesario realizar un proceso orientado a visualizar y a facilitar la interacción del usuario con la estructura obtenida. En este sentido, resulta crítico determinar qué paradigmas de visualización son más adecuados para representar los contenidos de unas estructuras tan ricas y, al mismo tiempo, tan desconocidas por los usuarios habituales de este tipo de sistemas. En esta Tesis Doctoral proponemos dos aproximaciones para la visualización y navegación sobre este tipo de estructuras que el usuario podrá utilizar de forma complementaria. La sección 6.7 describe en detalle cada una de ellas.

La figura 6.1 muestra la arquitectura de un sistema genérico de este tipo y el modo en el que estos procesos interactúan entre sí para dar lugar a la estructura de clustering que finalmente es visualizada por el usuario. Inicialmente el usuario realiza una consulta que es procesada por un *Módulo de Recuperación de Información* (MRI) con el fin de extraer un conjunto de documentos relevantes ordenados de acuerdo a un ranking de relevancia. De este conjunto, el MRI selecciona los n documentos más relevantes y se los envía a un *Módulo de Extracción de Descriptores* (MED) cuya finalidad es la de, a partir del conjunto de documentos, extraer el conjunto de descriptores que éstos contienen. Este conjunto de descriptores es procesado por un *Módulo de Selección de Descriptores* (MSD) que evalúa los descriptores extraídos y selecciona aquellos más apropiados para describir adecuadamente los n documentos recuperados. Finalmente, los conjuntos $Docs$ y $Desc$ son enviados al *Módulo de Aplicación del Modelo* (MAM) cuya finalidad es la de obtener el clustering asociado a los n documentos. Para ello primero es necesario generar el contexto formal (mediante el *Módulo de Generación del Contexto* MGC) asociado a la información recuperada y, a continuación, aplicar las técnicas necesarias para la generación del retículo (mediante el *Módulo de Generación del Retículo* (MGR)). En este proceso también se lleva a cabo la generación de los nodos de información con el objeto de facilitar la tarea de visualización y representación que se lleva a cabo en el siguiente módulo. Finalmente, y una vez aplicado el modelo, la información relativa al retículo generado, así como el conjunto de nodos de información asociados son pasados al *Módulo de Visualización* (MV), cuyo objetivo será el de construir una representación adecuada que permita al usuario navegar y explorar el clustering generado.

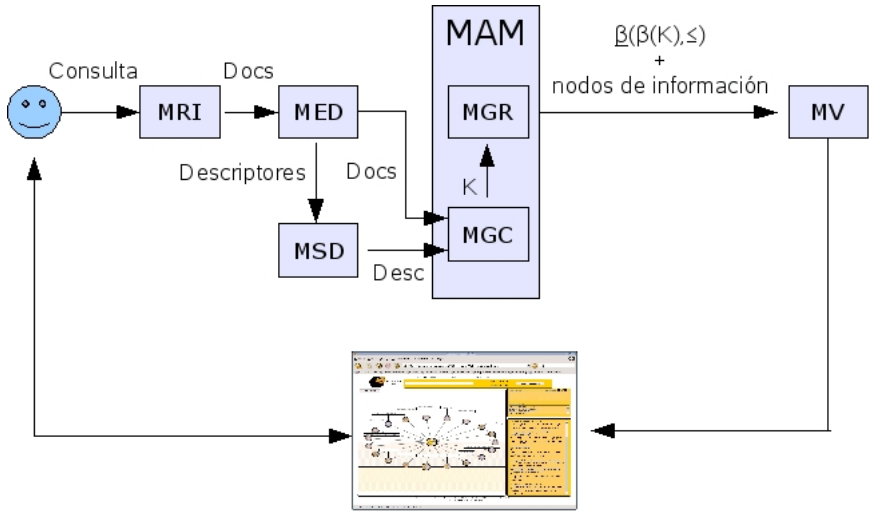


Figura 6.1: Arquitectura general de un sistema de RI basado en el modelo de clustering propuesto

6.2. Proceso de recuperación de información

Tal y como se expuso en el capítulo 1 de esta Tesis Doctoral, un sistema de recuperación de información se caracteriza por facilitar el acceso, sobre un corpus de documentos, a la información relevante a unas necesidades concretas del usuario que éste expresa mediante la formulación de una consulta. Este proceso de acceso a la información puede dividirse en tres subprocesos cuya realización implica tomar decisiones acerca del modo en el que deberán realizarse las consultas, el modelo aplicado para representar e indexar tanto el conjunto de documentos como las consultas realizadas y, finalmente, el procedimiento aplicado para llevar a cabo la recuperación de los documentos relevantes.

6.2.1. Formulación de las consultas

En un sistema de recuperación de información la formulación de las consultas puede llevarse a cabo de muy diversas formas, desde la utilización de lenguajes formales hasta el uso de texto libre en lenguaje natural. En principio, y aunque podría aplicarse cualquiera de estas dos aproximaciones a nuestra propuesta, el procedimiento para la formulación de las consultas propuesto en esta Tesis Doctoral está basado en texto libre. Más concretamente, permitiremos al usuario expresar sus necesidades haciendo uso de un vocabulario no controlado, lo cual permite crear consultas utilizando cualquier término independientemente de su aparición o no dentro del corpus de documentos sobre el cual se lleva a cabo el proceso de recuperación.

6.2.2. Proceso de indexación

Cualquier sistema de recuperación de información debe llevar a cabo un proceso de representación tanto de la información del corpus como de la información de las consultas realizadas por el usuario que permita un procesamiento eficaz a la hora de llevar a cabo el proceso de recuperación [108]. Los dos formatos de representación de la información más extendidos son:

- *Representación basada en facetas o metadatos.* Orientada a representar el contenido de los documentos mediante un conjunto de atributos o facetas predeterminados, a los que se les asigna valores específicos de acuerdo con la información existente en el documento. En esta aproximación, el proceso de análisis de los documentos y la asignación de valores a las facetas se realiza habitualmente de forma manual o semiautomática [23]. La representación basada en facetas tiene una aplicación muy útil en bibliotecas digitales, donde la documentación en lenguaje natural suele acompañarse de una serie de campos (e.g. autor, año de publicación, etc.) que pueden utilizarse como facetas para caracterizar la información.

- *Representación basada en términos o en palabras clave.* Propone la representación del contenido de los documentos mediante una lista de términos o descriptores. Este tipo de representación se presta a que el análisis de los documentos para la obtención de su representación se pueda realizar de manera automática. La indexación de estos términos se puede realizar mediante la utilización de vocabulario controlado (que consiste en definir previamente un conjunto de términos fijo a utilizar en el proceso de indexación) o de vocabulario no controlado (que permite utilizar cualquier término en el proceso de indexación)[108, 23].

El tipo de lenguaje utilizado por el usuario para expresar sus necesidades se encuentra fuertemente relacionado con el método utilizado para representar los documentos. Esto hace que un sistema basado en facetas resulta bastante complejo, e incluso inadecuado, para el procesamiento de las consultas expresadas en lenguaje natural [108, 33]. Por esta razón, además de por su sencillez a la hora de realizar el proceso de manera automática, en esta Tesis Doctoral adaptaremos el uso de una representación, tanto para la consulta como para los documentos del corpus, basada en palabras clave sobre un vocabulario no controlado.

Actualmente, esta aproximación es la más habitual en los sistemas de recuperación de información. Su objetivo es realizar un análisis inicial del corpus de documentos sobre el cual se lleva a cabo el proceso de indexación con el fin de construir un fichero de índices invertido, mediante el cual se relacionan cada uno de los términos con los documentos en los cuales aparece.

Para llevar a cabo el proceso de indexación se ha optado por la utilización de un conjunto de técnicas que son sencillas de implementar, a la vez que efectivas [108, 7]. Los elementos básicos utilizados tanto en el procesamientos de los documentos como en el de las consultas son:

- *Uso de listas de parada (stoplists).* Es deseable que no todos los términos que figuran en un documento sean considerados en el proceso de indexación. Esto es debido a que existe un gran número de términos con contenido semántico nulo y con frecuencias de aparición, tanto en la colección como en cada uno de sus documentos, muy elevadas (de las diez palabras

que aparecen más frecuentemente en Inglés pueden constituir entre un 20 % y un 30 % de los tokens de un documento [49]. Este tipo de términos reciben el nombre de palabras vacías (*stopwords*) por su baja capacidad discriminativa y, con el fin de mejorar la eficiencia de los sistemas de recuperación de información, suelen eliminarse en el proceso de indexación tanto de los documentos como de las consultas. Las listas de parada se obtienen a partir de estudios específicamente orientados a ello y a partir de corpus de texto suficientemente representativos del idioma considerado. Por ejemplo, en [119] puede encontrarse una lista de parada de 250 términos y en [49] una de 425 obtenida a partir del *Brown Corpus*.

- *Eliminación de sufijos (word stemming).* Los algoritmos de extracción de raíces (stemming), o de eliminación de sufijos, se encuentran orientados a obtener un único término a partir de diferentes palabras que constituyen esencialmente variaciones morfológicas con un significado similar [50]. El resultado de un algoritmo de este tipo es la obtención de una misma forma canónica, que no tiene que ser necesariamente la raíz lingüística, para las diferentes variantes morfológicas de una misma palabra. En [50] pueden encontrarse diferentes tipos de algoritmos de extracción de raíces. La elección de un algoritmo concreto para la realización de este proceso depende de la política seguida en la construcción del sistema de recuperación de información, aunque es deseable que éstos minimicen la *sobrerradicación* (la obtención de una misma forma canónica para palabras con significados diferente) y la *infrarradicación* (la obtención de diferentes formas canónicas para palabras que deberían compartir la misma por tener el mismo significado).

- *Pesado de los términos.* El concepto de *poder de resolución* de un término proporciona la base para los métodos de indexación centrados en frecuencias de aparición de términos [119] y hace referencia a la adecuación del término para ser considerado como término de indexación. Su definición se fundamenta en observaciones empíricas relativas a la frecuencia de aparición de las palabras en los textos, siendo la *Ley de Zipf* uno de los resultados de todas estas observaciones. Esta ley establece que, ordenados todos los términos de un texto (o un conjunto de textos) por su frecuencia de aparición, el producto de su frecuencia de uso por su posición dentro de la ordenación ($rango$) es una constante ($frecuencia * rango \simeq cte$). El poder de resolución se define en función de la frecuencia total de aparición de un término i en la colección de n documentos:

$$totfrec_i = \sum_{j=1}^{n} tf_{ij} \tag{6.1}$$

Donde tf_{ij} es la frecuencia de aparición del término i en el documento j. La figura 6.2 representa de forma genérica el poder de resolución para un conjunto de términos. En el eje de abscisas se representan los términos en orden decreciente de frecuencia de aparición dentro de la colección, mientras que el eje de ordenadas representa los valores de $totfrec_i$ obtenidos empíricamente y el poder de resolución estimado. Los términos que se encuentran a la izquierda de C suelen ser artículos, determinantes, etc. Estos términos se caracterizan por una alta frecuencia de aparición dentro de la colección y, por lo tanto, por tener poca capacidad

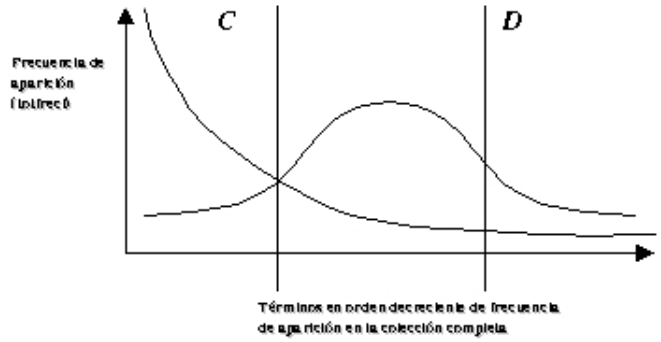


Figura 6.2: Poder de resolución de los términos

de discriminación y ser poco representativos como términos de indexación (lo que explica su consideración como palabras vacías). Por otra parte, los términos que se encuentran a la derecha de D son de aparición tan escasa que su presencia o ausencia no afecta de forma significativa al proceso de recuperación. Los términos que cuentan con mayor poder de resolución se encuentran entre C y D y son los más adecuados para utilizar como términos de indexación. El concepto de poder de resolución se presenta como la base sobre la cual se apoyan los métodos de indexación automática basados en las frecuencias de aparición de términos [119], aunque es necesario destacar las dificultades existentes a la hora de definir los umbrales C y D a partir de los cuales determinar si un término será o no indexado.

Con el fin de mejorar el proceso de indexación, aproximando ésta a las hipótesis propuestas por la Ley de Zipf y el poder de resolución, a lo largo de toda la literatura se han propuesto numerosas fórmulas de pesado de los términos que permiten asociar cada uno de éstos con un valor real que, en función de su frecuencia de aparición, son capaces de mostrar la importancia del término dentro de la colección. De todas ellas, una de las más habituales caracteriza cada uno de los términos en función de las siguientes fórmulas de pesado:

$$w_i = log_2 \left(\frac{n}{df_i} \right) \tag{6.2}$$

$$wd_{ij} = tf_{ij} * w_i \tag{6.3}$$

Siendo wd_{ij} el peso del término i dentro del documento j, w_i el peso del término i en la colección, df_i la frecuencia de documento del término i (el número de documentos en los

cuales aparece el término i), tf_{ij} el número de veces que el término i aparece en el documento j y n el número de documentos de la colección.

Las expresiones presentadas se caracterizan por hacer una asignación de pesos a los términos de la colección próxima al poder de resolución, asignando valores bajos de w_i a los términos que aparecen frecuentemente en la colección, así como a los que tienen muy baja frecuencia de aparición. Únicamente aquellos términos con una frecuencia de aparición media dentro de la colección obtendrán valores altos de acuerdo a las fórmulas de pesado aplicadas. Finalmente, debemos destacar que existen diferencias sutiles entre las expresiones presentadas y la definición del poder de resolución. Mientras que, en las fórmulas de pesado, el peso asociado a un término depende del número de documentos en los que aparece, en el poder de resolución este valor depende del número total de veces que aparece en el conjunto de documentos. Esta diferencia introduce mejoras en el proceso de recuperación, de acuerdo con estudios teóricos y experimentales [108].

- *Representación de los documentos y de las consultas.* Para la representación (indexación) tanto del conjunto de documentos como de las consultas asociadas optaremos por aplicar el *Modelo del Espacio Vectorial* (MEV). Este modelo se basa en una representación de la información a partir de descriptores o términos donde tanto los documentos de la colección como las solicitudes se representan mediante vectores de términos obtenidos automáticamente del propio corpus.

  Para cada documento d_j se calcula la frecuencia de aparición de cada término de indexación tf_{ij} , asignando a cada término un peso wd_{ij} en el documento mediante la expresión:

$$wd_{ij} = tf_{ij} * w_i \tag{6.4}$$

  De este modo, el documento d_j quedaría representado por un vector de dimensión m, con los pesos asignados a cada uno de los términos de indexación del siguiente modo:

$$d_j = \langle wd_{1j}, wd_{2j}, \ldots, wd_{mj} \rangle \tag{6.5}$$

  De igual modo, dada una consulta q_k, su representación de acuerdo al MEV pasaría por considerarla como un documento más del corpus. De este modo, sería necesario calcular los pesos wq_{ik} de cada uno de los términos que figuran en la consulta de igual manera a como se procedió para caracterizar los documentos.

$$wq_{ik} = tfq_{ik} * w_i \tag{6.6}$$

  Donde tfq_{ik} la frecuencia de aparición del término i dentro de la consulta q_k. La consulta quedaría representada por el siguiente vector:

$$q_k = \langle wq_{1k}, wq_{2k}, \ldots, wq_{mk} \rangle \tag{6.7}$$

6.2.3. Proceso de recuperación

Todo proceso de recuperación requiere de la existencia de un mecanismo, denominado *función de similitud*, capaz de determinar la proximidad entre los documentos y las consultas de los usuarios. Sobre una representación basada en vocabulario no controlado existen multitud de funciones de similitud [108]. En este trabajo se ha elegido como función de similitud la basada en el coseno del ángulo formado por el vector que representa a cada uno de los documentos y el vector que representa la consulta. Esta función ha sido ampliamente utilizada y, en término medio, proporciona buenos resultados [7]. Formalmente se expresa del siguiente modo:

$$sim(d_j, q_k) = \frac{\sum_{i=1}^{m} wd_{ij} * wq_{ik}}{\sqrt{\sum_{i=1}^{m} wd_{ij}^2 * \sum_{i=1}^{m} wq_{ik}^2}} \tag{6.8}$$

Donde la expresión representa el grado de similitud entre el documento d_j y la consulta q_k y su resultado está normalizado entre 0 y 1. Cuanto más próximo se encuentre este valor a 1, más relevante será el documento d_j a la consulta q_k.

6.3. Proceso de extracción de descriptores

Tal y como fue expuesto en la sección 6.1, una vez recuperado el conjunto de n documentos relevantes a una consulta específica es necesario realizar un procesamiento de los mismos con el fin de obtener un conjunto de descriptores candidatos a formar parte del conjunto $Desc$ del contexto formal. El objetivo principal del proceso de extracción de descriptores es el de obtener un conjunto de todos los descriptores válidos asociados al conjunto de documentos recuperado con el fin último de que una pequeña parte de éstos sean seleccionados en un proceso posterior.

Determinar el tipo de descriptores más adecuado para la construcción de un clustering de documentos basado en nuestro modelo es fundamental. Esto es debido a que el tipo de descriptor utilizado no sólo influye en la comprensión que el usuario tendrá del clustering obtenido, sino que también está ligado directamente con el número de clusters generados así como con su población (el número de documentos asignados a cada cluster). Desde este punto de vista, y con el fin de poder evaluar las aportaciones de nuestro modelo de clustering en escenarios diferenciados esencialmente en el modo en el que se lleva a cabo la etiquetación de los clusters, en este trabajo se propone la obtención de las descripciones asociadas a un clustering en base a tres tipologías diferentes de descriptor. En concreto, en esta Tesis Doctoral se presentan tres aproximaciones para la extracción y posterior selección de descriptores basadas en la utilización de unigramas, n-gramas de longitud no definida y, como caso particular, sintagmas terminológicos (tal y como se definen en [96]) como unidades básicas de información para la construcción de las etiquetas asociadas a cada cluster.

Los descriptores basados en unigramas se caracterizan por describir los clusters que etiquetan en base a unidades de información mínimas (palabras). Esto permite crear clusters con una población[1]

[1] En este contexto, entenderemos por *población* o *cardinalidad del cluster* el número de documentos contenidos en un cluster concreto

muy alta aunque, como contrapartida, la complejidad del clustering suele ser elevada (se genera un gran número de relaciones entre los clusters) y los descriptores suelen ser poco informativos. En contraste, el uso de descriptores basados en n-gramas y sintagmas terminológicos permite dotar a los clusters de etiquetas más significativas para el usuario debido a que habitualmente representan conceptos lexicalizados o acciones concretas de éstos. Su proceso de obtención es más complejo y suelen generar clusters con una población inferior a la obtenida mediante el uso de unigramas como descriptores. Como principal ventaja, las estructuras de clustering obtenidas suelen ser mucho más simples, informativas y útiles para el usuario.

Cada una de las aproximaciones propuestas influirá de manera directa en las características estructurales de las agrupaciones generadas, siendo uno de nuestros objetivos el de evaluar las mejoras que cada uno de ellas aporta desde el punto de vista del acceso a la información y de acuerdo a las consideraciones fundamentales presentadas en la sección 5.1.

6.3.1. Extracción de unigramas

Como primera aproximación, utilizaremos la representación mediante unigramas de los descriptores asociados al contexto formal. Entenderemos por unigrama cualquier término de longitud uno contenida en un documento y tal que no es una palabra vacía.

La representación de descriptores basada en unigramas implica un preprocesado inicial del conjunto de documentos con el fin de obtener una representación de éstos fácilmente manipulable. Una vez realizado el preprocesamiento inicial, se hace necesario realizar un índice que permita ordenar los unigramas obtenidos de acuerdo a criterios específicos relacionados con su frecuencia de aparición o con su peso dentro del conjunto de documentos recuperado. En el caso de los unigramas es relativamente sencillo obtener este índice. Esto es debido a que, teniendo acceso al fichero de índices invertido utilizado por el motor de búsqueda que realiza el proceso de recuperación, este tipo de información es directamente accesible. Sin embargo, y debido a que nuestro modelo pretende ser aplicado sobre cualquier tipo de escenario de recuperación de información, donde no siempre es posible acceder a este tipo de índices, debemos imponer que el requisito de disponer de un fichero de índices invertido no sea imprescindible. Por ejemplo, aplicar nuestro modelo de clustering al conjunto de resultados obtenidos a partir de una consulta web supone tener acceso a un conjunto de resúmenes (denominados *snippets*) que reproducen una pequeña parte del documento completo. En estos casos, el acceso al fichero de índices invertido no es posible y, por lo tanto, se hace necesario realizar un preprocesamiento inicial de los resúmenes con el fin de obtener dicha información.

El proceso completo para la obtención del conjunto de unigramas candidatos se describe detalladamente a continuación:

- *Preprocesado del conjunto de documentos recuperado.* Este proceso consiste en la eliminación de las palabras vacías, así como en la extracción de las raíces de los términos no eliminados.

- *Construcción de un índice.* El preprocesado de los documentos permite obtener una representación mucho más adecuada del conjunto de documentos recuperado. Una vez realizada esta

tarea, es necesario llevar a cabo la creación de un índice invertido que permita relacionar cada término con cada uno de los documentos en los cuales aparece y que, además, permita obtener su frecuencia relativa, así como su frecuencia de documento. Además, y con el fin de poder mostrar al usuario una representación completa y comprensible de los términos extraídos, el índice deberá asociar a cada término normalizado un *representante*. De entre todas las posibles variaciones morfológicas asociadas a un término, el representante será aquella variación cuya frecuencia de documento sea mayor dentro del conjunto de documentos recuperados.

- *Proceso de preselección.* Con el fin de mejorar el proceso de selección sobre el conjunto de unigramas extraído, resulta conveniente eliminar del índice generado aquellos términos cuya frecuencia de documento sea menor a un cierto umbral. Este umbral puede obtenerse a partir de las frecuencias de documento de los unigramas obtenidos, siendo su objetivo el de eliminar directamente del posterior proceso de selección aquellos descriptores cuyo poder para generar clusters significativos (que agrupen un número elevado de documentos) sea pequeño.

6.3.2. Extracción de n-gramas

La segunda aproximación propuesta se basa en la construcción del conjunto de descriptores a partir de un conjunto de n-gramas de longitud variable previamente extraídos de la colección de documentos. En este trabajo entendemos por n-grama de longitud no definida cualquier secuencia de palabras extraída de la colección de documentos tal que no comience ni termine por una palabra vacía.

El proceso de extracción de n-gramas es un poco más complejo presentado en la sección anterior. Su complejidad radica en que no se pretende extraer términos independientes (tarea relativamente sencilla si se dispone de un índice invertido), sino secuencias de palabras cuya frecuencia de aparición en el conjunto de documentos sea significativa. En este caso, el uso de un índice invertido no facilita la tarea de extracción, sino que se requiere de una estructura adicional capaz de almacenar todas las secuencias posibles de las palabras del texto.

El proceso para llevar a cabo la extracción de todos los n-gramas candidatos se describe detalladamente a continuación:

- *Preprocesado del conjunto de documentos recuperado.* Al igual que en el proceso de extracción de unigramas, el objetivo de esta subtarea es el de eliminar del texto original todas las palabras vacías y, sobre el conjunto de términos restante, realizar un proceso de extracción de raíces. En el caso concreto de los n-gramas, este proceso deberá obtener, además, una representación adecuada para cada uno de los documentos de manera que el acceso a las posiciones originales de las raíces dentro del documento original sea directa y permita extraer las secuencias completas de términos (incluyendo las palabras vacías y las variaciones morfológicas concretas de cada uno de los términos).

- *Construcción del árbol de sufijos.* El objetivo de este segundo proceso es el de extraer del conjunto de documentos recuperado el conjunto de n-gramas factibles que deberán ser seleccionados en un proceso posterior. En esta Tesis Doctoral se ha optado por el uso de *árboles de sufijos* (introducidos en el Anexo A.2 de este trabajo). Este tipo de estructuras se caracterizan

por generar y almacenar, con un coste computacional relativamente bajo, todos los sufijos posibles de las distintas subcadenas contenidas en un conjunto de documentos mejorando, además, su acceso y recuperación [9]. Este tipo de estructuras suelen aplicarse utilizando los caracteres como unidades básicas para la construcción del árbol. Sin embargo, en nuestra aproximación proponemos su uso utilizando como unidades básicas los términos normalizados (también denominados *tokens*) del conjunto de documentos. De esta manera, el árbol de sufijos construido permitirá acceder a las secuencias válidas de términos en lugar de a las secuencias válidas de caracteres. La sección 6.3.2 expone con detalle el modo en el que realizamos la construcción del árbol, así como su particularización al contexto sobre el cual lo aplicamos.

- *Construcción de un índice.* Una vez calculado el árbol de sufijos se dispone de información suficiente sobre los n-gramas contenidos en la colección, así como de los documentos en los cuales aparecen, sus frecuencias relativas y sus frecuencias de documento. Con el fin de facilitar el acceso al conjunto de sufijos extraídos, proponemos la creación de un índice invertido donde figuren las relaciones de cada uno de los n-gramas extraídos con los documentos en los cuales aparecen.

- *Proceso de preselección.* Finalmente, sobre el índice generado se eliminarán aquellos n-gramas cuya frecuencia de documento sea inferior a un cierto umbral. El objetivo es el de eliminar del índice aquellos n-gramas cuyo poder de resolución sea pequeño y, como consecuencia, su capacidad para generar clusters suficientemente significativos.

A continuación presentamos la teoría de los árboles de sufijos aplicada a nuestra aproximación. Si el lector no estuviera familiarizado con ella recomendamos una lectura previa del Anexo A.2.

Propuesta para la construcción de un árbol de sufijos

En contraste con las aplicaciones clásicas de árboles de sufijos, nuestra propuesta considera las raíces de los términos como unidades básicas para la construcción de cada nodo del árbol. En nuestra aproximación, cada uno de los nodos del árbol de sufijos representará un n-grama factible construido a partir de una secuencia de tokens extraída de los documentos recuperados. Además, cada nodo tiene asociado un subárbol donde estarán representadas todas las extensiones (nuevos n-gramas) aplicables a la secuencia de tokens inicial para la construcción de n-gramas mayores.

Nuestra propuesta no se orienta a la utilización de los árboles de sufijos para la recuperación de n-gramas específicos, sino que aplica este tipo de estructuras para facilitar la extracción de la totalidad de n-gramas contenidos en la colección de documentos con el fin de realizar un proceso de selección posterior.

La construcción del árbol de sufijos parte del conjunto de documentos recuperado pero, tal y como se indicó anteriormente, requiere de la realización de un proceso previo de procesamiento sobre la colección. Es importante destacar que aunque el árbol de sufijos trabaje con una representación reducida o simplificada de los documentos originales es necesario disponer de mecanismos adecuados para reconstruir las secuencias originales del texto. Con este fin, nuestra aproximación requiere de la

Docs	$\vec{docs}$
d_1	<Juan, ,corre, ,por, ,el, ,campo,.>
d_2	< Por, ,el, ,campo, ,corre, ,Juan,.>
d_3	< En, ,el, ,campo, ,hay, ,rios, ,que, ,corren, ,hacia, ,el, ,mar,.>

Cuadro 6.1: Vector $\vec{docs}$ correspondiente a un conjunto de documentos

Docs	$\vec{docs}'$
d_1	<juan,corre,campo >
d_2	< campo,corre,juan >
d_3	<campo,rio,corre,mar >

Cuadro 6.2: Vector $\vec{docs}'$ obtenido a partir del vector $\vec{docs}$ presentado en el cuadro 6.1

existencia de una correspondencia entre los documentos originales y su representación preprocesada que permita llevar a cabo esta transformación.

Formalmente, dado un conjunto de n documentos recuperados por un motor de búsqueda, denominaremos $\vec{docs} = \langle \vec{d_1}, \vec{d_2}, \ldots, \vec{d_n} \rangle$ al vector que contiene todos los documentos recuperados, y donde cada documento está representado por un vector, $\vec{d_j} = \langle t_1, t_2, \ldots, t_m \rangle$, que contiene la secuencia completa de palabras y separadores contenidos en éste (incluyendo palabras vacías, espacios en blanco y signos de puntuación). Llamaremos $\vec{docs}'$ al vector que representa el conjunto de n documentos preprocesado tal y como se ha descrito anteriormente (eliminación de palabras vacías y signos de puntuación y obtención de raíces). Es posible establecer una correspondencia entre los elementos de $\vec{docs}$ y de $\vec{docs}'$ que permita determinar de manera unívoca la posición de cada una de las raíces contenidas en cada uno de los del vectores de $\vec{docs}'$ sobre el texto original. Los cuadros 6.1, 6.2 y 6.3 ejemplifican esta aproximación. El cuadro 6.1 representa al vector $\vec{docs}$ inicial obtenido a partir de un conjunto de documentos recuperado (donde el índice i de cada uno de los elementos de $\vec{docs}'$ representa la posición del término sobre el texto original), mientras que el cuadro 6.2 muestra el vector reducido $\vec{docs}'$. Puede observarse como en este último vector sólo aparecen las raíces de los términos significativos en el orden en el que figuran en el texto original. Finalmente, el cuadro 6.3 muestra la correspondencia entre $\vec{docs}'$ y $\vec{docs}$ que permite localizar de manera directa las raíces de los documentos en el texto original y, por tanto, extraer el n-grama original del cual éstas han sido obtenidas.

La estructura básica sobre la cual se basa la construcción del árbol de sufijos propuesta es la del nodo. Su objetivo es el de almacenar la información relativa a una secuencia de tokens válida. Con el fin de optimizar el uso de memoria para el almacenamiento completo del árbol, un nodo sólo almacenará los índices sobre el vector $\vec{docs}'$ que definan la secuencia de tokens almacenada en lugar de sus correspondientes cadenas de caracteres. En el proceso posterior de reconstrucción de los n-gramas originales se hará uso de éstos índices y de la función de correspondencia para extraer las cadenas de caracteres completas. De forma un poco más detallada, exponemos los principales

Docs	$\vec{docs}' \longrightarrow \vec{docs}$
d_1	< 0,2,8 >
d_2	< 4,6,8 >
d_3	< 4,8,12,18 >

Cuadro 6.3: Correspondencia para obtener la posición de cada término del vector $\vec{docs}'$ sobre el texto original en $\vec{docs}$

Sufijos	**Frecuencia**
Juan corre por el campo	1
corre	3
corre por el campo	1
corre juan	1
corren hacia el mar	1
campo	3
campo hay rios que corren hacia el mar	1
campo corre juan	1
rios que corren hacia el mar	1

Cuadro 6.4: Conjunto de n-gramas extraídos del conjunto de documentos del cuadro 6.1 y sus frecuencias de documento asociadas

componentes de un nodo a continuación:

- *Documento.* Almacena el identificador del documento al cual se encuentra asociado la secuencia de tokens que define el nodo.
- *Documentos.* Almacena los identificadores de documento en los cuales aparece la secuencia de tokens asociada al nodo, así como la frecuencia de aparición de ésta en cada uno de los documentos.
- *Posición.* Posición de inicio de la secuencia de tokens dentro del documento almacenado en *Documento*. La posición se define en función de la posición del token dentro del vector del documento $\vec{docs}_i'$.
- *Número de Tokens.* Numero de tokens que componen la secuencia de tokens almacenada en el nodo.

La figura 6.3 muestra el árbol de sufijos completo correspondiente al ejemplo presentado en los cuadros 6.1, 6.2 y 6.3. El conjunto de sufijos extraídos se muestra en el cuadro 6.4 donde, además, se refleja la frecuencia de documento de cada uno de ellos.

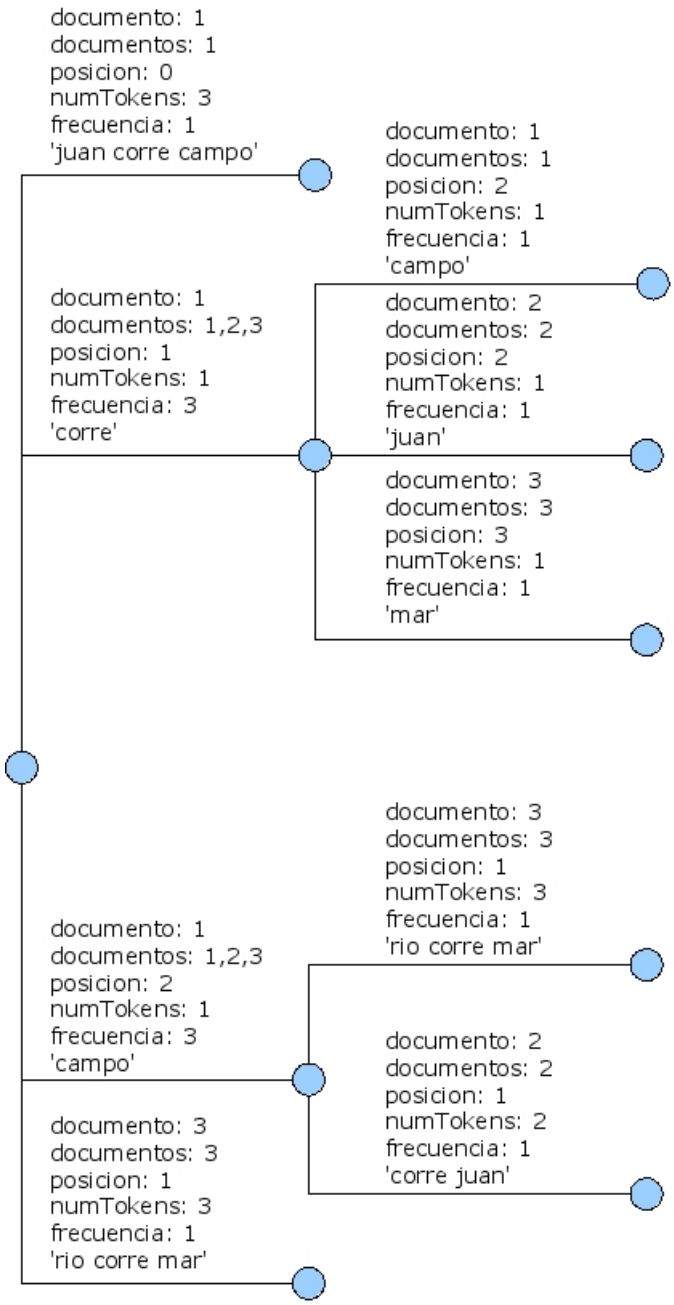


Figura 6.3: Árbol de sufijos correspondiente al conjunto de los n-gramas extraídos del conjunto de documentos del cuadro 6.1

6.3.3. Extracción de sintagmas terminológicos

Tal y como se ha expuesto al comienzo de este sección, en este trabajo consideramos la extracción de un último tipo de descriptor que puede considerarse como un caso particular de n-grama y que denominaremos sintagma terminológico.

Los sintagmas terminológicos, tal y como se definen en [96], pueden entenderse como sintagmas nominales tales que su frecuencia de documento es suficiente como para representar conceptos lexicalizados. Esta propuesta está basada en que la expresión lingüística de un concepto suele corresponderse con un sintagma nominal.

Las principales diferencias de los sintagmas terminológicos con respecto a la aproximación basada en n-gramas radica en dos aspectos principales:

- *El proceso de extracción.* En el caso de los sintagmas terminológicos se requiere la aplicación de técnicas de procesamiento del lenguaje natural que permitan categorizar sintácticamente cada una de las palabras que aparecen en el texto de los documentos [96]. Este tipo de técnicas y herramientas son dependientes del idioma en el que se lleva a cabo el proceso de extracción y su aplicación directa sobre un conjunto de documentos obtenido dinámicamente suele traducirse en una inversión de tiempo de cálculo que, en algunas ocasiones, no justifica los resultados obtenidos. En contraste, las técnicas de extracción basadas en n-gramas presentadas anteriormente requieren de un procesamiento mucho más ligero y no hacen necesario disponer de herramientas lingüísticas específicas para cada idioma permitiendo, como consecuencia, la generalización de nuestra aproximación a cualquier escenario de recuperación.
- *La estructura sintáctica de los descriptores extraídos.* La extracción de sintagmas terminológicos limita la estructura sintáctica de los descriptores extraídos a combinaciones de sintagmas nominales y sintagmas preposicionales. Sin embargo, en la aproximación basada en n-gramas esta restricción no se impone, permitiendo cualquier secuencia de términos como n-grama válido (siempre y cuando no comience ni termine por palabra vacía).

En esta Tesis Doctoral se pretende ilustrar la aplicación de los sintagmas terminológicos al modelo propuesto, así como llevar a cabo una evaluación de su idoneidad para la realización de agrupaciones de documentos adecuadas, tanto desde el punto de vista de su estructura como de su capacidad informativa, sin centrarnos en describir su proceso de extracción. Por esta razón, en este trabajo haremos uso de listas terminológicas previamente extraídas a partir de grandes colecciones de documentos, remitiendo para los detalles acerca de su obtención a [96].

6.4. Proceso de selección de descriptores

El siguiente paso consiste en seleccionar un pequeño conjunto de descriptores capaz de describir de la manera más completa posible el contexto formal sobre el cual se construirá la estructura final de clustering.

Dado que el clustering obtenido depende de manera directa del contexto formal que se genera a partir de la información recuperada, resulta crucial determinar qué descriptores son los más apropiados

para que el retículo obtenido cumpla las consideraciones fundamentales presentadas en la sección 5.1. Tal y como expusimos entonces, un buen proceso de clustering debe tener como objetivos principales separar adecuadamente la información relevante de la información no relevante, generando una estructura lo suficientemente simple como para que el usuario pueda acceder a la información requerida de forma sencilla.

El objetivo principal de esta sección es el de presentar un conjunto de técnicas orientadas a la obtención del conjunto de descriptores más adecuado tanto para describir el conjunto de documentos recuperados como para producir un clustering navegable y suficientemente informativo.

6.4.1. Relación entre el proceso de selección de descriptores y la estructura de clustering generada

En el proceso de construcción del clustering la definición de un contexto formal resulta imprescindible y su objetivo es el de establecer las relaciones iniciales entre el conjunto de documentos recuperado y un conjunto de atributos, previamente seleccionado, capaces de describir de la manera más exhaustiva posible la información que se desea agrupar. El retículo obtenido a partir del contexto inicial tiene como característica principal la de presentar el conjunto de relaciones iniciales de forma mucho más intuitiva, describiendo cada uno de los clusters mediante combinaciones de los atributos inicialmente asignados al contexto. Obviamente, la definición inicial del contexto formal es definitiva a la hora de obtener estructuras de clustering con un número de clusters adecuado, así como con un número de relaciones entre los clusters que sea razonable y comprensible por el usuario.

Idealmente, el conjunto de descriptores utilizado para la generación de un clustering debería:

- *Minimizar el número de conceptos objeto maximizando la precisión del clustering.* Tal y como expusimos al presentar el modelo, de todos los conceptos formales obtenidos los más interesantes desde el punto de vista del clustering son los conceptos objeto. Este tipo de conceptos definen la extensión de los nodos de información presentados en el modelo. Su principal característica es la de albergar la información (documentos) que ha recuperado el sistema y, por lo tanto, el objetivo final al que desea acceder el usuario. Es lógico pensar que un clustering capaz de separar correctamente la información relevante en un gran número de conceptos objeto obliga al usuario a visitar una gran cantidad de clusters con muy pocos documentos en su interior. Este tipo de escenarios, aunque no sean incorrectos desde el punto de vista del clustering, pueden suponer para el usuario un aumento innecesario del coste de exploración. La figura 6.4 muestra un ejemplo donde puede verse el efecto de tener un gran número de conceptos objeto. En la primera de las figuras se observa una estructura de clustering correcta (están correctamente separados los documentos relevantes de los documentos no relevantes) pero donde los clusters descritos por los descriptores $d2$, $d3$, $d5$ y $d6$ presentan una población muy baja (únicamente contienen un documento en su interior). En contraste, la segunda de las figuras representa una estructura de clustering igualmente válida caracterizada, en este caso, por tener un menor número de descriptores y, como consecuencia, un menor número de conceptos objeto. Esta situación produce una mayor concentración de documentos en los

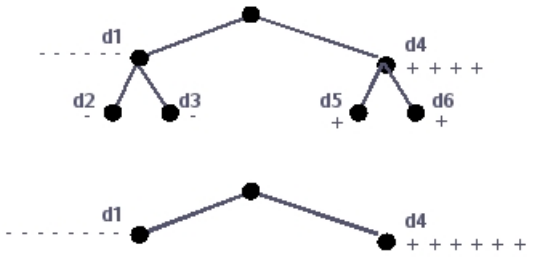


Figura 6.4: Ejemplo de dos agrupaciones válidas diferenciadas en el número de conceptos objeto que contienen

conceptos objeto existentes mejorando, como consecuencia, las características navegacionales del clustering generado.

- *Compactar los conceptos atributo en clusters que también sean conceptos objeto*. Los conceptos atributo se caracterizan por definir el concepto más genérico descrito por alguno de los atributos de su intensión. El que un concepto formal sea un concepto atributo no implica que también sea un concepto objeto y, de hecho, esto no ocurre en los retículos construídos a partir de grandes colecciones de datos. De acuerdo al modelo propuesto, la situación en la que un cluster es concepto atributo pero no es concepto objeto se puede entender como un cluster etiquetado con uno o varios descriptores que no contiene documento alguno. Consideramos que esta situación, desde el punto de vista del usuario, no es la más intuitiva debido a que éste interpreta un nodo etiquetado como un nodo importante dentro del clustering y, por tanto, espera encontrar documentos en su interior. La solución a esta situación pasa por generar un retículo donde la mayor parte de los conceptos atributo cumplan la condición de ser también concepto objeto. La idea intuitiva que se esconde detrás de esta condición es la de proveer al contexto formal de algunos documentos descritos de forma completa por un único descriptor evitando, de este modo, su aparición en clusters mucho más especializados dentro del clustering obtenido. La figura 6.5 muestra un ejemplo donde puede verse como los clusters etiquetados con los descriptores $d1$, $d2$, $d3$ y $d4$ representan conceptos atributo. En este caso, dichos clusters no se caracterizan por ser conceptos objeto y, como consecuencia, al ser explorados el usuario no encontrará documento alguno asociado a dichos descriptores. Tal y como exponíamos anteriormente, los documentos aparecen en clusters mucho más especializados (combinación de más de uno de los descriptores) debido a que en el contexto formal estos documentos están representados por más de un descriptor.

- *Minimizar el número de descriptores que no son concepto objeto ni concepto atributo*. De acuerdo a nuestro modelo, los únicos conceptos que realmente aportan información útil al clustering generado son aquellos que, o bien son conceptos objeto, o bien son conceptos atributo. No obstante, no debemos olvidar que la teoría del AFC puede generar, a partir del contexto formal inicial, conceptos tales que no sean ni concepto objeto ni concepto atributo. Su extensión e intensión se crea a partir de los conceptos padre realizando la intersección

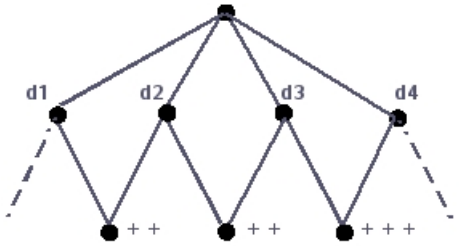


Figura 6.5: Ejemplo de un clustering donde los conceptos objeto y los conceptos atributo se muestran desacoplados

de las extensiones y la unión de las intensiones respectivamente. Intuitivamente, los clusters asociados a este tipo de conceptos se caracterizan por estar descritos por un conjunto de atributos heredados de las intensiones de sus clusters padre y por no contener documento alguno. Nuevamente, consideramos que esta situación no es deseable desde el punto de vista de la usabilidad del clustering y, por tanto, una buena estrategia de selección de descriptores debería estar orientada a minimizar el número de clusters de este tipo. La figura 6.6 muestra un ejemplo donde el cluster central (descrito por los atributos d_2 y d_3) se caracteriza por no ser ni concepto objeto ni concepto atributo, siendo su fin último el de relacionar partes disjuntas del retículo.

- *Minimizar la población del nodo raíz del clustering.* El último requisito que vamos a exigir al proceso de selección de descriptores es el de reducir el número de documentos contenidos en el nodo raíz del clustering. El nodo raíz se caracteriza por se el cluster que primero visita el usuario cuando comienza su exploración sobre la estructura de clustering generada. Habitualmente, y salvo que se haya seleccionado un descriptor común a todos los documentos recuperados, el nodo raíz contiene aquellos documentos que no pueden ser descritos por alguno de los descriptores seleccionados. Por esta razón, un clustering con un nodo raíz que contenga una gran cantidad de documentos no es conveniente desde el punto de vista del clustering. En el caso de utilizar unigramas como descriptores de los nodos del clustering este requisito no es tan importante debido a que el número de documentos asociados a este tipo de descriptores suele ser elevado (es habitual que un unigrama figure en un gran número de documentos) y por lo tanto, únicamente una pequeña parte de éstos quedarán sin describir por alguno de los descriptores seleccionados. Por el contrario, en el caso de trabajar con n-gramas o sintagmas terminológicos este requisito toma especial relevancia debido a que este tipo de descriptores suelen afectar a un menor número de documentos y, como consecuencia, resulta más probable encontrar documentos que al final del proceso no estén descritos por algún descriptor. En este último caso tiene especial relevancia diseñar estrategias de selección que presten especial atención al número de documentos que finalmente quedan sin describir en el nodo raíz.

Además de los objetivos presentados, existe un objetivo principal que cualquier aproximación para la selección de descriptores debe tratar de cumplir. Este objetivo se orienta a generar estructuras

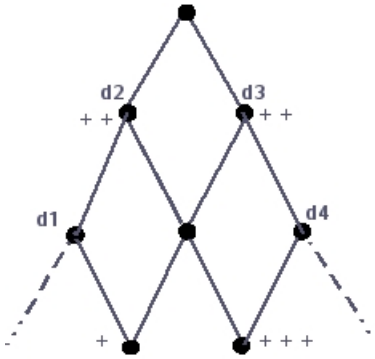


Figura 6.6: Ejemplo de un clustering con un cluster que no es ni concepto objeto ni concepto atributo

de clustering con un número de clusters razonable como para justificar el proceso realizado. Desde nuestro punto de vista, consideramos que las estructuras de clustering generadas utilizando nuestra propuesta en ningún caso deberían obligar al usuario a explorar un número de clusters superior al 50 % del número de documentos sobre los cuales se lleva a cabo el proceso de organización. Un proceso de exploración que implique acceder a un número de clusters superior supondría un coste cognitivo muy superior al necesario para acceder a la información relevante directamente a través de la lista original de documentos por lo que, en estos casos, no tendría sentido realizar el clustering.

6.4.2. Aproximaciones para la selección de descriptores

En esta sección se presenta un conjunto de aproximaciones factibles para llevar a cabo la selección de descriptores, cuya adecuación a la tarea global será evaluada en el capítulo 8.

A continuación se detallan las tres aproximaciones propuestas en este trabajo para llevar a cabo el proceso de selección de descriptores.

6.4.3. Aproximación basada en $tf - idf$

Como primera propuesta para la selección de descriptores presentamos una aproximación basada en el pesado de los términos utilizando $tf - idf$ (consultar sección 6.2).

Este pesado de descriptores ha sido ampliamente estudiado en el área de la Recuperación de Información y existen numerosas aproximaciones capaces de asociar un peso concreto a cada término o frase contenida en un documento. Las fórmulas 6.2 y 6.3, presentadas anteriormente, son uno de los ejemplos más sencillos y, al mismo tiempo, más utilizados en RI para la representación de los documentos de una colección.

Sin embargo, nosotros hemos optado por utilizar una fórmula un poco más compleja derivada de la familia de fórmulas denominadas Best Match (BM) [102, 103, 104] y, en concreto, del caso particular de la fórmula BM25 [104]. La primera de las fórmulas perteneciente a esta familia (BM1) originalmente fue propuesta por Robertson-Sparck Jones [102], dando lugar a un gran número de

variantes cuya mejora en la eficiencia en tareas de RI ha sido ampliamente demostrada en competiciones tan conocidas como el TREC o el CLEF. Esta forma de pesar los términos de un documento para la realización de tareas de RI recibe el nombre genérico de pesado OKAPI BM25, siendo sus características principales las de determinar el peso de cada uno de los términos en función de parámetros extra, además de los habituales $tf - idf$, tales como la longitud media de los documentos de la colección o la longitud de cada uno de los documentos. Este tipo de pesado de términos dispone de una serie de constantes que deben ser ajustadas a las características específicas de cada una de las colecciones sobre las cuales se lleva a cabo el proceso de recuperación.

En concreto, nuestra aproximación para el pesado de términos opta por una variante de la fórmula BM25 que se presenta en las ecuaciones 6.9,6.10, 7.4 y 6.12. Donde b, k_1 y $avdl$ son constantes que dependen de las características propias de la colección. Debemos destacar que para el cálculo de la fórmula, y dado que lo que perseguimos es calcular un único peso para cada término, hemos considerado el conjunto de documentos recuperado como un único documento. De acuerdo a esto, tf_i representa la frecuencia del término i en el documento (que en nuestro caso será su frecuencia en el conjunto de documentos recuperados) y df_i la frecuencia de documento del término i en la colección completa (nuevamente, este valor deberá tener en cuenta nuestra restricción). Finalmente, l_{ret} representa la longitud total (medida en número de términos) del conjunto de documentos recuperado.

$$w_i = tf * idf \tag{6.9}$$

$$idf = \frac{ln\left(\frac{N + 0{,}5}{df_i}\right)}{ln(N+1)} \tag{6.10}$$

$$df = \frac{(k_1+1)tf_i}{K + tf_i} \tag{6.11}$$

$$K = k_1 * \left[(1-b) + b * \frac{l_{ret}}{avdl}\right] \tag{6.12}$$

La metodología propuesta para llevar a cabo la selección de los k mejores descriptores utilizando esta aproximación sería la siguiente:

1. Siendo $D = \{doc_1, doc_2, \ldots, doc_n\}$ el conjunto de los n documentos más relevantes recuperado, y siendo $P = \{desc_1, desc_2, \ldots, desc_m\}$ el conjunto de descriptores candidatos extraído de dichos documentos. Realizamos el pesado, utilizando las fórmulas 6.9 y 6.12, del conjunto de descriptores P.

2. Ordenamos el conjunto de descriptores por orden descendente de pesos.

3. Finalmente se lleva a cabo la selección de los k primeros descriptores.

6.4.4. Aproximación terminológica

En contraste con la propuesta anterior, aquí proponemos una aproximación para el pesado de descriptores obtenida del área de la extracción de terminología para la construcción automática de diccionarios terminológicos. Su objetivo final es el de destacar el conjunto de descriptores característico o diferenciador de un conjunto de documentos específico a partir de la comparación con un corpus de documentos genérico.

En nuestra aproximación pretendemos aplicar esta estrategia a la comparación del conjunto de documentos recuperado con respecto al corpus de documentos completo sobre el cual se lleva a cabo el proceso de recuperación y de clustering. Como resultado final de este proceso se pretende obtener aquellos descriptores característicos del conjunto de documentos recuperado.

Para materializar el comportamiento descrito, proponemos aplicar la fórmula 6.13 que originalmente fue propuesta en [97]. En esta fórmula w_i representa el peso terminológico asignado al descriptor i, $tf_{i,ret}$ es la frecuencia relativa del descriptor i en el conjunto de documentos recuperado, $df_{i,ret}$ representa la frecuencia de documento del descriptor i sobre el conjunto de documentos recuperado y $tf_{i,col}$ es la frecuencia relativa del descriptor i en la colección completa sin tener en cuenta el conjunto de documentos recuperado.

$$w_i = 1 - \frac{1}{log_2\left(2 + \frac{tf_{i,ret} df_{i,ret} - 1}{tf_{i,col} + 1}\right)} \tag{6.13}$$

En esta segunda aproximación, la metodología propuesta para llevar a cabo el proceso de selección sería el mismo que el que hemos expuesto para realizar la selección de descriptores utilizando un pesado $tf - idf$.

6.4.5. Aproximación basada en un algoritmo balanceado

Finalmente, y con el fin de aumentar la población de los clusters generados (sobre todo en el caso concreto de utilizar descriptores basados en n-gramas de longitud no definida), proponemos una aproximación orientada a maximizar la distribución de los documentos recuperados sobre el conjunto final de clusters.

Tal y como demostraremos en el capítulo 8, las aproximaciones propuestas en las secciones anteriores producen muy buenos resultados en el caso de utilizar unigramas como descriptores en el contexto formal. Sin embargo, en el caso de utilizar n-gramas de longitud variable o sintagmas terminológicos como descriptores, las aproximaciones propuestas generan retículos caracterizados por organizar únicamente una pequeña parte del conjunto de documentos recuperado y, como consecuencia, por disponer de nodos raíz excesivamente poblados. Con el fin de poder aplicar nuestro modelo a este tipo de descriptores(mucho más comprensibles por parte del usuario) optamos por diseñar una estrategia de selección que permita controlar de manera mucho más directa la distribución de los documentos sobre el conjunto de descriptores seleccionado.

La idea es, por tanto, cubrir la mayor parte del conjunto de documentos recuperado seleccionando un conjunto de k descriptores. Para ello trabajaremos con la frecuencia de documento de cada uno

de los descriptores como indicador directo de la cantidad de documentos cubierta por cada uno de ellos, sin realizar cálculo de pesos alguno sobre el conjunto de descriptores extraído. Como ventajas adicionales es necesario destacar que en esta aproximación, al contrario que en las anteriores, no es necesario conocer a priori la colección completa sobre la cual se lleva a cabo el proceso de recuperación y de clustering haciendo que el proceso global sea, de este modo, mucho más independiente y aplicable a cualquier tipo de escenario.

El algoritmo propuesto para llevar a cabo el proceso de selección es el siguiente:

- Sea $D = \{doc_1, doc_2, \ldots, doc_n\}$ el conjunto de los n documentos más relevantes recuperado, y $P = \{desc_1, desc_2, \ldots, desc_m\}$ el conjunto de descriptores candidatos extraído de dichos documentos. Definimos un conjunto $G = \emptyset$ que almacenará los documentos cubiertos y un conjunto $S = \emptyset$ que almacenará los descriptores seleccionados.

- Repetir hasta que $|S| = k$ o $|D| = 0$, donde k es el número máximo de descriptores que el algoritmo debe seleccionar para representar el conjunto de documentos recuperado.

  1. Extraer de P el descriptor $desc_i$ con mayor frecuencia de documento en D.
     a) En el caso de existir más de un descriptor con la misma frecuencia de documento se seleccionará aquel descriptor perteneciente al documento más relevante de D.
     b) En el caso de que la frecuencia de documento de $desc_i$ supere un cierto umbral (e.g. el 50 % de los documentos recuperados) seleccionaremos el descriptor y realizaremos únicamente los pasos 3 y 5.
  2. Almacenar en un conjunto auxiliar vacío (AUX) aquellos documentos que, perteneciendo a D, contengan el descriptor $desc_i$.
  3. Borrar el descriptor seleccionado del conjunto de descriptores candidatos, es decir, $P = P \setminus desc_i$.
  4. Borrar los documentos seleccionados del conjunto de documentos, lo que implica $D = D \setminus AUX$.
  5. Añadir el descriptor procesado al conjunto final de descriptores seleccionados, es decir, $S = S \cup \{desc_i\}$.
  6. Añadir los documentos seleccionados al conjunto de documentos utilizados. $G = G \cup AUX$.

- El conjunto S contendrá los k descriptores con mayor frecuencia de documento sobre el conjunto de documentos recuperado.

Los cuadros 6.5, 6.6, 6.7, 6.8 muestran el proceso completo de aplicación del algoritmo sobre un conjunto de documentos. El cuadro 6.5 representa un conjunto de 10 documentos recuperados, y ordenados de acuerdo a los criterios de relevancia de un motor de búsqueda, sobre los cuales se han extraído 8 descriptores diferentes. La inicialización del algoritmo produce un conjunto de descriptores $S = \emptyset$, $D = \{D_1, D_2, D_3, D_4, D_5, D_6, D_7, D_8, D_9, D_{10}\}$ y $G = \emptyset$. En la primera iteración

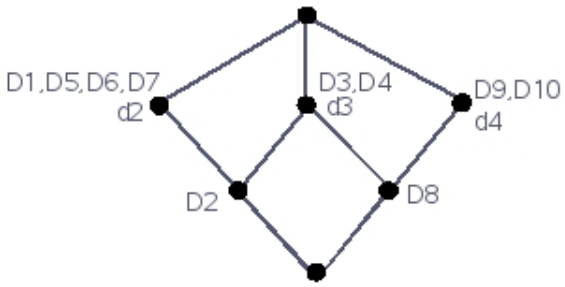


Figura 6.7: Clustering correspondiente al conjunto de descriptores seleccionado del cuadro 6.8

se selecciona el descriptor d_2, que es el que tiene mayor frecuencia de documento sobre D, y se reduce el número de documentos en este conjunto. El proceso de reducción consiste en eliminar de D aquellos documentos que contengan el descriptor que se acaba de seleccionar. Como resultado de este proceso $S = \{d_2\}$, $D = \{D_3, D_4, D_8, D_9, D_{10}\}$ y $G = \{D_1, D_2, D_5, D_6, D_7\}$. Debemos destacar cómo la eliminación de algunos de los documentos de D produce el efecto de variar las frecuencias de documento de los descriptores que todavía no han sido seleccionados, promoviendo que descriptores que anteriormente tenían una frecuencia de documento relativamente baja puedan ser considerados en las siguientes iteraciones del algoritmo. Por ejemplo, los descriptores d_1, d_3 y d_5 tenían una frecuencia de documento igual a 4 sobre el conjunto D antes de realizar la selección del descriptor d_2. Sin embargo, una vez realizada la selección y ajustado el conjunto D, la frecuencia de los descriptores mencionados varía (consultar cuadro 6.6), haciendo que el descriptor d_3 tenga una importancia mayor que d_1 y d_5. Nótese también como el descriptor d_4 cuya frecuencia era de 3 y, por lo tanto, no tenía porqué ser considerado, pasa a formar parte del conjunto de descriptores candidatos a seleccionar en la siguiente iteración. De hecho, en la siguiente pasada del algoritmo se deberá decidir entre los descriptores d_3 y d_4. Tal y como hemos expuesto en la descripción del algoritmo, se seleccionará aquel descriptor contenido en el documento más relevante de D. En nuestro caso, el nuevo descriptor seleccionado será d_3, donde $S = \{d_2, d_3\}$, $D = \{D_9, D_{10}\}$ y $G = \{D_1, D_2, D_5, D_6, D_7, D_3, D_4, D_8\}$. A continuación, el algoritmo debe decidir entre seleccionar los descriptores d_4 o d_8 (cuadro 6.7). En este caso, y debido a que ambos descriptores aparecen en los mismos documentos, el criterio de selección queda abierto y es posible seleccionar cualquiera de los dos. En nuestro caso optaremos por el descriptor d_4, quedando como resultado del proceso $S = \{d_2, d_3, d4\}$, $D = \emptyset$ y $G = \{D_1, D_2, D_5, D_6, D_7, D_3, D_4, D_8, D_9, D_{10}\}$. El cuadro 6.8 muestra el resultado del proceso de selección de descriptores. Tal y como exponíamos al comienzo de este apartado, el algoritmo ha conseguido seleccionar un conjunto de descriptores suficiente como para cubrir la totalidad de los documentos recuperados. Debido a esto, todos los documentos se encontrarán descritos por, al menos, uno de los descriptores seleccionados, haciendo que el nodo raíz del clustering no contenga documento alguno y repartiendo de forma equilibrada los documentos entre todos los clusters generados. La figura 6.7 muestra el clustering obtenido, donde puede apreciarse con un poco más de claridad el resultado de aplicar este método de selección de descriptores.

	Descriptores Seleccionables							
Conjunto D	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8
D_1	×	×						
D_2	×	×	×					
D_3	×		×		×			
D_4	×		×			×		
D_5		×				×		
D_6		×			×			
D_7		×			×		×	
D_8			×	×	×			
D_9				×				×
D_{10}				×			×	×
$FrecDoc$	4	5	4	3	4	2	2	2

Cuadro 6.5: Conjunto de documentos recuperados y sus descriptores asociados. El algoritmo selecciona el descriptor d_2

	Descriptores Seleccionables						
Conjunto D	d_1	d_3	d_4	d_5	d_6	d_7	d_8
D_3	×	×		×			
D_4	×	×			×		
D_8		×	×	×			
D_9			×				×
D_{10}			×			×	×
$FrecDoc$	2	3	3	2	1	1	2

Cuadro 6.6: Conjunto de documentos recuperados y sus descriptores asociados. El algoritmo selecciona el descriptor d_3

	Descriptores Seleccionables					
Conjunto D	d_1	d_4	d_5	d_6	d_7	d_8
D_9		×				×
D_{10}		×			×	×
$FrecDoc$	0	2	0	0	1	2

Cuadro 6.7: Conjunto de documentos recuperados y sus descriptores asociados. El algoritmo selecciona el descriptor d_4

Documentos-Descriptores	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8
D_1		×						
D_2		×	×					
D_3			×					
D_4			×					
D_5		×						
D_6		×						
D_7		×						
D_8			×	×				
D_9				×				
D_{10}				×				
$FrecDoc$	0	5	4	3	0	0	0	0

Cuadro 6.8: Aspecto final de las relaciones documento-descriptor sobre el conjunto de descriptores obtenido aplicando el algoritmo balanceado

6.5. Proceso de asignación de descriptores

Una vez realizado el proceso de extracción y selección de descriptores, nuestro modelo ya dispone de los conjuntos $Docs$ y $Desc$ sobre los cuales será posible llevar a cabo la construcción del contexto formal $K \equiv (Docs, Desc, I)$. Tal y como fue expuesto en el capítulo de introducción a la teoría del AFC, así como en la propuesta de nuestro modelo, un contexto K se caracteriza no sólo por el conjunto de documentos y descriptores sobre los cuales se genera el retículo, sino también por una relación de incidencia I encargada de definir el modo en el que cada uno de los documentos se relaciona con el conjunto de descriptores extraído. El objetivo de esta sección es el de presentar un conjunto de alternativas para la generación del contexto K y, más concretamente, para la obtención de la relación de incidencia I. A continuación se presentan brevemente nuestras propuestas:

1. *Construcción directa del contexto K.* Consiste en establecer una correspondencia directa entre el conjunto de descriptores seleccionados y el conjunto de documentos de manera que I refleje las relaciones de pertenencia $descriptor - documento$ que explícitamente figuran en la información recuperada.

2. *Construcción del contexto K aplicando Latent Semantic Indexing (LSI).* Esta segunda aproximación es un poco más compleja y pretende extraer nuevas relaciones de pertenencia $descriptor-documento$ con el fin de generar, un conjunto I mucho más rico cuyo objetivo será el de mejorar la población de los clusters generados.

6.5.1. Construcción directa del contexto K

Dados el conjunto de documentos recuperado $Docs = \{doc_1, doc_2, \ldots, doc_n\}$ y el conjunto de descriptores seleccionados $Desc = \{desc_1, desc_2 \ldots, desc_m\}$, la relación de incidencia I se define

como un conjunto de pares $(doc_i, desc_j) \in Docs \times Desc$ que define el conjunto de descriptores contenidos o pertenecientes a cada uno de los documentos recuperado.

En esta primera aproximación a la construcción del contexto K proponemos la obtención directa de la relación I a partir de la representación vectorial del conjunto de documentos recuperado en base a un modelo booleano.

El proceso propuesto para la construcción de I es el siguiente:

- Dados $Docs$ y $Desc$ es posible representar cada documento $doc_i \in Docs$ como un vector cuyas componentes representan los descriptores que éste contiene. Debido a que en esta aproximación es sólo necesario conocer si un descriptor está contenido o no en un documento concreto, una representación booleana de los descriptores en el vector será suficiente. Un documento doc_i quedará descrito por $\vec{doc_i} = \langle valDesc_1, valDesc_2, \ldots valDesc_m \rangle$, donde $valDesc_j$ valdrá *cierto* si el descriptor $desc_j$ figura de manera explícita en el documento doc_i y *falso* en caso contrario.
- Una vez construida la representación vectorial de los documentos, la obtención de I es directa de acuerdo a la siguiente fórmula:

$$(doc_i, desc_j) \in I \Longleftrightarrow \vec{doc_i}[valDesc_j] = cierto \tag{6.14}$$

- Conocida la relación I, el contexto K queda completamente definido, siendo posible realizar la agrupación automática de los documentos en base a la teoría del AFC.

6.5.2. Construcción del contexto K aplicando LSI

Con el fin de mejorar la población de los clusters generados aplicando nuestro modelo, en esta segunda aproximación se propone el uso de Latent Semantic Indexing (LSI) [38] (consultar anexo A) para extraer nuevas relaciones de pertenencia $descriptor - documento$ que no figuren explícitamente en los documentos recuperados. Como resultado de este proceso el contexto K obtenido se verá enriquecido con relaciones que no existían anteriormente.

Al igual que en la aproximación anterior, partiremos de los conjuntos $Docs = \{doc_1, doc_2, \ldots, doc_n\}$ y $Desc = \{desc_1, desc_2 \ldots, desc_m\}$ de documentos y descriptores obtenidos a partir del conjunto de documentos inicial. En este caso, y con el fin de mejorar los resultados finales, utilizaremos un vector de pesos para llevar a cabo la representación de cada documento. Esto es debido a que, según [40], la aplicación de LSI produce mejores resultados si la matriz de documentos se encuentra previamente pesada.

El proceso propuesto para la construcción de I se puede resumir en los siguientes pasos:

- Dados $Docs$ y $Desc$ representaremos cada documento $doc_i \in Docs$ como un vector cuyas componentes reflejen la importancia que tiene cada uno de los descriptores que éste contiene. Esto significa que, a diferencia de la aproximación anteriormente propuesta, en este caso es necesario realizar un pesado de los descriptores que describa la importancia relativa de cada

uno de ellos tanto dentro de un documento como respecto a la colección completa. En este caso, optaremos por aplicar el pesado propuesto en [40], que ha dado buenos resultados sobre sistemas de RI basados en LSI. El proceso de pesado consiste en calcular un peso global para cada uno de los descriptores que refleje la importancia del descriptor con respecto a la colección completa (en nuestro caso el conjunto de documentos recuperado). Calculados todos los pesos globales, es posible calcular un peso local para cada uno de ellos respecto a cada uno de los documentos en los que aparecen que refleje su importancia relativa sin tener en cuenta la colección completa. Finalmente, el peso de un descriptor respecto a un documento concreto se podrá definir como la relación existente entre su su peso local y su importancia con respecto a la colección completa. Las fórmulas 6.15 y 6.16 calculan el peso global de un descriptor i, la fórmula 6.17 su peso local respecto al documento j sin tener en cuenta la colección y, finalmente, la fórmula 6.18 su peso local teniendo en cuenta su importancia global.

$$GlobalWeight_i = \frac{1 + \sum_{\alpha=1}^{n} P_{i\alpha} \log P_{i\alpha}}{\log n} \tag{6.15}$$

$$P_{ij} = \frac{tf_{ij}}{GlobalFrequency_i} \tag{6.16}$$

$$LocalWeight_{ij} = \log\left(tf_{ij} + 1\right) \tag{6.17}$$

$$wd_{ij} = \frac{LocalWeight_{ij}}{GlobalWeight_i} \tag{6.18}$$

Donde tf_{ij} representa la frecuencia de aparición del descriptor i en el documento j, mientras que $GlobalFrequency_i$ es su frecuencia absoluta respecto al conjunto de documentos completo.

La representación vectorial de cada documento j se realizará mediante el peso w_{ij} calculado para cada uno de los descriptores.

- Una vez construida la representación vectorial de los documentos, procederemos a aplicar LSI obteniendo la correspondiente matriz A_k. El valor de k puede decidirse bien experimentalmente o aplicando el teorema de Eckart-Young descrito en el anexo A.1.

- La matriz A_k obtenida se caracteriza porque todos sus elementos han pasado a tener valores distintos de 0, lo que significa que, teóricamente, cualquier documento podría contener cualquiera de los descriptores seleccionados. Sin embargo, observando A_k podemos apreciar que muchos de los valores generados son prácticamente nulos (del orden de 10^{-10}), lo que significa que, a efectos prácticos, podrían considerarse como 0. Teniendo esto en cuenta, la relación de incidencia del contexto K podría generarse a partir de los resultados de A_k del siguiente modo:

$$(doc_i, desc_j) \in I \Longleftrightarrow A_k[desc_j, doc_i] > umbral \tag{6.19}$$

que determina si un documento doc_i contiene al descriptor $desc_j$ en función de su aparición en la matriz A_k con un valor que supere un cierto umbral que deberá determinarse experimentalmente.

- Conocida la relación I, el contexto K queda completamente definido, siendo posible realizar la agrupación automática de los documentos en base a la teoría del AFC.

6.6. Proceso de construcción del retículo

Finalmente, y una vez construido el contexto formal $K \equiv (Docs, Desc, I)$, el último paso que debemos realizar para obtener la agrupación del conjunto de documentos recuperados es construir el retículo asociado. El retículo, tal y como presentamos en el modelo, será la estructura intrínseca para llevar a cabo la estructuración y organización de la información recuperada. Existen diferentes aproximaciones para la construcción de un retículo de manera eficiente. De todas ellas, en este trabajo hemos optado por utilizar la presentada en [85], aunque también pueden tenerse en cuenta otras aproximaciones. [76] presenta un interesante estudio comparativo de los distintos algoritmos desarrollados para llevar a cabo esta tarea.

El algoritmo utilizado en este trabajo se caracteriza por lo siguiente:

- Permite calcular el conjunto completo de todos los conceptos formales $\beta(K)$ derivados de un contexto formal $K \equiv (Docs, Desc, I)$.
- No realiza un cálculo explícito de la estructura del retículo.
- Define un orden total $\prec$ sobre los conceptos calculados tal que $c_1 < c_2 \Longrightarrow c_1 \prec c_2$.
- Su coste asintótico temporal depende del número de conceptos formales generados, del número de documentos del contexto y del número de atributos seleccionados del siguiente modo $\Theta(|\beta(Docs, Desc, I)| \times |Docs|^2 \times |Desc|)$.

La base sobre la cual se apoya este algoritmo es el cálculo del conjunto de vecinos superiores de un concepto formal dado, siendo el coste de este proceso $\Theta(|Docs|^2 \times |Desc|)$ para cada concepto. Dado un concepto $c = (A, B)$, la expresión $(A \cup \{o_n\})''$ define la extensión del superconcepto generado por el objeto o_n. Decimos que $(A \cup \{o_n\})''$ es la extensión de un vecino superior del concepto c si y sólo si se cumple:

$$\big(\forall x \in (A \cup \{o_n\})'' \setminus A\big) \cdot (A \cup \{x\})'' = (A \cup \{o_n\})'' \tag{6.20}$$

De acuerdo a esta idea, la función *vecinos* presenta el modo en el que se calculan todos los vecinos superiores de un concepto c dado.

Para llevar a cabo el cálculo del conjunto completo de conceptos utilizaremos la función *retículo* que se apoya en la función *vecinos* para realizar el cálculo de los vecinos superiores de cada uno de los conceptos formales obtenidos. La función *retículo* realiza el proceso completo de cálculo inicializando el algoritmo sobre el concepto formal $(\emptyset'', \emptyset')$. A partir de este punto extrae los vecinos superiores, identificando aquellos que se encuentran compartidos por varios conceptos mediante un árbol de búsqueda L donde los diferentes conceptos $c = (A, B)$ son almacenados con la clave A y el orden total $\prec$ impuesto sobre éste respeta $c_1 < c_2 \Longrightarrow c_1 \prec c_2$. El algoritmo inserta los conceptos en L y realiza las búsquedas al mismo tiempo. Su coste asociado es de $\Theta(|\beta(Docs, Desc, I)| \times |Docs|^2 \times |Desc|)$.

input : Contexto formal $K \equiv (Docs, Desc, I)$, Concepto formal $c \equiv (A, B)$
output: Conjunto de conceptos vecinos superiores del concepto formal c que recibe como entrada
$min \leftarrow Docs \setminus A$;
$vecinos \leftarrow \emptyset$;
foreach $doc \in Docs \setminus A$ **do**
  $B_1 \leftarrow (A \cup \{doc\})'$;
  $A_1 \leftarrow B_1'$;
  if $(min \cap (A_1 \setminus A \setminus doc)) = \emptyset$ **then**
    $vecinos \leftarrow vecinos \cup \{(A_1, B_1)\}$;
  else
    $min \leftarrow min \setminus \{doc\}$;
  dev vecinos;

Function `vecinos` (*(A,B)* ,(Docs,Desc,I))

Donde $lookup$, $insert$ y $next$ son funciones de acceso y actualización del árbol de búsqueda L, x_* representa el supremo de todos los elementos menores que x y c^* representa el ínfimo de todos los elementos mayores que c. Formalmente, siendo v un elemento de un retículo completo V, v_* y v^* se definen del siguiente modo:

$$v_* \equiv \bigvee \{x \in V \cdot x < v\} \tag{6.21}$$

$$v^* \equiv \bigwedge \{x \in V \cdot v < x\} \tag{6.22}$$

6.7. Proceso de visualización y navegación

En las secciones anteriores hemos descrito detalladamente el proceso para obtener una estructura de clustering sobre un conjunto de documentos recuperados. Tal y como expusimos, la obtención del clustering se basa en un modelo formal cuyo resultado final es un retículo de conceptos. Dado que este tipo de estructuras se caracterizan por ser relativamente complejas y poco conocidas por

```
input  : Contexto formal K ≡ (Docs, Desc, I)
output: Conjunto de conceptos formales β(Docs, Desc, I)
c ← (∅'', ∅');
terminar ← false insert(c,L);
repeat
    foreach x in vecinos(c, (Docs, Desc, I)) do
        x ← lookup(x,L);
        if x=null then
            insert(x,L);
        x* ← x* ∪ {c};
        c* ← c* ∪ {x};
    c ← next(c, L);
    if c=null then
        terminar ← true
until terminar ;
dev L;
```

Function `reticulo` (*Docs,Desc,I*)

parte de los usuarios habituales de los sistemas de búsqueda, consideramos que este trabajo quedaría incompleto sin una propuesta orientada a definir el modo en el que se puede llevar a cabo su visualización, así como un paradigma de navegación para acceder a la información recuperada.

Con este objetivo en esta sección se proponen dos alternativas para la representación del clustering generado y para su posterior navegación por parte del usuario. Debemos remarcar que, aunque ambas propuestas pueden implementarse de manera independiente en sistemas diferentes, éstas pueden combinarse e interactuar en un único sistema, facilitando al usuario la tarea de acceder al conjunto de clusters generado. Brevemente, las propuestas presentadas son las siguientes:

- *Paradigma basado en retículos.* Presenta al usuario una vista simplificada del retículo generado de acuerdo a las reglas para la representación de este tipo de estructuras mediante diagramas de Hasse. Debido a que la comprensión de la información relacionada mediante un retículo puede llegar a convertirse en una tarea excesivamente compleja (cuando el número de conceptos y relaciones obtenidas es muy elevado), nuestra propuesta pretende familiarizar al usuario con este tipo de estructuras pero centrando su atención únicamente en la zona del retículo que éste se encuentre explorando en cada momento y en sus clusters relacionados.

- *Paradigma basado en directorios web.* Nuestra segunda propuesta para la visualización del clustering se basa en un paradigma de visualización y navegación mucho más familiar para los usuarios. La aproximación propuesta pretende transformar el clustering generado en una estructura de directorio similar a la utilizado por Open Directory Project o Yahoo! Directories [94, 140]. Al igual que en la aproximación anterior, en este caso nuestro objetivo es el de conservar la estructura del retículo en la visualización, permitiendo al usuario navegar directamente por el conjunto de clusters relacionados sin necesidad de realizar transformación

alguna sobre ésta.

6.7.1. Paradigma de navegación basado en retículos

Tal y como acabamos de exponer, una de las motivaciones principales de este trabajo es dar al usuario la posibilidad de explotar la riqueza de los retículos a la hora de explorar y relacionar la información recuperada. Desde este punto de vista, la representación clásica de un retículo mediante diagramas de Hasse es la forma más habitual de tener acceso a toda esa información. Sin embargo, una representación de los clusterings obtenidos aplicando nuestra propuesta y basada en diagramas de Hasse puros presenta, bajo nuestro punto de vista, los siguientes inconvenientes:

- Los diagramas de Hasse representan el retículo completo, permitiendo el acceso directo a la totalidad de la estructura generada. Cuando el número de conceptos formales generados es pequeño y existen pocas relaciones entre ellos, los diagramas de Hasse se presentan como una alternativa válida. Sin embargo, cuando el número de conceptos aumenta, la representación generada puede volverse excesivamente compleja haciendo que su utilización requiera de un cierto tiempo para la interpretación y comprensión de sus contenidos.

- Para un usuario medio, el tener que enfrentarse a representaciones excesivamente complejas de la información recuperada puede dificultar la usabilidad de los sistemas basados en nuestro modelo.

Con el fin de aliviar estos problemas, y aunque nuestra propuesta se basa en la metodología de representación de los diagramas de Hasse, nuestro objetivo es reducir la complejidad de las agrupaciones generadas focalizando al usuario únicamente en aquellas partes del retículo que se encuentre explorando en cada momento.

La visualización se llevará a cabo mostrando al usuario los nodos de información obtenidos a partir del retículo de conceptos formales, tal y como expusimos en el capítulo 5. Mientras que en el caso del retículo de conceptos la extensión de cada concepto contiene todos los objetos sobre los cuales éste se aplica, en el caso de los nodos de información cada nodo contendrá en su extensión únicamente aquellos objetos para los cuales éste es concepto objeto. Esta diferencia, aunque sutil, supone que el usuario va a localizar los documentos únicamente en aquellos clusters que los describan completamente, permitiéndole distinguir claramente las características diferenciadoras de un conjunto de documentos que se encuentre ubicado en un área concreta del retículo. La metodología aplicada para materializar este tipo de visualización será la misma que la utilizada para obtener los diagramas de Hasse, debido a que éstos etiquetan los conceptos representados en base a los atributos y objetos respecto a los cuales éstos son concepto atributo y concepto objeto.

Teniendo en cuenta estas consideraciones de visualización preliminares, proponemos un paradigma de navegación que muestra al usuario los clusters, así como sus relaciones, conforme éste vaya interactuando con el sistema en función a sus necesidades de búsqueda. En concreto, nos basamos en las siguientes ideas:

1. *Navegación basada en un refinamiento sucesivo de los nodos de información*. Proponemos un proceso de navegación del usuario basado en un refinamiento sucesivo a partir del nodo de información seleccionado en cada momento. De este modo se pretende que el usuario únicamente tenga acceso a una pequeña parte del retículo relacionada directamente con la zona del mismo que éste está explorando. Para realizar la selección del conjunto de clusters, así como de sus relaciones, que deben mostrarse al usuario en cada momento proponemos una estrategia basada en la definición de filtro principal o up-set de la teoría de conjuntos ordenados. Siendo $(M, \leq)$ un conjunto ordenado, si $a \in M$ y $Q \subseteq M$, entonces los conjuntos:

$$\uparrow a \equiv [a) \equiv \{x \in M | x \geq a\} \tag{6.23}$$

$$\uparrow Q \equiv [Q) \equiv \{x \in M | (\exists y \in Q) x \geq y\} \tag{6.24}$$

reciben el nombre de filtros principales o up-sets de a y Q respectivamente y representan todos los elementos mayores o iguales que a y Q dentro del conjunto M (de manera dual podemos definir el concepto de ideal principal o down-set).

Aplicado a nuestro modelo, siendo c_j el concepto formal actualmente seleccionado perteneciente al retículo L, siendo ni_j su correspondiente nodo de información y siendo N el conjunto de conceptos vecinos inferiores de c_j ($N \equiv \{c \in L | c \prec c_j\}$). Definimos el conjunto de conceptos accesibles desde el concepto c_j como $\uparrow N$. Este conjunto realiza una partición en el retículo que, debido a la relación de orden definida sobre L, cubre un subconjunto de conceptos desde los vecinos inferiores de c_j hasta el top del retículo (considerado el cluster raíz del clustering). Sus principales ventajas son las siguientes:

- La partición generada permite diferenciar de manera muy clara el área de interés para el usuario y el conjunto de clusters que todavía no pueden ser considerados de su interés debido a que, o bien están relacionados con clusters mucho más especializados que los vecinos inferiores de c_j y, por lo tanto, serán visualizados en futuras interacciones, o bien se encuentran en una zona disjunta del subretículo generado por c_j y, por lo tanto, no serán accesibles conforme a la estrategia de exploración actualmente aplicada por el usuario.
- El usuario tiene acceso en todo momento a cualquier elemento de $\uparrow N$. De este modo, el proceso de generalización del cluster actualmente seleccionado queda abierto al usuario, permitiéndole modificar su estrategia de navegación sobre la información relacionada de manera sencilla.
- Debido a que en cada interacción del usuario hay que calcular $\uparrow N$ sobre el concepto que acaba de ser seleccionado, la estructura y el tamaño del conjunto de clusters accesibles irá modificándose y adaptándose a las decisiones de exploración tomadas por éste, lo que le supone un proceso incremental de comprensión de la información organizada mucho más asequible para el usuario que el que supone enfrentarse directamente al retículo completo.

2. *División del espacio de navegación*. Con el fin de focalizar al usuario únicamente sobre aquella parte del clustering que se encuentra explorando en cada momento, proponemos dividir el espacio de navegación en dos áreas bien diferenciadas:

   - *Área de búsqueda*. El objetivo del área de búsqueda es el de representar únicamente aquella parte del retículo que el usuario se encuentra explorando en un instante dado. De este modo se pretende focalizar la atención del usuario únicamente sobre el cluster que éste acaba de seleccionar mostrando únicamente la zona del retículo implicada en su proceso de búsqueda de acuerdo a la estrategia que hemos descrito en los párrafos anteriores. El conjunto de clusters representado en este área se corresponderá con el conjunto $\uparrow N$ asociado al cluster seleccionado en cada momento.
   - *Área de clusters no relacionados*. El hecho de representar en el área de búsqueda únicamente una pequeña parte del retículo no permite que el usuario pueda modificar de manera radical su estrategia de exploración debido a que sólo podrá seleccionar aquellos clusters contenidos en el conjunto de clusters accesibles actualmente desplegado. Con el fin de permitir al usuario la selección de clusters no representados en el área de búsqueda, proponemos la creación de un área de clusters no relacionados donde figuren aquellos clusters más genéricos (vecinos inferiores del cluster raíz) y que no pertenezcan a $\uparrow N$. Formalmente, Siendo c_{top} el concepto raíz de L y ni_{top} su nodo de información asociado, definimos N_{top} como el conjunto de conceptos vecinos inferiores de c_{top}. Debido a que pueden existir elementos en c_{top} pertenecientes a $\uparrow N$ y que, por lo tanto, deben aparecer en el área de búsqueda, representaremos en el área de clusters no relacionados únicamente aquellos conceptos de N_{top} que pertenezcan al conjunto $N_{top} \setminus \uparrow N$. De este modo, el área de clusters no relacionados representa un conjunto de clusters que definen alternativas de exploración a la actualmente en curso, permitiendo al usuario detectar y navegar de manera sencilla hacia dominios disjuntos.

En las figuras 6.9 y 6.8 podemos observar la materialización de esta aproximación sobre la interfaz del sistema JBraindead (uno de los prototipos implementados en esta Tesis Doctoral para demostrar la validez de nuestras propuestas y que será presentado en detalle en el capítulo 8). En la parte izquierda de la interfaz (figura 6.9) se encuentra implementada nuestra propuesta de visualización y navegación basada en retículos, donde pueden diferenciarse claramente las áreas de búsqueda y de clusters no relacionados que acabamos de exponer. La zona superior, más clara, representa el área de clusters no relacionados, mientras que el área inferior, más oscura (con el gradiente de color), representa el área de búsqueda. El ejemplo muestra la visualización obtenida para la consulta *'clustering'*, donde el nodo actualmente explorado es el descrito por el atributo *'hierarchical clustering'*. Como puede observarse, el área de búsqueda no muestra el retículo completo que, en el caso concreto de ésta consulta resulta bastante complejo, sino que únicamente muestra aquella parte (utilizando un método de visualización basado en diagramas de Hasse) relacionada con el cluster actualmente seleccionado. Nótese como el conjunto de vecinos inferiores N abarca tres nuevos clusters que permiten dirigir la búsqueda hacia documentos mucho más específicos descritos por *'clustering software'*+*'based clustering'*, *'data clustering'*+*'data mining'* y *'means clustering'* y relacionados con

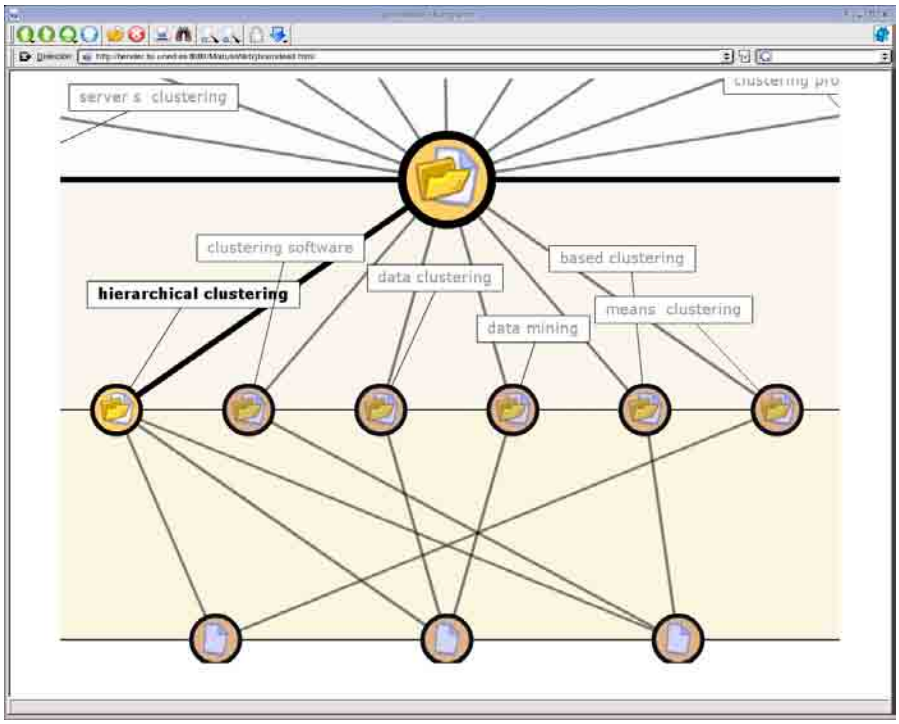


Figura 6.8: Área de búsqueda del sistema JBraindead para la consulta *'clustering'*

el descriptor *hierarchical clustering*. Por otra parte, $\uparrow N$ descubre cinco nuevos clusters mucho más genéricos que, sin estar directamente relacionados con *'hierarchical clustering'*, permiten generalizar los clusters pertenecientes al conjunto N, demostrando que la representación de los clusters en base a una parte del retículo completo permite al usuario acercarse de una manera natural y sencilla a la estructura conceptual sobre la cual se organiza la información recuperada.

Por otra parte, el conjunto de clusters representado en el área de clusters no relacionados da al usuario acceso a un conjunto de clusters que, sin estar directamente relacionados con la exploración realizada hasta el momento, pudieran resultar de su interés en caso de cambiar sus criterios de navegación. En este ejemplo concreto es destacable la aparición del descriptor *'linux clustering'* que permite al usuario acceder a un área del retículo disjunta que contiene información acerca de la realización de clustering de máquinas con sistema operativo Linux.

Desde el punto de vista de la eficiencia, implementar el paradigma propuesto es relativamente sencillo debido a que el retículo completo ha sido calculado con anterioridad y, por lo tanto, únicamente es necesario acceder a la relación de orden definida sobre éste para obtener los conjuntos de conceptos necesarios en cada momento.

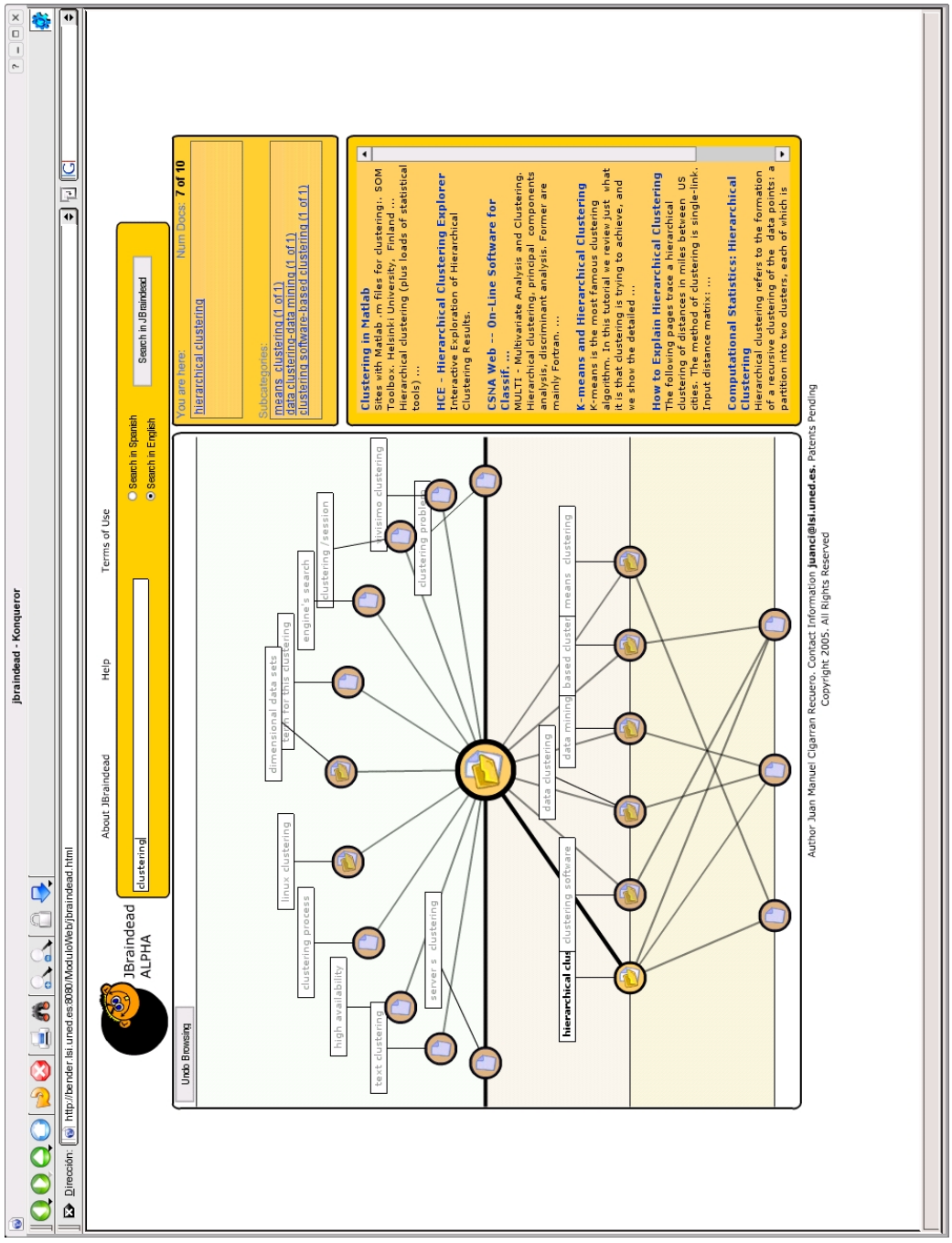


Figura 6.9: Clustering obtenido con el sistema JBraindead para la consulta *'clustering'*

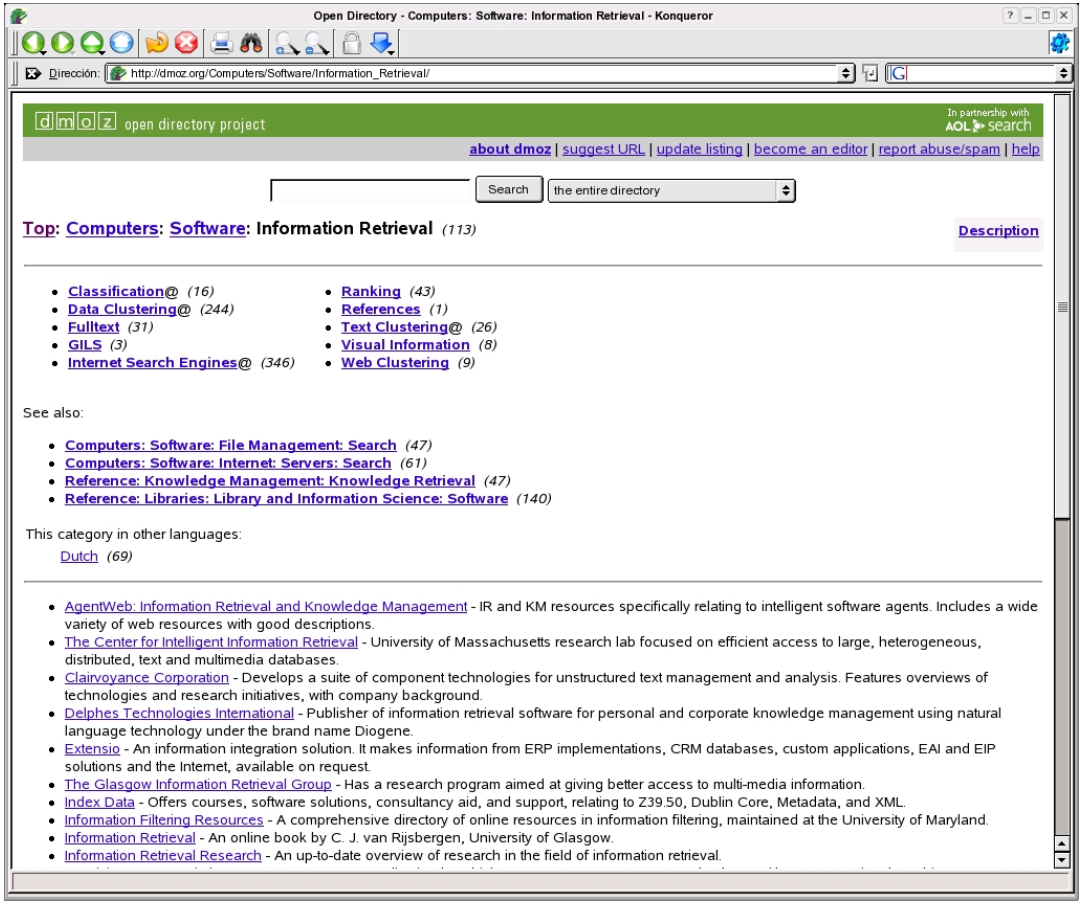


Figura 6.10: Documentos relacionados con la categoría *top - computers - software - information retrieval* en Open Directory Project (ODP)

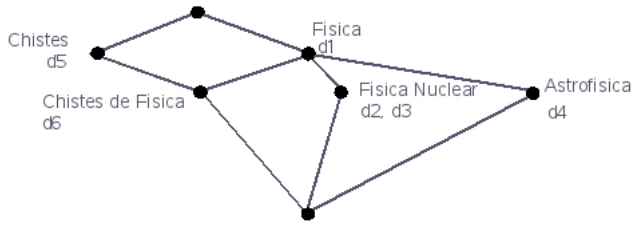


Figura 6.11: Clustering asociado a documentos descritos por los atributos *'Física'*, *'Chistes'*, *'Chistes de Física'*, *'Física Nuclear'* y *'Astrofísica'*

6.7.2. Paradigma basado en Directorios Web

Como complemento para facilitar la comprensión del paradigma de visualización basado en retículos que acabamos de exponer proponemos una aproximación alternativa basada en directorios web. Consideramos que el paradigma de navegación basado en este tipo de directorios resulta mucho más familiar al usuario y, en una primera aproximación para el desarrollo de un sistema basado en nuestro modelo, puede resultarle de gran ayuda para comprender el significado de los retículos mostrados. Además, resulta más manejable cuando el retículo es excesivamente complejo.

Los directorios web [140, 94] se caracterizan por organizar información (páginas web) de acuerdo a una serie de categorías (asignadas manualmente) en base a una relación jerárquica de generalización especialización muy similar a la presentada en este trabajo.

Nuestra aproximación pretende, utilizando una metodología de visualización similar a la aplicada sobre estos directorios, adaptar el paradigma de navegación que hemos propuesto en el apartado anterior con el fin de explotar las características estructurales de los retículos sobre los directorios web. Un directorio de este tipo apoya su paradigma de navegación sobre un sistema de visualización extremadamente simple que muestra las categorías o descriptores del nodo actualmente seleccionado (atributos seleccionados), así como una lista con un conjunto de descriptores que permiten llevar a cabo su especialización (atributos seleccionables). El paradigma de navegación se basa en que, en cualquier momento, el usuario puede especializar el nodo actualmente explorado mediante la selección de cualquiera de los atributos seleccionables, o puede realizar su generalización mediante alguno de los elementos del conjunto de atributos seleccionados mostrados por el directorio para dicho nodo. La figura 6.10 muestra un ejemplo del directorio web Open Directory Project (ODP). Como puede apreciarse, el nodo actualmente seleccionado hace referencia a la categoría *'Information Retrieval'* y es visualizado mostrando el conjunto de descriptores heredados desde el nodo de raíz de la jerarquía hasta éste (*top:computers:software:information retrieval*). Así mismo, la interfaz también muestra el conjunto de descriptores que es posible seleccionar para especializar la categoría que actualmente está explorando el usuario (con descriptores como *'classification'*, *'data clustering'*, *'ranking'*, etc.). La interacción del usuario con la información organizada se produce, por lo tanto, o bien seleccionando sobre el conjunto de descriptores propuestos como especialización del nodo actual o bien seleccionando alguno de los atributos utilizados para describirlo. Nuestra aproximación pretende adaptar este paradigma de navegación y visualización al modelo propuesto.

Sin embargo, el hecho de que un directorio concreto pueda tener varios padres produce dificultades en sitios como Yahoo! o ODP, y a veces se confunde el camino *canónico* que define el directorio con el camino por el que se ha llegado a él. En este trabajo proponemos una visualización alternativa. En los retículos, un cluster puede ser generado a partir de un conjunto de nodos no comparables entre sí (la relación $\leq$ no está definida entre ellos). Por ejemplo, de acuerdo al retículo de la figura 6.11, el nodo descrito por *'Física'*, *'Chistes'* y *'Chistes de Física'* hereda los descriptores *'Física'* y *'Chistes'* de dos clusters no comparables. Esto supone que el acceso a dicho nodo desde la raíz del clustering puede producirse por dos caminos diferentes (desde el nodo *'Chistes'* o desde el nodo *'Física'*), que generarán dos posibles secuencias de descriptores igualmente válidas para llevar a cabo un proceso de generalización posterior. De hecho, ambas secuencias permitirán, dependiendo de la estrategia de exploración utilizada por el usuario, acceder a zonas del retículo que, a ese nivel de descripción, podrían considerarse disjuntas.

Nuestra opción es no presentar al usuario la totalidad de las secuencias de descriptores validas para cada uno de los nodos (puesto que en algunos casos podrían ser muchas) sino que, en su lugar, utilizaremos el conjunto completo de atributos que lo describen. Esto hace que para llevar a cabo una generalización del nodo actual, el usuario deba realizar una selección sobre su conjunto de descriptores, considerados estos como entidades independientes. Formalmente, siendo $c_j = (A_j, B_j)$ el nodo actual seleccionado y ni_j su correspondiente nodo de información, definimos el conjunto S de atributos que lo describen, y que por lo tanto el usuario podrá seleccionar para llevar a cabo su generalización, como $S = B_j$. Es decir, el conjunto de atributos S coincide con la intensión del nodo actual que se encuentra explorando el usuario.

Por otra parte, para la construcción del conjunto de descriptores seleccionables que permitirán especializar el nodo actual optamos por trabajar con los descriptores asociados al conjunto de vecinos inferiores N del nodo actualente seleccionado. En concreto, de la intensión de cada uno de los elementos de N, proponemos mostrar únicamente aquellos atributos que no figuren en la intensión de cualquier concepto perteneciente al conjunto $\uparrow N \setminus N$. Formalmente, siendo $c_j = (A_j, B_j)$ el nodo actual seleccionado y ni_j su correspondiente nodo de información, y siendo $N = \{c_1, c_2, \ldots, c_k\}$ el conjunto de vecinos inferiores de c_j, definimos el conjunto de descriptores o categorías seleccionables CS de acuerdo a la siguiente fórmula:

$$CS = \bigcup_{i=1}^{k} B_i \setminus B_j \tag{6.25}$$

Las figuras 6.13 y 6.12 materializan esta aproximación sobre el sistema JBraindead (presentado en detalle en el capítulo 8) para la consulta *'Madonna'* La parte superior derecha de la interfaz (figura 6.13) muestra la aproximación propuesta basada en directorios web. Como puede observarse, este área se encuentra dividida en dos partes que muestran, para el nodo actual, el conjunto de descriptores seleccionados (cuadro etiquetado con la frase *'You are here'*) y el conjunto de descriptores seleccionables respectivamente (cuadro etiquetado con el término *'Subcategories'*). Tal y como acabamos de exponer, el conjunto de descriptores seleccionados representa la intensión del cluster que actualmente está explorando el usuario. Nótese como cada uno de los descriptores se muestran independientemente debido a las razones expuestas permitiendo al usuario seleccionar cualquiera de

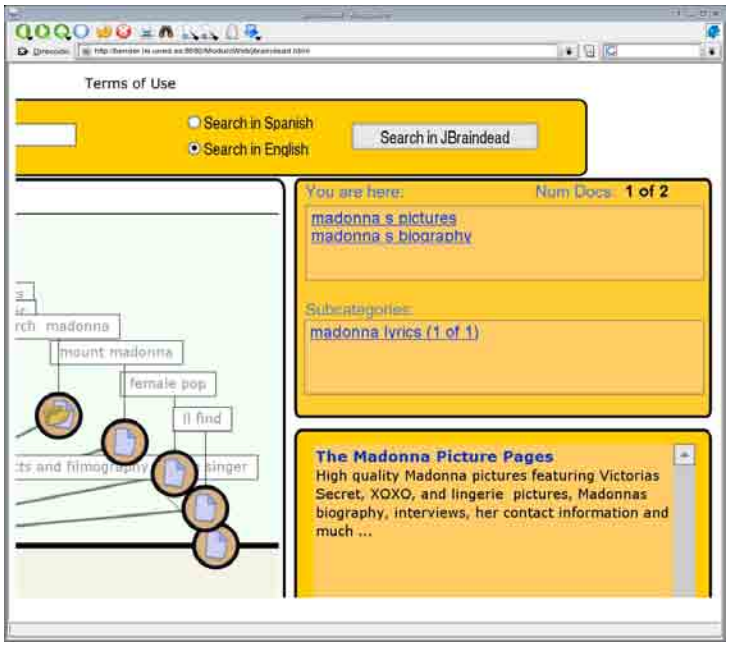


Figura 6.12: Area de navegación basada en directorios del sistema JBraindead para la consulta *'Madonna'*

ellos con el fin de llevar a cabo un proceso de generalización. En nuestro caso concreto, el nodo actual se encuentra descrito por los atributos *'Madonna's pictures'* y *'Madonna's bibliography'*, que hereda de dos clusters no comparables. Esta es la razón por la cual antes apuntábamos que es posible acceder a un mismo cluster desde diferentes caminos y que, por lo tanto, no es posible describir mediante una única secuencia de descriptores el camino desde el cluster raíz hasta el cluster actualmente seleccionado. Por otra parte, el conjunto de categorías seleccionables viene descrito, en este caso, por un único atributo (*'Madonna's lyrics'*) que permite especializar el nodo explorado hacia un cluster mucho más específico, con documentos que, además de incluir fotografías e información acerca de la bibliografía de Madonna, permiten obtener las letras de sus canciones. Nótese como este conjunto se ha obtenido a partir del conjunto de vecinos inferiores N que, en esta ocasión, está formado únicamente por un cluster.

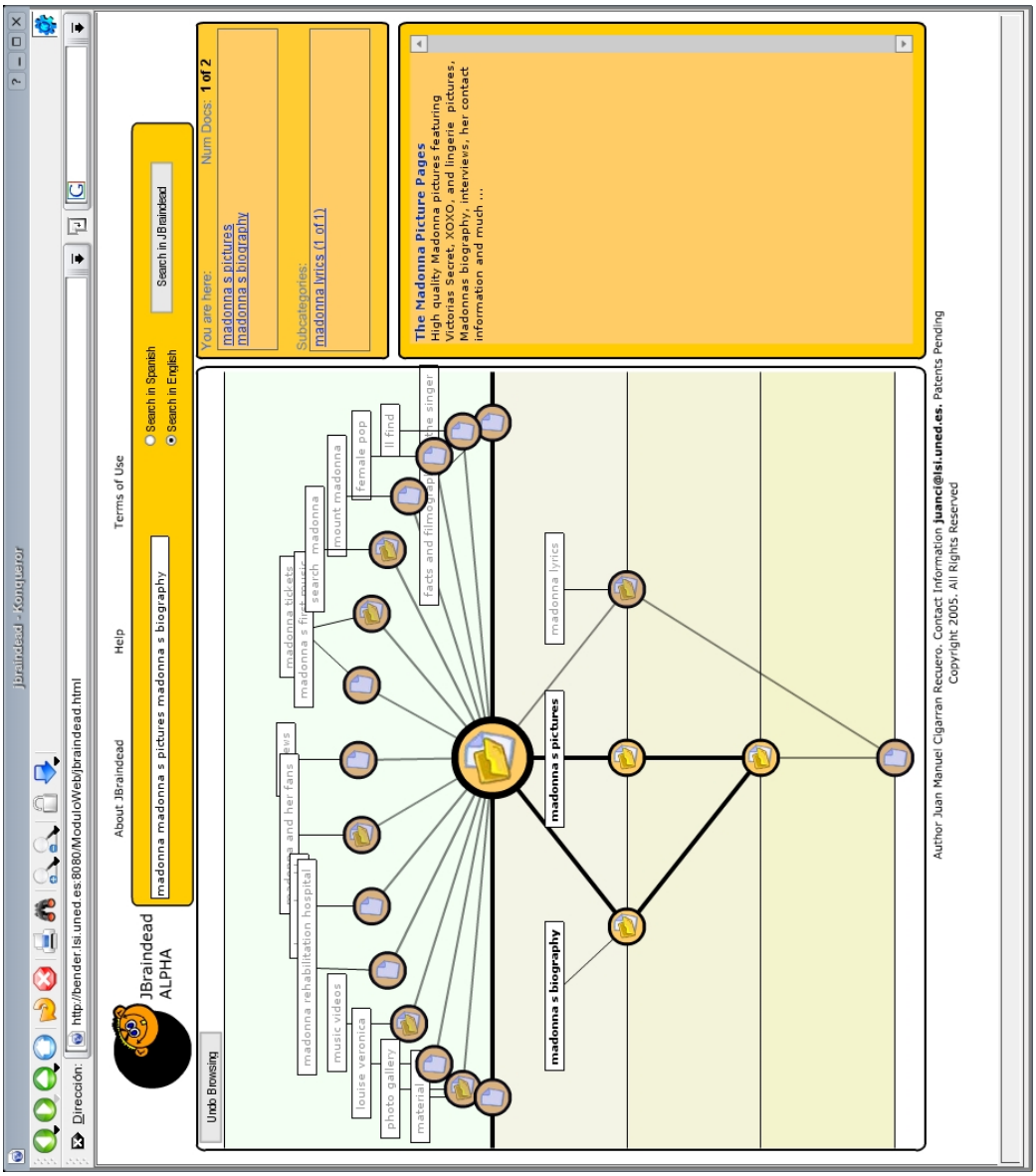


Figura 6.13: Clustering obtenido con el sistema JBraindead para la consulta *'Madonna'*

6.8. Recapitulación

En este capítulo hemos expuesto una metodología para la construcción de un sistema de clustering basado en el modelo presentado en el capítulo 5. En concreto, hemos descrito cada uno de los procesos que un sistema de este tipo debe resolver, haciendo especial hincapié en los procesos de extracción y selección de descriptores, así como en los procesos de construcción del contexto formal. Hemos presentado un conjunto de alternativas de visualización novedosas para mejorar la representación del clustering obtenido, así como la interacción del usuario. Finalmente, debemos destacar que nuestra metodología incluye que comparemos empíricamente distintas aproximaciones para cada uno de los procesos involucrados en un sistema de clustering. En el siguiente capítulo presentaremos el conjunto de medidas necesario para evaluar un sistema construido a partir de esta propuesta.

Capítulo 7

Propuesta de una Medida de Evaluación Basada en el Coste Cognitivo

Aunque el clustering de resultados de búsqueda es un área de gran actividad (tanto comercial como de investigación), todavía no existe consenso acerca de cómo realizar evaluaciones y de cómo comparar de forma sistemática distintas aproximaciones.

A diferencia del área de la recuperación de información, donde existen medidas clásicas (como la precisión y la cobertura) ampliamente aceptadas y que permiten medir automáticamente la efectividad de los sistemas, en el área del clustering de documentos existen diferentes propuestas que, bajo nuestro punto de vista, no reflejan adecuadamente la calidad de un sistema de clustering orientado a la tarea que proponemos en esta Tesis Doctoral.

La prueba última de la validez de este tipo de sistemas es la realización de test con usuarios reales. No obstante, optar por evaluaciones de este tipo cuando el sistema se encuentra en una fase de desarrollo no es una buena opción debido a la gran inversión en tiempo y dinero que éstas requieren. En su lugar, la posibilidad de aplicar un marco de evaluación automático permitiría acelerar el proceso de construcción de este tipo de sistemas, facilitando la toma de decisiones a la hora de optar entre diferentes estrategias de clustering posibles.

En general, las medidas de evaluación aplicables al clustering no fueron concebidas para la evaluación de resultados de búsqueda ni consideran la interacción con el usuario (consultar sección 2.5). Por lo tanto, su aplicación en nuestro contexto no permite evaluar de manera precisa el grado en el que nuestra propuesta mejora el proceso de acceso a la información. Es por esta razón por la que en esta Tesis Doctoral hemos optado por desarrollar un marco de evaluación propio capaz de cubrir las expectativas de calidad expuestas en la sección 5.1 y adaptado a las características específicas de nuestro modelo.

En este capítulo presentaremos nuevas medidas de evaluación que no sólo tienen en cuenta la capacidad de las estructuras de clustering para separar correctamente la información relevante sino que también consideran el coste cognitivo asociado a su exploración. Este último punto resulta relevante puesto que un clustering que separe correctamente la información relevante puede obligar al usuario

a recorrer previamente un gran número de clusters con información no relevante.

Comenzamos exponiendo los aspectos a tener en cuenta para poder realizar la evaluación del modelo propuesto, presentando los problemas que plantean otros tipos de evaluación. A continuación presentaremos los objetivos del marco de evaluación propuesto. En la siguiente sección introduciremos el concepto de área de navegación mínima, sobre el cual se fundamenta el cálculo de las medidas de evaluación propuestas, exponiendo detalladamente la manera en que se puede realizar su cálculo para, a continuación, presentar las dos medidas de evaluación desarrolladas en esta Tesis Doctoral, el *factor de destilación* (DF) y la *calidad del clustering* (CQ) (basadas en la noción de coste cognitivo), que serán explicadas y justificadas en detalle. Finalmente expondremos las conclusiones extraídas de las medidas de evaluación propuestas.

7.1. Consideraciones para la evaluación del modelo

El objetivo del modelo de clustering que hemos propuesto es el de facilitar y mejorar el acceso a la información y, por lo tanto, cualquier proceso de evaluación automática susceptible de ser considerado deberá estar claramente orientado a esta tarea. Esto supone que deberá cuantificar el grado en el que esta mejora en el acceso a la información se lleva a cabo en comparación con un baseline que, en nuestro caso, será la lista de documentos inicialmente devuelta por un motor de búsqueda.

La evaluación de la calidad de un clustering debe considerar un conjunto de factores que, directa o indirectamente, influirán en la calidad final del proceso. Estos factores se encuentran relacionados con el grado en el que la estructura de clustering es capaz de aislar correctamente la información relevante, así como de generar una estructura adecuada que permita acceder a esta información minimizando la cantidad de documentos no relevantes que el usuario se ve forzado a visitar. Estos factores fueron expuestos y desarrollados en la sección 5.1.

Dado que estos factores dependen de la estructura intrínseca del clustering resultan muy adecuados para ser evaluados mediante un proceso automático. Esto supone que los valores obtenidos en un proceso de evaluación de este tipo proporcionarán como resultado un valor máximo o una cota superior para la calidad del sistema que puede disminuir cuando el sistema se someta a una evaluación con usuarios reales. Esto es debido a que en una evaluación con usuario reales no sólo influye la capacidad del sistema para organizar correctamente la información, sino también la calidad de los descriptores generados, así como la capacidad del usuario para interpretarlos correctamente a lo largo de su proceso de navegación.

Tal y como expusimos en la sección 2.5, muchas de las métricas definidas en el área del clustering pueden considerarse de propósito general, y evalúan la calidad del mismo en función únicamente de su capacidad para agrupar correctamente la información sobre un conjunto de clases previamente definidas. Este tipo de medidas no tienen en cuenta el propósito para el cual el clustering es generado y, por lo tanto, no permiten obtener información acerca del grado de mejora que supone aplicar una aproximación de este tipo a una tarea concreta. En este sentido, medidas como la pureza o la pureza inversa, para la evaluación de clustering sobre la tarea propuesta no resultan adecuadas.

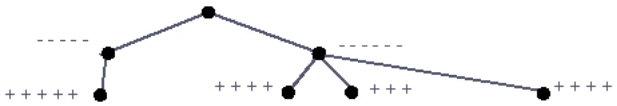


Figura 7.1: Clustering con buenos valores de pureza y pureza inversa pero poco adecuado para una tarea de recuperación de información

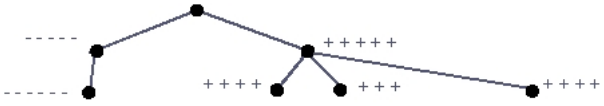


Figura 7.2: Clustering con buenos valores de pureza y pureza inversa adecuado para una tarea de recuperación de información

La figura 7.1 ejemplifica de manera clara esta situación. Como puede observarse, los documentos se encuentran distribuídos en el clustering de manera homogénea, de forma que cada cluster sólo contiene o bien información relevante o bien información no relevante, siendo la población de cada cluster razonable. Esto supone unos buenos valores tanto para la pureza como para la pureza inversa que, en ningún caso, reflejan la adecuación real del clustering a la tarea de recuperación de información. Aunque los documentos se encuentran correctamente agrupados, la propia estructura del clustering obliga al usuario a recorrer todos los clusters, y como consecuencia a consultar todos los documentos recuperados, para poder acceder a la información relevante.

En contraste, la figura 7.2 presenta una estructura de clustering alternativa a la que acabamos de exponer. En ambos casos el valor para las medidas de pureza y pureza inversa es el mismo que para la figura anterior, debido a que en este segundo ejemplo no varía el modo en el que la información se encuentra agrupada. Sin embargo, en este caso ha variado el modo en el que se relacionan los clusters, haciendo que esta estructura sea mucho más adecuada a nuestros propósitos desde el punto de vista de la tarea. Nótese como en este caso el usuario podría acceder a toda la información relevante sin necesidad de examinar ninguno de los documentos no relevantes.

Aún considerando este aspecto, no reflejamos el esfuerzo o coste cognitivo asociado al proceso de navegación que permitirá finalizar la tarea con éxito ni el número de clusters que deben atravesarse. En este sentido, consideramos imprescindible que cualquier métrica orientada a evaluar nuestro modelo, además de estar orientada a la tarea, considere los siguientes aspectos:

- *Número de clusters atravesados*. De acuerdo a lo que acabamos de exponer, resulta crítico considerar el número de clusters que el usuario debe atravesar para acceder a toda la información relevante, siendo este un factor relevante para determinar correctamente la calidad final del clustering. En este sentido, un clustering bien organizado pero que implique la exploración completa del mismo para acceder a la información relevante debería tener asociada una calidad baja.

- *Coste cognitivo*. Aunque la consideración del número de clusters permite reflejar en las métricas de evaluación un nuevo factor que determine la adecuación de nuestra propuesta a la tarea, debemos advertir que no resulta suficiente para determinar la calidad del clustering. Consideramos que en el proceso de navegación y exploración pueden distinguirse dos tareas bien diferenciadas: a) la tarea de seleccionar un cluster concreto, bien para inspeccionar sus documentos o bien para continuar su exploración desde ese nodo, lo que implica discernir acerca de la posible relevancia de sus contenidos y; b) la tarea de inspeccionar el contenido de un cluster concreto en busca de información relevante. Consideramos que el esfuerzo requerido para llevar a cabo ambas tareas, aunque depende directamente del modo en el que el sistema muestra la información al usuario, debe estar reflejado en la métrica de evaluación.

De acuerdo al conjunto de factores presentado para llevar a cabo el proceso de evaluación, las aproximaciones propuestas por [78, 79] y [75] (expuestas en la sección 2.5) podrían ser las más adecuadas para realizar la evaluación del modelo de clustering propuesto en esta Tesis Doctoral. Sin embargo, en este trabajo hemos optado por diseñar nuestras propias aproximaciones en lugar de realizar la adaptación de éstas medidas a nuestro modelo. Esto es debido principalmente a las siguientes razones:

1. Consideramos que el proceso de evaluación debe plantearse desde un punto de vista global, es decir, considerando el acceso a la totalidad de la información relevante en la tarea de recuperación. En este sentido, las aproximaciones [78, 79] y [75] plantean la evaluación considerando el acceso individualizado a cada uno de los documentos relevantes por separado, lo que supone considerar la evaluación de múltiples tareas de recuperación cuyo promedio representa la calidad global del clustering. En nuestro caso pretendemos evaluar la calidad del clustering para facilitar el acceso a todos los documentos relevantes que el sistema ha recuperado. No parece adecuado enfocar el proceso de evaluación suponiendo que el usuario *regresa* al cluster raíz cada vez que encuentra un documento relevante.

2. Las propuestas de [78, 79] y [75] evalúan los resultados de un clustering jerárquico. En contraste, la estructura sobre la que se apoya nuestro modelo es un retículo. Este tipo de estructuras posibilitan la existencia de diferentes alternativas para acceder a cada uno de los clusters, lo que en términos de evaluación se traduciría en diferentes resultados para la evaluación de un mismo clustering. Por lo tanto, resulta crítico dotar a nuestro marco de evaluación de mecanismos capaces de extraer del retículo el conjunto de caminos óptimos a la información relevante, con el fin de obtener unos resultados de evaluación óptimos.

De acuerdo a los aspectos relacionados con la evaluación que acabamos de exponer, a continuación presentaremos el conjunto de métricas desarrollado para evaluar el modelo de clustering propuesto. Este marco de evaluación se presenta como una propuesta de evaluación automática cuyo único requisito es el de disponer de juicios de relevancia para la colección sobre la cual se lleva a cabo el proceso de evaluación.

En la siguiente sección presentamos el concepto de *área de navegación mínima* (MBA), sobre el cual apoyaremos el desarrollo y justificación de las dos medidas propuestas en esta Tesis Doctoral.

7.2. Área de navegación mínima

El hecho de trabajar con un retículo como estructura subyacente añade una cierta dificultad al modo en el que se determina cual será el camino apropiado para acceder a cada uno de los clusters relevantes. De hecho, la posibilidad de considerar diferentes alternativas conduciría hacia distintas valoraciones de la calidad del clustering que no permitirían definir un marco de evaluación fiable y fácilmente interpretable.

Las figuras 7.3 y 7.4 ejemplifican esta situación. En ellas puede apreciarse como es posible acceder al cluster con la información relevante atravesando dos itinerarios diferentes. La primera de ellas muestra un camino que implica atravesar tres clusters con información no relevante, mientras que la segunda de ellas permite llegar al mismo nodo atravesando únicamente un cluster con información no relevante. Resulta obvio que, desde el punto de vista de una tarea de recuperación de información, la situación planteada en la segunda de las figuras es la más adecuada debido a que minimiza el número de documentos no relevantes visitados por el usuario.

En este sentido, el objetivo de esta sección es el de presentar una aproximación para la obtención de los caminos óptimos orientada al proceso de evaluación. Debemos remarcar que, tal y como expusimos en la sección 7.1, en nuestro caso se pretende realizar una evaluación de la tarea considerando que el usuario accederá a todos los documentos relevantes, por lo que no deberían considerarse los itinerarios a cada uno de los clusters relevantes de manera independiente, sino como un todo dentro del proceso de evaluación. En este sentido, en esta Tesis Doctoral presentamos el concepto de *área de navegación mínima* o *minimal browsing area* (MBA), cuyo objetivo será el de obtener el área del retículo que el usuario debe atravesar para acceder a toda la información relevante recuperada minimizando el número de documentos no relevantes visualizados.

De este modo, el área de navegación mínima contendrá todos los clusters con información relevante y aquellos clusters con información no relevante que deben ser obligatoriamente atravesados para poder acceder a la información relevante. Formalmente, siendo L un retículo (con sus correspondientes nodos de información asociados) y $L_{rel} \subseteq L$ el subconjunto de nodos de información relevantes[1], el usuario deberá acceder como mínimo a todos los elementos de L_{rel} con el fin de acceder a todos los documentos relevantes recuperados, lo que significa que $L_{rel} \subseteq MBA$. Sobre los ejemplos que acabamos de presentar, la figura 7.4 mostraría el área mínima de navegación correspondiente al clustering propuesto.

7.3. Cálculo del área de navegación mínima

La obtención del área mínima de navegación puede hacerse a través de la transformación del retículo L inicial (y sus nodos de información) en un nuevo grafo no dirigido caracterizado porque:

- *Sus nodos sólo contienen nodos de información relevantes*. Es decir, el grafo obtenido únicamente representa los clusters con información relevante del retículo original L.

[1] Definimos un nodo de información como relevante cuando éste contiene, al menos, un documento relevante en su extensión

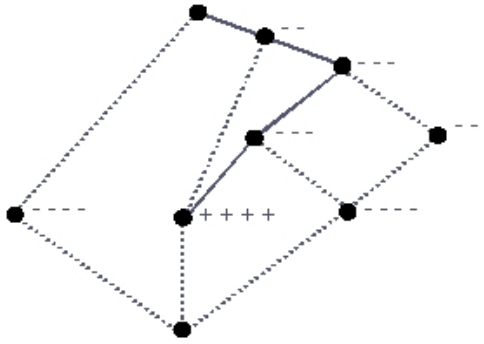


Figura 7.3: Posible itinerario para acceder a todos los documentos relevantes

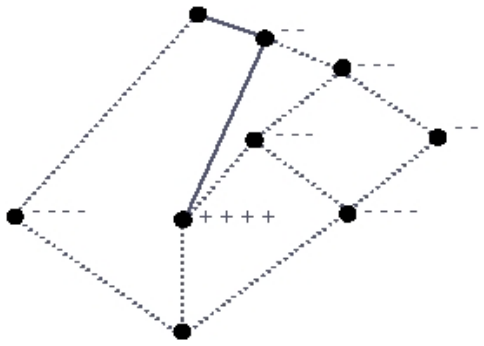


Figura 7.4: Itinerario óptimo para acceder a todos los documentos relevantes

- *Sus enlaces están pesados*. Su peso representará la cantidad de información irrelevante a la que el usuario accede al atravesar el enlace.

Una vez obtenido el grafo, el cálculo del área mínima de navegación se puede llevar a cabo de manera sencilla obteniendo el árbol de recubrimiento mínimo sobre dicho grafo.

El algoritmo para la obtención del grafo asociado se basa en la siguiente definición de coste asociado a un enlace. Dado un retículo L y sus correspondientes nodos de información, el coste asociado a pasar desde el nodo de información ni_i al nodo de información ni_j, tal que $ni_i \succ ni_j$, se define como el número de documentos no relevantes contenidos en ni_j. Por ejemplo, siendo ni un nodo de información tal que $AI = \{d_1, d_2, d_3\}$, donde d_1 y d_2 son documentos irrelevantes, el coste asociado a ir desde cualquier nodo de información vecino superior a ni tendría un coste de 2. A continuación presentamos el algoritmo completo:

1. Dado el retículo original L procedemos a pesar todos sus enlaces de acuerdo a la definición de coste que acabamos de presentar.

2. En un proceso iterativo, se van suprimiendo de L todos los nodos de información que no sean relevantes (es decir, que no contengan ningún documento relevante), seleccionando en cada iteración el nodo de información j más cercano a la raíz de L. El proceso de eliminación deberá mantener la relaciones existentes entre el conjunto de nodos antecesores y predecesores de j. Para ello sobre los conjuntos U_j y L_j, de vecinos superiores e inferiores respectivamente, se crea un nuevo enlace para cada par de nodos u y l tales que $(u, l) \in U_j \times L_j$. Así mismo, los enlaces creados se pesan asignándoles un coste definido de la siguiente manera:

$$coste(u, l) = coste(u, j) + coste(j, l) \tag{7.1}$$

   En caso de existir más de un enlace para un par de nodos concreto (u, l), para el cálculo del coste se seleccionará el enlace con menor coste y se suprimirán los otros. Igualmente, en caso de existir más de un arco con coste mínimo se seleccionará aquel que acumule menos nodos originales.

3. El resultado de la iteración descrita en el punto anterior es un grafo no dirigido conexo cuyos nodos son todos nodos de información relevantes. El área de navegación mínima se obtendrá entonces construyendo el árbol de recubrimiento mínimo sobre el grafo obtenido.

Las figuras 7.5 a 7.12 muestran el proceso descrito para obtener el área de navegación mínima sobre un ejemplo. Inicialmente (figura 7.5) partimos de un retículo L donde existen cuatro nodos de información relevantes (nodos b, f, g y h). Conforme al primer paso del algoritmo, se etiquetan los enlaces entre nodos de acuerdo a la definición de coste presentada (figura 7.6). Los enlaces no etiquetados tienen un coste cero, suprimido con el objeto de clarificar las figuras. En la figura 7.7 se procede a eliminar el primero de los nodos no relevantes, nodo d, para ello se suprime el nodo y se conservan las conexiones entre el conjunto de vecinos superiores y de vecinos inferiores, en este caso los nodos a e i. En las figuras 7.8, 7.9 y 7.10 se procede de igual modo, eliminando los

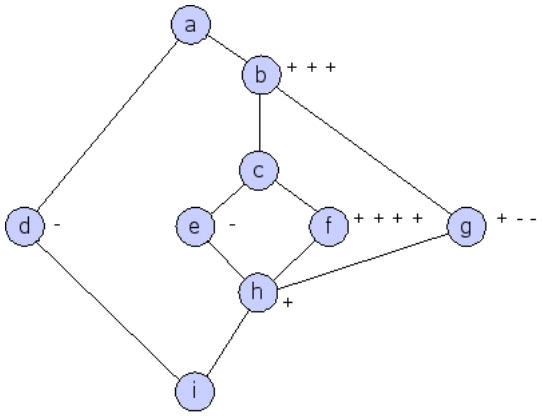


Figura 7.5: Retículo L inicial sobre el que aplicamos el algoritmo de cálculo del MBA

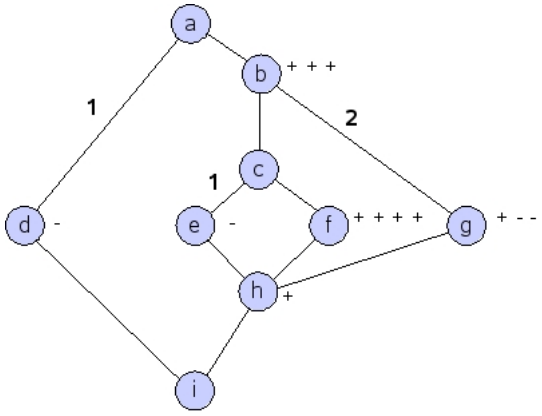


Figura 7.6: Asignación de pesos a los enlaces de L

nodos de información no relevantes c, e e i respectivamente. Nótese como la eliminación del nodo i implica eliminar el enlace con coste uno que lo unía con el nodo a, esto es debido a la no existencia de más vecinos inferiores. La estructura obtenida en la figura 7.10 representa el grafo asociado al retículo asociado L y describe el coste de los distintos caminos que permiten acceder a toda la información relevante agrupada. En la figura 7.11 se muestra el árbol de recubrimiento mínimo del grafo asociado, que refleja el área de coste mínimo (y como consecuencia con menor grado de acceso a información no relevante) sobre el grafo. Finalmente, la figura 7.12 representa la aplicación del árbol de recubrimiento mínimo sobre el retículo L y representa su área de navegación mínima. Como puede apreciarse en esta última figura, el área de navegación mínima cubre la totalidad de la información relevante, minimizando el acceso la información no relevante.

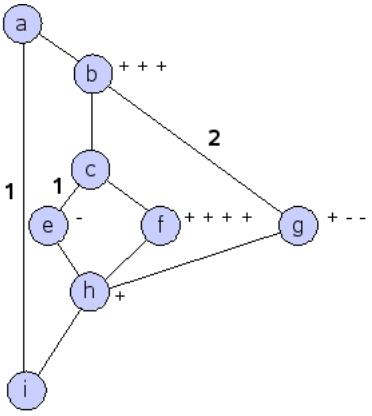


Figura 7.7: Primera iteración, se elimina el nodo d

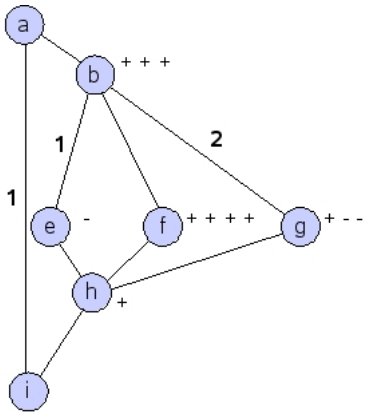


Figura 7.8: Segunda iteración, se elimina el nodo c

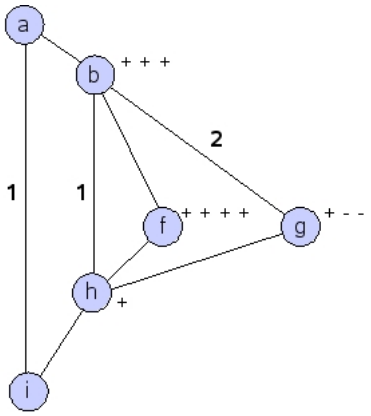


Figura 7.9: Tercera iteración, se elimina el nodo e

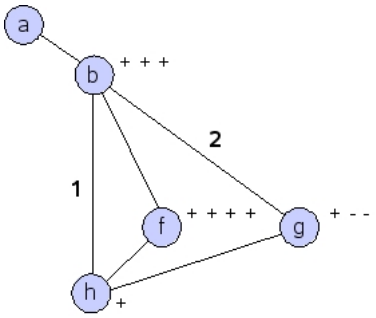


Figura 7.10: Cuarta iteración, se elimina el nodo i

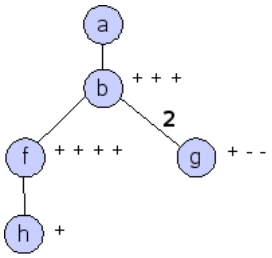


Figura 7.11: Árbol de recubrimiento mínimo correspondiente al grafo de la figura 7.10

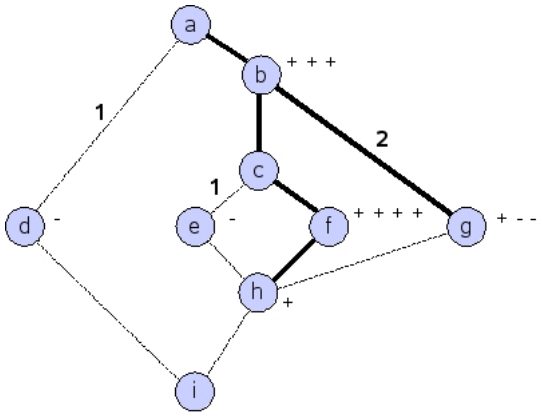


Figura 7.12: Area de navegación mínima (MBA) correspondiente al retículo L

7.4. Medidas de evaluación

El objetivo principal de las medidas de evaluación presentadas en esta Tesis Doctoral es el de cuantificar el grado de mejora en el proceso de recuperación del clustering obtenido respecto la lista de documentos devuelta por un motor de búsqueda. En este sentido, las medidas que vamos a proponer a lo largo de esta sección tomarán como base el conjunto $DOCS$ de los n documentos recuperados inicialmente y el clustering (retículo) L obtenido aplicando nuestro modelo.

En una primera aproximación, proponemos evaluar la calidad del retículo L como la proporción entre el coste cognitivo asociado a encontrar la información relevante en $DOCS$ y el coste cognitivo asociado a encontrar la información relevante en L.

$$Calidad(L) \equiv \frac{\text{coste cognitivo}(DOCS)}{\text{coste cognitivo}(L)} \tag{7.2}$$

El coste cognitivo asociado a encontrar todos los documentos relevantes en $DOCS$ será directamente proporcional al número de documentos, n, recuperados. Sin embargo, la estimación del esfuerzo requerido para navegar por el retículo no es una trivial y supone proponer alternativas que nos permitan atacar el problema teniendo en cuenta las consideraciones propuestas en la sección 7.1.

Aquí presentamos dos aproximaciones que permitirán abordar el problema de evaluación propuesto. La primera de ellas únicamente considera el coste cognitivo de examinar los documentos, mientras que la segunda tiene también en cuenta el coste cognitivo asociado a tomar decisiones de navegación, es decir, a examinar las descripciones de los nodos. Con el fin de obtener el grado óptimo de mejora, ambas están basadas en el concepto de área de navegación mínima presentado en la sección anterior.

7.4.1. Factor de destilación

Sea k_d el coste cognitivo promedio asociado a examinar un documento, es decir, el esfuerzo que debe realizar un usuario para relacionar la representación que el sistema hace del documento con su posible relevancia. De este modo, el coste cognitivo asociado a explorar el retículo se puede expresar como el coste cognitivo asociado a explorar todos los documentos contenidos en el área de navegación mínima del siguiente modo:

$$\text{coste cognitivo}(L) = \text{coste cognitivo}(MBA) = k_d|DOCS_{MBA}| \tag{7.3}$$

Donde $DOCS_{MBA} = \bigcup_j AI_j$, tal que AI representa la extensión de todos los nodos de información pertenecientes en el área de navegación mínima.

De igual manera, podemos expresar el coste cognitivo asociado a explorar la lista completa de documentos como un factor proporcional al número de documentos recuperado, es decir, coste cognitivo$(DOCS) = k_d|DOCS|$. De esta manera, la calidad del clustering, que en este caso denominaremos factor de destilación, puede expresarse a partir de la ecuación 7.2 de la siguiente manera:

$$Calidad(L) \equiv DF(L) = \frac{k_d|DOCS|}{k_d|DOCS_{MBA}|} = \frac{|DOCS|}{|DOCS_{MBA}|} \tag{7.4}$$

La medida DF recibe su nombre debido a que describe la habilidad del clustering para *destilar* o filtrar los documentos relevantes, de modo que se minimice el esfuerzo realizado por el usuario para acceder a ellos. Esta habilidad se expresa como un factor de mejora, en el rango $[1, +\infty)$, con respecto a la lista de documentos original. Un valor de uno para el factor de destilación significaría que el clustering no aporta ventajas frente al uso de un motor de búsqueda tradicional debido a que el proceso de exploración del clustering implicaría acceder a todos los documentos recuperados.

La figura 7.13 ilustra un ejemplo donde puede verse la utilidad de la medida que acabamos de definir. Suponemos que inicialmente se han recuperado siete documentos, de los cuales cuatro de ellos son considerados relevantes a las necesidades del usuario. La figura muestra en trazo continuo el área de navegación mínima, donde puede observarse cómo es posible acceder a toda la información relevante considerando únicamente un documento no relevante. Conocidos el número de documentos de la lista original, $|DOCS| = 7$, y el número de documentos contenidos en el área de navegación mínima, $|DOCS_{MBA}| = 5$, el cálculo de la medida DF es inmediato, arrojando un valor de $DF = \frac{7}{5} = 1{,}4$. El significado de este valor debe interpretarse como que el uso del clustering mejora la tarea de recuperación de información en un factor de $1{,}4$ respecto a realizar la misma tarea sobre la lista de documentos inicialmente recuperada por el motor de búsqueda.

7.4.2. Precisión del MBA y la medida DF

Desde el punto de vista de la recuperación de información, la medida DF tendría otra interpretación interesante. Es posible definir el factor de destilación DF como el grado de mejora en la precisión del clustering frente a la precisión de la lista ordenada original. De hecho, siendo $DOCS_{rel}$ y $DOCS_{MBA_{rel}}$ el conjunto de documentos relevantes contenidos en $DOCS$ y $DOCS_{MBA}$ respectivamente, podríamos expresar esta proporción del siguiente modo:

$$DF(L) = \frac{|DOCS|}{|DOCS_{MBA}|} = \frac{\frac{DOCS_{MBA_{rel}}}{|DOCS_{MBA}|}}{\frac{DOCS_{rel}}{|DOCS|}} = \frac{Precision_{MBA}}{Precision_{DOCS}} \tag{7.5}$$

Ya que $DOCS_{rel} = DOCS_{MBA_{rel}}$, por lo que la expresión obtenida finalmente para la medida DF resulta ser exactamente igual que la de la ecuación 7.4.

Volviendo al ejemplo presentado en la sección anterior (figura 7.13), el valor obtenido para la medida DF representa la mejora de la precisión del MBA con respecto a la precisión de la lista completa de documentos recuperados, siendo $Precision_{MBA} = \frac{4}{5}$ y $Precision_{DOCS} = \frac{4}{7}$.

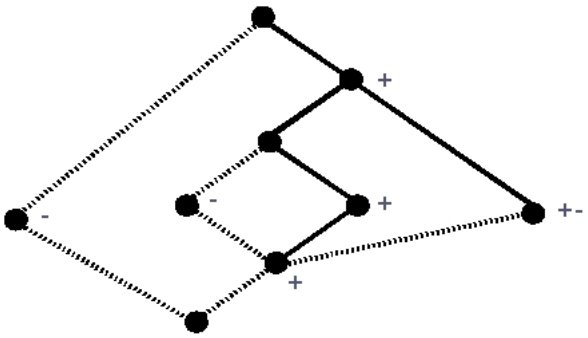


Figura 7.13: Calculo de la medida DF sobre un clustering

7.4.3. Calidad del clustering

La medida DF puede resultar una buena aproximación inicial para determinar la calidad de un clustering, permitiendo determinar de manera sencilla el grado en el que el éste es capaz de aislar los documentos relevantes. Sin embargo, su utilización no permite realizar una evaluación completa de todos los factores asociados a la calidad del retículo. Esto es debido a que la medida DF no considera:

- *La proporción de nodos contenida en el área de navegación mínima*. Tal y como expusimos en la sección 7.1, el número de nodos que el usuario debe explorar para alcanzar la información relevante es un factor determinante a la hora de evaluar la calidad del clustering. Aunque la medida DF hace uso del área de navegación mínima, ésta únicamente opera con los documentos contenidos en ella, sin tener en cuenta el número de nodos que conforman dicha área. Cuando el número de nodos contenidos en el área de navegación mínima sea muy elevado, la organización no puede ser óptima aunque la medida DF sea alta.

- *El coste cognitivo de examinar los descriptores asociados a cada cluster*. Considerar el número de clusters implicados en el proceso de navegación hasta la información relevante supone tener en cuenta el coste cognitivo que la exploración de sus descriptores supondrá al usuario.

La figura 7.14 ilustra este tipo de escenarios. En esta figura puede observarse el clustering obtenido para un conjunto de ocho documentos recuperados, donde el área de navegación mínima lo destacamos en línea continua. Como puede observarse, el valor para la medida DF (que es $\frac{8}{4} = 2$) indicaría una mejora notable de la calidad del clustering frente a la lista original de documentos, llegando a duplicar su precisión ($Precision_{DOCS} = \frac{4}{8}$ vs. $Precision_{MBA} = \frac{4}{4} = 1$). Sin embargo, el hecho de navegar hasta los clusters relevantes implica que el usuario debe considerar, a lo largo de este proceso, las descripciones de absolutamente todos los clusters generados (representados en color rojo en las figuras), con la consiguiente inversión de tiempo y esfuerzo cognitivo. En contraste, la figura 7.15 muestra una aproximación de clustering alternativa a la que acabamos de proponer con unas cualidades mucho más adecuadas para la tarea de recuperación. Como puede observarse en este segundo ejemplo, el valor para la medida DF es idéntico al caso anterior pero, sin embargo,

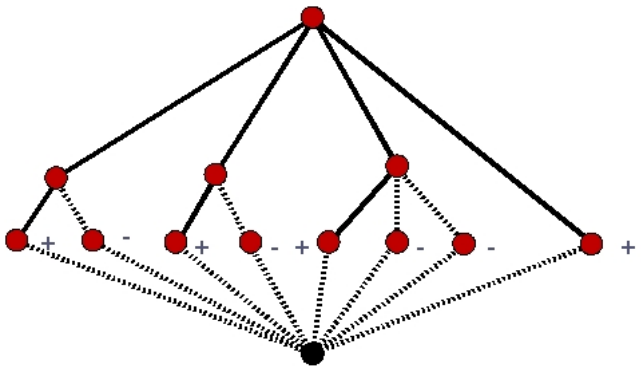


Figura 7.14: Ejemplo de un clustering con un buen valor para la medida DF pero con poca calidad desde el punto de vista de la tarea

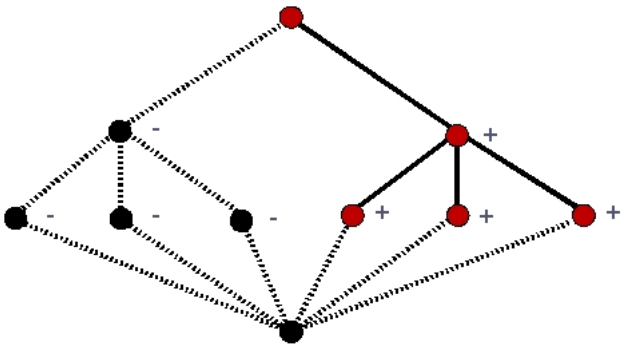


Figura 7.15: Ejemplo de un clustering con un buen valor para la medida DF con una calidad adecuada desde el punto de vista de la tarea

las propiedades del clustering implican un menor esfuerzo al usuario para alcanzar la información relevante, debido a que tiene que considerar un número de clusters muy inferior al caso anterior.

Con el fin de resolver los problemas planteados sobre la medida DF, en esta sección proponemos una medida más completa, denominada *calidad del clustering* o *clustering quality* (CQ), que teniendo en cuenta el coste cognitivo asociado a explorar un descriptor permita cuantificar el grado en el que el clustering mejora la tarea de recuperación de información respecto a la lista de documentos original. En este sentido, y con el fin de poder determinar el número de clusters que el usuario debe considerar en su proceso de navegación, definiremos una nueva magnitud denominada *área de visualización mínima* o *minimal viewing area* (MVA).

El área de visualización mínima se define como el número de clusters que el usuario debe considerar a lo largo del proceso de navegación por el área de navegación mínima. Formalmente, el área de visualización mínima se definirá a partir del conjunto de vecinos inferiores de todos los clusters del área de navegación mínima del siguiente modo:

$$MVA = \{x \in L | (\exists y \in MBA) x \prec y\} \tag{7.6}$$

Nótese como los clusters coloreados en rojo en las figuras 7.14 y 7.15 representan el área de visualización mínima asociada a los clusterings propuestos.

Una vez introducido el concepto de área de visualización mínima, es posible redefinir el coste cognitivo asociado al retículo, presentado en la ecuación 7.3, en función del número de documentos del área de navegación mínima y del número de clusters contenidos en el área de visualización mínima de la siguiente manera:

$$\text{coste cognitivo}(L) = \text{coste cognitivo}(MBA) = k_d|DOCS_{MBA}| + k_c|MVA| \tag{7.7}$$

Donde k_c representa el coste cognitivo promedio asociado a explorar la descripción asociada a un cluster con el fin de decidir acerca de su relevancia.

De esta manera, la definición de la calidad del clustering, o medida CQ, en este caso podría reescribirse a partir de las ecuaciones 7.2 y 7.4 del siguiente modo:

$$CQ(L) = \frac{k_d|DOCS|}{k_d|DOCS_{MBA}| + k_c|MVA|} = \frac{|DOCS|}{|DOCS_{MBA}| + \frac{k_c}{k_d}|MVA|} \tag{7.8}$$

Que nuevamente expresa la capacidad del clustering para mejorar la tarea de recuperación de información respecto a la lista ordenada inicialmente recuperada.

Nótese como, en este caso, el resultado de la evaluación dependerá de los valores asignados a k_d y a k_c, los cuales están estrechamente ligados a las características propias del sistema implementado y al escenario sobre el que se lleva a cabo la evaluación, y que reflejan la proporción entre el coste cognitivo que supone explorar la descripción asignada a un cluster y el coste cognitivo asociado a la exploración de la descripción de un documento ($k \equiv \frac{k_c}{k_d}$). Debemos destacar que en el caso en el que $k = 0$ entonces $CQ = DF$.

El rango de valores en los que opera la medida CQ pertenece al intervalo $(0, +\infty)$, donde un valor igual a uno representaría que el clustering no aporta ninguna mejora sobre la lista de documentos original. Sin embargo, y al contrario de lo que ocurría con la medida DF, la calidad del clustering puede obtener valores por debajo de uno. Este tipo de valores pueden interpretarse como que el coste cognitivo asociado a recorrer el clustering es superior al coste cognitivo de examinar la lista y, por lo tanto, el modelo no es susceptible de ser aplicado en ese caso. Estas situaciones pueden deberse principalmente a que, o bien el número de clusters a considerar es muy elevado, o bien el coste cognitivo asociado a examinar las descripciones de los clusters es muy similar, o incluso superior, al coste cognitivo asociado a examinar la representación de los documentos.

Dado que el parámetro k depende directamente de las característica propias del sistema (el modo en el que éste lleva a cabo la representación tanto de los documentos como de los descriptores asociados a los clusters), así como del tipo de información sobre el cual se aplica el proceso de clustering, este debe determinarse experimentalmente (en el capítulo 8 presentamos un modo de obtener su valor para un sistema concreto). No obstante, en el caso en el que su valor sea desconocido (por ejemplo

durante el desarrollo de prototipos), la medida CQ sigue siendo igualmente aplicable dado que permite realizar un estudio comparativo sobre diferentes aproximaciones de clustering en función de este parámetro cuyas conclusiones pueden ayudar a la toma de decisiones orientadas al desarrollo de un sistema final. En el siguiente apartado desarrollamos ambas aproximaciones.

Aplicación de la medida CQ

Desde el punto de vista de su aplicación, podemos establecer dos maneras de llevar a cabo la evaluación de un sistema basado en nuestro modelo utilizando la medida CQ:

- *Escenario 1*. En el caso de conocer los valores de k_c y k_d para un sistema concreto y un corpus de documentos específico es posible realizar un estudio comparativo entre diferentes aproximaciones de clustering basadas, por ejemplo, en distintos criterios de selección de descriptores o en diferentes aproximaciones para la extracción de los mismos. En este caso, el uso de la medida CQ obtendrá valores exactos que permitirán decidir acerca de cual de las estrategias genera estructuras de clustering de mejor calidad, así como cual es su mejora de calidad con respecto a la lista de documentos inicialmente devuelta por el motor de búsqueda.

- *Escenario 2*. En caso de no conocer los valores k_c y k_d es posible realizar un estudio de la medida CQ en función del parámetro $k = k_c/k_d$ para diferentes aproximaciones con el fin de determinar la adecuación de cada una de ellas sobre diferentes intervalos de k. Sobre este particular resulta especialmente interesante determinar el valor de corte k_{max} a partir del cual la medida CQ obtendrá valores inferiores a uno (y como consecuencia no tendrá sentido aplicar una aproximación basada en clustering). La obtención de este valor de corte puede llevarse a cabo de manera sencilla despejando de la ecuación 7.8:

$$k_{max} = \frac{|DOCS| - |DOCS_{MBA}|}{|MVA|} \tag{7.9}$$

  Cualquier valor de $k < k_{max}$ implicará una medida $CQ > 1$ y, por lo tanto, la adecuación del clustering frente a la lista original de documentos para resolver la tarea de recuperación de información.

  En las figuras 7.14 y 7.15 se han presentado dos alternativas de clustering para organizar ocho documentos, de los cuales cuatro son relevantes. El calculo de la medida CQ en este escenario se puede llevar a cabo de manera sencilla. Debido a que no disponemos de los valores reales para los parámetros k_c y k_d optaremos por realizar una evaluación sobre sus correspondientes valores k_{max}. En el primer caso obtenemos un valor para k_{max} igual a $\frac{1}{3}$, lo que significa que para obtener valores adecuados de calidad de clustering el coste cognitivo asociado a los documentos debe ser como mínimo tres veces mayor que el coste cognitivo asociado a los clusters ($k_d > 3k_c$). En contraste, en el segundo caso el valor para k_{max} será de $\frac{4}{5}$, lo que en este caso indica que los valores adecuados de calidad de clustering se obtendrán cuando el coste cognitivo asociado los documentos sea como mínimo $1{,}25$ veces mayor que el coste

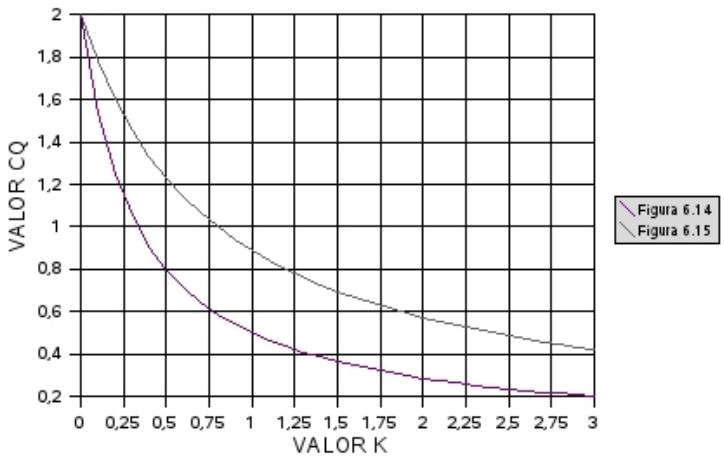


Figura 7.16: Evolución de la medida CQ para los retículos de las figuras 7.14 y 7.15

cognitivo asociado a los clusters ($k_d > \frac{5}{4}k_c$). De acuerdo a los valores obtenidos podríamos concluir que la segunda de las aproximaciones de clustering propuestas es la más adecuada.

No obstante, y con el fin de poder corroborar la conclusión obtenida para los retículos de las figuras 7.14 y 7.15, resulta conveniente realizar un estudio sobre los posibles valores de la medida CQ con el objeto de comprobar que no existen cruces en las gráficas que pudieran hacer más adecuadas ambas aproximaciones sobre intervalos diferentes. La gráfica 7.16 muestra los valores de la medida CQ para los retículos de las figuras 7.14 y 7.15. Como puede observarse, la gráfica correspondiente a la figura 7.15 se mantiene por encima de la correspondiente a la figura 7.14, lo que indica su adecuación para cualquier valor de k considerado sobre el intervalo $(0, k_{max})$.

7.5. Recapitulación

En este capítulo hemos propuesto un conjunto de medidas orientadas a evaluar la calidad de las estructuras de clustering obtenidas aplicando el modelo propuesto en este trabajo.

Como características principales de las medidas debemos destacar su orientación a una tarea de recuperación de información, así como la consideración de las características estructurales del clustering orientadas a reflejar en sus resultados el coste cognitivo asociado a la interacción del usuario con la estructura de clustering.

Con el fin de poder desarrollar las medidas presentadas, inicialmente hemos introducido el concepto de Area de Navegación Mínima o MBA, que resulta fundamental para determinar qué parte del clustering va a ser considerada a la hora de aplicar las medidas de evaluación. El área obtenida representa el conjunto de clusters y de caminos óptimo que el usuario debería recorrer para acceder

a toda la información relevante, por lo que cualquier medida de evaluación que trabaje sobre esta estructura proporcionará valores máximos o de cota superior con respecto a la mejora producida por el sistema de clustering con respecto a la lista ordenada original.

A continuación hemos definido las medidas DF y CQ orientadas a evaluar la calidad de un clustering y cuya principal diferencia radica en que, mientras que la primera únicamente considera el coste cognitivo asociado a los documentos, la segunda la extiende incluyendo el coste cognitivo asociado a explorar los clusters contenidos en el MBA.

Finalmente, debemos destacar que, hasta la fecha, esta es la primera aportación para la evaluación automática de estructuras de clustering basadas en AFC.

Parte III

DESARROLLO Y EVALUACION

Capítulo 8

Diseño Experimental

Los capítulos anteriores de esta Tesis Doctoral han presentado un modelo para la realización de clustering de documentos basado en análisis formal de conceptos, una metodología para llevar a cabo la construcción de este tipo de sistemas, así como un marco de evaluación orientado a cuantificar automáticamente la calidad de las estructuras generadas sobre escenarios de recuperación específicos.

Tal y como expusimos entonces, aplicar el modelo propuesto implica tomar decisiones relacionadas con la extracción y selección de descriptores que influirán en la estructura del clustering y, por tanto, en su calidad para aislar correctamente la información relevante y para reducir el esfuerzo realizado por el usuario para acceder a ella. Por esta razón, la construcción de un sistema de clustering completo y funcional implica el desarrollo incremental de un conjunto de prototipos que, basados en la selección de diferentes aproximaciones y criterios, puedan ser evaluados de manera rápida y sencilla.

En este capítulo y el siguiente demostraremos la validez de nuestras propuestas mediante el desarrollo de un conjunto de prototipos que implementen las diferentes aproximaciones presentadas en esta Tesis Doctoral y que evaluaremos utilizando las medidas propuestas. Esto nos ayudará a decidir acerca de su adecuación a un escenario concreto. En este capítulo presentaremos el diseño experimental utilizado para evaluar los prototipos presentados en el capítulo 9. En concreto, describiremos el corpus utilizado en los experimentos, así como la metodología utilizada para estimar experimentalmente el valor del parámetro k. Este parámetro será determinado mediante experimentos con usuarios reales y es necesario para realzar una discusión acerca de los valores de la medida CQ obtenidos. Su valor será utilizado en el siguiente capítulo para comparar los prototipos presentados.

8.1. Consideraciones acerca del proceso de evaluación

De acuerdo al conjunto de medidas presentado en el capítulo 7, la evaluación de un sistema de clustering obtenido a partir de nuestro modelo implica:

- Aplicar el proceso de clustering sobre un corpus de documentos controlado sobre el cual sea

posible conocer a priori el conjunto de documentos relevantes a una consulta concreta. En este sentido, todos los prototipos presentados en este capítulo (a excepción del sistema final, JBraindead) se caracterizan por realizar el proceso de clustering sobre un corpus cerrado y bien conocido para el cual se dispone de un conjunto de consultas con juicios de relevancia asignados manualmente por expertos.

- Disponer de una estimación de la proporción entre el coste cognitivo asociado a los descriptores de los clusters y el coste cognitivo asociado a los documentos (lo que implica conocer el parámetro k de la fórmula 7.8). Debido a que este parámetro está estrechamente ligado al modo en el que los clusters y los documentos son representados por el sistema, su obtención debe llevarse a cabo de manera experimental realizando una evaluación con usuarios reales. Para evaluar nuestros prototipos realizamos la estimación de k considerando el coste cognitivo de diferentes tipos de descriptores frente al coste cognitivo asociado a los documentos mediante un sencillo experimento con usuarios.

8.2. Corpus de Prueba

Para llevar a cabo la evaluación de los prototipos presentados hemos seleccionado el corpus de noticias periodísticas de la Agencia EFE correspondiente al año 1994 (EFE94), utilizado en las evaluaciones CLEF[1], cuyas características principales son las siguientes:

- *Temática del corpus.* La temática de las noticias incluidas en el corpus utilizado es de propósito general, permitiendo disponer de un marco de información variada adecuado para las tareas de clustering propuestas en este trabajo.

- *Volumen del corpus.* El corpus EFE94 dispone de aproximadamente 215.000 noticias que ocupan un total de 500 MBytes de espacio en disco, lo que permite aplicar nuestras aproximaciones sobre un entorno realista.

- *Idioma del corpus.* El idioma en el que se encuentran redactadas las noticias de EFE94 es el castellano. Al ser nuestra lengua materna nos ha facilitado la verificación de la corrección lingüística de las descripciones generadas (unigramas, n-gramas y sintagmas terminológicos), así como la correcta distribución de los documentos sobre las estructuras de clustering obtenidas.

- *Metadatos del corpus.* Cada una de las noticias de EFE94 está marcada inicialmente con metadatos SGML que permiten distinguir su estructura y que hemos convertido a XML. El cuadro 8.1 muestra la DTD correspondiente a esta colección, donde puede observarse como la información está etiquetada de acuerdo a un identificador de noticia, un texto que hace referencia al nombre del fichero donde ésta se encuentra, la fecha de la noticia, la categoría que le corresponde (asignada manualmente por el periodista), el título de la noticia y, finalmente,

[1] CLEF: www.clef-campaign.org

```
<!ELEMENT DOCUMENTS DOCUMENT*>
<!ELEMENT DOCUMENT (DOCID,DOCFILE,DATE,CATEGORY,
TITLE,TEXT)>
<!ELEMENT DOCID (#PCDATA)>
<!ELEMENT DOCFILE (#PCDATA)>
<!ELEMENT DATE (#PCDATA)>
<!ELEMENT CATEGORY (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT TEXT(#PCDATA)>
```

Cuadro 8.1: DTD correspondiente al corpus de noticias EFE94 utilizado en los prototipos

el cuerpo de la misma. El cuadro 8.2 muestra un ejemplo de noticia marcada de acuerdo a esta DTD. Es importante destacar que nuestra propuesta está orientada a la realización de clustering sobre texto libre, lo que implica que no es necesario disponer de información marcada para la realización del proceso de clustering. Los prototipos presentados procesarán la información relativa a cada una de las noticias ignorando sus metadatos, siendo únicamente considerados en el proceso de formateado de los documentos para su visualización en la interfaz de usuario.

8.3. Juicios de Relevancia

Debido a que el corpus EFE94 ha sido sido utilizado en diferentes campañas del CLEF (Cross Language Evaluation Forum) [100] para la evaluación de diversos tipos de sistemas de recuperación de información, la colección dispone de un amplio conjunto de consultas con juicios de relevancia asignados manualmente por expertos. En concreto, EFE94 dispone de un conjunto de 160 consultas orientadas a las tareas CLEF que están descritas por un título (i.e. que representa la consulta propiamente dicha), una descripción (i.e. que describe brevemente el tipo de documentos que deben ser recuperados por la consulta) y una narrativa (i.e. que extiende y complementa la descripción presentada). Para nuestros prototipos y experimentos hemos seleccionado el subconjunto de consultas de la 41 a la 87, proporcionando a los diferentes sistemas únicamente el título de la consulta. El cuadro 8.3 ejemplifica alguna de ellas.

8.4. Estimación del parámetro k

Las medidas de evaluación propuestas en el capítulo 7 hacen uso del concepto de coste cognitivo. Considerar este concepto en las medidas de evaluación supone considerar el esfuerzo que debe hacer el usuario para determinar si un item[2] concreto es relevante a sus necesidades. Obviamente, el valor del coste cognitivo dependerá del modo en el que la información relacionada con el item es

[2] a lo largo de esta sección, el término *item* hará referencia a un cluster o a un documento indistintamente.

<DOCUMENT>
<DOCID>EFE19940101-00044</DOCID>
<!ELEMENT DOCID (#PCDATA)>
<DOCFILE>EFE19940101.xml</DOCFILE>
<DATE>19940101</DATE>
<CATEGORY>POLITICA</CATEGORY>
<TITLE>
SUIZA-CANTONES.PANORAMA GEOGRAFICO SUIZO MODIFICADO CON LLEGADA NUEVO AÑO
</TITLE>
<TEXT>
Laufen (Suiza), 1 ene (EFE).- El panorama geográfico helvético quedó modificado en la medianoche del 31 de diciembre con la incorporación de la localidad de Laufen al cantón de Basilea y su desvinculación del de Berna, al que estaba unido desde 1815. Mediante la incorporación al semi-cantón de Basilea-Campo (junto a Basilea-ciudad configuran un cantón), Suiza experimenta su transformación territorial más importante desde la creación del cantón del Jura. Las trece comunas de Laufen y sus cerca de 15.000 habitantes se han convertido así en el quinto distrito de Basilea-campo y Laufen obtiene el rango del quinta capital del distrito, con Liestal, Arlesheim, Sissach y Waldenburgo. Los habitantes de Laufen decidieron el pasado año en referendum separarse del cantón que integra a la capital de Suiza y y vincularse al vecino de Basilea por 'una mayor semejanza y cercanía cultural'. La Confederación Helvética está dividida en 26 cantones, de los que resaltan por su importancia: Basilea, Zurich, Ginebra, Berna, Lausana, Lucerna, San Gallen, Winterthur y Lugano.EFE MCR/HA 01/01/11-45/94
</TEXT>
</DOCUMENT>

Cuadro 8.2: Ejemplo de un documento perteneciente al corpus EFE94

Número	Consulta
41	Pesticidas en alimentos para bebés
42	Naciones Unidas y Estados Unidos invaden Haití
43	'El Niño' y el tiempo
44	Indurain gana el Tour
45	El tratado de paz entre Israel y Jordania
46	Embargo sobre Irak
47	Intervención rusa en Chechenia
48	Fuerzas de paz en Bosnia
49	Caída de las exportaciones de coches en Japón
50	Levantamiento en Chiapas

Cuadro 8.3: Título de las consultas 41-50 con juicios de relevancia asignados correspondientes al corpus EFE94

mostrada al usuario y su valor podría expresarse como el tiempo que éste tarda en decidir acerca de su relevancia. Aplicado a la tarea propuesta, debemos destacar cómo este valor está directamente ligado al escenario sobre el cual se realiza el proceso de clustering, influyendo en su valor final aspectos tan importantes como la tipología de las descripciones utilizadas para los items, su calidad informativa o las propias características del corpus sobre el cual se realiza la tarea.

De acuerdo a nuestra propuesta de evaluación, consideraremos dos costes cognitivos diferentes: a) el asociado al esfuerzo realizado por el usuario para determinar la relevancia de un cluster a partir del conjunto de descriptores utilizados para representarlo (k_c), y; b) el asociado al esfuerzo realizado por el usuario para determinar la relevancia de un documento a partir de su descripción (k_d). El valor $k = \frac{k_c}{k_d}$ mostrará la proporción entre ambos.

De este modo, la estimación del valor global k para cada uno de los prototipos presentados en este capítulo es el primer objetivo para poder realizar su evaluación de manera adecuada. Su estimación puede plantearse de una manera exhaustiva, es decir, considerando el coste cognitivo correspondiente a todas y cada una de las aproximaciones implementadas, o bien se puede realizar una estimación un poco más general, considerando únicamente el coste cognitivo asociado a aquellas aproximaciones que supongan diferencias notables en el modo en el que la información es finalmente presentada al usuario. De este modo, una estimación generalista puede utilizarse para descartar rápidamente aproximaciones que generen estructuras de clustering poco adecuadas, pudiendo aplicar entonces una estimación exhaustiva para realizar un ajuste fino sobre aquellas aproximaciones con valores de calidad muy similares. Nótese como la estimación de k supone realizar una evaluación con usuarios reales, con sus consiguientes dificultades, por lo que, si una estimación generalista es suficiente para obtener resultados adecuados, no sería recomendable realizar una segunda estimación de k.

En este trabajo hemos optado por realizar una estimación generalista del parámetro k, diferenciando el coste cognitivo asociado a un cluster en función del tipo de descriptor utilizado para representarlo. En este sentido, el experimento que presentamos a continuación estima el coste cognitivo asociado a los clusters considerando que estos están descritos o bien por unigramas o bien por n-gramas. En este trabajo no hemos realizado una diferenciación entre el coste cognitivo asociado a los n-gramas

Número	Consulta
41	Pesticidas en alimentos para bebés
50	Levantamiento en Chiapas
59	Virus informáticos
71	Verduras, frutas y cáncer
82	El IRA ataca aeropuertos

Cuadro 8.4: Consultas utilizadas en el proceso de evaluación con usuario para la estimación del parámetro k

y a los sintagmas terminológicos por considerar que tanto su capacidad informativa como su morfología son muy similares. Respecto al coste cognitivo asociado a los documentos, consideramos para todos los prototipos que éstos están representados por su título y un pequeño fragmento de texto (denominado snippet) extraído automáticamente del documento completo y caracterizado por contener algún término de la consulta realizada.

8.4.1. Escenario de evaluación

La tarea de evaluación propuesta la realizaron ocho evaluadores diferentes sobre cinco consultas extraídas del conjunto de consultas con juicios de relevancia presentado en esta sección (ver cuadro 8.4). Los evaluadores no tenían un tiempo límite para realizar la tarea, aunque sí fueron informados que el objetivo era realizarla lo más rápidamente posible y de que deberían medir el tiempo empleado en la tarea.

8.4.2. Descripción de la tarea

La tarea de evaluación presentada a los evaluadores, y que tuvieron que realizar sobre las cinco consultas seleccionadas, fue enunciada del siguiente modo:

> 'Dada la siguiente consulta y tres listas de descripciones determinar, para cada una de las listas, el grado de relevancia de cada una de sus descripciones respecto a la consulta propuesta'

Las listas de items se definieron del siguiente modo:

- La primera lista representaba un conjunto de clusters descritos por unigramas. Estos fueron obtenidos de uno de los prototipos implementados, lanzando las cinco consultas objetivo sobre el sistema y seleccionando las descripciones asociadas a los conceptos atributo contenidos en el retículo. El número elementos contenidos en esta primera lista fue de 15 representaciones.

Lista de términos
informático
htlv, virus
virus
inmunodeficiencia
hiv, virus
virólogo, virus
ocid, virus
viruela, virus
skehel, virologo, virus, viruela
lcv, berghella, virus, informático
berghella, informático
santas, anyware, virus, informático
anyware, virus, informático
ansei, informático
gigabyte, informático

Cuadro 8.5: Conjunto de términos proporcionados a los evaluadores para la consulta 'virus informáticos'.

- La segunda lista representaba un conjunto de clusters descritos por n-gramas. Su proceso de obtención fue muy similar al utilizado para generar la primera lista, con la particularidad de que, en este caso, se utilizó uno de los prototipos basado en n-gramas para describir el clustering. Nuevamente, el número de elementos contenidos en la lista fue de 15 representaciones.
- La tercera lista representaba un conjunto de documentos descritos por una combinación de su título y un fragmento de texto obtenido automáticamente a partir del documento completo (i.e. snippet). En este último caso, el número de elementos presentados en la lista fue de 30 representaciones.

Para cada una de las listas se tomó el tiempo empleado por el evaluador en determinar el grado de relevancia de todos sus elementos. Los cuadros 8.5, 8.6 y 8.7 ejemplifican el tipo de información proporcionada a los evaluadores para la consulta *virus informáticos*.

Lista de snippets
expertos informaticos discrepan repercusion virus 'barrotes' madrid, 5 ene (efe).- la repercusión del virus informático de origen español 'barrotes ... de la información contenida en el disco duro. como medidas de prevención ante los virus informáticos ... entre los expertos informáticos consultados por efe. 'la reacción ha sido muy grande
ataque virus informatico 'barrotes' mañana 5 de enero madrid, 4 ene (efe).- el virus informático 'barrotes', de origen español, atacará mañana miércoles ... ', explicó a efe el director técnico de la empresa anyware seguridad informática s.a., carlos jiménez. el virus ... telemáticas). los virus informáticos son programas de ordenador creados con el objetivo

Lista de snippets
italia-virus detectado programa para creacion de 'virus' ... potencialmente infinito de 'virus' informáticos ha sido detectado por dos expertos italianos, que han advertido ... un 'virus a la carta' que ataque la memoria de un sistema informático o destruya la base de datos ... un cambio en las 'enfermedades' informáticas producidas por los 10.000 'virus' existentes
virus informatico 'barrotes' afecta ordenadores tve madrid, 5 ene (efe).- el virus informático 'barrotes' afectó hoy, entre otros ordenadores, a los del sistema informático de televisión española (tve), declararon a efe fuentes del departamento ... de sinutec, empresa distribuidora de productos para seguridad informática, pedro muñoz
mas un millon ordenadores amenazados por virus 'michelangelo' su información el próximo domingo, día 6, cuando se prevé que actúe el virus informático 'michelangelo', alertó hoy la asociación de técnicos en informática (ati). la acción destructora del virus ... de anyware, empresa de seguridad informática, carlos jiménez, para quien este virus es 'muy peligroso
mexico-elecciones identificados boicoteadores sistema informatico elecciones ... boicotear, con un 'virus informático', el sistema de cómputo electoral utilizado en los comicios ... próximo. el 21 de agosto de 1994, día de las elecciones presidenciales, se intentó 'infectar' con un 'virus informático' el sistema de cómputo para boicotear el recuento de votos que recibía de todo el país
ordenadores españoles amenazados por 2.000 virus existentes del contenido del disco duro. en los virus informáticos hay dos fases, según los expertos ... virus reconocidos, entre ellos el denominado 'michelangelo', que se espera ataque mañana. los virus más peligrosos que en la actualidad pueden afectar a los ordenadores españoles
china-deng xiaoping virus informatico daño segunda parte biografia 'supremo' chino pekín, 12 may (efe).- un virus informático, posiblemente debido a la escasa calidad y garantía de muchos de los disquetes que se venden en china, dañó seriamente la preparación de la segunda parte de la biografía del supremo dirigente chino, deng xiaoping, se informó hoy, jueves. el virus
ordenadores-honecker fallecido dirigente alemania oriental resucito en forma virus ' el pasado sábado en forma de virus informático en las pantallas de numerosos ordenadores de alemania ... nacional de la antigua alemania del este. no contento con ello, el creador del virus incluyó un mensaje que anunciaba la destrucción de los programas informáticos, 'por orden del consejo
mexico-elecciones-resultados resultados de comicios mantienen triunfo de partido gobernante ... de introducir en el sistema de ordenadores, encargado del recuento rápido, un virus informático. carpizo ... tomadas ante los intentos de introducir el virus y dijo que el ife ya tenía indicios de quienes podían
eeuu-literatura de los dinosaurios al acoso sexual de hombres por mujeres ... controvertido: trata del acoso sexual de un hombre por su jefa en una compañía de informática en estados ... strain' (los estragos de un virus extraterrestre), 'congo' (gorilas superinteligentes y asesinos

Lista de snippets
senado y ccaa en jornadas informatica parlamentaria palma , excepto extremadura y la rioja, han participado en palma en unas jornadas de informática ... hoy, es estudiar proyectos de cooperación relacionados con la informática aplicada a la actividad ... , entre otros temas, los sistemas documentales, la seguridad en entornos informáticos aplicada
fallo informatica provoca cortes telefonos barcelona madrid, 20 oct (efe).- un fallo en el sistema informático de una de las centrales de telefónica ... colapsadas con el fallo informático. técnicos de telefónica trabajan en estos momentos en la reparación del software (el prograna informático). efe eb/fam 10/20/19-13/94
constituida agrupacion española informatica y matematicas madrid, 11 abr (efe).- matemáticos e informáticos de españa han creado una asociación ... futuros programas europeos de investigación. la agrupación española de informática ... , desde los fundamentos de la informática hasta las nuevas arquitecturas de sistemas y computadores
femp pide responsabilidad en la utilizacion datos informaticos datos informáticos que 'sean rigurosos y respeten el derecho a la intimidad de los ciudadanos'. el vicepresidente de la femp, que inauguró hoy en salamanca las xvi jornadas informáticas de las administraciones locales (jial), insistió en que la informática debe estar al servicio
premio a la mejor tesis doctoral sobre informatica 'pc world' para distinguir la mejor tesis doctoral sobre informática que haya sido aprobada ... , a través de su programa de divulgación informática 'bienvenido mister chip', colabora en este premio ... , y contará también con la opinión de los decanos de varias facultades de informática. efe on.as
nuevos procesadores informaticos triplicaran velocidad equipos madrid, 5 may (efe).- los procesadores informáticos de nueva generación, denominados 'p6 ... compañía de capital español enfocada a la integración de informática, tecnología ... en equipos informaticos empezaría a principios de 1995.efe al/rs 05/05/13-34/94
criticas a regulacion delito informatico en nuevo codigo penal madrid, 8 nov (efe).- la comisión de libertades e informática (cli) considera que la regulación del delito informático, que por primera vez se incorpora al proyecto del nuevo código penal ... de justicia, recoge un endurecimiento de las penas para el delito informático -en referencia
banesto informaticos preocupados destino nuevas tecnologias banco ... y negligencias en informática y comunicaciones avanzadas (apedanica) expresó hoy su 'seria preocupación ... contables'. la asociación de informáticos, que asegura que no ha 'tenido acceso a ningún ordenador ... , con pantallas sensibles y seguridad informática. y agregan: 'estamos en permanente alerta
eeuu-informatica piratas informaticos se vengan de autores libro sobre tema informática, han sido víctimas de un ataque cibernértico: alguien se interfirió con su teléfono ... ' del que fueron objeto se parece a algunas de las tácticas empleadas por bandas rivales de piratas informáticos ... en informática. alguien desvió a un contestador automático todas las llamadas hechas al número
trabajo inpeccionara sus pc's detectar programas piratas general de informática y estadística del ministerio de trabajo, pedro maestre, explico ... del estado, para lo cual informarán al consejo superior de informática y la comisión interministerial de adquisición de bienes y servicios informáticos. la iniciativa del ministerio de trabajo

Lista de snippets
correos formara profesionales aplicar nuevas tecnologias a los servicios postales y telegráficos, como técnicas digitales, uso de satélites, informática ... programas de investigación conjuntos en los ámbitos de las nuevas tecnologías informáticas ... cinco becas cada año académico, para alumnos de quinto y sexto de la facultad de informática
profesionalidad y sectorizacion en el nuevo simo tcl tcl 94 -feria internacional de informática, multimedia y comunicaciones-, que se celebrará ... material de oficina e informática, incorpora este año las nuevas tecnologías de la comunicación y de la información. los productos no informáticos como mobiliario de oficina o elementos
nuevo metodo ayuda para aprobar selectividad . se trata de un libro y un programa informático para las distintas opciones ... con éxito por el alumno, este será premiado con el acceso a un juego informático. en esta obra, publicada por larousse planeta, han cooperado mas de 180 pedagógos, informáticos y profesores
siniestros informaticos, perdidas 80.000 millones año españa palma de mallorca, 17 may (efe).- las pérdidas económicas debidas a siniestros informáticos ... del v congreso internacional de seguridad en entornos informáticos, que comenzará mañana en mallorca. el presidente de la asociación nacional de la seguridad en entornos informáticos (ansei
sector informatico pierde 50.000 millones al año por pirateria madrid, 17 nov (efe).- la industria del software pierde en españa unos 50.000 millones de pesetas anuales por culpa de la piratería informática, denunció hoy la asociación española de empresas ... '. para el sector de la informática, las pérdidas alcanzadas como consecuencia del delito informático 'parecen
informatica-chile santiago acoge exposicion mas importante de telecomunicaciones santiago de chile, 3 jul (efe).- la exposición más importante de la informática ... a unas 250 empresas que expondrán 800 de las principales marcas del mercado mundial de la informática ... informáticos abiertos. efe mw/ltm/ag 07/03/15-33/94
china-ibm ibm participara infraestructura red informatica china pekín, 4 may (efe).- el gigante estadounidense de la informática ibm será la primera compañía extranjera en participar en un proyecto de infraestructura de una red de comunicación informática ... junto con la compañía de comunicaciones jitong que dirige el plan de infraestructura informática
caja ahorros enseña informatica amas casa para economia hogar de informática dirigido a amas de casa con el objetivo de que aprendan las nociones básicas ... de la entidad financiera. bajo el título genérico de 'aplicaciones informaticas familiares', el aula de informática de la caja de ahorros de terrassa ha abierto esta semana el periodo de inscripción
españa-informatica pirateria programas fue en 1993 de 46.500 millones pesetas londres, 27 abr (efe).- los fabricantes y distribuidores que operan en el sector informático ... por ciento de 1992. este incremento de la piratería informática en españa coincidió ... informática en europa ascendió al 61 por ciento frente al 66 por ciento de 1992, lo que supuso unas pérdidas

Cuadro 8.7: Conjunto de snippets proporcionados a los evaluadores para la consulta 'virus informáticos'.

Lista de n-gramas
inmunodeficiencia adquirida
infectar con un virus
contagio virus
virus del sida
eeuu-sida
virus de inmunodeficiencia
programas informáticos
virus informático
virus hiv
duplica a los portadores del virus
virus viruela
virus conocidos
aislar el virus
seguridad en entornos informáticos
tanzania-leones
piratas informáticos
virus htlv
virus gripe
servicios informáticos

Cuadro 8.6: Conjunto de n-gramas proporcionados a los evaluadores para la consulta 'virus informáticos'.

Representación	Tiempo (sg)
unigramas	3.49
n-gramas	2.15
titulo+snippet	5.47

Cuadro 8.8: Resultados del tiempo medio requerido por los evaluadores para decidir acerca de la relevancia de un item a partir de su representación

8.4.3. Formulario proporcionado

Por cada una de las consultas cada uno de los evaluadores recibió un formulario donde se especificaban los siguientes datos:

- *La tarea.* Descrita por un breve texto donde se explicaban claramente sus objetivos.
- *La consulta objetivo.* Descrita por el texto de la consulta y una breve explicación que proporcionaba un poco más de información acerca de los objetivos de la búsqueda.
- *Las tres listas a evaluar.* Que incluían para cada una de las representaciones un cuadro de texto donde el usuario debía especificar el grado de relevancia de acuerdo a las siguientes opciones:
  - NO. Para el caso en el que estaba seguro que el item representado no contenía información relevante.
  - IMPROBABLE. Para el caso en el resultaba improbable que el item representado contuviera información relevante pero, sin embargo, no podía asegurarse con toda certeza.
  - PROBABLE. Para el caso en el que resultaba probable que el item representado contuviera información relevante pero no podía asegurarse con toda certeza.
  - SI. En el caso de considerar con toda certeza la relevancia del item representado.
  - NS/NC. En el caso de no poder emitir un juicio debido a no disponer de información suficiente o del conocimiento del dominio necesario.

8.4.4. Resultados de la evaluación

Los resultados obtenidos en la evaluación propuesta fueron procesados y promediados para todos los evaluadores que participaron en el proceso, obteniendo los resultados que muestra el cuadro 8.8. Los tiempos representados en el cuadro han sido normalizados con el fin de extraer el tiempo que tarda el evaluador en decidir acerca de la relevancia de un item individual. La relación de tiempos obtenida se muestra a continuación:

$$t_{n-gramas} < t_{unigramas} < t_{titulo+snippets} \tag{8.1}$$

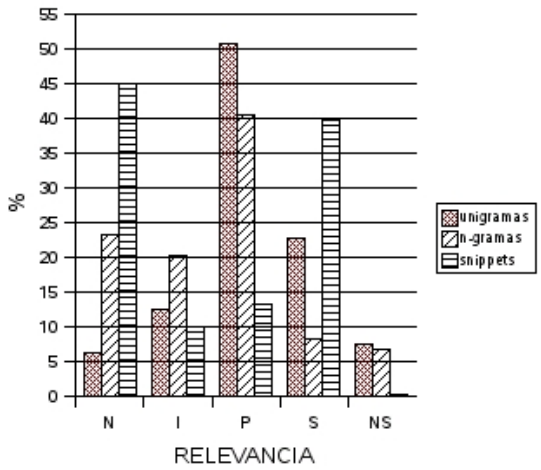


Figura 8.1: Resultados obtenidos por los evaluadores para la tarea propuesta.

Representación	**Valor k**
$k_{unigramas}$	0.64
$k_{n-gramas}$	0.40

Cuadro 8.9: Valor k considerando un sistema de clustering con descriptores basados unigramas y n-gramas

La figura 8.1 muestra el porcentaje de respuestas para caso obtenidas en el experimento. Observando estos resultados y los tiempos obtenidos podemos concluir lo siguiente:

- El coste cognitivo asociado a los documentos (descritos por su título y un pequeño fragmento de texto) es el más alto de todos. Teniendo en cuenta que este tipo de representaciones se caracterizan por tener una longitud mayor que cualquiera de las otras dos propuestas (unigramas y n-gramas) resulta lógico pensar que el tiempo empleado por el evaluador para determinar su relevancia sea mayor, aún considerando que en algunos casos le sea suficiente con explorar el título para determinar la relevancia del documento. No obstante, debemos destacar que la capacidad informativa de este tipo de representaciones es muy superior, lo que nos indica que la capacidad del usuario para determinar con certeza la relevancia de los documentos representados es mayor que en el caso de los unigramas y los n-gramas (en nuestro experimento, los evaluadores únicamente marcaron las opciones improbable o probable en un 23 % de las ocasiones).

- El coste cognitivo asociado a los clusters descritos por unigramas es superior al asociado a los clusters descritos por n-gramas. Este resultado nos sorprendió debido a que, atendiendo a la menor longitud de los unigramas respecto a los n-gramas, el tiempo empleado en determinar la relevancia de un item representado por un unigrama debería haber sido menor que su

correspondiente representado por un n-grama. La explicación a este fenómeno debe buscarse en la capacidad informativa de ambas aproximaciones y en el grado de conceptualización que éstas proporcionan. Los n-gramas se caracterizan por tener una capacidad informativa superior debida principalmente a su estructura sintáctica inherente, la cual permite presentar una información ya conceptualizada. En contraste, una representación basada en unigramas no permite reflejar de manera explícita las relaciones existentes entre éstos, lo que implica un trabajo extra de conceptualización que, en última instancia, deberá realizar el usuario. Aún así, la capacidad informativa de ambas representaciones es muy inferior a la obtenida por la representación de los documentos debido a que, en el experimento realizado, los evaluadores marcaron las opciones improbable o probable en un 62 % y en un 60 % de las ocasiones para unigramas y n-gramas respectivamente (compárese con el 23 % en el caso de los snippets).

A partir de los resultados obtenidos para k_c y k_d resulta inmediato calcular el valor del parámetro k considerando las dos aproximaciones planteadas en esta evaluación (unigramas y n-gramas). El cuadro 8.9 muestra los resultados estimados para el parámetro k considerando un sistema de clustering descrito por unigramas y n-gramas respectivamente. Estos valores serán utilizados en las discusiones posteriores sobre cada uno de los prototipos presentados.

Capítulo 9

Experimentos y Discusión

En este capítulo se presenta el conjunto de prototipos desarrollados en esta Tesis Doctoral, así como los experimentos realizados para evaluar la calidad de las estructuras de clustering generadas.

A lo largo de este trabajo hemos desarrollado cuatro prototipos que implementan las diferentes propuestas para la extracción y selección de descriptores. En primer lugar presentaremos el primer prototipo, cuyos descriptores están basados en unigramas. A continuación, describiremos el segundo prototipo, basado en sintagmas terminológicos como descriptores de los clusters, y un conjunto de experimentos orientados a evaluar la adecuación de diferentes técnicas de selección de descriptores. Así mismo, realizaremos una comparativa con los resultados obtenidos en el primer prototipo. En tercer lugar, presentaremos el tercer prototipo, basado en n-gramas como descriptores de los clusters, y una serie de experimentos orientados a evaluar la adecuación de diferentes técnicas para la construcción del contexto formal. En concreto, aplicaremos Latent Semantic Indexing (LSI) para el enriquecimiento de las relaciones n-grama documento obtenidas y también consideraremos la adecuación de fragmentos de texto (snippets), en lugar de los documentos completos, como fuentes de información para llevar a cabo el proceso de extracción de descriptores. Finalmente, presentaremos el último prototipo desarrollado en esta Tesis Doctoral, el sistema JBraindead, que pretende demostrar la funcionalidad de nuestra propuesta sobre un sistema on-line capaz de organizar en tiempo real un conjunto de páginas web recuperadas a partir de consultas sobre Internet.

9.1. Primer prototipo

Nuestro primer prototipo utiliza unigramas como descriptores del conjunto de clusters generado.

9.1.1. Objetivos del prototipo

Los principales objetivos de este prototipo se exponen a continuación:

- Evaluar el grado de mejora en la tarea de acceso a la información relevante que produce un sistema de clustering basado en unigramas.

- Evaluar diferentes aproximaciones para la selección de descriptores con el fin de establecer una comparativa que permita decidir en cada caso por una aproximación concreta. Para ello, este primer prototipo implementa dos estrategias de selección. La primera de ellas está basada en la fórmula okapi, mientras que la segunda utiliza la fórmula terminológica para llevar a cabo la selección de descriptores.

- Evaluar el efecto que produce sobre la calidad del clustering el aumento en el número de descriptores inicialmente seleccionados. En este sentido, deseamos determinar aquellos casos en los que resultaría conveniente seleccionar un mayor número de descriptores sin reducir de manera significativa la calidad del clustering.

9.1.2. Características del prototipo

El prototipo ha sido implementado enteramente utilizando Java para programar el código del sistema y Swing para la realización de la interfaz de usuario. El procesado del corpus se ha llevado a cabo utilizando la base de datos Berkeley SleepyCat para almacenar la indexación directa e invertida. Las características funcionales del prototipo son las siguientes:

- *Proceso de recuperación de información.* El prototipo implementa su propio motor de búsqueda basado en el modelo del espacio vectorial. Se ha desarrollado un módulo para la realización de la indexación del corpus completo que utiliza Berkeley SleepyCat como base de datos para almacenar los índices obtenidos. El módulo de recuperación utiliza la fórmula del coseno (formula 6.8), aplicada sobre una representación de los documentos y de las consultas pesada de acuerdo a las fórmulas 6.2 y 6.3. En este prototipo todas las consultas son procesadas aplicando una disyunción sobre sus términos, lo que implica que el sistema recuperará todos aquellos documentos que contengan, al menos, algún término de la consulta. Para la realización del proceso, tanto de indexación como de recuperación, el sistema realiza un preprocesado de la colección eliminando las palabras vacías y realizando la extracción de las raíces de los términos restantes.

- *Proceso de extracción de descriptores.* El prototipo implementa un módulo de extracción de unigramas basado en la aproximación propuesta en la sección 6.3.1 sobre el texto completo de los documentos. Debido a que disponemos del corpus previamente indexado, en este caso el proceso de extracción puede realizarse directamente sobre los índices almacenados en la base de datos, sin que sea necesario realizar una indexación del conjunto de documentos recuperado.

- *Proceso de selección de descriptores.* El prototipo implementa dos aproximaciones distintas para la selección de descriptores. La primera de ellas se basa en la fórmula okapi descrita en la sección 6.4.3, donde los parámetros b, k y $avdl$ toman los valores 0.5, 1.2 y 300 respectivamente. Estos valores han sido obtenidos a partir de la aplicación experimental de esta fórmula sobre el corpus EFE94 en las campañas del CLEF [100]. Por otra parte, la segunda de las aproximaciones implementadas es la basada en la fórmula terminológica descrita en

la sección 6.4.4. Nótese como ambas aproximaciones requieren del conocimiento de ciertos parámetros globales de la colección, lo que implica que su aplicación sobre cualquier prototipo hace necesario conocer a priori el corpus completo.

- *Proceso de construcción del retículo.* La obtención del retículo se aborda desde una perspectiva de construcción directa del contexto K, asignando a cada documento únicamente aquellos descriptores contenidos de manera explícita.

Interfaz del Primer Prototipo

La figura 9.1 muestra la interfaz del primer prototipo desarrollado en este trabajo. En él pueden distinguirse cuatro partes:

- *El área de búsqueda*. Descrita por la etiqueta *Information Retrieval Area*, su objetivo es el de recoger la consulta realizada por el usuario, así como determinar el número de documentos que serán organizados, el número de descriptores extraídos y la estrategia de selección utilizada.

- *El área de navegación principal sobre el clustering*. Es el área que aparece descrita en la interfaz con la etiqueta *'Conceptual Tree'*. Como puede observarse, en este primer prototipo optamos por representar los retículos generados mediante un árbol. Las razones de esta decisión se deben a que entonces no habíamos abordado el problema de la visualización de los resultados, optando por un paradigma de visualización similar al utilizado por otros sistemas de clustering comerciales como Vivisimo [122] que posteriormente descartamos por las razones expuestas en la sección 6.7.

- *El área de navegación sobre el conjunto de documentos*. Es el área que aparece descrita por la etiqueta *'Related Documents Topics'*. En ella aparecen todos los documentos contenidos en el cluster actualmente seleccionado descritos por su título (el texto correspondiente a la etiqueta `<TITLE>` del documento XML) y un pequeño fragmento del texto obtenido del documento completo de manera automática. Obsérvese como el título de cada uno de los documentos se encuentra acompañado por una barra de información que representa el grado de similitud del documento con respecto a la consulta de acuerdo a los criterios de relevancia implementados en el motor de búsqueda.

- *El área de información del documento seleccionado*. Descrita por la etiqueta *'Current Document'*, contiene la información completa del documento actualmente seleccionado en el área de navegación sobre el conjunto de documentos. Nótese como la interfaz utiliza los metadatos XML del documento para destacar con diferente tipografía y estilo las distintas partes del mismo.

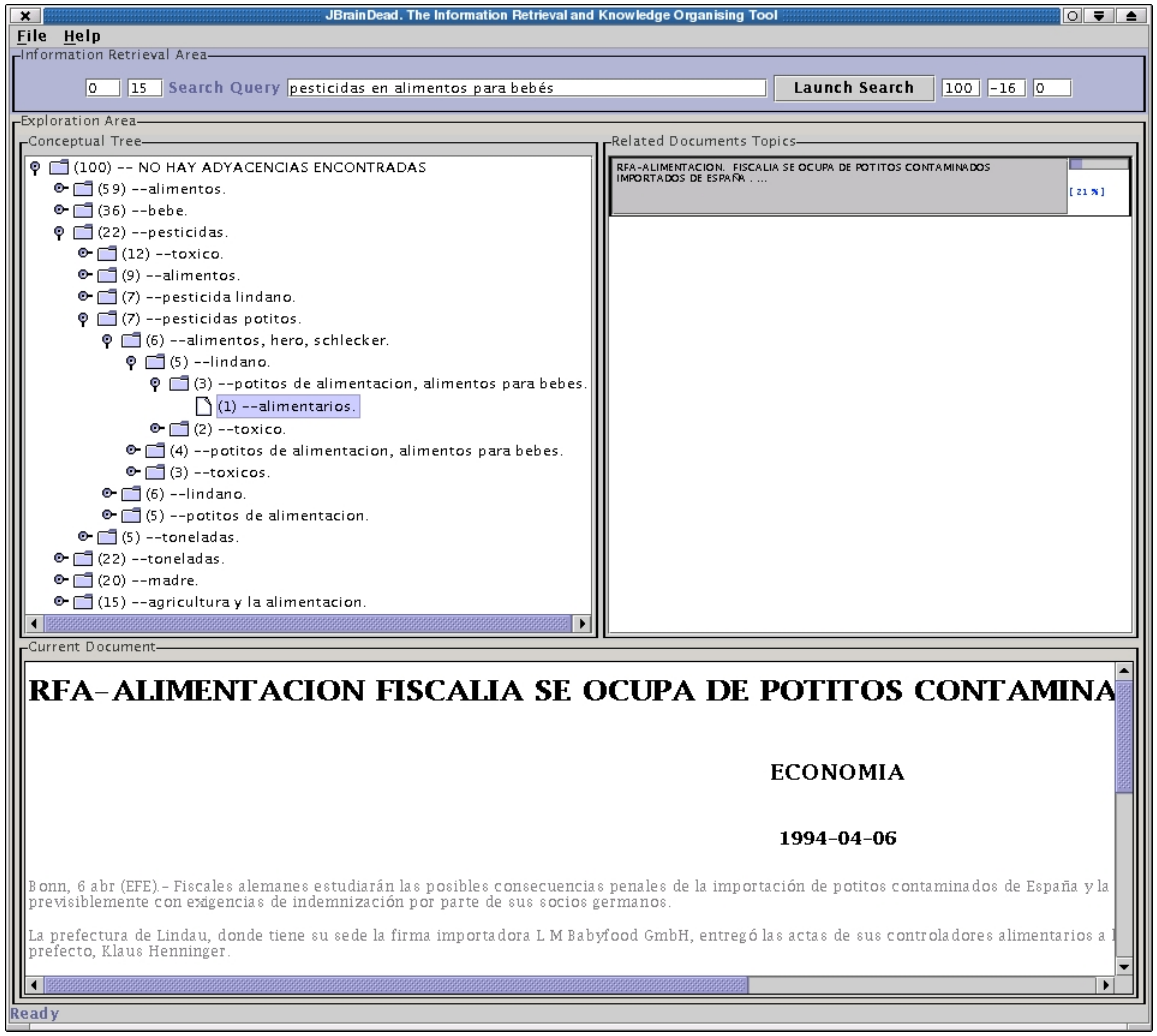


Figura 9.1: Interfaz del primer prototipo. Muestra los resultados del clustering correspondiente a la consulta '*pesticidas en alimentos para bebés*'

9.1.3. Experimentos

Los experimentos[1] propuestos en esta sección pretenden evaluar, sobre el prototipo presentado, la calidad de las estructuras generadas utilizando unigramas para representar el conjunto de clusters obtenido. Utilizando como marco de evaluación el conjunto de medidas propuesto en el capítulo 7 se realizaron los siguientes experimentos:

1. *Evaluación de la influencia del proceso de selección de descriptores sobre la calidad del clustering*. Dado que el prototipo presentado implementa las aproximaciones okapi y terminológica para realizar la selección de los descriptores, en este primer experimento se pretende evaluar la influencia de cada una de ellas sobre la calidad final del clustering. Variando la estrategia de selección pretendemos determinar, para un número fijo de documentos y de descriptores, cuál de las dos aproximaciones obtiene mejores estructuras de clustering. Tanto este experimento como los siguientes tendrán en cuenta el coste cognitivo (k) calculado para el escenario sobre el cual realizamos el experimento.

2. *Evaluación de la influencia del número de descriptores sobre la calidad del clustering*. Resulta lógico pensar que un aumento en el número de descriptores seleccionados debería producir estructuras de clustering mucho más precisas, caracterizadas por separar la información relevante de manera mucho más adecuada. El objetivo de este segundo experimento es determinar si esta apreciación es correcta y, en caso afirmativo, calcular el grado de mejora sobre la calidad de las estructuras de clustering generadas. En este segundo experimento se mantendrá constante el número de documentos sobre los cuales se lleva a cabo el proceso de clustering.

Evaluación de la influencia del proceso de selección

El objetivo del experimento presentado es el de cuantificar la influencia del proceso de selección de descriptores sobre la calidad de las estructuras de clustering obtenidas en el prototipo presentado. El experimento toma como base el conjunto de los 100 documentos más relevantes recuperados para el conjunto de consultas presentado en la sección 8.3, con una precisión media de $0{,}17$. Debemos destacar que las medidas de evaluación utilizadas trabajan a cobertura máxima ($recall = 1$), lo que significa considerar todos los documentos relevantes recuperados en el proceso de evaluación tanto para las estructuras de clustering generadas como para la lista de documentos inicialmente recuperada. Las estrategias de selección consideradas para la evaluación serán la estrategia $tf - idf$ (pesado okapi), ajustada a los parámetros expuestos en la sección 9.1.2, y la estrategia terminológica. El proceso de selección se realizará sobre los 10 descriptores más relevantes.

El cuadro 9.1 presenta los resultados obtenidos para ambas aproximaciones. Los valores de la medida DF son de $4{,}45$ y $6{,}8$ para las aproximaciones terminológica y okapi respectivamente, lo que

[1] Los resultados de las evaluaciones realizadas sobre los experimentos presentados tanto para este prototipo como para los de las siguientes secciones se han obtenido mediante una aproximación basada en macroaveraging. Es decir, promediando el observable sobre el conjunto de valores individuales obtenidos.

Estrategia de Selección	**DF**	**Num. MBA**	**Num. MVA**	k_{max}
Terminológico	4.45	11.26	19.36	7.02
Okapi	6.8	16.23	31.89	5.16

Cuadro 9.1: Resultados del experimento para la evaluación del proceso de selección de descriptores sobre unigramas. Precisión base del proceso de recuperación $0{,}17$ y número de descriptores seleccionados $|DESC| = 10$

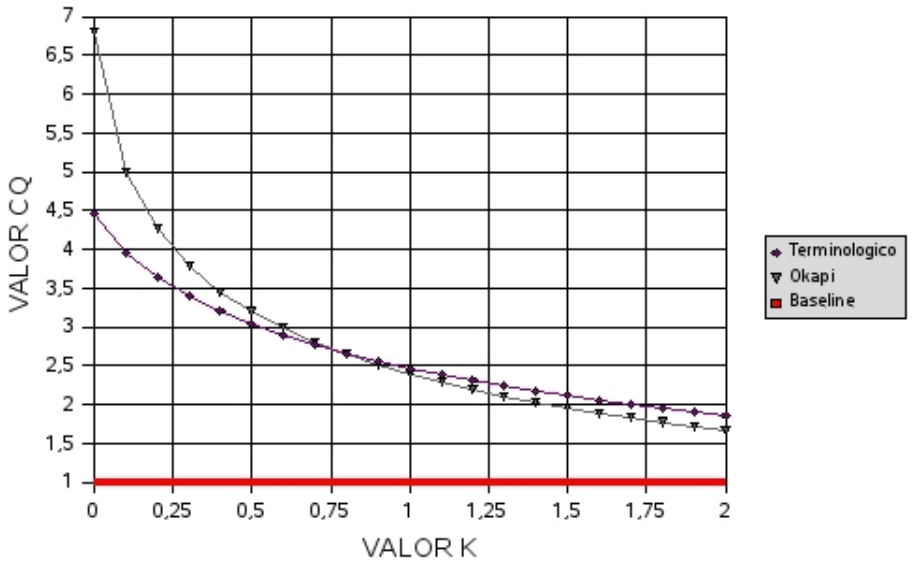


Figura 9.2: Evolución de la medida CQ para las aproximaciones de selección de descriptores terminológica y okapi sobre unigramas. Precisión base del proceso de recuperación $0{,}17$ y número de descriptores seleccionados $|DESC| = 10$

supone una mejora del $52{,}8\,\%$ de okapi respecto a la aproximación terminológica. El no considerar el coste cognitivo asociado a los descriptores de los clusters (medida DF) se traduce en que una estrategia de selección basada en la fórmula okapi permite generar estructuras de clustering mucho más precisas. Debemos remarcar, sin embargo, que ambas medidas obtienen una mejora sustancial con respecto a la precisión inicial del conjunto de documentos recuperados multiplicando por un factor promedio de $4{,}45$ y $6{,}8$ este valor.

La diferencia en el grado de mejora aportado por la aproximación okapi frente a la aproximación terminológica debe buscarse en la propia naturaleza de los pesados aplicados en el proceso de selección. Mientras que los descriptores obtenidos mediante okapi pueden caracterizarse por ser de propósito general (esta fórmula realiza un pesado $tf - idf$), los obtenidos por la fórmula terminológica son específicos y característicos del conjunto de documentos recuperado. Esto supone que, en el caso de aplicar la aproximación terminológica, el conjunto de descriptores seleccionado afectará a un menor número de documentos y, por lo tanto, la frecuencia de documento asociada a cada descriptor será pequeña. Esto supone retículos muy sencillos caracterizados por disponer de un nodo raíz con una gran cantidad de documentos en su extensión. En contraste, las estructuras obtenidas aplicando la fórmula okapi se caracterizan por disponer de un número de nodos más elevado, con un mayor grado de interrelación y por tener un nodo raíz con muchos menos documentos. Podemos concluir que los descriptores seleccionados mediante la estrategia okapi producen estructuras donde la información recuperada se distribuye de manera mucho más uniforme, facilitando el acceso a los documentos relevantes a través del MBA.

No obstante, la mejora de precisión obtenida mediante una selección okapi implica un aumento en el número de nodos que el usuario debe considerar para acceder a la información relevante que, en última instancia, podría perjudicar la calidad final de las estructuras de clustering obtenidas (19.36 nodos frente 31.89). Por esta razón se hace necesario un estudio más preciso que tenga en cuenta estos factores para cuantificar la calidad de las estructuras finalmente obtenidas en función del coste cognitivo asociado al escenario sobre el cual se lleva a cabo el proceso de clustering.

La figura 9.2 muestra la evolución de la medida CQ considerando diferentes valores de k. Ambas gráficas presentan un decrecimiento que, en el caso de la estrategia okapi, es mucho más pronunciado. Esto es debido a que okapi genera estructuras de clustering con más nodos. En este sentido, podemos considerar que la estrategia okapi será mucho más sensible a las variaciones del parámetro k. Los valores k_{max} se muestran en el cuadro 9.1 y representan los valores máximos de k que justifican el uso de técnicas de clustering frente a la exploración directa de la lista recuperada.

En la figura podemos distinguir dos intervalos bien diferenciados que suponen el cruce de las curvas CQ. En aquellos escenarios para los cuales $k < 0{,}8$, la estrategia okapi produce estructuras de clustering con una mejor calidad que las obtenidas mediante la estrategia terminológica. En estos casos, la mejora de precisión obtenida mediante okapi primaría frente al numero de nodos que el usuario debe considerar para acceder a la información relevante. No obstante, debemos destacar como para valores dentro del intervalo $k \in [0{,}4, 0{,}8]$, la ganancia de calidad entre ambas aproximaciones es muy pequeña (mejora de $CQ < 7{,}5\,\%$ como máximo), haciendo necesario considerar factores externos a la propia estructura del clustering, tales como la calidad informativa de los descriptores o el coste computacional asociado al proceso, para determinar cual de las dos estrategias de selección

es la más adecuada para la realización del clustering. En contraste, en aquellos escenarios donde $k > 0{,}8$, la influencia del coste cognitivo sobre el numero de nodos que el usuario debe considerar se hace más patente, lo que implica una mejor adecuación de la estrategia terminológica para la construcción de las estructuras de clustering.

El valor estimado para k en la colección utilizada para los experimentos es de 0.64 (sección 8.4), lo que supone valores de la medida CQ muy próximos para ambas aproximaciones ($CQ \simeq 2{,}8$). De acuerdo a estos resultados, podemos concluir que, en nuestro escenario de recuperación, ambas aproximaciones pueden considerarse igualmente válidas, siendo necesario tener en cuenta, tal y como ya hemos expuesto, factores externos a las propias estructuras para realizar la selección de una de ellas.

Evaluación de la influencia del número de descriptores

El objetivo del segundo experimento realizado sobre el prototipo presentado es el de determinar la influencia del número de descriptores seleccionados sobre la calidad final de las estructuras de clustering obtenidas. El experimento toma como base el conjunto de los 100 documentos más relevantes recuperados para el conjunto de consultas presentado en la sección 8.3, con una precisión media de $0{,}17$ y considerando, al igual que en el experimento anterior, que la evaluación se realiza a cobertura máxima. La estrategia de selección considerada para la evaluación será la estrategia terminológica sobre 10, 15 y 20 descriptores.

El cuadro 9.2 muestra los resultados obtenidos para las tres aproximaciones. Los valores para la medida DF suponen una mejora de la precisión con respecto a la lista de documentos original considerable (en un factor $4{,}45$, $5{,}93$ y $6{,}94$ respectivamente), siendo destacable el resultado obtenido sobre 20 descriptores. Estos valores son debidos principalmente a dos razones:

- Un aumento en el número de descriptores supone un aumento en el número de documentos descritos por, al menos, un descriptor. Esta situación implica una disminución de la población del nodo raíz, con la consiguiente mejora de precisión en el cálculo de la medida DF.
- Un aumento en el número de descriptores supone la generación de estructuras de clustering mucho más ricas, caracterizadas por especializar documentos que, en aproximaciones sobre menos descriptores, son agrupados en clusters genéricos. En estos casos existe una mayor probabilidad de mezclar información relevante con información no relevante.

No obstante, la mejora de las estructuras de clustering debido al aumento del número de descriptores produce irremediablemente un aumento en el número de nodos generados que hace necesario un estudio más exhaustivo que permita determinar hasta que punto esta mejora afecta negativamente a la calidad final del clustering.

La figura 9.3 muestra las curvas CQ correspondientes a las tres aproximaciones. Del estudio de las tres gráficas podemos extraer las siguientes conclusiones:

- Las tres gráficas decrecen, aunque el decrecimiento de las gráficas correspondientes a 15 y 20 descriptores es mucho más pronunciado. Al igual que ocurría en el experimento anterior,

Estrategia de Selección	DF	Num. MBA	Num. MVA	k_{max}
10 Descriptores	4.45	11.26	19.36	7.02
15 Descriptores	5.93	20.23	41.11	5.94
20 Descriptores	6.94	28.09	65.34	5.31

Cuadro 9.2: Resultados del experimento para la evaluación de la influencia del número de descriptores seleccionados mediante una estrategia terminológica sobre unigramas. Precisión base del proceso de recuperación 0,17. Número de descriptores seleccionados 10, 15 y 20

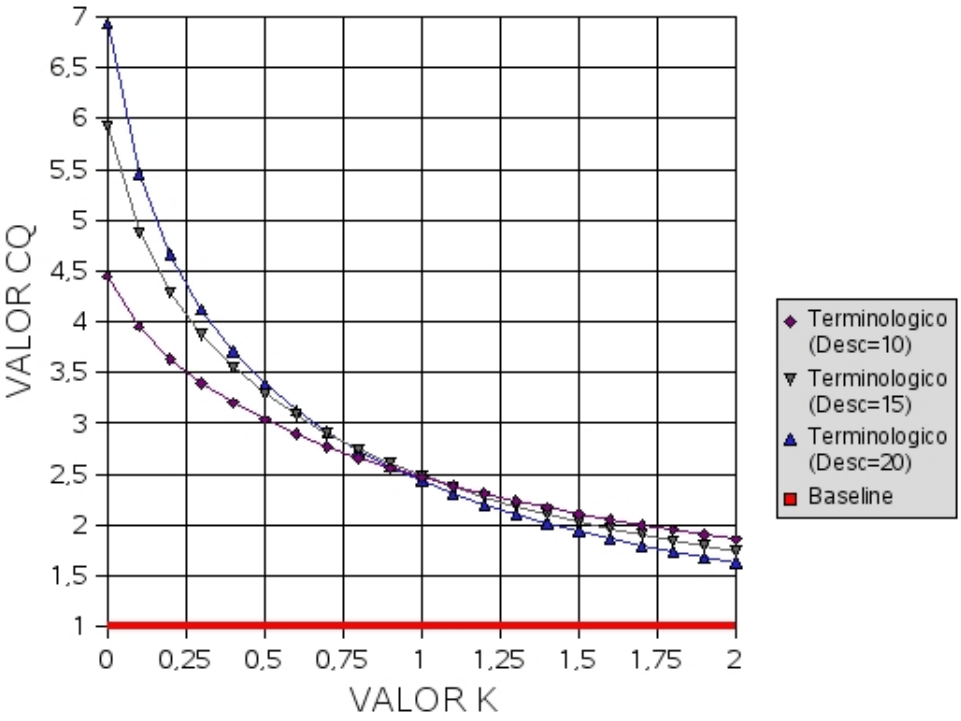


Figura 9.3: Evolución de la medida CQ para la aproximación de selección de descriptores terminológica sobre 10, 15 y 20 descriptores. Precisión base del proceso de recuperación 0,17

esto es debido a que en ambas aproximaciones el número de nodos del MVA es superior y, por lo tanto, la calidad de ambas aproximaciones es mucho más sensible a las variaciones del parámetro k.

- Las aproximaciones para 15 y 20 descriptores transcurren muy próximas, de hecho sus valores k_{max} son muy similares. Dado que la mejora de la medida DF es del $33{,}2\,\%$ respecto a la selección de 10 y 15 descriptores y del $17\,\%$ respecto a seleccionar 15 y 20 descriptores, el aumento en el número de nodos que el usuario debe considerar en el caso de seleccionar 20 descriptores (de un $38{,}8\,\%$ en el MBA y de un $58{,}9\,\%$ en el MVA respecto a seleccionar 15 descriptors) no justifica el aumento de precisión obtenido. Como consecuencia, utilizar una aproximación de selección sobre 15 descriptores permitiría obtener unos valores de calidad similares, reduciendo el coste computacional asociado al proceso de construcción del clustering. Este resultado nos permite intuir que el aumento en el número de descriptores no produce un aumento proporcional sobre la calidad del clustering para $k \neq 0$, lo que implica la existencia de un cierto umbral a partir del cual no resulta adecuado considerar un número mayor de descriptores para la realización del clustering.

- Para valores de $k > 1$ la aproximación sobre 10 descriptores produce los mejores resultados de calidad. Esto es debido a que la influencia del número de nodos a inspeccionar es mucho más relevante y, por lo tanto, priman aquellas estrategias de selección que, aún teniendo una menor precisión, no generen un número muy elevado de clusters.

Sobre nuestro escenario de recuperación, el valor estimado para $k = 0{,}64$ nos permite concluir que una estrategia de selección sobre 15 descriptores obtiene mejores resultados de calidad en comparación con la misma estrategia de selección aplicada únicamente sobre 10 descriptores.

Finalmente, y dado que en el experimento anterior hemos obtenido la curva CQ correspondiente a la estrategia de selección okapi aplicada sobre 10 descriptores, resulta interesante realizar un estudio que nos permita comparar los resultados obtenidos para esta aproximación frente a los obtenidos aplicando una aproximación terminológica sobre 15 descriptores. La figura 9.4 y el cuadro 9.3 muestran esta comparativa, donde pueden observarse las curvas CQ correspondientes a ambas aproximaciones. Los resultados indican como, para valores de $k \neq 0$, ambas gráficas evolucionan muy próximas, sin que exista una diferencia significativa de calidad entre ellas para algún valor de k. En este sentido, podría parecer lógico concluir que la selección de cualquiera de las dos aproximaciones sería adecuada sobre un sistema de clustering. Sin embargo, y dado que la estrategia terminológica se aplica sobre un mayor número de descriptores, consideramos que ésta sería la más adecuada. En contraste con el experimento anterior, en este caso el aumento en el número de nodos no resulta ser tan significativo (un $24\,\%$ en el MBA y un $28{,}9\,\%$ en el MVA respecto a la estrategia de selección okapi para 10 descriptores). Por esta razón, en este caso el aumento en el número de nodos se vería justificado por el aumento de la calidad informativa del clustering.

Estrategia de Selección	DF	Num. MBA	Num. MVA	k_{max}
Terminológico 15 Descriptores	5.93	20.23	41.11	5.94
Okapi 10 Descriptores	6.8	16.23	31.89	5.16

Cuadro 9.3: Resultados del experimento para la evaluación de la influencia del número de descriptores seleccionados mediante una estrategia terminológica (seleccionando 15 descriptores) y okapi (seleccionando 10 descriptores) sobre unigramas. Precisión base del proceso de recuperación $0{,}17$.

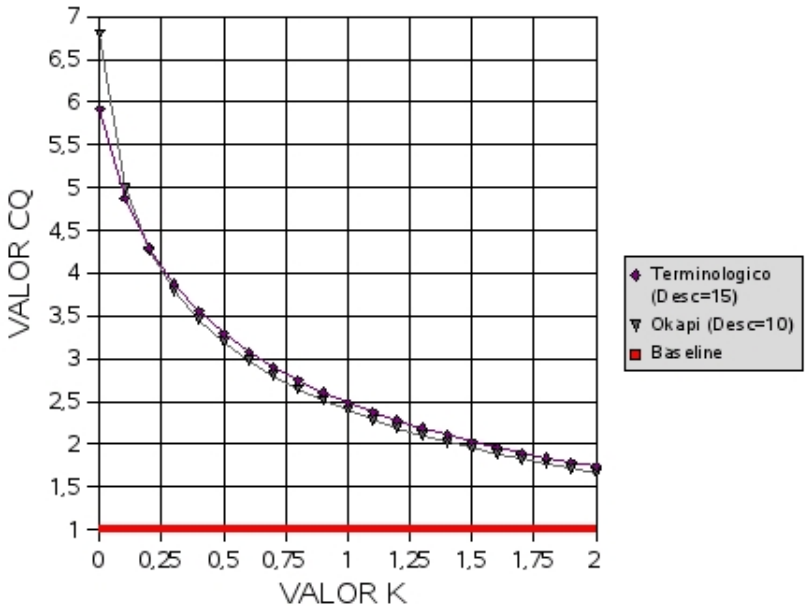


Figura 9.4: Evolución de la medida CQ para la aproximación de selección de descriptores terminológica sobre 15 descriptores y la aproximación okapi sobre 10 descriptores. Precisión base del proceso de recuperación $0{,}17$

9.1.4. Recapitulación

Los resultados obtenidos para la evaluación del primer prototipo, estimando un valor de $k = 0{,}64$, se presentan a continuación:

- Para un número de descriptores fijo (10 descriptores), Los valores de calidad obtenidos para ambas estrategias de selección de descriptores fueron muy similares, lo que implica considerar factores externos, tales como la calidad informativa de los descriptores seleccionados o el coste computacional asociado al proceso de construcción del clustering, para determinar cual de las dos propuestas es la más adecuada.

- Considerando un número de descriptores variable (10, 15 y 20 descriptores) sobre la estrategia de selección terminológica hemos demostrado como el aumento en el número de descriptores produce retículos de mayor precisión. No obstante, hemos podido comprobar como este aumento no es proporcional al aumento en la calidad de las estructuras de clustering generadas, lo que induce a pensar que existe un cierto umbral para el cual no resulta adecuado aumentar el número de descriptores para llevar a cabo la construcción del clustering. Sobre nuestro escenario de recuperación, la aproximación más adecuada para la realización del proceso de clustering se obtuvo seleccionando 15 descriptores.

- Considerando diferentes estrategias de selección sobre un diferente número de descriptores, hemos podido comprobar como valores similares de calidad, y no siendo significativo el aumento del número de nodos en el MBA y en el MVA, resulta más adecuado optar por aquella estrategia de selección que involucre un mayor número de descriptores. Esto es debido a que éstas producen estructuras de clustering con una mayor capacidad informativa.

- Finalmente, debemos decir que en todos los experimentos se produjeron mejoras de la calidad con respecto a la lista ordenada originalmente devuelta por el motor de búsqueda (es decir $CQ > 1$), lo que supone la adecuación de las distintas aproximaciones de clustering a la tarea de recuperación propuesta.

9.2. Segundo Prototipo

El segundo prototipo presentado en esta Tesis Doctoral realiza una implementación del modelo propuesto utilizando sintagmas terminológicos como descriptores del conjunto de clusters generado.

9.2.1. Objetivos

El diseño del prototipo presentado pretende alcanzar los objetivos expuestos a continuación:

- Evaluar el grado de mejora en la tarea de recuperación que produce un sistema de clustering basado en sintagmas terminológicos.

- Evaluar diferentes aproximaciones para la selección de descriptores sobre sintagmas terminológicos con el fin de establecer una comparativa que permita decidir en cada caso por una aproximación concreta. Se implementan dos estrategias de selección. La primera de ellas está basada en la aproximación terminológica que ya ha sido utilizada en el primer prototipo, mientras que la segunda utiliza la aproximación balanceada presentada en la sección 6.4.5.
- Evaluar el efecto de la población del cluster raíz sobre la calidad final del clustering. Con este fin, el prototipo presentado permite asignar a un cluster de propósito general (diferente al cluster raíz y que denominaremos *miscelánea*) todos los documentos que no se encuentren relacionados con alguno de los descriptores seleccionados, generando una estructura de clustering modificada donde la extensión del cluster raíz no contendrá documento alguno. La evaluación de estos nuevos retículos permitirá cuantificar el grado en el que el proceso de clustering aísla adecuadamente la información relevante.
- Evaluar el efecto que sobre la calidad del clustering tiene la tipología del descriptor utilizado para representar los clusters. Con este fin se pretende realizar una comparativa entre los resultados obtenidos por las aproximaciones basadas en sintagmas terminológicos que implementa el prototipo y los resultados obtenidos en el primer prototipo (sección 9.1.3) considerando unigramas como descriptores de los clusters.

9.2.2. Características del prototipo

El prototipo ha sido implementado enteramente utilizando Java para programar el código del sistema y Swing para la realización de la interfaz de usuario. El procesado del corpus se ha llevado a cabo utilizando la base de datos Berkeley SleepyCat para almacenar la indexación directa e invertida. Las características funcionales del prototipo son las siguientes:

- *Proceso de recuperación de información.* El prototipo implementa su propio motor de búsqueda basado en el modelo del espacio vectorial. Se ha desarrollado un módulo para la realización de la indexación del corpus completo que utiliza Berkeley SleepyCat como base de datos para almacenar los índices obtenidos. El módulo de recuperación utiliza la fórmula del coseno (formula 6.8), aplicada sobre una representación de los documentos y de las consultas pesada de acuerdo a las fórmulas 6.2 y 6.3. En este prototipo las consultas pueden procesarse aplicando una disyunción o una conjunción sobre sus términos según lo decida el usuario a través de la interfaz. No obstante, debemos remarcar que para los experimentos que presentaremos en la siguiente sección se optó por procesar las consultas de acuerdo a la disyunción de sus términos con el fin de obtener la misma precisión base en el proceso de recuperación. Para la realización del proceso, tanto de indexación como de recuperación, el sistema realiza un preprocesado de la colección eliminando las palabras vacías y realizando la extracción de las raíces de los términos restantes.
- *Proceso de extracción de descriptores.* En este caso, el prototipo trabaja sobre un conjunto de sintagmas terminológicos previamente extraídos de la colección EFE94 y que hemos indexado en tablas independientes sobre la base de datos SleepyCat. Para ello hemos utilizado

directamente el conjunto de sintagmas terminológicos que fueron extraídos sobre la colección de prueba para el trabajo [96]. Debemos remarcar que su proceso de extracción fue llevado a cabo considerando el texto completo de los documentos.

- *Proceso de selección de descriptores.* El prototipo implementa dos aproximaciones distintas para la selección de descriptores. En concreto, se ha implementado una aproximación terminológica sobre sintagmas, similar a la implementada en el prototipo anterior, y una aproximación basada en el algoritmo balanceado presentado en la sección 6.4.5. En este prototipo no se ha implementado la estrategia de selección okapi debido a que no disponíamos de los parámetros correspondientes ajustados a los valores adecuados adaptado a la colección EFE94 para pesar sintagmas en lugar de unigramas.

- *Proceso de construcción del retículo.* La obtención del retículo se aborda desde una perspectiva de construcción directa del contexto K, asignando a cada documento únicamente aquellos descriptores contenidos de manera explícita.

Interfaz del Segundo Prototipo

La figura 9.5 muestra la interfaz del segundo prototipo desarrollado en este trabajo. En él pueden distinguirse cinco partes bien diferenciadas:

- *El área de búsqueda*. Descrita en por la etiqueta *'Search Controls'*. Su funcionalidad es la de recoger la consulta del usuario, permitiendo elegir la estrategia de selección de descriptores, el número de documentos a recuperar, el número de descriptores seleccionados y el tipo de búsqueda realizada (considerando la conjunción de todos los términos de la consulta o su disyunción).

- *El área de navegación principal sobre el clustering*. Es el área que aparece descrita en la interfaz por la etiqueta *'Main Browser'*. En este segundo prototipo implementamos una aproximación para la visualización del clustering basada en retículos de acuerdo a la propuesta presentada en la sección 6.7. En esta representación se destacan el cluster actualmente seleccionado y los enlaces hacia el conjunto de clusters de su up-set en un color diferente. Así mismo, se han utilizado diferentes iconos para representar la tipología concreta del cluster, en concreto, la interfaz diferencia entre los clusters que son concepto objeto (representados por un icono circular con un documento)y los que no lo son (representados únicamente por el icono). De esta manera, el usuario puede diferenciar de manera sencilla los nodos que contienen información (documentos) y aquellos que, no conteniendo documento alguno, permiten especializar la búsqueda hacia clusters más específicos con información.

- *El área de camino actualmente seleccionado*. Descrita por la etiqueta *'Attributes Selected'*. Representa la intensión completa del nodo actualmente seleccionado.

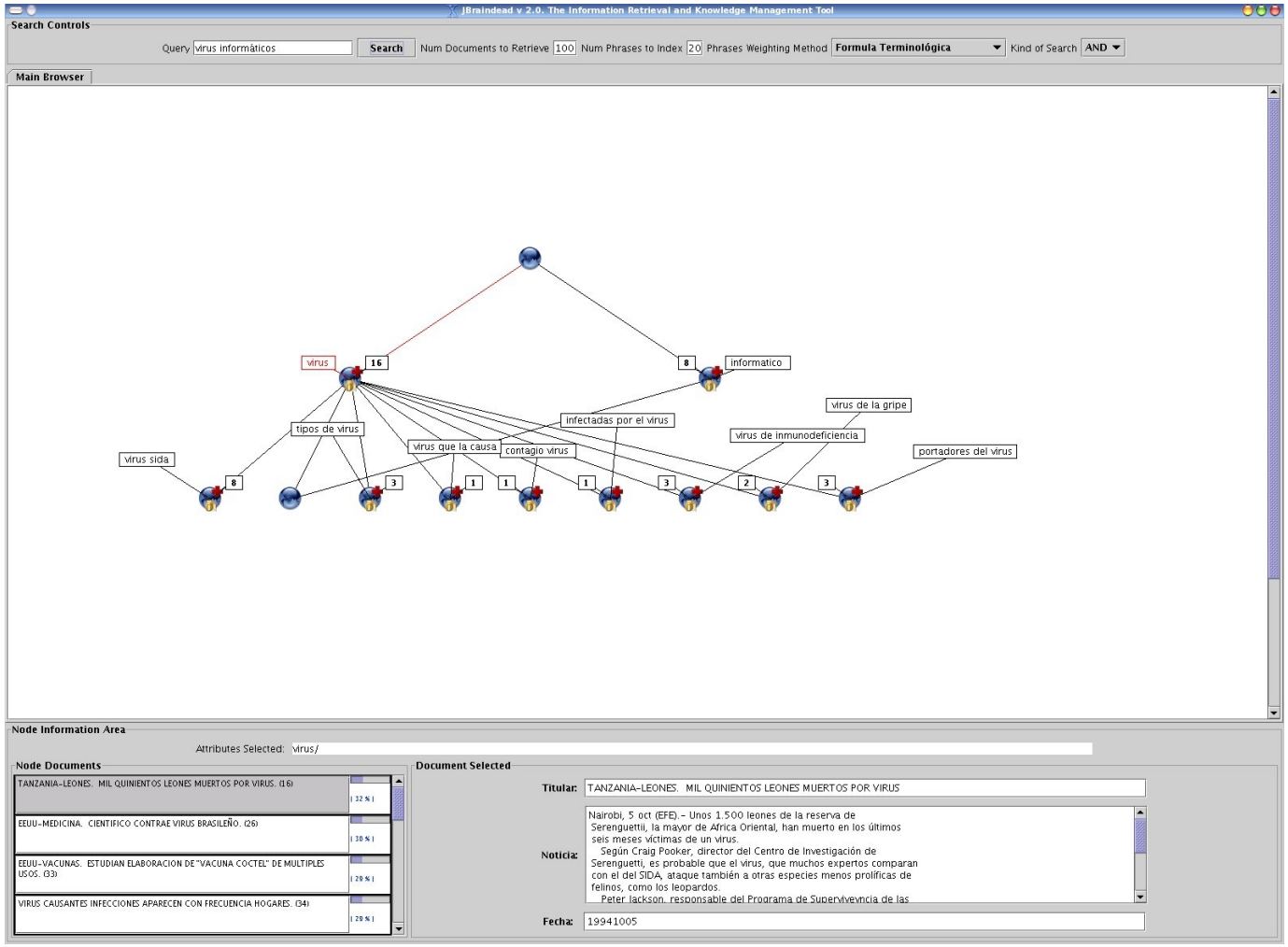


Figura 9.5: Interfaz del segundo prototipo. Muestra los resultados del clustering correspondiente a la consulta '*virus informáticos*'

- *El área de navegación sobre el conjunto de documentos*. Es el área que aparece descrita por la etiqueta *'Node Documents'*, su diseño y funcionalidad coinciden con los descritos para el prototipo anterior.

- *El área de información del documento seleccionado*. En la interfaz aparece descrita por la etiqueta *'Selected Document'* y representa la información del documento actualmente seleccionado. Su diseño y funcionalidad son los mismos que los descritos para el prototipo anterior.

9.2.3. Experimentos

Los experimentos propuestos en esta sección pretenden evaluar, sobre el prototipo presentado, la calidad de las estructuras generadas utilizando sintagmas terminológicos para representar el conjunto de clusters obtenido. Utilizando como marco de evaluación el conjunto de medidas propuesto en el capítulo 7, se realizaron los siguientes experimentos:

1. *Evaluación de la influencia del proceso de selección de descriptores sobre la calidad del clustering*. Este primer experimento sobre el prototipo presentado pretende evaluar la influencia de las estrategias de selección terminológica y balanceada sobre la calidad final de las estructuras de clustering generadas. Para ellos, se presenta un experimento que, variando la estrategia de selección, determine para un número fijo de documentos y de descriptores cual de las dos aproximaciones obtiene mejores estructuras de clustering en función del parámetro k asociado al escenario sobre el cual se lleva a cabo el proceso.

2. *Evaluación de la influencia de la población del cluster raíz sobre la calidad del clustering*. Tal y como expusimos en el capítulo 7, las medidas de evaluación propuestas se basan en el MBA para obtener los valores de calidad de las estructuras de clustering generadas. Esta estructura se caracteriza por contener siempre el nodo raíz del clustering, debido a que el usuario siempre deberá visitarlo al realizar la exploración del clustering. En este sentido, el experimento propuesto pretende determinar si los documentos contenidos en este cluster pueden influir negativamente en los valores DF y CQ obtenidos sobre las estructuras de clustering.

3. *Comparativa entre la calidad de las estructuras de clustering basadas en unigramas y sintagmas terminológicos*. El último experimento que presentamos pretende determinar cual es la influencia sobre la calidad del clustering de la tipología de los descriptores utilizados para representar los clusters generados. En este sentido, el experimento plantea una comparativa entre los resultados obtenidos sobre unigramas, para el prototipo anterior, y lo resultados que obtendremos sobre sintagmas terminológicos en este segundo prototipo.

Evaluación de la influencia del proceso de selección

El objetivo del experimento presentado es el de cuantificar la influencia del proceso de selección de descriptores sobre la calidad de las estructuras de clustering obtenidas en el prototipo presentado. El experimento toma como base el conjunto de los 100 documentos más relevantes recuperados para

Estrategia de Selección	DF	Num. MBA	Num. MVA	k_{max}
Terminológico	1.17	4.04	9.16	1.17
Balanceado	3.56	20.56	51.84	5.31

Cuadro 9.4: Resultados del experimento para la evaluación del proceso de selección de descriptores sobre sintagmas terminológicos. Precisión base del proceso de recuperación $0{,}17$ y número de descriptores seleccionados $|DESC| = 10$

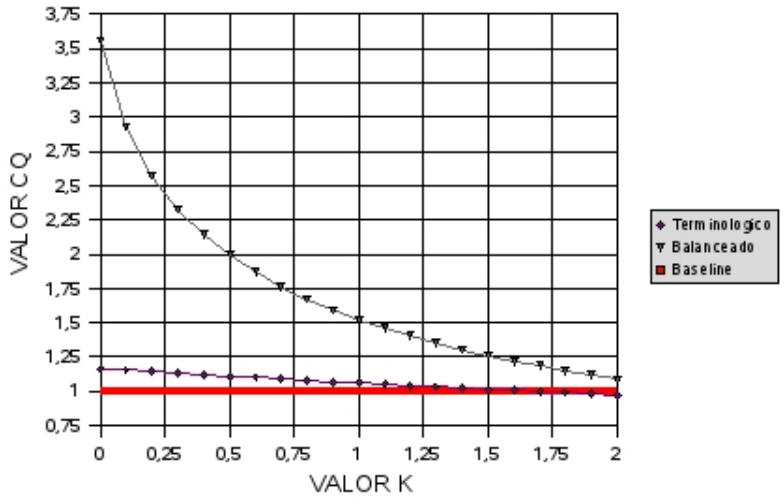


Figura 9.6: Resultados del experimento para la evaluación del proceso de selección de descriptores sobre sintagmas terminológicos. Precisión base del proceso de recuperación $0{,}17$ y número de descriptores seleccionados $|DESC| = 10$

el conjunto de consultas presentado en la sección 8.3, con una precisión media de $0{,}17$. Debemos destacar que las medidas de evaluación utilizadas trabajan a cobertura máxima ($recall = 1$), lo que significa considerar todos los documentos relevantes recuperados en el proceso de evaluación tanto para las estructuras de clustering generadas como para la lista de documentos inicialmente recuperada. Las estrategias de selección consideradas para la evaluación serán la estrategia terminológica y la estrategia balanceada, presentada en la sección 6.4.5, sobre 10 descriptores.

El cuadro 9.4 muestra los resultados obtenidos en el experimento para las dos aproximaciones propuestas. Podemos observar como, en el caso de no considerar el coste cognitivo asociado a los descriptores, la estrategia balanceada produce el mejor valor para la medida DF, siendo éste notablemente superior al obtenido por la estrategia terminológica ($3{,}56$ frente a $1{,}17$, lo que supone una mejora del $204{,}3\,\%$). De hecho, una mejora de precisión tan baja de la aproximación terminológica no justifica su coste computacional frente a la exploración completa del conjunto de documentos recuperado. La explicación a los resultados obtenidos se encuentra nuevamente en la baja frecuencia de documento de los descriptores seleccionados que, en el caso de trabajar con sintagmas terminológicos, se acentúa mucho más. Esto es debido a que los sintagmas terminológicos, y en general cualquier secuencia de más de un término, presentan frecuencias de documento muy inferiores a las obtenidas por unigramas. De este modo, la aplicación de una estrategia terminológica al proceso de selección escogerá un conjunto de sintagmas muy específicos que, como consecuencia, presentarán unas bajas frecuencias de documento. Este fenómeno se refleja claramente en los valores para $|MVA|$ y $|MBA|$ que indican la obtención de retículos excesivamente pequeños mediante esta aproximación.

La figura 9.6 muestra las curvas CQ correspondientes a cada una de las aproximaciones evaluadas. Podemos observar como, a lo largo de todo el intervalo considerado para k, la estrategia balanceada se encuentra claramente por encima de la estrategia terminológica. Esto indica que esta primera opción sería la más adecuada para construir un sistema de clustering sobre sintagmas terminológicos. Nótese, además, como el valor de k_{max} obtenido para la aproximación balanceada es cinco veces superior al obtenido para la estrategia terminológica, lo que indica que en el caso balanceado el rango de valores permitido para obtener valores $CQ > 1$ es muy superior. Para el valor de k estimado sobre nuestro escenario ($k = 0{,}4$), el valor de CQ obtenido sería de 2.10, lo que indica que las estructuras de clustering obtenidas mediante esta aproximación duplicarían la calidad con respecto a la lista original de documentos recuperados.

Evaluación de la población del cluster raíz

El objetivo de este segundo experimento es el de cuantificar la influencia del número de documentos contenidos en el cluster raíz sobre la calidad final de las estructuras de clustering generadas. Con este fin, el experimento obtiene estructuras de clustering alternativas caracterizadas por disponer de un cluster de propósito general (que denominaremos cluster *Dummy* o *Miscelánea*) que contenga todos aquellos documentos que, por no disponer de descriptor asociado alguno, son asignados al cluster raíz. El proceso de generación de estas estructuras se ha llevado a cabo asignando un descriptor genérico a todos aquellos documentos que, en el proceso de generación del contexto formal K, no estén relacionados con alguno de los descriptores finalmente seleccionados. El experimento toma

Estrategia de Selección	DF	Num. MBA	Num. MVA
Balanceado+Dummy	9.64	20.89	53.33
Balanceado	3.56	20.56	51.84

Cuadro 9.5: Resultados del experimento para la evaluación del proceso de selección de descriptores considerando la aproximación balanceada genérica con y sin nodo de propósito general. Precisión base del proceso de recuperación $0{,}17$ y número de descriptores seleccionados $|DESC| = 10$

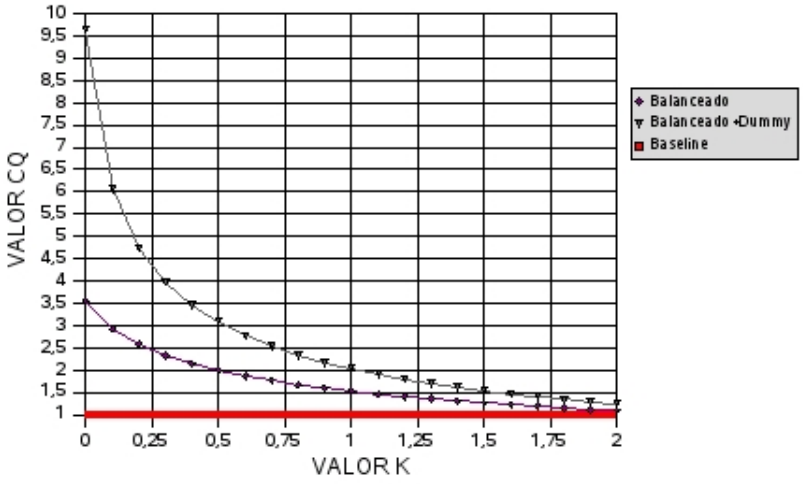


Figura 9.7: Resultados del experimento para la evaluación de la influencia del cluster raíz sobre la calidad del clustering generado. Precisión base del proceso de recuperación $0{,}17$ y número de descriptores seleccionados $|DESC| = 10$

como base el conjunto de los 100 documentos más relevantes recuperados para el conjunto de consultas presentado en la sección 8.3, con una precisión media de $0{,}17$. La estrategia de selección considerada para la evaluación será la estrategia balanceada sobre 10 descriptores.

El cuadro 9.5 muestra los resultados obtenidos en el experimento. Como puede observarse, la mejora en la medida DF en el caso de considerar un cluster de propósito general es muy significativa, llegando prácticamente a triplicar el valor DF obtenido por la aproximación que no considera este cluster (mejora del $170{,}8\,\%$). Las conclusiones que podemos obtener a partir de estos resultados resultan evidentes, y dejan patente que la calidad de las estructuras de clustering obtenidas mediante estrategia balanceada de selección aislan adecuadamente la información relevante. Esto es debido a que considerar el cluster de propósito general supone generar un nuevo nodo, que será hijo del cluster raíz, y que únicamente deberá ser considerado en el MBA en el caso de contener algún documento relevante. En este sentido, el grado de mejora obtenido para la medida DF indica que, en la mayor parte de las ocasiones, este nodo no tuvo que ser considerado en la construcción del MBA, lo que significa que los descriptores seleccionados fueron suficientemente adecuados como para aislar correctamente la información relevante. Obviamente, en el caso contrario el cluster raíz habría contenido una mezcla de información relevante con información no relevante, que habría provocado la generación de conjuntos MBA muy similares a los obtenidos en el caso de no considerar el cluster de propósito general, con la consiguiente similitud en los valores DF obtenidos.

La figura 9.7 muestra las curvas CQ correspondientes a ambas aproximaciones. La curva correspondiente a la aproximación que considera el cluster de propósito general obtiene los mejores valores CQ para todo el intervalo de k considerado, siendo su valor específico para nuestro escenario ($k = 0{,}4$) de 3.45 frente a 2.14 en el caso de no considerarlo, lo que supone una mejora del $61{,}21\,\%$ en la calidad del clustering.

Comparativa entre unigramas y sintagmas terminológicos

El objetivo del último experimento presentado es el de realizar una comparativa entre las estructuras de clustering descritas por sintagmas terminológicos y las descritas por unigramas, permitiendo determinar sobre un mismo escenario cual de ellas es la más adecuada para generar estructuras de clustering de calidad. En este sentido, el experimento realizará la comparación sobre los resultados obtenidos en el primer prototipo con una estrategia de selección basada en okapi sobre 10 descriptores y los resultados que acabamos de presentar para el segundo prototipo con una estrategia de selección basada en la aproximación balanceada sobre 10 descriptores generando un cluster de propósito general (cluster *dummy* o *miscelánea*). El experimento toma como base el conjunto de los 100 documentos más relevantes recuperados para el conjunto de consultas presentado en la sección 8.3, con una precisión media de $0{,}17$.

La figura 9.8 y el cuadro 9.6 muestran las curvas CQ correspondientes a ambas aproximaciones. Como puede observarse, en el caso de no considerar el coste cognitivo asociado a los descriptores, la aproximación sobre sintagmas terminológicos produce mejores resultados para la medida DF que la aproximación sobre unigramas (en concreto, una mejora del $41{,}8\,\%$), generando retículos caracterizados por aislar la información relevante de manera mucho más precisa. No obstante, y debido a que el coste cognitivo no tiene porqué ser el mismo para los unigramas que para los sintagmas ter-

Estrategia de Selección	DF	Num. MBA	Num. MVA
Balanceado+Dummy (sintagmas)	9.64	20.89	53.33
Okapi (unigramas)	6.8	16.23	31.89

Cuadro 9.6: Resultados del experimento para la evaluación del proceso de selección de descriptores considerando la aproximación balanceada con nodo de propósito general sobre sintagmas y la aproximación okapi sobre unigramas. Precisión base del proceso de recuperación $0{,}17$ y número de descriptores seleccionados $|DESC| = 10$

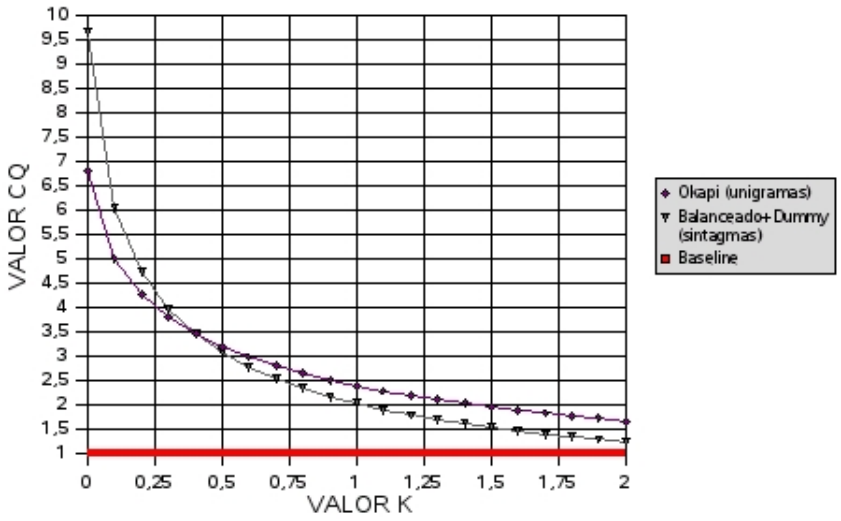


Figura 9.8: Resultados de la comparación entre una aproximación basada en unigramas y una aproximación basada en sintagmas terminológicos (considerando un nodo de propósito general). Precisión base del proceso de recuperación $0{,}17$ y número de descriptores seleccionados $|DESC| = 10$

minológicos (en la sección 8.4 hemos realizado su estimación experimental para nuestro escenario de clustering), la lectura de las gráficas no debe hacerse considerando un valor fijo de k para ambas aproximaciones y discutiendo su correspondiente valor CQ (tal y como hemos hecho en las evaluaciones anteriores). En una evaluación de este tipo, una lectura del eje de abscisas proporcionará el incremento de coste cognitivo necesario para obtener con ambas aproximaciones el mismo valor para la medida CQ. En este sentido, para valores de $k < 0{,}45$ podemos observar como la curva obtenida para los sintagmas se encuentra por encima de la curva correspondiente a los unigramas. De este modo, un mismo valor CQ implica que el coste cognitivo asociado a los unigramas sea menor que el coste cognitivo asociado a los sintagmas terminológicos. En contraste, para valores $k > 0{,}45$ las curvas se cruzan, lo que implica que iguales valores de la medida CQ supondrán un coste cognitivo para los sintagmas menor que el de los unigramas.

En nuestro escenario el valor de k para los sintagmas es de 0.4, mientras que para los unigramas es de 0.64. Esto significa que para obtener el mismo valor de CQ en ambos casos el valor de k para la aproximación basada en unigramas debería ser menor que el correspondiente a los sintagmas. Dado que en nuestro escenario esto no ocurre, podemos concluir que el valor de CQ sobre las estructuras de clustering basadas en unigramas siempre será menor y, por lo tanto, resultará mucho más adecuado realizar el proceso de clustering utilizando sintagmas terminológicos. Los valores de calidad correspondientes a cada una de las aproximaciones son de $3{,}45$ y $2{,}9$ utilizando sintagmas terminológicos y unigramas respectivamente, lo que supone una mejora del $18{,}9\,\%$.

9.2.4. Recapitulación

Los resultados obtenidos para la evaluación del segundo prototipo, estimando un valor de $k = 0{,}4$, se presentan a continuación:

- Para un número de descriptores fijo (10 descriptores), la evaluación de la calidad de las estructuras de clustering producidas utilizando las estrategias de selección balanceada y terminológica, aplicadas sobre sintagmas terminológicos, ha permitido seleccionar como más adecuada la primera de ellas. Esto se debe principalmente a que esta estrategia realiza una distribución mucho más uniforme de los documentos sobre los retículos generados. En contraste, la estrategia terminológica produce peores resultados de calidad sobre sintagmas terminológicos que sobre unigramas (expuestos en la evaluación del primer prototipo). Esto es debido a que la frecuencia de documento asociada a los sintagmas seleccionados es mucho menor y, como consecuencia, se generan estructuras de clustering muy pequeñas caracterizadas por tener un nodo raíz con una elevada población.

- Una evaluación de la población del cluster raíz nos ha permitido concluir que la estrategia de selección balanceada se comporta adecuadamente, obteniendo un conjunto de descriptores suficiente como para aislar de manera correcta la información relevante recuperada, siendo los documentos no descritos por alguno de los descriptores los que, al permanecer en el cluster raíz, disminuyen el valor de la medida DF.

- La comparación de los resultados de evaluación obtenidos para este prototipo con los obteni-

dos en el primer prototipo sobre unigramas nos ha permitido concluir que el uso de sintagmas es mucho más adecuado para construir sistemas de clustering sobre el escenario presentado. Para ello se han estudiado los valores CQ teniendo en cuenta el distinto coste cognitivo asociado a cada una de las aproximaciones y se ha llegado a la conclusión que el uso de sintagmas mejora en un $18{,}9\,\%$ la calidad del clustering con respecto al uso de unigramas.

9.3. Tercer Prototipo

El tercer prototipo presentado en esta Tesis Doctoral realiza una implementación del modelo propuesto utilizando n-gramas como descriptores del conjunto de clusters generado.

9.3.1. Objetivos

El diseño del prototipo presentado se ha orientado a cumplir los objetivos expuestos a continuación:

- Evaluar el grado de mejora en la tarea de recuperación que produce un sistema de clustering basado en n-gramas.
- Evaluar la influencia de considerar LSI en la construcción del contexto formal K. En este sentido, se pretende evaluar la mejora de la calidad de las estructuras de clustering obtenidas aplicando LSI al proceso de generación del contexto formal previo a la construcción del retículo.
- Evaluar la influencia de realizar el proceso de extracción de descriptores considerando únicamente fragmentos de los documentos recuperados. De este modo, se pretende evaluar la calidad de las estructuras de clustering obtenidas considerando únicamente los snippets de los documentos recuperados para el proceso de extracción de los descriptores.
- Evaluar la influencia del número de documentos considerados en el proceso de clustering. Con este fin se pretende evaluar la variación en la calidad de los retículos obtenidos cuando se incrementa el número de documentos sobre los cuales se realiza el proceso de clustering.

9.3.2. Características del prototipo

El prototipo ha sido implementado enteramente utilizando Java para programar el código del sistema y Swing para la realización de la interfaz de usuario. En este caso, y a diferencia de los prototipos anteriormente presentados, con el fin de mejorar la eficiencia y rapidez en el proceso de recuperación, se utilizó Apache Lucene como motor de búsqueda sobre el corpus de prueba. Por ello, los valores de precisión base obtenidos para los experimentos que presentaremos en las siguientes secciones son distintos a los obtenidos en los anteriores prototipos. Esto hace que los resultados que presentaremos aquí no puedan ser directamente comparables con las aproximaciones que hemos desarrollado en las secciones 9.1 y 9.2.

Las características funcionales del prototipo son las siguientes:

- *Proceso de recuperación de información.* El prototipo realiza este proceso sobre un motor de búsqueda externo (Apache Lucene) que ha sido integrado en el desarrollo del prototipo. Se caracteriza principalmente por estar desarrollado en Java y disponer de un API (Application Programing Interface) que permite realizar de manera muy sencilla cualquier operación relacionada con un motor de búsqueda. Lucene ha sido utilizado tanto para indexar y comprimir la colección de prueba como para llevar a cabo el proceso de recuperación a partir de las consultas realizadas por el usuario. Se han utilizado las herramientas de Lucene para la extracción de raíces y eliminación de palabras vacías para la realización de estos procesos. El procesamiento de las consultas considera únicamente la disyunción de sus términos para realizar el proceso de recuperación de información.

- *Proceso de extracción de descriptores.* El prototipo presentado utiliza n-gramas para describir los nodos generados en el proceso de clustering. Estos son extraídos mediante una aproximación basada en la construcción del conjunto de árboles de sufijos correspondiente al conjunto de documentos recuperado. Para ello puede utilizarse el texto completo de cada documento o un snippet representativo. El proceso de extracción de descriptores, éste puede ser realizado de acuerdo a cualquiera de estas dos posibilidades. Con el fin de conseguir esta funcionalidad, el proceso de extracción se lleva a cabo dinámicamente sobre el conjunto de documentos recuperado, es decir, no se realiza una extracción previa de la totalidad de los n-gramas sobre el corpus completo (que por otra parte consumiría una gran cantidad de recursos de memoria y de CPU), sino que el proceso de extracción se realiza sobre la información devuelta por el motor de búsqueda procesando, en cada caso, o bien el texto completo de los documentos recuperados o el snippet correspondiente obtenido de manera automática por Lucene.

- *Proceso de selección de descriptores.* Debido a que la estrategia balanceada presentada en el anterior prototipo fue la que obtuvo los mejores valores de calidad, en este prototipo hemos optado por implementar únicamente esta aproximación. Tal y como expusimos entonces, esta estrategia se caracteriza por realizar el proceso de selección sobre el conjunto de n-gramas extraídos minimizando el número de documentos que no serán representados por descriptor alguno.

- *Proceso de construcción del retículo.* En este prototipo hemos incorporado la posibilidad de aplicar Latent Semantic Indexing (LSI)sobre el proceso de construcción del contexto formal previo a la construcción de los retículos. En este sentido, este prototipo implementa las dos aproximaciones propuestas para la construcción del contexto formal K propuestas en la sección 6.5. De este modo, el prototipo permite generar estructuras de clustering a partir de un contexto formal obtenido directamente de las relaciones explícitas entre los descriptores seleccionados y los documentos recuperados o a partir de un contexto formal enriquecido mediante LSI. El cálculo de la descomposición en valores singulares se ha realizado en el prototipo utilizando el paquete Java JMP, que facilita las clases necesarias para procesar matrices dispersas y realizar las operaciones básicas LSI de manera eficiente

Interfaz del Tercer Prototipo

La figura 9.9 muestra la interfaz del tercer prototipo desarrollado en este trabajo. En él pueden distinguirse cuatro áreas:

- *El área de búsqueda*. Descrita en por la etiqueta *'Search Box'*. Su funcionalidad es la de recoger la consulta del usuario, permitiendo seleccionar la estrategia de selección de descriptores utilizada en el proceso de clustering, el número de documentos recuperados y el número de descriptores seleccionados.

- *El área de navegación principal sobre el clustering*. Es el área que aparece descrita en la interfaz por la etiqueta *'Explorer'*. Al igual que ocurría en el prototipo anterior, este prototipo implementa una aproximación para la visualización del clustering basada en retículos adaptada a los requisitos presentados en la sección 6.7. Sin embargo, con el fin de facilitar la comprensión de la estructura al usuario, en este caso hemos optado por eliminar los iconos asociados a los clusters, utilizando únicamente un código de colores que le permita determinar si un cluster concreto es o no terminal (los clusters representados en color amarillo indican la posibilidad de ser especializados, mientras que los representados en color rojo indican que se trata de un cluster terminal). Así mismo, en este prototipo hemos incluido etiquetas que muestran el número de documentos contenidos en cada uno de los clusters (el número de elementos contenidos en la extensión de su correspondiente nodo de información), así como el número de documentos total a los que el usuario puede acceder especializando dicho cluster (el número de elementos contenidos en la extensión de su correspondiente concepto formal).

- *El área de navegación sobre el conjunto de documentos*. Es el área que aparece en el área descrita por la etiqueta *'Main Search Results'*, su diseño y funcionalidad son exactamente los mismos que los descritos en el prototipo anterior. Recoge el conjunto de documentos contenidos en el cluster actualmente seleccionado, mostrando para cada uno de ellos su título y un breve fragmento del texto obtenido automáticamente.

- *El área de información del documento seleccionado*. Representa la información completa del documento actualmente seleccionado en el área de navegación sobre el conjunto de documentos. En este último prototipo el documento es mostrado al usuario mediante una ventana emergente cuando éste lo selecciona en el área de navegación sobre el conjunto de documentos.

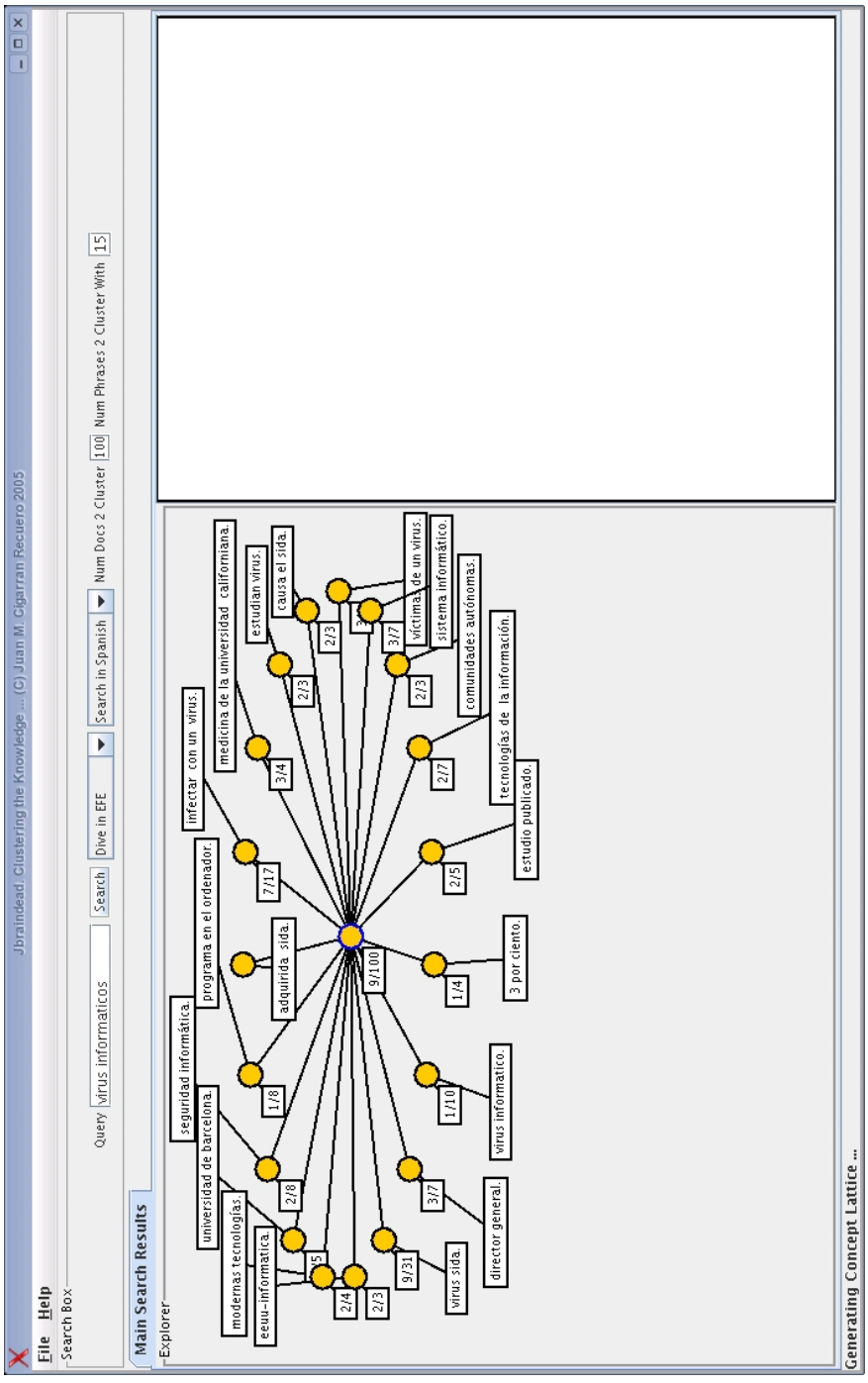


Figura 9.9: Interfaz del tercer prototipo. Muestra los resultados del clustering correspondiente a la consulta '*virus informáticos*'

9.3.3. Experimentos

Los experimentos propuestos en esta sección pretenden evaluar, sobre el prototipo presentado, la calidad de las estructuras de clustering generadas utilizando n-gramas para representar el conjunto de clusters obtenidos. Utilizando como marco de evaluación el conjunto de medidas propuesto en el capítulo 7 se realizaron los siguientes experimentos:

- *Evaluación de la influencia de considerar LSI en el proceso de construcción del contexto formal sobre la calidad del clustering*. Dado que el prototipo presentado implementa las dos aproximaciones propuestas en esta Tesis Doctoral para la construcción del contexto formal K, este primer estudio pretende evaluar el grado de mejora en las estructuras de clustering generadas a partir de contextos enriquecidos mediante la aplicación de LSI frente a aquellos contextos obtenidos de manera directa a partir de las relaciones explícitas entre el conjunto de descriptores seleccionado y el conjunto de documentos recuperado.

- *Evaluación de la influencia del número de documentos agrupados sobre la calidad del clustering*. En una primera aproximación podría pensarse que un aumento en el número de documentos recuperados, y por lo tanto agrupados, podría reducir la calidad de las estructuras de clustering generadas. El objetivo de este segundo experimento es el de determinar si esta apreciación es correcta o, por el contrario, un aumento en el número de documentos no influye sobre la capacidad para aislar la información relevante de las estructuras de clustering generadas.

- *Evaluación de la influencia de un proceso de extracción de n-gramas basado en snippets sobre la calidad del clustering*. Finalmente, la última evaluación presentada pretende comprobar el grado en el que varía la calidad de las estructuras de clustering generadas al considerar únicamente un pequeño fragmento de los documentos recuperados para realizar el proceso de extracción de descriptores.

Evaluación de la influencia del proceso de construcción del contexto

El objetivo del experimento presentado es el de cuantificar la influencia de considerar LSI para la construcción del contexto formal a partir del cual se obtendrán las estructuras de clustering finales. Sobre los 100 documentos más relevantes recuperados para el conjunto de consultas presentado en la sección 8.3, el experimento construye los retículos en base a un contexto formal generado a partir de las relaciones explícitas entre el conjunto de descriptores seleccionado y el conjunto de documentos recuperado y, también, a partir de un contexto formal enriquecido aplicando LSI. Debido a que en este prototipo el motor de búsqueda utilizado es distinto al implementado en los prototipos anteriores, la precisión media del proceso de recuperación varía, siendo ahora de 0.28. La estrategia de selección de descriptores utilizada será la estrategia balanceada sobre 10 descriptores.

El cuadro 9.7 muestra los resultados obtenidos en el experimento. Como puede observarse, en el caso de no considerar el coste cognitivo asociado a los descriptores, la estrategia de construcción

Construccion Contexto	**DF**	**Num. MBA**	**Num. MVA**	k_{max}
Directa	2.14	16.91	31.98	1.43
Aplicando LSI	2.3	18.31	34.93	1.41

Cuadro 9.7: Resultados del experimento para evaluar la influencia del proceso de construcción del contexto sobre la calidad de los retículos obtenidos. Estrategia de selección aplicada balanceada. Precisión base del proceso de recuperación $0{,}28$. Número de descriptores seleccionados $|DESC| = 10$

Construccion Contexto	**DF**	**Num. MBA**	**Num. MVA**	k_{max}
Directa	2.75	18.29	36.8	1.36
Aplicando LSI	3.01	20.09	40.84	1.32

Cuadro 9.8: Resultados del experimento para evaluar la influencia del proceso de construcción del contexto sobre la calidad de los retículos obtenidos. Estrategia de selección aplicada balanceada. Precisión base del proceso de recuperación $0{,}28$. Número de descriptores seleccionados $|DESC| = 15$

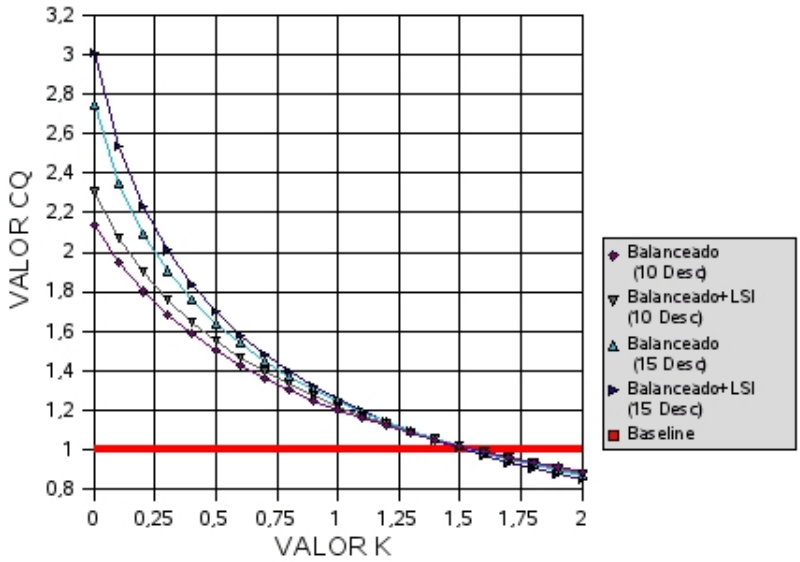


Figura 9.10: Resultados de la evaluación de la influencia de la construcción del contexto formal utilizando LSI sobre la calidad final del clustering. Aproximación de selección de descriptores balanceada. Precisión base del proceso de recuperación $0{,}28$. Número de descriptores seleccionados $|DESC| = 10$ y $|DESC| = 15$

del contexto formal basada en LSI obtiene unos resultados ligeramente superiores a los obtenidos por la estrategia de construcción directa, siendo la mejora del factor DF del $7{,}47\,\%$. Esto es debido a las frecuencias de documento de los descriptores seleccionados. El aplicar la estrategia balanceada para seleccionar un conjunto de descriptores pequeño (en nuestro caso 10 descriptores) implica que sus frecuencias de documento sean elevadas. De este modo, lose descriptores seleccionados muy probablemente aparezcan en la totalidad de los documentos con los cuales se encuentran semánticamente relacionados, lo que significa que el efecto de aplicar LSI sobre ellos no sea especialmente significativo.

Con el fin de comprobar esta apreciación, el cuadro 9.8 muestra los resultados del mismo experimento pero seleccionado 15 descriptores. Los resultados obtenidos muestran como, en este caso, el efecto de construir el contexto formal utilizando LSI ha producido una mejora de la calidad superior a la obtenida para 10 descriptores (en este caso la mejora en el factor DF es del 9.5 %). Nuevamente, la explicación de estos valores puede hacerse atendiendo a la frecuencia de documento asociada a los nuevos descriptores seleccionados. Estos se caracterizarán por tener una frecuencia de documento inferior a la de los 10 descriptores seleccionados en primer lugar, lo que implica una mayor probabilidad de que estén relacionados con documentos en los cuales no aparecen explícitamente y, por lo tanto, la influencia de LSI sobre ellos sea mucho más significativa.

La figura 9.10 muestra las curvas CQ correspondientes a las aproximaciones tomando 10 y 15 descriptores respectivamente. Dado que el aumento en el número de nodos que el usuario debe considerar sobre el clustering no es muy significativo al pasar de 10 a 15 descriptores, las gráficas correspondientes a cada aproximación no llegan a cruzarse para valores de $k < 1$. En todo el intervalo considerado las aproximaciones LSI obtuvieron valores de calidad superiores a los obtenidos al generar el contexto formal de manera directa, aunque el grado de mejora no puede considerarse significativo. Como consecuencia, independientemente del número de descriptores seleccionado, los resultados del experimento indican que una estrategia para la construcción del contexto formal basada en LSI sería adecuada, aunque no aporta mejoras sustanciales.

Evaluación de la influencia del número de documentos considerados

El objetivo de este segundo experimento presentado es el de evaluar la influencia del número de documentos considerados para la realización del clustering sobre su calidad. El experimento aplica un proceso de clustering sobre los 100 y 150 documentos más relevantes recuperados para el conjunto de consultas presentado en la sección 8.3. La estrategia de selección de descriptores aplicada será la estrategia balanceada sobre 10 descriptores y se utilizará LSI para la construcción de los contextos formales. La precisión base del proceso de recuperación será 0.28 y 0.23 en función del número de documentos recuperados, trabajando a cobertura máxima en ambos casos.

El cuadro 9.9 muestra los resultados obtenidos en el experimento. De acuerdo a las precisiones base obtenidas, puede observarse como el aumento en el número de documentos supone una disminución de la precisión general en el proceso de recuperación, lo que implica un mayor aumento en el número de documentos no relevantes recuperados que en el de documentos relevantes (como apunte destacar que en nuestros experimentos se pasó de procesar 71.8 documentos no relevantes en media a 114.91, lo que supone un incremento del $60\,\%$). Resulta lógico pensar que esta disminución de

Num. Docs Recuperados	**DF**	**Num. MBA**	**Num. MVA**	k_{max}
100 Documentos	2.3	18.31	34.93	1.41
150 Documentos	2.92	26.49	56.53	4.05

Cuadro 9.9: Resultados del experimento para evaluar la influencia del número de documentos procesados para la realización del clustering. Estrategia de selección aplicada balanceada. Construcción del contexto basada en LSI. Precisión base del proceso de recuperación $0{,}28$ y $0{,}23$. Número de descriptores seleccionados $|DESC| = 10$

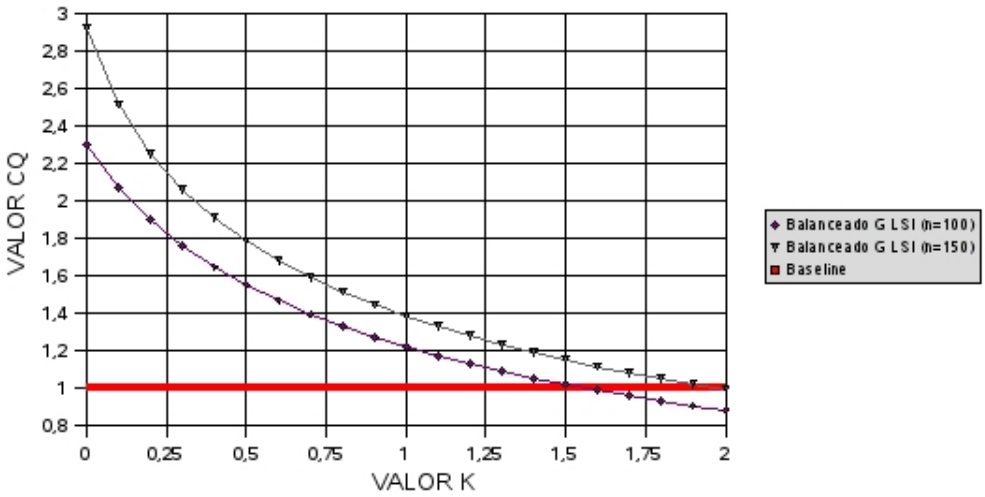


Figura 9.11: Evaluación del aumento del número de documentos sobre la calidad de las estructuras de clustering obtenidas. Estrategia de selección aplicada balanceada. Construcción del contexto basada en LSI. Precisión base del proceso de recuperación $0{,}28$ y $0{,}23$. Número de descriptores seleccionados $|DESC| = 10$. Número de documentos recuperados 100 y 150

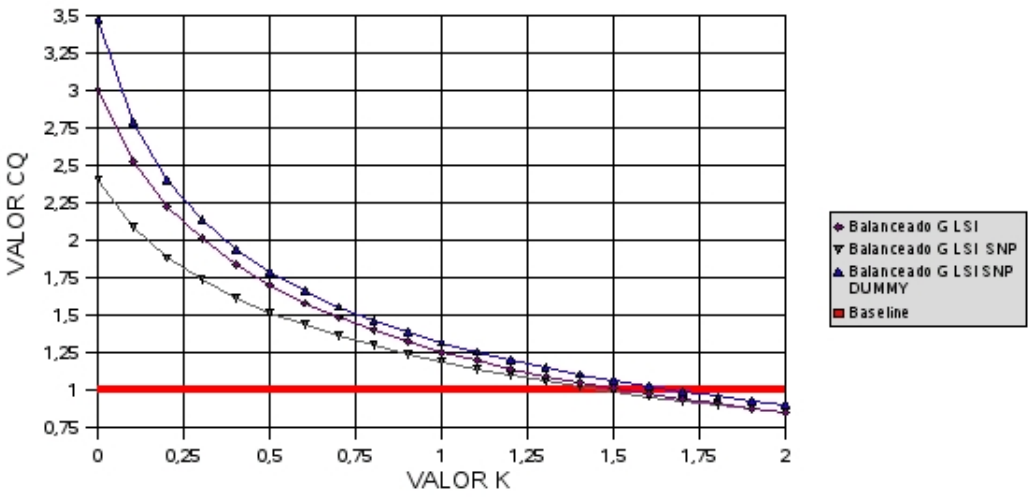


Figura 9.12: Evaluación de la influencia de los snippets en el proceso de obtención de las estructuras de clustering. Estrategia de selección aplicada balanceada. Construcción del contexto basada en LSI. Precisión base del proceso de recuperación 0.28. Número de descriptores seleccionados $|DESC| = 15$

la precisión en el proceso de recuperación debiera implicar una disminución en la calidad de las estructuras de clustering generadas, disminuyendo su capacidad para aislar correctamente la nueva información no relevante recuperada. Sin embargo, observando los resultados obtenidos podemos constatar que esto no ocurre así y que la calidad de los retículos obtenidos sobre 150 documentos es superior a la obtenida para únicamente 100 documentos (el valor de la medida DF se incrementa en un $26{,}9\,\%$). Esto es debido a que en la aproximación de clustering sobre 150 documentos la proporción entre el número de documentos contenido en el conjunto de documentos recuperados y el número de documentos contenidos en el MBA aumenta, lo que implica una mejor adecuación de los retículos obtenidos para aislar la información relevante que en el caso de considerar únicamente 100 documentos. En definitiva, el aumento de información recuperada al considerar un mayor número de documentos, no supone un aumento significativo en la cantidad de documentos contenidos en el MBA, lo que supone que, globalmente, la calidad del clustering aumente.

La figura 9.11 muestra las curvas CQ correspondientes a ambas aproximaciones. Resulta curioso observar como ambas transcurren prácticamente paralelas, lo que indica que, sobre este escenario, su variación de calidad se comporta de igual manera frente a las variaciones del parámetro k.

Evaluación de la influencia de considerar los snippets de los documentos

El objetivo del último experimento presentado es el de evaluar la influencia, sobre la calidad de las estructuras de clustering obtenidas, de realizar el proceso de extracción de los n-gramas sobre un fragmento de los documentos recuperados, en lugar de sobre su texto completo. Si la diferencia de

Procesamiento	**DF**	**Num. MBA**	**Num. MVA**	k_{max}
Documentos Completos	3.01	20.09	40.84	1.36
Snippets	2.41	17.84	36.31	1.41

Cuadro 9.10: Resultados del experimento para evaluar la influencia de considerar los snippets en el proceso de obtención de las estructuras de clustering. Estrategia de selección aplicada balanceada. Construcción del contexto basada en LSI. Precisión base del proceso de recuperación $0{,}28$. Número de descriptores seleccionados $|DESC| = 15$

Procesamiento	**DF**	**Num. MBA**	**Num. MVA**	k_{max}
Snippets + Cluster Dummy	3.48	18.2	36.91	1.52

Cuadro 9.11: Resultados del experimento para evaluar la influencia de considerar los snippets en el proceso de obtención de las estructuras de clustering. Estrategia de selección aplicada balanceada. Construcción del contexto basada en LSI. Precisión base del proceso de recuperación $0{,}28$. Número de descriptores seleccionados $|DESC| = 15$

calidad es pequeña, quedaría justificada la aplicación del clustering utilizando snippets de los resultados de búsqueda obtenidos de Internet (mucho más eficiente) en lugar de los propios documentos.

Con este fin se planteó el experimento aplicando el proceso de clustering sobre los 100 documentos más relevantes recuperados para el conjunto de consultas presentado en la sección 8.3 y aplicando el proceso de extracción de descriptores sobre los documentos completos y sobre un fragmento de los mismos obtenido de manera automática por el motor de búsqueda. La estrategia de selección utilizada fue la estrategia balanceada aplicada sobre 15 descriptores y el contexto formal fue construido aplicando LSI. La precisión base del proceso de recuperación, al igual que en los experimentos anteriores, fue de 0.28, realizando el proceso de evaluación a cobertura máxima.

El cuadro 9.10 muestra los resultados obtenidos en el experimento. Podemos observar como, en el caso de no considerar el coste cognitivo asociado a los descriptores, los mejores resultados para la calidad se obtienen aplicando una aproximación de extracción de descriptores que considere el texto completo de los documentos, siendo la mejora del factor DF de un 24.9 %. Estos resultados eran esperables y su explicación puede realizarse en base a que una aproximación para la extracción de n-gramas sobre fragmentos de texto procesará una menor cantidad de información y, como consecuencia, el número de n-gramas extraídos, así como su frecuencia de documento disminuirán.

Finalmente, y con el fin de demostrar la capacidad de las estructuras de clustering generadas a partir de fragmentos de texto, optamos por realizar el mismo experimento pero, en esta ocasión, separando los documentos contenidos en el cluster raíz sobre un nodo de propósito general, tal y como realizamos en el segundo experimento presentado en la sección 9.2.3. De este modo, sería sencillo comprobar la capacidad de los descriptores seleccionados para aislar la información relevante de manera adecuada. El cuadro 9.11 muestra los resultados obtenidos teniendo en cuenta esta nueva consideración. Como puede observarse, el valor para la medida DF ha aumentado, superando en este caso al valor obtenido a considerar el texto completo de los documentos. Este resultado indica que, incluso considerando una pequeña parte de los documentos, la información relevante es agrupada

de manera adecuada, mejorando en un factor $3{,}48$ la precisión obtenida sobre la lista inicialmente recuperada en el proceso.

La figura 9.12 muestra las curvas CQ correspondientes a las tres aproximaciones consideradas. A lo largo de todo el intervalo considerado para k las gráficas no llegan a cruzarse, lo que significa que el número de nodos que el usuario debe considerar en cada una de las aproximaciones no resulta ser especialmente significativo como para hacer que los valores CQ sean muy sensibles a las variaciones de k. De hecho, si observamos el número de nodos contenidos en el MVA, podemos concluir que la consideración de una u otra estrategia de selección no incrementa significativamente la cantidad de información que el usuario debe considerar para acceder a la información relevante, viéndose considerablemente reducido el tiempo de cálculo de las correspondientes estructuras de clustering.

Los resultados obtenidos en este último experimento son, sin duda, muy interesantes puesto que nos permiten concluir la adecuación de una aproximación basada en fragmentos de texto para la construcción de sistemas de clustering eficientes. En este sentido, una pequeña reducción en la precisión de los retículos obtenidos se ve plenamente justificada por una disminución considerable del tiempo asociado a su proceso de construcción (sobre todo en el proceso de extracción de los n-gramas mediante árboles de sufijos) que hacen viable la aplicación de nuestras propuestas de clustering sobre escenarios on-line que reciban dinámicamente la información recuperada.

9.3.4. Recapitulación

Los resultados obtenidos para la evaluación del tercer prototipo, estimando un valor de $k = 0{,}4$, se presentan a continuación:

- La aplicación de LSI en el proceso de construcción del contexto formal mejoró la calidad de los retículos obtenidos para las aproximaciones evaluadas en el experimento. Debemos destacar como el hecho de aumentar el número de descriptores seleccionados aumentó la influencia de LSI sobre la calidad final de las estructuras generadas. Esto fue debido a que, al aumentar el número de descriptores, se consideraron n-gramas con una menor frecuencia de documento y, por lo tanto, susceptibles de estar semánticamente relacionados con documentos en los que no aparecían explícitamente.

- La consideración de un mayor número de documentos para la realización del proceso de clustering produjo un aumento en la calidad de los retículos generados. Esto fue debido a que el aumento del número de documentos no produjo un aumento en la misma proporción del número de documentos no relevantes considerados en el MBA, lo que supuso un aumento global de la calidad de las estructuras.

- Finalmente, la consideración de los fragmentos de texto asociados a los documentos nos permitió determinar que, aunque se produce una ligera disminución en la calidad de los retículos (debida a que se procesa mucha menos información), el tiempo de proceso para obtener las correspondientes estructuras de clustering se ve reducido notablemente, por lo que resultan adecuadas para la construcción de este tipo de sistemas sobre escenarios dinámicos donde se

desconozca a priori el corpus completo y, además, sea necesario dar una respuesta rápida al usuario.

9.4. Sistema JBraindead

En esta sección presentamos el sistema JBraindead como un sistema plenamente funcional para la realización de clustering de documentos que toma como base para su diseño las conclusiones obtenidas sobre los prototipos presentados en las secciones anteriores. De este modo, JBraindead puede considerarse como el resultado final del proceso evolutivo y de refinamiento aplicado sobre los prototipos presentados.

Debido a que nuestro deseo era el de comprobar la viabilidad, tanto del modelo como de la arquitectura propuestos, sobre un escenario mayor y mucho menos controlado, el sistema presentado realizará el proceso de clustering sobre resultados de búsqueda obtenidos directamente de Internet. En concreto, JBraindead utiliza los resultados devueltos por motores de búsqueda comerciales (en concreto hemos utilizado los resultados devueltos por las APIs de Google y de Yahoo!) para llevar a cabo la construcción de retículos donde se organizará la información recuperada.

Dado que no entraba en nuestros objetivos el diseño de un motor de recuperación de información sobre Internet, decidimos que el sistema hiciera uso de las herramientas de recuperación de información que los motores de búsqueda comerciales ponen a disposición de sus usuarios para la integración de este tipo de funcionalidades en sus propios sistemas. El hecho de disponer de herramientas de este tipo nos ha permitido obviar el proceso de recuperación en el diseño de JBraindead y centrar nuestros esfuerzos en los aspectos más relacionados con el proceso de clustering que realiza el sistema. Actualmente, el sistema se encuentra disponible con acceso público en Internet[2]

El hecho de delegar el proceso de recuperación a un sistema externo supone perder el control del corpus sobre el cual se realiza el proceso de clustering, lo que implica plantear alguno de los procesos relacionados con la generación del clustering (en especial el proceso de extracción de descriptores) de manera dinámica sobre la información proporcionada para cada consulta. Así mismo, el hecho de no tener acceso al corpus completo y no disponer de consultas con juicios de relevancia asignados no nos ha permitido realizar una evaluación del sistema de acuerdo a la metodología utilizada en los anteriores prototipos.

9.4.1. Objetivos

Los objetivos principales del sistema JBraindead se exponen a continuación:

- Demostrar la viabilidad de nuestro modelo sobre un escenario no controlado, caracterizado por obtener de Internet la información sobre la que se realiza el proceso de clustering. Aunque el prototipo no ha sido evaluado cuantitativamente, pretendemos poner a disposición de los usuarios un sistema on-line real y eficiente que les permita realizar sus búsquedas sobre las

[2] http://bender.lsi.uned.es:8080/ModuloWeb/jbraindead.html

estructuras de clustering generadas utilizando AFC. De este modo, y aunque este trabajo queda pendiente, esperamos poder realizar una evaluación cualitativa del sistema a partir de las consultas y las interacciones realizadas por los usuarios sobre el sistema.

- Demostrar la viabilidad de las diferentes aproximaciones para los procesos involucrados en la construcción del clustering en tiempo real. En este sentido, el sistema implementa aquellas aproximaciones para la extracción y selección de descriptores, así como para la construcción del contexto formal, que mejores resultados obtuvieron sobre los prototipos que hemos presentado en las secciones anteriores.

Desafortunadamente, el hecho de no disponer de juicios de relevancia aplicables a una tarea de búsqueda sobre Internet no ha permitido evaluar este último sistema con el conjunto de medidas propuestas en este trabajo, siendo los comentarios realizados por los usuarios el único índice utilizado hasta el momento para realizar una evaluación subjetiva de la eficiencia tanto del sistema como de la usabilidad de su interfaz.

9.4.2. Características del sistema

El sistema JBraindead ha sido implementado utilizando Java para codificar todos los aspectos relacionados con el proceso completo de clustering. Debido a que el sistema ha sido diseñado para ser accesible desde Internet, el código Java se ha exportado como servicio web utilizando Apache Axis. La interfaz web se ha diseñado utilizando Macromedia Flash MX, utilizando Macromedia Remoting Components para realizar las peticiones al servicio web. El servidor sobre el cual se ha desplegado la aplicación ha sido Apache Tomcat.

Las características funcionales de JBraindead son las siguientes:

- *Proceso de recuperación de información.* Tal y como hemos expuesto, JBraindead delega el proceso de recuperación de información a motores de búsqueda comerciales. En concreto, el sistema hace uso de los resultados de búsqueda proporcionados por Google [59] y Yahoo! [139]. Estos sistemas, además de ser accesibles vía web, proporcionan acceso a sus motores mediante servicios web. Para ello, proporcionan a sus usuarios APIs (Application Programming Interfaces) en diferentes lenguajes de programación (principalmente en Java y C) que facilitan la integración de las operaciones de búsqueda sobre sistemas externos. Las características principales de este tipo de herramientas son:

  1. La posibilidad de ajustar cualquier parámetro de la búsqueda de igual manera a como se ajustan al manipular su interfaz web. En este sentido, ambas APIs permiten ajustar de manera sencilla el número de documentos recuperados, el idioma preferible sobre el que se realizará la búsqueda, la eliminación de páginas web similares, etc.
  2. Los resultados de la búsqueda se devuelven encapsulados en un objeto definido en el API cuya manipulación es muy sencilla. Ambos motores de búsqueda definen este objeto con un conjunto de propiedades que permiten acceder al título de la página, su snippet

o su dirección web de manera directa. Es importante destacar que en los resultados no se incluye el texto completo de cada página (debido a que ralentizaría demasiado el proceso de respuesta del sistema de búsqueda), por lo que JBraindead deberá operar con el snippet y el título de la página para realizar el proceso de clustering.

Debido a que vamos a utilizar dos fuentes de información distintas, con el objeto de eliminar las páginas repetidas en ambas búsquedas JBraindead realiza un sencillo procedimiento de identificación de páginas iguales comparando sus títulos y urls.

Finalmente, debemos destacar que el proceso de recuperación puede realizarse indistintamente en inglés o en castellano.

- *Proceso de extracción de descriptores.* El sistema presentado representa el conjunto de clusters obtenido mediante n-gramas. Dado que no es posible disponer del corpus de documentos, se hace necesario realizar el proceso de extracción de manera dinámica sobre el conjunto de documentos recuperados. En este sentido, el sistema tomará como fuente para la extracción de los n-gramas el conjunto de resultados obtenido en el proceso de recuperación, calculando el árbol de sufijos correspondiente al título y al snippet de cada página web. Tal y como expusimos al describir el proceso de obtención de estas estructuras, el sistema deberá normalizar el conjunto de resultados aplicando un proceso de extracción de raíces y eliminación de palabras vacías. Dado que el sistema debe ser capaz de procesar información en inglés y en castellano, JBraindead utilizará el conjunto de herramientas de Apache Lucene para realizar este preprocesamiento [3].

- *Proceso de selección de descriptores.* La estrategia de selección de descriptores implementada en el sistema ha sido la balanceada. Esto es debido a dos razones principales:

  - La estrategia no requiere el conocimiento explícito del corpus completo. En este sentido, el hecho de no tener acceso a la colección no permite aplicar otras estrategias de selección de descriptores como la estrategia terminológica o la estrategia okapi.
  - La estrategia balanceada obtuvo unos buenos resultados de calidad para los experimentos realizados en los prototipos presentados. De este modo, y aunque el escenario de recuperación es diferente, asumimos un comportamiento similar del sistema sobre documentos recuperados de Internet.

[3] Apache Lucene dispone de un completo conjunto de librerías para realizar extracción de raíces y procesamiento de palabras vacías sobre multitud de idiomas entre los que se encuentran el inglés, francés, italiano, alemán o castellano.

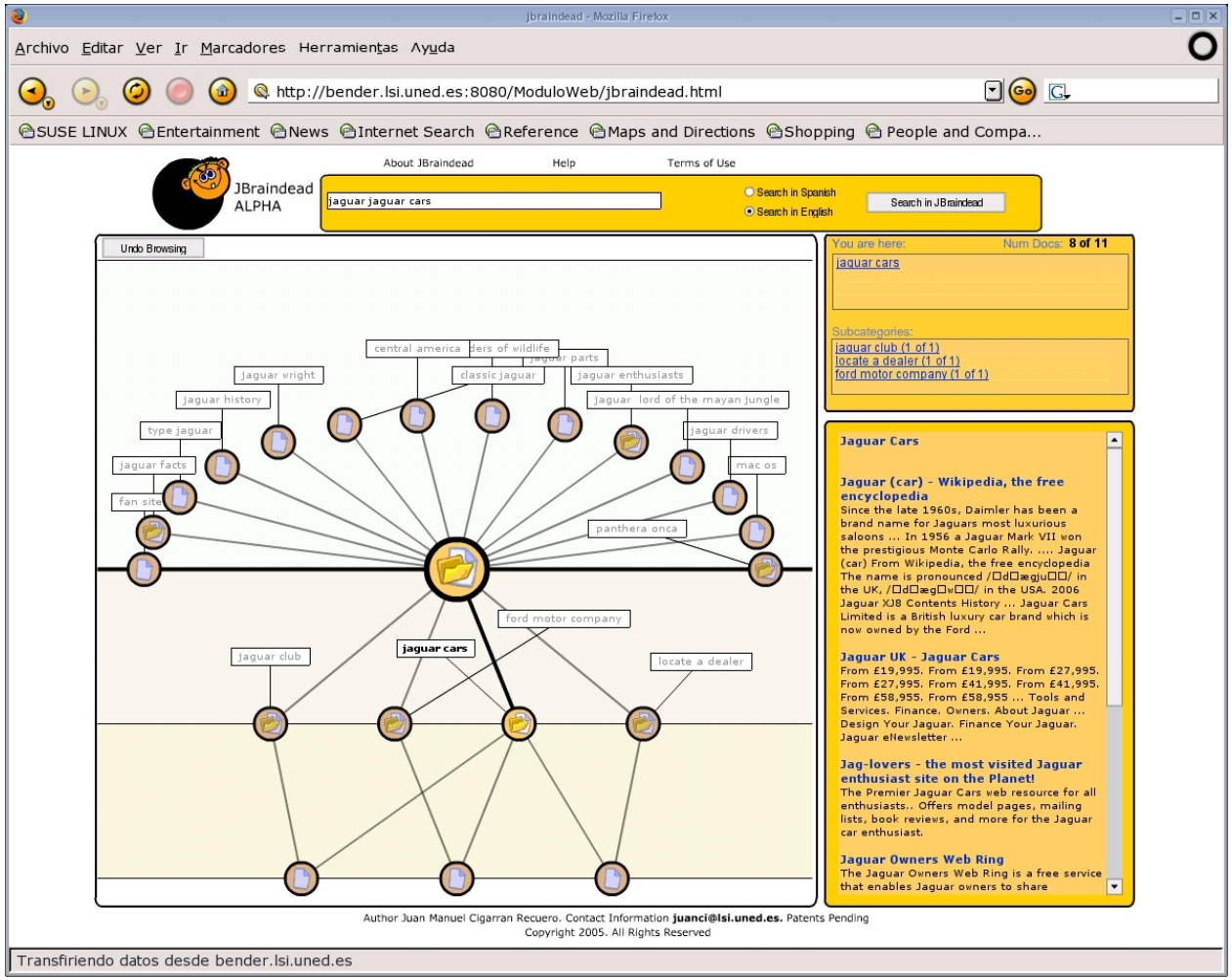


Figura 9.13: Interfaz del sistema JBraindead. Muestra el resultado del clustering de páginas web correspondiente a la consulta 'jaguar'. El cluster actualmente seleccionado está descrito por *'jaguar cars'*

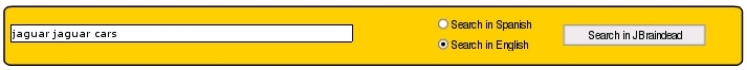


Figura 9.14: Area de búsqueda y refinamiento de la consulta del sistema JBraindead

- *Proceso de construcción del retículo.* El sistema JBraindead implementa una estrategia para la construcción del contexto formal basada en LSI. Dado que en los experimentos del tercer prototipo (sección 9.3.3) hemos demostrado que la construcción de los retículos a partir de un contexto enriquecido con LSI mejora la calidad de las estructuras de clustering generadas, asumimos un comportamiento similar del sistema sobre documentos recuperados de Internet.

9.4.3. Interfaz de JBraindead

La figura 9.13 muestra la interfaz del sistema JBraindead. En ella pueden distinguirse las siguientes partes bien diferenciadas:

- *El área de búsqueda y refinamiento de la consulta*. Se muestra en detalle en la figura 9.14, siendo su objetivo principal el de facilitar la interacción del usuario con el sistema para la realización de nuevas búsquedas. En este sentido, la realización de una búsqueda en JBraindead puede realizarse de dos maneras diferentes:

  - Introduciendo los términos de la nueva consulta directamente en la caja de texto y pulsando el botón *'Search in JBraindead'*. Este escenario puede identificarse con la necesidad del usuario de realizar una nueva búsqueda independiente de las realizadas hasta el momento (es la situación más común al interactuar con un motor de búsqueda).
  - Aprovechando los descriptores asociados al cluster actualmente seleccionado. En este sentido, JBraindead incluye una funcionalidad de búsqueda que permite realimentar los descriptores asociados al cluster explorado en cada momento, con el fin de realizar una búsqueda mucho más específica a la realizada inicialmente por el usuario. De este modo, las interacciones del usuario con el área de navegación sobre el clustering van reflejando en la caja de texto de la consulta la intensión asociada al cluster actualmente inspeccionado, permitiendo al usuario refinar su búsqueda con sólo pulsar el botón *'Search in JBraindead'*. Este escenario se identifica con la necesidad del usuario de encontrar más información relacionada con el cluster que se encuentra explorando.

  Dado que, tal y como ya hemos expuesto, el proceso de recuperación inicial se lleva a cabo utilizando los motores de búsqueda comerciales Google y Yahoo!, la consulta puede expresarse utilizando todas las opciones de búsqueda avanzadas que proporcionan estos buscadores.

  Así mismo, el área de búsqueda permite al usuario elegir el idioma en el que desea realizar su búsqueda, pudiendo elegir entre el inglés y el castellano. Nótese como la selección del idioma esta relacionada tanto con el modo en el que serán preprocesados los documentos y

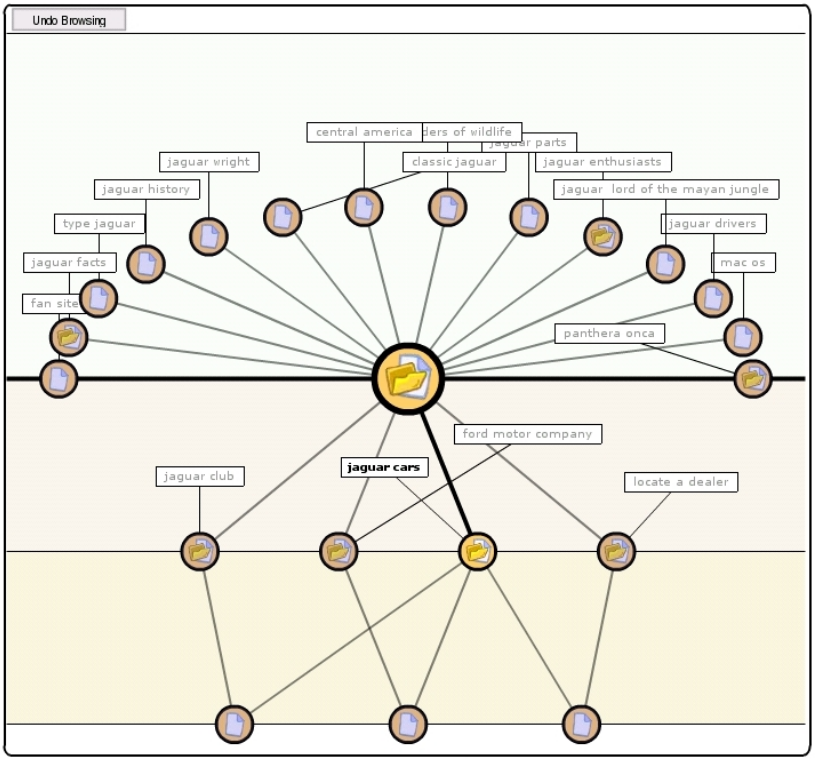


Figura 9.15: Área de navegación principal sobre el clustering del sistema JBraindead

las consultas como con el modo en el que se invoca a los motores de búsqueda, indicándoles el idioma de las páginas web que deben ser recuperadas.

- *El área de navegación principal sobre el clustering*. Su funcionalidad es la de representar el retículo obtenido a partir del conjunto de documentos recuperados para las consultas realizadas por el usuario. Las pautas seguidas para realizar esta representación se basan en el paradigma de navegación basado en retículos presentado en la sección 6.7, mostrando al usuario únicamente aquellos clusters contenidos en el up-set del conjunto de vecinos inferiores del nodo actualmente seleccionado y utilizando diagramas de Hasse para representarlo.

  La figura 9.15 muestra este área. Nótese como el cluster actualmente seleccionado, así como el conjunto de clusters de los cuales hereda su intensión (su up-set) aparecen destacados en la interfaz con un trazo más grueso.

  Para la representación de los clusters se ha utilizado un conjunto de iconos cuyo objetivo es el de informar al usuario acerca del contenido de los mismos, así como de su posibilidad de especialización. La figura 9.16 muestra su representación, siendo su significado concreto el siguiente:

  - El primero de ellos representa un nodo de información que contiene documentos (es

(a) (b) (c)

Figura 9.16: Iconos asociados a los clusters del área de navegación principal sobre el clustering

concepto objeto) pero que, sin embargo, es un nodo terminal y, como consecuencia, no es posible realizar su especialización.

- El segundo de ellos representa un nodo de información que no contiene documento alguno (no es concepto objeto) que puede ser especializado.
- El tercer icono representa un nodo de información que, además de contener documentos, puede ser especializado en clusters mucho más concretos.

El área de navegación sobre el clustering dispone de un botón *'Undo Browsing'*, similar al botón *'Atrás'* o *'Back'* de los navegadores web, que permite deshacer la navegación realizada, facilitando el acceso a estados de exploración anteriores. La utilidad de esta funcionalidad es especialmente importante sobre nuestro sistema. Esto es debido a que, a lo largo de un proceso de exploración, el usuario únicamente visualiza aquellas partes del retículo focalizadas sobre aquella zona del retículo en la que se encuentra en cada momento, por lo que disponer de un mecanismo que le permita volver a estados anteriores de la exploración sin necesidad de interactuar con el retículo resulta imprescindible.

- *El área de navegación secundaria sobre directorios web*. Tal y como expusimos en la sección 6.7, el hecho de utilizar retículos para mostrar los resultados del clustering, supone hacer que el usuario se enfrente a una tipo de estructura con la cual no está familiarizado. En este sentido, y con el fin de facilitar la interacción con el sistema en las primeras etapas de uso, JBraindead proporciona un modo de navegación alternativa que permite acceder a los nodos del clustering de acuerdo a un paradigma de navegación basado en directorios web. La figura 9.17 muestra su interfaz, donde pueden observarse dos partes bien diferenciadas:

  - El área descrita por la etiqueta *'You are here: '* muestra el conjunto de descriptores que representa el cluster actualmente seleccionado. Este conjunto se muestra al usuario mediante una lista de descriptores sobre la cual éste podrá generalizar su exploración. La lista se construye en base a los conceptos atributo de cada uno de los descriptores del cluster actualmente seleccionado facilitando, de este modo, la agrupación de varios descriptores en cada uno de los elementos de la lista (en caso de que así fuera necesario).
  - El área descrita por la etiqueta *'Subcategories: '* muestra el conjunto de descriptores seleccionables que permiten especializar el cluster actualmente seleccionado. La obtención de esta lista se lleva a cabo a partir de los clusters vecinos inferiores del cluster actualmente seleccionado, extrayendo los descriptores característicos de cada uno de estos clusters.

Figura 9.17: Área de navegación secundaria sobre directorios web del sistema JBraindead

Figura 9.18: Área de navegación sobre documentos del sistema JBraindead

Así mismo, en este área se proporciona al usuario información acerca del número de documentos contenidos en el cluster actualmente seleccionado, así como en número de documentos total al que podrá acceder realizando la especialización de dicho cluster.

- *El área de navegación sobre documentos*. Muestra el conjunto de documentos contenido en el cluster actualmente seleccionado descritos por su título y el snippet devueltos por el motor de búsqueda (consultar la figura 9.18). Sobre este área el usuario podrá seleccionar cualquiera de los documentos, pulsando sobre su título, haciendo que el sistema le muestre el contenido completo del documento en una nueva ventana del navegador.

9.4.4. Recapitulación

En esta última sección se ha presentado en sistema Jbraindead como una evolución natural del conjunto de prototipos propuesto en esta Tesis Doctoral. Las características principales del sistema son:

1. La aplicación de las técnicas de clustering basadas en AFC sobre resultados de búsqueda obtenidos dinámicamente de motores de búsqueda comerciales tales como Google o Yahoo!

2. La implementación de las técnicas de extracción y selección de descriptores que proporcionaron los mejores resultados para las medidas DF y CQ en los experimentos realizados sobre los distintos prototipos. En concreto, JBraindead implementa árboles de sufijos para llevar a cabo la extracción de n-gramas sobre los snippets obtenidos, la técnica de selección de descriptores balanceada y Latent Semantic Indexing para enriquecer las relaciones descriptor-documento obtenidas.

3. La implementación de una interfaz intuitiva que combina una navegación basada en retículos con una navegación más clásica basada en una estructura similar a la de los directorios web. Ambos modos de navegación pueden considerarse una aportación novedosa de este trabajo ya que no se limitan a proporcionar al usuario la estructura completa sobre la cual llevar a cabo el proceso de navegación, sino que focalizan su atención en aquellas zonas de la estructura relacionadas con el cluster que se encuentra explorando en cada momento, mostrando únicamente el subconjunto de clusters o de descriptores más próximos que le permiten refinar o generalizar su búsqueda.

Como trabajo pendiente relacionado con este sistema queda su evaluación utilizando las medidas propuestas en este trabajo. El hecho de no disponer de un entorno de evaluación similar al utilizado en la evaluación de los prototipos presentados, pero orientado a la web, no nos ha permitido concluir cuantitativamente acerca de la mejora que este sistema produce para una tarea de recuperación de información en comparación con la lista de documentos originalmente devueltos por los motores de búsqueda.

Parte IV

CONCLUSIONES FINALES

Capítulo 10

Conclusiones

A lo largo de esta Tesis Doctoral hemos presentado una aproximación para la organización de los resultados de búsqueda mediante Análisis Formal de Conceptos. En contraste con las estructuras de clustering jerárquico tradicionales, el AFC permite obtener estructuras de clustering mucho más informativas, facilitando al usuario la interpretación y acceso al espacio de información recuperado. En concreto, hemos focalizado los esfuerzos de este trabajo en resolver tres aspectos fundamentales:

- Formalizar un modelo matemático basado en AFC capaz de modelar un clustering de documentos.
- Presentar una metodología orientada a la aplicación del modelo sobre sistemas de recuperación y visualización de información.
- Desarrollar un conjunto de medidas adecuadas para evaluar un sistema de este tipo respecto a una tarea de recuperación de información.

Además, como parte final de esta Tesis Doctoral hemos presentado una evaluación que considera cada una de nuestras aproximaciones sobre el modelo presentado, lo que nos ha permitido definir una metodología para evaluar este tipo de sistemas, así como para demostrar el correcto funcionamiento de las medidas presentadas.

Finalmente, y como demostración de que una implementación de nuestras propuestas sobre un sistema real, plenamente funcional y público es viable hemos presentado el sistema JBraindead. Este sistema se encuentra actualmente a disposición de cualquier usuario en Internet[1], permitiéndole realizar un clustering basado en AFC a partir de un conjunto de documentos recuperados de la web. Aunque este sistema no ha sido evaluado formalmente, actualmente dispone de *logs de sesión* activos que nos permitirán, en un futuro, realizar una evaluación cualitativa de su calidad.

El capítulo está estructurado del siguiente modo. Inicialmente, presentamos por separado las conclusiones extraídas para cada uno de los aspectos desarrollados en este trabajo (modelo propuesto,

[1] http://bender.lsi.uned.es:8080/ModuloWeb/jbraindead.html

aproximaciones para la construcción de un sistema de clustering basado en nuestro modelo y medidas de evaluación). A continuación, repasaremos brevemente los prototipos desarrollados en esta Tesis Doctoral, así como los resultados experimentales extraídos de cada uno de ellos. Finalmente presentaremos la lista de publicaciones y congresos donde se han publicado parcialmente alguno de los resultados obtenidos en este trabajo y las líneas de investigación futuras.

10.1. Propuesta de un modelo de clustering basado en AFC

En el capítulo 5 hemos presentado un modelo específicamente desarrollado a partir de la teoría del AFC orientado a construir un clustering de documentos. Los objetivos perseguidos al desarrollar el modelo se dirigieron a respetar el conjunto de consideraciones fundamentales para un clustering de documentos presentado en la sección 5.1, así como el conjunto de restricciones propuestas en dicho capítulo. En concreto, el modelo respeta la *restricción del clustering a un 'universo abierto'*, con el objeto de presentar en cada uno de los clusters únicamente aquellos documentos descritos de manera completa por el conjunto de descriptores del mismo, así como la *restricción del clustering a considerar herencia múltiple*, que considera que un documento puede ser accedido desde zonas disjuntas del retículo, siendo considerado como componente de un único cluster.

El modelo presenta una alternativa a la representación de un clustering de documentos basado en AFC, transformando dicha representación en base a la definición de *nodo de información* y de las funciones de transformación presentadas en la sección 5.3. Como consecuencia, el clustering obtenido en base a nuestra propuesta presenta una serie de ventajas, no sólo desde el punto de vista de la estructura generada, sino también desde el punto de vista de su visualización y de su evaluación automática.

Consideramos que definir un modelo basado en AFC para la realización de un clustering de documentos es una aportación novedosa. Esto es debido a que en el área del AFC, hasta la fecha, las únicas aportaciones para la realización de tareas de recuperación de información se han llevado a cabo mediante aproximaciones ad hoc considerando directamente los retículos obtenidos y sin plantear un modelo formal capaz de definir el modo en el que tanto los clusters como sus documentos asociados son considerados.

10.2. Propuesta de una metodología para la aplicación del modelo sobre un sistema de recuperación y visualización de información

Una vez definido el modelo, en el capítulo 6 hemos presentado una metodología para el desarrollo de sistemas de clustering basados en nuestra propuesta. Para ello, hemos ido describiendo y profundizando en cada uno de los procesos asociados a este tipo de sistemas, presentando especial atención a las estrategias de extracción y de selección de descriptores, así como a las estrategias de construcción del contexto formal y de visualización de la estructura de clustering finalmente obtenida.

En relación con las estrategias de extracción, hemos presentado diferentes alternativas basadas en la

tipología de los descriptores obtenidos. En concreto, hemos considerado la extracción de unigramas y n-gramas de longitud no definida, presentando la extracción de sintagmas terminológicos como un caso particular de este último. En los tres casos, los sistemas desarrollados han obtenido buenos resultados en la tarea de recuperación de información, destacando especialmente el uso de n-gramas y sintagmas terminológicos debido a su mayor capacidad informativa con respecto a los unigramas.

En relación con las estrategias de selección de descriptores, hemos barajado diferentes alternativas obtenidas de diferentes áreas tales como la recuperación de información (utilizando un pesado $tf - idf$) o de la extracción terminológica (utilizando un pesado terminológico orientado a la construcción de diccionarios específicos). Debemos destacar que respecto a este particular, y con el objeto de mejorar la población de las estructuras de clustering obtenidas, en este trabajo hemos propuesto una estrategia de *selección balanceada*. Esta estrategia se caracteriza por dirigir el proceso de selección hacia la obtención de aquel conjunto de descriptores que represente el mayor número de documentos del conjunto inicialmente recuperado. En este sentido, consideramos esta estrategia como una aportación novedosa y efectiva, cuyas mejoras sobre una tarea de recuperación de información en un sistema de clustering han sido demostradas sobre los prototipos implementados.

En relación con las estrategias de construcción del contexto formal debemos destacar que la aplicación de LSI para mejorar los resultados sobre estructuras generadas utilizando AFC nunca ha sido probada, por lo que consideramos los experimentos realizados en este sentido, así como la propuesta para enriquecer las relaciones existentes en un contexto formal una aportación novedosa de nuestro trabajo. Los resultados obtenidos en los experimentos presentados indican que, aunque la mejora producida no resulta especialmente significativa, la aplicación de esta técnica puede mejorar la efectividad de un sistema de clustering basado en AFC.

Finalmente, en relación con las estrategias de visualización propuestas debemos destacar el hecho de que éstas explotan la propia estructura de retículo sobre la cual se organiza la información recuperada. En este sentido, nuestra aportación resulta novedosa frente a otras aportaciones basadas en clustering jerárquico. Hasta la fecha, únicamente el sistema Credo está basado en retículos como estructuras subyacentes para la organización de un conjunto de documentos pero, en contraste con nuestras propuestas, no utiliza aproximaciones para su visualización que se basen en dicha estructura para facilitar la navegación e interacción del usuario. Nuestra propuesta de *visualización basada en retículos* se caracteriza, además, por simplificar la navegación a través de dichas estructuras, mostrando al usuario únicamente aquellas partes del espacio de información organizado que se encuentran directamente relacionadas con el cluster que se está explorando en cada momento. Por otra parte, consideramos que la estrategia de *visualización basada en directorios web* es una aportación novedosa y única que, aprovechando un paradigma de navegación sobradamente conocido por los usuarios, permite mapear la estructura de un retículo de manera sencilla. Por desgracia, en este trabajo no se han podido realizar evaluaciones con usuarios reales que nos permitan concluir su adecuación a una tarea de recuperación de información, aunque la implementación del sistema JBraindead y su publicación en Internet con acceso público nos ha permitido obtener una idea inicial, a partir de entrevistas con algunos usuarios, de su funcionalidad e idoneidad para este tipo de tareas.

10.3. Propuesta de un conjunto de medidas de evaluación

Finalmente, en este trabajo hemos desarrollado y presentado un conjunto de medidas orientadas a evaluar la calidad, en una tarea de recuperación de información, de un sistema de clustering basado en retículos. Tal y como expusimos en el capítulo 7, hasta la fecha no existe en el área del AFC una propuesta concreta que permita evaluar la calidad de los retículos en tareas de este tipo, siendo únicamente las evaluaciones con usuarios reales las alternativas presentadas en este tipo de trabajos. Consideramos nuestra aportación como novedosa, ya que las medidas propuestas facilitan la definición de marcos experimentales para el diseño de prototipos basados en AFC que permitan evaluar rápidamente la adecuación de diferentes estrategias utilizadas para su construcción, así como comparar los resultados obtenidos a partir de diferentes prototipos. Además, nuestras medidas se caracterizan por ser generalizables a cualquier tipo de escenario de recuperación, siendo únicamente necesario estimar el valor del parámetro k para cada uno de ellos.

Las medidas presentadas en este trabajo están basadas en el concepto de *área de navegación mínima* (MBA), que representa el conjunto mínimo de nodos y enlaces que el usuario debería recorrer para acceder a toda la información relevante recuperada. Dado que este conjunto resulta ser óptimo sobre cada uno de los retículos evaluados, el valor proporcionado por las medidas propuestas debe entenderse como una cota superior a la mejora para la tarea de recuperación que experimenta el sistema de clustering con respecto a la lista inicialmente devuelta por el motor de búsqueda.

Las medidas presentadas han sido el *factor de destilación* y la *calidad del clustering*. La primera de ellas únicamente tiene en cuenta el coste cognitivo asociado a la exploración de los documento y, por lo tanto, no refleja el esfuerzo de navegación e interacción que realiza el usuario hasta acceder a toda la información relevante. No obstante, resulta una medida adecuada para reflejar la capacidad del clustering para aislar correctamente la información relevante y puede entenderse como la mejora de la precisión del retículo obtenido con respecto a la lista inicial de documentos. La segunda de ellas, está basada en el mismo planteamiento matemático pero incluye entre sus factores el coste cognitivo asociado a la exploración de los descriptores de cada cluster. En este segundo caso, el hecho de considerar el coste cognitivo asociado a los clusters permite reflejar de manera mucho más precisa la influencia del proceso de interacción del usuario sobre el clustering, haciendo que estructuras correctamente organizadas pero con una gran cantidad de clusters obtengan peores valores de calidad que estructuras de similares características pero con un menor número de nodos.

Debemos remarcar que, hasta la fecha, no conocemos ninguna medida de evaluación que tenga en cuenta los aspectos considerados en nuestra propuesta para la evaluación de una tarea de recuperación sobre un clustering de documentos. De igual modo, para el área del AFC constituye la primera aportación formal orientada a la evaluación automática de retículos de conceptos.

10.4. Prototipos y experimentos desarrollados

Finalmente, la última parte de esta Tesis Doctoral ha estado orientada a probar la viabilidad de todas nuestras propuestas sobre distintos prototipos y sistemas. Para ello, no sólo nos hemos limitado a realizar su desarrollo sino que hemos aplicado el conjunto de medidas propuestas con el fin de

evaluar cada una de las aproximaciones.

En concreto, en este trabajo se han desarrollado tres prototipos sobre una colección estática de noticias bien conocida y un sistema final orientado a la organización de resultados de búsqueda obtenidos de motores de búsqueda comerciales. Las conclusiones extraídas en cada uno de ellos se presentan a continuación:

- *Primer prototipo*. Esta basado en la construcción de un clustering sobre unigramas. En este caso aplicamos la estrategia de selección sobre pesado OKAPI y la estrategia de de selección sobre pesado terminológico. En este primer prototipo no se experimentó con el uso de LSI para la construcción del contexto formal. Los experimentos desarrollados se orientaron a evaluar la influencia del proceso de selección, así como la influencia del número de descriptores seleccionados sobre la calidad final del clustering. Los principales resultados fueron los siguientes:

  - En ambos experimentos los resultados obtenidos indicaron la adecuación de todas las aproximaciones respecto a la lista de documentos original.
  - Respecto al estudio comparativo entre ambas aproximaciones no pudimos concluir acerca de cual de las dos resultaba ser más adecuada debido a que los valores de calidad obtenidos resultaron ser muy similares para el escenario de recuperación sobre el que se desarrolló el experimento.
  - El estudio acerca de la influencia del número de descriptores nos permitió concluir que un aumento en el número de descriptores seleccionados produce un aumento de precisión en del sistema. No obstante, pudimos observar como este aumento de precisión se hacía menos significativo conforme el número de descriptores era mayor, lo que nos indujo a pensar en la existencia de un cierto umbral para el número de descriptores a partir del cual la mejora de precisión no justificaría el coste asociado al aumento en el número de descriptores.
  - Finalmente, pudimos comprobar como el comportamiento de las dos estrategias de selección con respecto al número de descriptores seleccionados era diferente, permitiéndonos concluir que a valores de calidad similares optar por la estrategia aplicada sobre un mayor número de descriptores resultaba ser más adecuado. Esta conclusión estuvo basada en la mejora de la capacidad informativa asociada a los retículos con un mayor número de descriptores.

- *Segundo prototipo*. Está basado en la construcción de un clustering sobre sintagmas terminológicos. En este caso aplicamos una estrategia de selección sobre pesado terminológico y el algoritmo balanceado propuesto en este trabajo. En este segundo prototipo tampoco experimentamos con el uso de LSI para la construcción del contexto formal. Los experimentos desarrollados se orientaron a evaluar la influencia del proceso de selección de descriptores, la población del cluster raíz y a realizar una comparativa entre el uso de unigramas y sintagmas terminológicos como descriptores para las estructuras de clustering obtenidas. Los principales resultados fueron los siguientes:

- En los tres experimentos los resultados obtenidos indicaron la adecuación de todas las aproximaciones respecto a la lista de documentos original.
- Respecto a la evaluación de la influencia del proceso de selección, los experimentos nos permitieron concluir como el uso de una estrategia balanceada permitía obtener clusters con una calidad muy superior. Esto fue debido principalmente a que los sintagmas seleccionados por la estrategia terminológica presentaban una frecuencia de documento muy pequeña, lo que implicaba estructuras de clustering extremadamente pequeñas y poco discriminativas con un gran número de documentos en su nodo raíz.
- Respecto a la evaluación de la población del cluster raíz, aplicamos una estrategia balanceada añadiendo todos los documentos no descritos por el conjunto de descriptores seleccionados a un cluster de propósito general (que denominamos cluster *dummy*). Los resultados nos indicaron, sin ningún género de dudas, que esta consideración producía estructuras de clustering mucho más precisas. Esto significaba que el proceso de selección de descriptores permitía generar estructuras de clustering capaces de separar de forma adecuada la información relevante de la información no relevante, siendo los documentos que finalmente quedaban sin describir los que hacían disminuir el resultado final de la evaluación obtenido en el experimento anterior.
- Finalmente, la comparación de los resultados obtenidos para el prototipo anterior (sobre unigramas) y los obtenidos en este prototipo (sobre sintagmas terminológicos) nos indicaron que una aproximación basada en sintagmas terminológicos obtenía mejores resultados de calidad para el escenario de recuperación sobre el que se realizaron los experimentos. Esto fue debido principalmente al distinto valor (obtenido experimentalmente) para el factor k considerando ambos tipos de descriptores.

- *Tercer prototipo*. Está basado en la construcción de un clustering sobre n-gramas de longitud variable. En este caso aplicamos una estrategia de selección basada en el algoritmo balanceado y experimentamos con la construcción del contexto formal utilizando LSI. Los experimentos desarrollados se orientaron a evaluar la influencia de LSI en el proceso de construcción del contexto formal, evaluar la influencia del número de documentos agrupados por el sistema de clustering, evaluación de considerar los snippets en lugar de los documentos completos en el proceso de extracción de los n-gramas. Los principales resultados fueron los siguientes:

  - En los tres experimentos los resultados obtenidos indicaron la adecuación de todas las aproximaciones respecto a la lista de documentos original
  - Respecto a la aplicación de LSI para la creación del contexto, obtuvimos resultados que indicaban que su uso mejoraba la calidad del clustering. Además, pudimos comprobar que el hecho de aumentar el número de descriptores aumentaba la ganancia de precisión entre una construcción del retículo directa y su correspondiente enriquecida mediante el uso de LSI.
  - La consideración de un mayor número de documentos produjo un aumento en la calidad de los retículos generados. Esto fue debido principalmente a que el aumento en el

número de documentos recuperado no produjo un aumento proporcional del número de documentos no relevantes considerados en el MBA, lo que supuso un aumento global de la calidad de las estructuras generadas.

- Finalmente, el uso de snippets produjo una disminución no muy importante en la calidad de las estructuras generadas dado que la cantidad de información sobre la que trabajaba basado en snippets era bastante menor a la utilizada por el sistema basado en texto completo. No obstante, la realización de un experimento considerando un cluster de propósito general nos permitió demostrar como el sistema, aun trabajando con snippets, producía una mejora de calidad bastante significativa que llegaba a superar el valor obtenido por la aproximación sobre texto completo sin considerar el cluster de propósito general.

Basados en las conclusiones extraídas de los diferentes prototipos desarrollados, finalmente presentamos el sistema JBraindead cuya funcionalidad principal es la de organizar resultados web en base a todas las propuestas presentadas en esta Tesis Doctoral. Se trata de un sistema on-line que genera las estructuras de clustering en tiempo real y que utiliza una aproximación basada en n-gramas extraídos a partir de los snippets proporcionados por los motores de búsqueda y seleccionados mediante el algoritmo balanceado propuesto en este trabajo. La construcción del contexto la lleva a cabo aplicando LSI y presenta un sistema de visualización novedoso basado en retículos y en directorios web. Actualmente se encuentra con acceso público y, aunque no ha sido evaluado cuantitativamente, esperamos realizar en un futuro una evaluación cualitativa del mismo en función de los logs de usuario recogidos.

10.5. Publicaciones del autor

El trabajo presentado en esta Tesis Doctoral ha sido parcialmente publicado en diferentes congresos y foros a lo largo de los años 2004 y 2005. En concreto, las publicaciones realizadas han sido las siguientes:

- *Browsing Search Results Via Formal Concept Analysis: Automatic Selection of Attributes* [20]. Publicado en el 2004 en la *International Conference on Formal Concept Analysis ICFCA04*, único foro internacional donde se exponen y discuten las aportaciones científicas en el área del AFC. Este primer artículo planteó principalmente dos aspectos. El primero de ellos se orientó a presentar diferentes estrategias para la selección automática de descriptores, considerando únicamente unigramas, mientras que el segundo se enfocó a plantear la necesidad de disponer de medidas de evaluación adecuadas y a proponerlas. En este primer artículo se presentó una primera versión de las medidas propuestas en esta Tesis Doctoral que, a raíz de experimentos posteriores, evolucionaron hacia las versiones finalmente presentadas.

- *Automatic Selection of Noun Phrases as Document Descriptors in an FCA-based Information Retrieval System* [21]. Publicado en el 2005 en el mismo foro que el artículo anterior. En este caso, se amplió el trabajo presentado en el año anterior, proponiendo la utilización de

sintagmas terminológicos como unidades básicas para la construcción de los descriptores necesarios en el proceso de clustering.

- *Evaluating Hierarchical Clustering of Search Results* [22]. Publicado en el 2005 en la *String Processing and Information Retrieval Conference SPIRE05*. En esta ocasión, presentamos las nuevas medidas de evaluación obtenidas como una modificación y evolución de las propuestas en [20].

10.6. Lineas Futuras de Trabajo

Con el fin de continuar las líneas de trabajo e investigación abiertas en esta Tesis Doctoral, en este apartado esbozamos aquellos trabajos que han quedado pendientes o que están proyectados para su futura realización:

- *Realización de una evaluación con usuarios reales.* Las medidas de evaluación propuestas en esta Tesis Doctoral permiten evaluar la calidad de las estructuras de clustering generadas de manera automática. Esto supone que, dado que trabajan considerando una parte del retículo óptima para acceder a la información relevante, el valor obtenido representa una cota superior al valor real de calidad que un usuario podría obtener navegando por el sistema. Obviamente, los resultados de la interacción de un usuario con la estructura de clustering no sólo dependen de lo correctamente agrupados que se encuentren los documentos relevantes, sino que también están relacionados con la capacidad informativa de los descriptores utilizados para describir cada uno de los clusters. En este sentido, la realización de una evaluación con usuarios de las estructuras de clustering generadas a partir de nuestra propuesta nos permitiría comprobar la proximidad de estos resultados con los obtenidos automáticamente, permitiéndonos extraer conclusiones acerca de la calidad de los descriptores utilizados, así como del grado de confianza de las medidas propuestas.

- *Propuesta de nuevas estrategias de selección de descriptores.* Tal y como hemos expuesto, en la metodología para la construcción de un sistema de clustering basado en AFC, el proceso de selección de descriptores resulta crítico para la obtención de estructuras suficientemente informativas y navegables para el usuario. En este sentido, el desarrollo de nuevas técnicas que nos permitan abordar el problema de la selección de descriptores puede mejorar la calidad de las estructuras finalmente obtenidas.

- *Propuesta y evaluación de nuevas estrategias de visualización.* El uso de retículos como estructuras subyacentes para la organización de documentos abre nuevas posibilidades de cara a su correcta visualización e interacción. En este sentido, consideramos que no sólo es necesario proponer nuevas alternativas para su visualización, sino también diseñar estrategias de evaluación que permitan determinar su calidad de cara a una tarea de recuperación de información.

- *Comparativa con otras aproximaciones de clustering de documentos.* Finalmente, a lo largo de todo este trabajo únicamente hemos considerado estructuras de clustering basadas en AFC,

realizando los experimentos de evaluación sobre distintas aproximaciones orientadas a su generación. En este sentido, en esta Tesis Doctoral no hemos considerado la posibilidad de realizar una comparativa con otro tipo de aproximaciones de clustering jerárquico propuestas en la literatura del área. De este modo, y como trabajo futuro, proponemos la realización de experimentos de este tipo que permitan comparar propuestas basadas en distintos tipos de aproximaciones a la realización del clustering. Esto supone, no sólo comparar distintas propuestas, sino también demostrar la adecuación de las medidas propuestas en este trabajo para evaluar correctamente estructuras de clustering que no estén directamente basadas en retículos.

Parte V

APENDICES

Apéndice A

Técnicas Adicionales Utilizadas en el Trabajo

En este anexo presentamos un conjunto de técnicas que, por ser necesarias para la realización de alguno de los procesos presentados en nuestra propuesta, consideramos deben ser brevemente explicadas. En concreto, nos centraremos en la definición y desarrollo de los fundamentos matemáticos de *Latent Semantic Indexing* (LSI), utilizado en esta Tesis Doctoral para enriquecer el contexto formal que dará lugar a las estructuras de clustering obtenidas a partir de un conjunto de documentos, y en la definición de árbol de sufijos, aplicado en esta Tesis Doctoral para la extracción automática de los n-gramas utilizados como descriptores del contexto formal.

A.1. Latent Semantic Indexing

LSI es una técnica matemática que permite extraer e inferir relaciones del uso contextual de las palabras dentro de los pasajes del discurso. Como características principales debemos destacar que no se trata de una técnica de procesamiento del lenguaje natural o de inteligencia artificial, que no hace uso de conocimiento externo al texto en si mismo como, por ejemplo, diccionarios, bases de conocimiento, redes semánticas, gramáticas, analizadores sintácticos o morfológicos y que trabaja únicamente con el texto completo considerando sus términos como cadenas de caracteres. En las siguientes secciones presentaremos los conceptos básicos relacionados con esta técnica, así como sus fundamentos matemáticos.

A.1.1. Introducción

Anderson [5] se percató de la analogía existente entre el proceso de recuperación de información y los procesos humanos relacionados con la memoria semántica. De este modo, un proceso de recuperación de información podría entenderse como un proceso en el cual el usuario expresa la percepción y comprensión de aquello que busca mediante una secuencia de palabras que denomi-

namos consulta. El éxito en una tarea de búsqueda de este tipo dependerá de la representación que haga el sistema de recuperación de la consulta realizada y de los documentos sobre los cuales se lleva a cabo el proceso. Lo más habitual en este tipo de sistemas es buscar correspondencias exactas entre los términos utilizados en la consulta y los términos que componen los documentos del corpus, no siendo posible recuperar documentos que estando semánticamente relacionados con la consulta no contengan alguno de sus términos. La aplicación de LSI al proceso de recuperación de información, sin embargo, permite resolver parcialmente este problema. Esto es debido a que LSI realiza una representación de los documentos del corpus no sólo por los términos que éstos contienen sino también por aquellos términos que, sin aparecer directamente en éstos, tienen relación semántica con el dominio de conocimiento sobre el cual éstos versan. Este proceso es realizado de manera automática utilizando únicamente la información contenida en el corpus de documentos y sin hacer uso de bases de conocimiento externas que permitan enriquecer de alguna manera los documentos. En su primera aplicación real a una tarea de recuperación de información, LSI produjo mejoras significativas con respecto al resto de los métodos de indexación, representación y recuperación [41]. Actualmente se apunta como una técnica factible capaz de mejorar notablemente la eficiencia en los sistemas de RI sobre los cuales se implementa.

La base sobre la cual se apoya la aplicación de LSI requiere representar el corpus de entrada como una matriz M donde cada fila representa un término o un n-grama perteneciente al corpus de documentos y cada columna representa cada uno de sus documentos. Cada elemento $m_{ij} \in M$ contiene la frecuencia con la que el término o n-grama i aparece en el documento j. Una vez construída la matriz M, los pasos para aplicar LSI podrían esquematizarse del siguiente modo:

- Se realiza una transformación de la matriz M, de acuerdo al proceso que describiremos a continuación, de modo que cada frecuencia sea pesada conforme a una función que exprese la importancia del término dentro del propio documento y el grado en el que dicho término aporta información relevante al corpus de documentos completo.

- Una vez realizado el pesado de los elementos de la matriz M, se lleva a cabo una descomposición en valores singulares (SVD) de la matriz obtenida. Este proceso permite descomponer la matriz rectangular M en el producto de tres nuevas matrices cuyo significado es el siguiente: a) una de las matrices describe las entidades de la fila original de M como vectores de los factores ortogonales derivados, b) otra de las matrices describe las entidades originales de las columnas como factores ortogonales derivados y, finalmente; c) la última de las matrices es una matriz diagonal que contiene los valores de escala (autovalores) de modo que cuando las tres matrices son multiplicadas se reconstruye la matriz original. Cualquier matriz puede descomponerse de este modo sin necesidad de utilizar factor adicional alguno.

- Conocida la descomposición en valores singulares y, como consecuencia, la matriz diagonal de autovalores, LSI propone la generación de una nueva matriz M', del mismo tamaño que la matriz M original, reduciendo la dimensión del espacio sobre el cual se encuentra definida M. La nueva matriz M' será una proyección de M sobre un espacio de dimensión inferior que produce el efecto de *añadir* a los documentos términos semánticamente relacionados que no aparecían en los textos originales sobre los que se construyó la matriz M. Debido a que la

reducción de la dimensión aplicada a M depende del número de autovalores que se supriman de la matriz diagonal obtenida, queda abierto a pruebas experimentales el determinar este valor para cada corpus en particular no siendo posible determinar con exactitud la dimensión óptima de manera general.

A continuación se presentan con un poco más de detalle los fundamentos matemáticos de LSI.

A.1.2. Fundamentos matemáticos de LSI

Entendiendo por un corpus una colección de m documentos, podemos definir cada documento como una colección de términos de un universo de n términos. De este modo, cada documento puede ser representado por un vector en $\Re^n$ donde cada eje representa un término. La coordenada i-esima de dicho vector modelará, mediante alguna función específica (modelo booleano, frecuencia, $tf - idf$, etc.), la importancia del término i-esimo en el documento representado por el vector.

Sea A una matriz de $n \times m$ de rango r cuyas filas representan términos y cuyas columnas representan documentos. Sean $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r$ los valores singulares de A (los autovalores de AA^T). La descomposición en valores singulares (SVD) de A expresa dicha matriz como el producto de tres matrices:

$$A = UDV^T \tag{A.1}$$

Donde $D = diag(\sigma_1, \ldots, \sigma_r)$ es una matriz de $r \times r$, $U = (u_1, \ldots, u_r)$ es una matriz $n \times r$ cuyas columnas son ortonormales y $V = (v_1, \ldots, v_r)$ es una matriz $m \times r$ cuyas columnas también son ortonormales.

Obsérvese como dados v_i y u_i (columnas i-esimas de V y U respectivamente) se cumple $Av_i = \sigma_i u_i$ y $A^T u_i = \sigma_i v_i$. Por esta razón denominamos al vector (u_i, v_i) un par vector singular. Los pares vector singulares están directamente relacionados con los autovectores de modo que u_i y v_i son los autovectores correspondientes a los i-esimos autovalores más grandes de las matrices AA^T y $A^T A$ respectivamente.

La aplicación de LSI propone considerar únicamente los k valores singulares más grandes de la descomposición anterior, siendo su objetivo el de reducir la dimensión del espacio de trabajo. Tal y como ya hemos expuesto, determinar el número de autovalores a considerar no es inmediato y supone experimentar con el corpus sobre el cual se aplica LSI con el fin de obtener los resultados óptimos que reflejen correctamente las características intrínsecas y la semántica del corpus de trabajo. Formalmente, siendo $D_k = diag(\sigma_1, \ldots, \sigma_k)$, $U_k = (u_1, \ldots, u_k)$ y $V_k = (v_1, \ldots, v_k)$, entonces A_k se define como:

$$A_k = U_k D_k V_k^T \tag{A.2}$$

Donde A_k es una matriz de rango k que LSI interpreta como una aproximación de A donde las filas de $V_k D_k$ representan los documentos del corpus o, en otras palabras, donde los vectores columna de

A (los documentos) son proyectados en el espacio k-dimensional definido por los vectores columna de U_k. Al nuevo espacio definido es habitual denominarlo *espacio LSI de* A.

La capacidad del espacio LSI creado para extraer la semántica del conjunto de documentos y eliminar ruido no necesario dependerá del valor de k seleccionado. No obstante, y aunque hemos expuesto que el valor de k puede determinarse experimentalmente sobre el corpus de documentos sobre el cual se aplica LSI, existen aproximaciones matemáticas que permiten obtener una aproximación para el valor de k. El siguiente teorema, originalmente propuesto por Eckart y Young, proporciona una de estas aproximaciones:

Teorema 1 *De entre todas las matrices C de $n \times m$ con rango de, al menos k, A_k es aquella que minimiza el error $\|A - C\|^2 = \sum_{i,j}(A_{i,j} - C_{i,j})^2$*

Donde la norma aplicada puede ser tanto la norma espectral como la norma de Frobenius definidas del siguiente modo:

$$\|A\|_2 = \sigma_1 \tag{A.3}$$

$$\|A\|_F = \sqrt{\sigma_1^2 + \ldots + \sigma_r^2} \tag{A.4}$$

Aplicando la definición de las normas al teorema obtenemos:

$$\|A - A_k\|_2 = \sigma_{k+1} \tag{A.5}$$

$$\|A - A_k\|_F = \sqrt{\sigma_{k+1}^2 + \ldots + \sigma_r^2} \tag{A.6}$$

Lo que lleva a la conclusión que $\|A - C\|^2$ representa la importancia de incorporar el par vector singular i-esimo en la aproximación generada por LSI.

A.2. Árboles de Sufijos

La idea de los árboles de sufijos surge para dotar a los sistemas de recuperación de información de mecanismos capaces de responder consultas basadas en secuencias de palabras o, incluso, en secuencias concretas de caracteres (tal y como puede ocurrir en sistemas de recuperación de información sobre bases de datos genéticas, donde el concepto de término o palabra no existe).

Un árbol de sufijos es una estructura de datos trie[1], construída a partir de todos los sufijos de un texto, y donde los punteros a los sufijos se encuentran almacenados en los nodos hoja del mismo. Habitualmente, y con el fin de mejorar el espacio de memoria requerido para su almacenamiento,

[1]Se utiliza este término para hacer referencia a estructuras válidas para llevar a cabo recuperación de información sobre ellas. Su nombre proviene de la palabra retrieval.

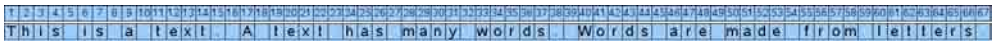


Figura A.1: Secuencia de caracteres de ejemplo con sus índices correspondientes.

Posicion	Sufijos
11	text. A text has many words. Words are made from letters.
19	text has many words. Words are made from letters.
28	many words. Words are made from letters.
33	words. Words are made from letters.
40	Words are made from letters.
50	made from letters.
60	letters.

Cuadro A.1: Lista de sufijos permitidos sobre la secuencia de caracteres presentada en la figura A.1 previamente normalizada.

los árboles de sufijos se suelen compactar mediante estructura denominada *árbol Patricia*. Estas estructuras se caracterizan porque los caminos unarios (aquellos caminos del árbol donde cada nodo tiene únicamente un hijo) se encuentran compactados en un único nodo. Esto implica indicar en cada uno de los nodos raíz de estos caminos cual es la siguiente posición a considerar dentro de la cadena de caracteres procesada en el árbol. Las mejoras en el coste de almacenamiento para este tipo de estructuras, mucho más compactas, permiten obtener un coste de $\Theta(n)$ nodos, en comparación con el coste $\Theta(n^2)$ obtenido en el caso peor para un árbol trie.

La figura A.1 muestra un texto[2] de ejemplo cuyos sufijos permitidos se muestran en el cuadro A.1. Nótese como en el ejemplo, la secuencia de caracteres ha sido previamente normalizada eliminando de la misma todas las palabras vacías que pudieran aparecer al comienzo del sufijo. La figura A.2 representa el árbol trie correspondiente a la secuencia de caracteres presentada, mientras que la figura A.3 muestra su correspondiente árbol de sufijos compactado.

[2]El ejemplo ha sido tomado de 'Modern Information Retrieval' (Baeza-Yates R.)[9]

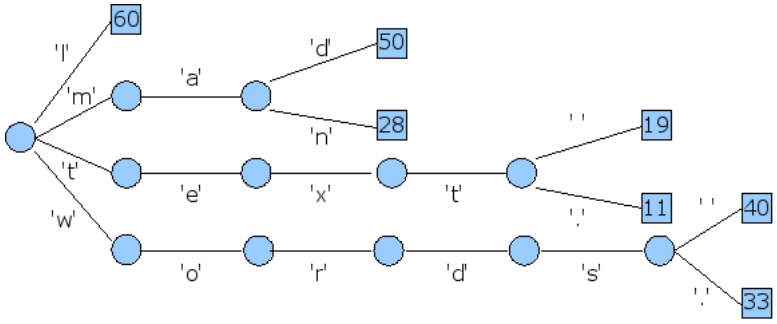


Figura A.2: Árbol trie correspondiente a la secuencia de caracteres de la figura A.1

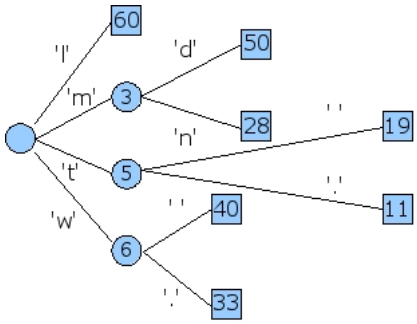


Figura A.3: Árbol de sufijos correspondiente a la secuencia de caracteres de la figura A.1

Figura A.4: Array de sufijos correspondiente a la secuencia de caracteres de la figura A.1 tras ordenar alfabéticamente la lista de sufijos

Aun así, el ahorro de espacio en memoria por parte de los árboles Patricia no es tan significativo como pudiera parecer inicialmente. Dependiendo de la implementación, cada nodo requerirá entre 12 y 24 bytes [9] necesitando, incluso en el caso de indexar únicamente los comienzos de las palabras, entre un 120 % y un 240 % más de memoria en comparación con el texto orginal sobre el cual se aplica el árbol de sufijos.

Con el fin de solventar este problema se presentan los *arrays de sufijos* como estructuras de datos alternativas para llevar a cabo una representación de los árboles de sufijos de manera mucho más eficiente. Los arrays de sufijos proporcionan la misma funcionalidad que los árboles de sufijos pero tienen unos requerimientos de memoria mucho menos ambiciosos. Estos basan su estructura en la definición de un array que contiene todos los punteros a los sufijos del texto ordenados alfabéticamente. Debido a que la estructura almacena un puntero por cada sufijo indexado, los requerimientos de espacio en memoria son mucho menores (un 40 % más de espacio en memoria que el texto original). La figura A.4 muestra el array de sufijos correspondiente al texto de ejemplo presentado en

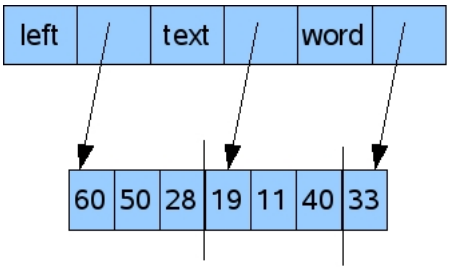


Figura A.5: Array de sufijos con supra-índice correspondiente a la secuencia de caracteres de la figura A.1.

la figura A.1.

Los arrays de sufijos pueden ser utilizados para la realización de búsquedas binarias mediante comparaciones con los contenidos apuntados por cada uno de los punteros almacenados en el array. Debido a que estas comparaciones habitualmente suponen accesos a memoria secundaria, con el consiguiente consumo de tiempo, la realización de este tipo de búsquedas sobre arrays de sufijos muy extensos puede producir serios problemas de rendimiento. Una posible solución a este problema pasa por el uso de *supra-índices*. Su finalidad principal es la de almacenar en memoria un subconjunto de las entradas almacenadas en el array, siendo sus requerimientos de memoria reducidos. De este modo, el supra-índice puede ser utilizado en una primera aproximación para guiar el proceso de búsqueda hacia un bloque específico del array de sufijos, lo que reduce el número de accesos a memoria secundaria. Un supra-índice que cubra el vocabulario completo sobre el cual se obtiene el array de sufijos requerirá un espacio equivalente al del almacenamiento de los índices invertidos obtenidos sobre dicho vocabulario [9].

La figura A.5 muestra cómo aplicar un supra-índice al array de sufijos correspondiente a la secuencia de caracteres de la figura A.1. Como puede observarse en el ejemplo, un supra-índice no necesita reproducir las entradas en intervalos fijos sobre el texto original.

Bibliografía

[1] G. Adamson and J. Bush. A comparision of the performance of some similarity and dissimilarity measures in the automatic classification of chemical structures. *Journal of Chemical Information and Computer Sciences*, 15:55–58, 1975.

[2] M. Agosti, M. Melucci, and F. Crestani. Automatic authoring and construction of hypertexts for information retrieval. *ACM Multimedia Systems*, 3:15–24, 1995.

[3] J. Allan. Building hypertext using information retrieval. *Information Processing and Management*, 33:145–159, 1997.

[4] E. Amigó, J. Gonzalo, V. Peinado, A. Peñas, and F. Verdejo. An empirical study of information synthesis task. In *Proceedings of the 42th Annual Meeting of the Association for Computational Linguistics*, Julio 2004.

[5] J. Anderson. *The Adaptive Character of Thought*. Hillsdale, NJ, 1990.

[6] K. Andrews, M. Pichler, and J. Wolte. Information pyramids: A new approach to visualizing large hierarchies. In *Proceedings of Late Breaking Hot Topics, IEEE Visualization 97*, pages 49–52, Phoenix, AZ, USA, 1997.

[7] J. Araya. Interactive query reformulation and feedback experiments in information retrieval. Master's thesis, Cornell University, Ithaca, NY, 1990.

[8] Ask. http://www.ask.com.

[9] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press, 1999.

[10] P. Becker. Toscanaj: Conceptual scaling of many-valued contexts. In *Proceedings of the 3rd International Conference of Formal Concept Analysis (ICFCA05)*, LNAI. Springer, 2005.

[11] D. Boley, M. Gini, R. Gross, S. Han, K. Hastings, G. K. pis, V. Kumar, B. Mobasher, and J. Moore. Partitioning-based clustering for web document categorization. *Decision Support Systems*, (3):329–341, 1999.

[12] G. Brajnik, S. Mizzaro, and C. Tasso. Evaluating user interfaces to information retrieval systems. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 128–136, Zurich, Switzerland, 1996.

[13] C. Buckley and A. Lewit. Optimizations of inverted vector searches. In *Proceedings of the annual international ACM SIGIR conference on Research and development in information retrieval*, 1985.

[14] C. Carpineto and G. Romano. *Topics in Artificial Intelligence*, chapter Automatic construction of navigable concept networks characterizing text databases, pages 67–78. Springer, 1995.

[15] C. Carpineto and G. Romano. A lattice conceptual clustering system and its aplications to browsing retrieval. *Machine Learning*, 24(2):95–122, 1996.

[16] C. Carpineto and G. Romano. Effective reformulation of boolean queries with concept lattices. In *Proceedings of the 3rd International Conference on Flexible Query-Answering Systems*, pages 83–94, Roskilde, Denmark, 1998.

[17] C. Carpineto and G. Romano. Order-theoretical ranking. *Journal of the American Society for Information Science*, 51(7):587–601, 2000.

[18] C. Carpineto and G. Romano. *Concept Data Analysis: Theory and Applications*. John Wiley and Sons, 2004.

[19] C. Carpineto and G. Romano. Exploiting the potential of concept lattices for information retrieval with credo. *Journal of Universal Computer Science*, 10(8):985–1013, 2004.

[20] J. Cigarran, J. Gonzalo, A. Peñas, and F. Verdejo. Browsing search results via formal concept analysis: Automatic selection of attributes. In P. Eklund, editor, *Concept Lattices*, number 2961 in LNAI, pages 74–87, Sydney, Australia, February 2004. Second International Conference on Formal Concept Analysis, ICFCA 2004, Springer. ISBN 3-540-21043 ISSN 0302-9743.

[21] J. Cigarran, A. Peñas, J. Gonzalo, and F. Verdejo. Automatic selection of noun phrases as document descriptors in an fca-based information retrieval system. In B. Ganter and R. Godin, editors, *Formal Concept Analysis*, number 3403 in LNAI, pages 49–63, Lens, France, February 2005. Third International Conference, ICFCA 2005, Springer. ISBN 3-540-24525-1 ISSN 0302-9743.

[22] J. Cigarran, A. Peñas, J. Gonzalo, and F. Verdejo. Evaluating hierarchical clustering of search results. In M. Consens and G.Ñavarro, editors, *String Processing and Information Retrieval. 12th International Conference, SPIRE 2005.*, volume 3772 of *Lecture Notes In Computer Science (LNCS)*, pages 49–54, Buenos Aires. Argentina, November 2005.

[23] C. Cleverdon. Optimizing convenient on-line access to bibliographic databases. *Information Service and Use*, 4(1):37–47, 1974.

[24] R. Cole. The management and visualization of document collections using formal concept analysis. Master's thesis, Griffith University, 2000.

[25] R. Cole and P. Eklund. Application of formal concept analysis to information retrieval using a hierarchical structured thesauris. In *Supplementary proceedings of International Conference on Conceptual Structures ICCS 96*, pages 1–12, University of South Wales, 1996.

[26] R. Cole and P. Eklund. Scalability in formal concept analysis. *Computational Intelligence*, 15(1):11–27, 1999.

[27] R. Cole and P. Eklund. A knowledge representation for information filtering using formal concept analysis. *Linkoping Electronic Articles in Computer ans Information Science*, 5(5), 2000.

[28] R. Cole and P. Eklund. Browsing semi-structured web texts using formal concept analysis. In *Proceedings of the 9th International Conference on Conceptual Structurs*, volume 2120 of *LNAI*, pages 319–332, Stanford, 2001. Springer.

[29] R. Cole, P. Eklund, and F. Amardeilh. Browsing semi-structured texts on the web using formal concept analysis. *Web Intelligence*, 2003.

[30] R. Cole, P. Eklund, and G. Stumme. Document retrieval for email search and discovery using formal concept analysis. *Applied Artificial Intelligence*, 17(3), 2003.

[31] J. Cooper and R. Byrd. Lexical navigation: visually prompted query expansion and refinement. In *Proceedings of the 2nd ACM Digital Library Conference*, pages 237–246, Philadelphia, USA, 1997.

[32] Credo. http://credo.fub.it/.

[33] W. Croft. *Organising and searching large files of documents*. PhD thesis, University of Cambridge, 1978.

[34] W. Croft and R. Thompson. I3r: A new approach to the design of document retrieval systems. *Journal of the American Society for Information Science*, 38:389–404, 1987.

[35] D. Cutting, D. Karger, and J. Pederson. Constant interaction-time scatter/gather browsing of very large document collections. In *Proceedings of th 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 318–329, 1993.

[36] D. R. Cutting, D. R. Karger, J. O. Pederson, and J. W. Turkey. Scatter/gather: A cluster-based approach to browsing large document collections. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 318–392, 1992.

[37] B. Davey and H. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, 1990.

[38] S. Deerwester, S. Dumais, W. Furnas, T. Landauer, and R. Harshman. Indexing by latent semantic indexing. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[39] D. Dubin. Document analysis for visualization. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 199–204. ACM Press, 1995.

[40] S. Dumais. Improving the retrieval of information from external sources. *Behavior Research Methods, Instruments and Computers*, 23(2):229–236, 1991.

[41] S. Dumais. Latent semantic indexing (lsi) and trec-2. In D. Harman, editor, *The Second Text Retrieval Conference (TREC-2)*, pages 105–116, 1994.

[42] D. Ebert, C. Shaw, A. Zwa, E. Miller, and D. Roberts. Two-handed volumetric document corpus management. *IEEE Computer Graphics and Applications*, 1997.

[43] P. Eklund and R. Cole. Structured ontology and ir for email search and discovery. In *Proceedings of the Sixth Australasian Document Computing Symposium*, Coffs Harbour, Australia, 2001.

[44] B. S. Everitt. *Cluster Analysis*. John Wiley and Sons, 1974.

[45] B. Fernandez-Manjon, J. Cigarran, A.Ñavarro, and A. Fernandez-Valmayor. Applying formal concept analysis to domain modeling in an intelligent help system. In *Proceedings of the Information Technology and Knowledge Systems. 5th IFIP World Computer Congress*, Vienna. Budapest, 1998.

[46] P. Ferragina and A. Gulli. The anatomy of a hierarchical clustering engine for web-page, news and book snippets. In *ICDM '04: Proceedings of the Fourth IEEE International Conference on Data Mining (ICDM'04)*, pages 395–398, Washington, DC, USA, 2004. IEEE Computer Society.

[47] P. Ferragina and A. Gulli. A personalized search engine based on web-snippet hierarchical clustering. In *WWW '05: Special interest tracks and posters of the 14th international conference on World Wide Web*, pages 801–810, New York, NY, USA, 2005. ACM Press.

[48] S. Ferre and O. Ridoux. A file system based on concept analysis. In *Proceedings of the 1st International Conference on Computational Logic*, pages 1033–1047, London, UK, 2000.

[49] C. Fox. *Information Retrieval: Data Structures and Algorithms*, chapter Lexical analysis and stoplists. Prentice Hall, London, UK, 1992.

[50] W. Frakes and R. Baeza-Yates. *Information Retrieval: Data Structures and Algorithms*. Prentice Hall, London, UK, 1992.

[51] B. Fung, K. Wang, and M. Ester. Large hierarchical document clustering using frequent itemsets. In *SDM03*, 2003.

[52] G. Furnas. Experience with an adaptative indexing scheme. In *Proceedings of ACM CHI'85 Conference on Human Factors in Computing Systems*, pages 130–135, San Francisco, USA, 1985.

[53] B. Ganter and R. Wille. *Formal Concept Analysis: Mathematical Foundations*. Springer, 1999.

[54] E. Gaussier, G. Goutte, K. Popat, and F. Chen. A hierarchical model for clustering and categorising documents. In *Proceedings of the 24th European Colloquium on IR Research (ECIR 02)*, pages 229–247, 2002.

[55] M. Giannotti, M.Ñanni, and D. Pedreschi. Webcat: Automatic categorization of web search results. In *11th Italian Symposium on Advanced Database Systems*, 2003.

[56] D. Gifford, P. Jouvelot, M. Sheldon, and J. O'Toole Jr. Semantic file systems. In *Proceedings of the 13th ACM Symposium on Operating Systems Principles*, pages 16–25, 1991.

[57] R. Godin, J. Gecsei, and C. Pichet. Design of a browsing interface for information retrieval. In *Proceedings of the 12th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 32–39. ACM Press, 1989.

[58] R. Godin, R. Missaoui, and A. April. Experimental comparison of navigation in a galois lattice with conventional information retrieval methods. *Internationa Journal of Man-Machine tudies*, 38:747–767, 1993.

[59] Google. http://www.google.com/.

[60] B. Gopal and U. Manber. Integrating content-based access mechanisms with hierarchical file systems. In *Proceedings of the 6th International Conference on Conceptual Structures*, pages 265–278, New Orleans, LA, USA, 1999.

[61] A. Gordon. *Classification*. Chapman & Hall/CRC, second edition, 1999.

[62] H. Griffiths, H. Luckhurst, and P. Willet. Using inter-document similarity information in document retrieval systems. *Journal of the American Society for Information Sciences*, 37:3–11, 1986.

[63] B. Groh, S. Strahringer, and R. Wille. Toscana-systems based on thesauri. In *Proceedings of the 6th International Conference on Conceptual Structures*, pages 127–138, Montpellier, France, 1998.

[64] D. Harman. *Information retrieval. Data structures and algorithms*, chapter Relevance Feedback and other query modification techniques, pages 241–263. Prentice Hall, 1992.

[65] M. Hearst. Tilebars: Visualization of term distribution information in full text information access. In *Proceedings of ACM CHI'95: Human Factors in Computing Systems*, pages 59–66, Denver, CO, USA, 1995.

[66] M. A. Hearst, D. R. Karger, and J. A. Pedersen. Scatter/gather as a tool for the navigation of retrieval results. In *Working Notes AAAI Fall Symp. AI Applications in Knowledge Navigation*, 1995.

[67] M. A. Hearst and J. O. Pedersen. Reexamining the cluster hypothesis: Scatter/gather on retrieval results. In *Proceedings of SIGIR-96, 19th ACM International Conference on Research and Development in Information Retrieval*, pages 76–84, Zurich, CH, 1996.

[68] A. Hotho and G. Stumme. Conceptual clustering of text clusters. In *Proceedings of the FGML Workshop*, Hannover, 2002.

[69] Infoseek. http://www.infoseek.com/.

[70] A. Jain and R. Dubes. *Algorithms for Clustering Data*. Prentice Hall, 1988.

[71] Z. Jiang, R. Joshi, R. Krishnapuram, and L. Yi. Retriever: Inproving web search engine results using clustering. In *Managing Business with Electronic Commerce*, 2002.

[72] B. Johnson and B. Shneiderman. Treemaps: A space-filling approach to the visualization of hierarchical information structures. In *Proceedings of IEEE Visualization 91*, pages 284–291, San Diego, CA, USA, 1991.

[73] A. Kaban and M. Girolami. Clustering of text documents by skewness maximisation. In *Proceedings of the 2nd International Workshop on Independent Component Analysis and Blind Source Separation (ICA 2000)*, pages 435–440, 2000.

[74] G. Kowalski. *Information retrieval systems. Theory and implementation*. Kluwer Academic Publishers, 1997.

[75] K. Kummamuru, R. Lotlikar, S. Roy, K. Singal, and R. Krishnapuram. A hierarchical monothetic document clustering algorithm for summarization and browsing search results. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 658–665, New York, NY, USA, 2004. ACM Press.

[76] S. Kuznetsov and S. Obiedkov. Comparing performance of algorithms for generating concept lattices. *Journal of Experimental and Theoretical Artificial Intelligence*, 14(2–3):189–216, 2002.

[77] E. Lagergren and P. Over. Comparing interactive information retrieval systems across sites: the trec-6 interactive track matrix experiment. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 164–172. ACM Press, 1998.

[78] D. Lawrie, W. B. Croft, and A. Rosenberg. Finding topic words for hierarchical summarization. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 349–357, New York, NY, USA, 2001. ACM Press.

[79] D. J. Lawrie and W. B. Croft. Generating hierarchical summaries for web searches. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 457–458, New York, NY, USA, 2003. ACM Press.

[80] A. Leuski. Evaluating document clustering for interactive information retrieval. In *Proceedings of the ACM CIKM 2001*, 2001.

[81] A. Leuski. Interactive information organization: Techniques and evaluation. Master's thesis, University of Massachusetts at Amherst, 2001.

[82] A. Leuski and J. Allan. Strategy-based interactive cluster visualization for information retrieval. *International Journal on Digital Libraries*, 3(2):170–184, August 2000.

[83] D. Lewis. *Text-based Intelligent Systems*, chapter Text representation for intelligent text retrieval: A classification-oriented view, pages 179–197. Lawrence Erlbaum, Hillsdale, NJ, 1992.

[84] C. Lindig. Concept-based component retrieval. In *Working Notes of the IJCAI-95 Workshop: Formal Approaches to Reuse of Plans, Proofs, and Programs*, pages 21–25, Montreal, Canada, 1995.

[85] C. Lindig. Fast concept analysis. In *Working with conceptual structures. Contribution to the 8th International Conference on Conceptual Structures*, pages 152–161, Darmstadt, Germany, 2000.

[86] D. Lucarella, S. Parisotto, and A. Zanzi. More: Multimedia object retrieval environment. In *Proceedings of ACM Hypertext'93*, pages 39–50, Seattle, WA, USA, 1993.

[87] Y. Maarek, D. Berry, and G. Kaiser. An information retrieval approach for automatically constructing software libraries. *IEEE Transactions on Software Engineering*, 17(8):800–813, 1991.

[88] Y. Maarek, R. Fagin, I. Ben-Shaul, and D. Pelleg. Ephemeral document clustering form web aplications. Technical Report RJ 10186, IBM Research, 2000.

[89] MailSleuth. http://www.mail-sleuth.com.

[90] Mooter. http://www.mooter.com/.

[91] MSN. http://www.msn.com.

[92] D.Ñorman. *User Centered Systems Design*, chapter Cognitive engineering, pages 31–61. Lawrence Erlbaum Associates, 1986.

[93] NorthernLight. http://www.nothernlight.com/.

[94] OpenDirectoryProject(ODP). http://dmoz.org.

[95] S. Osinski and D. Weiss. Conceptual clustering using lingo algorithm: Evaluation on open directory project data. In *New Trends in Intelligent Information Processing and Web Mining (IIPWM 04)*, 2004.

[96] A. Peñas. *Website Term Browser. Un Sistema Interactivo y multilingüe de Búsqueda Textual Basado en Técnicas Lingüísticas*. PhD thesis, ETSI Informatica. Universidad Nacional de Educación a Distancia, 2002.

[97] A. Peñas, F. Verdejo, and J. Gonzalo. Corpus-based terminology extraction applied to information access. In *Proceedings of Corpus Linguistics 2001*, Lancaster University, 2001.

[98] G. Pedersen. A browser for bibliographic information retrieval based on an application of lattice theory. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 270–279, Pittsburgh, PA, USA, 1993.

[99] F. Pereira, N. Tishby, and L. Lee. Distributional clustering of english words. In *Proceedings of the Association for Computational Linguistics*, pages 183–190, 1993.

[100] C. Peters, M. Braschler, J. Gonzalo, and M. Kluck, editors. *Evaluation of cross-language information retrieval systems*, volume 2406 of *LNCS*. Springer, 2002.

[101] U. Priss. Lattice-based information retrieval. In *Knowledge Organization*, volume 27, pages 132–142, 2000.

[102] S. Robertson and K. Spark Jones. Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27:129–146, 1976.

[103] S. Robertson and S. Walker. Some simple effective aproximations to the 2-poisson method for probabilistic weighted retrieval. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 311–317, Dublin, Ireland, 1994.

[104] S. Robertson, S. Walker, and M. Beaulieu. Okapi at trec-7: Automatic ad hoc, filtering, vlc, and interactive track. In *Proceedings of the 7th Text Retrieval Conference (TREC-7), NIST Special Publication 500-242*, pages 253–264, Gaithersburg, MD, USA, 1998.

[105] J. Rocchio. Document retrieval systems. optimization and evaluation. Master's thesis, Harvard, 1966.

[106] D. Rose and D. Levinson. Understanding user goals in web search. In *Proceedings of the 13th international conference on World Wide Web (WWW 2004)*, pages 13–19, New York, USA, May 2004. ACM Press.

[107] G. Salton. *The SMART retrieval systems*. Prentice Hall, 1971.

[108] G. Salton. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1989.

[109] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA, 1986.

[110] R. Sibson. Slink: an optimally efficient algorithm for the single link cluster method. *Computer Journal*, 16:30–34, 1973.

[111] C. Silverstein and J. O. Pedersen. Almost-constant-time clustering of arbitrary corpus subsets. In *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 60–66, 1997.

[112] H. Small and E. Sweeney. Clustering the science citation index using cocitations. *Scientometrics*, 7:391–409, 1985.

[113] SnakeT. http://snaket.di.unipi.it/.

[114] P. Sneath and R. Sokal. *Numerical Taxonomy*. San Francisco, Freeman, 1973.

[115] D. Soergel. Mathematical analysis of documentation systems. *Information storage and retrieval*, 3:129–173, 1967.

[116] A. Spoerri. Infocrystal: Integrating exact and partial matching approaches through visualization. In *Proceedings of RIAO 94: Intelligent Multimedia Information Retrieval Systems and Management*, pages 687–696, New York, USA, 1994.

[117] J. Stasko and E. Zhang. Focus+context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations. In *Proceedings of IEEE Symposium of InfoVis 2000*, pages 57–65, Salt Lake City, UT, USA, 2000.

[118] F. van der Merwe and D. Kourie. Compressed pseudo-lattices. *Journal of Experimental and Theoretical Artificial Intelligence*, 14(2–3):229–254, 2002.

[119] C. van Rijsbergen. *Information Retrieval*. Butterworths, London, second edition edition, 1979.

[120] A. Veerasamy and N. J. Belkin. Evaluation of a tool for visualization of information retrieval results. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 85–92, 1996.

[121] A. Veerasamy and R. Heikes. Effectiveness of a graphical display of retrieval results. In *Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 236–245, New York, NY, USA, 1997. ACM Press.

[122] Vivissimo. http://www.vivisimo.com.

[123] F. Vogt, C. Wachter, and R. Wille. *Classification, Data Analysis and Knowledge Organization*, chapter Data Analysis based on a conceptual file, pages 131–140. Springer, 1991.

[124] F. Vogt and R. Wille. *Graph Drawing'94*, chapter TOSCANA. A graphical tool for analysing and exploring data, pages 226–233. Springer, 1995.

[125] E. Voorhees. The cluster hypothesis revisited. In *Proceedings of the 8th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 188–196. ACM Press, 1985.

[126] E. Voorhees. Using wordnet to disambiguate word senses for text retrieval. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 171–180, Pittsburg, PA, USA, 1993.

[127] Y. Wang and M. Kitsuregawa. On combining link and contents information for web page clustering. In *13th International Workshop on Database and Expert Systems Applications (DEXA 02)*, 2002.

[128] W. Weili, X. Hui, and S. S. *Clustering and Information Retrieval*, volume 11 of *Network Theory and Applications*. Kluwer Academic Publishers, 2004.

[129] D. Weiss and J. Stefanowski. Web search results clustering in polish: Experimental evaluation of carrot. In *Intelligent Information Systems (IIS 03)*, 2003.

[130] R. Wille. *Ordered Sets*, chapter Restructuring lattice theory : An approach based on hierarchies of concepts, pages 445–470. Prentice Hall, 1982.

[131] R. Wille. Line diagrams of hierarchical concept systems. *International Classification*, 11(2):77–86, 1984.

[132] R. Wille. Concept lattices and conceptual knowledge systems. *Computers and Mathematics with Applications*, 23:493–522, 1992.

[133] R. Wille. *Classification in the Information Age*, chapter Conceptual Landscapes of knowledge: A pragmatic paradigm for knowledge processing, pages 344–356. Springer, 1999.

[134] P. Willet. Similarity coefficients and weighting functions for automatic document classification: an empirical comparision. *International Classification*, 10:138–142, 1983.

[135] P. Willet. Recent trends in hierarchic document clustering: A critical review. *Information Processing & Management*, 24(5):577–597, 1988.

[136] Y. Wu and X. Chen. Extracting features from web search returned hits for hierarchical classification. In *Proceedings of IKE 03*, 2003.

[137] J. Xu and W. Croft. Query expansion using local and global document analysis. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 4–11, Zurich, Switzerland, 1996.

[138] J. Xu and W. Croft. Improving the effectiveness of information retrieval with local context analysis. *ACM Transactions on Information Systems*, 18(1):79–112, 2000.

[139] Yahoo. http://www.yahoo.com/.

[140] YahooDirectory. http://dir.yahoo.com/.

[141] C. Yu, W. Meng, and S. Park. A framework for effective retrieval. *ACM Transactions on Database Systems*, 14(2):147–167, 1989.

[142] O. Zamir and O. Etzioni. Web document clustering: A feasibility demonstration. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 46–54, 1998.

[143] O. Zamir and O. Etzioni. Grouper: a dynamic clustering interface to web search results. *WWW8, Computer Networks (Amsterdam, Netherlands: 1999)*, 31(11–16):1361–1374, 1999.

[144] H. Zeng, Q. He, Z. Chen, and W. Ma. Learning to cluster web search results. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 210–217, Sheffield, United Kingdom, 2004.

[145] D. Zhang and Y. Dong. Semantic, hierarchical, online clustering of web search results. In *WIDM01*, 2002.