UNIVERSIDAD DE GRANADA

ESTUDIO Y EVALUACIÓN DE UN SISTEMA INTELIGENTE PARA LA RECUPERACIÓN Y EL FILTRADO DE INFORMACIÓN DE INTERNET

TESIS DOCTORAL

Juan José Samper Márquez

Granada 2005

Departamento de Arquitectura y Tecnología de Computadores

Editor: Editorial de la Universidad de Granada
Autor: Juan José Samper Márquez
D.L.: Gr. 1750 - 2005
ISBN: 84-338-3603-x

UNIVERSIDAD DE GRANADA

ESTUDIO Y EVALUACIÓN DE UN SISTEMA INTELIGENTE PARA LA RECUPERACIÓN Y EL FILTRADO DE INFORMACIÓN DE INTERNET

Memoria presentada por

Juan José Samper Márquez

Para optar al grado de

DOCTOR EN INFORMÁTICA

Fdo. Juan José Samper Márquez

D. Juan Julián Merelo Guervós, Profesor Titular de Universidad y **D. Pedro Ángel Castillo Valdivieso**, Profesor Asociado, del Departamento de Arquitectura y Tecnología de la Universidad de Granada.

CERTIFICAN

Que la memoria titulada "*Estudio y Evaluación de un Sistema Inteligente para la Recuperación y el Filtrado de Información de Internet*" ha sido realizada por **D. Juan José Samper Márquez** bajo nuestra dirección en el Departamento de Arquitectura y Tecnología de Computadores de la Universidad de Granada para optar al grado de Doctor en Informática.

Granada, a 30 de septiembre de 2005

Fdo. Juan Julián Merelo Guervós
Director de la Tesis

Fdo. Pedro Ángel Castillo Valdivieso
Director de la Tesis

A mi hijo.

Agradecimientos

Mi respeto y agradecimiento profundo a todas las personas que me han ayudado en algún momento durante la elaboración de esta Tesis, especialmente a mis Directores de Tesis, el profesor J.J. Merelo y el profesor Pedro Castillo, por su paciencia y dedicación.

Resumen

En esta tesis se desarrolla un nuevo sistema de recuperación y filtrado de información, denominado **NectaRSS**, que recomienda información a un usuario basándose en los intereses de éste. El método realiza automáticamente la tarea de adquisición de las preferencias del usuario, evitando la realimentación explícita.

Se realiza una revisión de todos los conceptos relacionados con el sistema, mostrando diferentes enfoques desde los que la comunidad científica ha abordado el problema, con especial incidencia en el contexto de la Web, donde se aplicará inicialmente.

Por último, se comprueba la efectividad del método propuesto aplicándolo a la implementación de un agregador inteligente, utilizado por diversos usuarios heterogéneos, demostrándose su capacidad para ofrecer la información personalizada según los intereses de cada individuo.

Abstract

In this thesis a new system called **NectaRSS** for information retrieval and filtering is presented. The system recommends information to a user based on his past choices. The method automatically accomplishes the task of user preferences acquisition avoiding explicit feedback.

In this work, a review of all the concepts related to the system is first performed, showing different approaches to the problem of user profile construction, emphasizing web information retrieval systems, where NectaRSS will be initially applied.

The efficiency of the proposed method is proved applying it to the implementation of an intelligent aggregator, used by different and heterogeneous users, proving its ability to offer the information personalized according to each individual's interests.

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

ACRÓNIMOS Y SÍMBOLOS MÁS UTILIZADOS EN LA PRESENTE MEMORIA

RI	Recuperación de Información
SRI	Sistema de Recuperación de Información
E-P	Par Exhaustividad-Precisión
P	Perfil de usuario
P_s	Perfil de sesión
P_r	Perfil de resumen
T	Conjunto de titulares
E(T)	Conjunto de titulares elegidos
D(T)	Conjunto de titulares destacados
CRS	Con Resumen – Sin resumen
DIV	Determinación del Intervalo de Vida
IRP	Importancia Relativa de los Perfiles
CRS2	Con Resumen – Sin resumen (2)[1]
PAU	Prueba del Algoritmo con diferentes Usuarios
PPA	Probar Puntuación Alternativa
tf_{ij}	Frecuencia de aparición del término t_j en el documento d_i
$tf_{h,k}$	Frecuencia del término t_k en el titular h
w_{ij}	Relevancia del término t_j en el documento d_i
w^h	Vector característica del titular h
$sim(P, w^h)$	Similitud entre el perfil P y el vector característica w^h
fol	Factor de olvido
C_P	Tasa que mide el porcentaje de titulares elegidos

[1] *Es un experimento similar a CRS pero utilizando los valores hallados empíricamente para ciertos parámetros.*

C_R	Tasa que mide el porcentaje de titulares ofrecidos destacados
C_T	Tasa que mide el porcentaje de titulares elegidos destacados
C_D	Tasa que relaciona la puntuación media de los titulares escogidos con la puntuación media máxima
$\lvert\overline{E}\rvert$	Error Absoluto Medio
σ	Desviación Estándar del Error
r	Coeficiente de Correlación entre titulares
RP(i)	R-Precisión en la sesión i

Capítulo 1

INTRODUCCIÓN

En pocos años, Internet se ha convertido en un medio de comunicación prácticamente indispensable y en la principal fuente de información para una parte importante de la población del mundo desarrollado.

Así, la Web[1], con más de 8 mil millones de páginas, según *Google*[2] a septiembre de 2005, se está convirtiendo rápidamente en la indiscutible opción de búsqueda cuando se tiene necesidad de información. Su uso resulta cada vez más importante para buscar o intercambiar información, para expresar o leer opiniones acerca de la actualidad en todo tipo de campos, y para estar al día en las noticias de todos los ámbitos procedentes de fuentes muy variadas.

En general, dada la gran cantidad de fuentes de información disponibles actualmente en la Web, es probable que un amplio subconjunto de éstas sea del interés de un usuario, encontrándose con tal cantidad información que le resulte prácticamente inabarcable. Así, en muchos casos el usuario se limitará a explorar la información hallada hasta cansarse aún cuando no haya cubierto su necesidad informativa. Si la información ofrecida es muy amplia, su revisión resultará probablemente una carga de trabajo más que una satisfacción. Además, tal cantidad de información contendrá con seguridad artículos más interesantes que otros para un usuario concreto. Por ello, se buscará una estrategia que pueda aliviar la sobrecarga de información a los usuarios y que ofrezca la información ordenada según las preferencias o necesidades del usuario, obteniendo éstas de forma automática.

Nuestro **objetivo primordial** es crear un sistema de filtrado o priorizado de información, que la presente a un usuario en orden de importancia según sus preferencias, que denominaremos **NectaRSS**.

---

[1] *"Web" es un término que proviene del inglés y significa "red informática", según [RAE, 2003]. En general se refiere a la "World Wide Web" o telaraña mundial. También puede referirse a un "documento situado en una red informática, al que se accede mediante enlaces de hipertexto" [RAE, 2003], y que normalmente se denomina página web.*

[2] *http://www.google.com*

Como **segundo objetivo**, buscaremos una forma de obtener las preferencias del usuario sin esfuerzo adicional para éste. Desarrollaremos un método automático basado en el historial de lectura de la información ofrecida. Así, nuestra propuesta será la confección incremental de un perfil de usuario en base a las selecciones de información que vaya realizando tal usuario.

Finalmente, como **tercer objetivo**, habrá que encontrar la forma óptima de crear ese perfil de usuario, y de usarlo para dar la información más relevante, y evaluar diferentes estrategias y opciones para que el resultado sea óptimo.

1.1. Organización de la tesis

Esta tesis se organiza de la forma siguiente:

- El **Capítulo 2** se dedica al estudio de los sistemas de recuperación de información y de los modelos utilizados para ello, incidiendo especialmente en el modelo vectorial de Salton. Así, se repasan los conceptos fundamentales de los sistemas de recuperación de información, el modelo conceptual, la realimentación de la relevancia, el agrupamiento o "clustering" de documentos, la extracción y el pesado automático de términos. La segunda parte del capítulo se dedica a la Web como sistema de recuperación de información, tratándose los métodos de recuperación específicos para ésta, las herramientas de búsqueda que se utilizan en dicho contexto y los sistemas de recomendación. La necesidad de este capítulo se fundamenta en el conocimiento de los sistemas de recuperación de información, de la Web en particular, y en conocer los modelos típicos para representar los documentos. NectaRSS es un sistema de recuperación de información que utilizará el modelo vectorial.

- En el **Capítulo 3** se estudian las principales técnicas de evaluación de los sistemas de recuperación de información y se definen conceptos como la *relevancia* y la *pertinencia*. Se comienza repasando los métodos tradicionales de evaluación, destacando las medidas basadas en la *relevancia*: la *precisión* y la *exhaustividad* principalmente, y la relación entre éstas. Se analizan diversos métodos para estimar la *exhaustividad*, así como las medidas promedio *exhaustividad-precisión*. También se tratan los valores sumarios simples, especialmente la *R-Precisión*, y otras medidas alternativas como la *exhaustividad y precisión normalizadas*, la *ratio de deslizamiento* y la

medida de Voiskunskii. El capítulo proporciona un conocimiento general de las técnicas de evaluación de los sistemas de recuperación de la información, necesario para aplicar dichas técnicas al sistema experimental NectaRSS.

- El **Capítulo 4** define y clarifica diversos aspectos de un perfil de usuario. Además se comentan los principales métodos para su creación. Se exponen diversas técnicas para adquirir los datos del usuario, tales como la información explícita, las reglas de adquisición, el reconocimiento del plan, la utilización de estereotipos y la adquisición de datos de utilización. Entonces se aborda la representación del perfil de usuario y las técnicas de inferencia asociadas, distinguiendo tres tipos de razonamiento: deductivo, inductivo y analógico. Otro tema tratado es la realimentación del usuario ya que ésta permitirá a dicho usuario actualizar su perfil correspondiente. Para finalizar el capítulo se comentan algunas técnicas alternativas utilizadas en la creación de perfiles de usuario, la utilización de agentes software, los modelos estadísticos, el razonamiento basado en reglas y la agrupación o "clustering" de perfiles, sin olvidar que un sistema puede combinar varias de ellas. También se comenta un ejemplo real de sistema de búsqueda adaptativa en la Web basado en un perfil de usuario automático, en el cual se inspirará parte de nuestro trabajo. En este capítulo se proporciona una visión amplia de los perfiles de usuario, que resultará útil para el diseño de un método propio que capte las preferencias de los usuarios. NectaRSS utilizará un perfil de usuario para representar las preferencias de éste.

- En el **Capítulo 5** se expone nuestra propuesta para un sistema de recuperación y recomendación de información de la Web, así como su aplicación en un agregador inteligente. Trataremos los diversos aspectos teóricos que fundamentan el sistema, comenzando por las estrategias que se utilizarán para la construcción de un perfil de usuario automático basado en su historia de navegación. Se considerará la utilización del modelo vectorial y el esquema *tf*, descritos en el Capítulo 2, y se verá cómo se puntúa la información que se ofrece al usuario mediante la medida del coseno propuesta por Salton. Se finaliza con una descripción general del sistema propuesto, que se denominará NectaRSS. Este capítulo es necesario para conocer la base teórica que subyace en dicho sistema.

- El **Capítulo 6** trata de la evaluación experimental del sistema propuesto, así se expondrá el esquema general de experimentación y se detallará la metodología

seguida. A continuación se comentan las distintas estrategias que se utilizarán en la experimentación, describiendo el tratamiento de las palabras y los experimentos que se desarrollarán. Entonces se proponen diversas medidas para la evaluación del sistema, en base a las variables consideradas en los experimentos, distinguiendo distintas tasas o medidas porcentuales de valor simple. Otras medidas estarán referidas a la puntuación que el sistema otorga a los distintos titulares de información. Se comparará también la distinta información que selecciona el usuario respecto a la que le ofrece el sistema, empleando para ello medidas como el *Error Medio Absoluto*, la *Desviación Estándar* del error, la *Correlación* entre titulares y la *R-Precisión* descrita por [Baeza, 1999]. Así, este capítulo servirá para conocer qué medidas se utilizan y cómo se evalúa el funcionamiento del sistema experimental propuesto, NectaRSS.

- En el **Capítulo 7** se exponen los experimentos realizados y los resultados obtenidos. Estos resultados se analizan y se representan gráficamente, para extraer conclusiones que permitan determinar diversos parámetros del sistema y para evaluar el funcionamiento del sistema propuesto con diversos usuarios, calibrando su funcionamiento en el "mundo real". Este capítulo servirá para comprobar la efectividad del sistema NectaRSS analizando los valores obtenidos por las medidas que evalúan su funcionamiento.

- Finalmente, el **Capítulo 8** presenta en forma sintética las conclusiones y principales aportaciones de esta tesis. Además se enumeran los objetivos que se han cumplido y se proponen diversas líneas de investigación identificadas en el desarrollo de la tesis. Es un resumen de los logros, aportaciones y posibles líneas a seguir, a partir de la investigación con NectaRSS.

Capítulo 2

LOS SISTEMAS DE RECUPERACIÓN DE INFORMACIÓN

En este capítulo se presentarán un conjunto de conceptos e ideas que se han desarrollado en el campo de los **sistemas de recuperación de información**, en adelante **sistemas RI** o **SRI**. Se abordará el concepto de **recuperación de información** y se expondrán distintos modelos sobre los que se basan los sistemas RI, destacando especialmente la recuperación de información en la Web y los sistemas de recomendación.

El fundamento de esta introducción teórica es proporcionar una base para la tesis; NectaRSS es un sistema RI: se pretenden identificar las informaciones relevantes en el área de interés de los usuarios, analizando para ello el contenido de los documentos, se realizarán correspondencias entre los contenidos de las fuentes analizadas y los intereses de cada usuario, destacando entonces las informaciones más relevantes. Asimismo se realizarán los ajustes necesarios en el sistema captando de manera automática las preferencias de los usuarios mediante un mecanismo de realimentación implícita. De esta manera se podrá recomendar la información a cada usuario.

2.1. Introducción

La *recuperación de información* "se trata de una disciplina que involucra la localización de una determinada información dentro de un almacén de información o base de datos" [Meadow, 1993]. Pérez-Carballo afirma que "una típica tarea de la recuperación de información es traer documentos relevantes desde un gran archivo en respuesta a una pregunta formulada por un usuario y ordenar estos documentos de acuerdo con su relevancia" [Pérez, 2000]. Para Grossman y Frieder "la recuperación de información es encontrar documentos relevantes, no encontrar simples correspondencias a unos patrones de bits" [Grossman, 1998].

Baeza-Yates utiliza la definición de recuperación de información elaborada por Salton: "la recuperación de la información tiene que ver con la representación,

almacenamiento, organización y acceso a los ítems de información" [Baeza, 1999]. Baeza define el problema de la recuperación de información como: "dada una necesidad de información y un conjunto de documentos, ordenar los documentos de más a menos relevantes para esa necesidad y presentar un subconjunto de aquellos de mayor relevancia" [Baeza, 1999].

Para Salton "la recuperación de información se entiende mejor cuando uno recuerda que la información que se procesa consiste en documentos", de esta manera se diferencian a los sistemas encargados de su gestión de otros tipos de sistemas, como los gestores de bases de datos relacionales. "Cualquier SRI puede describirse como un conjunto de ítems de información, un conjunto de peticiones y algún mecanismo que determine qué ítem satisface las necesidades de información expresadas por el usuario en la petición" [Salton, 1983]. Además considera "el uso de una clasificación o de un sistema de indización".

Otros autores como Croft consideran que la recuperación de información será "el conjunto de tareas mediante las cuales el usuario localiza y accede a los recursos de información que son pertinentes para la resolución del problema planteado" [Croft, 1987].

2.2. Modelos para la recuperación de información

Para realizar el diseño de un SRI se debe utilizar un modelo, en el que se definirá cómo se obtienen las representaciones de los documentos y de la consulta, la estrategia para evaluar la relevancia de un documento respecto a una consulta, los métodos para establecer la importancia u orden de los documentos de salida y los mecanismos que permiten una realimentación por parte del usuario para mejorar la consulta.

Una propuesta de clasificación de los modelos de recuperación es la realizada por [Dominich, 2000], que se muestra en la tabla 2.1.

Partiendo de la tarea inicial que realiza el usuario, es posible realizar una clasificación como la propuesta por Baeza-Yates, que considera la recuperación de información a partir de una ecuación de búsqueda, o bien mediante la consulta de documentos en busca de referencias interesantes [Baeza, 1999]. Así, en esta clasificación se introducen los modelos basados en la navegación entre páginas web: de estructura plana, de estructura guiada o de hipertexto, según puede verse en la tabla 2.2.

Modelo	Descripción
Clásicos	Booleanos, Probabilísticos y basados en el Espacio Vectorial.
Alternativos	Basados en la Lógica Fuzzy.
Lógicos	Basados en la Lógica Formal.
Basados en la interactividad	Posibilidades de expansión del alcance de la búsqueda y uso de retroalimentación por relevancia.
Basados en la Inteligencia Artificial	Redes neuronales, bases de conocimiento, algoritmos genéticos y procesamiento de lenguaje natural.

Tabla 2.1. Propuesta de clasificación de los Modelos de Recuperación de Información. Fuente [Dominich,2000].

Modalidad	*Vista lógica de los documentos*			
		Términos índice	**Texto Completo**	**Texto Completo + Estructura**
	Recuperación	Clásicos Conjuntos teóricos Algebraicos Probabilísticos	Clásicos Conjuntos teóricos Algebraicos Probabilísticos	Estructurados
	Navegación	Estructura plana	Estructura plana Hipertexto	Estructura guiada Hipertexto

Tabla 2.2. Otra propuesta de clasificación de los Modelos de Recuperación de Información, según la modalidad y la vista lógica de los documentos. Fuente: [Baeza, 1999].

2.2.1. El Modelo Vectorial

Este modelo es muy utilizado en los sistemas RI, el primer sistema que implementó el modelo vectorial fue el *SMART* de Salton [Salton, 1971, 1983]. En el sistema *SMART* cada documento estaba representado por un vector de términos, y cada componente del vector representaba el peso w_{ij} del término t_j presente en el documento d_i. De esta manera, la representación lógica de cada documento será un vector de pesos $d_i = (w_{i1}, w_{i2}, \ldots, w_{im})$, donde w_{ij} indicará el grado de relevancia de que el término t_j esté presente en el documento d_i. Este peso suele estar relacionado con la frecuencia de aparición del término.

Estos sistemas permiten añadir a los términos de las consultas distintos pesos en función de lo relevante que sea cada término de la consulta para el usuario. Así, una colección de documentos se puede representar por una matriz en la que cada fila se refiera a un documento y cada columna a un término, según se muestra en la figura 2.1.

	t_1	t_2	t_3	…	t_j	…	t_m
d_1	w_{11}	w_{12}	w_{13}	…	w_{1j}	…	w_{1m}
d_2	w_{21}	w_{22}	w_{23}	…	w_{2i}	…	w_{2m}
..	..	..	..	..	..	..	..
d_i	w_{i1}	w_{i2}	w_{i3}	…	w_{ij}	…	w_{im}
..	..	..	..	..	..	..	..
d_n	w_{n1}	w_{n2}	w_{n3}	…	w_{nj}	…	w_{nm}

Figura 2.1. Matriz de pesos de términos para el Modelo Vectorial. Fuente: [Llidó, 2002].

Una consulta podrá representarse de igual misma manera que un documento, asignándole un vector de pesos asociados a los términos, representando así la importancia de los términos en la consulta: $q_k = (w_{k1}, w_{k2}, \ldots, w_{km})$.

En el modelo vectorial se proponen las siguientes propiedades para los términos:

- tf_{ij}: es la frecuencia de aparición del término t_j en el documento d_i.
- df_j: indica el número de documentos en los que aparece el término t_j.

A partir de éstas, el peso w_{ij} se calcula frecuentemente según la siguiente función:

- $w_{ij} = tf_{ij} \cdot idf_j$, donde *idf* es la función inversa de *df* o *frecuencia inversa del documento*. Así, $idf_j = log_2 (N/df_j)$, siendo N el número total de documentos.

Un ejemplo de sistema que hace uso del modelo vectorial es el propuesto por [Crabtree y Soltysiak, 1998]. Este sistema monitoriza la navegación del usuario en la Web y su uso del correo electrónico para derivar sus intereses. Los documentos se representarán mediante vectores con el peso de las N palabras más representativas. Los pesos de las palabras se obtienen aplicando la regla *tf· idf*, donde *tf* representa la frecuencia del término e *idf* representa la *frecuencia inversa del documento*.

El modelo vectorial hace la suposición básica de que la proximidad relativa entre dos vectores es proporcional a la distancia semántica de los documentos. En la figura 2.2 [Salton, 1989] se muestran las distancias más utilizadas como medidas de similitud en los sistemas RI vectoriales.

Medida de Similitud	*Modelo Vectorial*
Producto escalar	$\sum_{i=1}^{m} X_i \cdot Y_i$
Coeficiente de Dice	$\frac{2 \cdot \sum_{i=1}^{m} X_i \cdot Y_i}{\sum_{i=1}^{m} X_i^2 + \sum_{i=1}^{m} Y_i^2}$
Coeficiente del coseno	$\frac{\sum_{i=1}^{m} X_i \cdot Y_i}{\sqrt{\sum_{i=1}^{m} X_i^2 \cdot \sum_{i=1}^{m} Y_i^2}}$
Coeficiente de Jaccard	$\frac{\sum_{i=1}^{m} X_i \cdot Y_i}{\sum_{i=1}^{m} X_i^2 + \sum_{i=1}^{m} Y_i^2 - \sum_{i=1}^{m} X_i \cdot Y_i}$

Figura 2.2. Medidas de similitud entre dos vectores de términos en el modelo vectorial. Fuente: [Salton, 1989].

Una de las medidas de similitud más utilizadas es la del coseno. La relación coseno medirá el coseno del ángulo entre documentos y consultas, ya que éstos se representarán como vectores en un espacio multidimensional de dimensión *t*. Así, podemos expresar la medida de similitud entre un documento d_i y una consulta q_k, siendo *m* el número de términos, como:

$$sim(d_i, q_k) = \frac{\vec{d}_i \cdot \vec{q}_k}{\left|\vec{d}_i\right| \cdot \left|\vec{q}_k\right|} = \frac{\sum_{j=1}^{m} w_{ij} \cdot w_{kj}}{\sqrt{\sum_{j=1}^{m} w_{ij}^2 \cdot \sum_{j=1}^{m} w_{kj}^2}} \qquad (2.1)$$

Un ejemplo de cálculo de la similitud, tomado de [Raymond, 2005], puede observarse en la figura 2.3 donde aparecen representados dos documentos d_1, d_2 y una consulta *q* respecto a los ejes t_1, t_2 y t_3.

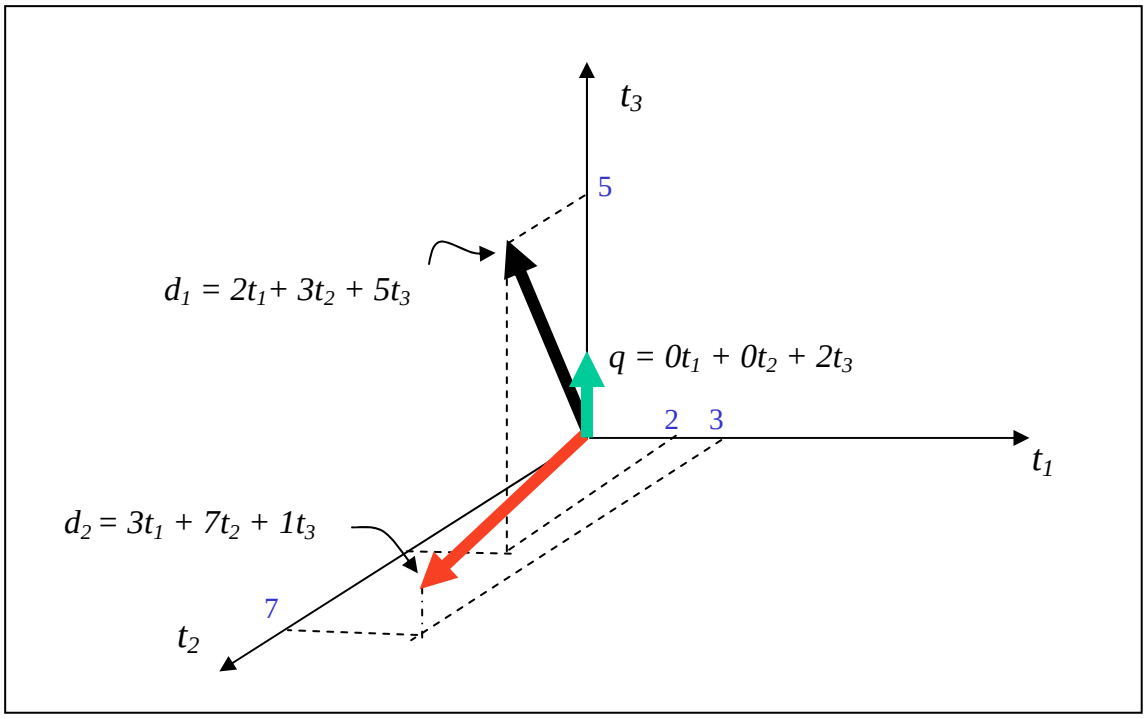


Figura 2.3. Representación gráfica de una consulta q junto a dos documentos d_1 y d_2 utilizando el modelo vectorial. Fuente: [Raymond, 2005].

El cálculo de la similitud entre los documentos d_1, d_2 y la consulta q del ejemplo, se efectuará como sigue:

$$sim(d_1, q) = \frac{2 \cdot 5}{\sqrt{(4+9+25) \cdot (0+0+4)}} = 0.81$$

$$sim(d_2, q) = \frac{2 \cdot 1}{\sqrt{(9+49+1) \cdot (0+0+4)}} = 0.13$$

teniendo en cuenta que $d_1 = (2, 3, 5)$, $d_2 = (3, 7, 1)$ y $q = (0, 0, 2)$.

De los resultados se deduce que el documento d_1 es bastante más similar a la consulta q que el documento d_2, o lo que es lo mismo, que el ángulo θ_1 entre el vector que representa a d_1 y el vector que representa a q es menor que el ángulo θ_2 entre el vector que representa a d_2 y el vector que representa a q, tal y como puede verse en la figura 2.4.

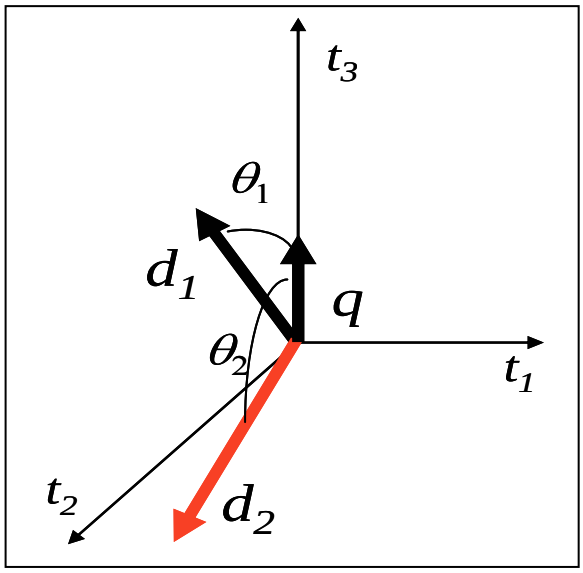


Figura 2.4. Representación gráfica de los ángulos θ_1 y θ_2 entre los vectores de los documentos d_1 y d_2 y la consulta q, para el ejemplo de cálculo de similitud en el modelo vectorial descrito. Fuente: [Raymond, 2005].

Al contar con una medida de similitud como la del coseno entre cada documento y una consulta dada, será posible considerar un umbral en la recuperación de los documentos, de forma que se consideren relevantes aquellos cuyo valor en la fórmula (2.1) sea, por ejemplo, mayor o igual a 0.6. De este modo podemos considerar búsquedas no exactas. Los documentos pueden entonces presentarse al usuario en un orden decreciente de similitud.

2.2.1.1 Realimentación de la Relevancia

Si se le presenta al usuario una lista de documentos relevantes, y dicho usuario realiza un juicio sobre la relevancia de los documentos recuperados con respecto a la consulta, esta información podrá ser utilizada por el sistema para construir nuevos vectores de consulta. A este proceso se le conoce como "relevance feedback", o realimentación de la relevancia. Entonces, las consultas reformuladas podrán compararse con los documentos de la base de documentos para obtener un nuevo conjunto de documentos relevantes. La finalidad de este proceso es obtener una nueva consulta que muestre un mayor grado de similitud con los documentos identificados previamente como relevantes, y al mismo tiempo, que sea menos similar a los documentos marcados como poco relevantes por el usuario. De esta manera, las consultas reformuladas deberán recuperar más documentos relevantes y menos documentos irrelevantes que las consultas previamente formuladas.

La reformulación de consultas se basa en las dos operaciones complementarias siguientes:

- Los términos que aparecen en los documentos identificados previamente como relevantes por el usuario se añadirán al vector de la consulta original, o su peso se incrementará por un factor si ya se encontraban en dicho vector.
- Los términos que aparecen en los documentos previamente identificados como no relevantes por el usuario se eliminarán del vector de la consulta o su peso será reducido.

Este proceso de realimentación de la relevancia podrá aplicarse tantas veces como se requiera para mejorar el resultado de la consulta.

2.2.1.2 Agrupación o "clustering" de documentos

La fórmula (2.1) de la medida del coseno se ha utilizado para medir la similitud entre un documento y una consulta, pero también se puede utilizar para determinar la similitud entre pares de documentos. Así, dados los vectores de dos documentos, d_i, y d_j, la similitud entre ellos puede definirse como:

$$sim(d_i, d_j) = \frac{\vec{d}_i \cdot \vec{d}_j}{\left|\vec{d}_i\right| \cdot \left|\vec{d}_j\right|} = \frac{\sum_{k=1}^{m} w_{ik} \cdot w_{jk}}{\sqrt{\sum_{k=1}^{m} w_{ik}^2 \cdot \sum_{k=1}^{m} w_{jk}^2}} \qquad (2.2)$$

Si determinamos la similitud entre pares de documentos se podrá construir un agrupamiento de documentos. Cada clase o "cluster" agrupará documentos similares a un representante de esa clase, denominado *centroide*.

Dado un conjunto de m documentos que constituyen una clase p, el centroide $C_p=(c_{p1}, c_{p2},\ldots, c_{pk})$ se puede calcular como la media aritmética de los vectores de los documentos incluidos en dicha clase. El peso del término k del centroide de la clase p puede calcularse como la media de los pesos del término k en todos los m vectores de documentos en la clase p:

$$c_{pk} = \frac{\sum_{i=1}^{m} w_{ik}}{m} \qquad (2.3)$$

De esta manera, al organizar los documentos en clases, la búsqueda de un documento se realizará en dos etapas. En primer lugar, la consulta se comparará con los centroides de cada clase, calculando los correspondientes coeficientes de similitud. Luego, los documentos pertenecientes a las clases que muestran cierta similitud con la consulta, se compararán con la consulta, según la fórmula (2.2), y se recuperarán aquellos documentos que resulten similares a la consulta.

Así, si existen n documentos en la colección que son clasificados en x clases, cada una de ellas aproximadamente con n/x documentos, entonces el número de comparaciones entre vectores se reducirá a $x + n/x$, en vez de las n comparaciones originales.

2.2.1.3 Extracción y Pesado Automático de términos

La construcción de los vectores asociados a cada documento se realiza durante el proceso de *indexado* de la colección de documentos. Dicha tarea consistirá en dos etapas, primero se determinan los términos representativos del contenido de un documento, y, segundo, se asigna a cada término un peso o valor que refleje su importancia como representante del contenido del documento.

La primera etapa es relativamente sencilla, se basa en la extracción de los términos que componen el texto de los documentos, pudiéndose considerar también el título, el resumen o cualquier otra fuente de información asociada al documento. La segunda etapa, la asignación de pesos a esos términos, será una tarea que necesita un análisis más profundo.

La mayoría de los intentos de indexación automática se basan en la idea de que la frecuencia de ocurrencia de un término en un documento tiene alguna relación con la importancia de ese término como representante del contenido del documento. Si ordenamos las distintas palabras de un documento en orden decreciente de frecuencia de aparición, la ocurrencia del vocabulario puede ser caracterizada por una constante z tal y como enuncia la ley de *Zipf* en [Zipf, 1949]:

$$frecuencia \cdot orden \approx z \tag{2.4}$$

Es decir, se cumple que la frecuencia de una palabra multiplicada por su puesto en el orden será aproximadamente igual a la frecuencia de cualquier otra palabra multiplicada por el suyo correspondiente.

Utilizando esta ley de *Zipf* se podrá obtener el factor de relevancia de un término, basándonos en las frecuencias de las palabras de la colección de documentos, siguiendo los siguientes pasos:

1. En una colección de n documentos, se calcula la frecuencia de cada término t_j en cada documento d_i: tf_{ij}.
2. Se determina la frecuencia de cada término t_j respecto a la colección completa, sumando sus frecuencias en los n documentos:

$$tot\_tf_j = \sum_{i=1}^{n} tf_{ij}$$

3. Se ordenan las palabras en orden decreciente de tot_tf_j y se eliminan aquellas que tengan un valor superior a un umbral dado, para excluir las palabras muy frecuentes.
4. Del mismo modo, se eliminan las palabras poco frecuentes.
5. Las palabras restantes, con una frecuencia media, se utilizarán para caracterizar los documentos indexados.

Para justificar estos pasos nos basamos en la conjetura del *poder de resolución*, que establece que el poder de resolución es máximo en el rango medio de frecuencias de aparición de las palabras, tal y como puede observarse en la figura 2.5. El poder de resolución será la habilidad de los términos de indexación para convertirse en ítems relevantes [Vegas, 1999].

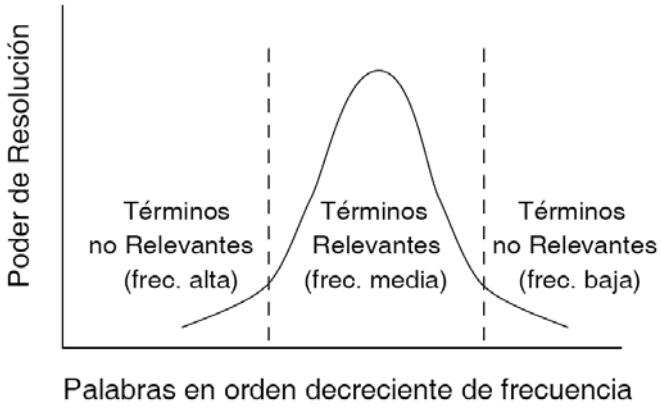


Figura 2.5. Gráfico del *poder de resolución* de los términos de un documento. Fuente: [Vegas, 1999].

Sin embargo, la eliminación de todas las palabras muy frecuentes puede producir pérdida en la *exhaustividad*, mientras que la eliminación de las palabras poco frecuentes puede ocasionar pérdidas en la *precisión*. Además, será necesario elegir los umbrales correctos que determinen un buen conjunto de palabras de frecuencia media. Todo esto nos conduce a reconsiderar la utilización de las frecuencias de aparición en modo absoluto y su sustitución por frecuencias relativas, mediante diversas estrategias:

La Frecuencia de Documento Inversa. Consiste en asumir que la importancia del término es proporcional a la frecuencia de ocurrencia de cada término t_j en cada documento d_i, tf_{ij}, e inversamente proporcional al número de documentos en los que se encuentra ese término, df_i. De esta manera, se puede considerar la medida del peso del término t_j en el documento d_i como:

$$w_{ij} = tf_{ij} / df_i \tag{2.5}$$

El Valor de Discriminación. Esta medida pretende cuantificar el grado en el que el uso de un término va a ayudar a distinguir un documento de otro. Dada una colección de documentos, y dos documentos d_i y d_j, podemos utilizar una medida de similitud, $sim(d_i, d_j)$ para representar la similitud entre esos documentos. Las funciones típicas de similitud generan valores entre 0, para documentos sin similitud, y 1 para documentos completamente iguales.

Obteniendo la similitud para todos los pares de documentos d_i y dj, con $i \neq j$, se puede calcular una similitud media para la colección:

$$\overline{sim} = c \sum_{i=1}^{n} \sum_{j=1}^{n} sim(d_i, d_j) \qquad \text{con } i \neq j \tag{2.6}$$

donde c es una constante, por ejemplo $1/n(n - 1)$. La fórmula (2.6) representa una medida de la densidad del espacio de documentos, el grado en que los documentos se agrupan en el espacio de documentos. Así, si todos los documentos fuesen iguales, $\overline{sim}$ tendría el valor $c \cdot n(n - 1) = 1$.

Para calcular de manera más eficiente la densidad del espacio de documentos se puede obtener un documento medio, $\bar{d}$, como centroide, cuyos términos se supone que poseen características de frecuencia media. Entonces, la frecuencia media del término t_j se definirá como:

$$\overline{tf_j} = \frac{1}{n}\sum_{i=1}^{n} tf_{ij} \tag{2.7}$$

En este punto, se calculará la densidad del espacio de documentos como la suma de las similitudes de cada documento con respecto al centroide, con la siguiente fórmula, menos costosa que la (2.6):

$$\overline{sim} = c\sum_{i=1}^{n} sim(\bar{d}, d_i) \tag{2.8}$$

Consideramos ahora el caso en el que se haya eliminado el término t_j de todos los documentos de la colección original. Sea $\overline{sim_j}$ la densidad del espacio de documentos en este caso. Si el término t_j fuera un término con alta frecuencia de aparición y con una distribución de frecuencias prácticamente constante, significaría que aparece en casi todos los documentos, entonces su eliminación reducirá la similitud media entre pares de documentos. Esta situación resulta desfavorable, ya que cuando un término como éste se asigne a los documentos, se incrementará la media de la similitud, comprimiendo el espacio de documentos. Por otra parte, si un término t_j hubiese obtenido un peso alto en unos documentos, pero no en otros, su eliminación producirá un incremento de similitud entre documentos.

Se puede calcular el valor de discriminación de un término t_j, dv_j como:

$$dv_j = \overline{sim}_j - \overline{sim} \tag{2.9}$$

Cuando se haya calculado el valor $\overline{sim_j}$ para todos los términos t_j, éstos podrán ordenarse en orden decreciente según su valor de discriminación. Entonces los que

aparezcan en el principio de la lista serán muy específicos, mientras que los del final de la lista serán muy comunes. De esta manera, los términos de indexación se pueden clasificar en tres categorías según su valor de discriminación:

- Buenos discriminadores: con un valor dv_j positivo, que al ser considerados en la indexación decrementan la densidad del espacio.
- Discriminadores neutros: con un valor dv_j cercano a cero y cuya eliminación o adición no varía la similitud entre documentos.
- Malos discriminadores: con un valor dv_j negativo, que hacen más similares a los documentos.

Mediante el cálculo del valor de discriminación obtenemos un método objetivo para determinar el umbral de frecuencia, así los términos con alta frecuencia y un valor de discriminación negativo serán pobres y no deberán utilizarse en la indexación. Los términos con baja frecuencia y un valor de discriminación cero pueden o no ser utilizados, su consideración no afectará a las prestaciones del sistema de recuperación, aunque si puede afectar a la eficiencia del sistema que deberá almacenar y manipular gran cantidad de términos poco frecuentes. Por último, los términos que son buenos discriminadores, con poder de resolución, tendrán un valor de discriminación positivo y deberán considerarse en la indexación, coincidiendo con los de frecuencia intermedia.

Ahora podemos definir una medida del peso de un término que tenga en cuenta la frecuencia relativa de aparición del mismo, combinando dicha frecuencia con el valor de discriminación:

$$w_{ij} = tf_{ij} \cdot dv_j \tag{2.10}$$

2.2.2. El Modelo Probabilístico

Este modelo se apoyará en la teoría de la probabilidad para construir y determinar el uso de una función de búsqueda capaz de diferenciar un documento relevante de otro que no lo sea [Rijsbergen, 1979]. Para componer esta función de búsqueda se examinará la distribución de los términos de indexación a lo largo de la colección de documentos o de

un subconjunto de ella. A la función de búsqueda se le podrá aplicar realimentación de la relevancia para automatizar el ajuste del valor de sus parámetros.

La función de búsqueda estará compuesta por una serie de pesos asociados a los términos de indexación, tal y como se introdujo en la sección dedicada al modelo vectorial. La diferencia entre ambos modelos reside en la forma de calcular el peso de los términos en la consulta. Así, en el modelo probabilístico, los pesos de los términos que aparezcan en los documentos relevantes de una consulta previa deberán incrementarse frente a los pesos de los términos que no aparezcan. Este cálculo se basará en los valores de la tabla 2.3, llamada *de contingencias*, que muestra la distribución del término *t* en los documentos relevantes y no relevantes para una consulta *q*, en donde *N* será el número total de documentos en la colección, *R* será el número de documentos relevantes para la consulta *q*, *n* será el número de documentos que incluyen el término *t* y *r* será el número de documentos relevantes que incluyen el término *t*. El contenido de la última fila y de la última columna será el resultado de sumar las filas y columnas correspondientes.

	doc relevantes	**doc no relevantes**	
t ∈ doc	r	$n - r$	n
t ∉ doc	$R - r$	$N - n - R + r$	$N - n$
	R	$N - R$	N

Tabla 2.3. Tabla de contingencias que muestra la distribución del término t en los documentos relevantes y no relevantes para una consulta q en el modelo probabilístico [Rijsbergen, 1979].

Apoyándose en esta tabla de contingencias, *Robertson* [Robertson, 1976] y *Sparck Jones* [Sparck, 1975, 1979] derivaron varias fórmulas para calcular el peso de un término basándose en los resultados de una consulta previa:

$$w_1(t) = \log \frac{(\frac{r}{R})}{(\frac{n}{N})} \tag{2.11}$$

$$w_2(t) = \log \frac{(\frac{r}{R})}{(\frac{n-r}{N-R})} \tag{2.12}$$

$$w_3(t) = \log \frac{(\frac{r}{R-r})}{(\frac{n}{N-n})} \tag{2.13}$$

$$w_4(t) = \log \frac{(\frac{r}{R-r})}{(\frac{n-r}{N-n-R+r})} \tag{2.14}$$

Estas cuatro fórmulas fueron estudiadas y probadas por diferentes autores, destacando los trabajos de *Sparck Jones* [Sparck, 1975, 1979], que las utilizó en una serie de experimentos sobre la *colección Cranfield*[1] indexada manualmente. La fórmula (2.14) proporcionó los mejores resultados, seguida de cerca por la fórmula (2.13).

2.3. La Web como sistema de recuperación de información

Berners-Lee [Berners, 1989] quiso desarrollar un método eficiente y rápido para intercambiar datos científicos combinando dos tecnologías existentes en 1991, el hipertexto y el protocolo de comunicaciones TCP/IP. Implantó un nuevo modelo de acceso a la información en Internet: la "World Wide Web", WWW, o la Web. Su objetivo básico era evitar la pérdida de información inherente a una gran organización así como facilitar el acceso a la información disponible. Dos características fundamentales de la propuesta han convertido a la Web en lo que es en la actualidad: su naturaleza distribuida y la posibilidad de establecer vínculos entre los documentos.

La propuesta original de Berners-Lee insistía en la necesidad de hacer el sistema suficientemente atractivo para animar a los usuarios a incorporar información al mismo, de tal forma que su utilidad creciese al añadirse nuevos documentos y esa utilidad creciente impulsase, a su vez, a seguir aumentando la base de documentos. "Un sistema con enlaces permitiría a los usuarios navegar a través de conceptos, documentos, sistemas y autores, permitiendo, asimismo, almacenar referencias entre documentos".

Se diseñó un sistema para crecer de un modo cada vez más acelerado sin incluir ningún tipo de mecanismo capaz de facilitar la localización de un documento en particular. No obstante, sería un error interpretar esto como una crítica hacia la forma en que se

[1] *Consiste en 1398 documentos sobre distintos aspectos de ingeniería aeronáutica y 225 preguntas para las que se conocen los juicios de relevancia [López, 2002].*

implementó finalmente la Web, esta decisión de diseño facilitó su desarrollo y posterior crecimiento, y desde la puesta en marcha del primer servidor Web aún transcurrieron tres años hasta que la necesidad de un sistema de búsqueda de información para la Web se hiciera apremiante.

Así, la Web es un nuevo contexto, con particularidades muy definidas, por lo que se precisará una adaptación del concepto de recuperación de información. Delgado Domínguez [Delgado, 1998] afirma que "se puede definir el objetivo de la recuperación como la identificación de una o más referencias de páginas web que resulten relevantes para satisfacer una necesidad de información". En este caso, los SRI que se empleen en la Web nos devolverán referencias a los documentos, en lugar de los propios documentos.

2.3.1. Métodos de recuperación de información en la Web

Las técnicas de RI que se utilizan en la Web proceden de las empleadas en los SRI tradicionales. Sin embargo, tanto el entorno de trabajo como las características de los datos almacenados son diferentes. Así, pueden surgir serios problemas al realizar operaciones de recuperación de información en la Web.

La Web "posee unas características, desde el punto de vista documental, que la configuran como un entorno singular y diferente de los clásicos. Algunas de estas características son las siguientes" [Delgado, 2001]:

- Gran tamaño de la base de datos documental: a septiembre de 2005 existen más de 8.000 millones de páginas web indizadas por el buscador *Google*.
- Heterogeneidad de las publicaciones en cuanto a:
  - Tipos de documentos, los artículos científicos coexisten con páginas personales y comerciales.
  - Tipos de datos, las páginas web pueden contener texto simple y elementos multimedia. Además admiten muchos formatos.
  - Estructura interna de las páginas, la mayoría están codificadas en *HTML*[2], y aunque existen unas especificaciones de dicho lenguaje publicadas por el

[2] *HTML es un lenguaje sencillo que controla la presentación y el comportamiento de documentos web. Para más información consultar la sección AI.1 del Anexo I.*

W3C[3], los autores de las páginas no suelen ser muy estrictos debido a que los navegadores son muy permisivos respecto a la sintaxis de los documentos. Esto dificulta su lectura e indización mediante un programa informático.

- Estructura externa, en muchas páginas no se puede identificar quién es el autor o su fecha de publicación, datos muy importantes en las referencias bibliográficas.
- Calidad, publicar en la Web es gratuito en muchos servidores, es fácil e instantáneo, esto conduce a que muchos páginas no tengan ninguna calidad científica, que puedan contener afirmaciones falsas o inventadas y errores tipográficos.
- Diseño hipertextual, una página web se identifica con un nodo de la estructura hipertextual de la Web. Puede coincidir con las partes clásicas de los documentos escritos: capítulos, secciones o párrafos, con la porción de texto que cabe en la pantalla sin realizar desplazamientos, con documentos completos, con el desarrollo de una idea. Un documento puede contener una o más páginas web, y por otra parte, una página web puede contener resúmenes o extractos de varios documentos.

- Audiencia, es muy fácil hacer que un documento esté accesible al mismo tiempo para cualquiera de los millones de internautas.
- Dinamismo y volatilidad, muchas páginas web se generan en tiempo real como resultado de consultas realizadas en buscadores, y su vida puede reducirse al tiempo de visualización del usuario; otras páginas cambian de URL[4], o incluso cambian totalmente de contenido manteniendo la misma URL.
- Invisibilidad, no todas las páginas web resultan susceptibles de ser encontradas como, por ejemplo, aquéllas que por deseo del autor no son indizadas, aquéllas que por estar en niveles muy profundos de la jerarquía de directorios de un servidor

[3] *W3C es un consorcio que desarrolla tecnologías inter-operativas (especificaciones, líneas maestras, software y herramientas) para guiar la Web a su potencialidad máxima a modo de foro de información, comercio, comunicación y conocimiento colectivo.*
[4] *URL es el acrónimo de "Uniform Resources Locator", o localizador uniforme de recursos, que permite localizar o acceder de forma sencilla a cualquier recurso de la Red.*

web no suelen ser tenidas en cuenta por un *robot*[5], aquéllas que sólo son accesibles mediante contraseña, o aquéllas que no son enlazadas por ninguna otra.

"En conclusión podríamos decir que el crecimiento explosivo de la Web, unido a la diversidad de información que contiene, su diversa procedencia y la anarquía de su organización, dificultan enormemente el hallazgo de información útil para un usuario determinado, más aún cuando es el propio usuario quien efectúa sus propias búsquedas" [Delgado, 2001].

2.3.1.1. Herramientas de búsqueda en la Web

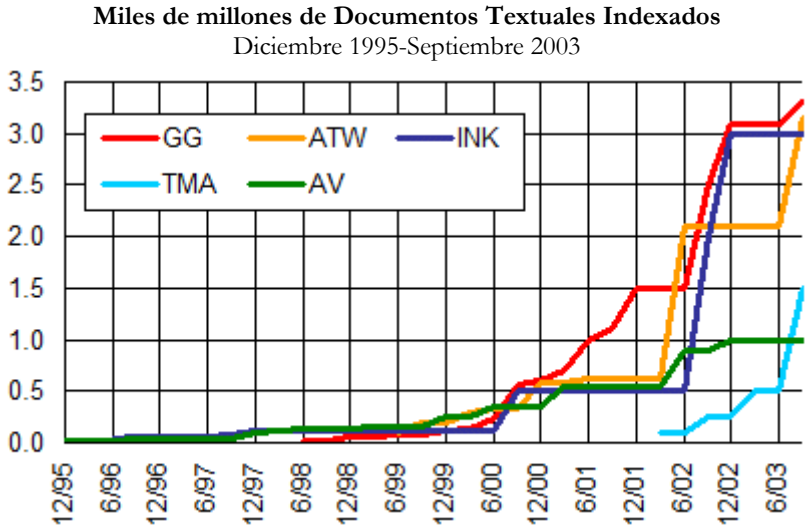
Según Baeza-Yates se pueden considerar tres maneras de buscar información en la Web: "la primera de ellas es utilizar los motores de búsqueda, que indexan una porción de los documentos existentes en la globalidad de la Web y permiten localizar información mediante la formulación de una pregunta. La segunda es utilizar directorios, sistemas que clasifican documentos Web seleccionados por materias y que nos permiten navegar por sus secciones o buscar en sus índices. La tercera es buscar en la Web mediante la explotación de su estructura hipertextual" [Baeza, 1999].

- **Motores de Búsqueda o Buscadores**

Los buscadores utilizan *robots* para rastrear la estructura hipertextual de la Web y localizar los recursos que incluirán automáticamente en su base de datos. Cada *robot* rastrea a su manera en la Web, de ahí que la información almacenada en cada base de datos sea diferente. Generalmente parten de una lista determinada y a partir de ahí, realizan un rastreo recursivo de los documentos que se referencian [Delgado, 2001].

Se puede observar el tamaño de la base de datos de los principales buscadores y su evolución en el gráfico de la figura 2.6 obtenido de *Searchenginewatch*[6]:

[5] *Un robot de la Web es un programa que recorre automáticamente la estructura de hipertexto de la Web buscando un documento y devuelve recursivamente los documentos a los que éste hace referencia, aplicándole a éstos el mismo proceso.*
[6] *http://searchenginewatch.com*



GG=Google, INK=Inktomi, AV=AltaVista, ATW=AllTheWeb, TMA=Teoma.

Figura 2.6. Comparación de la cantidad de documentos indexados por los buscadores más representativos desde el año 1995 hasta el año 2003. Fuente: *http://searchenginewatch.com/reports/article.php/2156481*, en línea.

Para utilizar un buscador, el usuario expresará su necesidad de información mediante un formulario. Este puede consistir desde una simple caja donde teclear las palabras clave hasta una búsqueda avanzada con multitud de opciones para expresar con un mayor detalle aquello que desea buscar. Las búsquedas avanzadas suelen ofrecer la posibilidad de utilizar operadores booleanos, de adyacencia, de existencia, de exactitud y, a veces, también se puede delimitar la búsqueda por fechas, por ciertas etiquetas de HTML, por tipo de fuente, por área geográfica o dominio y por idioma.

Los resultados de la búsqueda se mostrarán al usuario ordenados según algún criterio de relevancia. La ordenación suele calcularse según alguna función de similitud de la pregunta con respecto a los documentos, o en función de la popularidad de las páginas.

Una de las ventajas de los buscadores es que son muy exhaustivos gracias a que sus procesos de recogida de recursos y de indización son automáticos, sin embargo, estos recursos indexados automáticamente no pasan por ningún proceso de selección de calidad por lo que podemos encontrarnos con muchos resultados poco útiles.

✦ Directorios

Atendiendo a [Delgado, 2001], en los directorios la información está organizada en una estructura jerárquica, atendiendo a algún criterio de clasificación en categorías. Se pueden

utilizar esquemas de clasificación universalmente difundidos, como por ejemplo el "Dewey Decimal Classification" (DDC), el "Universal Decimal Classification" (UDC) o el "Library of Congress Classification" (LCC), aunque generalmente se aplican esquemas propios, y, en algunos casos, la clasificación se realiza de forma automática. Un esquema de clasificación estándar aportará ventajas para los profesionales de la búsqueda de información, y también para los usuarios asiduos de bibliotecas, familiarizados con tales esquemas.

En la recogida y selección de recursos se aplican criterios de pertinencia y calidad formal y de contenido para evaluar si un recurso merece ser incluido o no en el directorio. Además, se suele permitir que los usuarios remitan una URL para ser evaluada.

Los directorios se explorarán mediante navegación, es decir, los usuarios recorren la estructura ramificada para buscar la información que necesitan. De esta manera el usuario puede descender por distintos niveles de especificidad hasta encontrar la información adecuada a sus intereses sin necesidad de formular explícitamente su consulta.

Los directorios suelen ser más fáciles de utilizar que los buscadores, sólo hay que elegir la categoría que se ajuste a nuestro propósito, su contenido se puede examinar globalmente, podemos cambiar la especifidad de la búsqueda bajando o subiendo en la estructura del directorio, y los documentos hallados estarán en el contexto de la categoría en que se realiza la búsqueda. Sin embargo, cubren solo una pequeña parte de los recursos existentes en la Web y adolecen de una falta de criterios homogéneos para la selección y clasificación de los documentos.

- **Multibuscadores**

Para [Baeza, 1999] los multibuscadores son servidores Web que envían una pregunta dada a varios motores de búsqueda, directorios Web y otras bases de datos, entonces recolectan las respuestas y las unifican para mostrarlas al usuario. Ejemplos son *Metacrawler* [Selberg, 1995] y *SavvySearch* [Howe, 1997].

Según [Delgado, 2001] "los multibuscadores o metabuscadores proporcionan la posibilidad de buscar a través de un número determinado de herramientas de búsqueda de forma simultánea. No utilizan *robots* para recoger o mantener unas bases de datos propias individuales sino que utilizan las bases de datos de los buscadores o directorios sobre los que lanzan las peticiones de los usuarios. Existen multibuscadores que presentan los resultados de forma concatenada, es decir para cada motor interrogado se presenta una lista

de los resultados obtenidos; y otros que permiten obtener los resultados de forma integrada, eliminando los duplicados e indicando para cada resultado qué buscador o buscadores lo han proporcionado".

- **Búsquedas aprovechando la estructura hipertextual de la Web**

Para [Baeza, 1999], otras formas de búsqueda en la Web pueden llevarse a cabo utilizando lenguajes específicos para interrogar a la Web o "Web Query Languages", mediante Búsqueda Dinámica y empleando Agentes de Software.

La idea de los "Web Query Languages" es incluir en la pregunta la estructura de enlaces de las páginas Web, y no solamente el contenido de cada página. Por ejemplo, podríamos querer una búsqueda de todas las páginas Web que contengan al menos una imagen y que sean alcanzables desde un sitio siguiendo como mucho tres enlaces. Para posibilitar este tipo de búsqueda, se necesitarán diferentes modelos de datos, el más importante será un modelo de grafo etiquetado para representar las páginas Web (nodos) y los hiperenlaces (aristas) entre páginas, y un modelo de datos semi-estructurado para representar el contenido de las páginas Web. Lenguajes de este tipo son STRUQL [Fernández, 1997], FLORID [Himmeroder, 1997] y WebOQL [Arocena, 1998].

La Búsqueda Dinámica en la Web será equivalente a la búsqueda secuencial de texto. La idea es descubrir información relevante siguiendo los enlaces de las páginas. La principal ventaja es que se busca en la estructura actual de la Web y no en la almacenada en el índice de un buscador. Esta aproximación será lenta para toda la Web, pero podrá utilizarse en pequeños subconjuntos dinámicos de la Web. La primera heurística diseñada para esta función fue "fish search" [De Bra, 1994], que saca provecho de la intuición de que los documentos relevantes suelen tener como "vecinos" documentos relevantes. Así, la búsqueda seguirá los enlaces de los documentos relevantes. Esta heurística se mejoró con "shark search" [Hersovici, 1998], que realiza una mejor valoración de la relevancia de las páginas "vecinas".

Otros trabajos incluyen los Agentes de Software para buscar información específica en la Web [Ngu, 1997], [LaMacchia, 1997]. Esto implica el tratamiento con diversas fuentes heterogéneas de información que tienen que ser combinadas. Temas importantes a tener en cuenta serán cómo se determinan las fuentes relevantes y cómo se combinan los resultados recuperados [Baeza, 1999].

2.3.2. Navegando por la información de la Web

Los documentos hipertextuales de la Web pueden ofrecer información en forma de texto, sonido, imágenes, animaciones, vídeos y otras formas. A la operación de explorar en la Web para encontrar dicha información se le denomina genéricamente *navegar por la Web*. Existen diversas maneras de navegar por la información de la Web, la más común es utilizando programas navegadores. También será posible navegar en ésta a través de otros programas tales como los agregadores de contenidos. A continuación se comentarán las principales características de estos programas.

Navegadores

Un navegador web o "web browser" es una aplicación *software* que permite al usuario recuperar y visualizar documentos de hipertexto[7], comúnmente descritos en HTML, a través de Internet . Esta red de documentos es denominada "World Wide Web" o Telaraña Mundial. Los navegadores actuales permiten mostrar y/o ejecutar: gráficos, secuencias de vídeo, sonido, animaciones y programas diversos además del texto y los hipervínculos o enlaces.

La funcionalidad básica de un navegador web es permitir la visualización de documentos de texto, posiblemente con recursos multimedia incrustados. Tales documentos, comúnmente denominados páginas web, pueden poseer hipervínculos que enlazan una porción de texto o una imagen a otro documento, normalmente relacionado con el texto o la imagen. El seguimiento de enlaces de una página a otra, ubicada en cualquier ordenador conectado a Internet, se llama *navegación*.

El primer navegador, desarrollado en el CERN[8] a finales de 1990 y principios de 1991 por Tim Berners-Lee, era bastante sofisticado y gráfico, pero sólo funcionaba en determinados equipos de trabajo.

El navegador **Mosaic**, fue el primero que se extendió preparándose versiones para distintos sistemas operativos. Sin embargo, poco más tarde el navegador **Netscape Navigator** superó rápidamente a Mosaic en capacidad y velocidad.

[7] *Un hipertexto es un documento digital que se puede leer de manera no secuencial.*
[8] *La sigla CERN viene de su antiguo nombre: Centro Europeo para la Investigación Nuclear (Centre Européen pour la Recherche Nucléaire, en francés). Se trata de un laboratorio de investigación en física de partículas.*

Internet Explorer fue la apuesta de la empresa Microsoft para el mercado de los navegadores que finalmente consiguió desbancar a Netscape Navigator. En los últimos años se ha vivido una auténtica explosión del número de navegadores, y éstos ofrecen cada vez mayor integración con el entorno de ventanas en el que se ejecutan. "Netscape Communications Corporation" liberó el código fuente de su navegador, naciendo así el proyecto **Mozilla**.

A finales de 2004 aparece en el mercado **Firefox**, una rama de desarrollo de Mozilla que pretende hacerse con parte del mercado de Internet Explorer. Se trata de un navegador más ligero que su hermano mayor.

Agregadores de contenidos

Son un producto reciente en la Web, su función es aglutinar información de distintas páginas web que distribuyen los contenidos en lenguajes específicos, como por ejemplo **RSS**[9] o **Atom**[10], chequeando además la actualidad de esas fuentes de información. De esta manera, un agregador será un sistema que recupera información procedente de diversas fuentes de la Web, de forma que no sea necesario visitar las páginas en cuestión para obtener sus contenidos, centralizando así la información en un único lugar de consulta.

Existe una extensa lista de programas agregadores [RSS, 2005], [RSSfeeds, 2005], [Goo, 2005], la mayoría de ellos tienen un aspecto y funcionamiento muy parecido. Por una parte, permitirán subscribirse a las diferentes fuentes de información que resulten de interés para el usuario, y por otra, comprobarán periódicamente los contenidos ofrecidos en esas fuentes seleccionadas, para detectar si se han actualizado, en cuyo caso suelen presentar algún mensaje informativo al usuario acerca de la nueva información disponible. Ofrecerán aglutinada toda la información recuperada de las diversas fuentes a las que esté subscrito el usuario, evitando de esa manera la consulta individual de cada una de ellas. Un ejemplo de presentación de los contenidos recuperados por un agregador popular puede verse en la figura 2.8.

[9] *RSS es acrónimo de "Really Simple Syndication" o Sindicación Realmente Simple [Winer, 2005]. Para más información acerca de este lenguaje consultar el apartado AI.3 del Anexo I.*

[10] *Atom es otra tecnología para distribuir y actualizar contenidos. Para más información acerca de este lenguaje consultar el apartado AI.4 del Anexo I.*

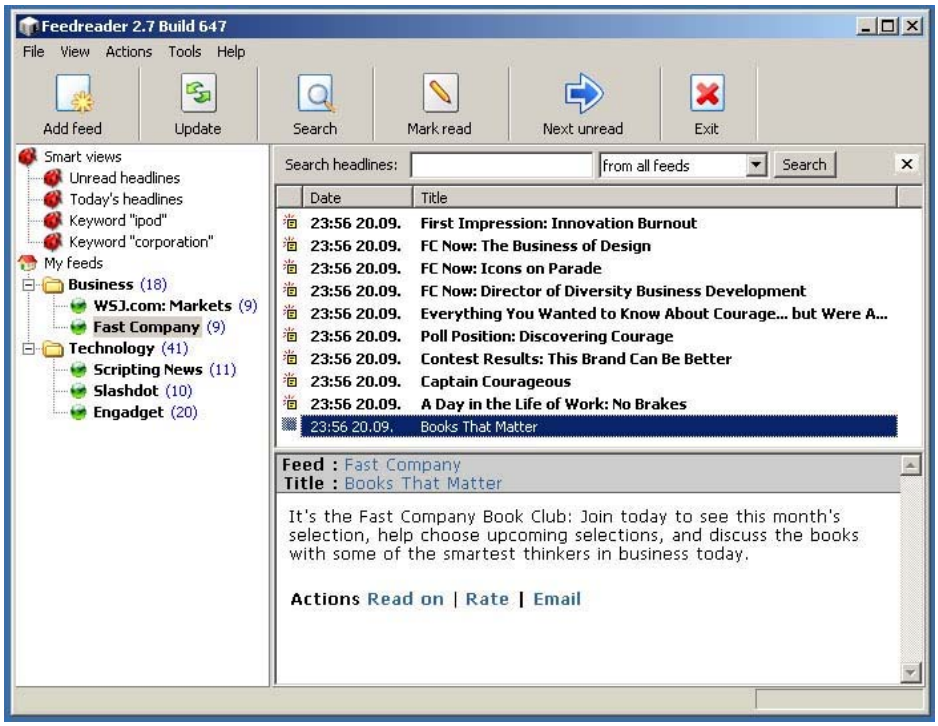


Figura 2.8. Aspecto típico de un agregador de contenidos. Fuente: http://feedreader.com/

Existen también agregadores en línea, como el proporcionado por *Feedster*[11], que proporcionan al usuario una serie de herramientas para agregar y modificar fuentes de información, con múltiples opciones de personalización.

Debido al auge de estos formatos de información, el número de fuentes disponibles en la Web se ha multiplicado rápidamente, sólo en *Feedster* [Feedster, 2005], a septiembre de 2005, se encuentran indexadas más de 10 millones de ellas. Un usuario típico puede desear subscribirse a cientos de estas fuentes, así que, aunque los agregadores típicos solucionan parcialmente el problema automatizando las consultas y aglutinando todos los contenidos recientes en un mismo lugar, este usuario puede llegar a sobrecargarse de información. De esta manera, normalmente el usuario seleccionará algunos contenidos que le resulten interesantes, dejando de escoger más información cuando su demanda se vea satisfecha o cuando se encuentre cansado de buscar sin llegar a cubrir su demanda informativa. Por ello, en muchos casos resultará interesante disponer de un mecanismo automático de selección de contenidos, por el cual se le recomiende al usuario aquella información que el sistema puntúe como interesante en base a sus intereses particulares.

[11] *http://my.feedster.com/login.php*

Nuestro enfoque en la tesis está encaminado en este sentido, el de un agregador inteligente de contenidos que ordene la información recuperada al usuario según sus intereses. Para ello, se necesitará algún tipo de marcaje sintáctico que indique la relevancia de diferentes partes del texto, por ejemplo el título y el resumen del contenido, características que poseen lenguajes del tipo RSS o Atom.

2.3.3. Sistemas de recomendación

En Internet existe una gran cantidad de sitios especializados que ofertan millones de productos y servicios para su consumo. Éste hecho puede resultar un importante inconveniente cuando se desea realizar una adquisición eligiendo entre todas las opciones existentes. Los sistemas de recomendación surgen como solución a este problema, así "un sistema de recomendación recibe información del usuario acerca de productos y/o servicios en los que el usuario se encuentra interesado y le recomienda aquéllos cercanos a sus necesidades" [García, 2002]. "La recomendación puede entenderse también como un proceso de filtrado en el que se deja pasar por el filtro únicamente los contenidos relevantes para cada usuario en concreto" [Serradilla, 2005].

Los sistemas de recomendación han evolucionado rápidamente dentro del entorno interactivo de la Web, especialmente en el sector del comercio electrónico, donde pueden albergarse inmensas bases de datos con productos ofreciendo soporte y atención a gran cantidad de usuarios, cada uno de ellos con un perfil determinado. En este sentido, Schafer et al. [Schafer, 2001] considera una taxonomía de sistemas de recomendación basada en tres categorías atendiendo a las funcionalidades de entradas y salidas, a los métodos de recomendación y al resto de aspectos del diseño.

García y Gil [García, 2002] describen un sistema de recomendación basado en agentes adaptativos que integra la personalización de las recomendaciones al usuario a la vez que la estrategia comercial del sitio web. El sistema de recomendación implementa una arquitectura propia de comercio electrónico denominada *e-CoUSAL* [García et al., 2002].

Un ejemplo de sistema de recomendación es el proyecto SIRLE [SIRLE, 2003], que recomienda lecturas de libros en español basándose en la correlación entre los perfiles de los usuarios, es decir, busca similitudes entre las preferencias de distintos usuarios. Los usuarios se representan como vectores en los que cada componente contendrá la valoración de un objeto particular por parte de dicho usuario. Según [Serradilla, 2005] este

proceso responde a la natural tendencia humana de recomendación de objetos entre amigos.

En [Merelo et al., 2004] se propone un sistema para recomendar a los lectores de un *weblog* otros *weblogs*[12] con temas relacionados, partiendo del resultado de una encuesta, empleando para ello reglas de asociación. Lo que se intenta es buscar condiciones del tipo atributo-valor que ocurren frecuentemente en un conjunto de datos. El sistema considera un conjunto de atributos compuestos por las URLs de los *weblogs* y una base de datos de encuestas donde se indicará si un usuario ha leído o no cada *weblog*.

En [Mizzaro, 2002] se emplean técnicas de personalización para implementar sistemas de acceso a publicaciones electrónicas. Para ello, distinguen entre personalización persistente y personalización efímera, describiendo cómo ambas pueden aplicarse en el filtrado de información y en sistemas de recuperación, a través de un portal Web especializado.

Para ayudar a los usuarios a encontrar documentos en la Web que sean relevantes a sus necesidades particulares, [Chaffee, 2000] considera una vista del mundo para cada usuario. Crea un perfil de usuario analizando las páginas Web que éste visita, y así puede suministrar la información clasificada individualmente, proporcionando un orden personalizado de conceptos para navegar por la Web. El sistema se construye utilizando las características de un sitio particular creado mediante el sistema denominado OBIWAN [OBIWAN, 1999], que permite a los usuarios explorar múltiples sitios utilizando la misma jerarquía de navegación. Un ejemplo de este sistema puede verse en la figura 2.7.

[Middleton, 2001] presenta un sistema de recomendación denominado *Quickstep* para encontrar artículos científicos y de investigación. Para adquirir las preferencias del usuario se monitoriza su comportamiento al navegar por la Web, empleando técnicas de aprendizaje automático asociadas a una representación ontológica.

Esta tesis también tiene un enfoque como sistema de recomendación. En este sentido, se monitorizarán las acciones del usuario para adquirir sus preferencias, se clasificará la información recuperada y se le ofrecerá ordenada. Sin embargo, el análisis del comportamiento del usuario al navegar por la Web se restringirá al conjunto de información recomendado por el sistema.

[12] *Los" weblogs" son sitios web, que suelen actualizarse varias veces al día, en los que uno o varios autores publican sus opiniones sobre temas de actualidad.*

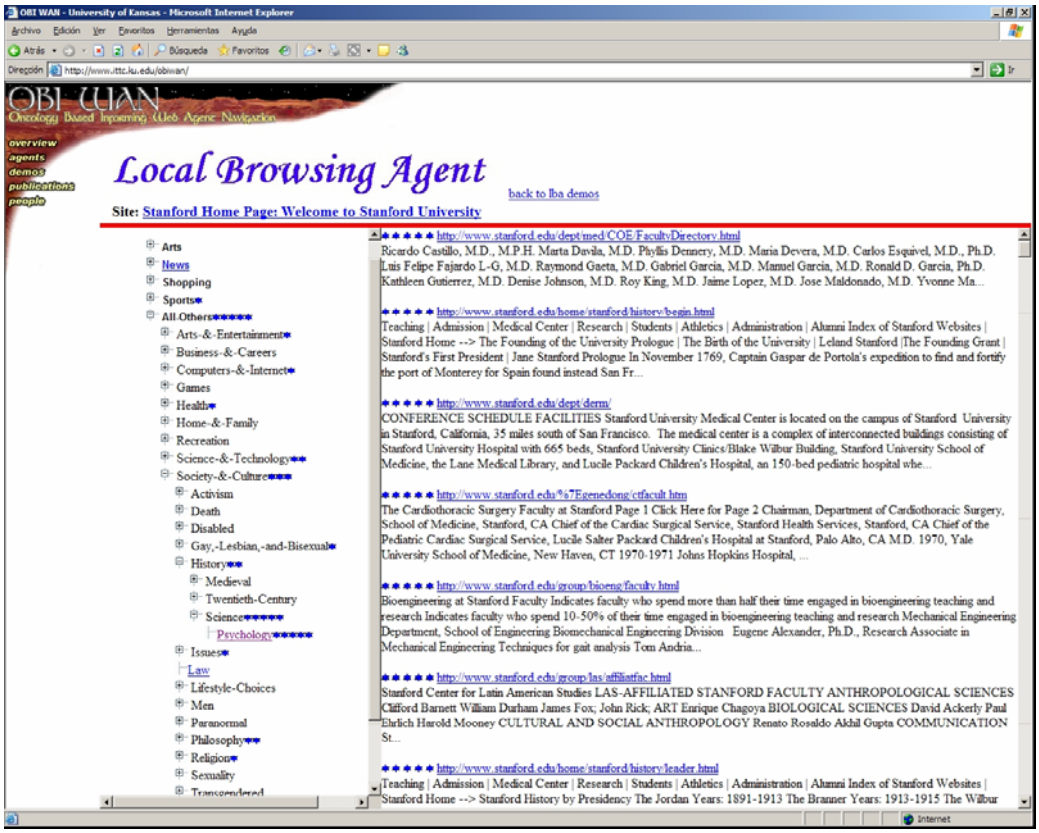


Figura 2.7. Ejemplo del sistema OBIWAN [OBIWAN, 1999] utilizado por [Chaffee, 2000]. Fuente: http://www.ittc.ku.edu/obiwan/

2.4. Resumen

En este capítulo se han visto varias definiciones del concepto de "recuperación de información" y de los sistemas de recuperación de información.

Se han expuesto varias propuestas de clasificación de los modelos para la recuperación de la información, para posteriormente analizar en detalle el modelo vectorial y el modelo probabilístico. El modelo vectorial hace la suposición básica de que la proximidad relativa entre dos vectores es proporcional a la distancia semántica de los documentos. Dentro de este modelo, se han analizado diferentes fórmulas para medir la similitud entre documentos y consultas, destacando la medida de similitud del coseno, ampliamente utilizada.

Se ha abordado también la realimentación de la relevancia por parte de un usuario para mejorar los resultados de las consultas, y la agrupación o "clustering" de documentos para organizar a éstos en clases, que puede realizarse aplicando medidas de similitud entre pares de documentos.

Para construir los vectores asociados a los documentos se necesita un proceso de indexado de éstos, extrayendo los términos que los componen y asignando pesos a esos términos. Así, para obtener la relevancia de un término se puede hacer uso de la ley de *Zipf*. Se exponen también estrategias para sustituir las frecuencias absolutas de los términos en un documento por frecuencias relativas como la frecuencia de documento inversa o el valor de discriminación.

El modelo probabilístico se diferencia principalmente en la forma de calcular los pesos de los términos en los documentos y en las consultas, que en este caso se basa en los valores de una tabla de contingencias.

Se ha dedicado también bastante atención a la Web como sistema de recuperación de información, diferenciando sus características singulares que nos obligan a considerar métodos de recuperación de información alternativos. Algunas herramientas de búsqueda de información en la Web son los buscadores, los directorios y los multibuscadores. Otros sistemas de búsqueda en la Web intentan aprovechar su estructura hipertextual, empleando lenguajes específicos, búsqueda dinámica o agentes de software.

Por otra parte, debido a la gran cantidad de información y de objetos de consumo disponibles en la Web, aparecen sistemas de recomendación que se encargan de filtrar la información recuperada, dejando pasar únicamente los contenidos u objetos relevantes para cada usuario. Podemos encontrarnos con sistemas de recomendación orientados al comercio electrónico, otros que recomiendan lecturas de libros, *weblogs*, publicaciones electrónicas, artículos científicos, y otros muchos enfoques.

Por último, se han comentado los agregadores de contenidos que recogen información de diversas fuentes de la Web, permitiendo la consulta simultánea de muchas páginas y aglutinando toda esa información en un mismo lugar. El auge de los lenguajes de marcado sintáctico como RSS o Atom han fomentado la aparición de grandes cantidades de información que se actualizan continuamente. Este volumen elevado de contenidos deberá gestionarse de manera inteligente para evitar la sobrecarga informativa del usuario.

La línea de trabajo de esta tesis se orientará al diseño de un sistema de recomendación. Se recuperará y puntuará el contenido de diversas fuentes de información para seleccionar automáticamente la información más relevante a cada usuario. Así, el sistema NectaRSS se aplicará a la elaboración de un agregador inteligente de contenidos, utilizando el modelo del espacio vectorial, que recomendará información al usuario: una especie de híbrido entre los sistemas de recomendación y los agregadores típicos.

Capítulo 3

EVALUACIÓN DE LOS SISTEMAS RI

Paralelamente al desarrollo de la tecnología de RI ha surgido un área de trabajo dedicada expresamente a establecer medidas para valorar su efectividad. Existen evaluaciones basadas en la relevancia de los documentos, otras basadas en los usuarios, y un tercer conjunto de medidas alternativas que evitan realizar juicios de relevancia.

Con objeto de sentar las bases necesarias para valorar el funcionamiento del sistema NectaRSS, se repasarán las técnicas empleadas habitualmente en la evaluación de los sistemas RI, distinguiendo en primer lugar entre **relevancia** y **pertinencia**, para posteriormente exponer los métodos tradicionales donde se emplean medidas basadas en la *relevancia*, tales como la **exhaustividad**, la **precisión** y la **R-Precisión**, utilizada para comparar el rendimiento de dos algoritmos. Por último, se presentarán una serie de medidas alternativas como la **exhaustividad y precisión normalizadas**, el **ratio de deslizamiento** y la **medida de Voiskunskii**.

3.1. Relevancia y Pertinencia

Es necesario definir con certeza cuando un documento es relevante porque esto marcará en gran medida los resultados de un proceso de evaluación. Así, el término *relevancia*, según [RAE, 2003], es "cualidad o condición de relevante, importancia, significación", y el término *relevante* se define como "importante o significativo" y "sobresaliente o destacado". Podemos entender entonces que un documento recuperado se considerará relevante cuando su contenido posea alguna importancia o significación en relación con la necesidad de información del usuario.

Aún conociendo de manera concisa el significado del término, pueden surgir problemas a la hora de determinar con exactitud cuándo un documento puede considerarse como relevante o no:

- El mismo documento puede ser considerado como relevante por una persona e irrelevante por otra, en función de la necesidad de información que posean ambas.

Incluso el mismo documento puede resultar relevante o no a la misma persona en momentos diferentes [Lancaster, 1993].

- Es difícil definir criterios a priori para determinar cuándo es relevante un documento, "resulta más fácil proceder a la determinación de la relevancia que explicar cómo se ha llevado a cabo" [Blair, 1990]. Se considera además que "el concepto de relevancia está afectado de gran dosis de subjetividad y puede ser explicado de múltiples maneras por distintas personas." [Blair, 1990].
- Es posible que los documentos resulten relevantes en alguno de sus apartados con una materia determinada pero no en el resto de sus contenidos. Esta relevancia parcial no se medirá solamente en términos binarios (sí/no), sino que podrá adquirir muchos valores intermedios, necesitando, por tanto, una función continua en lugar de una función binaria.

Estos problemas condicionan la viabilidad de la relevancia como criterio en la evaluación de la recuperación de información. Así, podemos considerar la idea de la "utilidad de un documento", es decir, "si el documento le va a resultar útil o no a un usuario" [Cooper, 1973]. La ventaja de este punto de vista es que un usuario puede tener problemas para definir qué es relevante y qué no lo es, pero tendrá pocos problemas para decidir si un documento le resulta útil o no.

Lancaster considera que la relevancia de un documento estará relacionada con la satisfacción del usuario ante una necesidad de información, y ante la "utilidad" que estos contenidos van a tener para él, y opina que en este caso es mejor hacer uso de la palabra "pertinencia". [Lancaster, 1993]. Es decir, *relevancia* quedará asociada con el hecho de relacionar los contenidos de un documento con un tema determinado y *pertinencia* se relacionará con la utilidad de un documento recuperado respecto a una necesidad de información individual. De esta manera, para Salton "el conjunto pertinente de documentos recuperados se puede definir como el subconjunto de documentos apropiado para la necesidad de información del usuario" [Salton, 1983].

Según [RAE, 2003], "pertinencia" significa "cualidad de pertinente", entendiendo como "pertinente" lo "que viene a propósito" o resulta oportuno. Podremos entonces decir que un documento será pertinente para un usuario cuando le resulte oportuno, proporcionándole información para algún propósito.

Asumiremos, por tanto, que un documento será relevante para nuestra necesidad de información cuando nos aporte algún contenido relacionado con nuestra petición; de esta

manera, cuando hablemos de *relevancia* se puede hablar de *pertinencia*, refiriéndonos al punto de vista del usuario que realiza la operación de recuperar información.

3.2. Métodos tradicionales de evaluación de SRI

La evaluación de los sistemas de recuperación de información puede enfocarse desde dos puntos de vista, por una parte se tendrán una serie de medidas orientadas a analizar el acceso físico a los datos, y por otra, existen medidas que pretenden analizar la pertinencia o no del contenido.

Para responder a la pregunta de *qué evaluar* en los SRI hacemos referencia al trabajo de Rijsbergen [Rijsbergen, 1979] que presenta las seis medidas de Cleverdon [Cleverdon et al., 1966]: "la cobertura de una colección, el tiempo de respuesta del sistema a una petición, la forma de presentación de los resultados, el esfuerzo realizado por el usuario, la *exhaustividad* del sistema y su *precisión*". Según el autor las cuatro primeras medidas son fácilmente estimables e intuitivas, y las dos últimas, la *exhaustividad* y la *precisión*, son las que medirán verdaderamente la efectividad del sistema.

Otro autor, Chowdhury, recoge las medidas anteriores y propone seis medidas divididas en dos grupos: el primer grupo formado por la cobertura, la *exhaustividad* y el tiempo de respuesta del sistema, y el segundo grupo formado por la *precisión*, la usabilidad y la presentación [Chowdhury, 1999].

Salton utiliza el conjunto de medidas de Cleverdon, manifestando sus dudas sobre el cálculo de la *precisión* y la *exhaustividad* [Salton, 1983]. Meadow sintetiza todas las medidas en tres grupos: las basadas en la relevancia, las medidas del proceso y las medidas del resultado [Meadow, 1993]. Estas medidas se muestran en las tablas 3.1, 3.2 y 3.3 siguientes:

Medidas basadas en la Relevancia	
Precisión	Número de documentos relevantes recuperados dividido entre el total de documentos recuperados
Exhaustividad	Número de documentos relevantes recuperados dividido entre el total de documentos relevantes
Promedio de la efectividad E-P	Promedios de la efectividad en pares de valores de *exhaustividad* y *precisión*

Tabla 3.1. Resumen de medidas basadas en la relevancia de los documentos recuperados. Fuente: [Meadow,1993].

Medidas basadas en el Proceso	
Selección	Mide cuántos documentos hay en la base de datos y el grado de solapamiento con otras relacionadas
Contenido	Tipo de documentos de la base de datos, temática de los documentos, frecuencia de actualización
Traducción de una consulta	Si el usuario puede plantear la consulta directamente o precisa intermediación
Errores en el establecimiento de la consulta	Media de errores sintácticos en la escritura de la búsqueda que propician la recuperación de conjuntos vacíos y erróneos
Tiempo medio de realización de la búsqueda	Tiempo medio de realización de una estrategia de búsqueda
Dificultad en la realización de la búsqueda	Problemas que los usuarios inexpertos se pueden encontrar
Número de comandos precisos para una búsqueda	Promedio de instrucciones necesarias para realizar una búsqueda
Coste de la búsqueda	Costes directos e indirectos en su realización
Nº de documentos recuperados	Extensión del resultado de una búsqueda
Nº de documentos revisados por el usuario	Promedio de documentos que los usuarios están dispuestos a revisar

Tabla 3.2. Resumen de medidas basadas en la evaluación de los procesos. Fuente: [Meadow, 1993].

Medidas de resultado	
Precisión	Número de documentos relevantes recuperados dividido entre el total de documentos recuperados
Exhaustividad	Número de documentos relevantes recuperados dividido entre el total de documentos relevantes
Promedio de la efectividad E-P	Promedios de la efectividad en pares de valores de *exhaustividad* y *precisión*
Medidas promedio de la satisfacción del usuario	Medidas que pretenden cuantificar la reacción de los usuarios ante el resultado de una búsqueda

Tabla 3.3. Resumen de medidas basadas en el resultado obtenido. Fuente: [Meadow, 1993].

El conjunto de medidas basadas en la relevancia es el que se considera más importante, las medidas basadas en el proceso sirven para diferenciar unos sistemas de otros basándose en las prestaciones de la aplicación informática y no permiten evaluar

aspectos relacionados con el contenido de los documentos. El tercer grupo de medidas, las basadas en el resultado, están muy relacionadas con las basadas en la relevancia, introduciendo algunos aspectos diferenciadores.

3.2.1. Medidas basadas en la relevancia

Después de realizar una operación de recuperación de información, un usuario obtendrá un conjunto de documentos. En este conjunto recuperado se distinguirá un subconjunto de documentos relevantes respecto a la necesidad de información del usuario y otro subconjunto de documentos no relevantes respecto a tal necesidad. Además, normalmente este usuario dejará de recuperar cierto conjunto de documentos relevantes y cierto conjunto de documentos no relevantes con el tema buscado. En la figura 3.1 se representan estos subconjuntos, observándose la inclusión del subconjunto de documentos recuperados en el conjunto formado por la totalidad de documentos.

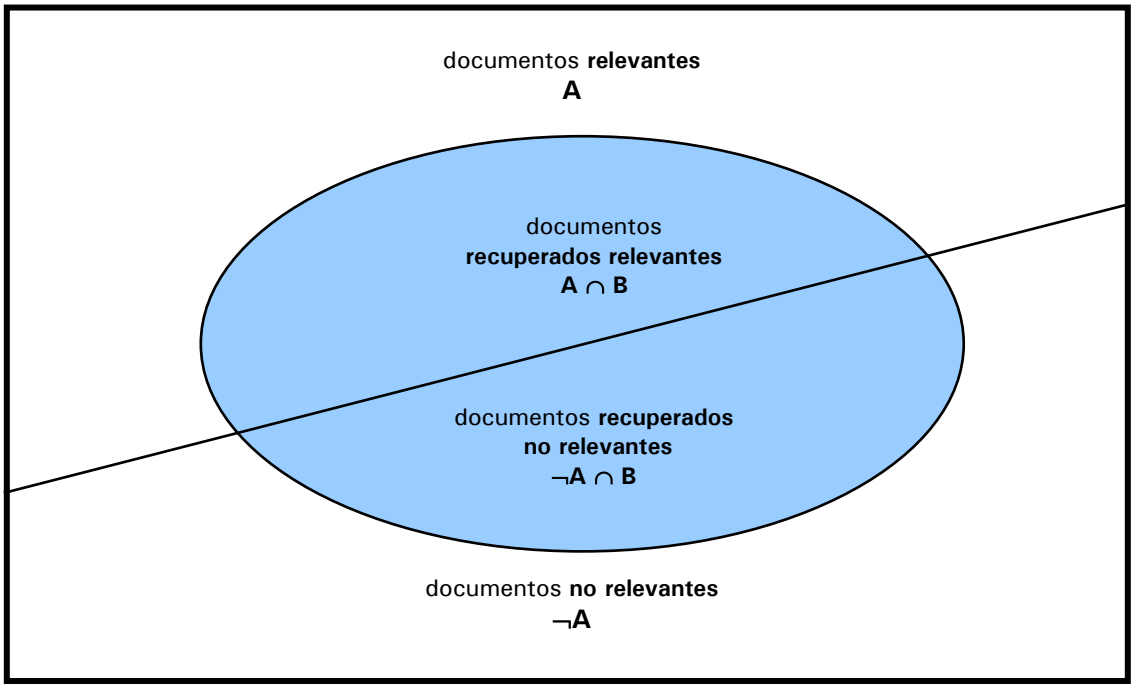


Figura 3.1. Subconjuntos de documentos considerados en una operación de recuperación de información. El color más oscuro indica el subconjunto *B* de documentos recuperados.

Rijsbergen considera esta serie de subconjuntos resultantes de una operación de búsqueda y los muestra en una *Tabla de Contingencia*, como puede verse en la tabla 3.4, en donde *A* representa el conjunto de documentos relevantes, *B* representa el conjunto de

documentos recuperados, $\neg A$ representa el conjunto de documentos no relevantes y $\neg B$ representa el conjunto de documentos no recuperados.

	RELEVANTES	**NO RELEVANTES**	
RECUPERADOS	$A \cap B$	$\neg A \cap B$	B
NO RECUPERADOS	$A \cap \neg B$	$\neg A \cap \neg B$	$\neg B$
	A	$\neg A$	

Tabla 3.4. Tabla de contingencia de Rijsbergen [Rijsbergen, 1979].

Esta *Tabla de Contingencia*, que además se puede encontrar en trabajos de otros autores [Korfhage, 1997], [Chowdhury, 1999], [Meadow, 1993] y [Frants, 1997], servirá como base para realizar una definición de las medidas de *exhaustividad*, *precisión* y de la *tasa de fallo* [Rijsbergen, 1979], tal y como se muestra en la tabla 3.5:

Precisión	$\frac{\lvert A \cap B \rvert}{\lvert B \rvert}$
Exhaustividad	$\frac{\lvert A \cap B \rvert}{\lvert A \rvert}$
Tasa de Fallo	$\frac{\lvert \neg A \cap B \rvert}{\lvert \neg A \rvert}$

Tabla 3.5. Fórmulas de la *Precisión*, *Exhaustividad* y *Tasa de Fallo* [Rijsbergen, 1979]

La *precisión* medirá el porcentaje de documentos recuperados que resultan relevantes con el tema, y se calculará dividiendo el número total de documentos relevantes recuperados entre el total de documentos recuperados.

La *exhaustividad* se calculará dividiendo el número de documentos relevantes recuperados entre el número total de documentos relevantes. Este denominador será muy difícil conocerlo de antemano, como mucho se puede inferir un número aproximado pero no se podrá afirmar esa cantidad con total seguridad.

La *tasa de fallo* representará el porcentaje de documentos recuperados no relevantes respecto al total de documentos no relevantes de la base de datos. Esta medida cobrará más

importancia cuando la *precisión* esté sujeta a variaciones en el contenido de la base de datos. Se observa que la *tasa de fallo* no depende tanto de dichas variaciones: "los cambios en la *generalidad* de una colección afectan menos a la *tasa de fallo* que a la *precisión*, que resulta más sensible" [Salton, 1983]. Salton hace referencia a una nueva medida, la *generalidad* o "el grado de documentos relevantes contenidos en una colección". Una colección con un alto grado de *generalidad* tendrá una mayoría de documentos relevantes.

Las medidas anteriores se encuentran relacionadas entre si, de tal manera que "la precisión podrá definirse en función de las tres restantes" [Salton, 1983], tal y como aparece en la siguiente expresión:

$$P = \frac{(E \cdot G)}{(E \cdot G) + F(1-G)} \tag{3.1}$$

en donde P= *precisión*, E= *exhaustividad*, G= *generalidad* y F= *tasa de fallo*.

Cuanto mayor sea el valor de la *precisión*, menor resultará el valor de la *exhaustividad*, así que estas dos medidas tenderán a relacionarse de forma inversa. Esto puede observarse en un gráfico *precisión-exhaustividad*, donde cada uno de los parámetros se coloca en un eje. Un ejemplo típico de este tipo de gráfico puede verse en la figura 3.2 tomada de [Rijsbergen, 1979]. El gráfico muestra que los dos parámetros están inversamente relacionados.

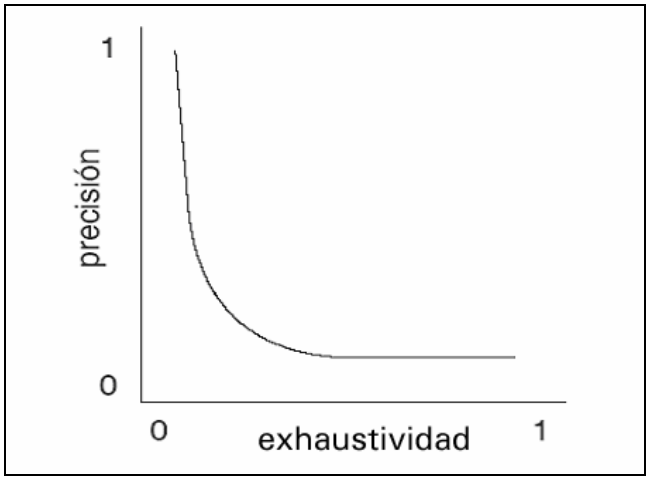


Figura 3.2. Ejemplo gráfico de la relación inversa entre *precisión* y *exhaustividad*. Fuente: [Rijsbergen, 1979].

Sin embargo, según Korfhage "no está claro que la *exhaustividad* y la *precisión* sean medidas significativas para el usuario" [Korfhage, 1997]. De hecho, la mayoría de los usuarios tienden a considerar mucho más importante la *precisión*, relegando la *exhaustividad* a un plano secundario; si una búsqueda proporciona información relevante en relación con la necesidad informativa del usuario, dicho usuario no se detiene a reflexionar sobre la cantidad de documentos relevantes que no recupera. Este razonamiento no se podrá considerar como regla general porque en ciertos ámbitos, como por ejemplo el jurídico, si que se querrá estar en posesión de todos los documentos relevantes que existan, es decir se buscará una gran *exhaustividad*.

3.2.2. Medidas orientadas al usuario

Las medidas basadas en la relevancia están muy relacionadas con el usuario que efectúa la evaluación y son difíciles de trasladar a otras personas, "se basan en el supuesto de que el conjunto de documentos relevantes para una respuesta es siempre el mismo independientemente del usuario que lleva a cabo la evaluación" [Baeza, 1999]. Pero la realidad es que diferentes usuarios podrán interpretar desigualmente qué documentos son relevantes y cuales no.

Por ello, diferentes autores presentan nuevas medidas partiendo del supuesto de que los usuarios forman un grupo homogéneo, con similar respuesta al determinar la relevancia del resultado de una operación de búsqueda [Salton, 1983], [Korfhage, 1997] y [Baeza, 1999]. Korfhage enumera estas medidas, propuestas por Keen al principio de los años setenta [Korfhage, 1997]. Se distinguen tres comunes:

- *Cobertura*, que será la proporción de los documentos relevantes conocidos que el usuario ha recuperado.
- *Novedad*, que será la proporción de los documentos recuperados relevantes que eran previamente desconocidos para el usuario.
- *Exhaustividad relativa*, que será la ratio de los documentos relevantes recuperados examinados por el usuario entre el número de documentos que el usuario está dispuesto a examinar.

Así, un valor alto de *cobertura* significará que se han encontrado la mayoría de documentos relevantes que el usuario esperaba encontrar, y un valor alto de *novedad* indicará que se ha recuperado una gran cantidad de documentos que el usuario desconocía.

Una cuarta medida orientada al usuario es el *esfuerzo de exhaustividad*, que será la ratio entre el número de documentos relevantes que el usuario espera encontrar y el número de documentos examinados al intentar encontrar esos documentos relevantes. Para ello se parte del supuesto: "la colección contiene el número deseado de documentos relevantes y el sistema permite al usuario localizar todos" [Korfhage, 1997].

3.2.3. Cálculo de la Exhaustividad y la Precisión

Según Blair, la *precisión* puede calcularse con facilidad, sin embargo, la *exhaustividad* se presenta inviable, su valor "solamente puede ser estimado" [Blair, 1990]. Este autor elaboró una revisión de los distintos métodos utilizados para estimar dicho valor, y que enumeraremos a continuación.

Un método que resultó de gran aceptación consiste en limitar el tamaño de la base de datos y calcular entonces el valor de la *exhaustividad* una vez analizados todos los documentos. Sin embargo, según Resnikoff [Resnikoff, 1976], "las pruebas a pequeña escala no dicen mucho sobre el rendimiento de un SRI o sobre las estrategias óptimas de recuperación para sistemas del mismo tipo pero mayores en tamaño".

Otro procedimiento para calcular la *exhaustividad* consiste en asignar a varias personas la tarea de analizar los documentos recuperados. Este procedimiento resulta complejo y costoso. Además contradice el sentido de la *pertinencia* de un documento para el usuario que realiza una búsqueda, dado que dos personas distintas emitirán distintos juicios de valor, y lo que sea interesante para una puede no serlo para la otra.

Una idea diferente es calcular la *exhaustividad* a partir de una muestra aleatoria de la colección de documentos. El usuario evaluará la pertinencia de los mismos y luego se estimará el número de documentos útiles de la colección empleando técnicas estadísticas. El principal problema de este método es determinar el tamaño de la muestra. Así Tague [Tague, 1994] avisa acerca de la dificultad para realizar esta tarea en bases de datos con muy bajo porcentaje de documentos relevantes; ya que en este caso el tamaño de la muestra debería ser muy grande lo que complica el análisis.

Salton apostó por calcular los valores de *exhaustividad* y *precisión* sobre una muestra de documentos de la colección total [Salton, 1983]. Este autor afirma con actitud positivista que no existen evidencias contrarias a que los resultados de este análisis puedan trasladarse sin problemas a una base de datos global y, por ello, sugiere que puede hacerse.

Un ejemplo de cálculo de la *exhaustividad* y la *precisión* sobre una muestra pequeña de una colección de documentos se expondrá a continuación. Primero suponemos que se elige una muestra constituida por los primeros siete documentos (d1, d2,…, d7) en la que resultan relevantes los documentos {d1, d3, d4, d7}. Siguiendo el método de Salton, los valores calculados para la *exhaustividad* y la *precisión* son los siguientes:

	Relevante	**E**	**P**
d1	X	0.25	1
d2	X	0.5	1
d3		0.5	0.66
d4	X	0.75	0.75
d5		0.75	0.6
d6		0.75	0.5
d7	X	1	0.57

Tabla 3.6. Ejemplo de cálculo de la *exhaustividad* y la *precisión*, según Salton, en una muestra de 7 documentos.

Según Salton, los cálculos del par *exhaustividad-precisión* (E-P en adelante) deben realizarse documento a documento. Así para el primer documento, d1, se ha recuperado un único documento pertinente, la *precisión* debe valer uno (un documento relevante para un documento recuperado) y la *exhaustividad* debe valer 0.25 (un documento relevante entre el total de documentos relevantes).

Para d2, la *precisión* resultará de dividir el valor de dos documentos relevantes recuperados entre el total de documentos recuperados hasta el momento, que también son dos, por ello su valor será uno nuevamente. La *exhaustividad* valdrá ahora 0.5 al dividir el número de dos documentos relevantes recuperados entre el total de cuatro documentos relevantes. Siguiendo este método se determina el resto de pares E-P, y se puede construir un gráfico como el que se muestra en la figura 3.3.

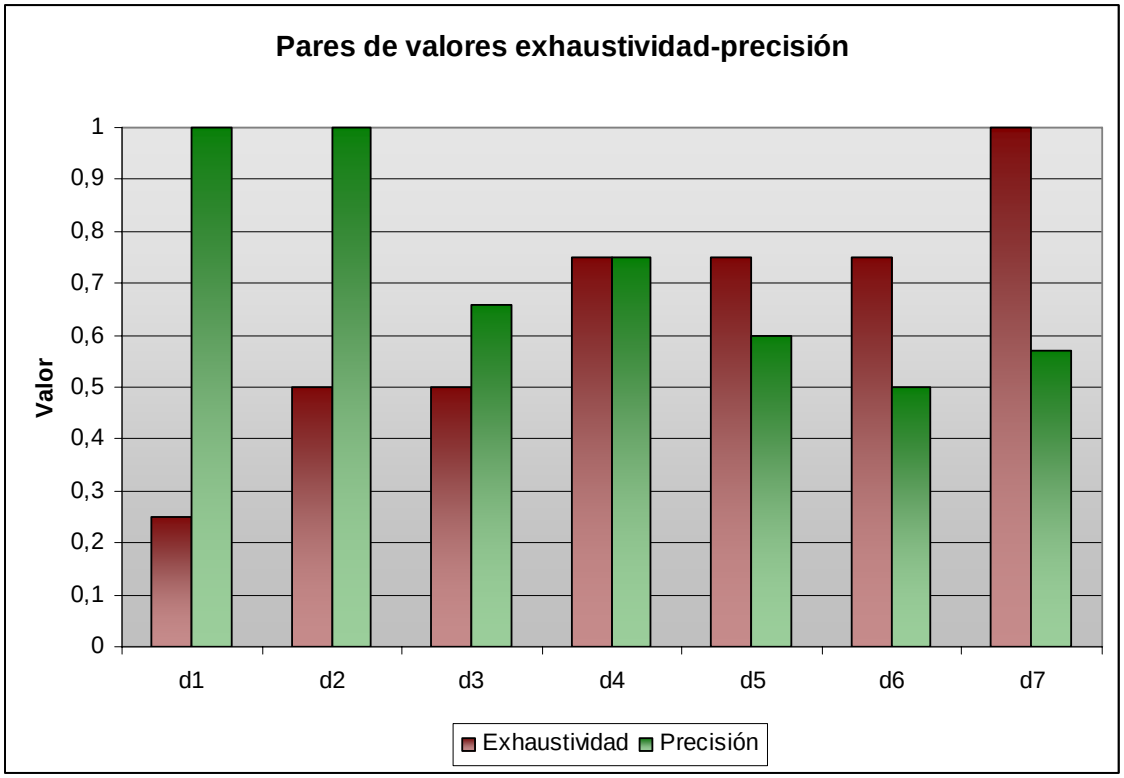


Figura 3.3. Representación gráfica de los pares de valores E-P del ejemplo de cálculo de la *exhaustividad* y la *precisión* según Salton, tomados de la tabla 3.6.

Este tipo de gráfico ha sido duramente criticado por considerase que no refleja claramente "el tamaño del conjunto de documentos recuperados y el tamaño de la colección" [Salton, 1983].

Además, en el gráfico se muestra una sucesión discreta de valores E-P en vez de una sucesión continua de los mismos. Así, por ejemplo, no se indica qué valor de *precisión* corresponde a un valor de *exhaustividad* de 0.5, ya que el mismo varía desde el valor inicial de 1 hasta el de 0.66.

3.2.4. Medidas promedio *exhaustividad-precisión*

Buscando solucionar los problemas anteriores, Salton propuso el cálculo de los pares de medidas E-P en términos de promedio, "el promedio que el usuario puede esperar de la realización de búsquedas por parte del sistema, puede ser calculado tomando la media aritmética sobre un número de *N* búsquedas de la *exhaustividad* y de la *precisión* individuales

de cada una de ellas". Según esta propuesta, la formulación de las medidas promedio E-P será:

$$Exhaustividad(D) = \frac{1}{N}\sum_{i=1}^{N}\frac{RecRel(D)_i}{RecRel(D)_i + NoRecRel(D)_i}, \qquad (3.2)$$

$$Precisión\ (D) = \frac{1}{N}\sum_{i=1}^{N}\frac{RecRel(D)_i}{RecRel(D)_i + RecNoRel(D)_i}, \qquad (3.3)$$

en donde *RecRel(D)* serán los documentos recuperados relevantes, *NoRecRel(D)* serán los documentos no recuperados relevantes y *RecNoRel(D)* serán los documentos recuperados no relevantes, siendo *D* el conjunto de documentos.

A partir de las fórmulas (3.2) y (3.3) se puede representar una curva E-P con valores diferentes de *exhaustividad* para cada valor de la *precisión*. Esta función será continua en vez de discreta y coincidirá con la curva propuesta por Rijsbergen [Rijsbergen, 1979]. En la figura 3.4 puede observarse una representación de este tipo correspondiente a los pares de valores E-P del ejemplo. A este método de cálculo de los valores E-P se le llama también como cálculo de *exhaustividad* y *precisión* relativa, entendiéndose estas medias como aproximaciones a los verdaderos valores de ambos ratios. Esta forma de representar la relación de los pares de valores E-P resultará también válida cuando se realiza una única búsqueda.

Korfhage propone dos métodos distintos para calcular el promedio de la *exhaustividad* y la *precisión*. El primero parte del supuesto de que se conocen a priori los documentos relevantes para cada conjunto de preguntas. Se supone además que cada pregunta no se realiza hasta que sea satisfecha determinada condición como, por ejemplo, recuperar un número determinado de documentos. Entonces se miden la *exhaustividad* y la *precisión* obteniendo un par de valores para cada pregunta. Finalmente se puede construir una tabla E-P aumentando en valor de 0.1 ambas medidas [Korfhage, 1997].

El otro método consiste en calcular los promedios de la *precisión* para un conjunto de tres o de once valores previamente establecidos de la *exhaustividad*. Estas dos técnicas se conocen como "promedio en tres puntos" y "promedio en once puntos".

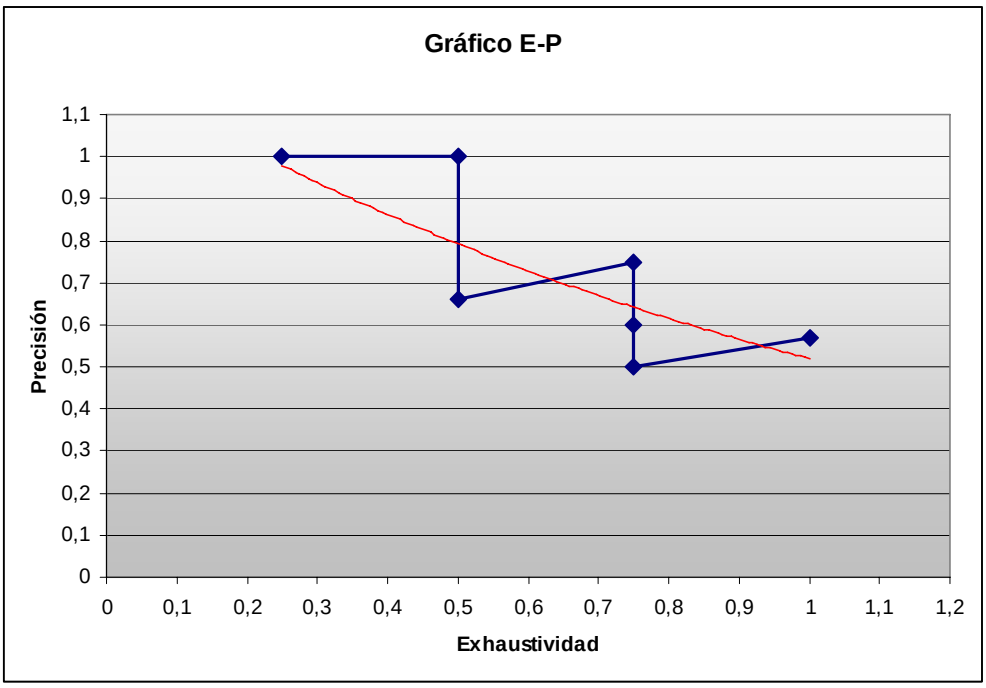


Figura 3.4. Representación gráfica de los pares de valores E-P del ejemplo descrito en la sección 3.2.3 junto con la curva propuesta por Rijsbergen en [Rijsbergen, 1979], en color rojo.

3.2.5. Valores sumarios simples

Según [Baeza, 1999], en ciertas situaciones se desea comparar el rendimiento en la recuperación de varios algoritmos para consultas individuales. Primero porque la precisión media sobre varias consultas puede disfrazar importantes anormalidades de los algoritmos en estudio, y segundo porque cuando comparamos dos algoritmos podemos estar interesados en investigar si uno de ellos funciona mejor para cada consulta en un conjunto dado de consultas. En estas situaciones se puede utilizar un valor simple de precisión, que podrá interpretarse como un resumen de la correspondiente curva precisión-exhaustividad. Normalmente, este valor simple se tomará como la precisión en un nivel determinado de exhaustividad.

3.2.5.1. Precisión media al observar documentos relevantes

Se obtendrá un valor sumario simple, para un conjunto de documentos ofrecidos en orden de relevancia, calculando la media de los valores de precisión obtenidos después de cada aparición de un documento relevante. Por ejemplo, si los valores de precisión al ir observando 5 documentos relevantes son 1, 0.6, 0.5, 0.4 y 0.3, entonces la precisión media

será (1+0.6+0.5+0.4+0.3)/5, es decir, 0.56. Esta medida favorecerá a los sistemas que recuperen documentos relevantes rápidamente. Algunos algoritmos pueden obtener un alto valor de precisión media al observar documentos relevantes y sin embargo tener un valor pobre de exhaustividad global.

3.2.5.2. La R-Precisión

La idea aquí será generar un valor sumario simple para un conjunto de documentos ofrecidos en orden de relevancia, calculando la precisión en la posición R del orden, siendo R el número total de documentos relevantes para la consulta actual. Por ejemplo, si consideramos R=10 y existen 4 documentos relevantes entre los diez primeros del orden, entonces se tendrá una *R-Precisión* de 0.4, al dividir los 4 documentos relevantes entre los 10 documentos recuperados. Esta medida puede utilizarse para observar el comportamiento de un algoritmo para cada consulta individual en un experimento. También se puede calcular la *R-Precisión* media de todas las consultas, no obstante, utilizar un número simple para resumir todo el comportamiento de un algoritmo de recuperación a lo largo de diversas consultas puede resultar impreciso.

3.2.5.3. Histogramas de Precisión

Las medidas de la *R-Precisión* para varias consultas podrán utilizarse para comparar la historia de recuperación de dos algoritmos. Así, considerando a $RP_A(i)$ y $RP_B(i)$ como el valor de la *R-Precisión* para un algoritmo *A* y un algoritmo *B* en la consulta *i*, respectivamente, podemos definir la diferencia entre ambos valores como:

$$RP_{A/B}(i) = RP_A(i) - RP_B(i) \qquad (3.4)$$

Un valor de $RP_{A/B}(i)$ igual a cero indicaría que ambos algoritmos tienen igual rendimiento para la consulta i, en términos de la *R-Precisión*. Si $RP_{A/B}(i)$ es positivo entonces indicaría un mejor rendimiento para el algoritmo *A*, y si el valor es negativo sería el algoritmo *B* el que ofrece mejor rendimiento para la consulta *i*. Estos resultados se pueden representar en un gráfico denominado *histograma de precisión*, que permitirá comparar rápidamente el rendimiento en la recuperación de los dos algoritmos, mediante una simple inspección visual, tal y como se muestra en el ejemplo de la figura 3.5.

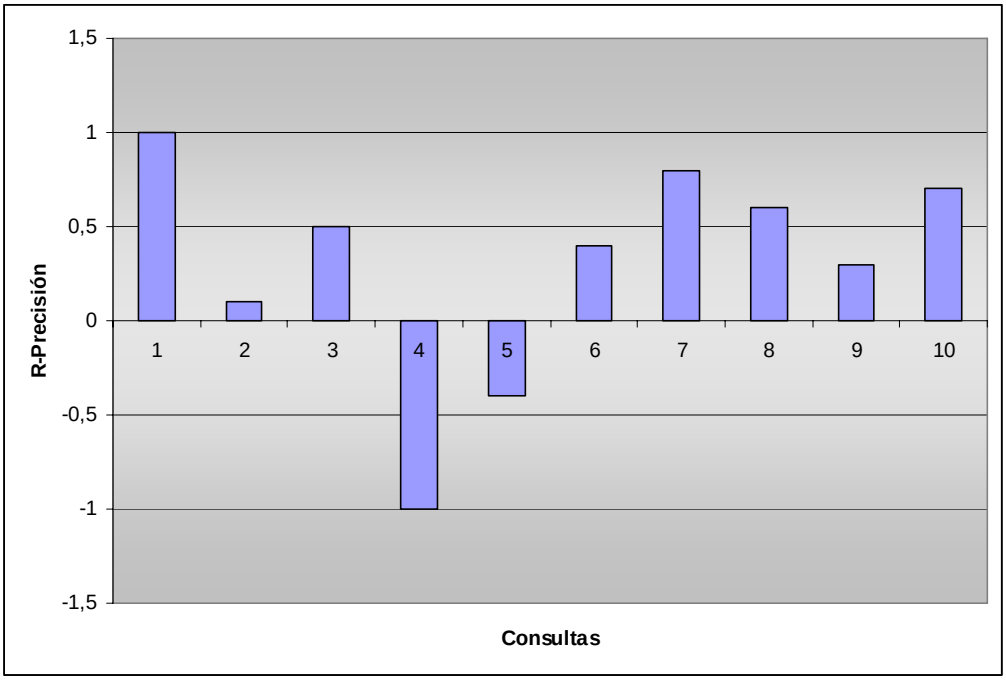


Figura 3.5. Histograma de precisión para dos algoritmos diferentes. El cálculo de los valores se realiza restando la *R-Precisión* calculada en diez consultas hipotéticas, según la fórmula (3.4). Fuente: [Baeza, 1999]

3.3. Otras medidas alternativas

Existe un amplio conjunto de medidas que intentan superar los problemas descritos en la sección 3.2.3 del cálculo de la *exhaustividad* y la *precisión*. Salton denomina a estas medidas "de valor simple" porque ya no se va a representar el resultado de una evaluación en función de un par de valores sino de un único valor [Salton, 1983]. Para este autor, las medidas alternativas deberían cumplir las siguientes condiciones:

- Deben ser capaces de reflejar la efectividad de la recuperación únicamente, de forma separada de otros criterios como el coste.
- Deben ser independientes de cualquier límite, es decir, el número de documentos recuperados no debe afectar a estas medidas.
- Deben ser expresadas en un número simple, en lugar de utilizar pares de valores.

3.3.1. Exhaustividad y precisión normalizadas

Uno de los problemas del uso de las medidas de *exhaustividad* y *precisión* proviene de la lectura secuencial de los resultados de una búsqueda, "los SRI típicos muestran los resultados al usuario formando una secuencia de documentos. Incluso en sistemas que no presentan así la información, el usuario suele examinar los documentos secuencialmente. Este modo de examinar afectará al juicio que el usuario dará sobre la relevancia o no de los documentos siguientes" [Korfhage, 1997].

Otro caso muy común sucede cuando al realizar una búsqueda los primeros documentos recuperados resultan relevantes con el tema de interés de un usuario. Este usuario tendrá una sensación positiva y no se preocupará del número de documentos no relevantes que también se hayan recuperado. Por el contrario, si hay muchos documentos no relevantes al principio, el usuario tendrá sensación de frustración aunque globalmente se le proporcionen más documentos relevantes que no relevantes. Estas reflexiones propician el desarrollo de medidas que tomen en cuenta la secuencia en que se presentan los documentos al usuario.

En esta línea, Rocchio [Rocchio, 1966] define la *exhaustividad* y la *precisión normalizadas*, para sistemas que presenten los documentos alineados según un criterio de clasificación, y donde el tamaño de la muestra analizada no afecta [Rijsbergen, 1979], [Korfhage, 1997].

Primero considera un sistema ideal donde los documentos relevantes se recuperan antes que los documentos no relevantes y representa en un gráfico la evolución de la *exhaustividad* de esta operación de recuperación de información. Así, por ejemplo, si se sabe que en una base de datos con 25 documentos existen cinco de ellos relevantes que han sido devueltos en las posiciones {3, 5, 10, 11, 15} podemos representar la *exhaustividad* como se muestra en la figura 3.6 siguiente.

Se observa que al analizar el tercer documento la *exhaustividad* alcanzará el valor de 0.2, un documento relevante divido entre el total de cinco documentos relevantes de la colección. Cada vez que se analice un documento relevante aumentará el valor de la *exhaustividad* hasta llegar a la unidad en el documento 15. En la misma figura se representa la gráfica de la mejor búsqueda posible: si los cinco documentos relevantes estuvieran en las cinco primeras posiciones de la secuencia, y la gráfica de la peor búsqueda posible: al presentarse los cinco documentos relevantes en las cinco últimas posiciones de la secuencia.

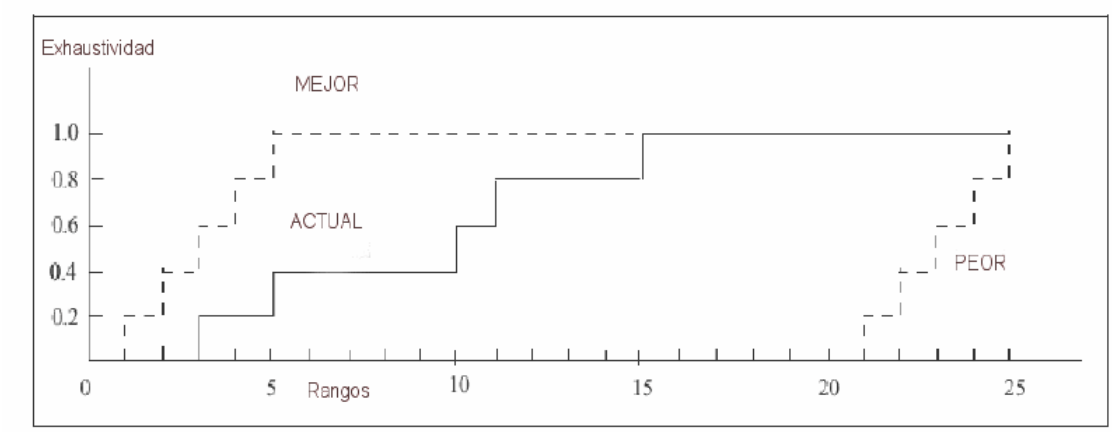


Figura 3.6. Ejemplo de *exhaustividad normalizada* para una búsqueda. En la misma gráfica se muestra la mejor búsqueda posible y la peor búsqueda posible. Fuente: [Rijsbergen, 1979].

Según Korfhage, "el área comprendida entre la búsqueda actual y la gráfica ideal representará una medida de la ejecución del sistema RI" [Korfhage, 1997]. Esta medida, la *exhaustividad normalizada*, se calculará restando a la unidad el resultado de dividir el valor de dicho área entre *(n1 * (N - n1))*, en donde *n1* es el número de documentos relevantes y *N* es el número total de documentos.

Para el cálculo de la *precisión normalizada* Rijsbergen propone "restar a la unidad el resultado de dividir el valor de este área por el valor del área existente entre la búsqueda ideal y la peor búsqueda" [Rijsbergen, 1979].

3.3.2. Ratio de deslizamiento

Esta medida "se basa en la comparación de dos listas ordenadas de documentos recuperados. Una lista es la salida del sistema actual y la otra representa un sistema ideal donde los documentos recuperados se muestran en orden descendente" [Salton, 1983]. Se permite la asignación de pesos a los documentos en función del grado de relevancia con la pregunta realizada por el usuario. La ratio se establece como el resultado de dividir la suma de los pesos de los documentos recuperados por el sistema real entre la suma de los pesos de los documentos que hubiera devuelto el sistema ideal.

En este modelo se sustituye la asignación binaria de relevancia de un documento por la asignación de un peso. La situación más favorable sería que la búsqueda realizada fuera exacta a la que ofrecería el sistema ideal, adquiriendo la ratio de deslizamiento el valor de uno.

A continuación veremos un ejemplo propuesto por [Korfhage, 1997]. Supongamos que un sistema ha recuperado 10 documentos con los siguientes pesos: 7.0, 5.0, 0.0, 2.5, 8.2, 4.5, 3.7, 1.1, 5.2 y 3.1, en el orden de recuperación. Con estos pesos se confecciona la columna "Σ pesos reales" que se muestra en la tabla 3.7. En un sistema ideal, estos documentos habrían sido recuperados y presentados en el orden descendente de pesos, formando la columna "Σ pesos ideales" de dicha tabla.

La *ratio de deslizamiento* se calcula dividiendo cada valor de la columna denominada "Σ pesos reales" entre el correspondiente valor de la columna "Σ pesos ideales". Así, por ejemplo, el resultado de 0.85 es el resultado de dividir el valor 7.0 entre el valor 8.2.

Ratio de Deslizamiento			
N	**∑ pesos reales**	**∑ pesos ideales**	**Deslizamiento**
1	7.0	8.2	0.85
2	12.0	15.2	0.79
3	12.0	20.4	0.59
4	14.5	25.4	0.57
5	22.7	29.9	0.76
6	27.2	33.6	0.81
7	30.9	36.7	0.84
8	32.0	39.2	0.82
9	37.2	40.3	0.92
10	40.3	40.3	1

Tabla 3.7. Ejemplo de cálculo de la *ratio de deslizamiento*. El Deslizamiento se calcula dividiendo la sumatoria de pesos reales entre la sumatoria de pesos ideales. Fuente: [Korfhage, 1997].

3.3.3. Medida de Voiskunskii

Este autor considera que los criterios para comparar los resultados de una búsqueda "deben proveer una comparación pragmática y justificada de los resultados de la búsqueda; y la cantidad de trabajo necesaria para determinar la información requerida para el establecimiento de estos criterios debe ser admisible" [Voiskunskii, 1997].

Tradicionalmente se ha empleado la medida de valor simple propuesta por Borko $I_1=E+P$, es decir, la suma de los valores de la *exhaustividad* y la *precisión*, aunque estas dos medidas no cumplen totalmente los criterios comentados, fundamentalmente porque se

infiere el valor de la *exhaustividad*. Para la medida I_1, una búsqueda será mejor que otra cuando mayor sea el valor de la suma. Sin embargo, esta medida puede conducir, a veces, a conclusiones equivocadas. Como ejemplo, expondremos un caso enunciado por Frants, Shapiro y Voiskunskii: "supongamos que sobre una colección de 10.000 documentos, de los cuales se consideran pertinentes 100, se llevan a cabo tres operaciones de búsqueda con los resultados siguientes:

a. Se recuperan 100 documentos, 50 de ellos son pertinentes y el resto no lo son.

b. Se recuperan 67 documentos, siendo pertinentes 40 de ellos.

c. Se recupera un solo documento que resulta ser pertinente.

Calculando los valores de *exhaustividad* y de *precisión* obtendremos los siguientes valores para la medida I_1:

Búsqueda	**E**	**P**	I_1
a	0.5	0.5	1
b	0.4	0.597	0.997
c	0.01	1	1.01

Tabla 3.8. Ejemplo de cálculo de la medida I_1 de Borko. Fuente: [Frants, 1997].

Interpretando los valores de la tabla, la mejor búsqueda resultaría ser la "c" al tener el valor más alto para I_1 [Frants, 1997]. Sin embargo, la búsqueda "c" difícilmente podrá considerarse como la mejor de las tres búsquedas para un usuario, máxime cuando sólo se le proporciona un único documento, por lo que será casi seguro que el usuario preferirá cualquiera de las otras dos búsquedas que le entregan más documentos, independientemente del valor matemático que nos devuelva la fórmula.

Frants, Shapiro y Voiskunskii proponen una nueva medida de valor simple para resolver este problema, la medida I_2 calculada a partir de la ratio entre el cuadrado de documentos relevantes recuperados y el número de documentos que conforman el resultado, "ratio cuya formulación analítica se corresponde con la raíz cuadrada del producto de los valores E-P" [Voiskunskii, 1997] y [Martínez, 2004]. Si aplicamos esta medida al anterior ejemplo planteado, los resultados serán los reflejados en la tabla 3.9.

En este caso, al analizar los resultados de la tabla, se observa que el valor más alto para I_2 corresponde a la búsqueda "a", considerando, por tanto, dicha búsqueda como la mejor, conclusión que resulta más lógica y coherente que la anterior.

En la práctica, la medida I_1 de Borko y la medida I_2 de Voiskunskii suelen coincidir en sus resultados, excepto en casos extraordinarios, como el descrito en el ejemplo.

Búsqueda	**E**	**P**	I_2
a	0.5	0.5	0.25
b	0.4	0.597	0.2388
c	0.01	1	0.01

Tabla 3.9. Ejemplo de cálculo de la medida I_2 de Voiskunskii. Fuente: [Frants, 1997].

3.4. Resumen

En este capítulo se repasan las técnicas y medidas empleadas en la evaluación de los sistemas de Recuperación de Información.

Se comienza distinguiendo los conceptos de *relevancia* y *pertinencia*, siendo relevante un documento cuando su contenido posea alguna importancia o significación en relación con nuestra necesidad de información, y siendo pertinente el documento cuando nos resulte oportuno, es decir, que nos proporcione información para algún propósito. Podemos asumir entonces que un documento será relevante para nuestra necesidad de información cuando nos aporte algún contenido relacionado con nuestra petición.

Posteriormente, se repasan los métodos tradicionales de evaluación de los sistemas RI, donde se emplean medidas basadas en la relevancia tales como la *exhaustividad* y la *precisión*, que están inversamente relacionadas. La *exhaustividad* relacionará el número de documentos relevantes recuperados con el número total de documentos relevantes y la *precisión* medirá el porcentaje de documentos recuperados que resultan relevantes con el tema.

En el supuesto de que los usuarios formen un grupo homogéneo, con similar respuesta al determinar la relevancia del resultado de una operación de búsqueda, se proponen otras medidas, orientadas al usuario, como la *cobertura*, la *novedad* y la *exhaustividad relativa*.

Se analiza con detenimiento el cálculo de la *precisión* y de la *exhaustividad* porque, según algunos autores, la *precisión* puede hallarse con facilidad pero el cálculo de la *exhaustividad* se presenta inviable, su valor solamente puede ser estimado. Algunos métodos para calcular la *exhaustividad*, como los manuales, resultan complejos y costosos. En otros casos se utiliza una muestra aleatoria de la colección de documentos. Para intentar solucionar estos problemas se proponen las medidas promedio *exhaustividad-precisión*.

Para comparar el rendimiento en la recuperación de varios algoritmos se proponen los valores sumarios simples, tales como la *precisión media*, la *R-Precisión*, donde se tendrá en cuenta la ordenación por relevancia de un conjunto de documentos, y los *histogramas de precisión* que se elaboran comparando los valores de *R-Precisión* de los algoritmos considerados.

Se proponen además otras medidas alternativas, tales como la *exhaustividad y precisión normalizadas*, para sistemas que presenten los documentos alineados según un criterio de clasificación, el *ratio de deslizamiento*, que se basa en la comparación de dos listas ordenadas de documentos recuperados, y la *medida de Voiskunskii*, calculada a partir de la ratio entre el cuadrado de documentos relevantes recuperados y el número de documentos que conforman el resultado.

Capítulo 4

PERFILES DE USUARIO

En este capítulo se da una visión global del estado del arte en la elaboración y utilización de los perfiles de usuario. Su consideración en el contexto de la Recuperación de Información está motivada en la necesidad de personalizar la información que se recupera y muestra a los usuarios, de forma que la información presentada sea lo más próxima posible a sus necesidades reales de información.

La tesis está encaminada a la propuesta de un sistema de recomendación, NectaRSS, que utilizará un perfil de usuario para representar las preferencias de éste. Por ello es importante conocer el concepto del perfil de usuario y los diversos métodos de creación y representación de perfiles, seleccionando con criterios suficientes las estrategias más adecuadas a nuestro trabajo. También es importante conocer los métodos de realimentación por parte del usuario, necesarios para que un sistema se vaya adecuando a sus intereses y circunstancias.

4.1. ¿Qué es un Perfil?

Perfil es una palabra que procede de la expresión latina "pro filare", que significa "diseñar los contornos". Un perfil será un modelo de un objeto, una representación compacta que describe sus características más importantes, que puede ser creado en la memoria de un ordenador y puede utilizarse como representante del objeto en las tareas computacionales. Las aplicaciones más conocidas que crean y gestionan perfiles incluyen la personalización, la gestión de conocimiento y el análisis de datos.

Pueden existir distintos tipos de perfiles, desde el perfil psicológico del comportamiento de un individuo, hasta el perfil del funcionamiento de un programa de ordenador. En principio, se puede hacer un perfil de todo, y por consiguiente, las características representadas en el perfil dependerán de la naturaleza del objeto modelado.

Muchos de los perfiles que se crean están referidos al usuario. Se realizan perfiles de los seres humanos como usuarios y también como clientes, éstos últimos con técnicas

específicas. El desarrollo de perfiles de clientes se ha incrementado mucho en los últimos años en las tiendas en línea y en aplicaciones de gestión de las relaciones con los clientes.

El perfil de usuario va a contener información modelada sobre el usuario, representada explícita o implícitamente, cuya explotación permitirá a un sistema incrementar la calidad de sus adaptaciones. Para obtener un perfil más actual y preciso, será necesario monitorizar las acciones del usuario de la forma más cercana posible. Esto refuerza la necesidad de emplear técnicas que automaticen de forma inteligente las tareas de creación y gestión de los perfiles de usuario.

4.2. Métodos de creación de perfiles

Pueden considerarse tres métodos principales para crear perfiles: el método explícito o manual; el método colaborativo o de composición a partir de otros perfiles, y el método implícito, que utiliza técnicas específicas para extraer las características automáticamente.

En el método explícito los datos serán introducidos directamente por el usuario, escribiéndolos en su perfil de usuario o respondiendo a formularios.

Mediante el método colaborativo se podrá crear y modificar un perfil de usuario a partir de su interacción colaborativa con otros perfiles con los que se relaciona, recurriendo a conocimiento específico del dominio y heurísticas inteligentes. En la figura 5.1 se muestra un esquema de las posibles interacciones entre distintos tipos de perfiles y sus fuentes de información.

Por último, en el método implícito, los perfiles de usuario se crearán y se modificarán automáticamente, recurriendo en la mayoría de los casos a técnicas de Inteligencia Artificial para dichas tareas.

Estos tres métodos no son excluyentes entre si, se podrán utilizar simultáneamente para producir perfiles más precisos y comprensibles.

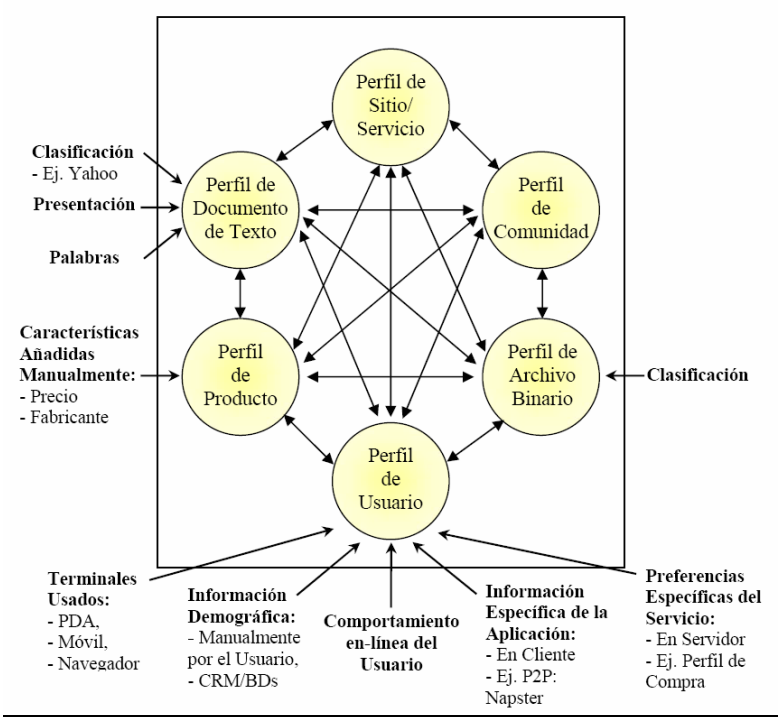


Figura 4.1. Interacciones entre diversos tipos de perfiles y sus fuentes de información, en el método colaborativo de creación de perfiles. Fuente: [Rui, 2003].

4.3. Métodos de adquisición de los datos del usuario

En esta sección se describirán algunos métodos basados en la introducción explícita de datos por el usuario, y en muchos casos basados en el comportamiento de adquisición activa del sistema. Posteriormente, se verán los métodos de adquisición pasiva: reglas de adquisición dependientes del dominio, reconocimiento del plan y objetivos, y estereotipos para la clasificación del usuario.

4.3.1. Información Explícita

La estrategia más obvia para obtener información del usuario sería aquella en la que sea el propio usuario quien proporcione los datos deseados. Estos datos se podrán obtener mediante preguntas que le realice el sistema. Algunos ejemplos de utilización de entrevistas iniciales los podemos encontrar en [Sleeman, 1985], [Rich, 1979], [Boyle y Encarnação, 1994] y [Fink et al., 1998]. Muchos sitios web recurren a entrevistas iniciales para asignar el usuario a un subgrupo de usuarios predefinido.

Un problema de este tipo de adquisición será la dificultad del usuario para autoevaluarse, sobre todo respecto a su nivel de experiencia y capacidades. Por ello, ciertos sistemas presentan al usuario un conjunto muy controlado de preguntas, tests o ejercicios, para tratar de obtener una visión objetiva del usuario. Un ejemplo de esta utilización de cuestionarios puede verse en [Akoulchina y Ganascia, 1997]. Otros sitios de la Web más orientados a un usuario consumidor pueden incorporar estas preguntas en actividades de entretenimiento, y pueden ofrecer incentivos para que el usuario las responda.

Otro problema es la *Paradoja del Usuario Activo* [Carrol y Rosson, 1987], según ésta, los usuarios se sienten motivados para comenzar la interacción y desean concluir su tarea inmediatamente. No pierden tiempo con cuestionarios, manuales o ayudas en línea. Resulta paradójico pues posiblemente ahorrarían tiempo a largo plazo "perdiendo" algún tiempo inicial para optimizar el sistema. Incluso ciertos usuarios no visitarán un sitio si tienen que responder primero a una entrevista. Por ello, "se debería permitir a los usuarios la iniciativa de proveer información personal, por ejemplo, como parte de un diálogo de preferencias" [Strachan et al., 2000], o "en momentos arbitrarios de la interacción" [Bares y Lester, 1997].

4.3.2. Reglas de Adquisición

Las reglas de adquisición servirán para generar presunciones acerca de un usuario y se ejecutarán normalmente cuando exista nueva información disponible sobre dicho usuario. En la mayor parte de los casos, estas reglas de adquisición estarán referidas a acciones observadas del usuario o a una interpretación de su comportamiento.

Las reglas de adquisición podrán ser específicas para un dominio de aplicación o independientes del dominio. Un ejemplo de adquisición independiente del dominio lo encontramos en [Chin, 1989], que utiliza heurísticas como "Si el usuario quiere conocer X entonces el usuario no conoce X". Otro ejemplo se encuentra [Kobsa y Pohl, 1995] donde se usan reglas de adquisición encajadas en actos de diálogo.

Respecto a las reglas de adquisición específicas, aunque pueden resultar de fácil implementación, su uso puede ser poco flexible y sus propiedades pueden ser difíciles de describir formalmente. Un ejemplo de su utilización puede verse en [Fink et al., 1998], y otro ejemplo detallado lo tenemos en [Strachan et al., 1997] y [Strachan et al., 2000], donde se describe el sistema TIMS. El modelo de usuario utilizado en este sistema consistirá en tres variables que representarán el nivel de experto del usuario con relación al dominio de la aplicación, su familiaridad con TIMS y con el sistema operativo. A cada una de estas

variables se les podrá asignar los valores: "principiante", "intermedio" o "experto", que serán actualizadas regularmente por el sistema utilizando reglas y heurísticas de adquisición específicas.

4.3.3. Reconocimiento del Plan

Se trata de explorar qué objetivos persigue el usuario y qué secuencia de acciones o plan realizará para lograr esos objetivos. En un sistema de reconocimiento de planes existirá una base de conocimiento de tareas para modelar las posibles acciones del usuario y las relaciones entre ellas, así como un mecanismo para identificar el plan actual y sus objetivos asociados. Los primeros sistemas de reconocimiento de planes fueron desarrollados sobre todo con métodos simbólicos. En los últimos años, se han ido aplicando cada vez más las técnicas numéricas [Albrech et al., 1997], [Bauer, 1996], y las técnicas basadas en grafos como en [Lesh, 1995].

El reconocimiento del plan de un usuario es especialmente efectivo en aplicaciones que tengan pocos objetivos posibles y pocas formas de lograrlos. En [Lesh et al., 1999] se muestra que el reconocimiento del plan del usuario acelera notablemente la interacción en una aplicación de gestor de mensajes.

4.3.4. Estereotipos

En este método, los usuarios se clasificarán en categorías y se harán predicciones sobre ellos en base a un estereotipo asociado a cada categoría. Se asumirá que si un usuario pertenece a una categoría entonces éste tendrá características y/o comportamientos semejantes a los miembros de esa categoría, bajo un conjunto determinado de circunstancias [Rich, 1979].

En un estereotipo se distinguirá, por una parte, el cuerpo, donde se mantiene la información "verdadera" para los usuarios a los que se aplica dicho estereotipo, y por otra, un conjunto de condiciones de activación del estereotipo que puede satisfacer un usuario.

Para razonar sobre la base de estereotipos se tendrán que evaluar las reglas de activación y si existen condiciones satisfechas por el usuario actual entonces se integran las presunciones correspondientes al estereotipo en el perfil de ese usuario. Por ejemplo, si el usuario "tiene interés en bebés" entonces se podría activar el estereotipo "padre" [Ambrosini et al., 1997].

Los estereotipos se han utilizado en gran cantidad de sistemas [Ambrosini et al., 1997], [Ardissono et al., 1999], [Fink et al., 1998], [Kobsa et al., 1994]. Un parámetro importante que determina la efectividad de este método va a ser la calidad de los estereotipos, es decir, cuántos diferentes estereotipos reconoce el sistema, con qué acierto atribuye los estereotipos a los usuarios y la calidad de las inferencias que se han diseñado para cada estereotipo.

4.3.5. Adquisición de Datos de Utilización

En algunos casos, además de observar el comportamiento del usuario, se intenta modelarlo para que sirva de fundamento en la adaptación del sistema. Ejemplos de sistemas que registran las acciones del usuario para obtener información de su comportamiento son *Flexcel* [Krogsaeter et al., 1994], que adapta los menús y ciertos parámetros del programa comercial Excel a un usuario concreto basándose en las tareas que éste realiza con la aplicación, y *Basar* [Thomas y Fischer, 1996], que asiste a un usuario en la manipulación de su información personal de la Web manejando sus listas de enlaces preferidos y su historia de navegación.

Otras técnicas son las empleadas por los *agentes de interfaz* y los *agentes personales* [Maes, 1994], [Mitchel et al., 1994]. "Estos sistemas serán más efectivos cuanto más aprendan los hábitos, intereses y preferencias del usuario" [Maes, 1994]. Se pretende que los agentes aprendan correlaciones entre las situaciones que el usuario encuentra y las acciones que realiza. Entonces, se utilizarán estos datos, por ejemplo, para prever el comportamiento del usuario en futuras situaciones, para recomendar acciones al usuario y para realizar automáticamente acciones por el usuario.

También se han construido perfiles de usuario orientados a su comportamiento mediante algoritmos de aprendizaje de máquinas. Una muestra es la aproximación de [Webb y Kuzmyez, 1996] en la que se pretenden aprender correlaciones situación-acción, para modelar al usuario en sistemas educacionales.

4.4. Representación del Perfil de Usuario

Una vez se haya adquirido un modelo del usuario, se necesitará una representación de ese modelo, el perfil de usuario, para que pueda ser utilizado por otros componentes del sistema. Se pueden utilizar estructuras simples para representar el modelo de usuario, como

pares "característica-valor" [Sleeman, 1985] o realizar adaptaciones directas de los contenidos que se le ofrecen al usuario, a partir de su perfil. Otros sistemas representarán los modelos adquiridos y emplearán inferencias para refinar los resultados iniciales.

Se abordarán los métodos más comunes de representación de modelos de usuario y las técnicas de inferencia asociadas. Distinguiremos epistemológicamente tres tipos de razonamiento: deductivo, inductivo y analógico.

4.4.1. Razonamiento Deductivo

La característica principal del razonamiento deductivo es que se progresará de lo general a lo particular. Dentro de este tipo de razonamiento trataremos el uso de métodos basados en la lógica y el razonamiento con incertidumbre.

4.4.1.1. Representación e Inferencia Lógica

El uso de métodos basados en la lógica ha sido analizado por diversos autores, una muestra bastante completa la podemos encontrar en [Pohl, 1998]. Un ejemplo de sistema adaptativo lo tenemos en [Kobsa y Pohl, 1995], denominado KN-AHS. Este sistema utilizará premisas sobre las creencias del usuario, representándolas mediante conceptos. Así, una premisa del tipo "usuario conoce el concepto X" se representará añadiendo una representación del concepto en la base de conocimiento del sistema.

Para representar el conocimiento del sistema sobre el dominio y el conocimiento del usuario sobre ese dominio, se pueden utilizar formalismos como los grafos de conceptos. También se pueden utilizar otros formalismos conceptuales como el cálculo de proposiciones y la lógica modal. Estos métodos no son capaces de gestionar la incertidumbre y alteran constantemente el perfil de usuario. Por ello, a veces se recurre a métodos basados en lógica no estándar como, por ejemplo, la técnica de la "manutención de verdad" [Brajnik y Tasso, 1994], [Paiva y Self, 1995].

4.4.1.2. Representación y Razonamiento con Incertidumbre

Para gestionar la incertidumbre asociada a la construcción de perfiles de usuario se pueden utilizar métodos numéricos basados en valores de evidencia [Jameson, 1996]. Un ejemplo, es HYDRIVE [Mislevy y Gitomer, 1996] que emplea redes neuronales Bayesianas.

Otra técnica basada en evidencias es la lógica borrosa que permitirá representar conceptos vagos. Un argumento de esta técnica es que los usuarios razonan en términos de conceptos vagos cuando se enfrentan con la incertidumbre y además la información que los usuarios pueden dar de sí mismos es vaga. Un ejemplo de este tipo de sistemas realiza recomendaciones de los productos más ajustados a un usuario, actuando como un asistente de ventas [Popp y Lodel, 1996].

4.4.2. Razonamiento Inductivo: Aprendizaje

En el razonamiento inductivo se progresará de lo particular a lo general, por ello se monitorizará la interacción del usuario con el sistema y se diseñarán conclusiones generales basadas en las observaciones.

En principio, los algoritmos de aprendizaje se podrán utilizar para inferir cualquier tipo de presunción sobre un usuario. En este caso, los perfiles de usuario representarán afinidades del usuario con objetos, basadas en el interés del usuario en alguna característica específica de dichos objetos. Entonces el sistema podrá realizar una recomendación personalizada de los objetos al usuario. Este tipo de recomendación se suele denominar *filtrado basado en características*. Se trata de descubrir qué preferencias tiene el usuario partiendo de determinadas características de los objetos y de clasificar los objetos como de mayor o menor interés para el usuario basándose en su perfil.

Podemos encontrar distintas técnicas de adquisición de los perfiles de intereses. En *Syskill and Webert* [Pazzani et al., 1996] se emplearon técnicas de aprendizaje automático para obtener el perfil de interés del usuario en base a clasificaciones explícitas de documentos.

En otros sistemas que utilizan aprendizaje inductivo, el perfil de interés del usuario se referirá a la información contenida en los documentos. Las características serán las palabras consideradas más o menos interesantes para el usuario. Ejemplos de estos sistemas adaptativos de recomendación basados en el interés del usuario son *Fab* [Balabanovic, 1997] y *Letizia* [Lieberman, 1995]. En [Balabanovic, 1997] se utilizan aproximaciones clásicas de los sistemas RI para describir los intereses del usuario. Los documentos y los perfiles de usuario se podrán describir mediante un modelo vectorial. Así, en el vector que represente a un documento cada peso podrá expresar la importancia de la palabra en tal documento, y en el vector que representa al perfil de usuario cada peso podrá expresar la importancia de la palabra para el usuario.

4.4.3. Razonamiento por Analogía

El razonamiento por analogía se basará en el reconocimiento de semejanzas entre usuarios. En esta sección se describirán dos aproximaciones relacionadas con el gran número de usuarios de la Web: el método de filtrado basado en grupos y la agrupación o "clustering" de perfiles de usuario.

4.4.3.1. Filtrado Basado en Grupos

En los sistemas de filtrado basado en características podemos encontrarnos con ciertos problemas: el contenido de los objetos puede no resultar fácil de analizar, dicho contenido puede no ser el único aspecto de interés por parte del usuario y puede ser difícil de expresar en forma de vectores. Además puede que los intereses del usuario no se basen en las características de los objetos. Para intentar solucionar estos problemas se proponen sistemas que buscan los usuarios que muestran un comportamiento interactivo similar. Estos sistemas se adaptarán al usuario basándose en el comportamiento de sus vecinos en intereses. Así, un perfil implícito para un usuario individual puede venir dado por el conjunto de usuarios semejantes. Esta aproximación se suele denominar *filtrado basado en grupos* [Alspector et al., 1997].

Un ejemplo de este tipo de sistema es *GroupLens* [Konstan et al., 1997] que calcula las correlaciones entre lectores de grupos de noticias de *Usenet*[1], utilizando para ello las clasificaciones de los nuevos artículos que realizan los usuarios. Estas clasificaciones se utilizarán para buscar usuarios con clasificaciones semejantes. En el sistema *Siteseer* [Rucker y Polanco, 1997] se confeccionan comunidades virtuales de usuarios basadas en sus marcadores de páginas o "bookmarks".

El rendimiento de los métodos de filtrado basado en grupos es difícil de cuantificar y muy dependiente de la distribución de clasificaciones en la población de usuarios. En [Breese et al., 1998] se puede encontrar una comparación de diferentes algoritmos de este tipo.

[1] *Usenet o Netnews es un servicio al que se puede acceder desde Internet, en el que los usuarios pueden leer o enviar mensajes, denominados artículos, a distintos grupos de noticias ordenados de forma jerárquica.*

4.4.3.2. Agrupación de Perfiles de Usuario

Al caracterizar un usuario mediante un conjunto de perfiles de otros usuarios lo que se está considerando es un perfil no explícito del usuario. En el caso de que se utilice un perfil de usuario explícito también existirán posibilidades de explorar las similitudes entre usuarios.

El sistema *Doppelganger* [Orwant, 1995] construye perfiles de usuario explícitos utilizando métodos estadísticos y de aprendizaje automático. Este sistema aplica un algoritmo de agrupación o "clustering" a los perfiles para descubrir usuarios semejantes, formando perfiles de grupos de usuarios.

[Paliouras et al., 1999] propone una aproximación híbrida, utiliza técnicas de aprendizaje para determinar el contenido de los estereotipos y para construir comunidades de perfiles de intereses. El método de aprendizaje automático que utiliza se denomina C4.5 [Quinlan, 1993] y realiza inducción en árboles de decisión. En este caso, cada árbol se corresponderá a un estereotipo para cierta variable dependiente del sistema, por ejemplo, una categoría de noticias.

El sistema de recomendación ELFI [Schwab y Kobsa, 2002] aprende explícitamente los intereses del usuario basándose en la navegación que realiza y en los documentos que selecciona. Primero obtiene estadísticamente las características del usuario, luego selecciona las características que representan los intereses del usuario para su perfil de usuario, y por último decide los documentos que recomendará basándose en dicho perfil. Esta decisión se basará en las características semejantes de los documentos o en las características semejantes de los usuarios. Para calcular la similitud entre usuarios, el sistema realizará grupos de perfiles de usuario y les aplicará la correlación de Pearson, que considera el peso de cada característica. Así, se determinará a qué grupo pertenece el usuario y se le recomendarán nuevos documentos, entre los ya visitados por el grupo y no visitados por el usuario, clasificados según una métrica propia de los autores.

4.5. Realimentación del usuario

Según [Rijsbergen, 1979], la actualización de un perfil de usuario podrá considerarse una secuencia de inferencias basadas en la observación de las interacciones del usuario, comúnmente llamadas de "feedback" o realimentación.

La realimentación del usuario puede ser de dos tipos, implícita y explícita. La realimentación implícita será difícil de detectar y de interpretar. En este caso el sistema

monitorizará el comportamiento del usuario de forma transparente para dicho usuario. En el dominio de la Web, se podrán interpretar distintos datos como realimentación implícita: seguir un enlace, el tiempo empleado en ver una página, el movimiento vertical de la página que realiza el usuario, imprimir la página, marcar la página como favorita. El problema es que este tipo de datos son muy vagos. Por ejemplo, un usuario puede seguir un enlace creyendo que le conduce a una página de interés y en realidad puede no serlo, el tiempo invertido en una página puede no ser realista, el usuario podría haberse distraído, imprimir o marcar una página como favorita puede ser debido a que el usuario tiene falta de tiempo.

Otro tipo de datos que se consideran como realimentación implícita serán los datos históricos de la actividad del usuario en el sistema. Esta fuente de información sobre el usuario puede proporcionarnos mucha información acerca de sus intereses. Así, por ejemplo, podrá utilizarse el historial de las selecciones de contenidos que realice un usuario para ir confeccionando automáticamente su perfil.

Respecto a la realimentación explícita, ésta se obtendrá preguntando directamente al usuario. Se le puede solicitar que rellene un cuestionario o que haga un juicio de valor con respecto a algo. Este tipo realimentación presentará bastantes desventajas, es muy común que un usuario no desee rellenar cuestionarios o responder a otras solicitudes. Por otra parte, la información que el usuario pueda proporcionar de sí mismo será poco fiable: puede querer dar buena imagen de sí mismo suministrando información que realmente no es la adecuada a sus intereses o necesidades. Además muchos usuarios simulan su interés en dar la realimentación y, sin embargo, responden de forma casi o totalmente aleatoria, y en ciertos casos el usuario puede no entender lo que se le solicita. De esta manera, puede suceder que el usuario y el sistema tengan modelos distintos del dominio y a su vez tener modelos distintos uno del otro [Rui, 2003].

Otro tipo de problemas estarán más relacionados con la naturaleza de la realimentación. Resulta un hecho bien conocido que el usuario ofrece realimentación positiva en muy pocas situaciones. Por otra parte, si ya ha encontrado lo que le interesa puede perder el interés en dar su opinión. En la realimentación negativa, la situación será aún peor dado que el usuario tendría que opinar sobre algo que no le interesa.

Estos inconvenientes de la realimentación explícita reafirman la conveniencia de utilizar, siempre que sea posible, una realimentación transparente para el usuario, sin que se requiera esfuerzo alguno por parte de éste.

4.6. Agentes Software y creación de perfiles

Según [Maes, 1995] "los agentes autónomos son sistemas computacionales que habitan en entornos dinámicos complejos, percibiendo y actuando de manera autónoma en ese entorno, y que realizan un conjunto de metas o tareas para las que han sido diseñados".

Los agentes se han utilizado ampliamente en distintos campos, comerciales, industriales, médicos e incluso para entretenimiento. Se han creado agentes para realizar de forma automática distintas tareas en la Web, tales como búsquedas, filtrado, resumen y presentación de información. Otros agentes recomiendan información mediante la colaboración del usuario o de usuarios que compartan intereses similares. Casi todos estos agentes, se basarán en algún modo de conocimiento del usuario.

Para [Akoulchina y Ganascia, 1997] los agentes se distinguirán del software convencional en los siguientes aspectos: autonomía, pueden deducir el estado de su ambiente y actuar de forma independiente para lograr sus objetivos; adaptabilidad, serán capaz de aprender y de adaptarse a distintas situaciones; y serán no-restrictivos, es decir, no impondrán ningún comportamiento a otras entidades como, por ejemplo, al usuario de un sistema.

La utilización de perfiles de usuario en la tecnología de agentes se centrará principalmente en las tareas de la gestión de información, donde encontraremos agentes que asisten en la navegación o en la búsqueda y agentes de recomendación. Estos agentes podrán aprender el perfil del usuario de forma automática recurriendo a técnicas de inteligencia artificial.

Un ejemplo de este tipo de agentes es *Apt Decision* [Shearin y Lieberman, 2000]. Este agente persigue el aprendizaje de las preferencias del usuario en un dominio de alquiler de pisos. Para ello, se observarán las críticas del usuario a los pisos que le vayan siendo presentados, y a partir de éstas realizará un conjunto de inferencias como base para la construcción del perfil de usuario. Cada característica de un piso tendrá un peso asociado, que será actualizado para cada usuario siempre que éste ubique esa característica en su perfil de usuario. La actualización del perfil puede ser manual, el usuario selecciona las características de los pisos que prefiere de una lista, o automática, se le sugiere al usuario que elija pisos prototipos en parejas para inferir automáticamente algunas preferencias del usuario y actualizar entonces su perfil.

4.7. Modelos Estadísticos

Estos modelos de creación de perfiles se caracterizan porque llevan a cabo diversos análisis estadísticos del comportamiento del usuario, por ejemplo, qué operaciones realiza, qué páginas visita, qué tiempo se entretiene en una página. Los datos obtenidos se emplearán para elaborar su perfil correspondiente.

Un sistema de este tipo será el propuesto por [Chan, 1999] que construye un perfil para reflejar los intereses de un usuario sin necesidad alguna de intervención por parte de éste, partiendo de la simple observación de su comportamiento. Se considera que un perfil de usuario estará formado básicamente por dos componentes: el estimador de interés en páginas, que clasificará las páginas Web por su contenido, analizando estadísticamente el comportamiento en accesos del usuario, y un grafo de accesos a la Web donde se mantendrán n-gramas de palabras o frases que aparecen en las páginas de interés y que servirán para describir dicho interés. Estas frases o n-gramas constituirán el perfil de usuario que servirá para clasificar el interés de las páginas devueltas por un motor de búsqueda. El análisis estadístico se basará en los datos del comportamiento del usuario obtenidos a partir de cuatro fuentes principales: el histórico, los marcadores de página, el contenido de cada página y los registros de acceso. A partir de estas fuentes de datos y un conjunto de presunciones probadas empíricamente se desarrollaron métricas estadísticas para evaluar el interés de una página para un usuario.

Las presunciones empíricas consideradas en [Chan, 1999] son:

1. Las direcciones más visitadas y más recientemente visitadas son las de mayor interés.
2. Las páginas que se encuentran marcadas tienen un gran interés.
3. Si las páginas tienen enlaces y el usuario sigue la mayoría de esos enlaces eso indicará que las páginas son de interés.
4. Cuanto más tiempo pase un usuario en una página más interés tendrá esa página, y cuanto más rápido sea el cambio de página menos interés tendrá esa página.

En este último punto será necesario tener en cuenta dos matices: un rápido cambio de página puede ser debido a que la página sólo esté compuesta por un conjunto de enlaces, pese a ser de interés; y por otra parte, permanecer mucho tiempo en una página puede ser deberse a una ausencia momentánea del usuario. Para prevenir estas situaciones

se marcará un tiempo máximo de permanencia en una página y los intervalos de tiempo superiores a dicho tiempo máximo se considerarán de otra sesión.

Otro ejemplo de sistema basado en un modelo estadístico es el denominado CASPER [Rafter y Smyth, 2001]. Éste utiliza un conjunto de métricas estadísticas para construir perfiles de los intereses del usuario en la búsqueda de empleo. Los perfiles de usuario se construyen monitorizando las selecciones que realiza el usuario y el tiempo que éste emplea en la lectura de la información suministrada. Estos datos se recogen de un servidor web denominado *JobFinder* donde se graban los registros de actividad de los usuarios.

4.8. Razonamiento Basado en Reglas

Los sistemas de razonamiento basados en reglas analizarán las características de problemas pasados efectuando asociaciones a lo largo de relaciones generales, para encontrar soluciones al problema presente.

Un método para adaptar la navegación en un hiperespacio estructurado basándose en el perfil de usuario se puede encontrar en [Hijikata et al., 2001]. En este hiperespacio existirán nodos, que representan las páginas, y enlaces entre los nodos. El perfil de usuario se obtendrá observando la actividad del usuario en el sistema, y estará formado por dos partes fundamentales: un conjunto de pares (propiedad, valor) o parámetros del usuario, y la secuencia de nodos o camino recorrido por el usuario hasta el momento. El sistema dispondrá de reglas de usuario basadas en el camino recorrido y de reglas de camino basadas en los parámetros del usuario. Con estas reglas y los elementos del perfil de usuario, se realizará una adaptación del camino a seguir por el usuario, eliminando ciertos enlaces que de otra manera estarían presentes en la página.

El principal problema de estos sistemas será la dificultad para describir y definir las reglas, así como la detección y prevención de errores en éstas.

4.9. Un sistema de búsqueda adaptativa en la Web basado en un perfil de usuario automático

Se examinará el sistema propuesto por [Kazunari, 2004] ya que reúne varias características que resultan de interés. En primer lugar la elaboración del perfil de usuario se llevará a cabo

sin esfuerzo alguno por parte de éste, simplemente analizando su historial de navegación por las páginas web, en segundo lugar el proceso de elaboración del perfil es relativamente sencillo y considera una evolución temporal de los intereses del usuario, y en tercer lugar su objetivo es facilitar la búsqueda de información al usuario ofreciéndole una serie de enlaces ordenados de mayor a menor puntuación, según su perfil.

Este sistema recoge una búsqueda de información del usuario y la lleva a cabo utilizando un buscador clásico como *Google*. Entonces adapta los resultados devueltos por el buscador seleccionando aquellas páginas relevantes para el usuario según su perfil. Para ir elaborando dicho perfil de usuario, monitoriza la navegación de éste por la Web, recopilando información acerca de los distintos términos que aparecen en cada página y su frecuencia.

Se distinguen dos aspectos de las preferencias del usuario: las preferencias persistentes P^{per} y las preferencias efímeras P^{today}. En las preferencias persistentes, el perfil de usuario se desarrolla a lo largo del tiempo y se almacena para utilizarlo en futuras sesiones. En las preferencias efímeras, la información utilizada para construir cada perfil de usuario se recoge solamente durante la sesión actual y se emplea inmediatamente para realizar procesos adaptativos destinados a personalizar la sesión. El perfil de usuario P se representará mediante un vector que se construye considerando ambos tipos de preferencias: $P=aP^{per} + bP^{today}$, donde a y b son dos constantes que satisfacen $a+b=1$. Para calcular P^{today} se considerarán las preferencias correspondientes a las sesiones del día anteriores a la actual, P^{br}, y las correspondientes a la sesión actual, P^{ur}. Entonces, se utiliza la fórmula $P^{today}=xP^{br} + yP^{ur}$, siendo x e y dos constantes que satisfacen $x+y=1$.

Cada página Web se representará mediante un vector w de pesos de los distintos términos que se encuentren en ella. Cada elemento de w se calculará según el esquema *tf*, o de la frecuencia del término.

La similitud entre una página w y el perfil de usuario P se calcula según la distancia del coseno entre ambos:

$$sim(P,w) = \frac{\vec{P} \cdot \vec{w}}{\left|\vec{P}\right| \cdot \left|\vec{w}\right|} \qquad (4.1)$$

De esta manera, los resultados de una búsqueda se adaptarán al usuario de acuerdo con su perfil, mostrando el sistema en primer lugar las páginas con mayor valor de similitud.

4.10. Resumen

En este capítulo se define el concepto de perfil de usuario y se enumeran distintos métodos para la creación de perfiles. Se han repasado también diversas metodologías de adquisición de los datos del usuario: la adquisición explícita o activa y la adquisición pasiva donde se incluyen las reglas de adquisición, el reconocimiento del plan y los estereotipos. En otros casos además se intenta modelar el comportamiento del usuario, registrando sus acciones, adquiriendo sus datos de utilización.

Una vez obtenidos los datos necesarios para el perfil de usuario, es necesaria una representación de dicho perfil para que pueda ser utilizado por otros componentes del sistema. Así, dentro del razonamiento deductivo nos encontraremos con representaciones e inferencias basadas en la lógica y para tratar con la incertidumbre con los métodos numéricos basados en valores de evidencia. Dentro del razonamiento inductivo o aprendizaje se considerará el filtrado basado en las características de los objetos, el aprendizaje automático y los sistemas adaptativos basados en los intereses de los usuarios. En éstos últimos, muchos autores han utilizado un modelo vectorial para representar los documentos y los perfiles de usuario. Dentro del razonamiento por analogía, se describen dos aproximaciones relacionadas con el gran número de usuarios de la Web, tales son el método de filtrado basado en grupos y el agrupamiento de perfiles de usuario.

Otro tema tratado es la realimentación del sistema por parte del usuario, que nos permitirá actualizar su perfil. Se distingue entre la realimentación implícita, que monitoriza el comportamiento del usuario de forma transparente para éste, y la realimentación explícita, que pregunta directamente al usuario. La primera será difícil de detectar e implementar y la segunda se enfrenta con problemas relativos al interés del usuario en proporcionar realimentación o no y la calidad de dicha realimentación.

Los perfiles de usuario también se utilizan en las tecnologías emergentes de agentes software, donde pueden encontrarse agentes que asisten en la navegación o en la búsqueda y agentes de recomendación. Estos agentes podrán aprender el perfil del usuario de forma automática recurriendo a técnicas de inteligencia artificial.

Otros modelos de creación de perfiles se caracterizan porque llevan a cabo diversos análisis estadísticos del comportamiento del usuario: modelos estadísticos, o porque analizan las características de problemas pasados para realizar asociaciones y encontrar soluciones al problema presente: sistemas de razonamiento basado en reglas.

Para finalizar, se expone un sistema propuesto por [Kazunari, 2004] que permite realizar búsquedas adaptativas en la Web basándose en un perfil de usuario automático, elaborado sin esfuerzo alguno por parte del usuario. En este sistema se emplea un modelo vectorial y valores de similitud basados en la medida del coseno para clasificar los resultados de una búsqueda.

Capítulo 5

NECTARSS, UN SISTEMA DE RECOMENDACIÓN DE CONTENIDOS BASADO EN PERFILES

En los capítulos anteriores se han presentado los conceptos generales sobre los SRI y su evaluación. Además se han tratado algunos lenguajes de definición de documentos y diversos aspectos sobre la creación y utilización de perfiles de usuario.

En este capítulo se exponen las bases teóricas del sistema **NectaRSS**. Se propone un sistema de recomendación que recupera información de la Web, la puntúa en base a un perfil de usuario elaborado automáticamente y presenta dicha información ordenada al usuario según su puntuación.

El capítulo se estructura de la siguiente manera: la sección 5.1 es una introducción, en la sección 5.2, tras definir la representación de la información y del perfil de usuario utilizando el modelo vectorial [Salton, 1971, 1983], se detalla la elaboración automática del perfil de usuario en base a la información que éste seleccione. En la sección 5.3 se verá cómo se puntúa la información utilizando la medida del coseno de Salton [Salton, 1989]. Finalmente, en la sección 5.4 se realiza una descripción general del sistema propuesto aplicándolo a la elaboración de un agregador inteligente.

5.1. Introducción

El sistema que proponemos, denominado NectaRSS, está encaminado a proporcionar un mecanismo de recomendación de información, ofreciendo ésta ordenada al usuario según la puntuación que el sistema le otorgue, en base a un perfil de usuario elaborado automáticamente.

Así, dado que el término "información" es muy general, resulta adecuado restringir su significado para acercarlo más al ámbito de nuestro sistema. Entonces, la información que recuperará el sistema se denominará genéricamente como **noticias**. Una noticia estará compuesta por un **titular**, un hiperenlace a su **contenido** y opcionalmente un **resumen** de dicho contenido.

En el sistema NectaRSS se considerará además el concepto de **sesión**. Una sesión será una ejecución completa del sistema, comprendiendo la recuperación de información disponible en la Web en ese momento, según las fuentes preferidas, la monitorización de las elecciones del usuario y el cálculo del perfil de usuario al término de la ejecución del sistema. Una sesión no está referida a un día concreto, sino que en un mismo día pueden darse varias sesiones o ninguna. Incluso puede que en una sesión no se recupere nueva información o que el usuario no seleccione noticia alguna. Así, la sesión estará limitada únicamente por el inicio y fin de la ejecución del sistema.

En la figura 5.1 se muestra una visión general de este sistema propuesto, donde puede observarse que el usuario simplemente navegará por las noticias que se le ofrecen y que el perfil de usuario servirá para puntuar la información recuperada de la Web, en forma de noticias, de manera que el sistema pueda ofrecerlas ordenadas por relevancia al usuario. Por otra parte la propia selección de noticias que realice el usuario servirá de retroalimentación al sistema que actualizará automáticamente su perfil.

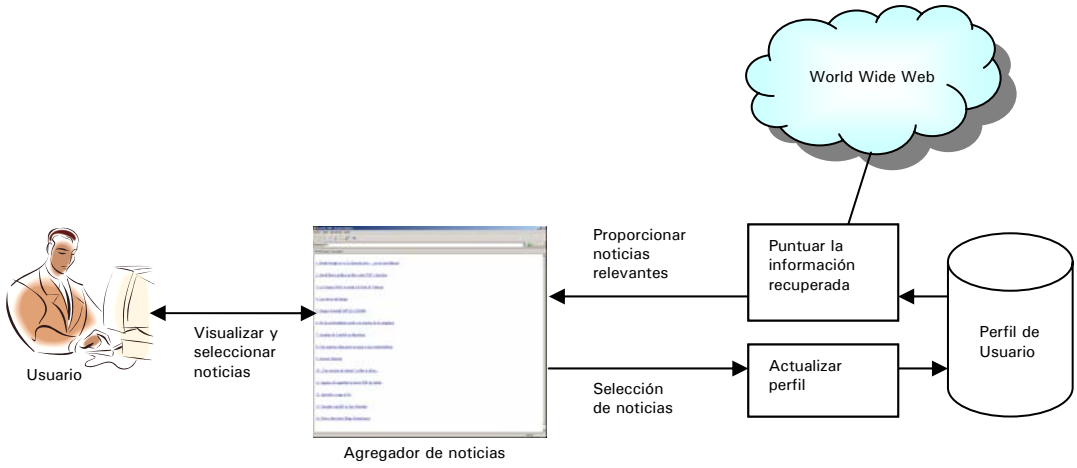


Figura 5.1. Vista general del sistema NectaRSS propuesto.

5.2. Construcción automática de un perfil de usuario basado en su historia de navegación

En nuestro enfoque, el perfil de usuario se construirá de manera implícita. En otras palabras, un usuario no deberá realizar esfuerzos explícitos como realimentación o evaluaciones para construir su perfil. Éste será elaborado de manera automática según su historial de navegación por los titulares de noticias que se le vayan ofreciendo.

El perfil de usuario *P* se desarrollará incrementalmente a lo largo de las distintas sesiones con el sistema y se guardará para utilizarlo en sesiones futuras. En cada sesión, se recopilará información acerca de las acciones del usuario y, al final de la sesión, esa información se trasladará al perfil de usuario. Así, podemos considerar un perfil de sesión P_s cuya información se recoge solamente durante la sesión actual. Un usuario puede realizar diferentes sesiones en un día y puede haber consultado diferentes titulares en ese periodo de tiempo. En nuestro método, asumiremos que las preferencias del usuario se construyen por acumulación de sus preferencias pasadas. De esta manera, iremos construyendo el perfil de usuario *P* considerando las preferencias acumuladas almacenadas en *P* y las preferencias de cada sesión almacenadas en P_s. Así, *P* reflejará un perfil de usuario construido con la historia de navegación por titulares durante *S* sesiones.

Para representar a las noticias y al perfil de usuario se utilizará el modelo vectorial propuesto por Salton [Salton, 1971, 1983], comentado en la sección 2.2.1 de esta tesis.

Así, definimos S_j *(j = 1, 2,…, N)* como el número de titulares que ha elegido el usuario en la sesión *j*. En cada sesión, P_s se construirá mediante el siguiente proceso. En primer lugar, denotaremos el vector característica w^h del titular *h (h = 1, 2,…, S_j)* como sigue:

$$w^h = (w^h_{t_1}, w^h_{t_2}, ..., w^h_{t_m}), \tag{5.1}$$

donde *m* es el número de distintos términos en el titular *h* y t_k denota cada término. Utilizando el esquema *tf*, o de la frecuencia del término, cada elemento $w^h_{t_k}$ de w^h se define como sigue:

$$w^h_{t_k} = \frac{tf_{h,k}}{\sum_{s=1}^{m} tf_{h,s}}, \tag{5.2}$$

donde $tf_{h,k}$ es la frecuencia del término t_k en cada titular *h*.

Entonces, definimos a P_s como:

$$P_s = (ps_{t_1}, ps_{t_2}, ..., ps_{t_s}), \tag{5.3}$$

donde s es el número de distintos términos en todos los titulares elegidos en la sesión j y t_k denota cada término.

Y definimos cada elemento ps_{t_k} utilizando la fórmula (5.2) como sigue:

$$ps_{t_k} = \frac{1}{S_j} \sum_{h=1}^{S_j} w_{t_k}^h \tag{5.4}$$

Cada usuario seleccionará S_j titulares en cada sesión. Ese valor S_j será diferente según el usuario. Por tanto, normalizaremos ps_{t_k} utilizando S_j como se muestra en la ecuación (5.4).

El perfil de usuario P se denotará también mediante un vector:

$$P = (p_{t_1}, p_{t_2}, ..., p_{t_n}), \tag{5.5}$$

donde n es el número de distintos términos en el perfil P y t_k denota cada término.

Cada elemento p_{t_k} se define:

$$p_{t_k} = \sum_{j=1}^{T} \frac{1}{S_j} \sum_{h=1}^{S_j} w_{t_k}^h, \tag{5.6}$$

siendo T el número total de sesiones que se hayan realizado hasta el momento.

Ahora se está en disposición de definir cómo se elaborará el perfil de usuario P al término de cada sesión. Sea P_j el perfil de usuario almacenado después de la sesión j. Entonces el perfil P_{j+1}, que se confeccionará al finalizar la sesión $j+1$, vendrá dado por las siguientes expresiones:

$$P_{j+1} = a \cdot P_j + b \cdot Ps_j \qquad para\ \forall\ p_{t_k} \subset Ps_j \qquad (5.7)$$

$$P_{j+1} = P_j \qquad para\ \forall\ p_{t_k} \not\subset Ps_j \qquad (5.8)$$

donde *a* y *b* son constantes que satisfacen *a + b = 1*. Para enfatizar la sesión actual, se le puede otorgar al parámetro *b* un peso mayor que al parámetro *a*.

Además, podemos definir un *factor de olvido fol* opcional, de manera análoga a como se propone en [Kazunari, 2004], asumiendo que ciertas preferencias del usuario decaen tras cada sesión:

$$fol(p_{t_k}) = p_{t_k} \cdot e^{-\frac{log2}{hl}} \qquad (5.9)$$

donde *hl* es un parámetro que mide el *intervalo de vida* [Kazunari, 2004].

En este caso, el perfil de usuario *P* que se calcula al final de cada sesión vendría determinado para $\forall\ p_{t_k} \subset Ps_j$ por la fórmula (5.7) anterior y para $\forall\ p_{t_k} \not\subset Ps_j$ por la fórmula (5.10) siguiente:

$$P_{j+1} = fol(P_j) \qquad para\ \forall\ p_{t_k} \not\subset Ps_j \qquad (5.10)$$

5.2.1 Consideración de los resúmenes opcionales de las noticias en la construcción del perfil de usuario

Algunas noticias pueden tener un resumen asociado. Este elemento es opcional y no estará presente necesariamente en todas las noticias que se recuperen. Aún así, se plantea la posibilidad de contar con dicha información extra en el proceso de elaboración automática del perfil de usuario. La cuestión será determinar si esta ampliación de información asociada a un titular aportará o no beneficios al perfil de usuario, y por ello al funcionamiento del sistema propuesto.

Utilizando el modelo vectorial, en este caso para los titulares que posean un resumen asociado, se considerará un vector característica w^b formado a partir de los

términos que aparezcan en el título de la noticia y un vector característica w^{hr} formado a partir de los términos que aparezcan en el resumen asociado.

Así, definimos Sr_j $(j = 1, 2,\ldots, R)$ como el número de titulares con resumen asociado que ha elegido el usuario en la sesión j. Para cada sesión, se elaborará un perfil P_r con los términos de los resúmenes, mediante el siguiente proceso. En primer lugar, denotaremos el vector característica w^{hr} del resumen asociado a un titular h $(h = 1, 2,\ldots, Sr_j)$ como sigue:

$$w^{hr} = (w^{hr}_{t_1}, w^{hr}_{t_2}, \ldots, w^{hr}_{t_v}), \tag{5.11}$$

donde v es el número de distintos términos en el resumen asociado al titular h y t_k denota cada término. Utilizando el esquema tf, de la frecuencia del término, cada elemento $w^{hr}_{t_k}$ de w^{hr} se define como sigue:

$$w^{hr}_{t_k} = \frac{tf_{hr,k}}{\sum_{s=1}^{v} tf_{hr,s}}, \tag{5.12}$$

donde $tf_{hr,k}$ es la frecuencia del término t_k en el resumen r asociado al titular h.

Entonces, definimos a P_r como:

$$P_r = (pr_{t_1}, pr_{t_2}, \ldots, pr_{t_v}), \tag{5.13}$$

y definimos cada elemento pr_{t_k} utilizando la fórmula (5.12) como sigue:

$$pr_{t_k} = \frac{1}{Sr_j} \sum_{h=1}^{Sr_j} w^{hr}_{t_k} \tag{5.14}$$

Cada usuario seguirá Sr_j titulares con resumen asociado en cada sesión. Ese valor Sr_j será diferente según el usuario. Por tanto, normalizaremos pr_{t_k} utilizando Sr_j como se muestra en la ecuación (5.14).

Entonces, si se considera la utilización de los resúmenes opcionales de las noticias en la confección del perfil de usuario, será necesario ampliar la fórmula (5.7) anterior. Ahora, el perfil P_{j+1} que se confeccionará al finalizar la sesión *j+1* vendrá dado por las siguiente fórmula:

$$P_{j+1} = (a \cdot P_j + b \cdot Ps_j) + Pr_j \qquad para\ \forall\ p_{t_k} \subset Ps_j \tag{5.15}$$

donde *a* y *b* son constantes que satisfacen $a + b = 1$.

5.3. Cálculo de la puntuación de los titulares

Para calcular la puntuación asociada a un titular *h*, compararemos su correspondiente vector característica $w^h = (w^h_{t_1}, w^h_{t_2}, ..., w^h_{t_m})$, donde *m* es el número de términos distintos en el titular *h* y t_k denota cada término, con el perfil de usuario $P = (p_{t_1}, p_{t_2}, ..., p_{t_n})$, donde *n* es el número de términos distintos y t_k denota cada término.

La similitud, $sim(P,w^h)$, entre el perfil de usuario *P* y el vector característica del titular *h*, w^h, se calculará según la siguiente fórmula de la medida del coseno, discutida en la sección 2.2.1 de esta tesis y propuesta por [Salton, 1989]:

$$sim(P, w^h) = \frac{P \cdot w^h}{|P| \cdot |w^h|} = \frac{\sum_{k=1}^{m} p_{t_k} \cdot w^h_{t_k}}{\sqrt{\sum_{k=1}^{m} (p_{t_k})^2 \cdot \sum_{k=1}^{m} (w^h_{t_k})^2}} \tag{5.16}$$

El valor de similitud obtenido mediante la ecuación (5.16) será la puntuación del titular *h* según el perfil de usuario *P*. Entonces los titulares de noticias se ordenarán para cada usuario de acuerdo con su perfil, mostrándole en primer lugar aquellos cuya puntuación sea mayor.

A continuación se expondrá un ejemplo de cálculo de la puntuación de un titular con la intención de clarificar la manera en que el sistema la lleva a cabo. Para más sencillez se considerará una noticia sin resumen asociado y no se va a considerar ningún *factor de olvido*.

Suponemos que el usuario ha seleccionado el siguiente titular *h*="Los anunciantes apuestan por los blogs". El sistema descartará las palabras vacías: "Los", "por" y "los". Entonces, se considerarán los siguientes 3 términos del titular *h*: t_1="anunciantes", t_2="apuestan" y t_3="blogs".

Según las fórmulas 5.1 y 5.2, el vector característica del titular *h* será:

$$w^h = (ps_{t_1} = 0.33,\ ps_{t2} = 0.33,\ ps_{t_3} = 0.33)$$

Ahora suponemos que se tienen los siguientes valores en el perfil de usuario, correspondientes a los términos del titular *h*:

$$P = (p_{t_1} = 0.03,\ p_{t_2} = 0.01,\ p_{t_3} = 0.09)$$

La puntuación del titular *h* respecto al perfil de usuario *P*, utilizando la fórmula de la medida del coseno (5.16), se calculará de la siguiente manera:

$$sim(P, w^h) = \frac{(0.33 \cdot 0.03) + (0.33 \cdot 0.01) + (0.33 \cdot 0.09)}{\sqrt{(0.33^2 + 0.33^2 + 0.33^2) \cdot (0.03^2 + 0.01^2 + 0.09^2)}} = 0.79$$

Entonces podemos decir que la similitud, o puntuación, entre el titular *h* y el perfil de usuario *P*, en este ejemplo, es de 0.79.

5.3.1. Puntuación alternativa de los titulares

Otra forma de calcular la puntuación asociada a un titular *h* puede realizarse utilizando la medida o coeficiente de Jaccard, visto en la sección 2.2.1 de la tesis y propuesto por [Salton, 1989].

Así, dado el correspondiente vector característica del titular *h*, $w^h = (w^h_{t_1}, w^h_{t_2}, ..., w^h_{t_m})$, donde *m* es el número de términos distintos y t_k denota cada término, y el perfil de usuario $P = (p_{t_1}, p_{t_2}, ..., p_{t_n})$, donde *n* es el número de términos distintos y t_k denota cada término, entonces la similitud, $sim(P,w^h)$, entre el perfil de usuario *P* y el vector característica del titular *h*, w^h, se podrá calcular según la siguiente fórmula de la medida de Jaccard:

$$sim(P, w^h) = \frac{\sum_{k=1}^{m} p_{t_k} \cdot w^h_{t_k}}{\sum_{k=1}^{m} (p_{t_k})^2 \cdot \sum_{k=1}^{m} (w^h_{t_k})^2 - \sum_{k=1}^{m} p_{t_k} \cdot w^h_{t_k}} \qquad (5.17)$$

El valor de similitud obtenido mediante esta ecuación (5.17) será la puntuación del titular *h* según el perfil de usuario *P*. Entonces los titulares de noticias se podrán ordenar para cada usuario mostrándole en primer lugar aquellos con mayor puntuación.

5.4. Descripción general del sistema NectaRSS

Apoyándonos en la elaboración automática del perfil de usuario descrita en la sección 5.2, y considerando el sistema de puntuación de titulares expuesto en la sección 5.3, se propone un sistema de recomendación de noticias recuperadas de la Web.

Inicialmente, el sistema NectaRSS se aplicará a la elaboración de un agregador inteligente de noticias procedentes de la Web en diversos formatos, como RSS[1] o Atom[2]. De esta manera, tendrá un aspecto y un funcionamiento similar a la mayoría de agregadores típicos, vistos en la sección 2.3.1.3 de la tesis. Una descripción del programa que lo implementa puede encontrarse en el Anexo II.

[1] *Para conocer más detalles del lenguaje RSS consultar el apartado AI.3 del Anexo I.*
[2] *Atom es otra tecnología para distribuir contenidos. Para más información consultar el Anexo I.*

En este sistema las noticias recuperadas se puntuarán de acuerdo con el perfil de usuario *P* y se mostrarán ordenadas según dicha puntuación, de mayor a menor relevancia. Así se pretende aliviar al usuario en la búsqueda de información.

El usuario no se tendrá que preocupar de nada más que seleccionar aquella información que le interese, es decir, la realimentación del sistema será implícita, sin esfuerzo alguno por su parte. Para ello se monitorizarán las selecciones que vaya realizando entre el conjunto de titulares de noticias que se le ofrecen. Con estas selecciones se irá confeccionando el perfil de la sesión P_s, definido en la expresión (5.3). Al término de cada sesión, se acumulará el perfil de sesión P_s al perfil de usuario *P*, definido en la expresión (5.5), mediante la fórmula (5.7).

Opcionalmente, el sistema puede utilizar un factor de olvido, definido en la fórmula (5.9), asumiendo que ciertas preferencias del usuario decaen tras cada sesión.

El perfil *P* se utilizará para puntuar los distintos titulares, tal y como se explica en la sección 5.3, utilizando la fórmula (5.16).

Si en la confección del perfil de usuario se consideran además los términos que aparecen en los resúmenes opcionales de las noticias, entonces se empleará la fórmula (5.15) en lugar de la (5.7), a fin de acumular al perfil de usuario *P* tanto el perfil de sesión P_s como el perfil P_r, elaborado con los términos de los resúmenes y definido en la expresión (5.13).

5.4.1. Características singulares del sistema

NectaRSS recoge algunas propuestas de [Kazunari, 2004] como la elaboración incremental del perfil de usuario de manera implícita y la presentación de la información adaptada según dicho perfil utilizando para ello una medida de similitud definida en la fórmula (5.16). Sin embargo, NectaRSS tiene varias diferencias significativas: el perfil de usuario se va elaborando al final de cada sesión utilizándose exclusivamente para personalizar la información ofrecida en la siguiente sesión, y cada sesión es independiente de las otras sin distinción alguna del día en que se han efectuado. Así, el cálculo incremental del perfil de usuario resulta más sencillo.

Además, NectaRSS distingue entre la información del titular de una noticia y la información opcional asociada a dicho titular en forma de resumen de esa noticia, reflejándolo entonces en la construcción del perfil de usuario mediante la fórmula (5.15).

Desde el punto de vista de los sistemas de recomendación, vistos en la sección 2.3.1.2 de la tesis, NectaRSS ofrece un enfoque distinto al de [García, 2002], orientado al comercio electrónico, al del [SIRLE, 2003], que realiza recomendaciones en base a las similitudes entre usuarios, y respecto a [Merelo et al., 2004], que recurre a encuestas para conocer las preferencias de los usuarios. NectaRSS puede recomendar una serie de noticias a un usuario concreto utilizando exclusivamente su perfil elaborado automáticamente.

Por otra parte, NectaRSS se ha aplicado en el ámbito de los agregadores de noticias, utilizándose para crear un agregador inteligente que recupera, filtra y recomienda información procedente de fuentes previsiblemente heterogéneas, presentándola ordenada según las preferencias de cada usuario. En dicho ámbito, no se conoce actualmente ninguna aplicación similar con estas funciones.

5.5. Resumen

En este capítulo se han expuesto las bases teóricas de un sistema de recomendación de información, denominado NectaRSS. La pretensión general de este sistema es aliviar a los usuarios en la tarea de encontrar la información que demandan.

NectaRSS se basa en la construcción automática e incremental de un perfil de usuario, en base a las distintas selecciones de titulares de noticias que vaya realizando tal usuario. Dicho perfil se utilizará en cada sesión para puntuar las noticias recuperadas por el sistema, con el objetivo de ofrecerlas ordenadas al usuario según esa puntuación calculada.

Si se considera que las preferencias del usuario decaen tras cada sesión, se plantea un *factor de olvido* opcional que se aplicará a la actualización del perfil de usuario al finalizar cada sesión con el sistema.

Además, también se propone el uso del resumen opcional de las noticias para "enriquecer" el perfil de usuario con nuevos términos al término de cada sesión.

Para representar las noticias y el perfil de usuario se utilizará el modelo vectorial propuesto por Salton [Salton, 1971, 1983]. Los elementos del vector característica de cada titular se calcularán mediante el esquema *tf* o de la frecuencia del término.

Finalmente, para calcular la puntuación de cada titular se comparará su correspondiente vector característica con el perfil de usuario, utilizando la medida del coseno [Salton, 1989] o, de manera alternativa, utilizando la medida de Jaccard [Salton, 1989].

Capítulo 6

EVALUACIÓN EXPERIMENTAL DEL SISTEMA PROPUESTO

En este capítulo se especifican las principales tareas llevadas a cabo para evaluar experimentalmente el sistema NectaRSS y se detallan las medidas utilizadas. Se comienza exponiendo el esquema general de la experimentación en la sección 6.1 y la metodología seguida en la sección 6.2. Posteriormente se comentan las estrategias empleadas para dicha experimentación, en la sección 6.3, distinguiendo dos fases principales: la primera para determinar ciertos parámetros de funcionamiento del sistema y la segunda para probar el sistema con distintos usuarios. En esta misma sección se muestra el tratamiento de las palabras y se describen los experimentos efectuados.

En la sección 6.4 se proponen distintas medidas para valorar el comportamiento del sistema, incluyendo tasas específicas y medidas tales como el *Error Medio Absoluto*, la *Correlación* entre titulares y la *R-Precisión*.

6.1. Objetivo general del sistema y esquema de su experimentación.

El objetivo de nuestro estudio será el desarrollo de un sistema para la recuperación y el filtrado inteligente de información de la Web, que recomiende noticias a un usuario en base a su perfil adquirido automáticamente, de tal manera que dichas recomendaciones satisfagan las necesidades informativas del usuario, encontrando éste más rápida y fácilmente la información que demande.

Para poder verificar este objetivo, ha sido necesario diseñar las siguientes tareas:

1. **Confección automática e incremental de un perfil de usuario basado en sus elecciones y cálculo de una puntuación asociada a cada titular de información recuperado en base al perfil de usuario**, descritas en el capítulo 5.
2. **Cálculo de diversas medidas para la evaluación del sistema**, en la sección 6.4. de este capítulo, incluyendo:

- Tasas basadas en la información que se le ofrece al usuario y la que éste selecciona.
- El *Error Medio Absoluto* y su *Desviación Estándar* basados en las diferencias de puntuación entre la información que se le ofrece al usuario y la que éste selecciona.
- La *Correlación* o similitud entre las elecciones del usuario y las propuestas informativas del sistema.
- La *R-Precisión* [Baeza, 1999] o Precisión en la posición R del orden, para cada sesión con el sistema.

3. **Determinación de los valores paramétricos más convenientes para el funcionamiento del sistema**. Para esta tarea se utilizarán los resultados obtenidos en los cuatro primeros experimentos propuestos, que se describirán en la sección 6.3.2. Los resultados de estos experimentos y los parámetros seleccionados se expondrán en las secciones 7.1, 7.2, 7.3 y 7.4 del capítulo siguiente.
4. **Estimación del funcionamiento del sistema con diferentes usuarios**, en base a las distintas medidas calculadas, y **prueba de un sistema alternativo de puntuación**. Para estas tareas se utilizarán los resultados obtenidos en los experimentos quinto y sexto propuestos, descritos en la sección 6.3.2, y cuyos resultados se expondrán en los apartados 7.5 y 7.6 del capítulo siguiente.

6.2. Metodología seguida.

Tras implementar el sistema descrito en el capítulo 5, utilizando el lenguaje C#, se procedió a su verificación y evaluación. Para ello se seleccionó la muestra objeto de estudio, formada por diversas fuentes de información a partir de las cuales se recuperan titulares de noticias actualizados. Estas fuentes de información seleccionadas se muestran en el Anexo II. Se ha procurado cierta variedad temática y que presentaran actualizaciones frecuentes. La mayoría de las fuentes de información seleccionadas emplean el idioma castellano, sin embargo se incluye un pequeño porcentaje de fuentes de información en idioma inglés.

En este punto, el sistema se puso a disposición de cualquier usuario de la Web en una página creada a tal efecto, comentada en el Anexo II, con la intención de seleccionar usuarios para su prueba.

Una vez diseñados los experimentos se preparó el sistema para cada uno de ellos y se llevaron a cabo. Los resultados obtenidos se almacenaron en una base de datos en formato XML[1] para su posterior análisis.

El número de sesiones de prueba realizadas para cada experimento ha sido de treinta, lo que no responde a un criterio arbitrario sino a una mera exigencia estadística. Para afirmar que el valor de la media aritmética de una distribución de valores representa fehacientemente a esta distribución se debe aplicar un contraste paramétrico conocido como la prueba *t de Student*, que exige ese número mínimo para su realización. Es por ello que todos los valores que se ofrecen como resultado de los experimentos han sido suficientemente contrastados por este método.

Para cada una de las diferentes sesiones de los experimentos se almacenará en la base de datos el nombre de cada titular seleccionado, su URL, el valor de la puntuación asignada al titular, la posición en que se ofrece al usuario y el ordinal en que el usuario lo selecciona. Un ejemplo de la base de datos, para un titular, se muestra en la figura 6.1.

```
<SESIÓN>
<Número_sesión>9</Número_sesión>
<Fecha_sesión>17/05/2005 1:50:50</Fecha_sesión>
<Número_titulares_elegidos>5</Número_titulares_elegidos>
<Número_titulares_ofrecidos>14</Número_titulares_ofrecidos>
<Titular_sesión>
<Título>Madrid 2012</Título>
<Url>http://www.ecuaderno.com/archives/000683.php</Url>
<Descripción>Un grupo de bloguers pone en marcha la bitácora colectiva Madrid 2012, cuyo objetivo fundamental es el apoyo a la candidatura de la ciudad de Madrid para la organización de los Juegos Olímpicos de 2012. Impulsan la iniciativa: Javier Morilla...</Descripción>
<Fecha>2005-05-17T09:12:49+01:00</Fecha>
<Valor_Puntuación>0.10293992241887566</Valor_Puntuación>
<Orden_elección>2</Orden_elección>
<Ofrecido_en_Posición>12</Ofrecido_en_Posición>
<Puntuación_Ideal>0.73849142501645082</Puntuación_Ideal>
<Error>0.6355515025975752</Error>
</Titular_sesión>
</SESIÓN>
```

Figura 6.1. Ejemplo de fragmento de la base de datos elaborada por sistema NectaRSS. La "<Puntuación_Ideal>" sería la que obtendría el titular si se encontrara en el lugar correspondiente al orden en que el usuario lo ha elegido.

[1] *XML es un lenguaje de marcado creado para organizar el contenido de un documento mediante etiquetas semánticas.*

Antes de las sesiones de prueba en cada uno de los casos considerados en los distintos experimentos se realizan dos sesiones de entrenamiento con el sistema, con el fin de inicializar el perfil de usuario correspondiente. Al final de cada experimento se analizan los resultados de la base de datos para verificarlos, analizarlos, contrastarlos y obtener conclusiones.

6.3. Estrategias de experimentación

Se distinguirán dos fases principales en la experimentación con el sistema propuesto, la primera para determinar los valores de ciertos parámetros iniciales, y la segunda para comprobar el comportamiento del algoritmo en diversos usuarios reales, contrastando los resultados de cada uno de ellos. Al comienzo de cada experimento, se dispone de un perfil de usuario vacío, el cual se irá elaborando y completando durante las distintas sesiones. Estas fases se describen más detalladamente a continuación:

- **Fase 1**. Consiste en **determinar diversos parámetros iniciales del sistema**. Así, se planteará la conveniencia o no de utilizar los resúmenes asociados a ciertos titulares para la elaboración del perfil de usuario, se probarán distintos valores en el *intervalo de vida* del *factor de olvido* definido en la fórmula (5.9), y se plantean distintas proporciones para la actualización del perfil definido en las fórmulas (5.7) y (5.15). Se realizarán distintas sesiones variando los parámetros. Al final de cada experimento se compararán los resultados, para comprobar si existen variaciones significativas, y cuál valor de entre los experimentados arroja mejores resultados. En esta fase los titulares se ofrecen desordenados aleatoriamente para no influir en las diferentes selecciones de la información. El usuario que experimentará con el sistema será el propio autor, y la elección de las noticias estará determinada por sus correspondientes preferencias temáticas, como cualquier otro usuario real. Una descripción más detallada de cada uno de los experimentos de esta fase se realiza en la sección 6.3.2.

- **Fase 2**. **Analizará el funcionamiento del sistema** utilizando los parámetros determinados en la fase 1. Para ello, se efectuarán distintas sesiones con distintos usuarios reales, contrastando los resultados para determinar su validez. En esta fase se le ofrecerán a cada usuario una lista de titulares ordenados por puntuación, y éste irá eligiendo los que le interesen. La cantidad de titulares ofrecida será tal que permita al usuario su visualización simultánea sin necesidad de realizar

desplazamientos verticales de la página. Se eligieron 15 usuarios para probar el sistema con el criterio de que sus intereses temáticos fuesen heterogéneos. También se probarán dos maneras distintas de puntuar la información. Una descripción más detallada de los usuarios experimentales y de los experimentos correspondientes a esta fase se encuentra en la sección 6.3.2.

6.3.1. Tratamiento de las palabras

Durante el funcionamiento del sistema, cada vez que se elija una noticia cualquiera, se analizarán los términos que aparezcan en el título y, si es el caso, los que aparezcan en la descripción o resumen de la noticia, mediante un sencillo analizador que irá extrayendo una a una todas las palabras.

En primer lugar se comprobará si el término extraído aporta alguna información o es una *palabra vacía*[2]. Para ello se comparará cada palabra extraída con un conjunto estándar de palabras vacías, formado por 561 palabras del castellano y 547 palabras inglesas de uso muy común. Estos conjuntos de palabras se han recopilado de diversas fuentes [Neu, 2005] y [Snow, 2005]. Antes de la comparación, cada palabra se convertirá completamente a minúsculas. Si dicha palabra pertenece al conjunto de palabras vacías se descarta. Si no es una palabra vacía se utilizará para ir formando el perfil de usuario, añadiéndola al mismo o modificando sus valores de perfil si ya está contenida.

El sistema no considerará números como palabras válidas pero se permitirá su inclusión en un conjunto de palabras que el sistema considerará necesariamente. También se podrá forzar al sistema para que excluya las palabras que se deseen.

Para evitar palabras erróneas o expresiones que pudieran escaparse a la acción del analizador, se efectuará una limpieza del perfil de usuario después de cada sesión, comparando cada uno de sus términos con un denso diccionario de castellano, formado por 650.817 palabras, y con otro menos denso pero también significativo, formado por 52.016 palabras inglesas. Ambos diccionarios se han confeccionado mediante la herramienta *ispell* [DATSI, 2005].

[2] *Existen palabras llenas, con significado independiente, y palabras vacías, aquellas que desempeñan funciones en compañía de otras. Una definición de palabra vacía es "una palabra sin significado por sí misma, como los artículos y preposiciones, también se denomina una palabra omitida". http://www.edym.com/books/esp/glosario.htm.*

6.3.2. Descripción de los experimentos

A continuación se exponen los distintos experimentos que se efectuarán con el sistema. Los cuatro primeros se corresponden con la primera fase, destinada a probar diversos parámetros del sistema, el quinto experimento irá destinado a analizar el comportamiento del algoritmo en distintos sujetos reales, para calibrar el sistema en el mundo real, y el último experimento comprobará si se producen diferencias significativas entre dos formas distintas de puntuar la información.

Los experimentos se realizarán en base a la información que se recupere en cada sesión procedente de las fuentes de información preseleccionadas, que se detallan en el Anexo II. En este contexto, cada sesión se corresponderá temporalmente con un día diferente, de esta manera puede decirse que se utilizarán los titulares de noticias de cada día. Para puntuar la información se utilizará inicialmente la medida del coseno propuesta en la sección 5.3 del capítulo 5. Es importante subrayar que los titulares que se empleen en el primer experimento se irán almacenando para ser utilizados en los siguientes, con el objeto de que en cada sesión correspondiente a cada experimento se dispongan exactamente de los mismos titulares de noticias.

- **Experimento 1. Con Resumen – Sin resumen (CRS)**

En este experimento se pretende evaluar cómo afecta al funcionamiento del sistema la consideración única del titular de cada noticia seleccionada para elaborar el perfil de usuario (ECON), respecto a la consideración del titular y de su resumen asociado, si éste lo posee (ESIN).

Para ello, se mantendrá una copia del sistema para cada estrategia y se realizarán exactamente las mismas selecciones de titulares en ambas. Finalmente, se analizarán los resultados, comparándolos para determinar si se encuentran diferencias significativas.

- **Experimento 2. Determinación del Intervalo de Vida (DIV)**

Se pretende probar ahora la utilización del *factor de olvido* definido en la fórmula (5.9). Se probará un rango de valores para su *intervalo de vida* y se analizarán los resultados obtenidos en cada uno de los casos, comparándolos para determinar cuál de los valores experimentados resulta más beneficioso para el sistema. Para este experimento el

sistema estará configurado con la mejor de las dos estrategias descritas en el experimento CRS anterior.

Los valores que se considerarán en el *intervalo de vida* son: 1, 2, 3, 4, 5, 6, 7, 10, 20 y 33. Esta muestra se fundamenta en la rápida tendencia a la unidad del *factor de olvido*, tal y como puede observarse en la figura 6.2.

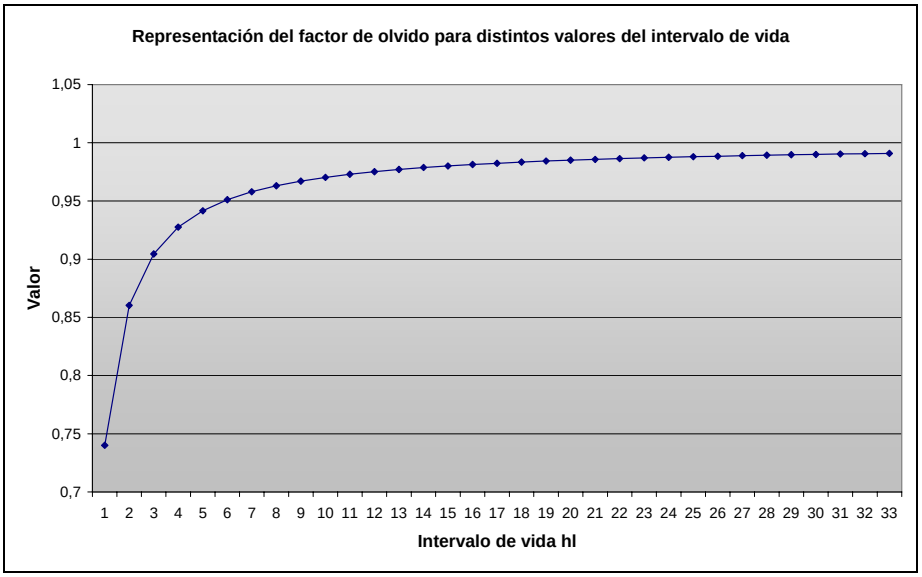


Figura 6.2. Representación gráfica del *factor de olvido*, según la fórmula (5.9), para distintos valores del intervalo de vida *hl*.

✦ Experimento 3. Importancia Relativa de los Perfiles (IRP)

En los experimentos anteriores, la estrategia seguida para calcular el perfil de usuario al finalizar cada sesión ha sido la de calcular el valor medio entre el perfil de sesión P_s y el perfil P acumulado en la sesión anterior. En este experimento se pretende probar con distintas importancias relativas para dichos perfiles, modificando sus parámetros multiplicadores tal y como se define en las fórmulas (5.7) y (5.15). Al final del experimento se analizarán los resultados ofrecidos por las distintas combinaciones consideradas para determinar cuál de ellas resulta más ventajosa para el sistema.

Se probarán los distintos pares de proporciones: (a=10, b=90), (a=20, b=80), (a=30, b=70), (a=40, b=60), (a=50, b=50), (a=60, b=40), (a=70, b=30), (a=80, b=20) y (a=90, b=10), abarcando uniformemente el intervalo [0, 100].

- **Experimento 4. Con Resumen – Sin resumen (2) (CRS2)**

Al igual que en el experimento 1, se pretende evaluar cómo afecta al funcionamiento del sistema la consideración única del titular de cada noticia seleccionada para elaborar el perfil de usuario respecto a la consideración del titular y de su resumen asociado, si éste lo posee. Este experimento será, por tanto, una repetición del experimento CRS pero ahora considerando los parámetros seleccionados en los experimentos 2 y 3. Con ello se pretenden reconfirmar las conclusiones obtenidas en el primer experimento.

Igualmente, se mantendrá una copia del sistema para cada estrategia y se realizarán exactamente las mismas selecciones de titulares en ambas. Finalmente, se analizarán los resultados, comparándolos para determinar si se encuentran diferencias significativas.

- **Experimento 5. Prueba del Algoritmo con diferentes Usuarios (PAU)**

Considerando los resultados obtenidos en los cuatro experimentos anteriores, se configurará un sistema tipo y se modificará para que presente al usuario una selección de titulares ordenados. Este sistema modificado será probado por diversos usuarios reales que deberán seleccionar cuantos titulares de noticias les resulten de interés en cada una de las sesiones. Al final del experimento se compararán los resultados que se hayan obtenido para cada uno de ellos, para determinar si el sistema posee un funcionamiento uniforme y válido. Se repetirá el experimento configurando el sistema para que presente al usuario una lista aleatoria de titulares, de entre los recuperados en cada sesión, con la intención de contrastar los resultados anteriores. El primer sub-experimento se denominará "ORDEN" y el segundo sub-experimento se denominará "AZAR".

En cada sesión del caso "ORDEN" se le presentarán al usuario una selección de 14 titulares ordenados por puntuación, cantidad elegida con la intención de presentar simultáneamente dichos titulares al usuario sin que éste deba realizar desplazamiento vertical alguno, según una resolución de pantalla concreta. Al repetir el experimento, la lista que se le presentará al usuario en el caso "AZAR" será de 14 titulares al azar de entre los recuperados en la sesión.

Se seleccionaron 15 usuarios con intereses heterogéneos, cada uno de los cuales debe efectuar 32 sesiones eligiendo la información de su interés de entre la ofrecida por el sistema. Las dos primeras sesiones serán de entrenamiento y las 30 sesiones restantes

proporcionarán los resultados que se exponen en el capítulo 7. Además, para comparar estos resultados, se realizarán otras 32 sesiones en las que cada usuario elegirá los titulares de su interés entre 14 ofrecidos al azar. Es necesario aclarar que en la primera sesión de cada sub-experimento, al no existir perfil de usuario alguno, se ofrecen todos los titulares recuperados.

Los usuarios fueron voluntarios anónimos que proporcionaron dos informaciones básicas: sus intereses preferidos, recogidos en la tabla 6.1, y los resultados de cada experimento.

USUARIO	INTERESES PREFERIDOS
1	Deportes y artículos en inglés
2	Internet, "blogosfera", "gadgets"
3	Tecnología, "gadgets", cine
4	Cine y noticias variadas
5	Deportes y cine
6	Sucesos en general y artículos en inglés
7	Internet, *software* y *hardware*
8	Artículos femeninos y "blogs"
9	Noticias, cine e Internet en general
10	Economía, noticias del Gobierno y generales
11	Deportes
12	Sucesos en general, política y coches
13	"Gadgets" y ciencia en general
14	Astronomía, ciencia e Internet en general
15	Cine y televisión

Tabla 6.1. Resumen de los intereses preferidos de los usuarios que efectúan el experimento 5.

- **Experimento 6. Probar Puntuación Alternativa (PPA)**

En este experimento, se selecciona al usuario que haya arrojado mejores resultados en el experimento PAU anterior y éste volverá a realizar 32 sesiones en el sistema configurado para puntuar la información según el coeficiente de Jaccard, propuesto como medida alternativa en la sección 5.3.1 del capítulo anterior.

En las 32 nuevas sesiones el usuario dispondrá de las mismas noticias que las empleadas para el experimento 5, donde se utilizó la medida del coseno para puntuar la información, al objeto de poder comparar sesión por sesión los resultados en ambos casos. Además también se le ofrecerán al usuario en cada sesión 14 titulares ordenados por puntuación para que escoja los que sean de su interés.

6.4. Medidas para la evaluación experimental del sistema

En este apartado se propondrán diversas medidas para cuantificar el funcionamiento del sistema propuesto, intentando reflejar desde diversos puntos de vista su ajuste a las preferencias del usuario. Cuanto más se acerque la recomendación de titulares ofrecida por el sistema a la elección de titulares que desea realizar el usuario en un momento determinado, mejor será dicha recomendación. Lo ideal es que el sistema mejore su funcionamiento cuantas más sesiones realice el usuario, ofreciendo cada vez mejores recomendaciones de titulares, y, por tanto, facilitando al usuario el acceso rápido a la información que más le interesa.

6.4.1. Tasas formadas por relaciones entre las variables observables

Durante el funcionamiento del sistema se monitorizarán las elecciones del usuario, almacenándose éstas en una base de datos para su posterior análisis, tal y como se mostró en el ejemplo de la figura 6.1. Determinaremos en esta sección las principales variables de interés que se observarán en los distintos experimentos; con éstas se definirán distintas medidas o tasas, cuyos resultados se analizarán después de cada experimento para evaluar el sistema.

Sea T el **conjunto de titulares de información** que se le ofrecen a un usuario en una sesión con el sistema. $E(T)$ será el **subconjunto de titulares que elige el usuario** en dicha sesión y $D(T)$ el **subconjunto de titulares con una puntuación asociada mayor que cero** en la sesión. Entonces, $E(T) \cap D(T)$ representará el **subconjunto de titulares con puntuación asociada mayor que cero elegidos por el usuario** en una sesión. En la figura 6.3 se muestran gráficamente éstos conjuntos. También podemos considerar dichos conjuntos como variables dependientes del sistema.

El número de titulares de una sesión será una cantidad variable que dependerá de las fuentes de información seleccionadas y de los titulares que devuelva cada una de ellas para esa sesión concreta. También se podría fijar una cantidad determinada de titulares para ofrecer al usuario, como sucede en el quinto experimento propuesto, descrito en el apartado 6.3.2. Así, una variable a considerar por el sistema será el **número de titulares que se le ofrecen al usuario** o *card(T)*.

En este conjunto de titulares ofrecidos podrá existir un porcentaje de titulares a los que el sistema haya otorgado una puntuación mayor que cero debido a su similitud con el

perfil de usuario, calculada según las fórmulas (5.16) y (5.17). El **número de titulares destacados con puntuación mayor que cero**, de entre los que se le ofrecen al usuario, será también una variable a considerar, su valor será *card(D(T))*.

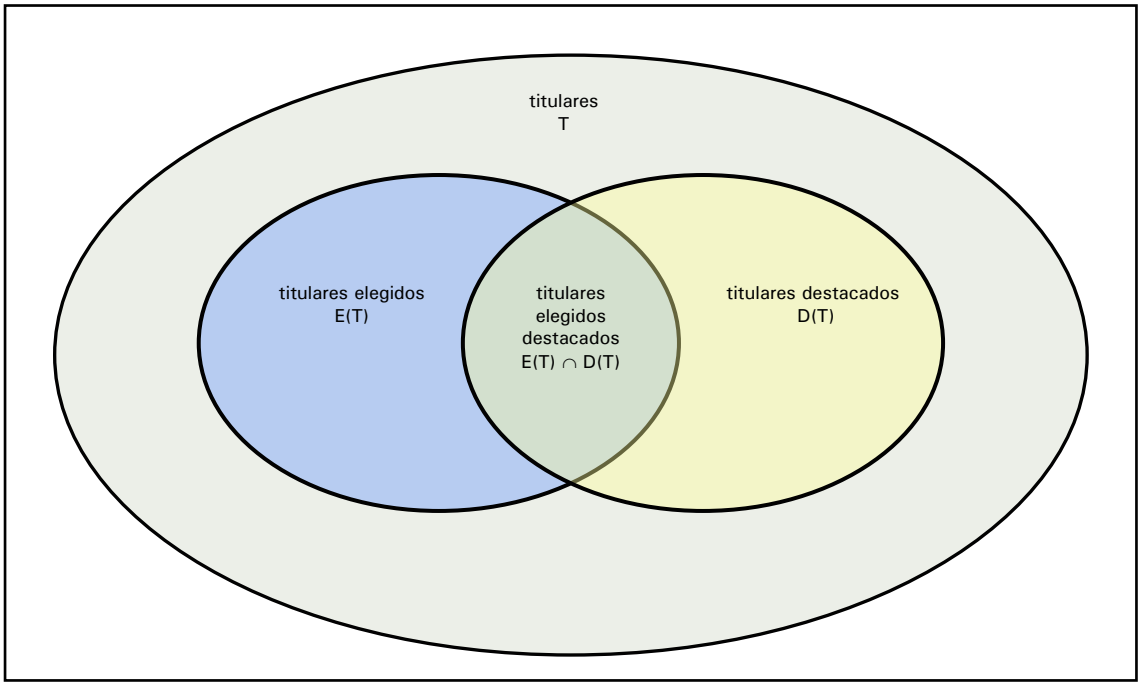


Figura 6.3. Relaciones consideradas entre los conjuntos de titulares, elegidos y destacados, comentados en la sección 6.4.1.

En cada sesión con el sistema el usuario elegirá los titulares que le interesen, por tanto, el **número de titulares que elija el usuario en una sesión** determinada será otra variable a considerar, siendo su valor el de *card(E(T))*.

Por otra parte, entre los titulares elegidos por el usuario en una sesión podrá existir un porcentaje de ellos que además tengan asociada una puntuación mayor que cero, tal cantidad variable será el **número de titulares destacados elegidos**, cuyo valor se corresponderá con *card(E(T) ∩ D(T))*.

Si relacionamos entre si estas variables podremos definir varias tasas de valor simple que nos ayuden a evaluar el sistema.

Así, para cuantificar el porcentaje de titulares elegidos por el usuario en una sesión respecto a los titulares que se le ofrecen en dicha sesión, se define la tasa C_p como:

$$C_P = \frac{E(T)}{T} \tag{6.1}$$

Valores bajos de esta tasa significarán que el usuario elige pocos titulares en la sesión, y valores altos de la tasa significarán que el usuario elige bastantes titulares.

Para calcular el porcentaje de titulares ofrecidos al usuario con puntuación asociada mayor que cero respecto al total de los titulares que se le ofrecen, se define la tasa C_R como:

$$C_R = \frac{D(T)}{T} \tag{6.2}$$

Valores altos de esta tasa significarán que se le ofrecen al usuario cantidades altas de titulares de noticias con puntuación calculada por el sistema mayor que cero respecto al total de titulares que se le presentan. Valores bajos pueden encontrarse en las sesiones iniciales debido a que el perfil de usuario se encuentra vacío o con poca información del usuario.

Para estudiar la relación entre el número titulares elegidos por el usuario con puntuación asociada mayor que cero y el total de titulares ofrecidos, se utilizará la tasa C_T definida como:

$$C_T = \frac{E(T) \cap D(T)}{T} \tag{6.3}$$

Si el valor de esta tasa es alto significará que el usuario elige bastantes titulares con puntuación asociada mayor que cero, y si el valor de la tasa es bajo es posible que los titulares puntuados por el sistema no sean los deseados por el usuario. Al igual que sucede con C_R, al inicio de los experimentos pueden esperarse valores bajos para esta tasa.

En la tabla 6.2 se muestra un resumen de estas relaciones de cardinalidad entre los conjuntos de titulares descritos, para obtener tasas que cuantifiquen ciertos aspectos del funcionamiento del sistema.

	titulares elegidos	titulares destacados	titulares elegidos destacados
titulares	Tasa C_P	Tasa C_R	Tasa C_T

Tabla 6.2. Tasas formadas a partir de las relaciones de cardinalidad entre los distintos conjuntos de titulares descritos en la sección 6.4.1. La relación se establece dividiendo la columna por la fila.

6.4.2. Puntuación media de un conjunto de titulares y puntuación media máxima

Como ya se ha comentado, cada titular ofrecido por el sistema tendrá asociada una puntuación, obtenida al calcular su similitud con el perfil de usuario, según las fórmulas (5.16) y (5.17). Así, aunque en la fase 1 de evaluación experimental del sistema los titulares se presentan al usuario desordenados aleatoriamente, para no influir en sus decisiones, éstos seguirán conservando un orden interno según esta puntuación calculada por el sistema.

En cada sesión se le ofrecerán al usuario cierta cantidad de titulares, o titulares ofrecidos, y éste elegirá los que le resulten interesantes, los titulares elegidos. Es posible calcular entonces un valor de puntuación medio $\overline{p(E(T))}$ para el conjunto de titulares escogidos por el usuario. Por otra parte, también se puede calcular un valor $\overline{p(T)}$ máximo que se obtendría cuando los N titulares escogidos por el usuario se correspondieran con los N primeros titulares en orden de puntuación ofrecidos por el sistema en una sesión determinada. Para cuantificar la relación entre el valor $\overline{p(E(T))}$ de los titulares elegidos por el usuario y el valor $\overline{p(T)}$ máximo se define la tasa C_D como:

$$C_D = \frac{\overline{p(E(T))}}{\overline{p_{max}(T)}} , \qquad (6.6)$$

en donde $\overline{p_{max}(T)}$ será la media de los N primeros valores de puntuación asociados a los N titulares con mayor puntuación de entre los ofrecidos al usuario, siendo N igual al número de titulares escogidos por el usuario.

6.4.3. El Error Medio Absoluto y la Desviación Estándar del Error

Estos criterios para evaluar el sistema son similares a los utilizados en [Moukas, 1996] y en [Lashkari, 1995]. Adoptando su notación, en nuestro sistema NectaRSS se asume que el conjunto $C = \{c_1, c_2, c_3, \ldots, c_N\}$ representa la puntuación de un subconjunto de titulares de noticias ofrecidos al usuario, y que el conjunto $F = \{f_1, f_2, f_3, \ldots, f_N\}$ representa la puntuación asociada a los titulares que selecciona el usuario. La idea es considerar la selección de titulares como una realimentación por parte del usuario. Entonces se define el conjunto error $E = \{e_1, e_2, e_3, \ldots, e_N\}$, y cada elemento de E se calculará según la expresión $e_i = c_i - f_i$, siendo N el número de titulares que escoge el usuario. De esta manera, consideramos las dos medidas siguientes:

- *Error Absoluto Medio*, cuanto menor sea su valor mejor será el rendimiento del sistema. Se calculará según la fórmula:

$$\left|\overline{E}\right| = \frac{\sum_{i=1}^{N} |e_i|}{N} \tag{6.7}$$

- *Desviación Estándar del Error*. Esta cantidad medirá la consistencia del rendimiento del algoritmo sobre el conjunto de datos. Cuanto menor sea su valor mejor será el algoritmo. Se definirá como:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N} \left(E - \overline{E}\right)^2}{N}} \tag{6.8}$$

6.4.4. La Correlación entre titulares

En [Moukas, 1996] se comparan las puntuaciones asignadas por el sistema *Amalthaea*, a ciertas páginas web, con las realimentaciones proporcionadas por el usuario. De manera análoga, compararemos las puntuaciones asignadas por nuestro sistema NectaRSS a los titulares de noticias con la realimentación implícita proporcionada por el usuario, al seleccionar titulares. El conjunto $C = \{c_1, c_2, c_3, \ldots, c_N\}$ representará la puntuación de un subconjunto de titulares de noticias ofrecidos al usuario, y el conjunto $F = \{f_1, f_2, f_3, \ldots, f_N\}$ representará la puntuación asociada a los titulares que selecciona el usuario. Así, se define la siguiente medida:

- *Coeficiente de Correlación*. Se pretende cuantificar la relación entre la puntuación de los titulares ofrecidos al usuario y la puntuación de los titulares que éste efectivamente escoge. Los valores de este coeficiente estarán comprendidos entre -1 y 1. Cuanto mayor sea este valor de la correlación, con valores más alejados de cero, mejor será el algoritmo [Hill, 1995]. Se definirá:

$$r = \frac{\frac{1}{N}\sum_{i=1}^{N}\left[(c_i - \bar{c})\cdot(f_i - \bar{f})\right]}{\sigma_c \cdot \sigma_f}, \qquad (6.9)$$

en donde σ_c y σ_f representan la desviación estándar de *C* y *F*, y el numerador de la expresión representa la covarianza.

6.4.5. La R-Precisión

Tal y como se expuso en la sección 3.2.5 del capítulo 3, de acuerdo con [Baeza, 1999], se generará un valor sumario simple para un conjunto de titulares ofrecidos en orden de puntuación, condición que sucede en los experimentos quinto y sexto propuestos. Para ello, se calculará la precisión en la posición R del orden, siendo R el número total de titulares relevantes de la sesión, en nuestro caso, el número de titulares que elija el usuario entre los ofrecidos por el sistema.

Así, por ejemplo, si R es igual a 6 y el usuario ha elegido tres titulares entre los seis primeros ofrecidos, se tendrá una *R-Precisión* de 0.5 al dividir los 3 titulares relevantes para el usuario entre los 6 elegidos en total. Esta medida se utilizará para observar el comportamiento del algoritmo para cada sesión *i* del experimento.

El valor de la *R-Precisión* podrá definirse en este caso como:

$$RP(i) = \frac{posR(E(T_i))}{card(E(T_i))}, \qquad (6.10)$$

en donde $posR(E(T_i))$ será el número de titulares elegidos entre los *R* primeros titulares ordenados ofrecidos al usuario en la sesión *i*, y el valor de $card(E(T_i))$ será igual al número total de titulares elegidos en dicha sesión.

6.5. Resumen

Se comienza el capítulo exponiendo el esquema general de la experimentación seguido para verificar nuestro objetivo: desarrollar un sistema de recomendación de información que la presente ordenada al usuario, en base a su perfil elaborado automáticamente, y que este sistema sea ventajoso para sus necesidades informativas. Para evaluar el funcionamiento del sistema se calcularán diversas medidas basadas fundamentalmente en las elecciones que realice el usuario y en la puntuación que el sistema haya otorgado a cada información.

Respecto a la metodología seguida, primero se implementó el sistema propuesto en el capítulo 5, para proceder posteriormente a su verificación y evaluación. Para ello se seleccionó una muestra de estudio compuesta por distintas fuentes de información y se realizaron diversos experimentos, analizando al final de cada uno de ellos los resultados obtenidos para valorar el funcionamiento del sistema propuesto.

En la experimentación se distinguen dos fases principales, la primera destinada a determinar empíricamente ciertos parámetros del sistema, y la segunda orientada a probar el funcionamiento del sistema con usuarios reales. Se llevaron a cabo seis experimentos, los cuatro primeros englobados en la fase 1, el quinto experimento destinado a probar el comportamiento del sistema con diferentes usuarios, lo que supone una calibración en el mundo real, y el sexto experimento donde se prueba una manera alternativa de puntuar la información. En la realización de todos estos experimentos se efectúa un tratamiento adecuado de las palabras o términos que irán conformando el perfil de usuario, eliminando las palabras vacías y contabilizando las que se vayan considerando.

Después de describir los experimentos, se proponen diversas tasas y medidas para cuantificar el funcionamiento del sistema, un grupo de ellas basadas en los conjuntos de titulares de noticias que se considerarán en cada sesión, tasas C_P, C_R y C_T, y otras relacionadas con la puntuación que el sistema asocia a los titulares en función de su similitud con el perfil de usuario. Entre éstas últimas se considera la tasa C_D, el *Error Absoluto Medio*, su *Desviación Estándar* y la *Correlación* entre titulares. Otra medida utilizada es la *R-Precisión*, o precisión en la posición R del orden, con la que puede observarse el comportamiento del sistema en cada una de las sesiones de los experimentos 5 y 6 mediante un valor simple.

Capítulo 7

RESULTADOS DE LOS EXPERIMENTOS

En este capítulo se presentan los distintos experimentos realizados, descritos en la sección 6.3.2 del capítulo anterior, indicando los parámetros a establecer y los valores numéricos obtenidos. Los resultados se representan gráficamente y se comentan, describiendo lo que se ve y a qué conclusiones se llegan por su análisis. La función del capítulo será, por tanto, comprobar la efectividad del sistema NectaRSS, analizando los valores obtenidos por las medidas que evalúan su funcionamiento.

En concreto, en la sección 7.1 se presentan los resultados obtenidos para el experimento **CRS**, destinado a determinar si es ventajosa la consideración de los resúmenes opcionales de las noticias para la elaboración del perfil de usuario. En la sección 7.2 se presentan los resultados del experimento **DIV** en el que se prueba el uso de un *factor de olvido* de los intereses del usuario. En la sección 7.3 se exponen los resultados para el experimento **IRP**, donde se prueban distintos porcentajes para el perfil de sesión y el perfil acumulado del usuario. En la sección 7.4 se muestra el experimento **CRS2**, análogo al CRS pero utilizando los valores de los parámetros determinados en los anteriores experimentos. En la sección 7.5 se prueba el sistema con diversos usuarios reales, experimento **PAU**, analizando el comportamiento del sistema desde perspectivas diferentes, y finalmente, en el experimento **PPA**, de la sección 7.6, se comparan dos maneras de puntuar la información: mediante la medida del coseno y mediante la medida de Jaccard.

7.1. Experimento 1. Con Resumen – Sin Resumen (CRS)

Este experimento, descrito en la sección 6.3.2, evalúa cómo afecta al funcionamiento del sistema la consideración o no de los resúmenes opcionales asociados a ciertas noticias para la elaboración del perfil de usuario. Para ello, se analizan los resultados obtenidos mientras se consideraban los resúmenes asociados, sub-experimento que se denota por **ECON**, y los resultados obtenidos sin su consideración, sub-experimento que se denota por **ESIN**.

Se utilizan las tasas C_P, C_R y C_T que se han definido en la sección 6.4.1 de esta tesis y que se resumen en la tabla 7.1. Además se utiliza la tasa C_D, definida en la sección 6.4.2, que se basa en el valor de puntuación que el sistema asigna a los titulares.

Para comparar los resultados de ambos sub-experimentos, en la tabla 7.2 se muestran los valores medios de las tasas calculadas en cada una de las 30 sesiones experimentales y se representan gráficamente estos valores medios junto con su desviación estándar en los gráficos de las figuras 7.1, 7.2 y 7.3.

	titulares elegidos	titulares destacados	titulares elegidos destacados
titulares	Tasa C_P	Tasa C_R	Tasa C_T

Tabla 7.1. Tasas formadas a partir de las relaciones de cardinalidad entre los distintos conjuntos de titulares considerados. La relación se establece dividiendo la columna por la fila.

Caso	**Experimento CRS – Valores medios de las tasas calculadas**			
	C_P	C_R	C_T	C_D
ECON	0.2312	0.6292	0.1572	0.5646
ESIN	0.2312	0.4248	0.1269	0.5192

Tabla 7.2. Valores medios obtenidos para las distintas tasas consideradas en el experimento 1 después de 30 sesiones experimentales.

En la tasa C_P, definida por la fórmula 6.1, se obtienen valores idénticos en ambos casos considerados, ECON y ESIN, debido a que se repite la misma selección de titulares, por ello no se tendrá en cuenta. Para la tasa C_R, definida en la fórmula (6.2), se comprueba que se obtienen mayores valores para el caso ECON, tal y como puede apreciarse en la figura 7.1. Esta es una consecuencia lógica ya que al considerar los resúmenes asociados a los titulares de noticias el perfil de usuario se enriquece con muchas más palabras que si no se consideran éstos. Al finalizar la sesión experimental 30 se obtuvieron 5342 términos en el perfil asociado al caso ECON en contraste con la cantidad de 1248 términos para el perfil asociado al caso ESIN. De esta manera, se obtienen más titulares de noticias con alguna puntuación pues será más probable que en ellos se encuentre alguna de las palabras del perfil con más términos. Por el mismo motivo se observan mayores valores medios en el caso ECON para la tasa C_T, definida en la fórmula (6.3), y representada en la figura 7.2.

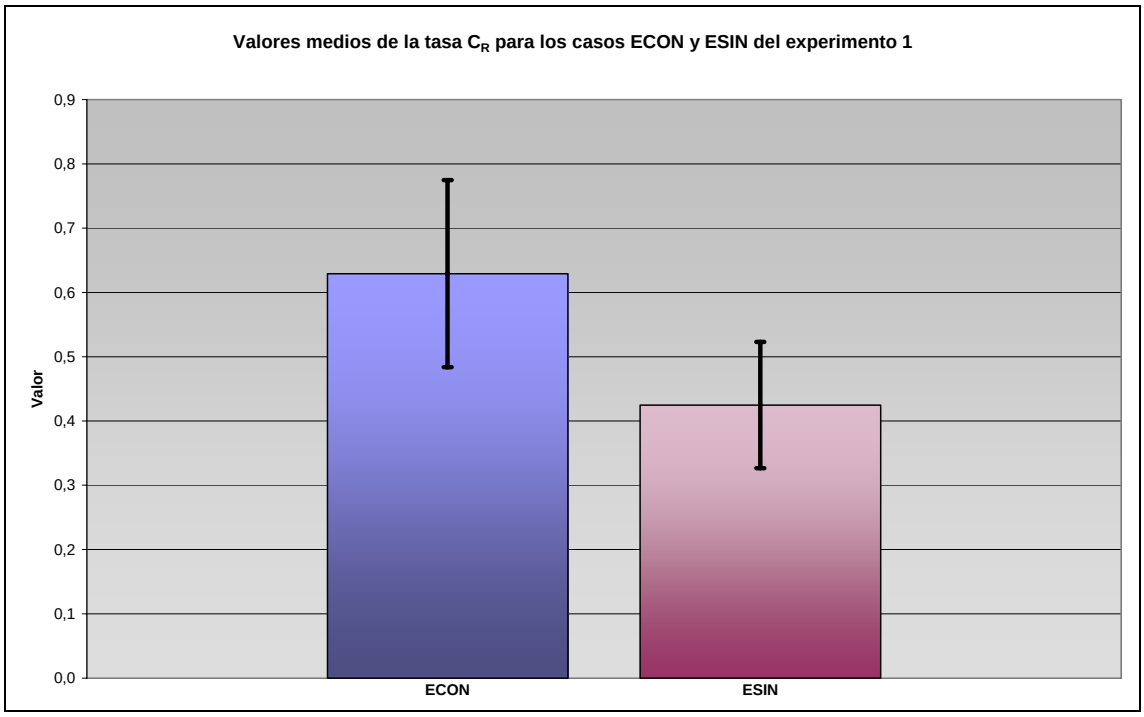


Figura 7.1. Comparación de los valores medios obtenidos por la tasa C_R calculada cuando el sistema utiliza los resúmenes asociados a los titulares (ECON) respecto a cuando no se utilizan (ESIN). Se representa además su desviación estándar. Se observa un mayor valor de la tasa para el caso ECON que para el caso ESIN.

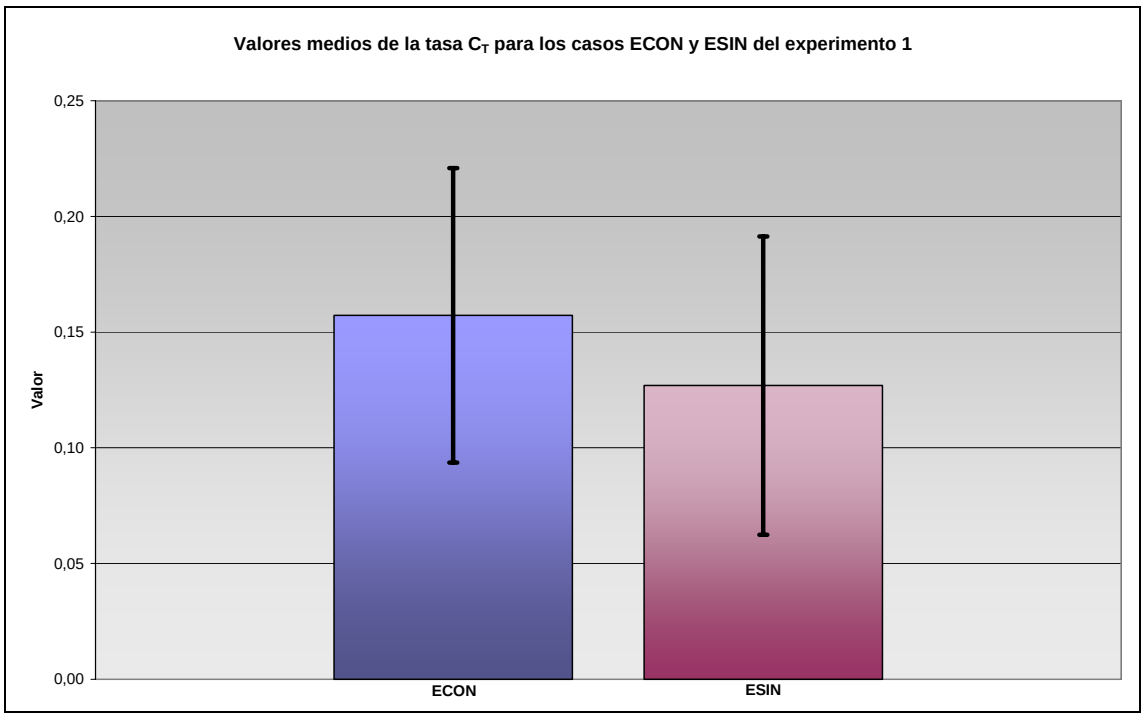


Figura 7.2. Comparación de los valores medios obtenidos por la tasa C_T calculada cuando el sistema utiliza los resúmenes asociados a los titulares (ECON) respecto a cuando no se utilizan (ESIN). Se representa además su desviación estándar. Se observa un mayor valor de la tasa para el caso ECON que para el caso ESIN.

Para la tasa C_D, fórmula (6.4), se observa un valor medio superior para el caso ECON, como puede verse en la figura 7.3. Esta tasa C_D tiene una naturaleza diferente a las anteriores, ya que lo que ahora se está comparando, en ambos casos, es la puntuación media asociada a la información que selecciona el usuario respecto a la puntuación media máxima ideal que se conseguiría si éste seleccionara la información mejor puntuada, tal y como se define en la fórmula (6.4).

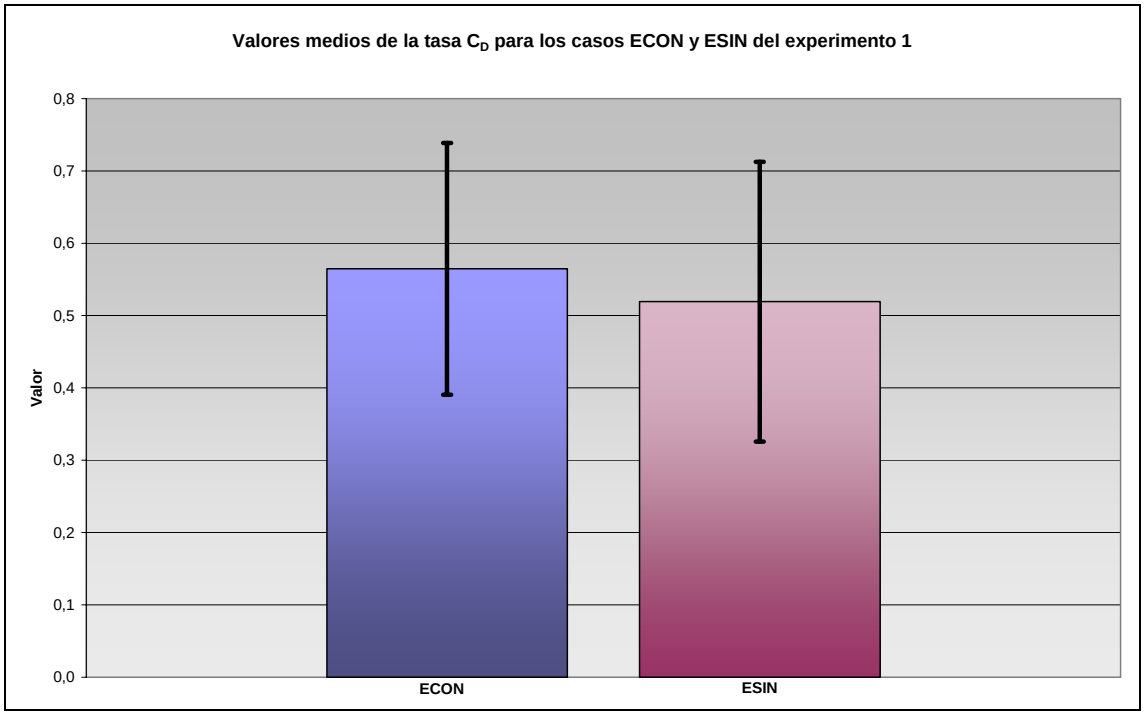


Figura 7.3. Comparación de los valores medios obtenidos por la tasa C_D, calculada cuando el sistema utiliza los resúmenes asociados a los titulares (ECON) respecto a cuando no se utilizan (ESIN). Junto a cada valor medio se muestra su desviación estándar. El valor medio para el caso ECON es mayor.

Para comprobar si existen diferencias significativas entre los dos tratamientos del perfil de usuario, ECON y ESIN, se utilizará la prueba *t-Student* con las dos series de datos obtenidas para la tasa C_D a lo largo de todas las sesiones consideradas. Se aplicará la prueba estadística de *Kolmogorov-Smirnov* a cada uno de los grupos de datos para comprobar su normalidad, condición indispensable para aplicar el *test de Student*.

Los resultados obtenidos para la prueba se muestran en la tabla 7.3. El resultado de 0.0025 obtenido para el *test de Student*, con $t = 3.312$ y 29 grados de libertad, se considera

muy significativo. Por lo tanto se considera que si existen diferencias significativas entre el caso ECON y el caso ESIN, según la tasa C_D.

Parámetros	**ECON**	**ESIN**
Media	0.5646	0.5192
Muestra	30	30
Desviación Estándar	0.1740	0.1934
P del test de Normalidad	0.0572	>0.10
Test t-Student (2 colas)	**0.0025**	

Tabla 7.3. Resultados estadísticos obtenidos para los grupos de valores de los casos ECON y ESIN destacando el valor de la prueba *t -Student* para la tasa C_D.

Comprobando los distintos resultados, cabe preguntarse qué es lo que importa en la práctica, que el usuario disponga de mayor número de titulares de noticias puntuados, hecho reflejado en la tasa C_R, con lo que es más probable que elija precisamente esos titulares, hecho que se refleja en la tasa C_T, o que el usuario vaya eligiendo los titulares con mejor puntuación. En el primer caso, la cantidad de titulares puntuados va a depender directamente del tamaño en palabras del perfil de usuario, así cuanto más se utilice el sistema mayor será dicho perfil y mayor cantidad de titulares se puntuarán. Las tasas C_R y C_T nos pueden dar una idea sobre todo de la densidad del perfil de usuario pero no ofrecerán demasiada información acerca de la calidad de las noticias que se le proporcionan al usuario. Por supuesto, los titulares puntuados contendrán términos del perfil y se puede esperar que sean de interés para dicho usuario pero las palabras pueden variar de significado según el contexto y por ello no está garantizado que todo titular puntuado sea de interés.

En el segundo caso, la tasa C_D debe reflejar cuándo se realizan selecciones de titulares con buena puntuación; esto implica, por una parte, que el usuario ha elegido las noticias mejor puntuadas por el sistema, es decir que la puntuación otorgada por el sistema a esas noticias resulta válida para ese usuario, y, por otra parte, si un usuario elige una noticia bien puntuada es más probable que esa noticia sea realmente de su interés puesto que algunos o todos los términos del titular deben encontrarse bien valorados en su perfil.

Por ello, la tasa C_D nos proporcionará más información acerca del funcionamiento del sistema, resultando además bastante más independiente respecto al tamaño en palabras del perfil de usuario que el resto tasas consideradas, así se tendrán en cuenta especialmente sus resultados.

Se puede afirmar que se requiere mayor esfuerzo computacional para manipular el perfil de usuario elaborado considerando los resúmenes opcionales de las noticias, estrategia ECON, respecto a su no consideración, estrategia ESIN. Esto se debe a la mayor cantidad de términos que formarán parte del perfil en el primer caso. Sin embargo, la mayor cantidad de palabras consideradas en un perfil permite puntuar mayor número de titulares de noticias, tal y como se ha comprobado en las tasas C_R y C_T analizadas, lo que a su vez conduce a que el usuario acabe eligiendo más noticias con puntuación mayor que cero.

Asimismo, se observa un mejor valor medio para la tasa C_D en la estrategia ECON respecto a la estrategia ESIN, y dada la representatividad de esta tasa sobre el funcionamiento del algoritmo, se comprobó mediante el test *t-Student* que sí existían diferencias significativas entre ambas estrategias. Por tanto, se considerará como mejor estrategia para el sistema propuesto la consideración de los resúmenes opcionales de las noticias en la elaboración incremental y automática del perfil de usuario basado en su historial de navegación. Esta característica se mantendrá durante los siguientes experimentos.

7.2. Experimento 2. Determinación del intervalo de vida (DIV)

En este experimento, descrito en la sección 6.3.2, se prueba el uso de un *factor de olvido*, fórmula (5.9), utilizando distintos valores para su *intervalo de vida hl*. Para ello, se realizaron 30 sesiones experimentales considerando distintos valores para *hl*: 1, 2, 3, 4, 5, 6, 7, 10, 20 y 33. La muestra se fundamenta en la rápida tendencia a la unidad del *factor de olvido*, como puede observarse en la figura 6.2 del capítulo 6. Además se considera el caso en que el sistema no utiliza ningún factor de olvido, denotando los resultados con *SINfol*.

Se empleará como criterio principal de análisis la tasa C_D, ya que el resto de tasas consideradas tomarán valores totalmente idénticos en la mayoría de los casos debido a que en cada sesión se realizan exactamente las mismas elecciones de titulares para cada valor de *hl*, sin que ello suponga variación alguna en el tamaño del perfil de usuario a diferencia del experimento 1 anterior.

Los valores medios obtenidos para la tasa C_D en los distintos casos considerados, después de 30 sesiones experimentales con el sistema, se muestran en la tabla 7.4. En la figura 7.4 se representan estos valores junto con su desviación estándar.

Experimento 2 – Valor medio de la tasa C_D										
hl = 1	*hl = 2*	*hl = 3*	*hl = 4*	*hl = 5*	*hl = 6*	*hl = 7*	*hl = 10*	*hl = 20*	*hl = 33*	*SINfol*
0.4882	0.5336	0.5510	0.5616	0.5650	0.5670	0.5681	0.5654	0.5648	0.5673	0.5652

Tabla 7.4. Valores medios obtenidos para la tasa C_D en el experimento 2 después de 30 sesiones experimentales con el sistema, con distintos valores para el *intervalo de vida hl* y sin considerar un *factor de olvido SINfol*.

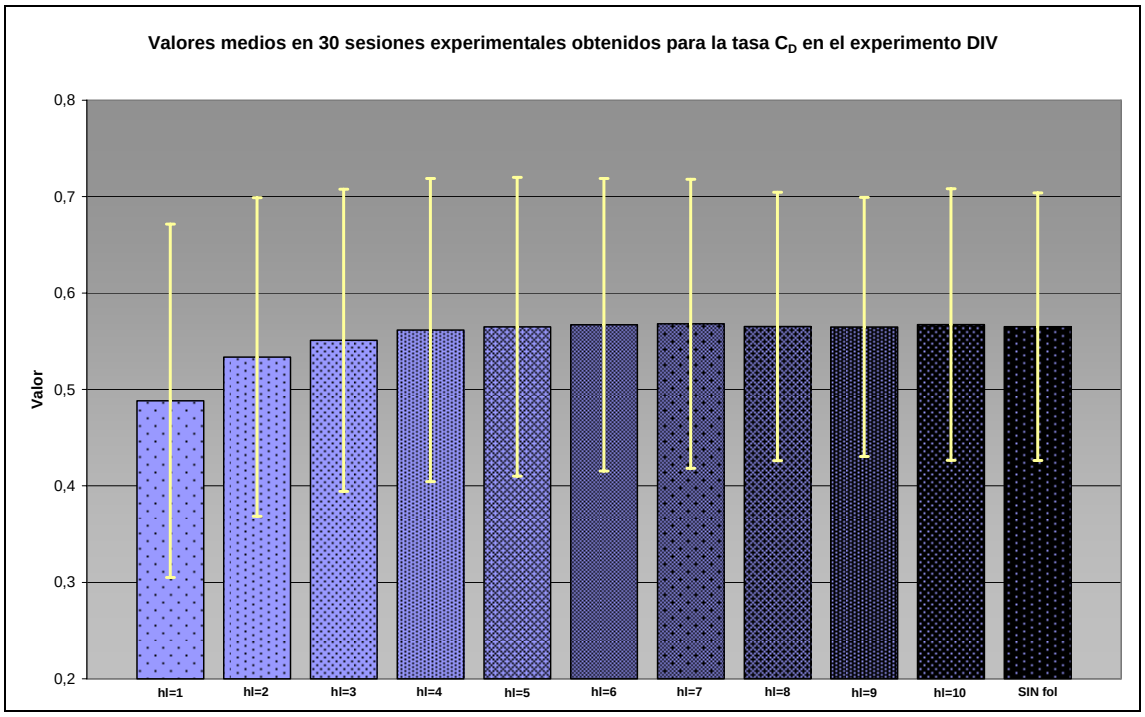


Figura 7.4. Comparación de valores medios obtenidos en la tasa C_D para distintos valores del *intervalo de vida hl*. Se muestra además el valor medio obtenido cuando no se utiliza una función de olvido *SINfol*. Se observan valores medios de la tasa muy similares a partir de *hl=4* y para el caso *SINfol*.

Se observa que los resultados obtenidos por la tasa C_D para los distintos valores del intervalo de vida *hl* son bastante similares. La mejor media entre las series de datos se ha calculado para un intervalo de vida "hl=7". Esta media, sin embargo, resulta similar a la obtenida en el caso en el que no se considera ningún factor de olvido, *SINfol*. Para

comprobar si existen diferencias significativas entre ambos casos, se aplicará a las dos series de datos la prueba *t-Student*. Se usará la prueba estadística de *Kolmogorov-Smirnov* con cada uno de los grupos de datos para comprobar su normalidad, condición indispensable para aplicar la prueba *t- Student*.

Los resultados obtenidos para la prueba se muestran en la tabla 7.5. El resultado de 0.6292 obtenido para el *test de Student*, con $t = 0.4880$ y 29 grados de libertad, se considera no significativo. Por lo tanto se considera que no existen diferencias significativas entre la consideración de un factor de olvido, con intervalo de vida "hl= 7", y la no consideración de tal factor de olvido, según la tasa C_D.

Parámetros	**Factor de olvido con *hl = 7***	**Sin factor de olvido**
Media	0.5681	0.5652
Muestra	30	30
Desviación Estándar	0.1500	0.1387
P del test de Normalidad	>0.10	>0.10
Test t-Student (2 colas)	**0.6292**	

Tabla 7.5. Resultados estadísticos obtenidos para la serie de datos cuando se considera un *factor de olvido* con *intervalo de vida hl= 7* y la serie de datos cuando no se considera un *factor de olvido*, destacando el valor de la prueba *t -Student* para la tasa C_D.

Teniendo en cuenta el resultado de la prueba *t-Student*, que indica la no existencia de diferencias significativas para los casos considerados, la adopción de un factor de olvido con un intervalo de vida *hl* = 7 no debe variar significativamente los resultados del sistema pero si que supone el cálculo de mayor número de operaciones, pues al final de cada sesión se deberán actualizar la mayoría de los términos del perfil de usuario con dicho factor. Es por ello que se optará por la opción más simple, la de no considerar un *factor de olvido* en el proceso incremental de elaboración del perfil de usuario. Esta característica se mantendrá durante los siguientes experimentos.

7.3. Experimento 3. Importancia Relativa de los Perfiles (IRP)

Este experimento, descrito en la sección 6.3.2 de la tesis, evalúa cómo afecta en el rendimiento del sistema la consideración de distintas proporciones para el cálculo del perfil de usuario acumulado al final de cada sesión, tal y como se describe en la fórmula (5.15). Las proporciones vienen dadas por los parámetros *a* y *b*. Un valor mayor para el parámetro *a* enfatizará el perfil acumulado y un valor mayor para el parámetro *b* enfatizará el perfil elaborado por la sesión en curso.

Así, se han probado distintos pares de proporciones para dichos parámetros durante 30 sesiones experimentales del sistema: (*a*=10, *b*=90), (*a*=20, *b*=80), (*a*=30, *b*=70), (*a*=40, *b*=60), (*a*=50, *b*=50), (*a*=60, *b*=40), (*a*=70, *b*=30), (*a*=80, *b*=20) y (*a*=90, *b*=10).

Como en el experimento 2, se ha utilizado como criterio principal de evaluación la tasa C_D. El resto de tasas consideradas tomarán valores totalmente idénticos en la mayoría de los casos puesto que en cada sesión se realizan exactamente las mismas elecciones de titulares para cada par de valores considerados, sin que ello suponga variación alguna en el tamaño del perfil de usuario. Los valores medios obtenidos para esta tasa C_D en los distintos casos considerados, después de 30 sesiones experimentales, se muestran en la tabla 7.6. En la figura 7.5 se representan estos valores junto con su desviación estándar.

Experimento 3 – Valor medio de la tasa C_D considerando distintos pares (a, b)								
(10,90)	*(20,80)*	*(30,70)*	*(40,60)*	*(50,50)*	*(60,40)*	*(70,30)*	*(80,20)*	*(90,10)*
0.6186	0.6240	0.6283	0.6306	0.6319	0.6315	0.6286	0.6223	0.6123

Tabla 7.6. Valores medios obtenidos para la tasa C_D en el experimento 3 después de 30 sesiones experimentales con el sistema, con distintos pares de valores para los parámetros *a* y *b*.

En la figura 7.5 se observan valores bastante cercanos de la tasa C_D para todos los casos considerados. Sin embargo, la mejor media se ha calculado para el par (*a*=50, *b*=50). La consideración de cualquier otro par de valores de entre los experimentados no tiene ningún efecto en el número de operaciones necesarias para calcular el perfil de usuario después de cada sesión. Por ello, se escogerá el par de valores que ofrece la mejor media para el coeficiente C_D, lo que indicará más selecciones de titulares con buena puntuación, aún cuando la media siendo irrelevante la aplicación de un test *t-Student* para determinar si existen diferencias significativas entre las distintas series de valores.

Así, en los siguientes experimentos se utilizará la proporción 50 para ambos parámetros *a* y *b*, lo que efectivamente equivale a calcular la media entre el perfil de sesión P_s y el perfil acumulado *P*, tal y como se define en la fórmula (5.15).

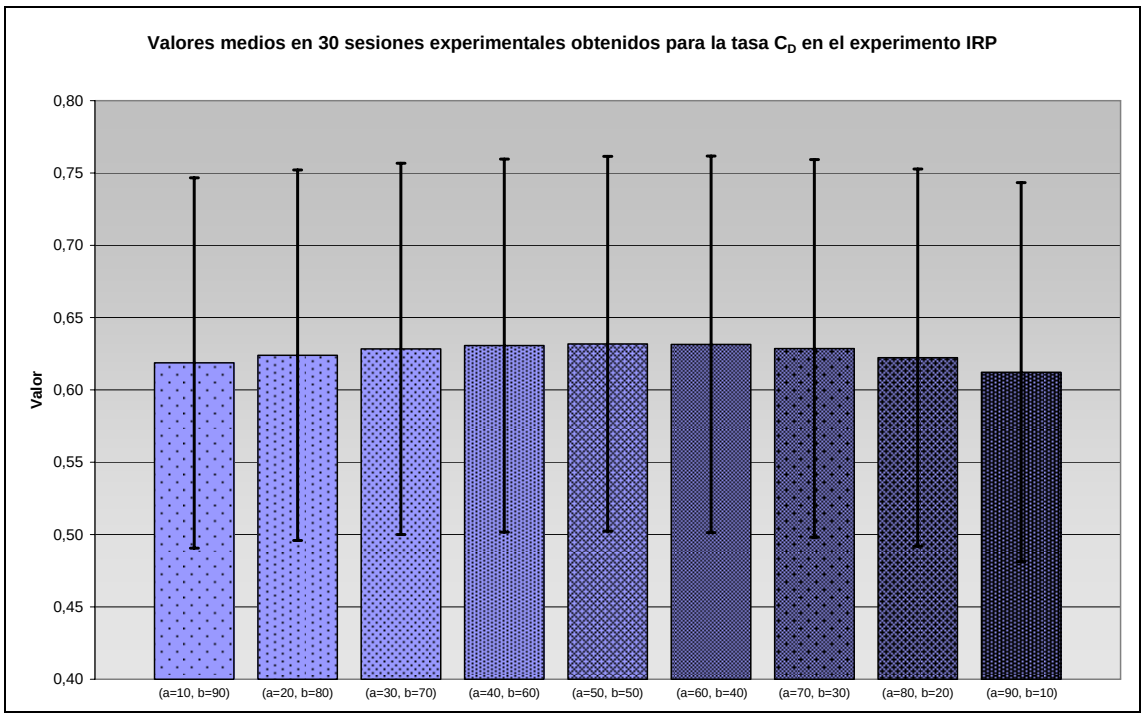


Figura 7.5. Valores medios de la tasa C_D para distintos pares de proporciones en el cálculo del perfil de usuario después de 30 sesiones experimentales con el sistema. La media más elevada se obtiene para el par (a=50, b=50). Se indica además la desviación estándar para cada media.

7.4. Experimento 4. Con Resumen – Sin Resumen (2) (CRS2)

Este experimento, expuesto en la sección 6.3.2, pretende evaluar nuevamente cómo afecta al sistema la consideración o no de los resúmenes opcionales de las noticias para la elaboración del perfil de usuario. La intención es confirmar los resultados obtenidos en el experimento 1. Se considera importante esta confirmación de las conclusiones debido a las diferentes consecuencias que sobre el perfil de usuario tienen ambos casos considerados.

Se utilizarán los valores de los parámetros determinados experimentalmente, según los experimentos 2 y 3, que son la no consideración de un factor de olvido y la proporción 50 para los parámetros *a* y *b* de la fórmula (5.15).

Se analizarán los resultados calculados para la tasa C_D durante 30 sesiones experimentales con el sistema, considerando el caso que denotaremos por **ECON2**,

cuando se tienen en cuenta los resúmenes opcionales, y el caso **ESIN2** cuando no se utilizan estos resúmenes en la elaboración del perfil de usuario. Esta tasa es la que se muestra más independiente respecto a variaciones en tamaño del perfil, como ya se ha observado en el experimento 1.

A diferencia de los experimentos anteriores donde se obtuvieron valores medios, en este experimento se va a considerar la evolución de la tasa C_D a lo largo de las 30 sesiones para comparar su tendencia en cada caso. Así en la figura 7.6 se muestran los resultados obtenidos por dicha tasa en cada una de las sesiones para los dos casos considerados "ECON2" y "ESIN2", junto con la línea de tendencia de cada uno, "Lineal(ECON2)" y "Lineal(ESIN2)". Estas líneas de tendencia se calculan por el método de mínimos cuadrados, según la ecuación $y = mx + b$, donde m es la pendiente y b es la intersección.

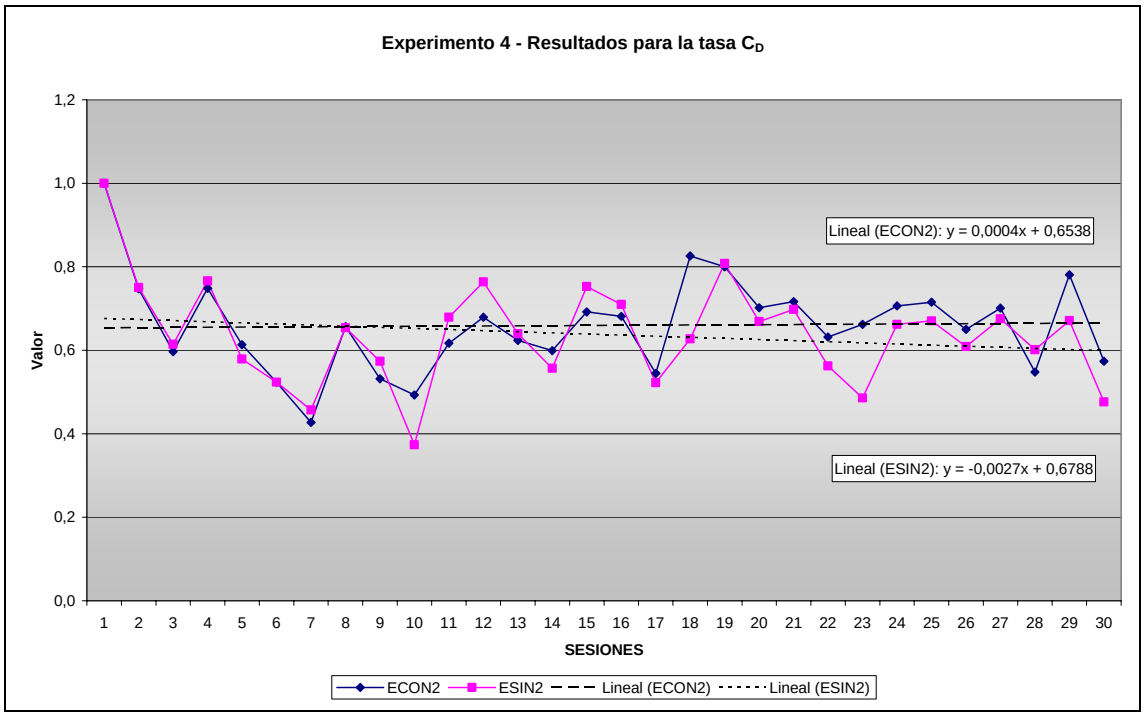


Figura 7.6. Resultados obtenidos para la tasa C_D durante 30 sesiones experimentales, considerando los resúmenes opcionales de las noticias "ECON2" y sin considerarlos "ESIN2". Se observa que la línea de tendencia correspondiente al caso "ECON2", "Lineal(ECON2)", es más favorable.

Observamos que entre las dos líneas de tendencia de la figura 7.6, correspondientes a las series de datos "ECON2" y "ESIN2", resulta más favorable la correspondiente a la serie "ECON2", "Lineal(ECON2)", debido a que su pendiente es positiva, frente a la

pendiente de "Lineal(ESIN2)" con valor negativo que indicaría una tendencia negativa a lo largo de las sesiones para este segundo caso.

Estos resultados nos confirman las conclusiones obtenidas para el experimento 1, donde se afirmaba mejor la estrategia en la que se considera el resumen opcional de las noticias para ir elaborando el perfil de usuario. Es decir, se tendrán en cuenta los términos de los resúmenes opcionales asociados a los titulares que seleccione el usuario en cada sesión con el sistema.

7.5. Experimento 5. Probar Algoritmo con diferentes Usuarios (PAU)

En este experimento se evaluará el funcionamiento del sistema propuesto con diferentes usuarios. Puede considerarse como una calibración del método en el "mundo real". Los resultados nos darán una idea de la eficacia del sistema NectaRSS y ayudarán a confirmar su adecuado funcionamiento como sistema de recomendación de información para distintos usuarios.

Partiendo de los resultados obtenidos en los cuatro experimentos anteriores, se configuró un sistema tipo con los mejores valores experimentales y se modificó para que presentara al usuario, en cada sesión, una selección de 14 titulares ordenados por puntuación, cantidad elegida en base a la intención de presentar simultáneamente dichos titulares al usuario según una resolución de pantalla concreta sin que éste deba realizar desplazamiento vertical alguno.

Cada uno de los 15 usuarios voluntarios efectuó 2 sesiones de entrenamiento y 30 sesiones experimentales, eligiendo la información de su interés de entre la ofrecida por el sistema. En las sesiones experimentales el sistema sigue elaborando incrementalmente el perfil de cada usuario. Los intereses de estos usuarios son los mostrados en la tabla 6.1 del capítulo anterior. Además, para comparar los resultados, los participantes realizaron otras 30 sesiones de prueba en las que cada usuario tenía que elegir los titulares de su interés entre 14 ofrecidos al azar. Es necesario aclarar que en la primera sesión de cada sub-experimento, al no existir perfil de usuario alguno, se ofrecen todos los titulares.

Los resultados obtenidos para las distintas tasas y medidas consideradas se recogen en las tablas y gráficos de las secciones siguientes.

7.5.1. Comparación de Tasas

En la tabla 7.7 se recogen los valores numéricos obtenidos para las tasas C_T y C_D en la sesión experimental 30 del experimento para los 15 usuarios. En las figuras 7.7 y 7.9 se representan estos resultados. También se han calculado los valores medios para estas tasas en las 30 sesiones experimentales. Dichos valores se exponen en la tabla 7.8 y se representan en las figuras 7.8 y 7.10. En todas las tablas y gráficos se denota por **ORDEN** a la serie asociada al sub-experimento en el que se le ofrece al usuario una lista ordenada de titulares según su puntuación, y se denota **AZAR** a la serie asociada al sub-experimento en el que se le ofrece al usuario una lista de titulares al azar de entre los recuperados en la sesión.

La tasa C_R no se ha considerado pues ofrece el valor 1 en todos los usuarios para el caso "ORDEN". Esto es debido a que en la sesión 30 todos los titulares aparecen como destacados para dicho caso. Por el mismo motivo no ha considerado la tasa C_P que ofrecerá los mismos resultados que la tasa C_T para el caso "ORDEN".

tasa	**Experimento 5 – Valores obtenidos para C_T y C_D en la sesión 30 por 15 usuarios**														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
C_T ORDEN	0.714	0.286	0.429	0.571	0.714	0.357	0.357	0.500	0.643	0.643	0.714	0.571	0.500	0.500	0.357
C_T AZAR	0.286	0.143	0.071	0.214	0.143	0.286	0.143	0.143	0.143	0.286	0.143	0.214	0.071	0.143	0.071
C_D ORDEN	0.936	0.876	0.939	0.866	0.890	0.817	0.847	0.838	0.972	0.871	0.974	0.852	0.822	0.915	0.927
C_D AZAR	0.725	0.426	0.097	0.238	0.489	0.580	0.634	0.241	0.479	0.250	0.536	0.709	0.635	0.535	0.022

Tabla 7.7. Valores obtenidos para las tasas C_T y C_D por los quince usuarios experimentales en la sesión 30, en los casos "ORDEN" y "AZAR".

tasa	**Experimento 5 – Valores medios obtenidos para C_T y C_D por 15 usuarios**														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
C_T ORDEN	0.726	0.300	0.414	0.50	0.743	0.402	0.412	0.340	0.564	0.574	0.757	0.495	0.338	0.355	0.267
C_T AZAR	0.138	0.062	0.093	0.233	0.195	0.198	0.095	0.100	0.179	0.183	0.136	0.193	0.086	0.067	0.062
C_D ORDEN	0.876	0.773	0.901	0.849	0.915	0.756	0.871	0.691	0.872	0.853	0.918	0.799	0.696	0.773	0.845
C_D AZAR	0.265	0.222	0.361	0.531	0.310	0.615	0.360	0.287	0.430	0.383	0.390	0.610	0.310	0.262	0.298

Tabla 7.8. Valores medios obtenidos para las tasas C_T y C_D por los quince usuarios en las 30 sesiones experimentales, distinguiendo los casos "ORDEN" y "AZAR".

Observando el gráfico de la figura 7.7, donde se representan los valores obtenidos por 15 usuarios para la tasa C_T en la sesión experimental 30, y el gráfico de la figura 7.8, donde se representan los valores medios calculados para dicha tasa en las 30 sesiones experimentales, vemos que para todos los usuarios se han obtenido mayores valores para el caso "ORDEN", que ofrece los titulares ordenados por puntuación, respecto al caso "AZAR", que ofrece los titulares al azar a cada usuario. Esto significa que en el caso "ORDEN" el usuario elige más titulares de noticias que el sistema ha puntuado. Es decir, mayor cantidad de titulares que el sistema evalúa como interesantes, según el perfil del usuario, serán efectivamente interesantes para tal usuario puesto que los selecciona. Así, podemos afirmar que el sistema ofrece en el caso "ORDEN" mejores titulares según el interés del usuario.

Para cuantificar la mejora del sistema en el caso "ORDEN" respecto al caso "AZAR" se compararán los valores medios de la tasa C_T obtenidos en ambos casos, tanto para la sesión 30 como cuando se consideran las medias de las 30 sesiones experimentales.

El valor medio de la tasa C_T para todos usuarios en la sesión experimental 30 es de 0.524 en el caso "ORDEN" y de 0.167 en el caso "AZAR". En la sesión 30 se constata, por tanto, un incremento de valor medio de la tasa C_T de 314% para el caso "ORDEN" respecto al caso "AZAR".

Asimismo, se tiene que el valor medio de la tasa C_T para todos los usuarios en las 30 sesiones experimentales es de 0.479 en el caso "ORDEN" y de 0.135 en el caso "AZAR". Entonces se constata que el valor medio de C_T en las 30 sesiones es un 355% mayor en el caso "ORDEN" que el correspondiente al caso "AZAR".

Observando el gráfico de la figura 7.9, donde se representan los valores obtenidos por 15 usuarios para la tasa C_D en la sesión experimental 30, y el gráfico de la figura 7.10, donde se representan los valores medios calculados para dicha tasa, vemos que para todos los usuarios se han obtenido mayores valores para el caso "ORDEN", que ofrece los titulares ordenados por puntuación, respecto al caso "AZAR", que ofrece los titulares al azar a cada usuario. Esto significa que en el caso "ORDEN" los titulares que elige el usuario tienen mayor puntuación que los que elige en el caso "AZAR". Es decir, mayor cantidad de titulares que el sistema califica con una buena puntuación, según el perfil del usuario, serán efectivamente interesantes para tal usuario puesto que los selecciona. Así, podemos afirmar que el sistema ofrece en el caso "ORDEN" titulares mejor puntuados según el interés del usuario.

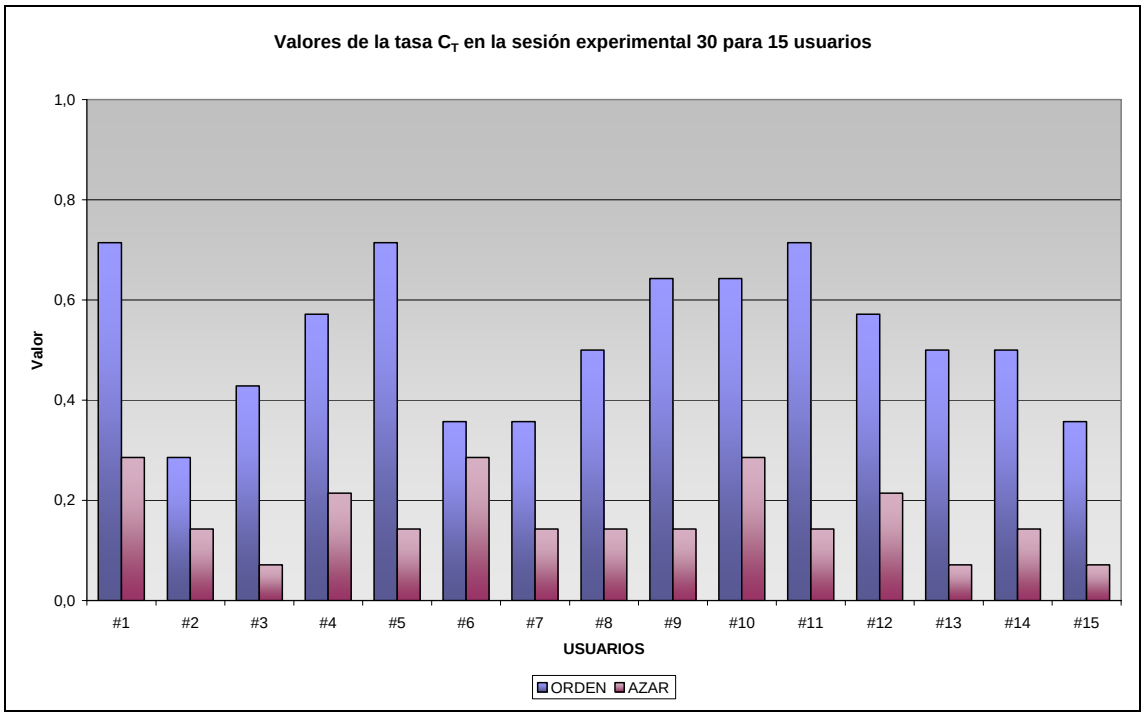


Figura 7.7. Resultados obtenidos en la sesión experimental 30 para la tasa C_T por 15 usuarios, cuando se ofrecen los titulares ordenados, caso "ORDEN", y cuando los titulares se ofrecen al azar, caso "AZAR". En dicha sesión 30 el valor de C_T es mayor en el caso "ORDEN" para todos los usuarios.

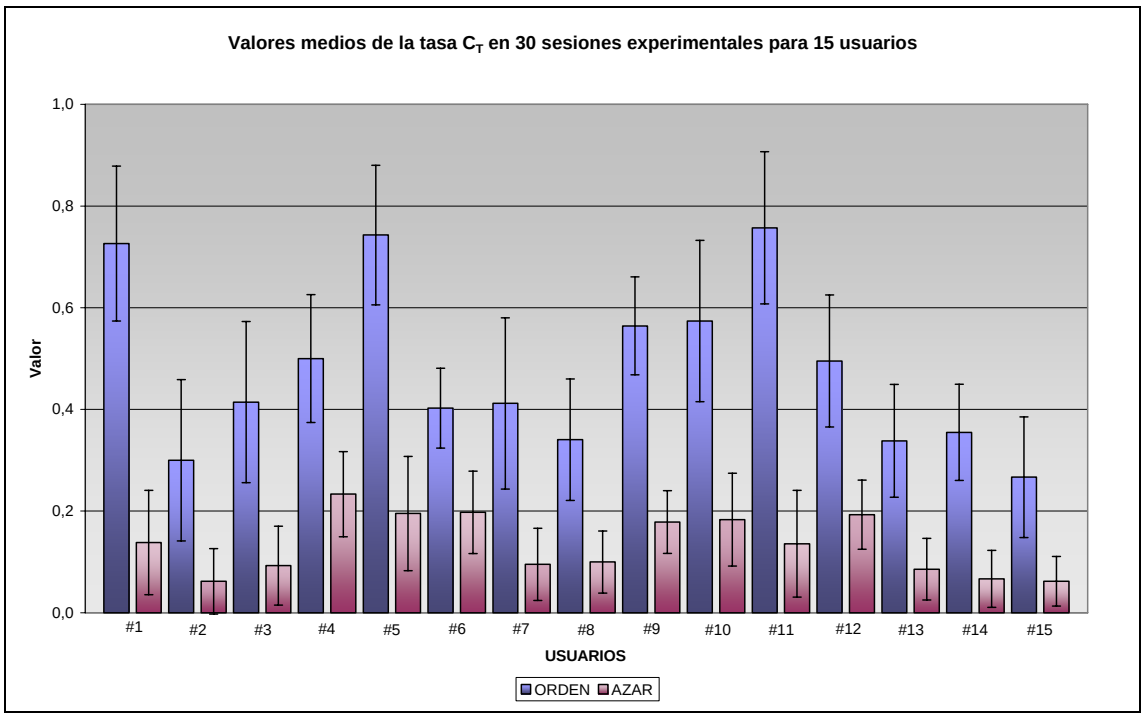


Figura 7.8. Valores medios de la tasa C_T obtenidos por 15 usuarios después de 30 sesiones experimentales cuando se ofrecen los titulares ordenados, caso "ORDEN", y cuando los titulares se ofrecen al azar, caso "AZAR". Para todos los usuarios se observa un valor más alto de la tasa en el caso "ORDEN".

A diferencia de la anterior tasa analizada, C_T, donde sólo se tenía en cuenta si los titulares tenían o no puntuación, para la tasa C_D se compara la puntuación media de los titulares elegidos por el usuario con la puntuación media ideal, que sucedería cuando el usuario escogiese todos los titulares recomendados por el sistema. De esta manera, se obtiene otro punto de vista, orientado a medir no la cantidad sino la calidad, en términos de puntuación, de las elecciones del usuario respecto a las recomendaciones del sistema.

Para cuantificar la mejora del sistema en el caso "ORDEN" respecto al caso "AZAR" se compararán los valores medios de la tasa C_D obtenidos en ambos casos, tanto para la sesión 30 como cuando se consideran las medias de las 30 sesiones experimentales.

El valor medio de la tasa C_D para todos usuarios en la sesión experimental 30 es de 0.889 en el caso "ORDEN" y de 0.440 en el caso "AZAR". En la sesión 30 se constata, por tanto, un incremento de valor medio de la tasa C_D de 202% para el caso "ORDEN" respecto al caso "AZAR". Asimismo, se tiene que el valor medio de la tasa C_D para todos los usuarios en las 30 sesiones experimentales es de 0.826 en el caso "ORDEN" y de 0.376 en el caso "AZAR". Entonces se constata que el valor medio de C_D en las 30 sesiones es un 220% mayor en el caso "ORDEN" que el correspondiente al caso "AZAR".

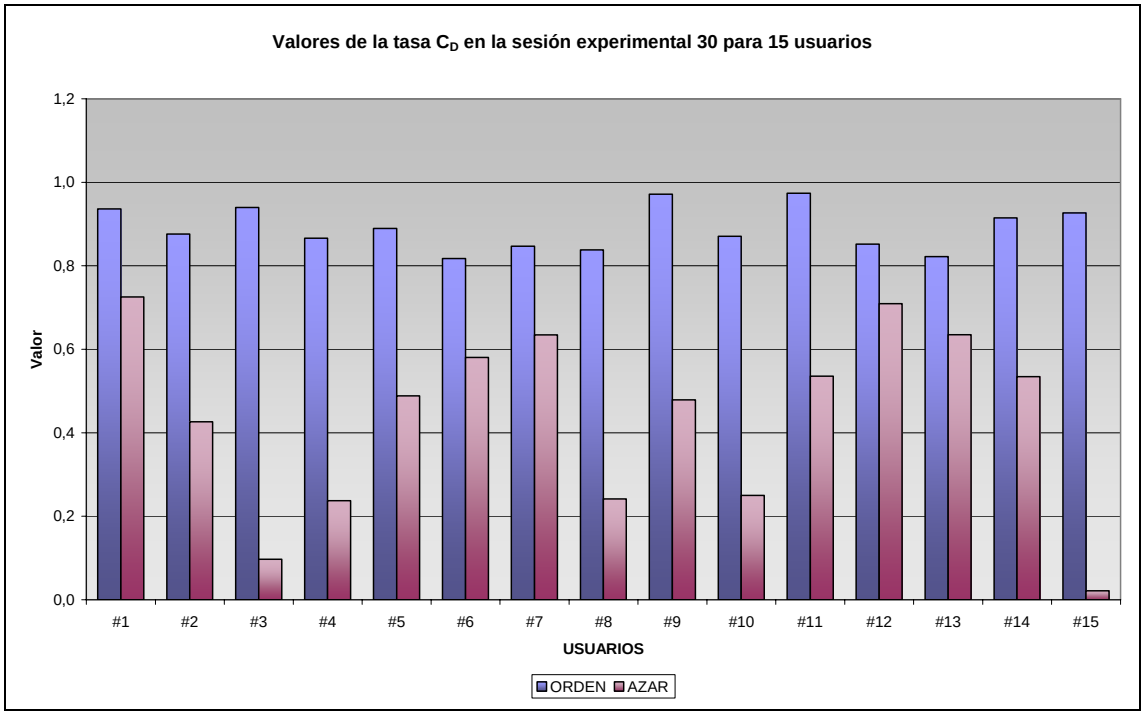


Figura 7.9. Resultados obtenidos por 15 usuarios para la tasa C_D en la sesión experimental 30, cuando se ofrecen los titulares ordenados, caso "ORDEN", y cuando los titulares se ofrecen al azar, caso "AZAR". En dicha sesión 30 el valor de C_D es mayor en el caso "ORDEN" para todos los usuarios.

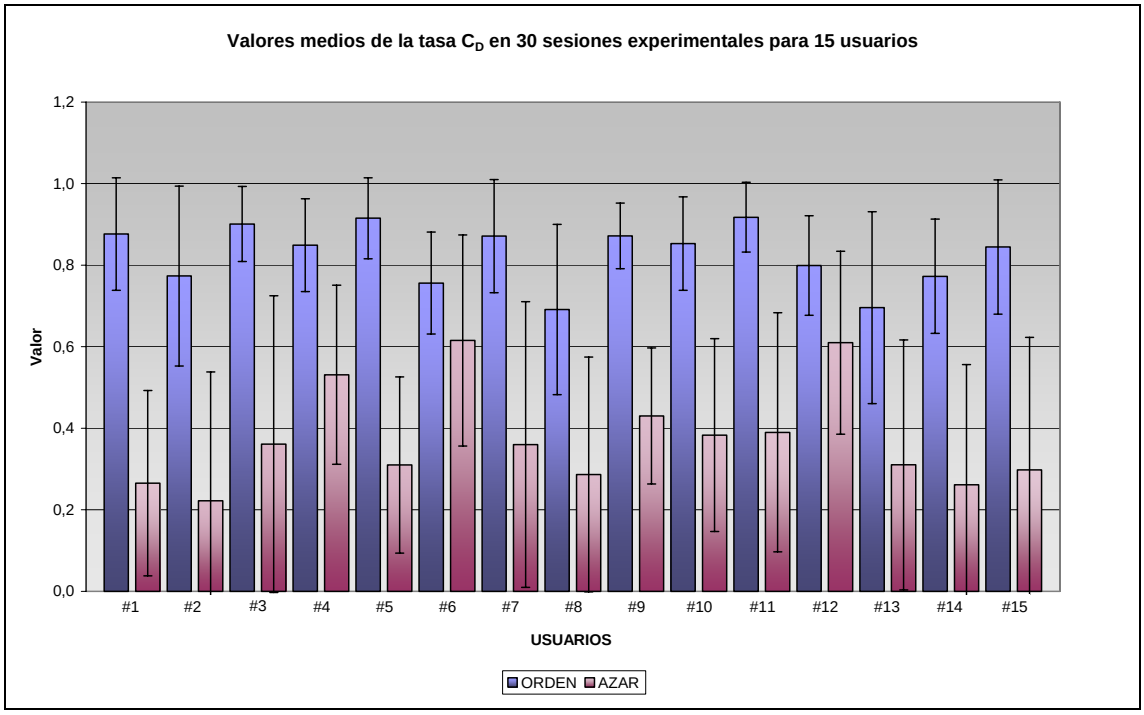


Figura 7.10. Valores medios de la tasa C_D obtenidos por 15 usuarios después de 30 sesiones experimentales cuando se ofrecen los titulares ordenados, caso "ORDEN", y cuando los titulares se ofrecen al azar, caso "AZAR". Para todos los usuarios se observa un valor más alto de la tasa en el caso "ORDEN".

7.5.2. Error Absoluto Medio y Coeficiente de Correlación

En la sección 7.5.1 anterior se ha visto la idoneidad del caso "ORDEN", donde se presentan los titulares de noticias ordenados por puntuación al usuario, respecto al caso "AZAR", donde se le presentan los titulares en orden aleatorio al usuario. Las siguientes medidas se aplicarán por tanto a dicho caso "ORDEN", por ser el de mayor interés y porque para su aplicación será necesario un orden de la información que se ofrece.

En la tabla 7.9 se recogen los valores numéricos obtenidos en la sesión experimental 30 para los 15 usuarios en el *Error Absoluto Medio* $\left|\overline{E}\right|$, definido en la fórmula (6.7), y en su *Desviación Estándar* σ, definida en la fórmula (6.8). En la figura 7.11 se representan estos resultados.

En la tabla 7.9 también se muestran los resultados obtenidos en la sesión experimental 30 por 15 usuarios para el *Coeficiente de Correlación r* entre titulares, definido en la fórmula (6.9). En la figura 7.12 se representan los resultados de este coeficiente.

	Experimento 5 – Valores obtenidos para $\left\|\overline{E}\right\|$, σ y r en la sesión 30 por 15 usuarios														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$\left\|\overline{E}\right\|$	0.062	0.095	0.210	0.123	0.144	0.244	0.193	0.173	0.224	0.206	0.026	0.197	0.158	0.073	0.051
σ	0.020	0.068	0.118	0.037	0.028	0.029	0.075	0.083	0.077	0.050	0.024	0.034	0.034	0.038	0.019
r	0.971	0.987	0.622	0.995	0.933	0.878	0.958	0.911	0.666	0.698	0.989	0.942	0.958	0.973	0.999

Tabla 7.9. Valores obtenidos para el *Error Absoluto Medio*, su *Desviación Estándar* y el *Coeficiente de Correlación* entre titulares en la sesión experimental 30 por 15 usuarios.

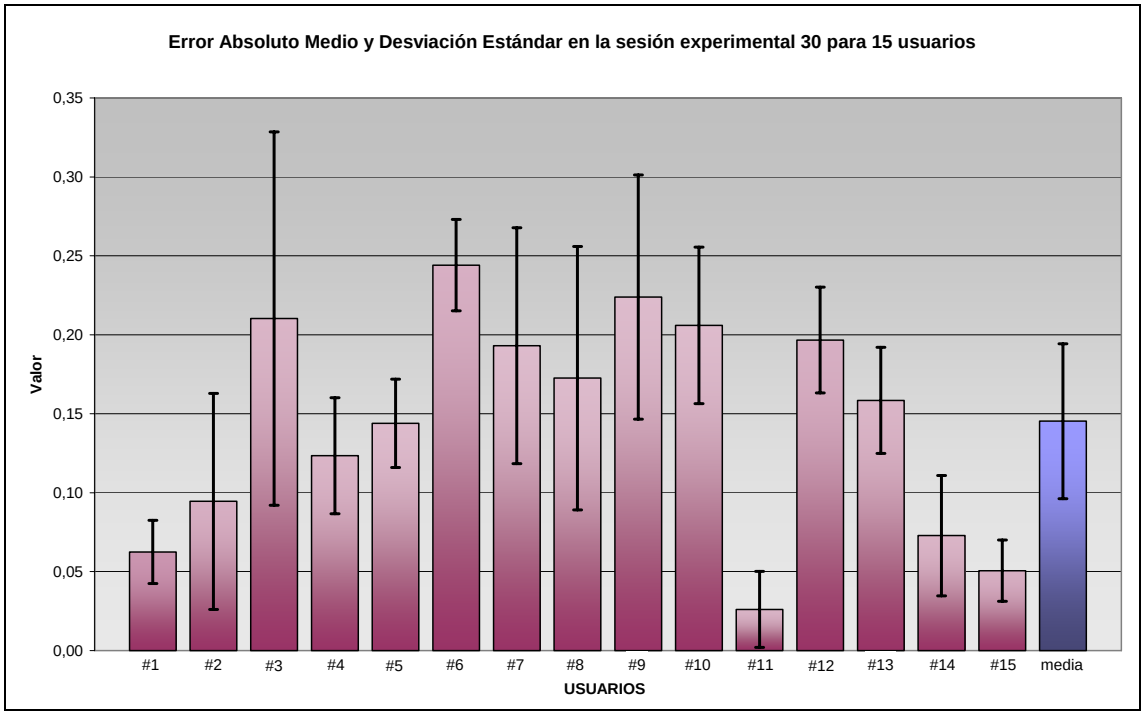


Figura 7.11. Resultados obtenidos en la sesión experimental 30 por 15 usuarios para el *Error Absoluto Medio* y la *Desviación Estándar* del Error. Se observan valores bajos para el *Error Absoluto Medio*, con una media inferior a 0.15 y una *Desviación Estándar* media inferior a 0.05.

Se observan valores bajos para el *Error Absoluto Medio* en los distintos usuarios experimentales. Ninguno de estos usuarios ha llegado a alcanzar el valor de 0.25, obteniéndose en varios casos valores cercanos a cero, como sucede con los usuarios 1, 2, 11, 14 y 15. Este hecho se interpreta como un buen funcionamiento del sistema para todos los usuarios. Asimismo, el valor medio de este *Error Absoluto Medio* para todos los usuarios

es menor que 0.15, con una *Desviación Estándar* media inferior a 0.05, lo cual refuerza la conclusión anterior.

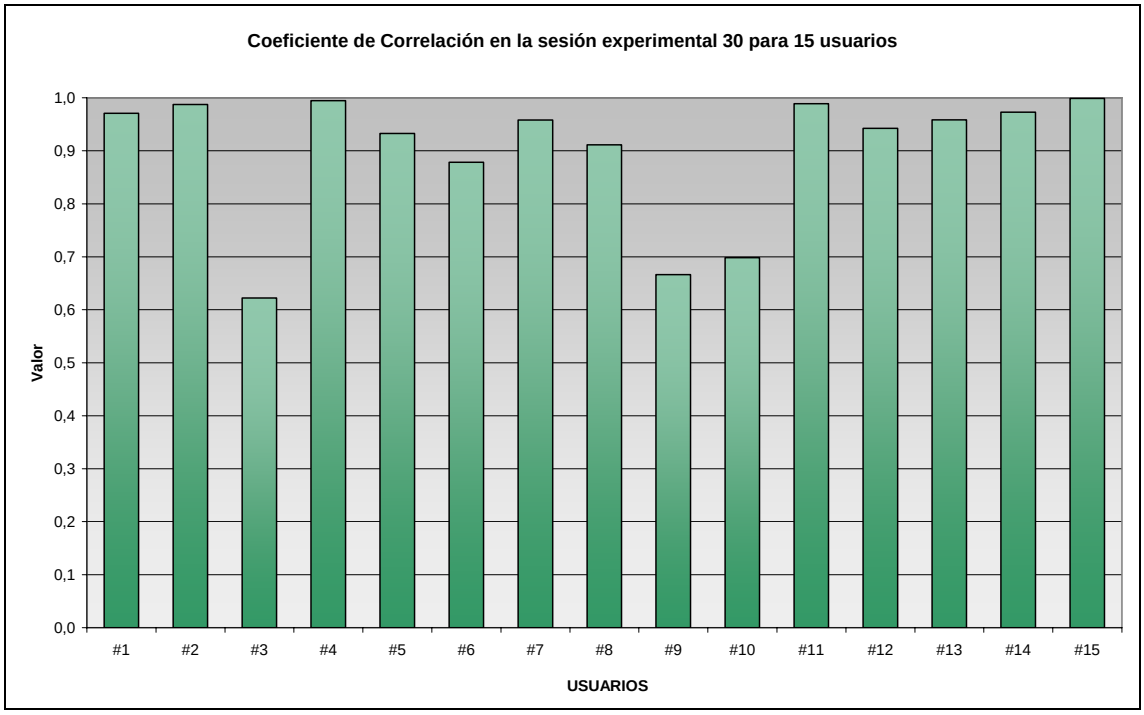


Figura 7.12. Resultados obtenidos en la sesión experimental 30 por 15 usuarios para el *Coeficiente de Correlación* entre titulares. Se observa que los valores de este coeficiente se aproximan a 1 para todos los usuarios.

En el gráfico de la figura 7.12 se observa que los valores del *Coeficiente de Correlación* entre titulares se aproximan a 1 para todos los usuarios, obteniendo la mayoría de los usuarios un resultado superior a 0.9 Además ningún usuario ha obtenido para el coeficiente un valor menor de 0.6. Estos hechos indican que, en general, la puntuación de los titulares propuestos es cercana a la de los que efectivamente elige el usuario en cada sesión.

7.5.3. La R-Precisión

Esta medida propuesta por [Baeza, 1999] y definida en la fórmula (6.10) también se aplicará al caso "ORDEN", como sucedía en la sección 7.5.2 anterior. Esto es debido a que el cálculo de la *R-Precisión* necesita un conjunto de titulares de noticias ordenados para poder calcular entonces la precisión en la posición R del orden.

La medida se utiliza para observar el comportamiento del algoritmo en cada sesión del experimento. Así, se ha calculado un valor de la *R-Precisión*, para las 30 sesiones experimentales efectuadas por los usuarios con el sistema, en las que se han ofrecido los titulares ordenados al usuario.

En la tabla 7.10 se recogen los valores medios para la *R-Precisión* obtenidos por los 15 usuarios considerados en las 30 sesiones experimentales. Estos resultados se representan en la figura 7.13.

	Experimento 5 – Valores medios de la R-Precisión en 30 sesiones para 15 usuarios														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R-Precisión	0.756	0.492	0.724	0.607	0.762	0.449	0.646	0.406	0.666	0.644	0.770	0.552	0.451	0.504	0.665

Tabla 7.10. Valores medios obtenidos por la *R-Precisión* en 30 sesiones experimentales para 15 usuarios.

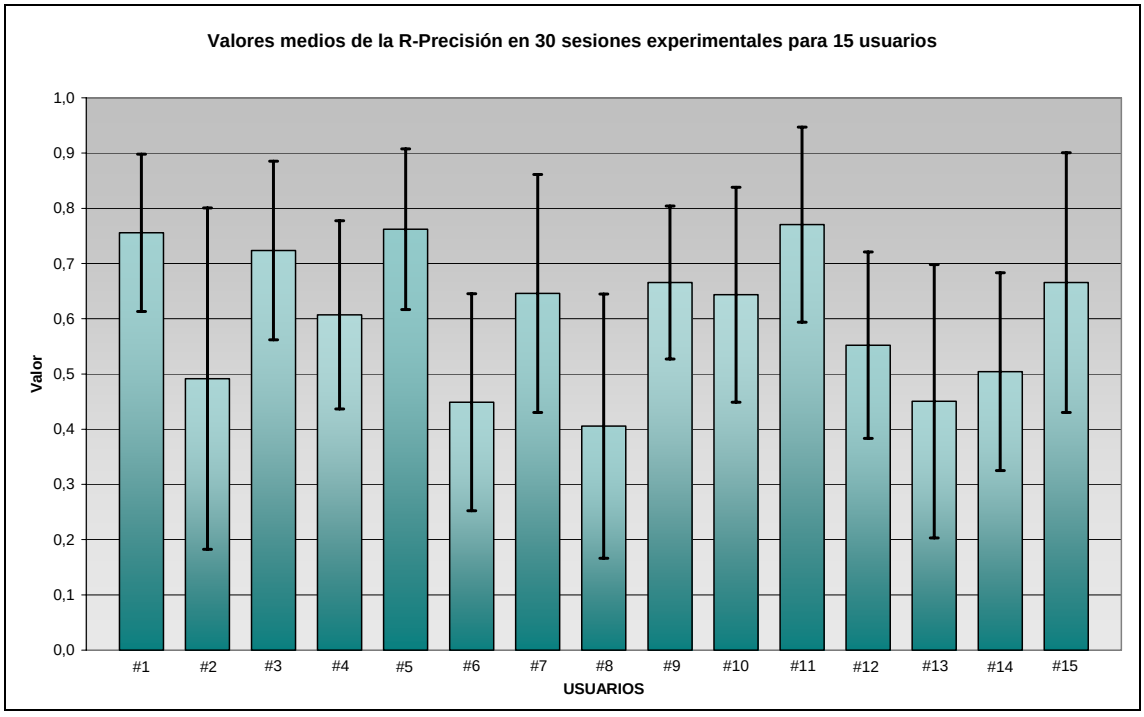


Figura 7.13. Valores medios obtenidos para la *R-Precisión* por 15 usuarios en 30 sesiones experimentales con el sistema. La media mayor es la del usuario #11 y la menor es la del usuario #8.

En el gráfico de la figura 7.13 se observan buenos valores medios de la *R-Precisión* para la mayoría de usuarios, ya que cuando ésta supera el valor de 0.5 puede afirmarse que más de la mitad de los titulares que haya escogido el usuario estarán en el intervalo [1, R]

del orden, siendo R el número de titulares que elige el usuario en la sesión. Ningún usuario ha obtenido un valor medio de la *R-Precisión* menor que 0.4, siendo el valor mínimo el de 0.406 obtenido por el usuario #8. Varios usuarios han superado un valor medio de 0.7 para la medida, siendo la mejor media la del usuario #11 con un valor de 0.770. La *R-Precisión* media para el resto de usuarios se encontrará entre estos dos valores mínimo y máximo.

Aunque las medias anteriores arrojan buenos resultados, la verdadera utilidad de la *R-Precisión* reside en observar su comportamiento a lo largo de las distintas sesiones experimentales con el sistema. Para comparar la *R-Precisión* a lo largo de las 30 sesiones experimentales se ha elegido el usuario con peor media, el #8, y el usuario con mejor media para esta medida, el #11.

En la figura 7.14 se representan gráficamente los valores de la *R-Precisión* obtenidos por los usuarios #8 y #11 en las 30 sesiones experimentales junto con la línea de tendencia de cada uno: "Lineal(Usuario #8)" y "Lineal(Usuario #11)". Estas líneas de tendencia se calculan por el método de mínimos cuadrados, según la ecuación $y = mx + b$, donde m es la pendiente y b es la intersección.

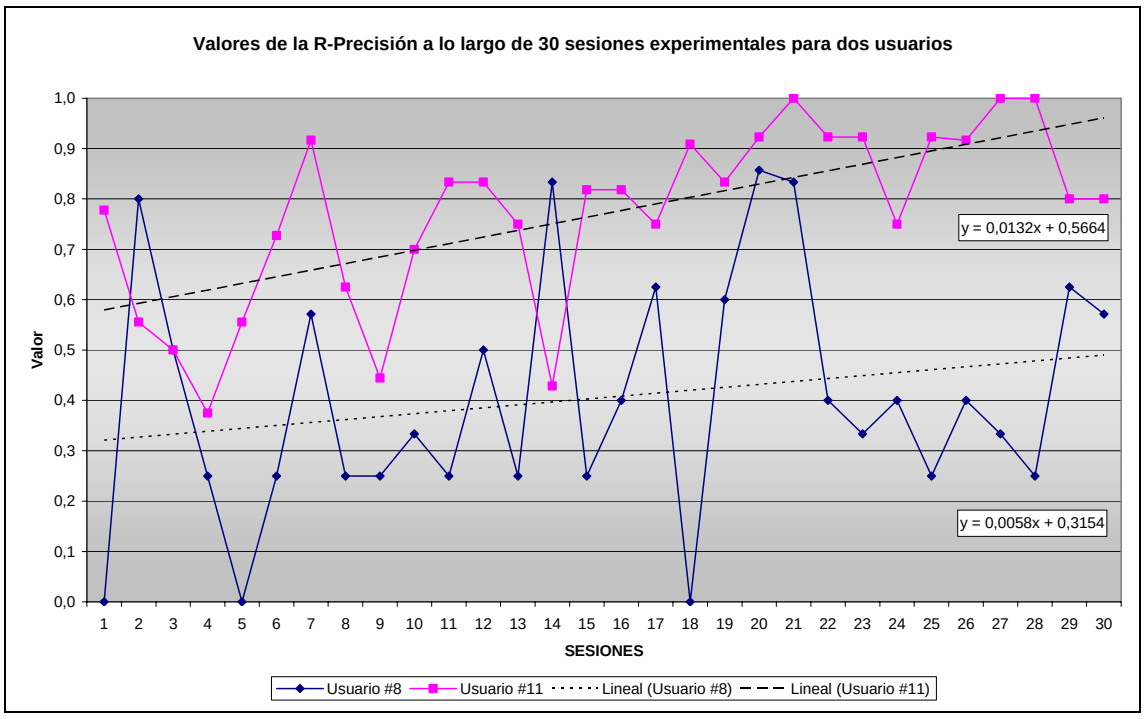


Figura 7.14. Resultados obtenidos por el usuario #8 y por el usuario #11 para la *R-Precisión* a lo largo de 30 sesiones experimentales, junto con las líneas de tendencia de los datos. Se observa en ambos casos una evolución favorable de la *R-Precisión*.

En el gráfico de la figura 7.14 se observa una tendencia de incremento del valor de la *R-Precisión* a lo largo de las distintas sesiones efectuadas. La pendiente de la línea de tendencia de cada usuario, "Lineal(Usuario #8)" y "Lineal(Usuario #11)", es positiva en ambos casos. Este hecho se interpreta como un comportamiento positivo del algoritmo para los usuarios, indicando que el sistema ofrece cada vez mejores ordenaciones de titulares.

7.6. Experimento 6. Probar Puntuación Alternativa (PPA)

En este experimento se selecciona al usuario que haya arrojado mejores resultados en el experimento PAU anterior, el #11, y éste vuelve a realizar 32 sesiones en el sistema configurado para puntuar la información según el coeficiente de Jaccard, propuesto como medida alternativa en la sección 5.3.1 del capítulo 5.

En las 32 nuevas sesiones con el sistema el usuario dispondrá de las mismas noticias que las empleadas para el experimento 5, donde se utilizó la medida del coseno para puntuar la información. Esto nos permitirá comparar los resultados obtenidos por el usuario #11 para el caso "ORDEN" del experimento 5 con los resultados que se obtengan en el experimento 6 utilizando la medida de Jaccard como puntuación de los titulares. De esta manera se tendrán dos casos a considerar: **COS**, formado por el conjunto de resultados obtenidos por el usuario #11 cuando el sistema puntúa la información mediante la medida del coseno, y **JAC**, formado por el conjunto de resultados obtenidos por el mismo usuario cuando el sistema utiliza la medida de Jaccard para puntuar la información.

Los valores numéricos obtenidos por el sistema en el caso "JAC" para las tasas C_P, C_R y C_T son exactamente iguales a los alcanzados por éste en el caso "COS". Por ello no resultará de interés su análisis. La conclusión que se deriva de este hecho es que de alguna manera el usuario ha escogido los mismos titulares entre los ofrecidos por el sistema en ambos casos. Para ello el sistema habrá ido ofreciendo al usuario un conjunto de titulares similar o idéntico en el caso "JAC" al del caso "COS".

Para la tasa C_D se observaron pequeñas diferencias entre ambos casos considerados, sin embargo, tanto el valor medio de la tasa en las 30 sesiones como el valor obtenido en la sesión experimental 30 han sido idénticos. De este hecho se deduce que en el caso "JAC" la puntuación media de los titulares que se van escogiendo se aproxima de igual manera a la puntuación media ideal que en el caso "COS".

Los valores obtenidos para el *Error Absoluto Medio* en la sesión experimental 30 y los valores medios en las 30 sesiones son también son idénticos en ambos casos, lo que indica que el rendimiento del sistema es similar en el caso "JAC" y en el caso "COS".

En la tabla 7.11 se muestran los valores obtenidos para el *Coeficiente de Correlación r* en la sesión experimental 30 junto con las medias de esta medida en las 30 sesiones. En la figura 7.15 se representan gráficamente estos datos.

	Experimento 6 – Valores de la Correlación en la sesión 30 y su medias	
caso	r	$\bar{r}$
COS	0.989	0.964
JAC	0.989	0.936

Tabla 7.11. Valores obtenidos por el usuario #11 para el *Coeficiente de Correlación* en la sesión experimental 30 junto con sus medias para los casos "COS" y "JAC".

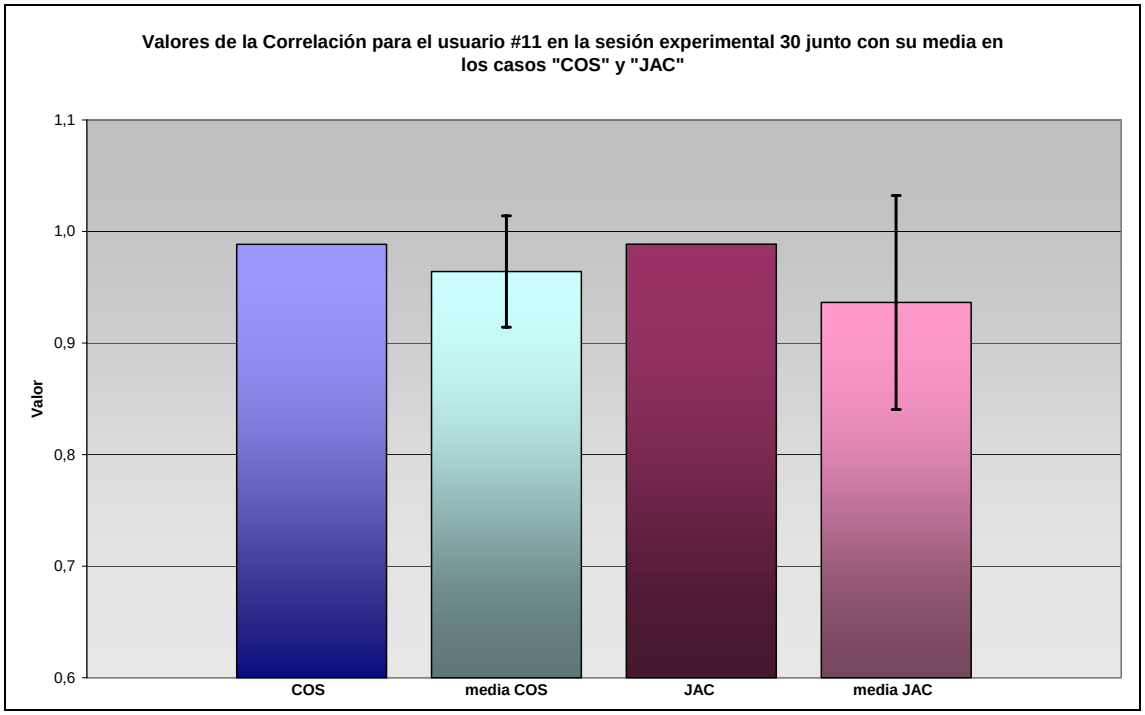


Figura 7.15. Resultados obtenidos en la sesión experimental 30 por el usuario #11 para el *Coeficiente de Correlación*, junto con sus valores medios. Se obtiene el mismo valor de Correlación para los casos "COS" y "JAC". Se observa un mayor valor medio del coeficiente para el caso "COS".

En el gráfico de la figura 7.15 se observa que se ha obtenido el mismo valor en la sesión experimental 30 para los dos casos considerados en el experimento, "COS" y "JAC". Y, aunque el valor medio obtenido en las 30 sesiones es algo mayor en el caso "COS", concretamente un 1.05%, que en el caso "JAC", valores tan cercanos para la *Correlación* indican que en ambos casos el usuario escoge principalmente los titulares bien puntuados por el sistema.

Por último, para la *R-Precisión* se obtuvieron valores idénticos en todas las sesiones en los dos casos considerados. Esto indica que el sistema ha tenido igual comportamiento al utilizar como puntuación de los titulares la medida del coseno que al utilizar la medida de Jaccard.

En general, se puede concluir que el funcionamiento del sistema es bastante independiente del método de puntuación de la información elegido, teniendo más peso la calidad del perfil de usuario. En este sentido, teniendo en cuenta los resultados de éste experimento y los resultados de los anteriores, tendrá bastante influencia la existencia o no de una palabra en dicho perfil de usuario.

7.7. Resumen

En este capítulo de la Tesis se han mostrado y se han analizado los resultados obtenidos en los distintos experimentos llevados a cabo para determinar algunos parámetros del sistema propuesto y su eficacia con diversos usuarios.

El primer experimento (CRS) evaluará si es más favorable para el sistema considerar los resúmenes opcionales de las noticias para enriquecer el perfil de usuario con nuevos términos de dichos resúmenes o si es mejor considerar solamente los términos de los titulares. Se efectuaron diversas sesiones con idénticas selecciones de titulares en dos versiones configuradas del sistema, una considerando los resúmenes y otra sin considerarlos, y se recogieron los valores de las tasas propuestas para su comparación, en concreto C_R, C_T y C_D, definidas en las secciones 6.4.1 y 6.4.2. Se observaron para todas ellas mejores resultados al considerar los resúmenes opcionales de las noticias. Para la tasa C_D que ofreció resultados más ajustados entre ambos casos se aplicó la prueba t-Student con el objeto de determinar que efectivamente existen diferencias significativas entre las dos alternativas experimentadas. Así, a tenor de los resultados, finalmente se escogió la opción de considerar los resúmenes en el proceso de elaboración del perfil de usuario, que se mantendrá para el resto de experimentos.

En el segundo experimento (DIV) se probaron diversos valores para el *intervalo de vida* que es un componente de un *factor de olvido* opcional, definido en la fórmula (5.9). En este caso, se analizaron los resultados obtenidos para la tasa C_D pues el resto de las tasas propuestas toman idénticos valores para este experimento en todos los casos, al realizarse exactamente las mismas selecciones de titulares en cada sesión. Examinando los resultados del experimento se llegó a la conclusión de que la adopción de un *factor de olvido* no favorece significativamente al sistema por lo que finalmente se desestimó su uso.

El tercer experimento considerado (IRP) está orientado a seleccionar la mejores proporciones consideradas en el cálculo del perfil acumulado al término de cada sesión, según las fórmulas (5.7) y (5.15). Se probaron distintos pares de valores, analizándose los resultados obtenidos para la tasa C_D durante distintas sesiones. Aunque con bastantes similitudes en el comportamiento de los pares considerados experimentalmente, se observó la mejor tendencia para las proporciones (a=50, b=50), consideradas como la media aritmética entre el perfil de sesión y el perfil acumulado.

El cuarto experimento (CRS2) se realiza para reafirmar las conclusiones obtenidas en el primer experimento (CRS), pero en este caso considerando los valores que se han determinado empíricamente según los resultados de los experimentos 2 y 3 anteriores. En este caso se analizó la evolución de la tasa C_D a lo largo de 30 sesiones experimentales para los dos casos ya comentados en el experimento 1. Se obtuvieron resultados más favorables cuando se consideraron los resúmenes opcionales de las noticias para ir formando el perfil de usuario, confirmando por tanto las conclusiones del primer experimento.

El experimento 5 (PAU) evaluará el funcionamiento del sistema propuesto con diferentes usuarios, pudiendo considerarse como una calibración del método en el "mundo real". Cada usuario efectuó 2 sesiones de entrenamiento y 30 sesiones experimentales. Todos los usuarios, que se seleccionaron con intereses heterogéneos, dispusieron de la misma colección de noticias, eligiendo éstos las más convenientes a sus correspondientes necesidades informativas. Así, en cada sesión se le ofreció a cada usuario una selección de titulares ordenados según su puntuación, calculada de acuerdo con su perfil de usuario correspondiente. Además, para poder contrastar los resultados se repitió cada sesión con el sistema configurado para que ofreciera los titulares aleatoriamente al usuario.

Para todos los usuarios del experimento 5 se observaron mejores resultados, según las tasas C_T y C_D, en el caso en que el sistema recomienda una selección ordenada de titulares. Se evaluaron otras medidas como el *Error Absoluto Medio*, su *Desviación Estándar* y la

Correlación entre titulares, determinando, según los resultados de las dos primeras, un buen funcionamiento del sistema para todos los usuarios, y, según la *Correlación*, que la puntuación que se le otorga a los titulares es cercana a la de los que efectivamente escoge cada usuario.

Otra medida analizada para cada usuario del experimento 5 ha sido la *R-Precisión*, obteniéndose buenos valores medios en general para todos los usuarios. De esta medida se analizó también su evolución a lo largo de las 30 sesiones experimentales para dos de los usuarios, el que ofrecía la peor media y el que ofrecía la mejor. Se observó en ambos casos una tendencia positiva de los datos, lo que nos permitió concluir que el algoritmo tiene un comportamiento positivo para los usuarios, indicando que el sistema ofrece sucesivamente mejores ordenaciones de titulares.

Por último, en el experimento 6 (PPA), se probó el sistema utilizando una medida distinta para puntuar la información, el coeficiente de Jaccard, en contraste con la medida del coseno utilizada en todos los experimentos anteriores. Para el usuario con mejores medias del experimento 5 se obtuvieron resultados prácticamente similares para las dos medidas, concluyendo, por tanto, que el funcionamiento del sistema es bastante independiente del método de puntuación elegido.

Capítulo 8

CONCLUSIONES

En el trabajo de tesis doctoral presentado en esta memoria se ha desarrollado un método para crear un sistema de priorizado de información periódica, procedente de una serie de fuentes preestablecidas, que la presenta a los usuarios en orden de importancia según sus preferencias.

En la primera parte de este trabajo se estudiaron los sistemas de recuperación de información y las principales técnicas de evaluación que se aplican a éstos.

Posteriormente se describieron los aspectos a tener en cuenta para definir y crear perfiles de usuario, cómo adquirir los datos del usuario, la representación del perfil de usuario y las técnicas de inferencia asociadas.

El análisis de dichos problemas y de los distintos enfoques encontrados en la bibliografía para resolverlos, nos llevó a establecer una metodología de diseño y a proponer un sistema de recuperación y filtrado de información de la Web, más concretamente un agregador inteligente que recomienda contenidos al usuario, denominado NectaRSS.

Dicho sistema, se basa en la utilización del modelo vectorial y el esquema *tf*, descritos en el capítulo 2, y puntúa la información que se le ofrece al usuario, en forma de titulares de noticias, mediante la medida del coseno propuesta por Salton o mediante la medida de Jaccard.

Finalmente, el sistema de recomendación propuesto se evaluó experimentalmente y se comprobó su validez.

Este capítulo es un resumen de los logros, aportaciones y posibles líneas de investigación a seguir, en base a la investigación realizada con el sistema NectaRSS.

8.1. Principales Aportaciones y Conclusiones

Las principales aportaciones y conclusiones obtenidas quedan resumidas a continuación:

- Se ha creado un sistema de filtrado o priorizado de información, capaz de recomendar ésta a un usuario según sus preferencias.

- Se ha desarrollado un método automático para captar las preferencias del usuario y confeccionar su perfil sin esfuerzo alguno por parte de éste, en base a su historial de selección de la información ofrecida.

- Se ha encontrado una forma óptima de crear ese perfil de usuario, y de usarlo para dar la información más relevante.

- Los procesos de adquisición de preferencias y de puntuación de la información se realizan de manera totalmente transparente al usuario.

- Se han evaluado diferentes estrategias y opciones para que el resultado del sistema sea óptimo.

- Los parámetros fijados experimentalmente para el sistema son válidos para distintos usuarios heterogéneos.

- Puntuar los titulares según un perfil de usuario resulta beneficioso ya que las ordenaciones de información que ofrece el sistema al usuario resultan mejores para éste que un orden aleatorio.

- Conforme el sistema obtiene más datos de las preferencias del usuario más se aproxima la puntuación de los titulares propuestos a la de los que efectivamente

elige el usuario en cada sesión, lo que redunda en una mejor ordenación de los titulares, desde el punto de vista del usuario.

- El sistema demuestra un funcionamiento adecuado para distintos usuarios.

- El rendimiento del sistema resulta independiente del método de puntuación de la información elegido.

- El uso del sistema propuesto proporciona más satisfacción a un usuario respecto a sus demandas informativas en comparación a una presentación al azar típica, puesto que cada vez encuentra más fácil y rápidamente la información que realmente le interesa, sin tener que realizar ninguna otra acción adicional.

8.2. Líneas de investigación futuras

El desarrollo del presente trabajo ha permitido identificar una serie de temas y líneas de investigación originales, que se considera de interés abordar:

- Determinar el rendimiento del sistema considerando conjuntos de palabras encadenadas, en la suposición de que puedan ser más relevantes para el usuario.

- Comprobar si resulta relevante otorgar mayor puntuación a las palabras o términos que se encuentren en la información seleccionada en primer lugar por el usuario, en la suposición de éstos serán más importantes para dicho usuario.

- Mostrar al usuario cierto porcentaje de titulares de información aleatorios, en la suposición de que se puedan encontrar nuevos temas de interés para dicho usuario.

- Desarrollar una aplicación del sistema "on-line" en la que, en el servidor web, se mantenga un perfil para cada usuario que visite la página de los titulares de

información, con el objeto de personalizar automáticamente dichos titulares la próxima vez que la visite. Esta forma de aplicar el sistema NectaRSS resultaría de especial interés en tiendas y periódicos "on-line".

- Aplicación de algoritmos evolutivos y de aprendizaje automático en la elaboración del perfil de usuario.

- Elaborar y utilizar varios perfiles del usuario para reflejar mejor sus intereses.

- Añadir capacidades "sociales" al sistema, teniendo en cuenta, por ejemplo, la información que eligen las personas en las que el usuario confía, o lo que eligen distintos usuarios con perfiles similares.

- Utilizar el perfil de usuario para recomendar noticias de otras fuentes diferentes a las que el usuario haya preseleccionado.

Bibliografía y Referencias

[Akolulchina y Ganascia, 1997] Akolulchina, I. y Ganascia, J.: 1997, *Satelit-Agent: An adaptive interface agent based on learning interface agent technology*. In: A. Jameson, C. Paris and C. Tasso, Proceedings of 6th International Conference on User Modeling, UM'97. Sardinia, Italy. Wien: SpringerWienNewYork, 22-32.

[Albrech et al., 1997] Albrech, D., Zukerman, I., Nicholson, A. y Bud, A.: 1997, *Towards a Bayesian model for keyhole plan recognition in large domains*. In A. Jameson, C. Parisand, C. Tasso (ed). Proceedings of 6th International Conference on User Modeling, UM'97. Sardinia, Italy. Wien: SpringerWienNewYork, 365-376.

[Alspector et al., 1997] Alspector, J., Kolez, A. y Karunanithi, N.: 1997, *Feature-based and clique-based user models for movie selection: a comparative study*. User Modeling and User Adapted Interaction 7(4), 279-304.

[Ambrosini et al., 1997] Ambrosini, L., Cirillo, V. y Micarelli, A.: 1997, *A hybrid architecture for user-adapted information filtering on the WWW*. In A. Jameson, C. Parisand, C. Tasso (ed). Proceedings of 6th International Conference on User Modeling, UM'97, Sardinia, Italy. Wien: SpringerWienNewYork, 59-61.

[Ardissono et al., 1999] Ardissono, L., Goy, A., Meo, R. y Petrone, G.: 1999, *A configurable system for the construction of adaptive virtual stores*. World Wide Web 2(3), 143-159.

[Arocena, 1998] Arocena G., Mendelzon A. *WebOQL: Restructuring documents, databases and Webs*. In Int. Conf. on Data Engineering, pages 24-33, Orlando, Florida, 1998.

[Baeza, 1999] Baeza-Yates, R. and Ribeiro-Neto, B. *Modern information retrieval*. ACM Press. Addison-Wesley, 1999.

[Balabanovic, 1997] Balavanovic, M.: 1997, *An adaptive web page recommendation service*. In Proceedings of the 1st International Conference on Autonomous Agents. Marina del Rey, USA, 378-385.

[Bares y Lester, 1997] Bares, W. y Lester, J.: 1997, *Cinematographic user models for automated real-time camera control in dynamic 3D environments*. In A. Jameson, C. Parisand, C. Tasso (ed). Proceedings of 6th International Conference on User Modeling, UM'97. Sardinia, Italy. Wien: SpringerWienNewYork, 215-226.

[Bauer, 1996] Bauer, M.: 1996, *A Dempster-Shapher approach to modeling agent preferences for plan recognition*. User Modeling and User Adapted Interaction 5(3-4), 317-348.

[Berners, 1989] Berners-Lee, T. *Information Management: A Proposal*. CERN, 1989.

[Blair, 1990] Blair, D.C. *Language and representation in information retrieval*. Amsterdam. Elsevier Science Publishers, 1990.

[Boyle y Encarnação, 1994] Boyle, C. y Encarnação, A.: 1994, *Metadoc: an adaptive hypertext reading system*. User Modeling and User Adapted Interaction 4(1), 1-19.

[Brajnik y Tasso, 1994] Brajnik, G. y Tasso, C.: 1994, *A shell for developing non-monotonic user modeling systems*. International Journal of Human-Computer Studies 40, 31-62.

[Bray, 2004] Bray, T., Paoli J., Sperberg-McQueen C. M., Maler E., Yergeau F. *Extensible Markup Language 1.1*. W3C Recommendation, 4 February 2004, edited 15 April 2004. http://www.w3.org/TR/2004/REC-xml11-20040204/

[Breese et al., 1998] Breese, J., Heckerman, D. y Kadie, C.: 1998, *Empirical analysis of predictive algorithms for collaborative filtering*. Proceedings of the 14th Annual Conference on Uncertainty in Artificial Intelligence (UAI-98). Morgan Kaufmann, 43-52.

[Carrol y Rosson, 1987] Carrol, J. y Rosson, M.: 1987, *The paradox of the active user*. In JM Carrol (ed) Interfacing thought: Cognitive Aspects of Human-Computer Interaction. MIT Press.

[Chaffee, 2000] Chaffee, J., Gauch, S. *Personal Ontologies for Web Navigation*. Proc. 9th Intl. Conf. on Information and Knowledge Management (CIKM'00) McLean VA, Nov. 2000, pp. 227-234.
http://www.ittc.ku.edu/obiwan/

[Chan, 1999] Chan, P.: 1999, *A non-invasive learning approach to building web user profiles*. Proceedings of the KDD-99 Workshop on Web Analysis and User profiling. Computer Science, Florida Institute of Technology, Melbourne, Australia.
http://citeseer.ist.psu.edu/chan99noninvasive.html

[Chin, 1989] Chin, D. *KNOME: modeling what the user knows in UC*. In A. Kobsa and W. Wahlster (eds). User Models in Dialog Systems. Springer-Verlag, 74-107. 1989.

[Chowdhury, 1999] Chowdhury, G. G. *Introduction to modern information retrieval*. London: Library Association, 1999.

[Cleverdon et al., 1966] Cleverdon, C.W., Mills, J., Keen, M., *Factors Determining the Performance of Indexing Systems*, Vol. 1, Design, Vol.II, Test Results, ASLIB Cranfield Project, Cranfield (1966).

[Cooper, 1973] Cooper, W.S. *On selecting a Measure of Retrieval Effectiveness*. Journal of the American Society for Information Science, v. 24, March-April 1973. p.87-92.

[Crabtree y Soltysiak, 1998] Crabtree, B. y Soltysiak, S.: 1998, *Identifying and tracking changing interests*. International Journal on Digital Libraries 2 (1), 38-53.

[Croft, 1987] Croft, W. B. *Approaches to intelligent information retrieval*. Information Proccesing & Management, 23, 4, 1987. p. 249-254.
[DATSI, 2005] Departamento de Arquitectura y Tecnología de Sistemas Informáticos (DATSI). Universidad Politécnica de Madrid. http://www.datsi.fi.upm.es/~coes/.

[De Bra, 1994] De Bra, P. M. E., Post, R. D. J. *Searching for arbitrary information in the WWW: The fish search for Mosaic*. In Proc. of the 2nd Int. WWW Conference, Chicago, 1994.
http://archive.ncsa.uiuc.edu/SDG/IT94/Proceedings/Searching/debra/article.html

[De la Fuente, 1998] De la Fuente, P. *Texto Estructurado en Internet: SGML HTML y XML*. Dpto. Informática. Universidad de Valladolid, 1998. Presentado en las VI Jornadas Iberoamericanas de Informática. Santa Cruz de la Sierra. Bolivia, del 7 al 11 de Septiembre de 1998.

[Delgado, 1998] Delgado Domínguez, A. *Mecanismos de recuperación de Información en la WWW*. Memoria de Investigación. Universitat Illes Balears. Mallorca, 1998.

[Delgado, 2001] Delgado Domínguez, A. *Herramientas de búsqueda para la WWW*. Congreso Internacional Virtual de Educación CIVE2001. Abril 2001. http://servidorti.uib.es/adelaida/CIVE/adecive.htm.

[Dominich, 2000] Dominich, S. *A unified mathematical definition of classical information retrieval.* Journal of the American Society for Information Science, 51 (7), 2000. p. 614-624.

[Feedster, 2005] *Feedster: Search Today's Internet for listings, news, and blogs*. 2005. http://www.feedster.com/

[Fernández, 1997] Fernández M., Florescu D., Levy A., Suciu D. *A query language for a Web-site management system*. SIGMOD Record, 26(3): 4-11, 1997.

[Fink et al., 1998] Fink, J., Kobsa, A. y Nill, A.: 1998, *Adaptable and adaptive information provision for all users, including disabled and elderly people*. The New Review of Hypermedia and Multimedia 4, 163-188.

[Frants, 1997] Frants, V.I. et al. *Automated information retrieval: theory and methods*. San Diego: Academic Press, cop.1997. XIV, 365 p.

[García, 2002] García, F.J., Gil, A.B. *Personalización de Sistemas de Recomendación*. Workshop de Investigación sobre Nuevos Paradigmas de Interacción en Entornos Colaborativos Aplicados a la Gestión y Difusión del Patrimonio Cultural, COLINE'02. Granada, 11-12 Nov. de 2002.

[García et al., 2002] García, F. J., Gil. A.B., Moreno, M.N., Curto, B. *A Web-Based E-Commerce Facilitator Intermediary for Small and Medium Enterprises: A B2B/B2C Hybrid Proposal.* In K. Bauknecht, A. Min Tjoa, G. Quichmayr (Eds.) E-Commerce and Web Technologies. Third International Conference, EC-Web 2002 Proceedings. Lecture Notes in Computer Science Series. Vol. LNCS 2455. Springer Verlag (2002) 47-56.

[Goo, 2005] Google Directory. *RSS News Readers*. Julio de 2005. http://directory.google.com/Top/Reference/Libraries/Library_and_Information_Science/Technical_Services/Cataloguing/Metadata/RDF/Applications/RSS/News_Readers/.

[Grossman, 1998] Grossman, D.A. and Frieder, O. *Information retrieval: algorithms and heuristics*. Boston: Kluwer Academia Publishers, 1998.

[Hersovici, 1998] Hersovici, M., Jacobi, M., Maarek, Y. S., Pelleg, D., Shtalhaim M., Ur, S. *The shark-search algorithm. An application: tailored Web site mapping*. In 7th WWW Conference, Brisbane, Australia, 1998.

[Herwijnen, 1994] Herwijnen, Eric van. *Practical SGML. 2nd edition*. Kluwer Academic Publishers, 1994.

[Hijikata et al., 2001] Hijikata, Y., Yoshida, T. y Nishida, S.: 2001, *Adaptive hypermedia system for supporting information providers in directing users through hyperspace*. Proceedings of the 3rd on Adaptive Hypertext and Hypermedia at the 12th ACM Conference on Hypertext and Hypermedia, 147-156.

[Hill, 1995] Hill, W., Stead, L., Resenstein, R., Furnas, G. *Recommending and evaluating choices in a virtual community of use*. In Proceedings of CHI 95, Denver, CO. 1995.

[Himmeroder, 1997] Himmeroder, R., Lausen G., Ludascher B., Schlepphorst C. *On a declarative semantics for Web queries*. In Proc. of the Int. Conf. on Deductive and Object-Oriented Database (DOOD), pages 386-398, Singapore, 1997.

[Howe, 1997] Howe, A., Dreilinger, D. *Savvysearch: A metasearch engine that learns which search engines to query*. AI Magazine, 18(2): 19-25, 1997.

[HTML, 1999] *HTML 4.01 Specification*. Technical report, WWW Consortium (W3C), 1999. http://www.w3.org/TR/html401/

[Jameson, 1996] Jameson, A., *Numerical uncertainty management in user and student modeling: an overview of systems and issues*. User Modeling and User-Adapted Interaction 5 (3-4), 193-251. 1996.

[Kazunari, 2004] Kazunari Sugiyama, Kenji Hatano, Masatoshi Yoshikawa. *Adaptive Web Search Based on User Profile Constructed without Any Effort from Users*. Proceedings of the 13th international conference on World Wide Web. 2004.

[Kobsa et al., 1994] Kobsa, A., Muller, D. y Nill, A.: 1994, *KN-AHS: an adaptive hypertext client of the user modeling system BGP-MS*. Proceedings of the 4th International Conference on User Modeling, 99-105.

[Kobsa y Pohl, 1995] Kobsa, A., Koenemann, J. y Pohl, W.: 1995, *The user modeling shell system BGP-MS*. User Modeling and User-Adapted Interaction 4 (2), 59-106.

[Konstan et al., 1997] Konstan, J., Miller, B., Maltz, D., Herlocker, J., Gordon, L. y Riedl, J.: 1997, *GroupLens: applying collaborative filtering to Usenet news*. Communications of the ACM 40(3), 77-87.

[Korfhage, 1997] Korfhage, R.R. *Information Retrieval and Storage*. New York: Wiley Computer Publisher, 1997.

[Krogsaeter et al., 1994] Krogsaeter, M., Oppermann, R. y Thomas, C.: 1994, *A user interface integrating adaptability and adaptativity*. In R. Oppermann (ed): Adaptive user support: ergonomic design of manually and automatically adaptable software. Lawrence Erlbaum, 97-125.

[LaMacchia, 1997] LaMacchia, B. *The Internet fish construction kit*. In 6th Int. WWW Conference, Santa Clara, CA, USA, 1997.

[Lancaster, 1993] Lancaster, F. W. and Warner, A.J. *Information Retrieval Today*. Arlington, Virginia : Information Resources, 1993.

[Lashkari, 1995] Lashkari, Y. *Webhound*. Master's thesis. MIT Media Laboratory, 1995.

[Lesh, 1995] Lesh, N., Etzioni, O.: 1995, *A sound and fast goal recognizer*. Proceedings of the 14th International Joint Conference on Artificial Intelligence, IJCAI'95, Montreal, 1704-1710.

[Lesh et al., 1999] Lesh, N., Rich, C. y Sidner, C.: 1999, *Using plan recognition in humancomputer collaboration*, In J Kay (ed). UM99 User Modeling: Proceedings of the 7th International Conference, Springer-Verlag, 23-32. http://www.cs.usask.ca/UM99/Proc/lesh.pdf

[Lieberman, 1995] Lieberman, H.: 1995, *Letizia: An agent assists web browsing*. Proceedings of the 14th International Joint Conference on Artificial Intelligence, IJCAI'95, Montreal, 924-929.

[Llidó, 2002] Llidó Escrivá, D. M. *Extracción y Recuperación de Información Temporal*. Tesis Doctoral. Universitat Jaume I, Castellón, 2002.

[López, 2002] López, C, Guerrero, V., Moya, F. R*etroalimentación por relevancia: nueva perspectiva desde la programación evolutiva*. Actas I Jorn. de Tratamiento y Recuperación de la Información (JOTRI). 2002.

[Maes, 1994] Maes, P.: 1994, *Agents that reduce work and overload*. Communications of the ACM 37 (7), 31- 40.

[Maes, 1995] *Intelligent Software*. Scientific American, vol. 273, no. 3, pp. 84-86.

[Meadow, 1993] Meadow, C. T. *Text Information retrieval Systems*. San Diego: Academic Press, 1993.

[Martínez, 2004] Martínez Méndez, F. J., Rodríguez Muñoz, J. V. *Reflexiones sobre la evaluación de los sistemas de recuperación de información : necesidad, utilidad y viabilidad*. Anales de Documentación Nº 7: pp. 153-170. 2004.

[Merelo et al., 2004] Merelo, J.J., Carpio, J., Tricas, F., Ferreres, G., Prieto, B. *Recomendación de weblogs utilizando reglas de asociación*. GT-43. Weblogs, ¿un nuevo género de comunicación?. II Congreso Online del Observatorio para la Cibersociedad. Barcelona. 2004.

[Middleton, 2001] Middleton S., De Roure D., Shadbolt N. *Capturing knowledge of user preferences: ontologies in recommender systems*. In Proceedings of the 1st International Conference on Knowledge Capture (K-Cap2001), Victoria, BC Canada, 2001.

[Mislevy y Gitomer, 1996] Mislevy, R. y Gitomer, D.: 1996, *The role of probability-based inference in intelligent tutoring systems*. User Modeling and User Adapted Interaction 5(3-4), 253-282.

[Mitchell et al., 1994] Mitchell, T., Caruana, R., Freitag, D., McDermott, J. y Zabowski, D. 1994, *Experience with a learning personal assistant*. Communications of the ACM 37 (7), 81-91.

[Mizzaro, 2002] Mizzaro S., Tasso C. (2002). *Ephemeral and persistent personalization in adaptive information access to scholarly publications on the Web*. Artificial Intelligence Laboratory Department of Mathematics and Computer Science , 2002.

[Moffat, 2003] Moffat, Malcolm. *RSS-a primer for publishers and content providers*. EEVL Development Officer, Heriot-Watt University, Edinburgh, UK. 2003.

[Moukas, 1996] Moukas, A., Maes, P. *Amalthaea: An Evolving Multi-Agent Information Filtering and Discovery System for the WWW*. MIT Media Laboratory. Cambridge, USA, 1996.

[Neu, 2005] Institut Interfacultaire D'informatique. University of Neuchatel. http://www.unine.ch/info/clef/.

[Ngu, 1997], D., Wu, X. *SiteHelper: a localized agent that helps incremental exploration of the World Wide Web*. In 6th Int. WWW Conference, Santa Clara, CA, USA, 1997.

[OBIWAN, 1999]. *OBIWAN Project*. University of Kansas. 1999. http://www.ittc.ku.edu/obiwan/.

[Orwant, 1995] Orwant, J.: 1995, *Heterogeneous learning in the Doppelganger user model system*. User Modeling and User Adapted Interaction 4 (2), 107-130.

[Paiva y Self, 1995] Paiva, A. y Self, J.: 1995, *Tagus: a user and learner modeling workbench*. User Modeling and User Adapted Interaction 4 (3), 197-226.

[Paliouras et al., 1999] Paliouras, G., Karkaletsis, V., Papatheodorou, C. y Spyropoulos, C.: 1999, *Exploiting learning techniques for the acquisition of user stereotypes and communities*. In J Kay (ed.), UM99, User Modeling: Proceedings of the 7th International Conference, Springer-Verlag, 45-54.

[Pazzani et al., 1996] Pazzani, M., Muramatsu, J. y Bilsus, D.: 1996, *Syskill and Webert: Identifying interesting web sites*. Proceedings of the 13th National Conference on Artificial Intelligence, AAAI'96, Portly, OR, 54-61. http://www.ics.uci.edu/~pazzani/Syskill.html

[Pérez, 2000] Pérez-Carballo, J. and Strzalkowski, T. *Natural language information retrieval: progress report*. Information Processing and Management 36, 2000. p. 155-178.

[Pohl, 1998] Pohl, W.: 1998, *Logic-based representation and reasoning for shell systems*. St. Augustin, Germany.

[Popp y Lodel, 1996] Popp, H. y Lodel, D.: 1996, *Fuzzy techniques and user modeling in sales assistants*. User Modeling and User Adapted Interaction 5(3-4), 349-370.

[Quinlan, 1993] Quinlan, J. R. *C4.5: Programs for Machine Learning*. Kaufmann, 1993.

[RAE, 2003] Real Academia Española. *Diccionario de la Lengua Española*. En línea: http://www.rae.es.

[Rafter y Smyth, 2001] Rafter, R. y Smyth, B.: 2001, *Passive profiling from server logs in online recruitment environment*. Smart Media Institute, University College Dublin. Ireland. maya.cs.depaul.edu/~mobasher/itwp01/papers/rafter.pdf

[Raymond, 2005] Raymond J. Mooney. CS 378: *Intelligent Information Retrieval and Web Search*. http://www.cs.utexas.edu/users/mooney/

[Resnikoff, 1976] Resnikoff, H.L. *The national need for research in information science*. ST1 Issues and Options Workshop. House subcommittee on science. research and technology, Washington, D.C.. Nov. 3, 1976.

[Rich, 1979] Rich, E.: 1979, *User modeling via stereotypes*. Cognitive Science 3, 329-354.

[Rijsbergen, 1979] C. J. van Rijsbergen. *Information Retrieval*. Butterworths, London, second edition, 1979. http://www.dcs.gla.ac.uk/Keith/.

[Robertson, 1976] Robertson, S.E., Sparck Jones, K. *Relevance weighting of search terms*. Journal of American Society for Information Science. 27(3):129-46, 1976.

[Rocchio, 1966] Rocchio, J.J., *Document retrieval systems - optimization and evaluation*, Ph.D. Thesis, Harvard University. Report ISR-10 to National Science Foundation, Harvard Computation Laboratory (1966).

[RSS, 2005] *RSS at Harvard Law*. Syndication technology hosted by the Berkman Center. Editor: Dave Winer. En línea, julio de 2005.
http://blogs.law.harvard.edu/tech/directory/5/aggregators

[RSSfeeds, 2005] RSSfeeds. *The RSS, Atom and XML directory and resource*. 2005.
http://www.rssfeeds.com/readers.php.

[Rucker y Polanco, 1997] Rucker, J. y Polanco, M. J.: 1997, *Siteseer: personalized navigation for the web*. Communications of the ACM, 40(3), 66-73.

[Rui, 2003] Rui Alexandre P. P. da Cruz, R., García Peñalvo, F. J., Alonso Romero, L. *Perfiles de usuario: en la senda de la personalización*. Informe Técnico. DPTOIA-IT-2003-001. Enero, 2003.

[Salton, 1971] Salton, G.. *The SMART Retrieval System*. Prentice-Hall, 1971.

[Salton, 1983] Salton, G., McGill, M. J. *Introduction to Modern Information Retrieval*. Computer Science Series. McGraw-Hill, 1983.

[Salton, 1989] Salton, G. *Automatic Text Procesing – The Analysis, Transformation and Retrieval of Information by-Computer*. Addison-Wesley, 1998.

[Sánchez, 2002] Sánchez Fernández, L.; Delgado Kloos, C. *XML: el ASCII del siglo XXI*. NOVATICA, nº 158, pag. 5-9. 2002.

[Schafer, 2001] Schafer, J. B., Konstan, J., Riedl, J. *Electronic Commerce Recommendation Applications*. Journal of Data Mining and Knowledge Discovery, vol. 5 Nos 1/2, (2001) pp. 115-152.

[Schwab y Kobsa, 2002] Schwab, I. y Kobsa, A.: 2002, *Adaptivity through Unobstrusive Learning*. KI 3 (2002), Special Issue on Adaptivity and User Modeling.

[Selberg, 1995] Selberg, E., Etzioni, O. *Multi-service search and comparison using the MetaCrawler*. 4th Int. WWW Conference, 1995.

[Serradilla, 2005] Serradilla García, F. *Sistemas de Recomendación*. Escuela Universitaria en Ingeniería de Sistemas y Automática. UPM. Madrid. 2005.
http://www.sia.eui.upm.es/grupos/Ainfo2.pdf

[Shearin y Lieberman, 2000] Shearin, S. y Lieberman, H.: 2000, *Intelligent profiling by example*. MIT Lab, Cambridge, USA.

[SIRLE, 2003] Serradilla García, F., Teruel, J. *SIRLE: Sistema Inteligente de Recomendaciones sobre Literatura en Español*. 2003.
http://peterpan.eui.upm.es/index.html.

[Sleeman, 1985] Sleeman, D.: 1985, *A user modeling front-end subsystem*. International Journal of Man-Machine Studies 23, 71-88.

[Snow, 2005] Snowball. http://snowball.tartarus.org/.

[Sparck, 1975] Sparck Jones, K. *A performance yardstick for test collections*. Journal of Documentation, 31(4):266-72, 1975.

[Sparck, 1979] Sparck Jones, K. *Experiments in relevance weighting of search terms*. Information Processing and Management, 15(3):133-44, 1979.

[Sperberg, 1996] Sperberg-McQueen, C. M., Burnard L. *A gentle introduction to SGML*. Technical report, Text Encoding Initiative, 1996.

[Strachan et al., 2000] Strachan, L., Andersen, J., Sneesby, M. y Evans, M.: 2000, *Minimalist user modeling in a complex commercial software system*. User Model and User-Adapted Interaction 10 (2-3), 109-146.

[Strachan et al., 1997] Strachan, L., Andersen, J., Sneesby, M. y Evans, M.: 1997, *Pragmatic user modeling in commercial software system*. In: A. Jameson, C. Paris and C. Tasso, Proceedings of 6th International Conference on User Modeling, UM'97. Sardinia, Italy. Wien: SpringerWien. NewYork, 189-200.

[Tague, 1994] Tague-Sutcliffe, J. *The pragmatics on information retrieval experimentation, revisited.* Information Processing and Management, 28, 4, pp. 467-490. 1994.

[Thomas y Fischer, 1996] Thomas, C. y Fischer, G.: 1996, *Using agents to improve the usability and usefulness of the WWW*. 5th International Conference on User Modeling, 5-12.

[Vegas, 1999] Vegas Hernández, J. Tesis Doctoral. *Un Sistema de Recuperación de Información sobre Estructura y Contenido*, 1999.

[Voiskunskii, 1997] Voiskunskii, V. G. *Evaluation of search results: a new approach*. Journal of the American Society for Information Science. 48(2) 1997. p.133-142.

[Webb y Kuzmyez, 1996] Webb, G. y Kuzmyez, M.: 1996, *Feature based modeling: a methodology for production coherent, consistent, dynamically changing models of agent's competencies*. User Modeling and User Adapted Interaction 5 (2), 117-150.

[Winer, 2005] Winer, D. *RSS 2.0 Specification*. Syndication technology hosted by the Berkman Center. En línea, julio de 2005. http://blogs.law.harvard.edu/tech/rss

[Zipf, 1949] Zipf, G. K.. *Human Behavior and the Principle of Least Effort*. Addison-Wesley, 1949.

Anexo I. Lenguajes de definición de documentos

En la tesis se hace referencia a la recuperación de información en general y a la recuperación de información en la Web en particular. Dado que la mayoría de documentos de la Web se encuentran estructurados en formato HTML, y que el lenguaje XML será parte importante de la implementación del sistema propuesto, dedicaremos este Anexo I a introducir ambos lenguajes. También se hará una introducción a dos subconjuntos de XML, el primero denominado RSS, que se utiliza para sindicar noticias en la Web, y el segundo denominado Atom, con un cometido muy parecido al RSS.

Entre los **lenguajes de estructuración de documentos** más utilizados destacan tres: SGML, HTML y XML, [De la Fuente, 1998]. Estos lenguajes insertan etiquetas en los documentos para delimitar los elementos de estructura. Por una parte, diferenciaremos entre SGML y XML que son metalenguajes, y permitirán crear lenguajes de definición de distintos tipos de documentos, y las instancias de éstos como HTML, que es un lenguaje de definición de un tipo de documento concreto, es decir, una instancia de SGML.

SGML o *Standard Generalized Markup Language*, se definió en los años 80 por iniciativa de las editoriales de los EE.UU. Pretendía separar dos funciones principales del mundo editorial, que son los contenidos y la forma de presentar esos contenidos, en este caso los libros o publicaciones. El autor de una publicación sería el especialista en el contenido y la editorial es la que definirá cómo ha de presentarse ese contenido. SGML permitirá definir lenguajes concretos de marcado, es decir, se trata de un metalenguaje, un lenguaje o notación para definir lenguajes. SGML será por tanto un lenguaje que no tiene nada que ver con Internet ni con las redes [Sánchez, 2002]. Una buena introducción a este lenguaje se tiene en [Sperberg, 1996] y una referencia sobre su uso puede encontrarse en [Herwijnen, 1994].

AI.1. Hypertext Markup Language

HTML, acrónimo de "HyperText Markup Language", es un lenguaje simple de marcado que se utiliza para crear documentos de hipertexto para la Web, de los cuales describe su estructura y contenido.

"Aunque no es un lenguaje de descripción de estructura de uso general, su amplia difusión y el número de documentos estructurados según sus normas es tan grande que su consideración como lenguaje de definición de estructura se hace obligatoria" [Vegas, 1999].

El lenguaje HTML no sólo permitirá establecer hiperenlaces entre diferentes documentos, sino que describirá las páginas independientemente de la plataforma en que sean utilizadas. Es decir, un documento HTML contendrá toda la información necesaria sobre su estructura junto con la interacción con el usuario, y será el programa navegador que se utilice el responsable de asegurar que el documento tenga un aspecto coherente, independientemente del tipo de máquina desde donde se acceda al documento. De esta manera, todos los documentos compartirán un mismo aspecto y una única interfaz, lo que facilita enormemente su manejo por cualquier persona

HTML es un lenguaje muy sencillo que permite preparar documentos Web insertando en el texto de los mismos una serie de etiquetas, o *tag*s, que controlan los diferentes aspectos de la presentación y el comportamiento de sus elementos. Las etiquetas que controlan el comportamiento del documento son fragmentos de texto encerrados entre ángulos como <*etiqueta*>. Existen diferentes tipos de etiquetas: algunas controlan simplemente la presentación del texto del documento, otras la forma en que se incluirán imágenes, hiperenlaces con documentos o con diferentes partes del mismo documento. Como todo lenguaje, HTML está en constante evolución, apareciendo versiones nuevas con una cierta frecuencia. La última versión a junio de 2005 es la 4.01 [HTML, 1999].

AI.1.2. Evolución del Lenguaje HTML

El lenguaje HTML fue creado en 1991 por Tim Berners-Lee del CERN, con el único objetivo de servir como medio de transmisión de información en forma de hipertexto entre físicos. En 1993 Dan Connelly escribe la primera especificación SGML describiendo el lenguaje HTML. En 1994 el sistema había tenido tal aceptación que la especificación se había quedado ya obsoleta. Es entonces cuando nace el HTML 2.0 en un borrador realizado también por Dan Connelly. El crecimiento exponencial que comienza a sufrir el sistema lleva a organizar la "First International WWW Conference" en Mayo de 1994. Desde entonces el lenguaje ha seguido creciendo a medida que se difundía su uso y se descubrían nuevas necesidades. De este modo, a finales de 1993 se comienza a hablar de HTML+ propuesto por Dave Raggett, de HEP Labs, Bristol, que evoluciona a un nuevo borrador en Marzo de 1994 para la versión HTML 3.0 incorporando nuevas posibilidades

como la realización de tablas complejas, control de proceso de formatos e incorporación de expresiones matemáticas.

Actualmente, la mayoría de los documentos de la Web se almacenan y transmiten en HTML, lenguaje apropiado para elaborar de manera sencilla documentos con posibilidades de hipertexto y multimedia mediante un conjunto de etiquetas. Sin embargo tal simplicidad tiene un coste, que se refleja en una serie de limitaciones del HTML:

- No se permite que el usuario especifique su propias etiquetas o atributos, para parametrizar o cualificar semánticamente sus datos.
- No soporta la especificación de estructuras complicadas para representar esquemas de bases de datos o jerarquías orientadas al objeto.
- No se soporta ninguna clase de especificación de lenguaje que permita comprobar la validez estructural de los datos en el momento de su importación.

AI.2. Extensible Markup Language

Para responder a los requisitos que precisaba el sistema de publicación comercial a través de la Web y posibilitar su expansión en nuevos dominios, el "WWW Consortium" o W3C, creó un grupo de trabajo en 1996, presidido por Jon Bosak de *Sun Microsystems*, para desarrollar el "Extensible Markup Language" (XML) o *lenguaje de marcado extensible*, para las aplicaciones que requerían una funcionalidad no cubierta por HTML. Se trataba de construir un conjunto de especificaciones que permitieran utilizar de una forma fácil y directa las posibilidades que proporcionaba SGML. El objetivo principal era disponer de estructuras de datos autodescriptivas de complejidad y profundidad arbitraria, para ser utilizadas en las aplicaciones que lo requiriesen. La última definición de XML a junio de 2005 es la 1.1 [Bray, 2004].

Así, XML es un subconjunto de SGML adaptado específicamente para su uso en la Web, manteniendo todas las ventajas de SGML pero más fácil de aprender y de utilizar. Este subconjunto diferirá de HTML en tres aspectos fundamentales:

1. Se pueden definir nuevas etiquetas y atributos.
2. Las estructuras de los documentos pueden anidarse hasta cualquier nivel de complejidad.

3. Cualquier documento XML puede contener una descripción opcional de su gramática para ser utilizada por aquellas aplicaciones que precisen realizar una validación estructural.

El lenguaje XML no se desarrolló para crear páginas Web, sino para organizar el contenido de un documento mediante etiquetas semánticas. Sus objetivos de diseño fueron [Bray, 2004]:

- Debía ser directamente utilizable sobre Internet.
- Debía ser compatible con una amplia variedad de aplicaciones.
- Debía ser compatible con SGML.
- Debía ser fácil la escritura de programas que procesaran documentos XML.
- Sus características opcionales debían ser mínimas, idealmente cero.
- Los documentos XML deberían ser legibles y razonablemente claros.
- Un diseño de XML debería poderse preparar rápidamente.
- El diseño de XML debía ser formal y conciso.
- Los documentos XML deben ser fáciles de crear.

AI.2.1. Estructura de XML

Un documento XML contendrá exclusivamente información en forma de texto, nunca de otro tipo. En él se encontrarán etiquetas o delimitadores con un aspecto parecido a los empleados en HTML, pero con la libertad de elegir la denominación que se desee, normalmente reflejando el tipo de contenido que delimitan.

Un ejemplo de sencillo documento XML se muestra a continuación:

```
<persona>
      <nombre_completo>
            <nombre>Juan</nombre>
            <apellidos>Pérez Fernández</apellidos>
      </nombre_completo>
      <trabajo>fontanero</trabajo>
</persona>
```

En el ejemplo se observa que existe un elemento raíz denominado `persona` y dos elementos hijos del anterior denominados `nombre_completo` y `trabajo`. En un documento XML sólo puede existir un elemento raíz o "root".

Aunque no es estrictamente obligatorio, los documentos XML deben tener una declaración inicial, en ésta aparecerán atributos como la versión de XML: `version`, la codificación del texto del documento: `encoding`, y la autonomía del documento: `standalone`. Si el valor de `standalone` fuese "no" entonces se requerirá una definición externa para determinar los valores apropiados de ciertas partes del documento. Una declaración ejemplo es la siguiente:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes"?>
```

Los elementos XML pueden tener atributos. Un atributo será un par nombre-valor adjunto a una etiqueta de inicio. Los valores irán encerrados entre comillas. Por ejemplo, un elemento `persona` puede tener un atributo `nacida` con el valor `"23-06-1912"`:

```
<persona nacida="23-06-1912">
      Alan Turing
</persona>
```

AI.2.2. Documentos XML bien-formados

Cada documento XML, sin excepción, debe estar bien-formado. Esto implica que debe cumplir las reglas sintácticas especificadas en el lenguaje. Algunas de estas reglas son:

- Cada etiqueta o marca inicial "<" debe corresponderse con una etiqueta o marca final "</".
- Los elementos pueden estar anidados pero no superpuestos.
- Sólo puede existir un elemento raíz.
- Los valores de los atributos deben ir entrecomillados.
- Un elemento no puede tener dos atributos con el mismo nombre.
- Los comentarios y las instrucciones de proceso no pueden aparecer entre las marcas.

AI.2.3. Especificaciones XML

Además de la propia definición del lenguaje [Bray, 2004], podemos encontrar diversas especificaciones para XML, destacando las siguientes:

- **DTD** ("Document Type Definition"): definición del tipo de documento. Contendrá una definición formal de un tipo de documento y, a la vez, una especificación de la estructura lógica. Define tanto los elementos de una página como sus atributos. Esta notación, necesaria para definir un lenguaje de marcado concreto, fue estandarizada por el W3C en 1998[1]. El DTD del XML es opcional, en tareas sencillas no será necesario. Cuando un documento XML, además de estar bien formado, se ajusta una estructura y una semántica determinada por un DTD se dice que el documento XML es *válido*.

- **XML Schema**. Es una manera de definir tipos de documentos alternativa a DTD, resultando más potente, expresiva y completa que la anterior [Sánchez, 2002]. Fue especificada en mayo de 2001 por el W3C. La última versión de XML Schema está fechada a junio de 2005[2].

- **XSL** ("eXtensible Stylesheet Language"): define o implementa el lenguaje de estilo de los documentos escritos para XML. Permite modificar el aspecto de un documento. Está dividido en dos partes: "XSL Transformations" o XSLT[3] y "XSL Formatting Objects" o XSL-FO[4]. XSLT es una aplicación XML que permitirá definir transformaciones, en forma de reglas, para convertir un documento XML en otro documento XML. Por su parte, XSL-FO es una aplicación XML para definir el diseño preciso del texto en una página. Tiene elementos que representan páginas, bloques de texto en las páginas, gráficos y muchos otros.

- **Xpath**[5]. Es un lenguaje no XML utilizado para identificar o direccionar partes particulares de un documento XML. Como soporte para este objetivo principal, también proporciona facilidades básicas para manipulación de cadenas, números y booleanos. XPath obtiene su denominación por el uso que hace de una notación de

[1] *W3C Recommendation. http://www.w3.org/XML/1998/06/xmlspec.dtd.*

[2] *W3C Architecture Domain. http://www.w3.org/XML/2005/xsd-versioning-use-cases/.*

[3] *W3C Recommendation 16 November 1999. http://www.w3.org/TR/1999/REC-xslt-19991116.*

[4] *W3C Recommendation. http://www.w3.org/TR/xsl/slice6.html#fo-section.*

[5] *W3C Recommendation. http://www.w3.org/TR/xpath.*

caminos, como en las URLs, para navegar a través de la estructura jerárquica de un documento XML.

- **Xlink**[6]. Es una sintaxis basada en atributos para añadir enlaces a los documentos XML. Los enlaces podrán ser simples, como los habituales en HTML, bidireccionales, enlazando dos documentos en ambas direcciones, y multidireccionales, presentando varios caminos diferentes entre cierto número de documentos XML. Los documentos que se enlazan también pueden no ser XML.

AI.3. Rich Site Summary

"Rich Site Summary" o RSS es un formato basado en XML utilizado para compartir fácilmente el contenido de la Web. Ciertos contenidos están especialmente indicados para utilizar este formato: titulares de noticias, mercadotecnia, anuncios de trabajo y otros muchos, tales como los *blogs*[7] o diarios personales en la Web.

Un archivo RSS, también denominado un "feed" RSS o una fuente RSS, consiste en una lista de items, cada uno de los cuales contiene un título, una descripción y un enlace a una página Web. Normalmente el contenido completo está disponible por separado y es accesible mediante el enlace del fichero RSS.

Existen diferentes versiones de RSS, así se hablará de "Rich Site Summary", "RDF Site Summary" o de "Really Simple Syndication" dependiendo de la versión con la que estemos tratando. Una definición de "Syndication" es "distribuir una noticia a través de una coalición de empresas o sindicato para su publicación en cierto número de periódicos simultáneamente" [Moffat, 2003].

AI.3.1. Historia y Origen de RSS

Netscape introdujo en 1999 el formato RSS 0.90[8] para ofrecer un canal de contenidos en su portal "my.netscape.com". El objetivo era crear una plataforma y un vocabulario basado

---

[6] *W3C Recommendation. http://www.w3.org/TR/xlink/*

[7] *"No está en el diccionario de la RAE, pero el término blog corre de boca en boca, incluso ha sido palabra del año 2004. Básicamente un blog, weblog o bitácora es una dirección de Internet en la que el autor escribe, en forma de diario, sobre temas que le llaman la atención con enlaces a otras páginas webs que considera interesantes". Fuente: http://www.20minutos.es/noticia/181/0/blogs/weblogs.*

[8] *My Netscape Network. http://www.purplepages.ie/RSS/netscape/rss0.90.html.*

en RDF[9] para poder sindicar los datos en el portal de Netscape y en su navegador, ofreciendo una forma muy simple de publicar contenidos y permitiendo a los desarrolladores web obtener visitas gracias a los contenidos ofrecidos en "My Netscape". Posteriormente Netscape diseñó RSS 0.91[10] con la intención de estandarizar la versión anterior. Sin embargo, Netscape decidió no continuar el proyecto RSS, lo que provocó la aparición de diferentes formatos RSS. Básicamente se pueden dividir en dos grupos:

- **RSS 1.0**[11]: esta especificación, que se basa por completo en RDF, se publicó como propuesta en diciembre de 2000. Se elaboró a iniciativa privada en el grupo liderado por Rael Dornfest de O'Reilly. Se concibe para aprovechar las posibilidades de extensión que ofrece sin tener que actualizar las versiones de la especificación constantemente. Generalmente, los ficheros se guardan con extensión RDF.
- **RSS 0.92**[12]**, 2.0**[13]: Desarrolladas por Dave Winner, estas especificaciones están basadas en XML. El autor modificó el significado de RSS y le otorgó el significado de "Really Simple Syndication", o sindicación realmente simple, que da una idea de su objetivo: proporcionar una herramienta para publicar contenidos de una forma rápida y sencilla en la Web.

AI.3.2. RSS 0.92

Fue publicada en Diciembre del 2000 por Dave Winner. Esta especificación es totalmente compatible con RSS 0.91 ya que los nuevos elementos incorporados por esta versión son opcionales. Por tanto, un fichero RSS 0.91 es también un fichero RSS 0.92 válido.

Elementos obligatorios

En la parte superior del archivo, debe existir la etiqueta <*rss*> y la versión que cumple el documento XML. Subordinado a la etiqueta <*rss*> se encuentra el elemento <*channel*> o canal. Todo canal debe contener al menos los tres primeros elementos que se enumeran a continuación:

---

[9] *RDF (Resource Description Framework) es un lenguaje de marcado creado en 1997 por Ramnathan V. Guha. La especificación del lenguaje puede encontrase en http://www.w3.org/RDF/.*

[10] *Netscape Communications. http://my.netscape.com/publish/formats/rss-spec-0.91.html.*

[11] *RDF Site Summary (RSS) 1.0. http://www.rddl.org/rss10.htm.*

[12] *UserLand, RSS 0.92. http://backend.userland.com/rss092.*

[13] *RSS at Harvard Law. RSS 2.0 Specification. http://blogs.law.harvard.edu/tech/rss.*

- <*title*> -- El nombre del canal, será como los usuarios identifican el servicio.
- <*link*> -- Dirección Web que apunta al lugar identificado en <*title*>.
- <*description*> -- La frase que describe el canal.

Elementos opcionales:

- <*image*> -- Es un elemento XML que contiene varios sub-elementos, tres de ellos son opcionales y otros tres son requeridos:
  - <*url*> -- Dirección Web de un archivo de imagen que representa al canal.
  - <*title*> -- Describe la imagen.
  - <*link*> -- Es la dirección Web donde se encuentra el canal. En la práctica los elementos <*title*> y <*link*> de la imagen deberían ser los mismos que los del canal.
  - Los elementos opcionales de <*image*> incluyen <*width*> y <*height*>, que son números que indican el ancho y alto de la imagen en *pixels*. <*description*> contendrá un texto relacionado con el renderizado de la imagen en HTML.
- <*language*> -- Indica el idioma en que está escrito el canal. Esto permite a los agregadores de noticias agrupar los sitios con el mismo idioma, por ejemplo, en una única página. Para el idioma español, será "es".
- <*copyright*> -- Aviso de derechos de autoría para el contenido del canal.
- <*managingEditor*> -- La dirección de correo del editor del canal, la persona de contacto para cuestiones de edición.
- <*webMaster*> -- La dirección de correo del desarrollador del canal, la persona de contacto si existen problemas técnicos.
- <*rating*> -- "PICS[14] Rating" del canal. Es un control de contenido del canal.
- <*pubDate*> -- La fecha de publicación del contenido del canal. Todas las fechas en RSS estarán conformes a la especificación RFC 822[15].

[14] PICS, "*Platform for Internet Content Selection". "W3C Specification". http://www.w3.org/PICS/#Specs.*

- <*lastBuiltDate*> -- La última fecha en que se modificó el contenido del canal.
- <*docs*> -- Es una dirección Web que apunta a la documentación para el formato utilizado en el fichero RSS.
- <*textInput*> -- Es un elemento XML que sirve para que un usuario proporcione realimentación en forma de texto. Contiene varios sub-elementos que son requeridos:
  - <*title*> -- Es la etiqueta del botón a presionar para enviar el texto.
  - <*description*> -- Describe el area de texto donde se escribe.
  - <*name*> -- Nombre del objeto de texto.
  - <*link*> -- Dirección Web del script CGI[16] que procesa la entrada de texto.

- <*skipDays*> -- Es un elemento XML que puede contener hasta siete sub-elementos del día, que pueden ser: *Monday*, *Tuesday*, *Wednesday*, *Thursday*, *Friday*, *Saturday* o *Sunday*. Los lectores de noticias no leerán el canal durante los días especificados en este elemento.
- <*skipHours*> -- Es un elemento XML que puede contener hasta 24 sub-elementos de hora, que representan la hora en formato GMT[17]. Los lectores de noticias no leerán el canal durante las horas especificadas en este elemento.

[15] *Standard for the format of ARPA Internet text messages. http://asg.web.cmu.edu/rfc/rfc822.html.*

[16] *CGI, "Common Gateway Interface", es un protocolo para la transmisión de información hacia cierto compilador instalado en un servidor Web.*

[17] *GMT, "Greenwich Meridional Time", es la hora con referencia al meridiano de Greenwich.*

¿Qué es un ítem?

Este es uno de los elementos más importantes, ya que todos los ficheros RSS deben contener al menos un <*item*>. Un canal puede contener varios elementos <*item*>, cada uno de ellos apuntará a una noticia diferente, con una descripción opcional. El <*item*> estará compuesto por los siguientes elementos opcionales:

- <*title*> Es el título de la noticia.
- <*link*> Dirección Web que apunta a la noticia.
- <*description*> Es el resumen de la noticia.

Nuevos elementos respecto a la versión RSS 0.91:

- <*source*> -- Es un nuevo sub-elemento opcional del <*item*>. Es el nombre del canal RSS de donde proviene el item, se deriva del título.
- <*enclosure*> -- Es un nuevo sub-elemento opcional del <*item*>. Describe un objeto adjunto al item. Posee tres atributos requeridos. Así, *url* indicará donde se encuentra <*enclosure*>, *length* indicará cuanto ocupa en bytes, y *type* indicará el tipo que es según el estándar MIME[18].
- <*category*> -- Es un nuevo sub-elemento opcional del <*item*>. Posee un atributo opcional, *domain*, que identificará la categoría en una taxonomía.
- <*cloud*> -- Es un nuevo sub-elemento opcional del <*channel*>. Especificará un servicio Web. Su propósito es permitir la notificación de actualizaciones en el canal.

[18] *MIME, "Multipurpose Internet Mail Extensions" define la estructura de un mensaje de e-mail. Esto se consigue mediante campos en formato ASCII que identifican el contenido de diversas partes del mensaje.*

Un ejemplo de fichero RSS 0.92:

Se muestra a continuación un ejemplo simplificado de fichero RSS 0.92 que consta de un canal y un elemento *item*:

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<rss version="0.92">
  <channel>
       <title>ELPAIS.es</title>
       <link>http://www.elpais.es</link>
       <description>RSS de ELPAIS.es</description>
       <language>es-es</language>
       <item>
          <title>España consigue sus primeros oros en los Juegos del
          Mediterráneo</title>
          <link>http://www.elpais.es/articulo.html?xref=2005062</link>
          <description>La delegación española vivió el sábado una
          exitosa jornada de competición donde sumó un total de 23
          medallas.</description>
       </item>
  </channel>
</rss>
```

En este ejemplo puede observarse la declaración de documento XML, la indicación de la versión de RSS y varios elementos del canal como el título, el enlace, la descripción y el lenguaje del documento. Además se dispone de un item con su título, enlace y descripción correspondientes.

AI.3.3. RSS 2.0

Esta especificación fue publicada en Octubre de 2002 por Dave Winner. Es compatible con RSS 0.91 y RSS 0.92 . Por tanto, un fichero RSS 0.91 es también un fichero RSS 2.0 válido.

Nuevos elementos respecto a la versión anterior:

Se permiten crear tantos elementos como sean necesarios, siempre y cuando se hayan definido correctamente. El elemento <*category*> pasa a ser opcional en <*channel*>. Se han incorporado los siguientes:

- <*comments*> -- Es un nuevo sub-elemento opcional del <*item*>. Contendrá la dirección Web donde se encuentran los comentarios acerca del item.
- <*generator*> -- Es un nuevo sub-elemento opcional del <*channel*>. Indicará el programa que ha generado el archivo RSS.
- <*author*> -- Es un nuevo sub-elemento opcional del <*item*>. Especificará la dirección de correo del autor del item. Para un periódico o revista el autor es la persona que ha escrito el artículo.
- <*ttl*> -- Es un nuevo sub-elemento opcional del <*channel*>. Define el tiempo de vida del canal. Se expresa en minutos e indica cuánto tiempo puede guardarse el canal en memoria antes de ser refrescado.
- <*pubDate*> -- Es un nuevo sub-elemento opcional del <*item*>. Es una fecha que indica cuándo fue publicado el item.
- <*guid*> -- Es un nuevo sub-elemento opcional del <*item*>. Es un identificador unívoco del item. Si está presente, un agregador puede utilizarlo para decidir si el item es nuevo o no.

Un ejemplo de fichero RSS 2.0:

Se muestra a continuación un ejemplo simplificado de fichero RSS 2.0 que consta de un canal y dos elementos *item*:

```
<?xml version="1.0" encoding="utf-8" ?>
<rss version="2.0">
	<channel>
		<title>El Blog Salmón</title>
		<link>http://www.elblogsalmon.com/</link>
		<description>El Blog Salmón</description>
		<copyright>Copyright 2005</copyright>
		<lastBuildDate>Sun, 26 Jun 2005 01:36:04
		+0100</lastBuildDate>
		<generator>http://www.movabletype.org/?v=3.16</generator>
		<docs>http://blogs.law.harvard.edu/tech/rss</docs>
		<item>
			<title>Bolivia, sus recursos y las empresas
			extranjeras</title>
			<description>La situación en Bolivia, como se ha
			podido comprobar en las últimas semanas por la
			información emitida en la televisión, es
			complicada.</description>
			<link>http://www.elblogsalmon.com/2005/06/26-
			bolivia.php</link>
			<category>Entorno</category>
			<pubDate>Sun, 26 Jun 2005 01:36:04 +0100</pubDate>
		</item>
		<item>
			<title>Vuelven las nacionalizaciones</title>
			<description>El gobierno francés continúa con la
			privatización a la francesa, que es su proceso de
			vender partes de sus empresas estatales a inversores
			privados mientras mantienen control sobre el
			nombramiento de los altos ejecutivos y sobre la
			estrategia a seguir.</description>
			<link>http://www.elblogsalmon.com/2005/06/24-
			naciona.php</link>
			<category>Entorno</category>
			<pubDate>Fri, 24 Jun 2005 12:33:57 +0100</pubDate>
		</item>
	</channel>
 </rss>
```

Observamos la aparición de nuevos elementos respecto a la versión 0.92 de RSS, tales como <*generator*> y <*pubDate*>.

AI.4. Atom

Atom también es un sublenguaje XML. No se corresponde ni se basa en ninguna versión de RSS, pero tiene un formato muy similar a éste y tiene el mismo objetivo: permitir la distribución de contenidos y noticias de sitios web.

Se creó para resolver la confusión creada por la existencia de diversos estándares similares para sindicación (RSS y RDF). Sin embargo, más que resolver el problema de múltiples estándares, ha creado uno nuevo que convive con los anteriores. Está aún en proceso de desarrollo y ha recibido diferentes nombres, denominándose finalmente Atom. La última versión del estándar es Atom 1.0[19], publicada en julio de 2005.

Las mejoras que supone Atom respecto a RSS han hecho que su uso se extienda rápidamente a pesar de ser algo más complicado. Un documento Atom puede contener más información y más compleja. También es más consistente que un documento RSS.

Un ejemplo de Atom 1.0:

Se muestra a continuación un ejemplo simplificado de fichero Atom 1.0 que consta de una sola entrada. En Atom el elemento entrada o `<entry>` es equivalente al elemento `<item>` de RSS. Además cada entrada tendrá un título o `<title>`.

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom">
 <title>Ejemplo de entrada</title>
 <link href="http://example.org/"/>
 <updated>2003-12-13T18:30:02Z</updated>
 <author>
 <name>Juan J.</name>
 </author>
 <id>urn:uuid:60a76c80-d399-11d9-b93C-0003939e0af6</id>
 <entry>
 <title>Los robots potenciados con Atom corren furiosamente</title>
 <link href="http://example.org/2003/12/13/atom03"/>
 <id>urn:uuid:1225c695-cfb8-4ebb-aaaa-80da344efa6a</id>
 <updated>2003-12-13T18:30:02Z</updated>
 <summary>Texto del resumen.</summary>
 </entry>
</feed>
```

[19] *http://www.atompub.org/2005/08/17/draft-ietf-atompub-format-11.html*

Anexo II. Un Agregador Inteligente

Con el fin de situarnos en el contexto en que se llevaron a cabo los experimentos diseñados, se comentarán las características y principales funciones del programa desarrollado para implementar y probar el sistema NectaRSS, y que denominaremos con el mismo nombre por simplicidad.

La interfaz de usuario de NectaRSS dispone de un menú con todas las funciones que puede realizar el usuario y de una barra de botones con las acciones más importantes o usuales. El área de trabajo puede mostrar cualquier página web a la que se desee navegar y será ahí donde se muestren los titulares de noticias ordenados, puesto que dicho resumen es en sí mismo una página en HTML confeccionada por el sistema. Por último, como cualquier navegador estándar, se dispone de una barra de estado donde se informa al usuario del estado de carga de las páginas, entre otras informaciones. En la figura AII.1 se muestra el aspecto usual del programa:

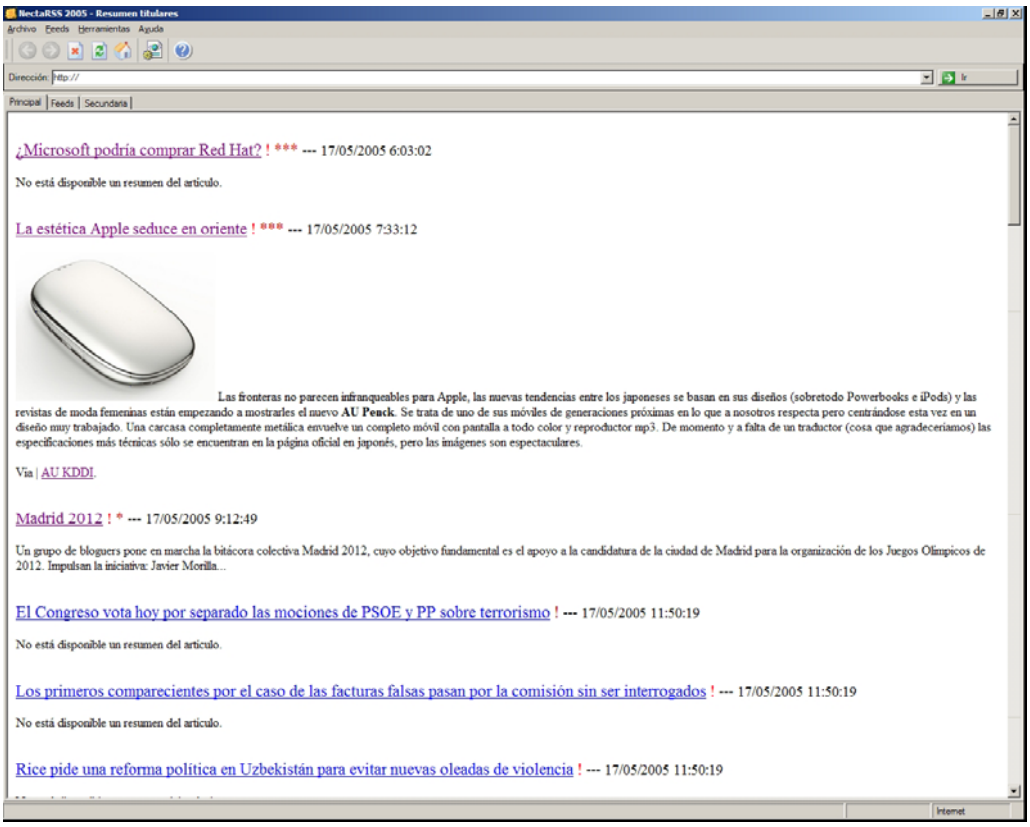


Figura AII.1. Aspecto principal del programa NectaRSS.

Será necesario gestionar de algún modo las fuentes de información a las que desea acceder el usuario, así como los titulares de cada una de esas fuentes. Para ello se diseñó otra pantalla, donde se muestran las distintas fuentes de información a las que se haya subscrito el usuario y los titulares de la fuente de información o "feed" que se encuentre seleccionado. Se podrá navegar por los titulares como en cualquier agregador de contenidos típico. El aspecto de la pantalla "Feeds" se muestra en la figura AII.2:

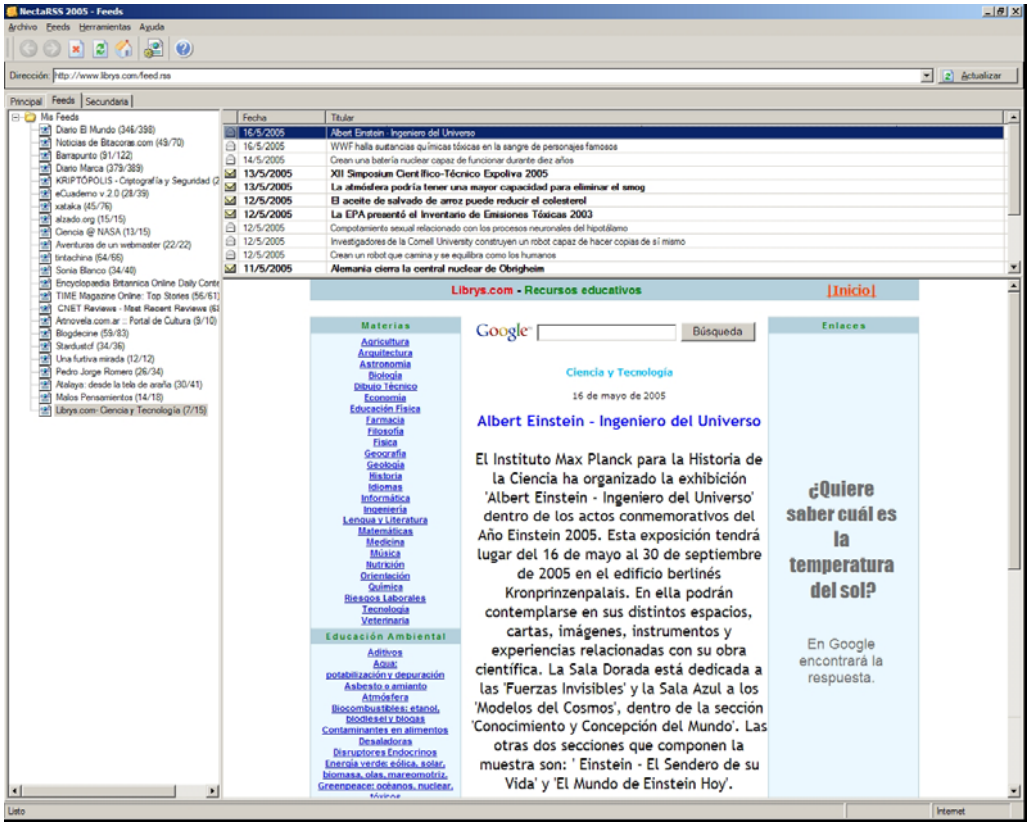


Figura AII.2. Gestión de "feeds" en el programa NectaRSS.

Para efectuar los experimentos, se dotó al programa de un modo de trabajo especial, el *modo experimento,* en el que los titulares de noticias no se muestran ordenados ni destacados, sino en un orden aleatorio y sin distinción alguna de su importancia. Así se ha considerado para no condicionar en modo alguno las decisiones del usuario experimental a la hora de elegir un titular u otro. En este caso el programa ofrecerá el aspecto de la figura AII.3.

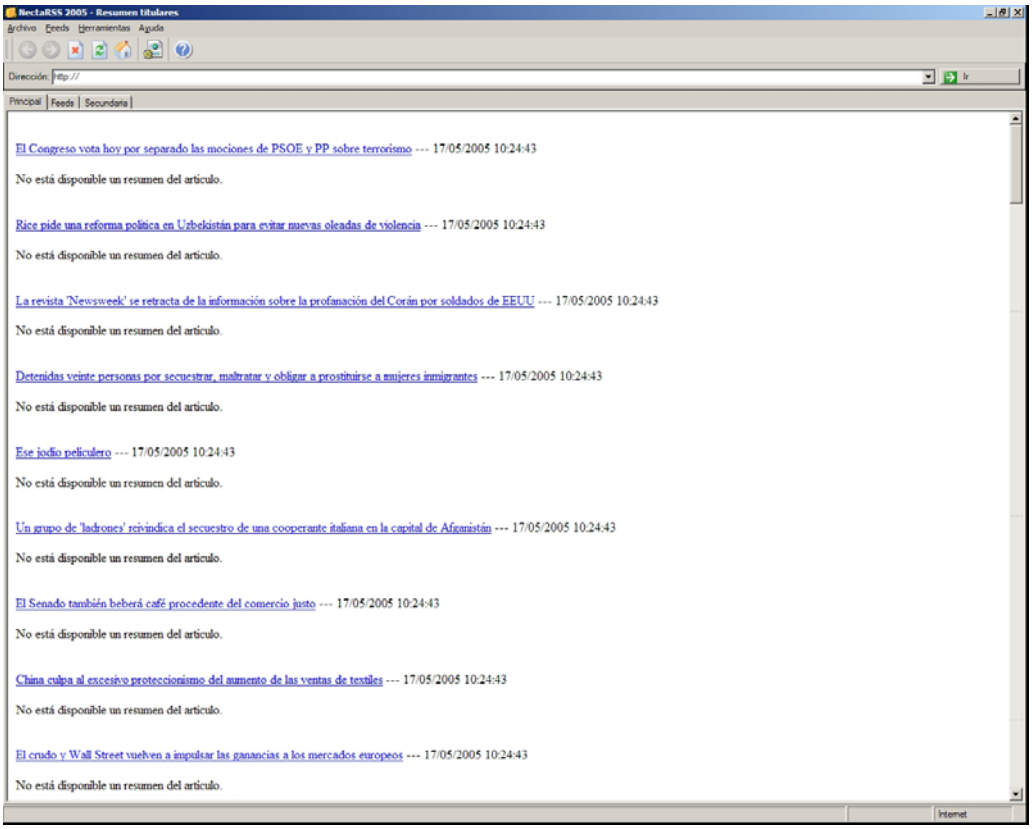


Figura AII.3. Aspecto del programa NectaRSS en *modo experimento*.

Adicionalmente, el programa genera una página web con las recomendaciones de titulares de cada sesión. Esta página se envía a un dominio creado expresamente este fin: http://www.neoyet.com. Se accede a ella pulsando el enlace denominado "Titulares del día". Se controló el número de visitas diarias para tener una idea relativa del interés de los visitantes ante la recomendación de noticias ofrecida. Si bien, tal resumen se encontrará personalizado para un usuario concreto, puede resultar interesante a personas que compartan intereses. El aspecto de esta página web es también muy sencillo y se refleja en la figura AII.4.

Figura AI.4. Aspecto de la página web para acceder a la recomendación de noticias elaborada por el programa NectaRSS.

A través de esta página web se solicitaron usuarios voluntarios para colaborar en la evaluación experimental del sistema. A éstos se les ofreció una versión experimental del programa NectaRSS, junto con instrucciones detalladas. Después de la realización de los experimentos cada usuario seleccionado devolvió la base de datos con los distintos resultados. Se comprobó la validez de los experimentos realizados y se utilizaron los valores numéricos obtenidos para evaluar la eficacia del sistema. En ningún caso se obtuvo información personal de ningún usuario, respetando estrictamente su privacidad.

AII.2. Fuentes de información o "feeds" utilizadas con el sistema

Se realizó la siguiente preselección de fuentes de información de la Web:

- Diario El Mundo (http://abraldes.net/feeds/elmundo.xml)
- Noticias de Bitácoras (http://bitacoras.com/noticias/index.xml)
- Barrapunto (http://backends.barrapunto.com/barrapunto.rss)
- Diario Marca (http://abraldes.net/feeds/marca.xml)
- Kriptópolis (http://www.kriptopolis.org/rss)
- eCuaderno (http://www.ecuaderno.com/index.xml)
- xataka (http://xataka.com/es/index.xml)
- alzado.org (http://www.alzado.org/xml/alzado.xml)
- Aventuras de un webmaster (http://www.maestrosdelweb.com/blog/index.rdf)
- tintachina (http://www.tintachina.com/index.xml)
- Sonia Blanco (http://www.filmica.com/sonia_blanco/index.xml)
- Enciclopedia Britanica (http://www.britannica.com/eb/dailycontent/rss)
- TIME Magazine (http://rss.time.com/web/time/rss/top/index.xml)
- CNET reviews (http://reviews.cnet.com/4924-5_7-0.xml)
- Artnovela (http://www.artnovela.com.ar/backend.php)
- Blogdecine (http://www.blogdecine.com/index.xml)
- Stardustcf (http://www.stardustcf.com/rdf.asp)
- Una furtiva mirada (http://furtivos.bloxus.com/rdf.xml)
- Pedro Jorge (http://www.pjorge.com/rss)
- Atalaya (http://atalaya.blogalia.com/rdf.xml)
- Malos Pensamientos (http://mp.blogalia.com/rdf.xml)
- Librys.com (http://www.librys.com/feed.rss)
- El Blog Salmón (http://www.elblogsalmon.com/index.xml)