

UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA

Recomendación de Contenidos Digitales basada en divergencias del lenguaje

Diseño, Experimentación y
Evaluación

Ángel Castellanos González

Directores: Ana García Serrano
Juan Manuel Cigarrán Recuero

Índice

Índice	1
Índice de Ilustraciones	7
Resumen	13
Abstract.....	15
Introducción.....	17
0.1. Motivación	18
0.2. Objetivos	20
0.3. Metodología.....	20
Parte 1: Estado del arte	23
Capítulo 1. Clases de Sistemas de Recomendación	25
1.1. Recomendación basada en Filtros Colaborativos	27
1.1.1. Basados en usuarios.....	27
1.1.2. Basados en Ítems	29
1.1.3. Basados en modelos	30
1.1.4. Problemas asociados a SR basados en Filtros Colaborativos.....	31
1.2. Recomendación basada en Contenido	32
1.3. Recomendación Híbrida	35
1.4. Comentarios finales	36
Capítulo 2. Modelado de Usuarios para Sistemas de Recomendación	39
2.1. Modelado de usuarios multiplataforma	41
2.2. Modelado de usuarios sensible al contexto.....	44
2.3. Modelado de usuarios teniendo en cuenta información social	45
2.4. Evaluación	46
2.5. Comentarios Finales	46
Capítulo 3: Medidas de similitud	49
3.1. Función coseno [133].....	49
3.2. BM25 [117].....	50
3.3. Otras medidas de similitud	51
3.4. Comentarios Finales	53

Capítulo 4. Técnicas de Representación de Información Textual.....	55
4.1. Técnicas no probabilísticas	55
4.1.1. VSM (Vector Space Model).....	55
4.1.1.1. Función de Pesado.....	56
Funciones de pesado locales	57
Funciones de pesado globales	57
4.1.2. LSA (Latent Semantic Analysis).....	58
4.1.2.1. Base teórica de LSA.....	59
4.2. Técnicas probabilísticas	60
4.2.1. pLSA (Probabilistic Latent Semantic Analysis).....	61
4.2.2. LDA (Latent Dirichlet Allocation).....	61
4.3. Comentarios finales	62
Capítulo 5. Tecnología: Herramientas y Recursos	65
5.1. Herramientas	65
5.1.1. <i>Solr</i>	65
5.1.2. <i>Cassandra</i>	66
5.1.2.1. Características	66
5.1.2.2. Modelo de datos	68
5.1.3. <i>GWT</i>	70
5.2. Recursos y Colecciones	71
5.2.1. Colecciones enfocadas en Recomendación	71
5.2.1.1. Colección Yahoo Webscope	71
5.2.1.2. Colección VideoLectures	72
5.2.1.3. Colección Delicious	74
5.2.1.4. Mendeley.....	76
5.2.1.5. MovieLens.....	77
5.2.1.6. Last.fm.....	80
5.2.2. Colecciones no enfocadas en Recomendación	81
5.2.2.1. Colección ImageCLEF - Photo Annotation	81
5.2.2.1.1. Información Visual.....	82
5.2.2.1.2. Información Textual.....	82
5.2.2.2. Tass.....	84

5.2.2.3.	Colección Medical Retrieval Task	85
5.2.3.	Colecciones propias	86
5.2.3.1.	Colección extraída de Storify	86
5.2.3.2.1.	Acceso	89
5.2.3.2.2.	Consultas	89
5.2.3.2.3.	Autenticación	89
5.2.3.2.4.	Errores	89
5.2.3.2.	Colección de Noticias Buscamedia	90
5.2.3.2.	Colección descargada de Pinterest	97
5.2.4.	Comentarios finales	100
Capítulo 6.	Evaluación de Sistemas de Recomendación.....	103
6.1.	Evaluación experimental.....	103
6.1.1.	Medidas de Exactitud	103
6.1.2.	Medidas de Precisión.....	105
6.1.3.	Medidas de Satisfacción	107
6.1.4.	Medidas de Diversidad y Novedad.....	107
6.1.5.	Medidas de Evaluación Online.....	110
6.1.6.	Comentarios finales	111
6.2.	Evaluación centrada en el usuario.....	112
6.3.	Comentarios Finales	114
Parte 2.	Propuesta de trabajo	115
Capítulo 7 - Descripción de la Propuesta		119
7.1.	Modelado de los contenidos	119
7.1.1.	Aproximación de modelado.....	119
7.1.2.	Almacenamiento de los modelos.....	120
7.2.	Modelado de los usuarios	122
7.3.	Aproximaciones de recomendación.....	123
7.3.1.	Recomendación utilizando únicamente los contenidos vistos por el usuario con anterioridad.....	123
7.3.2.	Recomendación utilizando los contenidos consultados anteriormente y los nuevos que el usuario está consultando.....	125
7.3.2.1.	Recomendación relacionada con una búsqueda de contenidos	125

7.3.2.2.	Recomendación relacionada con un contenido específico	126
7.4.	Colección para experimentación.....	128
7.4.1.	Elección de la colección	129
7.4.2.	Generación de usuarios.....	129
7.4.2.1.	Proceso de generación.....	130
7.4.2.2.	Desarrollo del <i>Ground Truth</i>	133
7.4.3.	Indexado de la colección	134
Capítulo 8.	Evaluación	137
8.1.	Evaluación de la Aproximación de Modelado.....	137
8.1.1.	Configuración de la evaluación	137
8.1.2.	Metodología de Evaluación	138
8.1.3.	Análisis de Relevancia Estadística	141
8.1.4.	Resultados.....	142
8.2.	Evaluación Extendida de la Aproximación de Modelado	146
8.2.1.	Metodología de Evaluación	147
8.2.2.	Análisis de Relevancia estadística.....	149
8.2.3.	Resultados.....	149
8.3.	Evaluación del Recomendador	151
8.3.1.	Evaluación experimental	151
8.3.1.1.	Metodología de Evaluación.....	152
8.3.1.2.	Justificación de uso	153
8.3.2.	Evaluación centrada en el usuario	153
8.3.2.1.	Configuración de la Evaluación	153
8.3.2.1.1.	Tareas planteadas	154
8.3.2.1.2.	Cuestionario	155
8.3.2.2.	Metodología de Evaluación.....	157
8.3.2.2.1.	Información recopilada mediante los logs.....	157
8.3.2.2.2.	Información recopilada mediante el cuestionario.....	157
8.3.2.3.	Resultados	158
8.3.2.3.1.	Logs.....	158
8.3.2.3.2.	Cuestionario	162
Capítulo 9.	Conclusiones y Trabajo Futuro	173

Anexos.....	177
Anexo A: Instalación y Configuración de Herramientas.....	177
A.1: <i>Apache Cassandra</i>	177
A.2: <i>Solr</i>	177
Anexo B: Manual de Usuario del Recomendador	178
B.1: Introducción	178
B.2: Descripción de la Evaluación.....	178
B.3: Utilización del Sistema.....	178
Alta de Usuario y Login.....	178
Presentación de las tareas.....	182
Realización de la tarea.....	183
Búsqueda de Contenidos	185
Exploración de Contenidos	188
Puntuación de un Contenido	189
Finalizar Tarea.....	190
Finalización de la Evaluación	190
Referencias	193

Índice de Ilustraciones

Figura 1 - Representación de la Función Coseno	50
Figura 2 - Comparativa - Consulta de complejidad baja	67
Figura 3 - Comparativa - Consulta de Complejidad Media.....	67
Figura 4 - Complejidad - Consulta de Complejidad Alta.....	68
Figura 5 - Modelo de Datos con <i>Columns</i> [27]	69
Figura 6 - Modelo de Datos con <i>Supercolumns</i> [27].....	70
Figura 7 - Ejemplo de Imagen y Etiquetas asociadas	83
Figura 8 - Ejemplo de Imagen e información de su propietario asociada	83
Figura 9 - Ejemplo de Imagen de la Colección y Descripción asociada	85
Figura 10 - Panel de creación de historias	87
Figura 11 - Ejemplo de historia social.....	88
Figura 12 - Secciones antes de su agrupación	93
Figura 13 - Secciones tras su agrupación	94
Figura 14 - Representación de la colección.....	96
Figura 15 - Interfaz de <i>Pinterest</i>	97
Figura 16 - Explicación Gráfica <i>Markedness</i>	106
Figura 17 – Recomendación utilizando el perfil de los usuarios.....	124
Figura 18 - Recomendación utilizando una búsqueda de contenidos.....	126
Figura 19 - Recomendación por niveles	127
Figura 20 - Recomendación basada en la noticia consultada	128
Figura 21 - Términos asociados a un usuario genérico	132
Figura 22 - Términos asociados a un usuario interesado en una categoría (economía, negocios y finanzas)	132
Figura 23 - Términos asociados a un usuario interesado en una subcategoría (crimen organizado).....	133
Figura 24 - Rangos de aceptación y rechazo	141
Figura 25 - Evaluación del modelado de las categorías	142
Figura 26 - Evaluación del modelado de la Subcategorías.....	143
Figura 27 - Representatividad del Modelado de Categorías.....	150
Figura 28 - Representatividad del Modelado de Subcategorías	150

Figura 29 - Cuestionario de Evaluación	156
Figura 30 - Respuesta Pregunta 1	162
Figura 31 - Respuesta Pregunta 2	163
Figura 32 - Respuesta Pregunta 5	164
Figura 33 - Respuesta Pregunta 6	165
Figura 34 - Respuesta Pregunta 9	166
Figura 35 - Respuesta Pregunta 10	167
Figura 36 - Pantalla de Acceso al Sistema	179
Figura 37 - Pantalla de Registro	180
Figura 38 - Usuario registrado correctamente	180
Figura 39 - Acceso al sistema.....	181
Figura 40 - Botón de Salir	181
Figura 41 - Ventana de Confirmación	181
Figura 42 - Interfaz de acceso a las tareas	182
Figura 43 - Interfaz del Recomendador (Primer Uso).....	183
Figura 44 - Interfaz del Buscador	183
Figura 45 - Interfaz de Recomendación	184
Figura 46 - Interfaz una vez completada la tarea.....	185
Figura 47 - Búsqueda de Contenidos mediante el Buscador	186
Figura 48 - Búsqueda de Contenidos mediante el Recomendador	187
Figura 49 - Exploración de Resultados mediante el Buscador	188
Figura 50 - Exploración de Resultados mediante el Recomendador.....	189
Figura 51 - Vista antes de realizar una puntuación	189
Figura 52 - Vista después de realizar una puntuación	190
Figura 53 - Puntuación de Recursos en la Interfaz de Búsqueda	190
Figura 54 - Vista de todas las tareas completadas	191
Figura 55 - Cuestionario	192

Ecuación 1	28
Ecuación 2	29
Ecuación 3	29
Ecuación 4	34
Ecuación 5	49
Ecuación 6	50
Ecuación 7	51
Ecuación 8	51
Ecuación 9	52
Ecuación 10	55
Ecuación 11	57
Ecuación 12	57
Ecuación 13	57
Ecuación 14	57
Ecuación 15	58
Ecuación 16	59
Ecuación 17	59
Ecuación 18	59
Ecuación 19	61
Ecuación 20	61
Ecuación 21	103
Ecuación 22	104
Ecuación 23	104
Ecuación 24	105
Ecuación 25	105
Ecuación 26	105
Ecuación 27	106
Ecuación 28	106
Ecuación 29	106
Ecuación 30	108
Ecuación 31	108
Ecuación 32	108

Ecuación 33	109
Ecuación 34	110
Ecuación 35	110
Ecuación 36	110
Ecuación 37	110
Ecuación 38	120
Ecuación 39	147
Ecuación 40	148
Ecuación 41	148

Tabla 1 - Problemas asociados a los SR	35
Tabla 2 - Comparativa entre los diferentes modelos de representación	63
Tabla 3 - Comparativa <i>MySQL</i> vs. <i>Cassandra</i>	66
Tabla 4 - Estadísticas de la Colección VideoLectures	74
Tabla 5 - Estadísticas de la Colección Delicious	75
Tabla 6 - Estadísticas de la Colección <i>Mendeley</i>	77
Tabla 7 - Estadísticas de la Colección MovieLens	78
Tabla 8 - Estadísticas de la Colección de <i>Last.fm</i>	80
Tabla 9 - Estadísticas de la Colección Photo Annotation	82
Tabla 10 - Estadísticas de la Colección TASS	84
Tabla 11 - Estadísticas de la Colección Medical	85
Tabla 12 - Resumen de las colecciones	102
Tabla 13 - Perfiles para la generación de usuarios	131
Tabla 14 - Análisis de los logs	159
Tabla 15 - Análisis de los logs normalizado por número de tareas	161
Tabla 16 - Respuesta Pregunta 3	169
Tabla 17 - Respuesta Pregunta 4	169
Tabla 18 - Respuesta Pregunta 7	170
Tabla 19 - Respuesta Pregunta 8	171

Resumen

Para enfrentar el problema del descubrimiento de información en grandes repositorios de datos, como Internet, surgieron los sistemas de recomendación. Estos sistemas ofrecen a los usuarios contenidos relevantes de acuerdo a su actividad previa. El funcionamiento de un sistema de recomendación consta de dos etapas: el modelado de la actividad previa de los usuarios, para identificar sus preferencias, y la recomendación de contenidos de acuerdo a estas preferencias. En el estado del arte de sistemas de recomendación, una de las aproximaciones seguidas es la de los sistemas basados en contenido, que recopilan los contenidos previamente vistos y recomiendan contenidos similares. En este contexto, el modelado de usuarios se plantea como una recopilación de términos de los contenidos vistos, ponderados de acuerdo a alguna aproximación.

Este tipo de sistemas (basados en contenido) presentan un problema conocido como sobre-especialización (i.e. recomendación de contenidos muy similares entre sí), afectando sensiblemente a la satisfacción de los usuarios con este tipo de sistemas. A este respecto, en este trabajo se propone una aproximación utilizando técnicas de divergencias del lenguaje como método para modelar la actividad previa de los usuarios. Con ello se espera poder identificar los términos originales de los contenidos consultados previamente por los usuarios, favoreciendo la recomendación de contenidos originales. De esta manera se espera mejorar la satisfacción del usuario en aspectos relacionados con la sobre-especialización

En el estado del arte, la evaluación en sistemas de recomendación habitualmente se ha llevado a cabo teniendo en cuenta únicamente la etapa de recomendación. Sin embargo, en este trabajo se ha tenido en cuenta también la etapa previa de modelado. Por ello, se ha planteado una metodología de evaluación para evaluar la representatividad de los modelos generados, utilizando una nueva métrica desarrollada para tal fin: representatividad normalizada. Los resultados han demostrado como la aproximación de modelado propuesta mejora el rendimiento de otras técnicas del estado del arte.

El modelado propuesto ha sido aplicado a varias aproximaciones de recomendación de contenidos, basadas en Recuperación de Información. Para la evaluación de estas aproximaciones se ha planteado una tarea de evaluación con usuarios reales, mediante la utilización de un prototipo de evaluación, desarrollado para tal fin. Los resultados muestran como el rendimiento del sistema de recomendación mejoran los de un sistema de recuperación de información en términos de satisfacción de usuario, en un contexto susceptible de ser afectado por la sobre-especialización de las recomendaciones.

Abstract

To face the information discovering problem in large data repositories, like Internet, it appeared the recommender systems in early 90s. These systems offer relevant contents to the users, according to their previous activity. The running of recommender systems consists of two steps: user modelling, to identify their preferences, and recommendation according to the identified preferences. In the state of the art of recommender systems, one of the most common approaches is the so-called content-based systems, which collects the contents previously viewed and recommends new similar contents. In this context, user modelling can be seen as a compilation of terms of the viewed contents, weighted according some approach.

This kind of systems (content-based) presents a problem known as over-specialization (i.e. recommendation of contents very similar among them), significantly affecting to the user satisfaction. Respect to this, this work proposes a modelling approach using divergences techniques as method to model the previous activity of the users. With this it is expected to identify the original terms of the contents previously viewed by users, allowing the recommendation of new original contents. In this way it is expected to improve user satisfaction in aspects related to over-specialization.

In the state of the art, the evaluation in recommender systems has been normally conducted taking into account only the recommendation step. However, in this work, the modelling step has been also taken into account. For this, an evaluation methodology has been proposed to evaluate the representativeness of the models, using a new measure, specially developed for this work: Normalized Representativeness. The results have demonstrated how the modelling approach improves the performance of other common techniques in the state of the art, in terms of representativeness.

The proposed modelling has been applied in several recommendation approaches, based on Information Retrieval. To the evaluation of these approaches it has been posed an evaluation task with real users. The results show that the performance of the recommender system improves the results of an information retrieval system in terms of user satisfaction, in a context susceptible of being affected by over-specialization.

Introducción

Desde hace unos años la web se ha convertido en el principal, y casi único, repositorio de información para todos los ámbitos del conocimiento. La gran cantidad de datos y relaciones que existen entre ellos representan un conocimiento impensable antes de la existencia de la web. Sin embargo, este tamaño representa también el principal problema a la hora de tratar de acceder a información en la web.

Para intentar superar el problema de la búsqueda de información en la web surgieron los motores de búsqueda, mediante los cuales los usuarios pueden realizar consultas y recibir un conjunto de documentos que, potencialmente, respondan a las consultas. Con el crecimiento ingente de la web los documentos devueltos por estos sistemas son miles o incluso millones, haciendo imposible que los usuarios los revisen todos para encontrar la información deseada. Además, para los usuarios es complicado encontrar nuevos contenidos de su interés debido, de nuevo, a la gran cantidad de contenido presente en la web.

Para intentar enfrentar este problema, a mediados de los 90, surgieron los sistemas de recomendación (SR) de contenidos [113]. Los SR ofrecen a los usuarios, de manera automática, contenidos que consideran que pueden ser de su interés. Con ello se espera poder descubrir al usuario nuevos contenidos, los cuáles no hubiese sido capaz de encontrar por sí solo, o le hubiese resultado muy difícil hacerlo. La idea básica tras los sistemas de recomendación no ha variado mucho respecto a la de aquellos primeros sistemas. Sin embargo, el contexto en el que se mueven es muy diferente: actualmente existe un número mucho mayor de contenidos, más aún desde el auge de la web 2.0 y las redes sociales; información que a su vez no están centralizada en una única aplicación, sino que están distribuida a lo largo de diferentes plataformas y herramientas.

En este contexto actual, la recomendación de contenidos es una de las principales áreas de investigación en el ámbito de la recuperación de información debido a la gran cantidad de líneas de investigación abiertas: obtención de información sobre usuarios y posterior procesamiento de la misma; extracción de patrones de comportamiento de los usuarios, enfoques de recomendación que puedan utilizar efectivamente el conocimiento adquirido o desarrollo de metodologías estandarizadas para la evaluación de SR.

0.1. Motivación

En la actualidad los SR suponen una de las principales áreas de investigación, favorecida por la necesidad y el interés de grandes sitios de Internet en este tipo de sistemas. Como curiosidad, *Netflix*, un servicio de alquiler de películas online, organizó en 2006 un concurso que premiaba con un millón de dólares a aquel sistema que mejorase sustancialmente su sistema de recomendación¹.

También a nivel académico el interés en SR ha crecido en los últimos años, impulsado por: conferencias relacionadas con este campo, como por ejemplo *The ACM Conference Series on Recommender Systems*²; la publicación de diferentes ediciones especiales en las principales revistas de investigación en el área de los sistemas de información, como *ACM Transactions on Intelligent Systems and Technology (ACM TIST) Special Issue on Social Recommender Systems*; o la publicación de libros monográficos como *Recommender Systems Handbook* [115] o *Recommender Systems: an Introduction* [67].

Sin embargo los SR presentan también una serie de problemas y limitaciones. El funcionamiento de los SR requiere de la existencia de una cantidad importante de información previa, ya sea sobre los contenidos o sobre la actividad de los usuarios. Especialmente importante también es la información de *feedback* generada por los usuarios de manera explícita o implícita, ya sea mediante revisiones, comentarios, puntuaciones u otro tipo de fuentes. Otro de los principales problemas, es el de la sobre especialización [148], los SR a menudo ofrecen contenidos poco originales a los usuarios, o lo que es lo mismo, la mayoría de las veces recomiendan el mismo conjunto de contenidos. Esto es debido a que su funcionamiento se basa en contenidos genéricos visitados por un gran número de usuarios y no reflejan aquellos más específicos, que pueden ser también de interés para los usuarios. El problema de falta de originalidad a la hora de recomendar contenidos afecta a la satisfacción del usuario respecto al sistema de recomendación.

Más allá de las limitaciones que puedan aparecer en su funcionamiento, hay una serie de funcionalidades que, a día de hoy, un sistema de recomendación debe ofrecer. En este sentido, Meyer et al en [95] presentan 4 aspectos en los que un sistema de recomendación debe ayudar al usuario: ayuda para **decidir** si un objeto será de su interés, ayuda para **comparar** entre varios ítems, para saber cuál será más interesante para el usuario, ayuda para **descubrir** ítems interesantes entre un gran volumen de datos y ayuda para **explorar** y encontrar objetos relacionados dado un contenido de su interés.

¹ <http://www.netflixprize.com/>

² <http://recsys.acm.org>

Con todo esto en mente: información previa, contexto actual, limitaciones y funcionalidades; en el trabajo planteado se pretende desarrollar un sistema de recomendación que aborde el modelado de contenidos para SR basados en contenido, tratando de evitar los problemas característicos de estos sistemas; entre ellos el de la sobre especialización. Para ello, apoyándose las conclusiones extraídas del análisis del estado del arte, se estudiará el proceso completo de recomendación, desde la extracción de la información más relevante de entre los contenidos, pasando por la representación de esta información hasta su posterior uso en diferentes aproximaciones de recomendación.

Para la extracción de la información más relevante de los contenidos se plantea el uso de modelos del lenguaje y *Kullback-Leibler Divergence* como función de pesado que defina la relevancia de cada término. Para la representación de la información asociada a los contenidos se ha realizado un estudio de las principales técnicas empleadas hasta la fecha, tanto algebraicas como probabilísticas (*Vector Space Model*, *Latent Semantic Analysis*, *Probabilistic Latent Semantic Analysis* y *Latent Dirichlet Allocation*). Mediante estas técnicas se busca desarrollar una aproximación de modelado de la actividad de los usuarios, teniendo en cuenta tanto los contenidos visitados como la relevancia que para el usuario tienen dichos contenidos.

En cuanto a la recomendación de contenidos, se pretenden definir y probar varias aproximaciones de recomendación de contenidos para estudiar cuál de ellas proporciona mejores resultados, en cuanto a satisfacción de usuario, tanto a nivel general como centrándose en el aspecto de la originalidad de contenidos recomendados. Estas aproximaciones son:

- Recomendación teniendo en cuenta los contenidos previamente consultados por el mismo usuario, etiquetados por relevancia.
- Recomendación teniendo en cuenta la información asociada a una serie de objetos (como por ejemplo documentos devueltos por un sistema de búsqueda).
- Recomendación de contenidos teniendo en cuenta la información del usuario y la información de un objeto que el usuario ha considerado relevante.

Estas aproximaciones pretenden abarcar los enfoques comúnmente utilizados en los SR; esto es, tener en cuenta la información del contenido de los objetos y tener en cuenta la información previa del usuario, para obtener un conjunto de resultados significativos de los que extraer conclusiones.

Otro de los aspectos más importantes en recomendación de contenidos, estudiados también en este trabajo, es la evaluación de la satisfacción de los usuarios. Pero, no solo la evaluación de los métodos de recomendación es importante, la aproximación utilizada para modelar los contenidos también es de importancia capital en el funcionamiento final del sistema. Por ello, tanto para el modelado como para las aproximaciones de recomendación propuestas se plantea su evaluación de acuerdo a técnicas tradicionales del estado del arte de la evaluación de este tipo de sistemas.

0.2. Objetivos

El objetivo de este trabajo se centra en la aplicación de una técnica basada en divergencias del lenguaje para modelar los contenidos en un sistema de recomendación, permitiendo mejorar el rendimiento del mismo frente a problemas clásicos de este tipo de sistemas, como la sobre-especialización. Para abordar este objetivo se pretende abordar los aspectos presentados a continuación:

- En primer lugar se define una aproximación de modelado de contenidos, aplicable también al modelado de perfiles de usuario, que permita extraer la terminología más representativa de los contenidos, que permita definirlos.
- Para la aplicación del modelado desarrollado se plantea la experimentación con diferentes técnicas de recomendación, basadas en técnicas de recuperación de información. Con ello se pretende cubrir diferentes situaciones en las cuales un SR puede resultar de utilidad para un usuario.
- Finalmente, se pretende definir un marco de evaluación que permita respaldar la propuesta. Esta evaluación estará basada en la metodología propuesta habitualmente en el estado del arte, tanto en términos de evaluación experimental como de evaluación centrada en el usuario.

0.3. Metodología

Para lograr la consecución de los objetivos planteados se va a seguir la metodología expuesta a continuación:

- Estudio detallado del estado del arte de cada una de las temáticas relacionadas con este trabajo, como son: Sistemas de Recomendación, Modelado de Usuarios, Medidas de Similitud y Representación de Información.
- Estudio del estado tecnológico actual en el contexto del trabajo planteado. Este estudio estará enfocado a estudiar las técnicas más interesantes para dar solución a los aspectos técnicos. Dentro de esta parte se hará especial atención al estudio de colecciones de experimentación, en el ámbito de la recomendación de contenidos.
- Definición de una técnica de modelado, basada en divergencias del lenguaje, para tratar de extraer la terminología original que mejor represente los contenidos textuales. Para ello se propone la utilización de una técnica de extracción de terminología, conocida como Kullback Leibler Divergence (KLD).
- Desarrollo de un sistema de recomendación de contenidos, basado en un sistema de búsqueda y consulta de noticias, que permita la aplicación del método de modelado desarrollado. En este punto se ha desarrollado un prototipo de dicho sistema, para la experimentación con el mismo.

- Definición de un marco de evaluación para la evaluación del método de modelado, así como de su posterior aplicación a la recomendación de contenidos. Para la evaluación del modelado se va a aplicar un enfoque experimental; mientras que para la evaluación del recomendador se desarrollará una evaluación centrada en el usuario.

En lo que sigue, el trabajo se estructura en dos partes principales. La primera de ellas lleva a cabo un estudio detallado del estado del arte, tanto de sistemas de recomendación como de modelado de usuarios para este tipo de sistemas; así como el modelado de ambos. Además se exponen y se reflexiona sobre las principales medidas de similitud y técnicas de representación de información textual utilizadas en este tipo de sistemas. Por último en esta primera parte, se presenta el contexto tecnológico en esta área. La segunda parte se reserva para la definición del trabajo planteado, la evaluación del mismo y las conclusiones extraídas del desarrollo de este trabajo.

Parte 1: Estado del arte

Capítulo 1. Clases de Sistemas de Recomendación

Los SR son las herramientas y técnicas desarrolladas para ayudar a los usuarios a encontrar contenidos que puedan ser de su interés en grandes repositorios de datos, mediante la sugerencia de ítems para que el usuario los consulte. En la actualidad existe una gran variedad de SR, muchos de ellos integrados dentro de otras aplicaciones, páginas, sistemas o plataformas; siendo algunos de gran popularidad como Youtube³, Amazon⁴ o Netflix⁵. En [99] se lista una amplia variedad de SR y sus dominios de aplicación, tanto académicos como empresariales.

El funcionamiento de un sistema de recomendación se puede ver como **la estimación del interés de un usuario sobre un determinado contenido**. Si el interés estimado es suficientemente alto, el contenido será considerado como relevante y le será recomendado al usuario. Una manera común de estimar este interés del usuario sobre un contenido es la utilización de puntuaciones de usuarios sobre dichos contenidos. Las puntuaciones de un usuario sobre un ítem pueden ser principalmente de 3 tipos: **unarias, binarias y escalares**. Las puntuaciones unarias hacen referencia a la existencia de una interacción directa de un usuario con un ítem (e. g. acceso, visionado, compra). Las puntuaciones binarias hacen referencia al interés explícito en un ítem por parte de un usuario (e. g. me gusta/no me gusta). Por último, las puntuaciones escalares tratan de reflejar el grado de conformidad de un usuario con un ítem (e. g. puntuaciones de 1 a 5). Otras formas de estimar el interés del usuario son, por ejemplo: acceso a un contenido o recomendación de un contenido por parte del usuario (e.g. retweets en *Twitter*).

Otra manera de entender el funcionamiento de los sistemas de recomendación es como una **particularización de un sistema de Recuperación de Información (RI)** [136]. En un sistema de RI, un usuario introduce una consulta (que refleja una necesidad de información) y el sistema le devuelve una serie de resultados relevantes. Un SR puede verse como un sistema de RI, el cual ofrece resultados relevantes sin esperar a que el usuario introduzca una consulta; utilizando como consulta las preferencias del usuario. También un SR puede entenderse como un sistema de RI que, dada una consulta de usuario, “re-rankea” la lista de resultados relativos a dicha consulta.

³ <http://www.youtube.com/>

⁴ <http://www.amazon.com/>

⁵ <http://www.netflix.com/>

Las aproximaciones de recomendación, en su funcionamiento, se enfrentan a una serie de retos. Alguno de los más importantes, según se expone en [114], son:

- **Escalabilidad** ante grandes conjuntos de datos. Debido al aumento de la complejidad computacional al incrementar el tamaño de los datos los SR pueden ver afectado su rendimiento. Soluciones que funcionen bien cuando son comprobadas experimentalmente puede que sean ineficientes o totalmente inaplicables sobre grandes conjuntos de datos reales [114]. El caso típico son sistemas que en su fase de investigación se demuestran eficientes resultan inútiles cuando se aplican a un ámbito comercial.
- **Proactividad** de los SR, es decir, ofrecer recomendaciones sin que sean requeridas por el usuario. Un par de ejemplos de este tipo son el sistema de Sae-Ueng et al [118], para la recomendación de compras en una tienda *online* o el de Puerta Melguizo et al [112] para la recomendación en un sistema de escritura. Éste último trabajo propone un sistema que, de manera pro activa, envía consultas a sistemas de búsqueda web basándose en el perfil del usuario y en lo que en ese momento el usuario esté tecleando. En este último trabajo se profundiza sobre un aspecto importante en la proactividad de un recomendador: que no sea molesto para el usuario, interrumpiéndole constantemente. Las interrupciones hacen que el usuario desvíe su atención de la tarea principal, perdiendo luego tiempo en retomarla. Melguizo et al defienden que si un SR interfiere demasiado en la actividad del usuario, este dejará de usarlo.
- **Privacidad** de los usuarios. Es posible que los usuarios no quieran compartir sus preferencias o su actividad con otros usuarios. Estas cuestiones son tratadas en el trabajo de Lam et al [80], así como en el de McSherry y Mironov [94], donde se estudia la relación entre las restricciones de privacidad y el funcionamiento de los SR.
- **Diversidad** de los contenidos recomendados, para favorecer el descubrimiento de nuevos contenidos a los usuarios. Sobre este aspecto se habla en el trabajo presentado en [145].
- **Integración de información** que ya se tiene sobre las preferencias de un usuario e información nueva que llega sobre la actividad reciente del usuario (*long term* y *short term*) [57] [101]. La división entre estos dos tipos de preferencias cada vez es más difusa y se está yendo hacia un enfoque híbrido que tenga en cuenta ambos tipos de información.
- **Integración de diferentes sistemas y plataformas** para enriquecer los perfiles de los usuarios [66][18]. Para integrar información multiplataforma es especialmente importante una estrategia de modelado de usuarios que lo permita (ver apartado de esta memoria, dedicado al Modelado de usuarios multiplataforma).

- SR diseñados para nuevos soportes y dispositivos que hagan uso de la información específica de contexto, como por ejemplo dispositivos móviles [114].

Aunque los SR se pueden clasificar con diversos criterios [99], en la literatura tradicional se han clasificado en tres tipos [9]: Recomendación basada en Filtros Colaborativos, Recomendación basada en Contenidos, Recomendación Híbrida. A continuación se detallan el funcionamiento de cada uno de estos tipos, así como las ventajas y desventajas asociadas y se exponen algunos ejemplos de sistemas representativos de cada tipo.

1.1. Recomendación basada en Filtros Colaborativos

Los sistemas de filtro colaborativo se basan en la clasificación de usuarios por un lado y de los contenidos por otro en conjuntos similares (“*neighbourhood*” o “vecindario”). Para generar estos conjuntos se utilizan las interacciones que los usuarios hayan llevado a cabo sobre los contenidos. Las interacciones se almacenan en una matriz de usuario-ítem, en la cual cada usuario es asociado con una fila, cada ítem con una columna, y cada interacción de un usuario sobre un ítem es reflejada en la celda de la matriz correspondiente.

Para agrupar usuarios o ítems similares se puede tener en cuenta dos tipos de información: información **explícita** e información **implícita**. La información explícita se refiere a la evaluación directa que los usuarios llevan a cabo sobre los contenidos (revisiones, comentarios, puntuaciones...). La información implícita por otro lado se refiere a las evaluaciones que el usuario realiza de manera indirecta sobre los contenidos, mediante el uso de los mismos (navegación, compras o descargas de productos, acceso a los contenidos, etc...). Linden et al [84] utilizan la información implícita asociada a los usuarios para mejorar el sistema de recomendación de Amazon.com. Baltrunas y Amatriain en [10] también estudian la utilización de información implícita de los usuarios para desarrollar un sistema de recomendación dependiente del tiempo.

Dentro de los SR basados en filtros colaborativos se distinguen tres aproximaciones: Basados en usuarios, Basados en ítems y Basados en modelos.

1.1.1. Basados en usuarios

Estos sistemas se apoyan en la idea de agrupar usuarios que hayan encontrado relevantes (o no relevantes) al mismo conjunto de ítems. Es de esperar que usuarios que en el pasado hayan compartido intereses los sigan compartiendo en el futuro y, por tanto, un nuevo contenido que un usuario considere relevante es susceptible de ser recomendado a sus usuarios “vecinos”.

Un ejemplo de este tipo de sistemas es el de Barman y Dabber, quienes proponen un sistema donde los contenidos recomendados son aquellos que son más populares entre el conjunto de usuarios similares [11]. Este sistema es tomado como referencia por Bellogín et al para desarrollar su sistema que expande el trabajo anterior mediante la utilización de información no binaria [15].

A la hora de desarrollar este tipo de sistemas es necesario establecer por un lado el tamaño K del “*neighbourhood*” o “vecindario” y por otro una medida de similitud para comparar usuarios. De cara a estimar K , Herlocker et al exponen en [59] que mientras mayor sea K , más información se tendrá respecto a un usuario o contenido y mejor funcionará el sistema. Sin embargo, a medida que crezca K también crecerá el tiempo de ejecución del recomendador, por lo que se debe llegar a un compromiso entre rendimiento y complejidad (entre 20 y 60 usuarios/ítems según su experimentación). Otros trabajos, como el de Bellogín et al [15], contradicen este planteamiento y defienden que, cuando se hace uso de métricas como la precisión y la cobertura, se obtiene mejores resultados con tamaños de K pequeños.

En cuanto a las medidas de similitud, estas se tratarán en detalle en la sección de esta memoria correspondiente a Medidas de Similitud. Por ahora simplemente se referirá $sim(u_i, u_j)$, como la similitud entre el usuario i y el usuario j . Una vez establecida la medida de similitud y el tamaño del “*neighbourhood*”, se puede predecir la puntuación de un usuario u sobre un contenido o ítem i , p_{ui} de acuerdo a la Ecuación 1:

$$p_{ui} = \frac{\sum_{\{v \in V_u | i \in S_v\}} sim(u, v) * r_{vi}}{\sum_{\{v \in V_u | i \in S_v\}} |sim(u, v)|} \quad \text{Ecuación 1}$$

siendo V_u el “*neighbourhood*” del usuario u , S_v el conjunto de ítems que el usuario v ha puntuado y r_{vi} las puntuaciones del usuario v sobre el ítem i . El problema con utilizar las puntuaciones es que, como se ha comprobado empíricamente, a menudo las puntuaciones de los usuarios no son representativas [24]. Este problema se puede ejemplificar de la siguiente forma:

Supóngase un usuario cuyas puntuaciones sobre los contenidos están siempre entre 1 y 5 y otro usuario que puntúe siempre entre 1 y 4; esto hará que exista una desviación entre las puntuaciones de ambos usuarios, a pesar de que ambos usuarios quieran expresar lo mismo con sus puntuaciones.

Para resolver este problema se puede modificar la Ecuación 1 teniendo en cuenta la desviación de las puntuaciones de la siguiente manera:

$$p_{ui} = \bar{r}_u + \frac{\sum_{\{v \in V_u | i \in S_v\}} \text{sim}(u, v) * (r_{vi} - \bar{r}_v)}{\sum_{\{v \in V_u | i \in S_v\}} |\text{sim}(u, v)|} \quad \text{Ecuación 2}$$

donde \bar{r}_u es la media de las puntuaciones del usuario u .

Una aproximación más avanzada para seleccionar el “*neighborhood*” de un usuario objetivo es presentada por Cleger-Tamayo et al en [34]. Cleger-Tamayo et al proponen utilizar un criterio de similitud basado en la idea de que si un usuario ha sido útil prediciendo las puntuaciones del usuario objetivo, entonces este usuario será también valido para predecir puntuaciones futuras. Por esto, para seleccionar a los mejores candidatos de pertenecer al *neighborhood*, se propone la utilización de la distancia entre las puntuaciones pasadas del usuario objetivo y aquellas que hubiesen sido predichas por cada uno de los usuarios candidatos, siendo el usuario o usuarios candidatos más recomendable aquellos cuyas predicciones se ajusten más a las puntuaciones realmente dadas por el usuario.

1.1.2. Basados en Ítems

Los SR de filtro colaborativo basados en ítems siguen un enfoque similar a los basados en usuarios a diferencia que en este caso lo que se agrupan son los ítems en lugar de los usuarios. Si un grupo de ítems ha sido considerado relevante para un usuario, un ítem similar al conjunto es probable que sea también relevante para este usuario. Dos ejemplos típicos de sistemas de este tipo son los planteados en [84] y [43].

De manera análoga al caso anterior se puede entender el proceso de recomendación de un ítem como la predicción de la puntuación del usuario sobre ese ítem de acuerdo a la siguiente fórmula:

$$p_{ui} = \bar{r}_i + \frac{\sum_{\{j \in S_u \cap V_i\}} \text{sim}(i, j) * (r_{uj} - \bar{r}_j)}{\sum_{\{j \in S_u \cap V_i\}} |\text{sim}(i, j)|} \quad \text{Ecuación 3}$$

siendo V_i el “*neighbourhood*” del ítem i , S_u el conjunto de ítems que el usuario u ha puntuado, r_{ui} las puntuaciones del usuario u sobre el ítem i y \bar{r}_i la media de las puntuaciones que los usuarios han dado al ítem i .

1.1.3. Basados en modelos

Las dos aproximaciones presentadas anteriormente comparten un **problema** que dificulta su uso en SR reales, su **complejidad** es muy alta. En el caso de los sistemas basados en usuarios la complejidad es de $O(N^2 \times M \times K)$ siendo N el número de usuarios, M el número de ítems o contenidos y K el número de usuarios que forman parte del “*neighbourhood*” de cada usuario. El caso de los sistemas basados en ítems es similar, la complejidad es de $O(M^2 \times N \times K)$.

Ambos métodos tienen una complejidad cuadrática, lo que no es admisible para sistemas reales con gran cantidad de datos. Para intentar solucionar este problema surgieron los sistemas basados en modelos, planteados por Breese et al en [23]. La idea base de estos sistemas es generar un modelo de los datos de manera off-line para facilitar el cálculo online de las predicciones.

En general, esta aproximación se basa en la aplicación de técnicas de **clustering**, ya sea sobre los usuarios o sobre los contenidos, para agrupar a los que sean similares. El clustering de usuarios se lleva a cabo para agrupar a usuarios que tengan preferencias similares. De manera análoga el clustering de ítems agrupa contenidos similares de tal manera que a la hora de recomendar, de nuevo solo se tendrán en cuenta los contenidos dentro de un mismo clúster. También se han presentado trabajos utilizando otras técnicas de clustering. Una de ellas es el uso de algoritmos de **clustering probabilísticos**, los cuales permiten que un usuario pertenezca a varios clusters, Candillier et al incluyen algunos en [24]. Este enfoque parece razonable ya que, por ejemplo, a un usuario le pueden gustar contenidos de varias temáticas, por lo que debería estar en varios grupos. Otra técnica de clustering propuesta es el uso de **jerarquías de clústeres**, Kelleher y Bridge desarrollan esta técnica en [69], la cual mejora tanto la precisión como la cobertura de otro tipo de algoritmos, ya que si en un clúster no se encuentran opiniones (o puntuaciones) sobre un contenido particular se busca en el clúster superior en la jerarquía.

Además de los modelos basados clustering, se pueden citar otro tipo de modelos que han sido utilizados como: modelos bayesianos, modelos basados en reglas de asociación, Latent Semantic Analysis, Latent Dirichlet Allocation, Singular Value Decomposition o Support Vector Machines. Para encontrar más información y ejemplos se puede consultar [24] y [66].

El uso de técnicas de clustering permite reducir la **complejidad** cuadrática de los métodos anteriores a $O(N \times M \times K \times L)$, siendo L el número de ejecuciones necesarias para generar el modelo de datos. Por otro lado los sistemas basados en modelos han demostrado mejor rendimiento que los sistemas basados en ítems o en usuarios a la hora de predecir puntuaciones [76]. Por el contrario, los sistemas basados en modelos necesitan de una fase previa de entrenamiento para generar el modelo, la cual, a menudo, es bastante costosa en términos computacionales. Este problema se acentúa más en SR en producción que necesitan ejecutar regularmente este entrenamiento, debido a la continua introducción de nuevos datos.

1.1.4. Problemas asociados a SR basados en Filtros Colaborativos

Una de las desventajas asociadas al uso de SR basados en filtros colaborativos es la **complejidad** [132]. Este problema se puede abordar mediante la creación de modelos que representen los datos de manera offline para facilitar la ejecución del sistema. Otro enfoque para abordar el problema es presentado por Koren y Bell en [77], en dónde se hace uso de métodos de factorización de la matriz de puntuaciones usuario-item para extraer la información semántica latente.

El otro problema principal asociado a los SR de filtro colaborativo es el conocido como *cold start* [123]. Este problema ocurre cuando se añade un nuevo elemento al sistema, ya sea un usuario o un ítem, ya que no se dispone de información asociada a él. La falta de información imposibilita el funcionamiento del sistema de recomendación ya que no se puede establecer el “*neighbourhood*” del objeto ni aplicar sobre él ninguna medida de similitud. El problema es más grave cuando el nuevo ítem es un usuario en lugar de un objeto. Si un nuevo ítem es añadido no será recomendado hasta que no se disponga de suficiente información sobre él, pero el sistema seguirá recomendando el resto de ítems a los usuarios. Sin embargo, si un nuevo usuario es añadido, el sistema no podrá recomendarle nada hasta que el usuario no interactúe lo suficiente con el sistema.

Para este problema, Nguyen et al investigan en [100] sobre el uso de información demográfica asociada al objeto tal como, edad, localización, ocupación, etc. para intentar proporcionar datos que permitan tenerlo en cuenta en el recomendador. Otra manera de afrontar el problema es requerir al usuario que puntúe un número mínimo de contenidos para que el sistema pueda comenzar a recomendarle otros nuevos contenidos. Sinha y Swearingen, en su estudio sobre seis sistemas diferentes de recomendación, llevado a cabo en [129], concluyen que los usuarios están dispuestos a proporcionar estas puntuaciones iniciales si con ello esperan recibir mejores recomendaciones. Relacionado con esto, Cremonesi et al investigan en [37] acerca del número mínimo de puntuaciones que un usuario debería llevar a cabo para que el sistema de recomendación tenga la precisión suficiente.

El problema del *cold start* también puede ser solucionado mediante la mejora del sistema de recomendación, incluyendo información del contenido de los objetos, dando lugar a SR híbridos (ver Sección 1.3).

Otro problema importante asociado a estos sistemas es el de la **escasez de datos**. A medida que el sistema incluya más variedad ítems a ser recomendados, cada vez será más complejo encontrar ítems en común entre usuarios. El caso extremo se daría si el conjunto de ítems con los que un usuario ha interactuado no tiene ningún ítem con el que otro usuario haya interactuado también, haciendo imposible la generación del *neighbourhood*. Una solución común a este problema es la creación de un sistema híbrido que aproveche el contenido de los ítems para así poder generar algún tipo de similitud entre usuarios.

Por el funcionamiento de este tipo de algoritmos los sistemas basados en filtro colaborativo no sufren tanto el problema de la **falta de diversidad** en las recomendaciones en tanta medida como los sistemas basados en contenido. Aun así este aspecto se ha estudiado en algunos trabajos, siendo uno de los más interesantes el de Boim et al [22]. En este trabajo se propone un nuevo algoritmo para diversificar las listas de recomendaciones que utiliza la categorización de los ítems recomendados: dada una lista de recomendaciones, se busca que contenga el mayor número de ítems posibles de diferentes categorías. Una aproximación muy similar para diversificar los resultados, llamada *Topic Diversification*, es llevada a cabo por Ziegler et al en [150].

Vargas et al también abordan el problema de la falta de diversidad en [136], adaptando modelos, métodos y métricas aplicados en RI para tal propósito al campo de los SR. Para ello plantean el concepto de *aspect space* como método para trasladar dos nociones clave de la diversidad en RI *document similarity* y *query intents* a sus correspondientes *ítem similarity* y *profile aspects*. En RI, *document similarity* mide lo parecidos que son dos documentos recuperados tras una consulta y *query intents* los diferentes *subtopics* relacionados con la consulta de un usuario; por su parte, los correspondientes propuestos en SR, *ítem similarity* hace referencia a los similares que son dos *ítems* recomendados y *profile aspects* son las diferentes temáticas que interesan a un usuario, reflejadas en su perfil. Con este contexto, se puede trasladar la diversificación de resultados en RI a SR: en RI dada una consulta, que incluye varios *query intents*, se trata de devolver documentos que cubran todos los *intents*, utilizando la similitud entre ellos para eso; análogamente, en SR se trata de devolver *ítems* que cubran todos los *aspects* reflejados en un perfil de usuario, aprovechándose de la similitud entre ellos.

1.2. Recomendación basada en Contenido

El funcionamiento de estos sistemas se basa en el contenido asociado a los ítems que el sistema recomienda y a la información disponible de los usuarios. El sistema recomienda ítems a usuarios si la **información asociada a los ítems** concuerda con las **preferencias de los usuarios**. Una preferencia representa una relación entre un determinado usuario y los datos, y más concretamente entre un usuario y un atributo asociado a los datos.

Las preferencias asociadas a los usuarios se pueden representar de dos maneras:

- Mediante un **perfil del usuario**, que se construye recopilando las preferencias del usuario, expresadas mediante, por ejemplo, la actividad previa del usuario o la opinión del mismo, recogida directamente a través de un cuestionario.
- Mediante un **modelo del usuario** que modela la información almacenada en su perfil, para extraer algún tipo de conocimiento de esta información que defina mejor las preferencias del usuario.

En [104] Pazzani y Billsus discuten sobre diferentes maneras de representar el contenido de los ítems, así como maneras de crear perfiles y modelos de usuario. Se puede equiparar el funcionamiento de este tipo de sistemas con un sistema de Recuperación de Información (RI). Al igual que en los sistemas de RI se tiene un corpus de documentos representados por su contenido textual y un conjunto de consultas que expresan una necesidad de información, ya sean *queries* en un sistema de RI o las preferencias de los usuarios (representadas en un perfil o modelo).

Si bien es cierto que las preferencias extraídas pueden ser utilizadas para mejorar los SR; también sufren ciertas limitaciones, siendo la principal de ellas la **cobertura**. La cobertura de una preferencia depende de la cobertura del atributo que define dicha preferencia, es decir, la cantidad de veces que aparece el atributo en los ítems. El problema reside en que la mayoría de los atributos no aparecen frecuentemente en los contenidos. Sirva para ejemplificar este problema la siguiente situación:

Dado un usuario, se ha extraído la siguiente preferencia: *Me gustan las películas de Spielberg*. Con esta preferencia se sabe que, en principio, siempre que se tenga una película de *Spielberg* debería ser recomendada al usuario. Sin embargo, si raramente aparecen películas de *Spielberg* a lo largo de los datos, tener identificada esta preferencia no sería de mucha ayuda. Una preferencia con mayor cobertura, por ejemplo: *Me gustan las películas de acción*, sería más recomendable en esta situación.

Debido al problema de la cobertura, los SR basados en contenidos que únicamente hacen uso de preferencias necesitan un gran número de detalles acerca de los usuarios. Una solución común es **extender** la cobertura con atributos similares al que es definido por la preferencia. Por ejemplo, dado el ejemplo anterior, se podría extender la preferencia *Me gustan las películas de Spielberg* con directores similares a *Spielberg* o con películas del mismo tipo a las películas de *Spielberg*. De nuevo entra en juego el concepto de similitud entre objetos, por lo que es necesario también en este tipo de SR plantear **medidas de similitud** entre contenidos. Este tema es tratado en la sección correspondiente (ver Capítulo de Medidas de Similitud).

Otro enfoque para el funcionamiento de los sistemas basados en contenido, diferente del uso de preferencias, es el uso de **clasificadores** como *Naive Bayes*, los cuales clasifiquen los ítems del sistema en relevantes o no relevantes para cada usuario tomando como base los ítems similares puntuados por el usuario previamente [1].

Los SR basados en contenidos cubren las limitaciones de los basados en filtros colaborativos derivadas de la falta de información ante la entrada de un nuevo usuario o ítem o ante la falta de puntuaciones coincidentes sobre estos. Por contra, estos sistemas tienen asociados nuevos problemas, el principal de ellos es la necesidad de tener una **descripción completa de los ítems y de los usuarios**. Los ítems raramente están estructurados correctamente en la mayoría de los dominios y a menudo su descripción está incompleta. Además se da la circunstancia de que para tener una correcta descripción de los usuarios es necesario pedir a los propios usuarios que indiquen sus requerimientos de manera estructurada y compleja. Esto representa un reto debido a su falta de especialización o a su rechazo a rellenar largos formularios.

Otro problema de estos sistemas es el conocido como **sobre especialización**, planteado por Zhang et al en [148], por el cual los sistemas recomiendan ítems muy parecidos a los que ya han sido considerados por los usuarios, dando lugar a una **falta de originalidad**. Para favorecer la **diversidad** de las recomendaciones se han planteado varias propuestas; una de las más interesantes es la de Zhang y Hurley, presentada en [146] y extendida en [147], donde proponen un método para maximizar la diversidad de los ítems recomendados, disminuyendo lo menos posible la precisión. Para ello proponen la utilización de una métrica de “disimilitud” entre dos ítems recomendados para diversificar las listas de resultados. La formulación de esta métrica se puede ver a continuación:

$$D(R) = \frac{1}{N(N-1)} \sum_{i \in R} \sum_{j \in R, j \neq i} d(i, j) \quad \text{Ecuación 4}$$

donde R es la lista de recomendaciones, N es el tamaño de la lista de recomendaciones, $d(i, j)$ es la “disimilitud” de los ítems i y j , la cual se define como $1 - sim(i, j)$.

Otros métodos de diversificación, como el uso de algoritmos genéticos, el filtrado de contenidos similares o el uso de medidas redundantes, se pueden consultar en [87] en la sección correspondiente.

Un aspecto relacionado con la sobre especialización es la **serendipia** (*serendipity*), el cual hace referencia a la recomendación de un ítem inesperado para el usuario [87]. No se debe confundir la serendipia con la **novedad** (*novelty*). La novedad hace referencia a la recomendación de un ítem desconocido pero que podría haber sido descubierto por el usuario (i.e. a un usuario que le gusten discos de *The Beatles* se le recomienda un disco de ellos que no conocía); sin embargo, la serendipia se refiere a la recomendación de contenidos que no podrían haber sido descubiertos por el usuario (i.e. a un usuario que le gusten discos de *The Beatles* se le recomienda un disco de los *Rolling Stones*).

Un trabajo interesante sobre serendipia se presenta en [62]. En este trabajo se implementa un clasificador que etiqueta los ítems como interesantes o no interesantes para un usuario, produciendo una lista ordenada de acuerdo a su relevancia. En la parte superior de esta lista estarán los ítems de más relevancia para el usuario (cuya puntuación será positiva) mientras que en la parte inferior estarán los menos relevantes (con puntuación negativa) y en la parte media de la lista estarán aquellos ítems cuya puntuación sea cercana a cero. Esto significa que estos ítems que tienden a cero apenas guardan relación con el perfil del usuario; por tanto, es lógico pensar que son desconocidos para el usuario, haciéndolos buenos candidatos de ser ítems con una alta serendipia.

1.3. Recomendación Híbrida

En las secciones anteriores se han expuesto las dos clases de SR más comúnmente utilizadas. Ambas presentan una serie de desventajas, los basados en filtros colaborativos el problema del *cold start*; mientras que los basados en contenido, a pesar de solucionar el problema del *cold start*, presentan nuevas desventajas, principalmente la necesidad de tener una extensa descripción de los contenidos, lo cual no es sencillo de conseguir. Algunos de los principales problemas relacionados con estos dos tipos de SR se recogen en la siguiente tabla:

Filtro Colaborativo	Basados en contenido
Complejidad de ejecución	Cobertura
<i>Cold-Start</i>	Sobre especialización
Escasez de Datos	Falta de Novedad/Serendipia
Falta de Diversidad	Falta de Diversidad
	Necesidad de gran cantidad de información

Tabla 1 - Problemas asociados a los SR

Para intentar aprovechar las ventajas de ambas clases de SR, a la vez que se solucionan las desventajas de ambos, nacen los **sistemas de recomendación híbridos**. Estos sistemas utilizan tanto información colaborativa como información basada en contenido para generar las recomendaciones. Si bien la idea básica detrás de este tipo de sistemas es simple, la clasificación y desarrollo de estos sistemas es compleja, debido a la amplia variedad de aproximaciones existentes para combinar la información colaborativa y basada en contenido.

Un primer método para desarrollar SR consiste en **ejecutar de manera separada e independiente dos recomendadores**, uno basado en filtro colaborativo y otro basado en contenido, y combinar los resultados de ambos. Uno de los primeros trabajos que siguen este método es el de Balabanovic y Shoham [9]; en su sistema *Fab* combinan los resultados haciendo que los ítems tengan que coincidir con el perfil de los usuarios y además tener puntuaciones altas del “*neighbourhood*” de dicho usuario. Pazzani et al [103] utilizan los perfiles de usuario para generar el “*neighbourhood*” de los usuarios y después poder aplicar un filtro colaborativo, aplicando también información demográfica de los usuarios. Vozalix y Margatis [139] aplican un método similar a este último pero en lugar de sobre los usuarios lo hacen sobre los ítems para agrupar aquellos contenidos que son similares y posteriormente aplicar un filtro colaborativo basado en ítems. En [139] también se estudia cómo mejorar el funcionamiento de diferentes algoritmos de recomendación utilizando información demográfica, obteniendo mejoras significativas. Han y Karypis [52] y Wang et al [141] se enfocan también en la idea de agrupar los ítems de acuerdo a su contenido para después aplicar técnicas colaborativas.

El otro método utilizado para desarrollar SR híbridos es seguir un enfoque basado en contenido y utilizar los datos generados por un sistema de recomendación basado en filtro colaborativo para enriquecer la descripción de los ítems usada para generar las recomendaciones. Un ejemplo de este método es el sistema de recomendación desarrollado por Jack y Duclaye presentado en [63].

Berkovsky et al [18] presentan otro enfoque de hibridación que utiliza un sistema de recomendación basado en contenido que utiliza un sistema basado en filtro colaborativo para generar recomendaciones cuando se tenga poca información (o no se tenga) sobre los contenidos consultados por el usuario.

1.4. Comentarios finales

Existen dos grandes tipos de SR si se tiene en cuenta el funcionamiento de los mismos. Por un lado están los sistemas basados en filtro colaborativo, que tratan de encontrar usuarios con perfiles similares a un usuario objetivo. Se basan en la idea de que si en el pasado a dos usuarios les han interesado contenidos similares, en un futuro seguirá siendo igual. Por otro lado, los sistemas basados en contenido en lugar de buscar similitudes entre usuarios, la buscan entre contenidos, normalmente texto. Se entiende que si a un usuario le han interesado un tipo de contenidos en el pasado le seguirán interesando en un futuro.

La principal diferencia que existe entre ambos tipos de SR es el tipo de información que utilizan. Los basados en filtro colaborativo, únicamente necesitan un registro de actividades previas de usuarios sobre un conjunto de ítems (i.e. películas puntuadas por un usuario), sin que sea necesario tener acceso al contenido de los mismos. Sin embargo, los basados en contenido, como su propio nombre indica, necesitan tener acceso al contenido completo de los ítems para poder llevar a cabo recomendaciones satisfactorias.

Este último aspecto constituye uno de los principales problemas a la hora de utilizar este tipo de sistemas, frente a los sistemas basados en filtro colaborativo. Además, pese a que en principio los SR son independientes del dominio, la necesidad de acceder al contenido puede limitar el rango de dominios en los que aplicar los sistemas basados en contenido (e.g. recomendación de canciones). Por otra parte los sistemas basados en filtro colaborativo tienen asociado otro problema relacionado con la información necesaria para su funcionamiento, el llamado *cold start*. El *cold start* se produce cuando llega un nuevo usuario al sistema del que no se tienen datos previos; hasta que no se produzcan similitudes entre las interacciones recogidas en su perfil y las recogidas en el de otros usuarios no se le podrá recomendar nada. Se tiene por tanto, para ambos tipos de sistemas, un problema común relacionado con la necesidad de tener una **cantidad crítica de información** para poder ofrecer recomendaciones satisfactorias

Aparte de esta consideración, relativa a las necesidades de información, ambos tipos de sistemas sufren otros tipos de problemas. Por un lado la ejecución de sistemas de filtro colaborativo tienen una complejidad alta, debido a la necesidad de calcular *a priori* similitud, bien entre iconos o bien entre ítems. Los sistemas basados en contenido no padecen este problema; sin embargo, puesto que la recomendación se basa únicamente en los contenidos previamente consultados, este tipo de sistemas tiende a recomendar contenidos iguales, o muy similares, a los ya consultados, dando lugar a problemas como la falta de originalidad, de novedad o de serendipia.

Debido a estos problemas, inherentes a ambos tipos de SR, la decisión de utilizar una u otra no es trivial, dependiendo de factores como: dominio de actuación, originalidad de las recomendaciones necesaria o, sobretodo, información disponible. Cabe destacar a este respecto que cada vez es mayor la hibridación de ambos tipos de sistemas para intentar solventar los problemas citados, aunque en este caso se presenta el problema de cómo combinar los resultados de ambos sistemas (filtro colaborativo y basado en contenido).

Más allá de estas consideraciones a la hora de ponderar qué tipo de SR utilizar, un factor común a tener en cuenta es el modelado de la información recogida en los perfiles de usuario, es decir, extraer patrones de comportamiento, reflejados en esta información. La dificultad de modelar los diferentes aspectos reflejados en los perfiles de usuario representa una de las principales dificultades a la hora de desarrollar un SR eficiente. En el capítulo siguiente se reflexiona acerca de diferentes temas relacionados con modelado de usuarios de cara a su utilización en sistemas de recomendación

Capítulo 2. Modelado de Usuarios para Sistemas de Recomendación

El **modelado de usuarios** es un campo de investigación multidisciplinar que aborda la construcción de modelos que reflejen el comportamiento de los usuarios en un contexto determinado. Los primeros trabajos en este campo, que suelen ser tomados como relevantes, son los desarrollados por Allen, Cohen y Perrault [106] y por Rich [116]. Tomando como inspiración estos trabajos originales, pronto se desarrollaron numerosos trabajos en este campo. Estos trabajos cubren diferentes áreas de investigación, como por ejemplo: Sistemas educativos, HCI [46], Sistemas Expertos y, el que aquí nos ocupa: Sistemas de Recomendación. Debido a esto, no es posible tener una aproximación de modelado de usuario genérico sino que, es específica de cada aplicación.

El objetivo principal del modelado es el de la adaptación de sistemas a las necesidades de los usuarios. Para esto, es necesario recopilar una gran cantidad de información sobre los usuarios; que representan las preferencias de los mismos. Toda esta información es referida típicamente como **modelo de usuario**. De manera más formal, modelo de usuario es definido por Wahlster y Kobsa [142] como:

Una fuente de conocimiento, en un sistema con capacidades de diálogo, que contiene, asunciones explícitas de todos los aspectos del usuario que podrían ser relevantes para el comportamiento dialogante del sistema.

Existen dos tipos de modelos, de acuerdo a la estrategia seguida para su generación: **empíricos**, los cuales se basan en observaciones empíricas de la actividad del usuario; y **analíticos**, los cuales intentan simular el proceso cognitivo que ocurre durante la interacción del usuario y el sistema. La manera común de obtener toda esta información necesaria para modelar a los usuarios es mediante el *feedback* recibido de los mismos. Este *feedback* puede obtenerse de manera **implícita**, observando la actividad del usuario y recopilándola de manera transparente a éste, o de manera **explícita**, preguntándole directamente al usuario por sus gustos (mediante puntuaciones, comentarios, críticas, formularios, etc...) [70]. Independientemente del tipo de información o la obtención de la misma, para la generación de estos modelos, hay una serie de consideraciones que deben ser tenidas en cuenta [74]:

- Independencia del dominio
- Expresividad
- Rápida adaptación
- Extensibilidad
- Importación de información externa
- Manejo de información distribuida
- Manejo de aspectos de privacidad

Existe un gran número de aproximaciones para modelar esta información, dependiendo del contexto de utilización. Algunas de las más destacables son recopiladas en los estudios presentados en [86] y en [21]. Centrándose únicamente en el contexto que aquí nos ocupa, la recomendación de contenidos, el modelado de usuarios es un campo de investigación mucho más acotado, siguiendo básicamente el enfoque empírico para su generación. Teniendo esto en cuenta, las aproximaciones de modelado dependerán tanto del área de aplicación como de la aproximación de recomendación seguida (basadas en contenido o filtro colaborativo).

El área de aplicación de los SR es muy amplia, incluyendo recomendación de páginas webs, música, películas, productos en comercio electrónico, o destinos turísticos entre otros. El modelado de usuarios en cada uno de estos ámbitos puede seguir aproximaciones diferentes, tanto por las características dependientes del dominio como por la estrategia de recomendación a seguir [50]. Es decir, en general, diferentes aproximaciones de recomendación requieren diferentes estrategias de modelado.

Los modelos de usuarios planteados para los SR basados en contenido consisten en una recopilación de aquellos rasgos que caracterizan los contenidos consultados por el usuario. Estos modelos pueden ir desde un simple conjunto de palabras ponderadas de acuerdo a algún criterio (e.g. frecuencia de aparición, TF-IDF) a modelos más complejos teniendo en cuenta aspectos temporales o de polaridad acerca de los contenidos.

Por otra parte, los modelos utilizados en SR colaborativos van más enfocados a recoger las interacciones de los usuarios sobre los contenidos (e.g. acceso, puntuación, compra,...), sin tener en cuenta sus características. En estos sistemas los usuarios se suelen modelar como un vector de interacciones sobre los contenidos. Para ello, para cada usuario se tiene un vector de igual longitud al total de contenidos. En este vector se almacena en cada posición el valor (e.g. acceso/no acceso, puntuación de 1 a 5, número de consultas,...) correspondiente al ítem al que refiere esa posición, almacenando un valor nulo para los ítems que no se tengan valores asociados.

La hibridación de diferentes modelos de usuarios de recomendación también ha sido planteada, como base de los SR híbridos. Un ejemplo de este modelado híbrido es el trabajo de Kim et al [70], donde se plantea un modelo de usuario basado en contenido, mediante la creación de patrones de aparición de términos en los contenidos visitados previamente. Para enriquecer el modelo de un determinado usuario objetivo se utilizan patrones extraídos de usuarios con similares preferencias: si en los modelos de los usuarios similares aparecen patrones parecidos a los ya reflejados en el modelo del usuario objetivo, se añaden a dicho modelo estos nuevos patrones encontrados. La utilización de este modelado en un sistema de recomendación consigue mejorar los resultados de un sistema basado únicamente en contenido.

Hay otros tipos de modelos, como por ejemplo, para los SR demográficos, donde el modelado consiste en una lista de características demográficas de los usuarios (e.g. localización, sexo, edad...). Modelos basados en ontologías u otras estructuras que soporten inferencia también han sido planteadas [108], siendo *WordNet*⁶ uno de los recursos más utilizados [42]. Otro tipo de propuesta interesante es la de Lakiotaki et al [78], quienes proponen un modelo de usuario híbrido que incorpora técnicas de Análisis de Decisiones Multi-Criterio (MCDA) a un modelo basado en filtro colaborativo, transformando la tarea de recomendación en un problema de decisión de entre un conjunto de contenidos. Para esto, para cada usuario, se toman sus puntuaciones sobre diferentes criterios relacionados con un contenido, en este caso películas, para posteriormente aplicar un algoritmo UTA (Utilités Additives) [130]

A menudo se ha planteado la idea de crear un *framework* común a varios enfoques de recomendación, que sea capaz de abstraerse del funcionamiento específico de cada uno de ellos. Sin embargo, no se trata de un problema de fácil solución; Godoy et al proponen en [50] un sistema de recomendación que permite abstraerse de los tipos de modelos asociados a los diferentes enfoques de recomendación, permitiendo generar agentes que mezclen distintas técnicas de recomendación.

Godoy et al plantean un marco de trabajo organizado en capas que intercambian información entre ellas, utilizando un formato común. Cada una de las capas maneja una etapa en el proceso de recomendación. De esta forma se pueden utilizar diferentes enfoques y métodos de recomendación, construyendo una nueva capa que procese la información, siguiendo el método requerido y pase la información a la siguiente capa.

El modelado de usuarios para SR aborda otros aspectos, y no solo el tipo de recomendador en el cuál se van a utilizar. Estos aspectos pueden verse como ejes transversales, los cuales pueden ser aplicados a mejorar los modelos presentados sea cual sea su tipo. Algunos de estos aspectos, expuestos en las siguiente subsecciones, son: Modelado de usuarios multiplataforma, Modelado de usuarios sensible al contexto o Modelado de usuarios teniendo en cuenta información social.

2.1. Modelado de usuarios multiplataforma

Recientemente se ha comenzado a plantear el problema del **aislamiento** de los modelos de usuario. En la actualidad es común que lo usuarios utilicen diferentes plataformas o sistemas a través de internet. Cada uno de estos sistemas almacena su propio modelo del usuario, recopilando únicamente la información de la que dispone del usuario en el sistema. Por ejemplo, un usuario puede tener almacenada información acerca de su experiencia laboral en su perfil de *LinkedIn*, información sobre su vida privada en su perfil de *Facebook*, e información pública, más formal, en su perfil de *Twitter*.

⁶ <http://wordnet.princeton.edu/>

Esta situación da lugar a una serie de problemas, planteados por Viviani et al en [138]:

- Pérdida de información sobre los usuarios. Solo una parte del conocimiento total del usuario está disponible, aquella que el usuario haya introducido en el sistema.
- Al ser modelos específicos de cada aplicación, el usuario no suele tener control sobre lo que el sistema almacena sobre él. Con un modelado multiplataforma, independiente de los sistemas, al usuario le sería más sencillo acceder a su modelo. Al desconocer qué tipo de información se almacena, y no poder acceder a ella, también se plantea el problema de la privacidad. Un usuario puede querer que cierto tipo de información sobre él no sea recopilada.
- A la hora de recopilar información sobre un usuario, tanto de manera implícita, recogiendo la información de manera automática de los diferentes perfiles, como explícita, pidiéndole al usuario que la introduzca, cierto porcentaje de información podría estar duplicada en diferentes sitios. Este problema es aún mayor si la información se recopila de manera explícita, ya que se obliga al usuario a introducir la misma información cada vez que se registra en un nuevo servicio o plataforma.

De cara a mejorar el modelado de los usuarios, sería interesante poder disponer de toda la información que sobre él se tiene en las diferentes plataformas en las que está presente. En este sentido en [138] se puede consultar una detallada revisión de trabajos sobre el modelado de usuarios para entornos multiplataforma, desde los primeros sistemas, desarrollados en los años 70, hasta los más modernos. También en [138] se reflexiona sobre los requerimientos para conseguir una correcta integración en sistemas multiplataforma. Para esto se deben cumplir una serie de requisitos, entre otros:

- **Precisión del perfil del usuario:** La información recopilada a lo largo de las plataformas debe ser lo más precisa posible y así evitar incluir en el perfil multiplataforma información ruidosa.
- **Independencia del dominio:** Las diferentes plataformas a través de las cuales se recopilan los datos pueden pertenecer a diferentes dominios. El modelado multiplataforma debe ser capaz de gestionar estos diferentes dominios de conocimiento e integrarlos en el modelo común.
- **Expresividad:** El sistema debe ser capaz de expresar tantos tipos de patrones acerca de los usuarios como sea posible.
- **Adaptación rápida:** El sistema debe tener la habilidad de adaptarse rápidamente a la aparición de nuevos usuarios, así como a la aparición de diferentes perfiles del mismo usuario. De igual modo, el modelo único se debe adaptar a los cambios que se puedan producir en los distintos perfiles de los que recoge información.

- **Importación de datos externos:** Los datos importados de otras plataformas deben ser fácilmente integrables con los datos que ya se tienen del usuario en el sistema.
- **Escalabilidad:** El sistema debe soportar la gestión, de manera eficiente, de tantos usuarios como concurran en un momento dado en el sistema.
- **Extensibilidad:** La capacidad de integrar una gran variedad de técnicas de modelado debe estar presente en el sistema, mediante el uso de interfaces de programación (API).
- **Privacidad:** El sistema debe contemplar la utilización de políticas de privacidad a la hora de usar y compartir información del usuario, presente en diferentes plataformas.

A su vez, existen una serie de problemas asociados a los sistemas multiplataforma, relacionados principalmente con la interoperabilidad entre sistemas. En [138], se recogen algunos de ellos:

- **Interoperabilidad Sintáctica:** Hace referencia a la posibilidad para dos o más sistemas de comunicarse e intercambiar información entre ellos, superando las diferencias entre ellos a *nivel de aplicación*, es decir, que compartan un formato de datos común que pueda ser entendido por ambos.
- **Interoperabilidad Semántica:** Refiere a la capacidad de superar diferencias entre sistemas a *nivel de conocimiento*. Es decir que los sistemas sean capaces de reconocer conceptos o entidades sintácticamente similares que estén presentes en ellos.

Un posible problema a la hora de modelar información obtenida desde varias plataformas es: ¿Qué representación de datos elegir? ¿La de una de las fuentes? ¿Una mezcla de la representación de cada fuente? ¿Una nueva representación? Con relación a esto, De Luca et al presentan en [40] un modelado para representar información multiplataforma basado en una ontología desarrollada para tal fin. Mediante esta ontología, los autores definen un conjunto de relaciones que enlazan los diferentes atributos existentes en cada una de las plataformas, en las cuales se encuentran los perfiles a modelar. La ontología definida también tiene en cuenta el aspecto multilingüe; esto es, los perfiles del usuario pueden estar en diferentes idiomas. Para ello, De Luca et al proponen utilizar EuroWordNet⁷ para enlazar diferentes representaciones del mismo atributo en diferentes idiomas.

⁷ <http://www.illc.uva.nl/EuroWordNet/>

2.2. Modelado de usuarios sensible al contexto

Tradicionalmente los SR han utilizado modelos de usuarios muy simples, los cuales simplemente recopilan información previa de los usuarios sin ninguna otra consideración. Sin embargo, este tipo de modelos no tienen la capacidad de capturar el conocimiento asociado a la información recopilada sobre los usuarios [2]. A este respecto, Mobasher [96], establece el problema de la **acción situada** (*situated action*) del usuario, mediante el cual un ítem que puede ser relevante para un usuario en un determinado contexto puede no serlo en otro contexto diferente.

Para hacer frente al problema de la acción situada, una aproximación común es utilizar la información de contexto disponible para dividir al perfil del usuario en sub-perfiles para cada concepto. Siguiendo este enfoque Said et al [120] utilizan una aproximación para generar sub-perfiles de usuarios según cuándo y en qué lugar (en el cine o en casa) el usuario haya visto la película que recomienda. Estos sub-perfiles son posteriormente utilizados en un sistema basado en filtro colaborativo, consiguiendo mejorar el rendimiento tanto en términos de precisión como de cobertura.

Al abarcar un contexto más amplio, la creación de los modelos de los usuarios se vuelve más compleja. Se deben tener en cuenta aspectos como: **heterogeneidad** de los datos debido a que estos proceden de diversas fuentes, **relación** entre los tipos de información contextual, necesidad de manejar **información histórica** del usuario, o **errores** en los datos, entre otros.

Este tipo de cuestiones son abordadas en la revisión del estado del arte publicada por Bettini et al [19]. En este trabajo se hace un recorrido por las aproximaciones de modelado de usuarios teniendo en cuenta el contexto: se comienza con las más primitivas basadas en *keywords* o en *object-roles*, también conocido como *Object Role Modeling* (ORM) [56], que proveen una notación gráfica para definir formalmente los requerimientos de cada uno de los atributos contextuales (lugar, fecha...); a continuación se tratan los modelos espaciales, los cuales dan especial relevancia a los atributos espaciales relacionados con el usuario (localización del mismo); finalmente se aborda la utilización de modelos basados en ontologías, que explotan el poder de representación y razonamiento de conceptos del cual proveen las ontologías, aplicándolo sobre conceptos relacionados con el contexto de un usuario.

Bettini et al también abordan aspectos relacionados como la incertidumbre de la información de contexto o la hibridación de las diferentes aproximaciones, citadas con anterioridad en esta misma sección.

2.3. Modelado de usuarios teniendo en cuenta información social

La utilización de plataformas sociales cada vez está más extendida; por ello, son una importante fuente de información a la hora de recopilar información sobre los usuarios y mejorar el modelado de sus preferencias. Con la utilización de este tipo de información se persigue resolver diferentes problemas asociados al modelado de usuarios para SR, como son: la **escasez de datos**, a este respecto Ma et al en [89] aplican factorización probabilística de matrices empleando información sobre las relaciones entre los usuarios en redes sociales; la **elección de usuarios** similares a un usuario objetivo [82]; o el **incremento de la confianza** de los usuarios en el sistema [11] [15] puesto que, como Sinha y Swearingen establecen en [129], aunque los objetos recomendados por amigos tienden a ser menos novedosos los usuarios confían más en estas recomendaciones.

La información utilizada normalmente son las **relaciones** sociales entre usuarios [17]; sin embargo, otros tipos de información, como el **etiquetado social**, también se han utilizado: Bellogín et al en [14] realizan un análisis de diferentes fuentes de información, entre las que incluye la utilización de etiquetado social; Konstas et al utilizan también información de etiquetas [75] para su sistema de recomendación musical; mientras que Shepitsen et al hacen uso de clustering jerárquico para la recomendación en sistemas de etiquetado social (folksonomías) [127].

Popescu y Grefenstette en [109] presentan otra forma de utilizar etiquetado social, extraído de *Flickr*: las anotaciones de las fotografías se utilizan para identificar lugares y recomendar destinos relacionados. Para llevar a cabo la recomendación, en primer lugar, identifican los usuarios con preferencias similares teniendo en cuenta los destinos visitados que tengan en común e identifican los lugares turísticos más significativos de cada uno de los destinos mediante Wikipedia. Con esta información, dado un usuario que desee visitar un nuevo destino, el sistema le recomienda lugares para ese destino teniendo en cuenta lo que han visitado otros usuarios similares.

A pesar de que la información social es una importante fuente de conocimiento acerca de los usuarios, su utilización no es trivial. Esto es debido principalmente a dos factores: por un lado a la gran cantidad de información disponible que hay que procesar y, por otro, a la elevada frecuencia de actualización de estos datos. Estos aspectos son tenidos en cuenta en el trabajo de Diaz-Aviles et al [41] donde se presenta una aproximación para analizar el *stream* de *Twitter* y modelar perfiles de usuario en tiempo real. Otro trabajo relacionado es el de Das et al [38], quienes tratan el manejo de grandes cantidades de información, procedentes de *Google News*⁸, para la generación de perfiles de usuario.

⁸ <https://news.google.com/>

2.4. Evaluación

Si bien es cierto que la evaluación en el campo del modelado de usuarios ha sido un aspecto ampliamente estudiado [29] [30], ya sea siguiendo un enfoque experimental o un enfoque basado en el usuario. Sin embargo, en el campo de la recomendación de contenidos la evaluación de la etapa de modelado apenas ha sido motivo de estudio. En su lugar la evaluación de los SR ha estado enfocada a medir la calidad de las recomendaciones ofrecidas, lo cual implícitamente también medirá de alguna manera el rendimiento de la etapa de modelado.

No obstante, sería interesante conocer con detalle cuál es el rendimiento del proceso de modelado, independientemente de su posterior aplicación en la recomendación de contenidos. Este análisis sería especialmente interesante en los SR basados en contenido, puesto que gran parte de su rendimiento es debido a la capacidad de modelar los contenidos. Algún trabajo interesante en este contexto ha sido llevado a cabo, como por ejemplo el trabajo de Shmueli-Scheuer et al [128], donde se propone una metodología para la evaluación de la aproximación de modelado que proponen en el trabajo.

2.5. Comentarios Finales

El modelado de usuarios aborda la tarea de representar la información que se tiene sobre un usuario, obtenida mediante sus interacciones, para facilitar el tratamiento de la misma. Se trata de una tarea esencial en un SR, ya que serán estos modelos generados los que se utilicen para encontrar contenidos susceptibles de ser interesantes para un usuario.

Dado que el modelado de usuarios está estrechamente relacionado con la recomendación de contenidos en sí, la información utilizada para modelar a un usuario dependerá o influirá en la aproximación de recomendación que se vaya a utilizar. Mientras que para los SR basados en contenido el modelado utiliza los contenidos consultados para extraer características representativas, para los SR colaborativos se tiene en cuenta otro tipo de información como puntuaciones de los usuarios sobre los contenidos o datos de navegación de los usuarios.

La representación de la información modelada es otro aspecto importante que se debe tener en cuenta, de cara a su utilización para recomendación. Los trabajos a este respecto van desde la utilización de un vector de componentes, el cual almacena cada una de las características que se desean modelar junto con su “importancia”, hasta las representaciones más complejas, que utilizan ontologías para modelar las diferentes características interesantes, así como las relaciones entre las mismas.

En el modelado, al igual que ocurre con las aproximaciones de recomendación, se ha planteado la hibridación de aproximaciones basadas en contenido y en filtro

colaborativo para, de esta manera, cubrir las posibles limitaciones que puedan tener por separado. Algunos trabajos interesantes a este respecto como [70] o [78] son citados en este capítulo.

Más allá de este tipo de consideraciones, la elección del tipo de modelado a utilizar queda definida por el tipo de información que se modela según el tipo de aproximación de recomendación utilizada. Las posibles variaciones entre las aproximaciones de modelado vendrán únicamente marcadas por la manera en que la información es tratada y representada.

En este capítulo también se han abordado otros aspectos de interés que deben ser tenidos en cuenta a la hora de llevar a cabo el modelado de los usuarios. Estos aspectos son independientes tanto del tipo de información utilizada como de la representación de la misma y tienen que ver con la agregación de información a los modelos generados. Más concretamente se trata la utilización de información presente en diferentes plataformas en las que el usuario tenga un perfil, la utilización de información relacionada con el contexto en el cual el usuario realiza las interacciones con el sistema (i.e. localización, fecha, dispositivo...) y la utilización de información que se tiene del usuario en redes sociales. Si bien es cierto que este tipo de información es muy interesante, a menudo el acceso a la misma resulta imposible, o muy difícil. Además dado que este tipo de información viene dada por un flujo constante de datos, una vez se accede el tratamiento de la misma es muy complejo. Este impedimento hace que, a menudo, este tipo de información no sea utilizado en sistemas reales más allá de para propósitos de investigación.

Por último se ha reflexionado sobre la evaluación del modelado de usuarios, centrándose en el ámbito de la recomendación de contenidos. A este respecto es destacable la importancia de llevar a cabo algún tipo de evaluación sobre el modelado para comprobar su rendimiento independientemente de los resultados obtenidos posteriormente en la recomendación de contenidos.

Capítulo 3: Medidas de similitud

Los SR, sea cual sea el tipo de representación utilizado, persiguen agrupar contenidos o usuarios en conjuntos **similares**. En este punto, el problema queda reducido a definir qué es similar, dados dos conjuntos de datos y sus representaciones. Para ello se debe definir una **medida de similitud** que establezca cuantitativamente lo cercanos (o lejanos) que están dos representaciones entre sí. Las medidas de similitud más utilizadas en SR son detalladas a continuación.

3.1. Función coseno [133]

La función coseno es una medida utilizada para medir la similitud entre dos representaciones vectoriales. Para ello, dados una pareja de vectores, se mide el coseno del ángulo que forman las representaciones espaciales de los mismos. Observando la Figura 1, se tienen 2 vectores de características (Vector 1 y Vector 2) representados en un espacio bidimensional. Dichos vectores que forman un ángulo α entre ellos, cuyo coseno, $\cos(\alpha)$, representa la similitud (o disimilitud) entre ambos.

Generalizando el ejemplo, esta función puede ser aplicable a vectores n-dimensionales. La representación de la función coseno como medida de similitud entre los vectores X e Y n-dimensionales se muestra en la Ecuación 5:

$$sim(X, Y) = \cos(\alpha) = \frac{\sum_{i=1}^N X_i * Y_i}{\sqrt{\sum_{i=1}^N X_i^2} * \sqrt{\sum_{i=1}^N Y_i^2}} \quad \text{Ecuación 5}$$

Donde el coseno tendrá valor máximo ($\cos(\alpha) = 1$) cuando las dos representaciones X e Y sean iguales.

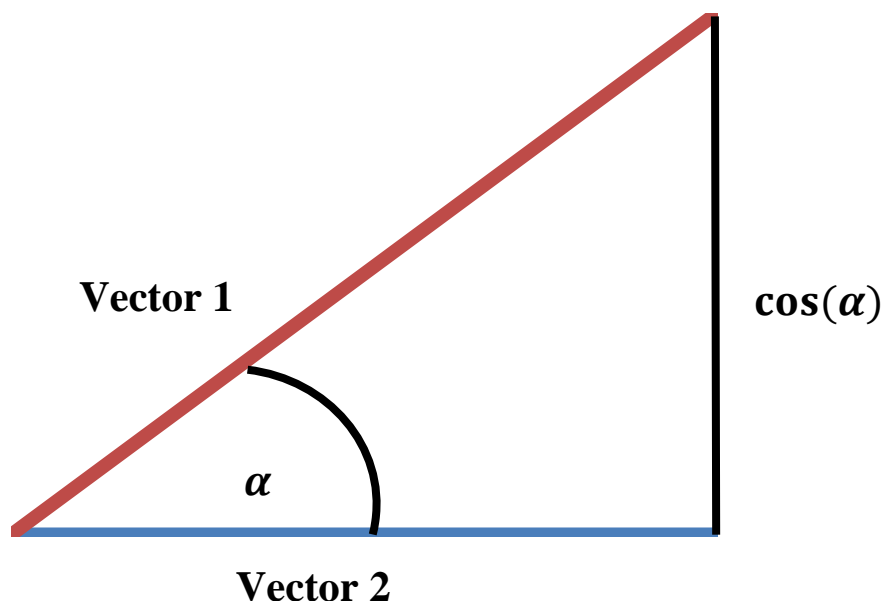


Figura 1 - Representación de la Función Coseno

Esta función se ha utilizado comúnmente en minería de textos para medir la similitud entre dos variables (documentos, consultas,...) representadas como vectores de componentes. Esta medida también ha sido aplicada al área de SR para medir la similitud entre perfiles de usuarios [44], cuando estos son representados como vectores.

3.2. BM25 [117]

BM25 es una función de similitud utilizada para generar un *ranking* de documentos de acuerdo a su relevancia con una consulta de información. BM25 busca términos de la consulta en los documentos a devolver para establecer la puntuación de estos.

De manera formal, dada una consulta Q conteniendo un conjunto de términos (q_1, \dots, q_n) , la puntuación de un documento D establecida por BM25 es:

$$score(D, Q) = \sum_{i=1}^n IDF(q_i) \cdot \frac{f_{q_i,D}(k_1 + 1)}{f_{q_i,D} + k_1(1 - b + b \frac{|D|}{avgdl})} \quad \text{Ecuación 6}$$

donde $IDF(q_i)$ es la frecuencia inversa del término q_i de la consulta (ver Ecuación 15), $f_{q_i,D}$ es la frecuencia del término q_i en el documento D , $|D|$ es la longitud del documento D y $avgdl$ es la longitud media de los documentos que forman la colección. k_1 y b son dos términos para ajustar el funcionamiento de la ecuación, usualmente los valores asignados son $k_1 = 2.0$ y $b = 0.75$.

3.3. Otras medidas de similitud

En la literatura se han empleado otras medidas de similitud. Una de las primeras, propuesta por Resnick et al en [113], fue la correlación de *Pearson*. La correlación de Pearson es una medida de similitud para cuantificar la relación lineal entre dos variables. Su formulación se muestra en la Ecuación 7:

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} \quad \text{Ecuación 7}$$

donde $\rho_{X,Y}$ es la correlación de Pearson entre las variables X e Y , $cov(X,Y)$ es la covarianza de (X,Y) y σ_X es la desviación típica de X .

Otro tipo de medida de similitud, muy similar a la función coseno, es la similitud *Manhattan* [144]. La función de similitud Manhattan entre dos variables X e Y , representadas como vectores, se define como la distancia entre los puntos de sus vectores, más formalmente en la Ecuación 8:

$$d(X,Y) = \sum_{i=1}^N |X_i Y_i| \quad \text{Ecuación 8}$$

Este tipo de medidas (Pearson, Coseno, Manhattan) tienen una desventaja común- A la hora de comparar variables solo se tiene en cuenta aquellos atributos que las variables tengan en común. Esta circunstancia puede llevar a un funcionamiento del sistema no deseado. En [24] Candillier et al ejemplifican este problema, llevándolo al extremo, de la siguiente manera:

En el contexto de un sistema de recomendación de películas, se suponen dos usuarios con preferencias completamente distintas, a uno le gustan las películas de ciencia ficción y al otro las películas de comedia, y no comparten ninguna puntuación en común. Suponer ahora que ambos usuarios puntúan de igual modo una película "*Men in Black*", una comedia de ciencia ficción. Ahora estos usuarios compartirán una única puntuación igual por lo que aplicando las medidas anteriores el sistema los considerará iguales (o al menos muy parecidos), recomendándoles el mismo tipo de contenidos, a pesar de que el resto sus preferencias no coinciden en absoluto.

Para solucionar este problema se ha probado el uso de otro tipo de medidas de similitud como la distancia *Jaccard*. A la hora de comparar dos vectores, la distancia *Jaccard* tiene en cuenta tanto los atributos que son coincidentes en ambos vectores, como los que no lo son. La definición formal de la distancia *Jaccard*, entre dos variables X e Y , se puede ver en la **Ecuación 9**:

$$J_{\delta}(X, Y) = \frac{|X \cup Y| - |X \cap Y|}{|X \cup Y|} \quad \text{Ecuación 9}$$

Sin embargo, la distancia *Jaccard* presenta otra desventaja. Solo tiene en cuenta la coocurrencia de atributos dados dos vectores, sin tener en cuenta el valor de estos atributos. Esto conlleva el problema de que, por ejemplo, en el mismo contexto de sistema de recomendación de películas planteado con anterioridad, dos usuarios que puntuasen las mismas películas serían considerandos iguales aun cuando sus puntuaciones fueran completamente diferentes.

Debido a este problema, la distancia *Jaccard* se utiliza en combinación con alguna de las medidas anteriores (*Pearson*, *Coseno* o *Manhattan*), utilizándose la distancia *Jaccard* como función de pesado. Los resultados obtenidos por Candillier et al en [24] muestran que el uso de la distancia *Jaccard* como función de pesado proporciona resultados mucho más satisfactorios, a la vez que reduce el tamaño óptimo del “*neighbourhood*” tanto para los contenidos como para los usuarios en sistemas de filtro colaborativo.

El uso de fuentes externas de conocimiento para esta tarea también se ha planteado. Se espera que al añadir conocimiento al ya disponible, el funcionamiento de las medidas de similitud sea más preciso. Una de las fuentes de conocimiento planteadas son las **ontologías del dominio** construidas por expertos. La ontología tiene categorizados a los objetos en una jerarquía, en la cual se definen las relaciones entre los conceptos. Mediante las ontologías es posible medir la similitud entre dos objetos en función de su categoría. Si bien esta aplicación es muy útil en dominios reducidos, cuando el tamaño de los datos es muy grande este tipo de medida no es práctico.

Otra importante fuente de conocimiento es Internet. La ventaja de Internet es que se dispone de una ingente cantidad de información (prácticamente ilimitada) independiente del dominio, a diferencia de las ontologías. Para aprovechar las posibilidades de Internet a la hora de medir la similitud entre objetos, se ha definido una medida de similitud conocida como *Normalized Google Distance* [32] o *NGD*. *NGD* calcula la similitud entre términos textuales usando sus coocurrencias en los sitios web (según Google). *NGD* ha sido utilizada por Jack y Duclayee en [64] con resultados muy satisfactorios. El principal problema de esta medida es la dificultad de su uso cuando se pretende aplicar, ya que necesita grandes cantidades de datos, y debido a las restricciones de uso del *API* de Google no es posible adquirirlos de este.

3.4. Comentarios Finales

Si bien es cierto que las métricas de similitud no constituyen uno de los principales componentes de un sistema de recomendación, la importancia de éstas en el rendimiento final del sistema es capital. Esto es así dado que, independientemente de la aproximación de recomendación utilizada, el funcionamiento final de un SR se basa en la similitud entre un modelo de usuario y un contenido: un contenido será recomendado sólo si es similar sobre la base del modelo que almacena las preferencias del usuario. Por esto, el funcionamiento de la métrica de similitud utilizada determinará el funcionamiento final del SR.

Pese a que a lo largo de los trabajos en esta área se han presentado una gran variedad de medidas, como por ejemplo: Distancia *Jaccard*, Distancia *Manhattan*, Correlación de *Pearson* o, la más sofisticada, *Normalized Google Distance*; la utilización de la Función Coseno es todavía común. Esto se debe a que el rendimiento obtenido con esta medida es similar al obtenido con otras métricas más complejas.

Capítulo 4. Técnicas de Representación de Información Textual

Las aproximaciones de modelado se emplean para extraer información desde el perfil que define las preferencias de cada usuario y obtener una representación o modelo de dicho perfil. Sin embargo, en aquellos tipos de modelado que utilizan información textual es necesario representar términos (o características) para establecer la importancia relativa de cada uno con respecto al resto. Para representar estos modelos se plantean, principalmente dos tipos de técnicas: técnicas probabilísticas y técnicas no probabilísticas.

4.1. Técnicas no probabilísticas

La aproximación clásica para representar los modelos consiste en utilizar modelos que no utilizan técnicas basadas en probabilidades, como por ejemplo VSM (*Vector Space Model*) o LSA (*Latent Semantic Analysis*).

4.1.1. VSM (Vector Space Model)

VSM o Modelo de espacio vectorial es un modelo de representación de documentos textuales mediante vectores [121]. En VSM se define un documento como un vector:

$$d_j = (w_{1,j}, w_{2,j}, \dots, w_{t,j}) \quad \text{Ecuación 10}$$

donde cada dimensión del vector ($w_{1,j}$) se corresponde con cada uno de los términos existentes en la colección; si un término ocurría en el documento su valor en el vector es diferente de cero. Existen diferentes maneras de establecer el peso de cada término (ver Sección Función de Pesado).

El modelo VSM ha sido extensamente utilizado en IR para comparar una consulta dada con documentos textuales, representando también la consulta como un vector de términos, así como para comparar documentos entre sí, [91]. Para ello se calcula la diferencia de ángulos de los vectores que representan a los documentos, basándose en el coseno de estos ángulos. De esta manera un valor de coseno igual a 0 significa que ambos vectores, y por tanto ambos documentos, son ortogonales y por tanto no hay coincidencia alguna.

Pese a que esta aplicación ha sido ampliamente utilizada, presenta una serie de limitaciones:

- Los documentos muy largos son difícilmente comparables entre sí ya que tienen muy pocos valores comunes.
- Sensibles al vocabulario. Dos documentos con una temática común pero cuyo vocabulario difiera no serán elegidos como similares cuando realmente lo son.
- Problemática asociada a los términos. Hay dos aspectos referentes a este problema, la **polisemia** y la **sinonimia**.
 - Por sinonimia se define el uso de diferentes términos para referirse a un mismo concepto, haciendo que dos sinónimos sean considerados diferentes. La existencia de sinonimia afecta al *recall* de un sistema, ya que, ante una búsqueda, se recuperarán menos objetos relevantes de los deseados.
 - Por polisemia se entiende el hecho de que un mismo término pueda referirse a más de un concepto diferente, haciendo que conceptos diferentes sean considerados iguales por estar representados por el mismo término polisémico. La existencia de polisemia afecta a la precisión de un sistema, ya que se recuperarán objetos que no son relevantes, dada una consulta.
- No se tiene en cuenta el lugar en el cual los términos aparecen en el documento. Sin embargo, no es un aspecto tan problemático como pueda parecer en un principio. Launder afirma en [79] que el 80% de la información potencial del lenguaje se encuentra en la elección de las palabras y no en el orden de estas.

4.1.1.1. Función de Pesado

Las funciones de pesado, o ponderación, se utilizan para medir la importancia de un término dentro de un texto de acuerdo a un determinado criterio, normalmente relacionado con la frecuencia de aparición del término. Esto es así dado que, es lógico pensar que si un término aparece frecuentemente en un documento será más representativo de este documento que otro que ocurra con menor frecuencia.

En la literatura se distingue entre dos tipos de funciones de pesado, funciones de pesado locales y funciones de pesado globales.

Funciones de pesado locales

Son aquellas que para ponderar la importancia de un término sólo tienen en cuenta la información del documento donde ocurre el término. Formalmente definida, una función de pesado local es aquella función

$$f(t_i, d_j) \text{ siendo } (t_i | t_i \in d_j) \quad \text{Ecuación 11}$$

donde t_i es un término i que ocurre dentro de un documento j .

Una de las funciones de pesado locales más utilizada es la **Función Binaria**. Esta función de pesado considera únicamente la ocurrencia o no de un término dentro de un documento $\{0,1\}$ para ponderar la relevancia de un término. Se expresa de manera formal en la Ecuación 12:

$$F: \text{Bin}(t_i d_j) = \begin{cases} 1, & \text{si } t_i \text{ ocurre en } d_j \\ 0, & \text{si no aparece} \end{cases} \quad \text{Ecuación 12}$$

Funciones de pesado globales

Se define función de pesado global como aquella función que, a diferencia de las locales, tiene en cuenta información de todos los documentos de la colección para determinar la importancia de un determinado término. Formalmente definida, una función de pesado global es aquella función:

$$f(t_i, d_j, C) \text{ siendo } (t_i | t_i \in d_j \text{ y } d_j \in C) \quad \text{Ecuación 13}$$

donde t_i es un término que ocurre dentro de un documento d_j , el cual forma parte de una colección de documentos C .

La función de pesado global más conocida es TF-IDF (*Term Frequency – Inverse Document Frequency*). TF-IDF trata de reflejar la importancia de un término dentro de un documento teniendo en cuenta la frecuencia de aparición del término en el documento y la frecuencia de aparición en el resto de documentos de la colección. La intuición detrás de TF-IDF es que un término será más representativo de un documento cuantas más veces aparezca en el documento y menos veces aparezca en el resto de documentos de la colección. La fórmula que define TF-IDF es:

$$TFIDF_{i,j} = f_{i,j} * idf_i \quad \text{Ecuación 14}$$

donde $f_{i,j}$ es la frecuencia del término i en el documento j y idf_i es la **frecuencia inversa** del término i definida por la fórmula:

$$idf(i) = \log\left(\frac{N}{n_i}\right) \quad \text{Ecuación 15}$$

donde N es el total de documentos de la colección y n_i es el número de documentos en los que un término i ocurre.

4.1.2. LSA (Latent Semantic Analysis)

LSA puede resolver algunos de los problemas inherentes de VSM y de los sistemas de representación clásicos, los cuáles basan su funcionamiento en la coincidencia de términos. Para ello **LSA** pretende identificar patrones que relacionen términos que aparecen en un corpus de textos mediante la extracción del contenido conceptual de un texto estableciendo asociaciones de sus términos con otros términos que ocurren en contextos textuales similares.

LSA fue planteado por primera vez por Deerwester et al [39]. La solución que proponían se basa en estimar la estructura latente, analizando estadísticamente las co-ocurrencias de palabras iguales a lo largo de diferentes documentos. La idea es que si dos documentos son similares se producirán co-ocurrencias de términos entre los documentos.

Cuando se realiza una consulta textual sobre un sistema basado en LSI, éste devolverá documentos que contengan los términos de la consulta (como un sistema IR clásico) y también documentos cuya similitud con los documentos devueltos sea alta, pudiendo ser que no compartan ningún término en común con la consulta.

Un **ejemplo** de funcionamiento de LSA se muestra a continuación: supongamos dos ecuaciones:

$$\mathbf{a} + \mathbf{b} + \mathbf{c} = \mathbf{e}$$

$$\mathbf{a} + \mathbf{b} + \mathbf{d} = \mathbf{e}$$

a y **b** co-ocurren en el mismo contexto mientras que **c** y **d** no lo hacen; sin embargo, se puede inferir que **c** y **d** son equivalentes dado el contexto. Si a este conjunto añadimos dos ecuaciones más:

$$\mathbf{a} + \mathbf{b} + \mathbf{f} = \mathbf{h}$$

$$\mathbf{a} + \mathbf{b} + \mathbf{g} = \mathbf{h}$$

De igual manera que antes se puede inferir que **f** y **g** son equivalentes en el contexto dado. Por último se consideran las dos ecuaciones siguientes:

$$\mathbf{c} + \mathbf{f} = \mathbf{j}$$

$$\mathbf{d} + \mathbf{g} = \mathbf{k}$$

Sabiendo por las ecuaciones anteriores que **c** y **d** por un lado y **f** y **g** por otro son equivalentes, se puede determinar que en estos dos pasajes son equivalentes (y por extensión **j** y **k** también lo son) a pesar de que no exista ningún término en común entre ellos.

4.1.2.1. Base teórica de LSA

LSA trata de representar los términos (ya sean palabras, n-gramas, párrafos, etc.) y documentos que conforman un corpus como puntos en un **espacio semántico latente** multidimensional. Para ello se utiliza una técnica conocida como: **descomposición en valores singulares** (*Singular value decomposition*) o SVD, que se basa en la posibilidad matemática de descomponer una matriz rectangular en el producto de otras tres matrices de acuerdo a la siguiente ecuación:

$$M = U\Sigma V^* \quad \text{Ecuación 16}$$

siendo M una matriz $m \times n$ conocida como matriz término-documento, la cuál representa al corpus, dónde cada fila representa un término y cada columna un documento (ver Ecuación 17). En M cada elemento t_{ij} contiene la importancia del término i (de acuerdo a alguna función de pesado) en el documento j .

$$M = \begin{pmatrix} t_{11} & t_{12} & \dots & t_{1n} \\ t_{21} & t_{22} & \dots & t_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ t_{m1} & t_{m2} & \dots & t_{mn} \end{pmatrix} \quad \text{Ecuación 17}$$

De las otras matrices presentes en la Ecuación 16, U es una matriz $m \times r$ cuyas columnas son ortonormales, V^* es una matriz $r \times n$ con columnas también ortonormales y Σ es una matriz diagonal $r \times r$ con los valores singulares de M (aquellos distintos de 0).

LSA propone considerar únicamente los k valores singulares mayores de Σ , generada mediante SVD, fijando el resto de valores a 0. Aplicando esta reducción se obtiene una matriz M_k definida en la Ecuación 18 como:

$$M_k = U_k \Sigma_k V_k^* \quad \text{Ecuación 18}$$

donde M_k es una matriz de rango k que puede ser interpretada como una de la matriz término documento M en un nuevo espacio k -dimensional llamado **espacio semántico latente** de M . Los documentos, representados por las filas de $V_k^* \Sigma_k$, son proyectados en este espacio semántico latente por los vectores columna de U_k .

De la selección del parámetro k dependerá la capacidad de LSA para extraer la semántica latente de los documentos. El parámetro k se puede determinar de manera empírica y también se han propuesto métodos formales para estimar este parámetro. Uno de estos métodos es el teorema planteado por Eckart y Young:

Teorema Eckart-Young: *Dada una matriz M de $m \times n$ y su descomposición en valores singulares*

$$U \Sigma V^*$$

existe una matriz M_k $m \times n$ de rango k que minimiza el sumatorio de los errores cuadrados entre los elementos de M y M_k cuando

$$M_k = U_k \Sigma_k V_k^*,$$

siendo Σ_k la matriz diagonal que contiene los k valores mayores de Σ .

Definido formalmente:

$$\exists M_k / \min \left(\|M - M_k\|^2 = \sum_{i,j} (M_{i,j} - M_{k,i,j})^2 \right)$$

Aplicando esta reducción (definida por k) se consigue disminuir la dimensionalidad de tal manera que aquellos documentos que comparten términos co-ocurrentes tenderán a tener una representación similar en el espacio semántico latente de menor dimensión, incluso si no tienen términos en común entre ellos.

4.2. Técnicas probabilísticas

Estas técnicas hacen uso de modelos probabilísticos para representar el contenido de los perfiles de usuario. Las técnicas más importantes de este tipo son PLSA (*Probabilistic Latent Semantic Analysis*) y LDA (*Latent Dirichlet Allocation*). Tratan de solucionar los problemas derivados de la falta de una base estadística sólida [61].

4.2.1. pLSA (Probabilistic Latent Semantic Analysis)

Se trata de una expansión de LSA utilizando técnicas probabilísticas. Basa su funcionamiento en un modelo estadístico llamado *aspect model* [61]. *Aspect model* es un modelo para la co-ocurrencia de datos que asocia una variable no observada:

$z \in \mathcal{Z} = \{z_1, \dots, z_{\mathcal{K}}\}$ (conjunto de los temas o documentos del corpus)

a cada observación; siendo una observación la ocurrencia de un término:

$w \in \mathcal{W} = \{w_1, \dots, w_{\mathcal{N}}\}$ (conjunto de términos del corpus)

en un documento:

$d \in \mathcal{D} = \{d_1, \dots, d_{\mathcal{M}}\}$ (conjunto de documentos del corpus).

El funcionamiento de pLSA tiene las siguientes fases:

- Seleccionar un documento d con probabilidad $P(d)$
- Escoger una clase latente z con probabilidad $P(z|d)$
- Generar un término w con probabilidad $P(w|z)$

En base a este modelo se puede calcular la probabilidad de que se dé una co-ocurrencia de un término en un documento de acuerdo a la Ecuación 19:

$$P(d|w) = P(d)P(w|d), \quad \text{Ecuación 19}$$

siendo $P(w|d)$:

$$P(w|d) = \sum_{z \in \mathcal{Z}} P(w|z)P(z|d) \quad \text{Ecuación 20}$$

4.2.2. LDA (Latent Dirichlet Allocation)

LDA es un modelo generativo que permite describir un conjunto de interacciones mediante una serie de clases no observadas, relacionando las observaciones similares que se hayan producido. Trasladado al caso de estudio, las observaciones serían palabras y las clases no observadas los *topics* (temáticas) a los cuales pertenecen.

LDA representa cada documento como una mezcla de *topics* como en PLSI; la diferencia reside en que en LDA las distribuciones de los *topics* siguen una distribución de Dirichlet (una generalización de una distribución Beta para el caso multinomial) definidas por: α , que es el parámetro de la distribución de *topics* por documento (θ), y β , que es el parámetro de la distribución de palabras por *topic* (φ).

La estimación de los parámetros α y β es un problema irresoluble, por ello es necesario usar métodos aproximativos como la estimación basándose en muestreos de Gibbs, descrito por Griffiths y Steyvers en [49].

Con un modelo generativo LDA se puede caracterizar a un documento como una mezcla aleatoria de *topics* latentes, en el cuál, cada *topic* es caracterizado a su vez por una distribución sobre las palabras. Más concretamente, para cada documento d :

- Dada una distribución θ , elegir un $\theta_d \sim Dir(\alpha)$,

donde $d \in \mathcal{D} = \{d_1, \dots, d_M\}$ representa un documento del total de la colección y $Dir(\alpha)$ es una distribución Dirichlet para el parámetro α .

- Dada una distribución φ , elegir un $\varphi_k \sim Dir(\beta)$,

donde $k \in K = \{k_1, \dots, k_K\}$ representa un topic del conjunto de *topics* de la colección y $Dir(\beta)$ es una distribución Dirichlet para el parámetro β

- Para cada una de las palabras que vayan a formar del documento d :
 - o Elegir un *topic* $z \sim \theta_d$
 - o Elegir una palabra $z \sim \varphi_z$

4.3. Comentarios finales

La representación textual no está limitada a la Recomendación de Contenidos sino que es un área multidisciplinar. Por ello, la investigación en esta área es muy extensa, dando lugar a una gran cantidad de trabajos al respecto. Una manera común de clasificar las representaciones textuales es en aquellas que usan técnicas probabilísticas y aquellas que no lo hacen.

En la Tabla 2 se detalla una comparativa entre las técnicas de recomendación presentadas de acuerdo a diferentes criterios. Observando la tabla se puede contemplar que ninguna de estas aproximaciones tiene en cuenta el orden de las palabras; es decir, para estas representaciones, tendrá la misma representación textos con los mismos términos en diferente orden. Como ya se ha planteado anteriormente, esta circunstancia no afecta en gran medida a la calidad de la representación (ver [79]).

MODELO	Sensible al vocabulario	Tiene en cuenta orden de las palabras	Complejidad asociada al tamaño
VSM	SI	NO	NO
LSI	NO	NO	NO
pLSI	NO	NO	SI
LDA	NO	NO	NO

Tabla 2 - Comparativa entre los diferentes modelos de representación

Si atendemos a la complejidad asociada al tamaño de la colección a representar se observa que solo para pLSI representa un inconveniente. Esto es, dado un incremento del tamaño de la colección la complejidad asociada a la ejecución de pLSI crecerá en gran medida con respecto a las otras representaciones. Así mismo el uso de VSM para grandes colecciones de datos puede acarrear una complejidad alta si no se aplica algún tipo de técnica para reducir la dimensión de los datos.

Por último destacar que solo VSM es sensible al vocabulario, dado que el resto de representaciones trata de buscar los *topics* latentes con independencia de con qué términos estén representados. Este inconveniente sería suficiente para descartar el uso de VSM frente al resto de representaciones; sin embargo, dada su sencillez de ejecución, VSM sigue siendo una técnica cuyo uso es recomendable (de hecho sigue siendo una técnica común, tanto para Recuperación de Información como para Sistemas de Recomendación).

De manera general, se puede concluir que las técnicas probabilísticas (PLSI o LDA) ofrecen un rendimiento mayor (en términos generales) a la utilización de otro tipo de técnicas, como VSM. Debido a esto, en los últimos años la utilización de técnicas como LDA se ha convertido en un estándar de facto en diversas áreas de investigación como la Recuperación de Información o la Clasificación de Textos. Sin embargo, la ejecución de las mismas es muy costosa en términos computacionales, haciendo que su utilización en sistemas reales sea poco práctica. Es por esto que todavía a día de hoy técnicas como VSM son extensamente utilizadas, con notable éxito, a la hora de representar contenidos textuales.

Capítulo 5. Tecnología: Herramientas y Recursos

5.1. Herramientas

5.1.1. Solr⁹

Solr es una plataforma de búsqueda de código abierto basado en el proyecto *Apache Lucene*¹⁰ para la indexación y recuperación de información textual. *Solr* está desarrollado en Java y funciona como un servidor de búsqueda dentro de un contenedor de servlets como *Apache Tomcat*¹¹.

Las principales características que proporciona *Solr* son:

- Potente motor de búsqueda textual con múltiples opciones
- Permite la búsqueda utilizando facetas
- Permite llevar a cabo clustering dinámico de los resultados
- Resaltado de coincidencias encontradas
- Integración con sistemas de bases de datos

El funcionamiento de *Solr* se basa en la librería de búsqueda de *Apache Lucene*, e incluye todas sus funcionalidades, tanto de búsqueda como de indexado. De cara a generar el ranking de resultados ante una consulta de información *Solr* utiliza la implementación propia de *Apache Lucene* de la función coseno [53], que no es exactamente la implementación teórica de esta técnica.

La principal ventaja de *Solr* frente a *Apache Lucene* es que facilita el trabajo de indexación y búsqueda, sin tener que desarrollar ningún programa que haga uso de *Lucene*. *Solr* también es más sencillo de configurar (a través de archivos XML) que *Lucene*. La configuración de *Solr* ofrece la posibilidad de ajustar parámetros del proceso de indexado, como la estructura de datos a utilizar o los analizadores que se aplicarán sobre el contenido de los documentos, así como de la recuperación de información.

⁹ <http://lucene.apache.org/solr/>

¹⁰ <http://lucene.apache.org/core/>

¹¹ <http://tomcat.apache.org/>

5.1.2. Cassandra¹²

Cassandra es un software de código libre para la gestión de bases de datos distribuidas, respaldado por *Apache Software Foundation*¹³. Está diseñado para poder manejar un gran número de datos, distribuidos a lo largo de muchos servidores, con facilidad. Grandes corporaciones con una gran cantidad de datos a su disposición como *Facebook*¹⁴ y *Twitter*¹⁵ utilizan *Cassandra* para gestionar sus bases de datos.

5.1.2.1. Características

Las principales características de *Cassandra* que le hacen destacar sobre otros sistemas de Bases de datos son:

- **Descentralizada:** Es la característica más destacable de *Cassandra*, no existe un único nodo central, en lugar de ello, todos los nodos tienen el mismo peso. Esto evita que haya un punto de fallo en la estructura de la base de datos. Ante la caída de un nodo, cualquier otro nodo puede seguir atendiendo a las peticiones, evitando la caída del sistema.
- **Alta velocidad:** Debido al cambio de modelo de datos (detallado en la Sección 5.1.2.2), *Apache Cassandra* representa un significativo aumento de velocidad con respecto a un gestor de base de datos tradicional. En la Tabla 3 se muestra una comparación entre los tiempos de lectura y escritura de *Apache Cassandra*, frente a *MySQL* para una base de datos de 50 Gb de tamaño:

	<i>MySQL</i>	<i>Cassandra</i>
Lectura	350 milisegundos	15 milisegundos
Escritura	300 milisegundos	0.12 milisegundos

Tabla 3 - Comparativa *MySQL* vs. *Cassandra*

Como se puede ver en la tabla, la velocidad tanto de escritura como de lectura es mucho mayor en *Cassandra* que en *MySQL*. La comparación se ha realizado sobre *MySQL* por destacar su velocidad sobre la de otros gestores de bases de datos. En la Figura 2, Figura 3 y Figura 4 se puede ver una comparativa de *MySQL* frente a otros gestores de bases de datos ante consultas de diferente complejidad.

¹²<http://cassandra.apache.org/>

¹³ <http://www.apache.org/>

¹⁴ <http://www.Facebook.com/>

¹⁵ <https://Twitter.com/>

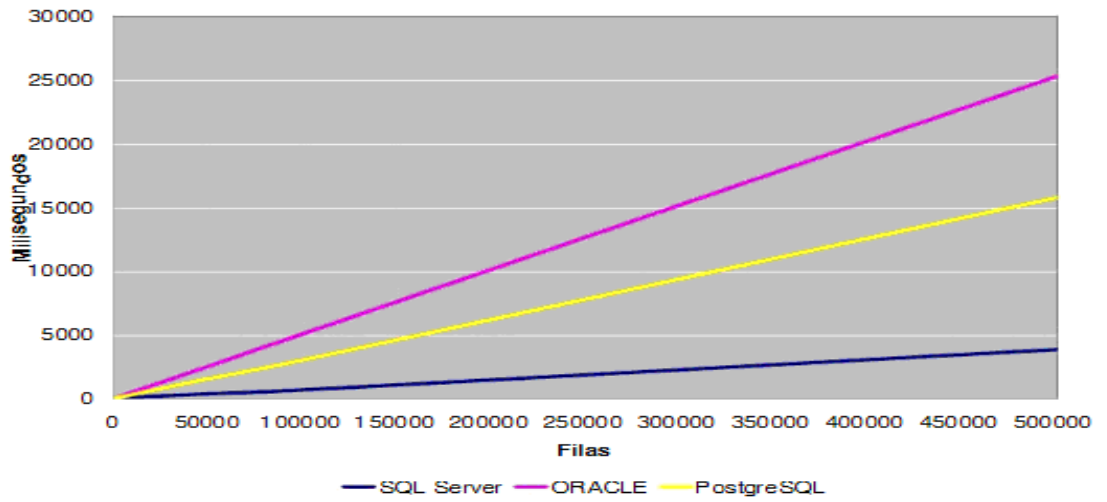


Figura 2 - Comparativa - Consulta de complejidad baja

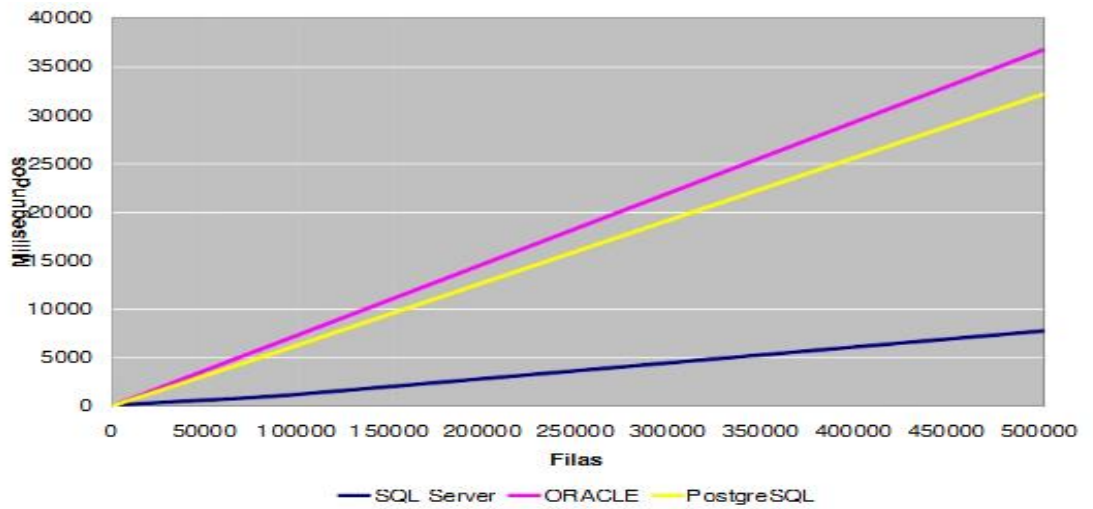


Figura 3 - Comparativa - Consulta de Complejidad Media

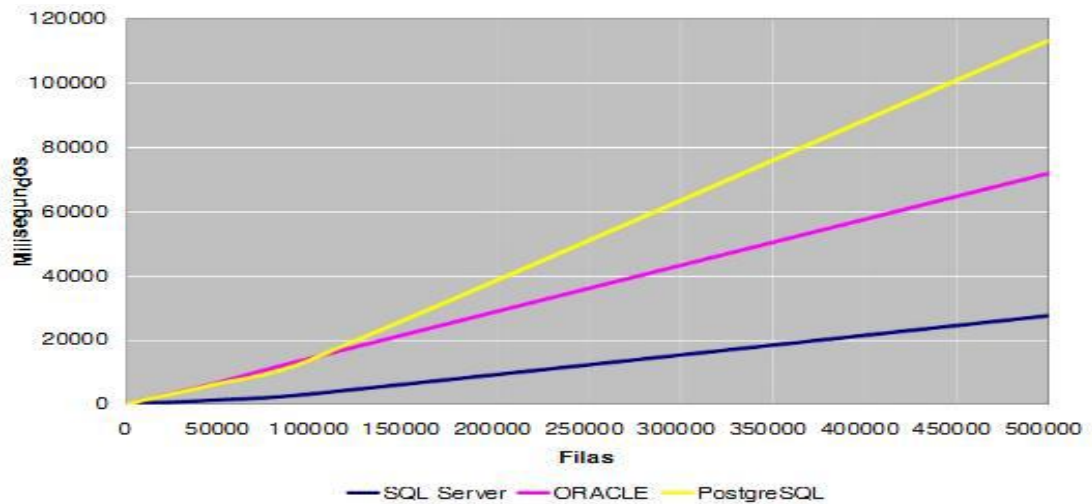


Figura 4 - Complejidad - Consulta de Complejidad Alta

- **Tolerancia ante fallos:** De manera automática los datos son replicados a lo largo de diferentes nodos de la estructura. Si un nodo falla puede ser remplazado inmediatamente, de manera transparente para el funcionamiento del sistema
- **Elástica:** Se pueden agregar nuevas máquinas, incluso mientras el sistema está funcionando. Ello no supone un aumento significativo en los tiempos de lectura y escritura ni modifica la estructura de la base de datos.
- **Altamente configurable:** El funcionamiento y la estructura es fácilmente modificable, otorgando una gran flexibilidad al sistema.
- **Probada:** Su funcionamiento está respaldado por la experiencia y la adquirida en su uso por parte de grandes sistemas como *Digg*¹⁶, *Facebook*¹⁴, *Twitter*¹⁵ o *Reddit*¹⁷. Su uso se está extendiendo también en el ámbito académico. Todo ello hace que exista una importante cantidad de documentación y *feedback* proveniente de sus usuarios.
- **Fácilmente integrable:** La integración y gestión de *Apache Cassandra* con otros lenguajes y sistemas está ampliamente cubierto. Existen clientes de *Apache Cassandra* para *Python*, *Java*, *Scala*, *.NET*, *Ruby* o *PHP* entre otros.

5.1.2.2. Modelo de datos

La principal novedad de *Cassandra* respecto a otros sistemas tradicionales de gestión de bases de datos es que rompe con el modelo de base de datos relacional. En su lugar utiliza un modelo de datos basado en claves (*Keys*) y valores (*Values*), pudiendo cada clave estar asociada a uno o varios valores.

¹⁶ <http://digg.com/>

¹⁷ <http://www.reddit.com/>

Comenzando desde abajo hacia arriba la estructura de los datos en *Cassandra* se organiza en:

- **Columns:** Es la estructura de datos más simple. Está compuesta por una terna de nombre, valor y *timestamp* (usado este último para la resolución de conflictos)
- **SuperColumns:** Son columnas cuyos valores a su vez son otras columnas.
- **ColumnFamilies y SuperColumnFamilies:** Es un contenedor de columnas (o supercolumnas) que agrupa a aquellas que están relacionadas. Es un concepto similar al de tabla en el modelo de bases de datos relacionales.
- **KeySpaces:** Es un contenedor de *ColumnFamilies* o *SuperColumnFamilies*. Es la estructura principal en el modelo de datos de *Cassandra* y será único para cada aplicación.

En la Figura 5 y en la Figura 6 se puede observar un ejemplo de un modelo de datos de columnas y súper columnas respectivamente.

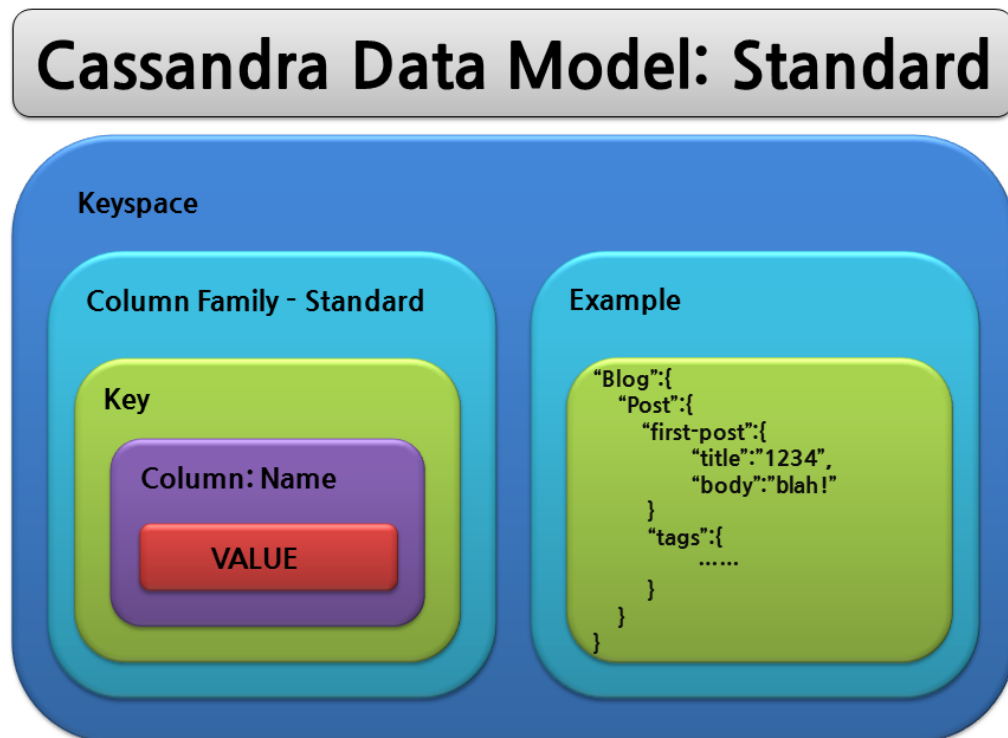


Figura 5 - Modelo de Datos con *Columns* [28]

Cassandra Data Model: Super

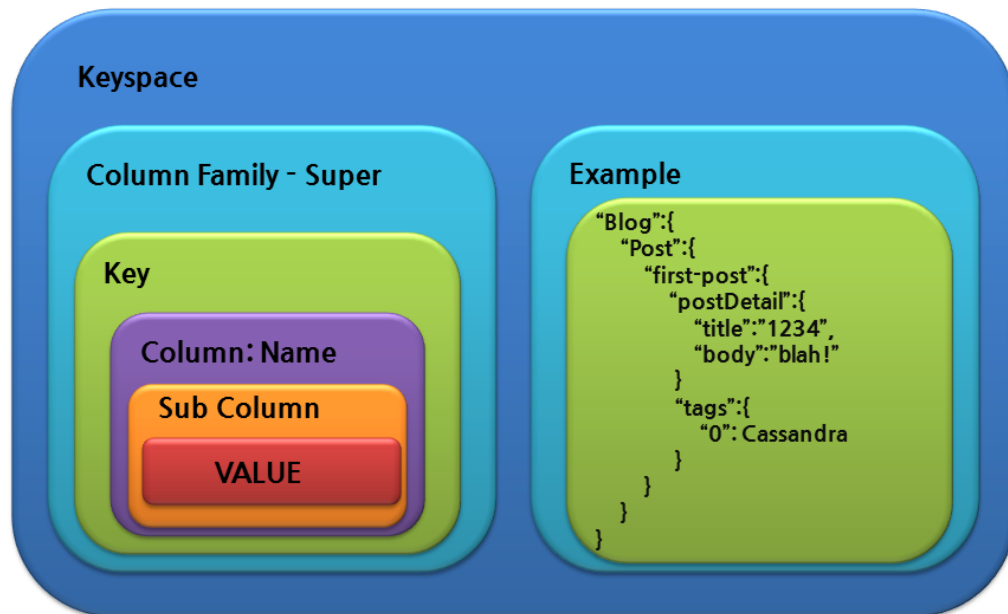


Figura 6 - Modelo de Datos con *Supercolumns* [28]

5.1.3. GWT¹⁸

GWT o *Google Web Toolkit* es un kit de desarrollo para la construcción y optimización de aplicaciones basadas en navegador. La idea de la utilización de GWT es permitir el desarrollo de aplicaciones web complejas sin necesidad de tener que programar en lenguajes tales como JavaScript o AJAX.

Para esto, GWT proporciona un SDK (*Software Development Kit*) que provee de un conjunto de APIs Java y Widget, permitiendo al desarrollador escribir aplicaciones AJAX en lenguaje Java. GWT genera el código JavaScript correspondiente a la aplicación, abstrayendo de este proceso al desarrollador, el cual únicamente programará utilizando lenguaje JAVA.

Además de la capacidad de abstracción, la cual facilita en gran medida la labor del desarrollador incrementando así su productividad, la creación de aplicaciones utilizando GWT da lugar a un código mucho más optimizado.

¹⁸ <https://developers.google.com/web-toolkit/?hl=es>

De cara a facilitar el desarrollo de aplicaciones, GWT también ofrece un plugin para Eclipse, uno de los IDEs más utilizados de programación en Java. Gracias a este plugin se facilita, entre otras cosas, la creación de interfaces web, la importación y actualización de las APIs de GWT o el despliegue de la aplicación.

5.2. Recursos y Colecciones

En este apartado se presenta una recopilación de recursos y colecciones que hacen posible la evaluación de técnicas y sistemas. Para cada uno de ellos se recogen sus características, así como sus problemas tanto generales como aquellos relacionados con la propuesta que se presenta en este trabajo.

5.2.1. Colecciones enfocadas en Recomendación

Las colecciones presentadas en esta sección han sido recopiladas específicamente para su experimentación en el área de SR. Todas ellas proceden de las bases de datos de plataformas sociales interesadas en recomendar contenidos a sus usuarios.

A continuación se detallan algunas de las principales colecciones utilizadas en la investigación en el área de SR que son accesibles online.

5.2.1.1. Colección Yahoo Webscope

*Yahoo Webscope*¹⁹ es un programa enfocado a recopilar y distribuir colecciones de datos útiles para la investigación científica en diferentes áreas, entre ellas los SR. *Yahoo Webscope* ofrece varias colecciones en el área de los SR; de entre ellas la más interesante es *Yahoo! Front Page Today Module User Click Log Dataset*. Esta colección contiene una fracción del *log* de *clicks* de usuario en noticias a través de la herramienta *Today Module* de *Yahoo*.

La colección está compuesta por 45 millones de visitas de usuarios, incluyendo información de los usuarios así como de las noticias visitadas. *Yahoo* también ofrece una versión reducida de esta colección formada por 28 millones de visitas y la información asociada a ellas. La colección posee una entrada por cada visita de usuario. Las entradas tienen el siguiente formato:

- **timestamp:** e.g., 1241160900
- **id_artículo_mostrado:** e.g., 109513
- **click_usuario:** 0 si no hay click y 1 si lo hay
- **“|usuario” y “{|id_artículo}”:** indica el comienzo de las características del usuario y el artículo

¹⁹ <http://webscope.sandbox.yahoo.com/>

- **características:** Las características están codificadas como “id_característica: valor_característica”.

Aunque la colección presenta una importante fuente de datos de usuarios reales, su utilización presenta varios problemas. En primer lugar, por cuestiones de anonimización, la información de los usuarios ha sido eliminada. Cada uno de los usuarios está únicamente identificado con la cadena “*user*”. Otro problema que presenta la colección es la falta del contenido textual de los artículos. De cada uno de los artículos, en lugar del texto, se tiene un conjunto de características, construidas de acuerdo al trabajo descrito en [29].

5.2.1.2. Colección VideoLectures²⁰

VideoLectures es una plataforma de acceso libre y gratuito para visionar conferencias científicas. Utilizando información de VideoLectures se ha generado para SR, dentro del ECML/PKDD Discovery Challenge de 2011 [7], una colección accesible a través de la siguiente dirección:

<http://lis.irb.hr/challenge/index.php/dataset/>

La colección está compuesta por 8105 videos, de los cuales 5286 han sido categorizados manualmente por expertos mediante una taxonomía que contiene 350 temáticas aproximadamente. La colección está dividida en dos partes: un conjunto de entrenamiento formado por 6983 videos anteriores al 1 de Julio de 2009 y un conjunto de test formado por 1122 videos posteriores al 1 de Julio de 2009. La información contenida en la colección es la siguiente:

- **Datos sobre las conferencias:** Cada conferencia tiene asociada información acerca de:
 - Título
 - Tipo (conferencia, keynote, tutorial, conferencia de prensa, etc.)
 - Identificador del idioma (en, es, fr, etc.)
 - Numero de visionados
 - Fecha de publicación
 - Identificador del evento
 - Conjunto de autores

Alguno de los vídeos también incluye una breve descripción, así como los títulos de las diapositivas de la presentación expuesta en la conferencia. El conjunto de datos de entrenamiento posee información adicional asociada a los videos tal como: frecuencias de visionados conjuntos de una pareja de videos (es decir

²⁰ <http://videolectures.net/>

videos que han sido vistos juntos en una misma sesión) o secuencias combinadas de visionados.

- **Datos sobre los autores:** El conjunto de datos tiene información acerca de 8092 autores:
 - Nombre
 - Dirección de correo electrónico
 - Dirección de la página de inicio
 - Género
 - Afiliación
 - Lista de videos correspondiente
- **Datos sobre los eventos:** Las conferencias pueden estar asociadas a un determinado evento (una conferencia específica). Estos eventos tienen asociada información similar a la de una conferencia:
 - Título
 - Tipo
 - Identificador del idioma
 - Fecha de publicación
 - Identificador de meta-evento.

La base de datos contiene información acerca de 519 eventos y meta-eventos, de los cuales 245 han sido categorizados de manera manual y 437 han sido enriquecidos con descripciones textuales procedentes de Wikipedia.

- **Datos sobre las categorías:** Las representación de las categorías se basa en una taxonomía científica utilizada en la propia página de VideoLectures. Cada categoría está definida por:
 - Identificador
 - Nombre
 - Identificador de la categoría padre en la taxonomía
 - *URL* de Wikipedia (en el caso de que exista)
- **Estadísticas sobre los visionados:** El sistema de VideoLectures almacena la secuencia de videos que cada usuario visiona en cada visita a la página. El conjunto de datos contiene datos anonimizados de 329.481 sesiones diferentes. Las estadísticas que se proporcionan de estas sesiones son:
 - Frecuencia de veces que un par de videos son visionados conjuntamente (no necesariamente de forma consecutiva)
 - Frecuencia para una secuencia de visionados combinada/tripletas de videos vistos juntos antes de una secuencia de 10 videos. Esta última estructura está diseñada especialmente para su uso en la competición

VideoLectures	
Autores	8.092
Vídeos	8.105
Vídeos Categorizados	5.286
Temáticas	350
Eventos	519
Visionados	329.841

Tabla 4 - Estadísticas de la Colección VideoLectures

La colección fue generada para su utilización en el VideoLectures.Net Recommender System Challenge. Este Challenge, organizado dentro de la ECML/PKDD, persigue mejorar los SR actualmente utilizados en la web y, más concretamente, los aspectos relacionados con el problema de *cold start* y con la utilización de los *streams* de clicks llevados a cabo por los usuarios. La temática del Challenge es aplicable tanto a sistemas basados en contenido como a aquellos basados en filtro colaborativo. De hecho, los trabajos más interesantes presentados al Challenge utilizan ambas aproximaciones para sus sistemas [8].

De cara a nuestra propuesta, a pesar de que la colección tiene información textual ésta generalmente no hace referencia al contenido de las conferencias sino a metadatos relacionados. Por ello, la información textual no es adecuada para modelar el comportamiento de los usuarios con la aproximación planteada en este trabajo. Por otro lado está el problema de la ausencia de correlación entre el interés del usuario por un vídeo y su descripción textual, como ocurre con las colecciones de películas (i.e. dos vídeos con la misma descripción textual pueden ser de interés muy distinto para un usuario en función de otros factores: autor, fecha, evento asociado,...).

5.2.1.3. Colección Delicious²¹

Delicious es un gestor social de marcadores que permite a los usuarios agregar marcadores correspondientes a páginas web, estableciendo su categoría. Este etiquetado de categorías llevado a cabo de manera social da lugar a una estructura de categorías conocido como folcsonomía. El sistema permite ver, para cada etiqueta de la folcsonomía, qué *URLs* han etiquetado los usuarios. De esta forma es posible descubrir nuevas páginas que, según otros usuarios, pertenecen a la misma categoría. Delicious

²¹ <http://delicious.com/>

también proporciona información acerca de qué páginas son más populares o han sido añadidas recientemente. Esto permite recopilar los temas más tratados en cada momento de manera similar a los *trending topics* de *Twitter*.

La colección está compuesta por información de aproximadamente 2000 usuarios y cerca de 70.000 *URLs* (ver Tabla 5):

Delicious	
Usuarios	1.867
<i>URLs</i>	69.226
<i>URLs</i> principales	38.581
Relaciones bi-direccionales entre usuarios	7.668
Media Relaciones bi-direccionales-usuario	8.236
Etiquetas	53.388
Asignaciones de etiquetas	437.593
Media Asignaciones-usuario	234,383
Media Asignaciones- <i>URL</i>	6.321
Marcadores (par usuario- <i>URL</i>)	104.799
Media marcadores-usuario	56.132
Media marcadores- <i>URL</i>	1.514

Tabla 5 - Estadísticas de la Colección Delicious

La colección se ofrece dividida en varios archivos:

- **Archivo *user_contacts.dat*:** Contiene las relaciones de contactos entre los usuarios existentes en la base de datos, definiéndose relación de contacto entre dos usuarios como una *mutual fan relation* existente en Delicious. También se dispone de información del *timestamp* de las relaciones.
- **Archivo *bookmarks.dat*:** Contiene información sobre los marcadores de las *URL*.
- **Archivo *tags.dat*:** Contiene información de las etiquetas presentes en la base de datos.

- **Archivo user_taggedbookmarks.dat:** Contiene las etiquetas establecidas por un determinado usuario a un marcador. También está disponible el *timestamp* en el cuál se ha producido la asignación.
- **Archivo bookmark_tags.dat:** Contiene las etiquetas asociadas a un marcador y el número de veces que la etiqueta ha sido asignada.

La idea detrás del uso de Delicious para SR es que, si un usuario añade un nuevo marcador es equiparable a que marque como relevante un contenido. Teniendo esto en cuenta, gracias a Delicious se puede tener un conjunto de **interacciones de usuarios reales** que incluya sus preferencias, y poder recomendarle nuevos contenidos.

Delicious proporciona una gran cantidad de información asociada a las interacciones de los usuarios: dado que los usuarios están recomendando páginas Web, se puede acceder a su contenido y obtener **gran cantidad de contenido multimedia**, clasificado de manera social por los propios usuarios. Todas estas posibilidades convierten a Delicious en una de las fuentes de información más recomendables para su utilización en el área de SR basados en contenido.

5.2.1.4. Mendeley

Mendeley es una plataforma (Web y en versión Escritorio) para gestionar y compartir artículos científicos, permitiendo descubrir nuevos artículos relacionados. *Mendeley* permite a cada usuario recopilar los artículos científicos que sean de su interés y clasificarlos para su posterior lectura, así como relacionarse con otros usuarios que tengan intereses comunes.

En este contexto nace el *Mendely's DataTEL Challenge* [55], centrado en fomentar la investigación en SR de artículos científicos. Mientras que en otros dominios, como las películas, se ha llevado a cabo una extensa investigación en recomendación colaborativa y de contenidos, en el dominio de los artículos de investigación apenas se han desarrollado trabajos. Por ello, *Mendeley* ofrece una colección para la experimentación en SR, generada con la información de sus propios usuarios.

La colección consta de 4.8 millones de artículos de investigación seleccionados de una muestra de 50.000 usuarios activos en la plataforma (ver Tabla 6). De cada usuario se tiene el conjunto de artículos que aparece en su librería (al menos 20 artículos por usuario), así como información de cuáles de estos artículos han leído y cuáles han marcado como interesantes. Del total de artículos de la colección tan solo un 30% han sido leídos y únicamente un 13% han sido puntuados alguna vez.

Mendeley	
Usuarios	50.000
Artículos	4.848.724
Artículos Leídos	1.466.489
Artículos Puntuados	615.308

Tabla 6 - Estadísticas de la Colección Mendeley

El dominio de los artículos de investigación apenas ha sido abordado con anterioridad en SR, lo cual lo hace muy interesante para su utilización. Sin embargo, resulta imposible aplicar nuestra aproximación a la colección ya que no dispone de información textual asociada a los artículos. Para tratar de subsanar este problema se intentó expandir la colección añadiendo información acerca de los artículos, dado que *Mendeley* dispone de un *API* para acceder a sus contenidos. Sin embargo, debido a la anonimización de la colección, los identificadores de los usuarios y los artículos habían sido cambiados, imposibilitando la utilización del *API* para conocer que había visto cada usuario.

5.2.1.5. MovieLens

Esta colección está extraída de la base de datos de *MovieLens*²², una página para la recomendación de películas basándose en las películas marcadas como vistas por los usuarios anteriormente. La colección ha sido generada por el laboratorio de investigación del Departamento de Informática e Ingeniería de la Universidad de Minnesota, *GroupLens*²³ y ha sido utilizada dentro del *Workshop HetRec 2011*²⁴.

La colección está compuesta por más de 850.000 puntuaciones de 2133 usuarios sobre un total de 10.197 películas (ver Tabla 7). La información acerca de las películas que contiene la colección ha sido expandida mediante enlaces a *Internet Movie Databaes*²⁵ (IMDb), una base de datos con información de millones de películas, y a *RottenTomatoes*²⁶ (RT), otra base de datos de películas similar a IMDb.

²² www.movielens.org

²³ <http://www.grouplens.org>

²⁴ <http://ir.ii.uam.es/hetrec2011>

²⁵ <http://www.imdb.com/>

²⁶ www.rottentomatoes.com/

MovieLens	
Usuarios	2.133
Películas	10.197
Géneros de película	20
Asignaciones género→película	20.809
Media Asignaciones genero-película	2,040
Directores	4.060
Actores	95.321
Media Actores-película	22,778
Etiquetas	13.222
Asignaciones etiquetas→usuario, película	47.957
Media asignaciones etiquetas-usuario	22,696
Media asignaciones etiquetas-película	8,117
Puntuaciones	855.598
Media puntuaciones-usuario	404,921
Media puntuaciones-película	84,637

Tabla 7 - Estadísticas de la Colección MovieLens

El conjunto de datos es provisto en diferentes archivos estructurados de la siguiente forma:

- **Archivo movies.dat:** Contiene la descripción de las películas de la base de datos; esta descripción incluye:
 - Título en español.
 - ID de la película en IMDb.
 - *URLs* de las fotos asociadas a la película en IMDb.
 - ID de la película en *RottenTomatoes*.
 - *URLs* de las fotos asociadas a la película en *RottenTomatoes*.
 - Información asociada a la película en *RottenTomatoes* (críticas, puntuaciones, puntuaciones medias, numero de revisiones, ...).
 - Información asociada a las puntuaciones de la película en

RottenTomatoes.

- **Archivo `movies_genres.dat`:** Contiene los géneros asociados a cada película.
- **Archivo `movie_directors.dat`:** Contiene los directores asociados a cada película.
- **Archivo `movie_actors.dat`:** Contiene a los principales actores y actrices que aparecen en las películas. Los actores vienen ordenados de acuerdo al orden de aparición en cada película.
- **Archivo `movie_countries.dat`:** Contiene el país de origen de las películas.
- **Archivo `movie_locations.dat`:** Contiene las localizaciones de las películas.
- **Archivo `tags.dat`:** Contiene el conjunto de etiquetas presentes en el conjunto de datos.
- **Archivo `user_taggedmovies.dat`:** Contiene las etiquetas que cada usuario ha asignado a una película. Contienen también el *timestamp* en el cuál la asignación fue hecha.
- **Archivo `movies_tags.dat`:** Contiene las etiquetas asociadas a cada película, también contiene el número de veces que la etiqueta se ha asociado a la película.
- **Archivo `userRatedmovies.dat`:** Contiene las puntuaciones de los usuarios sobre las películas. También se incluye el *timestamp* en el cuál la puntuación fue asignada.

La colección ha sido extraída de MovieLens, un sistema de recomendación basado en filtro colaborativo y, por tanto, la información disponible la haría solo apta para experimentación en este tipo de sistemas. Por ello, de cara a su utilización en el HetRec de 2011, la colección fue ampliada incluyendo información textual asociada a las películas.

La colección liberada en el *HetRec*, ya ampliada, dispone de información textual asociada a los contenidos así como de información de los usuarios, haciéndola adecuada para todo tipo de aproximaciones de recomendación, como se puede comprobar en los *Proceedings* del *Workshop* [24]. Sin embargo, de cara a nuestra aproximación, de nuevo aparece el problema presentado anteriormente de la ausencia de correlación entre el interés de un usuario por un video y la descripción textual de éste: un usuario no ve una película únicamente por la descripción textual de ésta, sino que influyen otros factores (director, actores, efectos especiales,...).

5.2.1.6. Last.fm²⁷

Last.fm es un sitio web musical que permite a sus usuarios escuchar pistas de música individuales de su catálogo de más de 12 millones. Last.fm también tiene un sistema de recomendación que recopila las pistas escuchadas por cada usuario para generar un perfil y recomendar nuevos contenidos, utilizando un algoritmo basado en filtro colaborativo. Utilizando como fuente Last.fm se ha generado una colección para su utilización en el *Workshop HetRec 2011*.

La colección, obtenida Last.fm, está formada por unos 2000 usuarios y la información asociada a ellos como por ejemplo: información social de los usuarios o información del etiquetado social (ver Tabla 8).

LastFM	
Usuarios	1.892
Artistas	17.632
Relaciones entre usuarios (bi-direccionales)	12.717
Media Relaciones-usuario	13.443
Relaciones usuario escuchando a un artista	92.834
Media artistas escuchados-usuario	49.067
Media usuarios escuchando a cada artista	5.265
Etiquetas	11.946
Asignaciones de etiquetas (usuario, etiqueta, artista)	186.479
Media asignaciones-usuario	98.562
Media asignaciones-artista	14.891
Media etiquetas distintas usadas por cada usuarios	18.930
Media etiquetas distintas asociadas a cada artista	8.764

Tabla 8 - Estadísticas de la Colección de *Last.fm*

²⁷ <http://www.lastfm.es/>

La colección está compuesta por los siguientes archivos:

- **Archivo artists.dat:** Contiene la información acerca de los artistas escuchados y etiquetados por los usuarios.
- **Archivo tags.dat:** Contiene las etiquetas presentes en la colección.
- **Archivo user_artists.dat:** Contiene los artistas que han sido escuchados por cada usuario. También se dispone de la información temporal (*timestamp*) en el cuál se ha escuchado al artista.
- **Archivo user_taggedartists.dat:** Contiene las asignaciones de etiquetas a los artistas establecidas por cada usuario. También se dispone de la información temporal de cuándo se produjo la asignación.
- **Archivo user_friends.dat:** Contiene las relaciones de amistad existentes entre los usuarios de la base de datos.

El sistema de recomendación de Last.fm y, por consiguiente, la colección generada están enfocados únicamente a recomendaciones basadas en filtro colaborativo. La colección, además de los datos típicos de una colección enfocada a su uso en sistemas de filtro colaborativo, contiene información de contexto interesante como: información de las relaciones sociales de los usuarios o anotaciones sociales de los contenidos. Pese a que la colección es muy interesante, la limitación a sistemas de filtro colaborativo hace que no sea posible utilizarla para nuestra aproximación de modelado y recomendación.

5.2.2. Colecciones no enfocadas en Recomendación

Las colecciones presentadas en esta sección no han sido generadas pensando en utilizarlas en experimentación en el área de SR; sin embargo, se ha estudiado emplearlas dadas sus características.

5.2.2.1. Colección ImageCLEF - Photo Annotation

Photo Annotation es una de las tareas desarrolladas dentro de la edición de 2012 de *ImageCLEF*²⁸, uno de los *LAB* llevados a cabo dentro del *CLEF 2012*²⁹ para la evaluación experimental de sistemas de recuperación y clasificación de imágenes. Dentro de *Photo Annotation* se llevan a cabo dos subtareas, siendo interesante para este trabajo una de ellas: *Visual Concept detection, annotation and retrieval using Flickr photos*³⁰. Esta subtarea está enfocada a la detección, anotación y recuperación de conceptos presentes en fotografías mediante la información visual o textual de cada foto. Para esta subtarea la organización ha proporcionado un subconjunto de la

²⁸ <http://imageclef.org/2012>

²⁹ <http://clef2012.org/>

³⁰ <http://imageclef.org/2012/photo-flickr>

colección *MIRFLICKR* (compuesta por 1 millón de imágenes extraídas de *Flickr*³¹) formado por 200.000 imágenes.

La colección utilizada para esta tarea está formada por 200.000 imágenes, las cuales están acompañadas de información textual y visual que las describen.

Photo Annotation	
Fotos	200.000
Conceptos	91

Tabla 9 - Estadísticas de la Colección Photo Annotation

5.2.2.1.1. Información Visual

La información visual incluye varios tipos de características visuales que recogen las propiedades visuales de la imagen:

- **SIFT, C-SIFT, RGB-SIFT, OPPONENT-SIFT:** Se proporcionan una serie de características, basadas en SIFT, extraídas mediante la herramienta ISIS Color Descriptor. Estas características son provistas en varios archivos, cada uno de ellos correspondientes a un tipo de características: *train_features_sift*, *train_features_csift*, *train_features_rgbsift*, *train_features_opponentsift*.
- **SURF:** Se ha usado la herramienta *OpenSURF* para extraer estos descriptores. Estos descriptores se proveen en el archivo *train_features_surf*.
- **TOP-SURF:** Descriptores que representan las imágenes con *bag-of-words* basadas en SURF. Estos descriptores se proporcionan en el archivo *train_freatures_topsurf*.
- **GIST:** Se ha usado la herramienta *LabelMe* para extraer los descriptores. Estos descriptores están incluidos en el archivo *train_metadata_gist*.

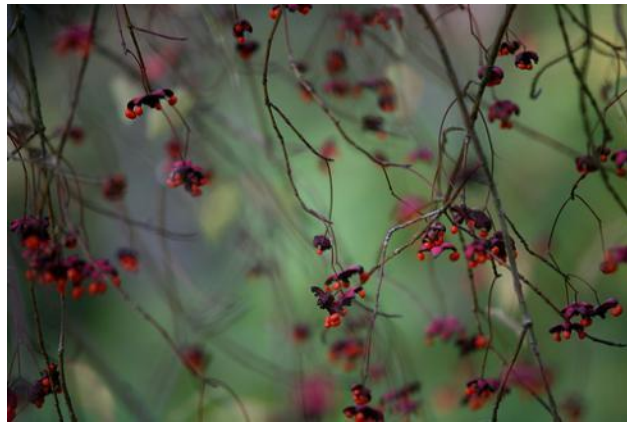
5.2.2.1.2. Información Textual

La información textual de cada imagen incluye:

- **Etiquetas de los usuarios de Flickr:** Etiquetas asignadas por los usuarios que las subieron a *Flickr*. A continuación se muestra un ejemplo de imagen de la colección y etiquetas asociadas a esta:
- **Metadatos EXIF:** Cada imagen tiene asociada la información EXIF de la cámara que realizó la fotografía.

³¹ <http://www.flickr.com/>

- **Información de usuarios:** Cada imagen tiene asociada la información del usuario propietario de la fotografía. A continuación se muestra un ejemplo de imagen e información de este tipo asociada:



Etiquetas (tags): berries, red, green, bokeh, blur, arboretum, westonbirt, fall, autumn, 510fav, 1025fav, fv10, fv5, bigpicture2008

Figura 7 - Ejemplo de Imagen y Etiquetas asociadas



Información asociada:

Date uploaded: 2008-02-15

License: Attribution License

License URL: <http://creativecommons.org/licenses/by/2.0/>

Picture title: Better Homes and Gardens

Owner name: decor8 ☆ holly

Owner username: decor8

Owner id: 21784738@N00

Figura 8 - Ejemplo de Imagen e información de su propietario asociada

Esta colección presenta dos **inconvenientes**: en primer lugar, como ya se comentó, no es una colección específicamente enfocada a recomendación, aunque pueda ser utilizada como tal. El segundo inconveniente está relacionado con el método de evaluación utilizado en la tarea. Debido al gran tamaño de la colección no es posible disponer de un *ground truth*, por lo que la evaluación se va a realizar mediante *pooling*; esto es, se evaluarán los sistemas en función de los resultados de todos los participantes, siendo considerado mejor a aquel sistema que devuelva resultados parecidos al resto. Puesto que en este trabajo se plantea desarrollar un sistema que encuentre contenidos originales, es posible que estos contenidos originales queden fuera del conjunto de resultados devueltos por otros sistemas, haciendo que el sistema obtenga una mala puntuación a la hora de evaluarse. Dados estos inconvenientes, pese a que esta colección sea de gran utilidad para sistemas de recuperación de información, su utilización es SR no parece aconsejable.

5.2.2.2. Tass

El TASS (o Taller de Análisis de Sentimientos en la SEPLN) es un Taller desarrollado dentro de la SEPLN (Sociedad Española de Procesamiento del Lenguaje Natural) para la investigación y experimentación en el análisis de reputación y sentimientos en la Web. Para el Taller se ofrece una colección extraída de *Twitter*.

La colección presentada está compuesta por 70.000 *tweets* en Español escritos por aproximadamente 200 personalidades del mundo de la política, economía, comunicación o cultura entre otras áreas. El total de los *tweets* se encuentra categorizado en 10 temáticas: política, otros, entretenimiento, economía, música, fútbol, cine, tecnología, deportes y literatura.

TASS	
Usuarios	158
Tweets	70.000
Temáticas	10

Tabla 10 - Estadísticas de la Colección TASS

La colección proporcionada en el TASS está enfocada al análisis de sentimientos en *Twitter* pero, dadas las características de la colección puede adaptarse al campo de la recomendación de contenidos. La colección contiene un conjunto de ítems (*tweets*), con un contenido textual que los define (contenido textual del *tweet*) y un conjunto de usuarios que interaccionan sobre estos contenidos. Pese a que sobre el papel la colección parece adecuada para la aproximación presentada en este trabajo, inspeccionando ésta en más detalle se ve observan dos problemas: el tamaño de la colección es escaso y el contenido textual asociado a cada *tweet* es muy reducido (debido a la limitación de 140 caracteres por *tweet*).

Aunque en principio esta colección no es adecuada para recomendación de contenidos, otro tipo de colecciones extraídas de *Twitter* podría ser útiles: colecciones más grandes o colecciones con la posibilidad de expandir el contenido textual de cada *tweet* (e.g. a través de enlaces a otras fuentes que los *tweets* puedan tener).

5.2.2.3. Colección Medical Retrieval Task³²

La *Medical Retrieval Task* es una tarea organizada dentro del *CLEF 2012 (Conference and Labs of the Evaluation Forum)*, centrada en la investigación de recuperación de imágenes médicas. Para esta tarea se ha generado una colección extraída desde la base de datos de *PubMed Central*³³, un repositorio de artículos médicos con imágenes asociadas.

La colección de la *Medical Retrieval Task* está formada por 75.000 artículos médicos de temática biomédica, conteniendo más de 300.000 imágenes. Cada una de las imágenes de la colección tiene una descripción textual de la imagen y están clasificadas utilizando una taxonomía de dos niveles compuesta por 31 clases. A continuación se muestra un ejemplo de imagen de la colección junto con su descripción textual:

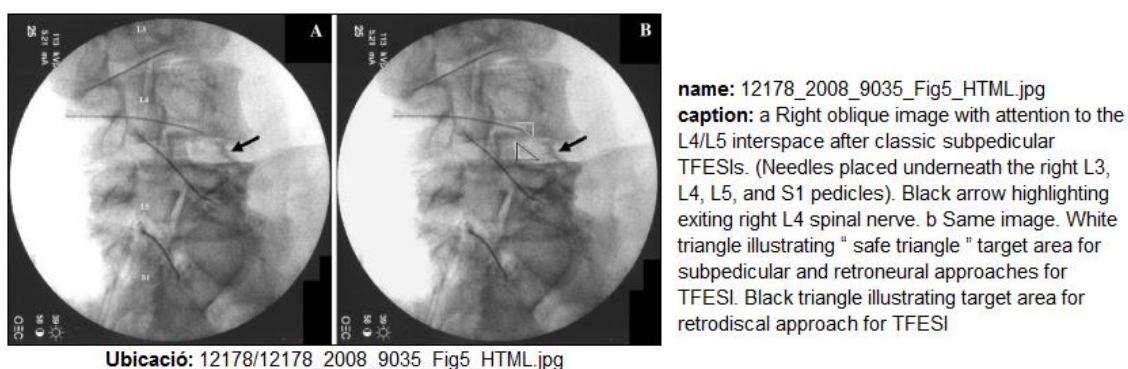


Figura 9 - Ejemplo de Imagen de la Colección y Descripción asociada

Medical	
Imágenes	300.000
Artículos	75.000
Categorías	31

Tabla 11 - Estadísticas de la Colección Medical

³² <http://www.imageclef.org/2012/medical>

³³ <http://www.ncbi.nlm.nih.gov/pmc/>

La colección está muy centrada para utilizarla en el ámbito de la recuperación de información multimedia por lo que no encaja en el ámbito de los sistema de recomendación de contenidos, pese a que la información textual disponible es bastante extensa.

5.2.3. Colecciones propias

En esta sección se presentan colecciones que han sido generadas dentro de este trabajo. Se ha planteado su descarga y posible utilización dado el interés de la información disponible en las fuentes de datos de las cuáles han sido extraídas. Las colecciones estudiadas se detallan a continuación.

5.2.3.1. Colección extraída de Storify³⁴

Storify es un sitio web para crear **historias sociales**, recopilando contenidos multimedia desde otras redes sociales (*Facebook, Twitter,...*) o desde la Web. De esta manera se tiene en un mismo panel todos los datos acerca de un determinado tema. Las historias sociales pueden ser embebidas también en otros sitios o difundidas de manera sencilla para llegar al mayor número de usuarios posibles. Además, *Storify* también permite al usuario generar contenidos propios para dar un formato estructurado a la historia (ver Figura 10).

En la Figura 11 se muestra un ejemplo de historia social extraído de *Storify* sobre los *Oscar*. En él se pueden ver los principales componentes de una historia social: contenidos agregados por el usuario, en este caso texto para introducir la historia y crear un hilo conductor de esta; agregación de contenidos externos, *tweets* en este caso; estadísticas asociadas a la historia (seguidores, visitantes, suscriptores); historias relacionadas y también otras historias generadas por el usuario que ha creado la historia mostrada. *Storify* permite acceder a los contenidos originales a través de la interfaz de la historia, de igual modo también se puede acceder a los perfiles de los usuarios que hayan generado el contenido original.

³⁴ <http://storify.com/>

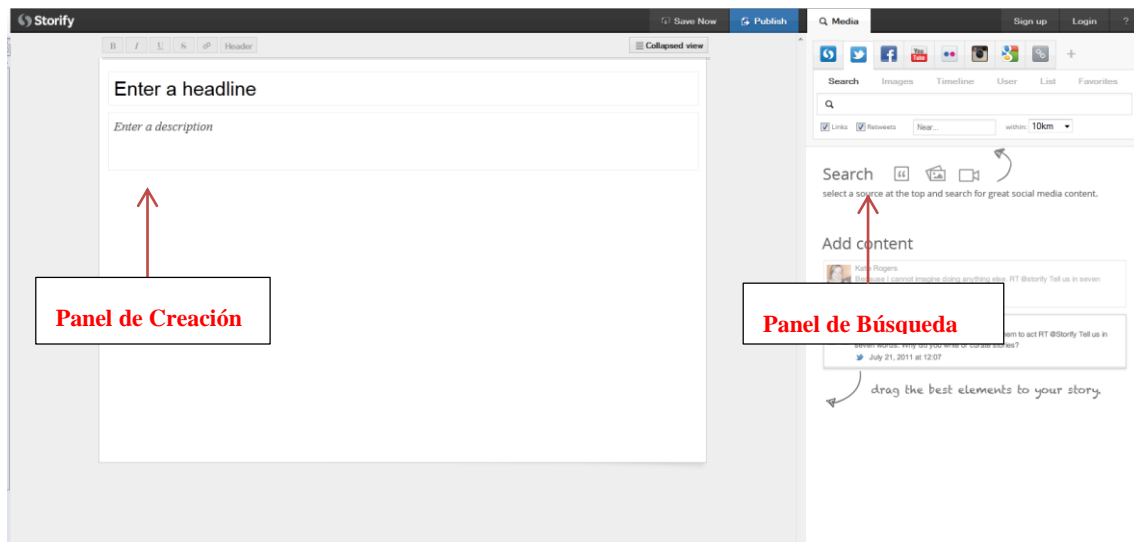


Figura 10 - Panel de creación de historias

Storify enables anyone to create social stories.

Get started

Storified by Halief Ardiasyah
2 hours ago · 49 views · 110 · 20

#Oscars 2012

Got up early this morning to catch *The 84th Academy Awards*. Caught the Red Carpet and midway through the Oscars itself in the morning and caught the second half on a rerun in the evening.

Just heard Vin Diesel will be walking the Oscar red carpet in character as Dominic Toretto from *Fast 5*. Excited.
2 days ago · Aziz Ansari

Er, right.

Moi's big day is here! #Oscar: here I come!
2 days ago · Miss Piggy

And we're LIVE from the Red Carpet NOW! #ERedCarpet #Oscars
1 day ago · Halief Ardiasyah

Piggy and Kermit arriving on the carpet! Longest-lasting relationship in Hollywood! #eredcarpet
1 day ago · E! Live Events

Miss Piggy and Kermit on the red carpet: Best Dressed, Muppet category. bzfd.it/s:SGok #bestdressed (via @BuzzFeed) Awww...
1 day ago · Halief Ardiasyah

We missed Kermit The Frog & Glee's @DarrenCris' performance of 'Rainbow Connection' during the E! #Oscars Pre-Show an hour ago! Le sigh.
1 day ago · Halief Ardiasyah

The Artist's Penelope Ann Miller: A long way since starring alongside Alec Baldwin in... *The Shadow*. #Oscars <http://lockerz.com/s/187507494>
1 day ago · Halief Ardiasyah

halief
Halief Ardiasyah
+ Subscribe

Mostly talks about film, music, TV, video games & sweet eats. Tweets in English, Indonesian, and glibberish.

Statistics

Published stories	2
Subscribers	0
Subscribed to	4
Total views	49
storify.com	44
other	5

RELATED STORIES

Live coverage: Golden Globes 2012

Oscar night's true Twitter star? Angelina's leg

TODAY's Oscar watch party

More from halief

On A Sunday: Jimmy Eat World - Live In Jakarta

halief
11 months ago · 292

feedback

Figura 11 - Ejemplo de historia social

Storify dispone de un API para el acceso a sus contenidos y poder generar una colección con ellos. En la siguiente dirección se puede ver una descripción completa del API:

<http://dev.storify.com/API/summary>

Los aspectos más importantes del API se resumen a continuación:

5.2.3.2.1. Acceso

El acceso principal al *API* se realiza a través de la dirección:

<http://API.storify.com/v1>

5.2.3.2.2. Consultas

Las consultas al *API* se realizan mediante la ejecución de peticiones *POST*³⁵ y *GET*³⁶ y las respuestas se reciben en un formato *JSON*³⁷.

5.2.3.2.3. Autenticación

En ocasiones es necesario estar autenticado para acceder a ciertos contenidos del *API*. Para autenticarse es necesario proporcionar un *API_key* y un *token* de usuario. Para obtener el *token* es necesario realizar una llamada al *API* proporcionándole un nombre de usuario válido y su contraseña. Para obtener el *API_key* hay que ponerse en contacto con storify y solicitarla a través de la dirección API@storify.com.

Para poder manejar respuestas que sean muy extensas, el *API* pagina los resultados devueltos, estableciendo el número de resultados por página y la página que ha sido devuelta.

5.2.3.2.4. Errores

Cada respuesta tiene un campo *code* que contiene el código HTML de respuesta del servidor para comprobar si se ha producido algún error.

Storify presenta varias características que la hacen interesante para su uso en este trabajo:

- Existencia de gran cantidad de **contenido multimedia clasificado** en diferentes categorías, incluyendo gran cantidad de texto asociado a estos contenidos multimedia.
- Posibilidad de acceder a **datos reales de usuarios**: historias que han creado, historias que han visitado, historias que son de su interés,... . Esto resulta útil a la hora de evaluar SR ya que se dispone tanto de la información sobre los gustos del usuario como del contenido de estos gustos.
- Posibilidad de **acceder a contenidos de diferentes redes sociales** (entre otras *Facebook* o *Twitter*) unificados en un mismo sitio. Además se puede acceder al contenido original presente en las redes sociales referenciadas.

³⁵ http://en.wikipedia.org/wiki/POST_%28HTTP%29

³⁶ <http://opensta.org/docs/sclref10/imp3ln43.htm>

³⁷ <http://www.json.org/>

- **Facilidad de acceso** a sus contenidos ya que dispone de un *API* bien estructurada y los datos tanto de entrada como de salida son formatos fácilmente tratables (consultas HTML y JSON).

A pesar de lo interesante de la información disponible, inspeccionando ésta en más detalle se vio que, en general, la información era muy dispersa: se tenían pocos usuarios con suficiente cantidad de contenidos en sus perfiles o las relaciones entre usuarios eran escasas. Por ello a pesar de que las posibilidades que ofrece la plataforma son muchas, la aplicación real de los datos limita la usabilidad de la colección generada en la forma necesaria para la experimentación en este trabajo.

5.2.3.2. Colección de Noticias Buscamedia

Dentro del proyecto Buscamedia [27] se ha generado una colección de noticias, extraídas de las ediciones digitales de periódicos de temáticas variadas, (deportes, política, nacionales, locales,...) tomando todo el contenido de dichas noticias (título, subtítulo, etiquetas, contenido textual,...). Para la obtención de las noticias, se ha empleado los canales RSS (*Really Simple Syndication*) de los periódicos seleccionados.

La colección tiene un carácter monolingüe (todas las noticias están en español) para facilitar una primera aproximación al problema, pero en un futuro podría extenderse el trabajo a colecciones multilingües (ya se cuenta con transcripciones de noticias en catalán y euskera para su futura inclusión en la colección).

La colección está compuesta por 3 tipos de documentos:

- **Transcripciones de Videos.** Compuesta por 21 documentos referentes al texto transcrito de videos en castellano. Aunque algunos de los videos están en catalán, sus transcripciones son en castellano.
- **Páginas Web.** Compuesta por un conjunto de 30 páginas web en formato HTML en castellano, cuya temática está relacionada con los videos anteriores.
- **Noticias.** Conjunto de 5.360 noticias en castellano en formato HTML de 14 fuentes diferentes.

Para este trabajo, se decidió seleccionar de la colección original sólo el conjunto de las 5360 noticias, descartándose tanto las páginas web como las transcripciones de video. Esto se hizo así porque la información textual de los videos y las páginas era escasa y, por tanto, poco representativa.

Ya que los documentos/noticias anteriores provienen de diferentes fuentes, es necesario un preprocesamiento de cada documento, unificando el formato de todas las noticias. Cada noticia ha sido representada como un fichero con formato XML, el cual define un campo para cada uno de los contenidos de la noticia (sección, título, contenido,...)

Para el trabajo posterior sobre la colección era necesario que cada noticia tuviese definida una categoría. La colección poseía una anotación de categorías para cada periódico/fuente basada en la sección del periódico en la que estaba incluida. Sobre estas secciones se realizó una primera clasificación de la colección pero se descubrieron varios problemas:

- **Falta de normalización:** Las secciones no estaban normalizadas, en su lugar cada periódico tenía sus propias secciones, lo que hacía imposible generar secciones generales para todos los periódicos:

Ejemplo: nacional \approx España

- **Diferente criterio de clasificación:** La clasificación de las noticias dependía de la opinión subjetiva del periódico, lo que propicia que secciones de una temática parecida puedan estar en secciones diferentes, según de que periódico procedan.
- **Ausencia de clasificación real:** En algunos periódicos, como por ejemplo El País, no existía una clasificación por secciones real, lo que hacía imposible su clasificación.

Ejemplo de categoría de El País:

http://www.elpais.com/articulo/cataluna/Demasiado/politicos/elpepiespcat/20101019elpcat_4/Tes?print=1

Estos problemas hacían que el número de categorías fuese excesivo, aproximadamente unas 400 para poco más de 5000 noticias. Debido a esto se intentó reducir las categorías de las noticias, re-etiquetando algunas categorías de manera manual, teniendo en cuenta el conocimiento previo acerca de los problemas de la categorización planteados. Tras esta agrupación el número de secciones quedó reducido a aproximadamente 100 secciones. En la Figura 13 y Figura 14 se pueden ver los repartos de las noticias antes y después del re-etiquetado. En estas gráficas se puede comprobar que la mayoría de noticias están agrupadas en un número reducido de secciones, mientras que hay una gran cantidad de secciones que apenas contienen noticias. Como era de esperar al agrupar las secciones esta agrupación es aún más acusada.

Aunque la agrupación de noticias dio lugar a un número mucho más reducido de secciones, consiguiendo una categorización más adecuada, los problemas planteados siguen persistiendo. Por todo ello se decide desechar esta clasificación, dependiente de la fuente y emplear en su lugar un clasificador para, en función del contenido de la noticia, establecer un conjunto de categorías normalizado a todas las noticias.

La herramienta empleada para este propósito es *Stilus Class*³⁸ que proporciona una categorización de dos niveles (categoría y subcategoría) basándose en el contenido de la noticia y en un conjunto de categorías predefinidas. *Stilus* utiliza una aproximación híbrida entre un clasificador basado en kNN (*k-Nearest Neighbor*) y un sistema experto basado en reglas [137]. A continuación se muestra un ejemplo de clasificación de *Stilus* de un artículo:

Ejemplo de categorización de Stilus:

```
<categoria>deporte</categoria>  
  <subcategoria>baloncesto</subcategoria>
```

Además de las categorías de *Stilus* se han creado dos nuevas: miscelánea y error. *Miscelanea* se emplea cuando el clasificador de *Stilus* no es capaz de definir una categoría o subcategoría al tener la noticia un carácter muy genérico. Error se emplea cuando se produce un fallo en la clasificación.

³⁸ <http://api.daedalus.es/stilusclass-info>

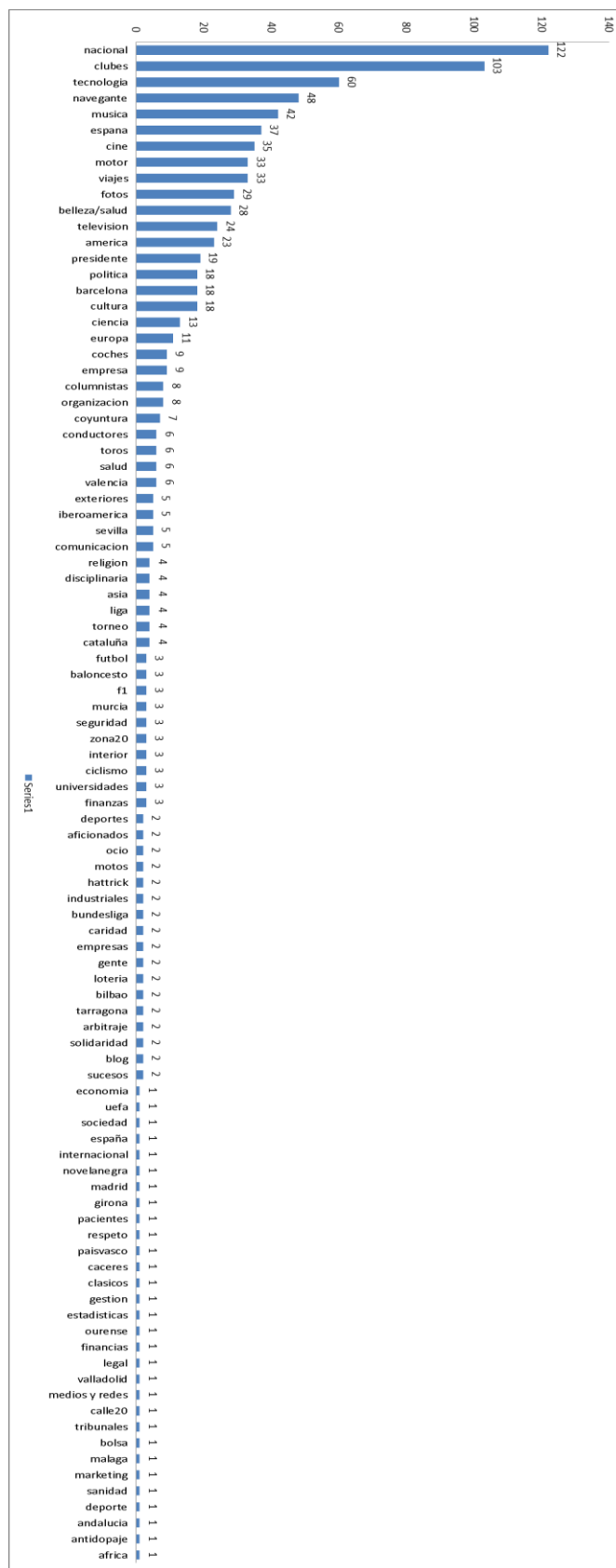


Figura 12 - Secciones antes de su agrupación

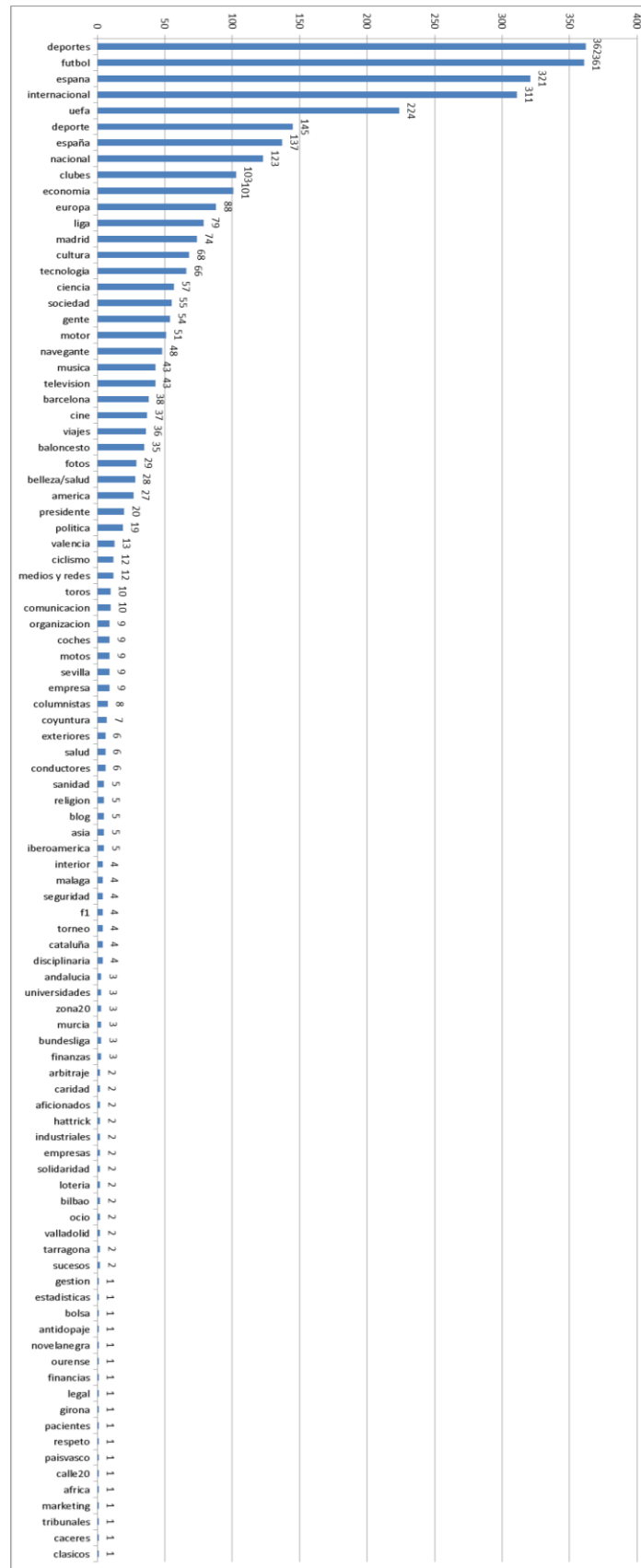


Figura 13 - Secciones tras su agrupación

Con la categorización previamente realizada, se ha generado un archivo de categorías para facilitar la clasificación de la colección. En este archivo se tienen las noticias clasificadas por categoría y subcategoría. La estructura de este archivo se puede ver en Figura 14.

El archivo está organizado en categorías, por cada categoría se tiene una lista de subcategorías, en las cuales están incluidas las noticias (todas las noticias tienen que estar dentro de una subcategoría). Cada noticia está representada a su vez por una entrada con su nombre y el lenguaje de la noticia (en la actualidad solo se tienen noticias en español). Por cada categoría y subcategoría se tiene un atributo con el número de apariciones de esta en el total de la colección.

Los datos contenidos en esta colección se adaptan perfectamente a las necesidades de experimentación para el enfoque de recomendación presentado en este trabajo. A diferencia de lo que ocurre con las películas, el interés de un usuario por una noticia se basa en el contenido de esta principalmente.

La principal desventaja a la hora de utilizar esta colección es que no dispone de información de usuarios que interactúen con los contenidos presentes. Esto imposibilita la ejecución de ninguna aproximación de recomendación, tomando como base esta colección. Una posible solución al problema es simular un conjunto de usuarios interactuando con la colección; sin embargo, como ya se ha visto, es importante contar con datos de usuarios reales para la experimentación en SR (datos reales dan lugar a conclusiones contrastadas). Aún a pesar de la falta de datos de usuarios, esta colección se puede utilizar para generar un sistema de consulta y recomendación de noticias que recopile información de los usuarios que lo utilicen, pudiendo tener así información real.

```

<categoria apariciones="34">
  <nombre>educación</nombre>
  <subcategoria apariciones="21">
    <nombre>universidad</nombre>
    <id lang="es">20minutos_es_-191231287642875.6303</id>
    <id lang="es">20minutos_es_113941287427637.1245</id>
    <id lang="es">20minutos_es_494341287614271.8772</id>
    <id lang="es">abc_es_-122701287699481.5607</id>
    <id lang="es">abc_es_-262851287733379.5059</id>
    <id lang="es">abc_es_-642571287528464.4997</id>
    <id lang="es">abc_es_757851287476516.288</id>
    <id lang="es">elmundo_es_-404321287699365.9917</id>
    <id lang="es">elmundo_es_270271287733238.1609</id>
    <id lang="es">elpais_es_-620111287655151.3281</id>
    <id lang="es">elpais_es_490031287555946.0042</id>
    <id lang="es">elpais_es_542121287410064.3262</id>
    <id lang="es">larazon_es_609021287733430.7445</id>
    <id lang="es">publico_es_281611287699419.3249</id>
    <id lang="es">publico_es_83681287699398.7802</id>
    <id lang="es">upi_es_-160821287498836.9561</id>
    <id lang="es">upi_es_-218721287433443.524</id>
    <id lang="es">upi_es_-370081287732470.9311</id>
    <id lang="es">upi_es_-534921287585162.3812</id>
    <id lang="es">upi_es_-910841287732495.3332</id>
    <id lang="es">upi_es_231471287527652.2435</id>
  </subcategoria>
  <subcategoria apariciones="10">
    <nombre>escuelas</nombre>
    <id lang="es">20minutos_es_-294291287642563.1594</id>
    <id lang="es">20minutos_es_-497971287441306.8546</id>
    <id lang="es">20minutos_es_132021287427211.2568</id>
    <id lang="es">abc_es_-156051287559534.8688</id>
    <id lang="es">elmundo_es_-296531287607594.9381</id>
    <id lang="es">elpais_es_-52251287607064.8665</id>
    <id lang="es">elpais_es_520381287410003.1385</id>
    <id lang="es">larazon_es_341541287733431.2238</id>
    <id lang="es">publico_es_-632251287607622.6545</id>
    <id lang="es">upi_es_-250041287606887.841</id>
  </subcategoria>
  <subcategoria apariciones="3">
    <nombre>miscelanea_educación</nombre>
    <id lang="es">20minutos_es_-46431287698888.7135</id>
    <id lang="es">20minutos_es_126221287410473.0055</id>
    <id lang="es">20minutos_es_312561287699234.6565</id>
  </subcategoria>
</categoria>

```

Figura 14 - Representación de la colección

5.2.3.2. Colección descargada de Pinterest³⁹

Pinterest es un sitio web social que permite compartir fotos al estilo de un tablón de anuncios. Mediante *Pinterest* los usuarios pueden crear y gestionar tableros enfocados en contenidos específicos en los cuales publicar sus *PINs*. Cada uno de los *PIN* está compuesto por una imagen, la cual puede ser vista, distribuida (mediante re-*PINs*) o valorada (de manera similar al “me gusta” de *Facebook*) por los usuarios. Además, los usuarios también pueden añadir comentarios a los *PINs*. *Pinterest* permite también compartir sus contenidos en otras redes sociales, como *Twitter* o *Facebook*, o en las páginas personales de los usuarios, mediante el botón de Pin It. En la Figura 15 se puede ver el aspecto que presenta la interfaz de *Pinterest*:

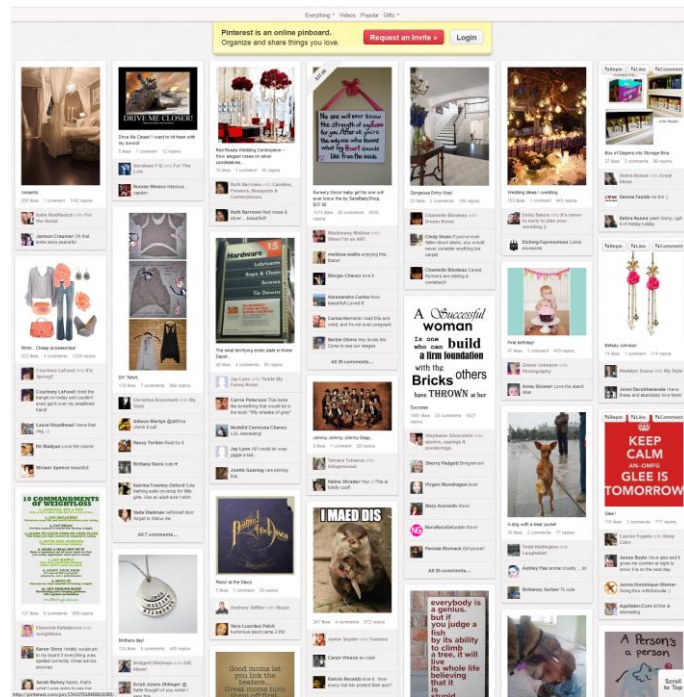


Figura 15 - Interfaz de *Pinterest*

Algunas de las características más importantes a destacar de *Pinterest* son:

- **Gran número de Usuarios:** En Enero de 2012 *comScore* estimaba que *Pinterest* tenía 11.7 millones de usuarios únicos [35] [12], siendo uno de los principales sitios sociales para compartir imágenes.
- **Rápido Crecimiento:** El crecimiento de *Pinterest* desde su creación a principios de 2010 ha sido espectacular, alcanzando un 87% durante 2011. Esto ha hecho que *Pinterest* sea el sitio web que más rápido ha alcanzado los 10 millones de visitantes únicos [35].

³⁹ <http://Pinterest.com/>

- **Integración con otras redes sociales:** *Pinterest* permite compartir sus contenidos en otras redes sociales como *Facebook* o *Twitter*. Esto permite conseguir más información sobre los perfiles de los usuarios, ampliándolos con los contenidos de estas otras redes sociales
- **Existencia de un API para obtener sus contenidos:** Es posible acceder a los contenidos de *Pinterest* mediante su *API*, facilitando la obtención de información. A la fecha de redacción de este documento el acceso al *API* está restringido, lo que imposibilita trabajar sobre *Pinterest*.
- **Clasificación del contenido en categorías:** Un aspecto interesante de *Pinterest* es que todas sus imágenes están clasificadas en categorías con el añadido de que esta clasificación se ha llevado a cabo por los propios usuarios.
- **Contenido textual asociado a las imágenes:** Las imágenes presentes en *Pinterest* tienen asociadas descripciones de los autores, así como los comentarios realizados por otros usuarios.
- **Información de las relaciones entre usuarios:** Se puede tener acceso a la información relativa a las relaciones entre usuarios (seguidores de un usuario, seguidores de una determinada imagen o categoría,...).

Pinterest tiene un *API* a través del cual se puede acceder a sus contenidos. En la actualidad este *API* no está público debido a modificaciones que se están realizando sobre ella. Las consultas al *API* se realizan mediante consultas POST a la dirección anteriormente dada y los datos de respuesta son proporcionados en formato JSON, por lo que su tratamiento es muy sencillo. Una información detallada sobre el *API* se puede encontrar en la siguiente dirección:

<http://tijn.bo.lt/Pinterest-API>

Para poder generar una colección mediante *Pinterest*, las partes del *API* más interesantes son: **Obtención de usuarios, Obtención de Categorías y Obtención de Tablones (*Boards*):**

- **Obtención de Usuarios:** La obtención de información de los usuarios se realiza a través de la dirección:

<https://API.Pinterest.com/v2/users/nombreusuario/>

donde *nombreusuario* es el nombre de usuario del que se quiere obtener la información. La información de un usuario viene estructurada en el conjunto de *boards* en los que el usuario participa.

- **Obtención de Categorías:** La obtención de los datos referentes a una categoría se obtienen mediante la dirección:

<https://API.Pinterest.com/v2/all/?category=nombrecategoría&limit=limite&page=página>

donde *nombrecategoría* es la categoría de la que se quiere conocer la información y *limite* y *página* son parámetros de la paginación de los resultados referidos al límite de resultados por página y al número de página que se está mostrando. La información de una categoría obtenida viene dada por el conjunto de *PINs* que la conforman.

- **Obtención de Tablones:** La obtención de la información sobre un tablón (o board) se lleva a cabo mediante la dirección:

<https://API.Pinterest.com/v2/boards/nombreusuario/nombreboard/>

donde *nombreusuario* es el usuario propietario del tablón del que se quiere obtener la información y *nombreboard* es el nombre del propio tablón. De cada tablón se obtiene, al igual que en las categorías, el conjunto de *PINs* que conforman el tablón.

Para almacenar la información obtenida a través del *API* se ha decidido utilizar una base de datos *Apache Cassandra*. Teniendo en cuenta la estructura de los JSON y los posibles usos futuros de la información recopilada se ha diseñado la estructura de base de datos. Además de almacenar la información textual sobre usuarios, categorías y *boards*, se ha decidido almacenar también el conjunto de imágenes asociadas a los *PINs*. Las imágenes se han descargado por separado y han sido almacenadas en formato JPG en un directorio por cada categoría de imágenes.

Las características expuestas con anterioridad hacen de *Pinterest* una fuente de datos muy interesante de cara a generar colecciones tanto para búsqueda multimedia como para recomendación. Por un lado, *Pinterest* ofrece un gran repositorio de imágenes expandidas con información textual por los propios usuarios y, además, categorizada de manera manual por el creador de la imagen. Por otra parte gracias a *Pinterest* se puede disponer de un corpus de usuarios con su actividad y las relaciones entre ellos. Dado el éxito y gran crecimiento de *Pinterest*, representa una de las fuentes de información sobre usuarios más atractiva de la actualidad.

Como este tipo de colecciones no está enfocado a recomendación propiamente dicha habría que estudiar cómo procesar los datos almacenados para que puedan ser aprovechados para tal fin. Otra de las limitaciones que puede tener *Pinterest* como fuente de datos multimedia es que, si bien las imágenes disponen de una información textual asociada, a menudo esta es muy escasa lo que dificultaría su utilización. Esto haría que también fuese difícil generar perfiles de usuario que realmente fuesen significativos de sus intereses.

5.2.4. Comentarios finales

En la Tabla 12 se muestra un resumen de las colecciones presentadas de acuerdo a las principales características son interesantes para la aproximación planteada.

De las características presentadas en la tabla, las dos más interesantes son: la existencia de perfiles de usuarios y de texto asociado a los ítems de la colección (videos, noticias, *tweets*,...). Para poder llevar a cabo el trabajo planteado es indispensable que la colección cumpla ambos requisitos. A este respecto, únicamente 3 colecciones no cumplen ambas condiciones, imposibilitando su uso: *Mendeley*, que no tiene texto asociado a los artículos; y *Photo Annotation* y *Medical*, las cuales no tienen perfiles de usuario. Otras tres colecciones a pesar de cumplir ambas condiciones tienen limitaciones a destacar: *VideoLectures* y *Tass* tienen una información textual asociada a los ítems muy escasa, dificultando la generación de los modelos de los usuarios; y *Buscamedia* cuyos perfiles de usuarios son simulados sin basarse en información real.

Otro aspecto más sutil que haría desechar una colección es la correlación entre el texto asociado a los ítems de la colección y el interés del usuario por dichos ítems, esto es, el interés del usuario por un ítem en algunos dominios no se basa en la descripción textual de éste. En las colecciones presentadas, este hecho se da en las colecciones de *VideoLectures*, *MovieLens* y *Last.fm*. *VideoLectures* se compone de conferencias científicas; en este campo el usuario normalmente está más interesado en el autor de la conferencia o el evento/congreso al que se asocia. *MovieLens* se compone de películas; en este dominio los usuarios suelen estar interesados, por citar algún aspecto, en los actores que participan, el director de la película o en los efectos de la película. Por último, *Last.fm* está compuesta por canciones; en este ámbito los aspectos que usualmente interesan al usuario son autor/es de las canciones o género musical.

Por último, en la tabla se incluyen otros dos aspectos a tener en cuenta sobre las colecciones: la existencia de una clasificación sobre la colección y la posibilidad de acceder a información del contexto de los usuarios. Si bien no son aspectos decisivos a la hora de decidirse por una colección, su existencia sí que haría más interesante el uso de la colección, de cara a mejorar los modelos de usuario generados.

Observando la tabla, las únicas colecciones que cumplen con todos los requisitos, convirtiéndolas en las mejores candidatas, son *Delicious* y *Buscamedia*, aunque en el caso de *Buscamedia* tanto los perfiles de usuario como la información de contexto se simulan en base a datos no reales. *Delicious* por su parte presenta el problema de la

gestión de los contenidos. La colección no tiene los contenidos propiamente dichos, sino que tiene los enlaces (URLs) a los contenidos. Esto presenta el problema de gestionar la descarga y, una vez descargados, preprocesar los contenidos y extraer aquellos que realmente sean relevantes.

Del lado contrario, la falta de información textual, perfiles de usuario o de correlación entre texto e interés descarta el uso de las colecciones *VideoLectures*, *Mendeley*, *MovieLens*, *Last.fm*, *PhotoAnnotation* y *Medical*.

Características Colección	Perfiles de usuario	Texto asociado a los ítems	Colección clasificada	Relación Texto asociado / Interés	Contexto
Yahoo	X		X	X	X
VideoLectures	X	X* ¹	X		X
Delicious	X	X	X	X	X
Mendeley	X			-	
MovieLens	X	X	X		
Last.fm	X	X			
Photo Annotation		X	X	X	
Tass	X	X* ¹	X	X	X
Medical		X	X	X	
Storify	X	X		X	X
Buscamedia	X* ²	X	X	X	X* ²
Pinterest	X	X* ¹	X	X	

Tabla 12 - Resumen de las colecciones

*¹ Existe información pero es muy escasa

*² Existe esta información pero es simulada sobre datos no reales

Capítulo 6. Evaluación de Sistemas de Recomendación

La evaluación de los SR, y por consiguiente de las medidas de evaluación, ha sido uno de los temas clave en la investigación en SR y ha sido ampliamente estudiada a lo largo de la literatura. La evaluación, y por tanto las medidas, se pueden clasificar en dos tipos: Evaluación experimental de los resultados y evaluación centrada en el usuario.

6.1. Evaluación experimental

Esta evaluación se basa en la utilización de métricas que examinan el rendimiento del sistema comparando unos resultados experimentales con un conjunto de juicios de relevancia, sin que sea necesaria la interacción del usuario para llevar a cabo la evaluación. En este tipo de evaluación se obtiene una valoración cuantitativa del funcionamiento de un sistema, de acuerdo a ciertos parámetros, mediante la utilización de alguna medida de evaluación. Para definir qué medida de evaluación utilizar se debe tener en cuenta qué característica del sistema se desea medir y escoger el tipo de medida que mejor se adapte. Según el tipo de parámetros utilizados para medir el rendimiento, las medidas de evaluación se pueden clasificar en: Medidas de Exactitud, Medidas de Precisión, Medidas de Satisfacción y Medidas de Diversidad y Novedad.

6.1.1. Medidas de Exactitud

En gran parte de las investigaciones, las medidas utilizadas para llevar a cabo la evaluación experimental se basan en medir la **exactitud** de un sistema, comparando las puntuaciones estimadas por el mismo con las puntuaciones reales de los usuarios (siempre y cuando éstas estén disponibles). Para la ejecución de este tipo de medidas es necesario que tanto las predicciones del sistema como las puntuaciones reales de los usuarios estén representadas de manera numérica.

La idea detrás de utilizar este tipo de medidas es que si un sistema produce recomendaciones más exactas, éste será preferido por los usuarios. Las medidas basadas en exactitud más comúnmente utilizadas son: **MAE** y **RMSE** [24].

MAE (*Mean Absolute Error*) evalúa el error producido en las predicciones comparándolas con los resultados esperados. Se define formalmente como:

$$MAE = \frac{1}{n} \sum_{i=1}^n abs(f_i - y_i) \quad \text{Ecuación 21}$$

donde n es el total de predicciones y $abs(f_i - y_i)$ es el error absoluto o e_i , definido por la diferencia entre la predicción dada por el sistema f_i y el resultado real y_i . Mediante el uso del valor absoluto, MAE mide los errores en ambos sentidos, tanto por exceso como por defecto, en la recomendación.

RMSE (*Root Mean Square Error*), al igual que MAE, mide el error entre los valores predichos por el sistema y los valores reales observados. La formulación se puede ver en la Ecuación 22:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n abs(f_i - y_i)^2}{n}} \quad \text{Ecuación 22}$$

Dado que los errores se elevan al cuadrado antes de que se aplique la media, RMSE otorga un peso relativo más alto cuanto mayor sea el error, haciendo que sea más útil cuando se quiere penalizar especialmente los errores más grandes. RMSE siempre tendrá un valor igual o mayor que MAE, siendo mayor la diferencia cuanto mayor sea la varianza entre los errores.

Ambas medidas (MAE y RMSE) consideran que todos los errores tienen el mismo peso, lo cual puede no ser válido para todos los casos de aplicación (i.e. en un sistema de recomendación de noticias es lógico pensar tendría más importancia un error en noticias de reciente aparición). Para abordar este problema Cleger-Tamayo et al proponen una extensión de MAE (también aplicable a RMSE), la cual permite otorgar diferente importancia a cada error: **weighted Mean Absolute Error (wMAE)** [33] cuya formulación se puede ver en la Ecuación 23:

$$wMAE = \frac{\sum_{i=1}^U \sum_{j=1}^{N_i} w_{i,j} * abs(p_{i,j} - r_{i,j})}{\sum_{i=1}^U \sum_{j=1}^{N_i} w_{i,j}} \quad \text{Ecuación 23}$$

donde U representa el número de usuarios; N_i , el número de predicciones para el usuario $i^{\text{ésimo}}$; $r_{i,j}$, la puntuación establecida por el usuario $i^{\text{ésimo}}$ sobre el ítem I_j ; $p_{i,j}$, la puntuación predicha por el algoritmo y $w_{i,j}$ representa el peso asociado a cada predicción. Destacar que cuando todos los pesos $w_{i,j}$ son iguales, la fórmula coincide con la fórmula de MAE (ver Ecuación 21).

6.1.2. Medidas de Precisión

La utilización únicamente de métricas basadas en exactitud ha sido ampliamente discutida. Hay el consenso en la comunidad investigadora de que estas métricas por sí solas no son suficientes para capturar la efectividad de un recomendador [6]. En [93], McNee et al plantean el problema de utilizar medidas basadas en errores, como MAE o RMSE, a la hora de evaluar SR. Defienden que estas medidas favorecen a aquellos sistemas que recomiendan contenidos muy similares a aquellos que ya han sido consultados por el usuario, en lugar de aquellos sistemas que ofrezcan nuevos contenidos relevantes para el usuario. Por ello plantean el uso de **medidas de precisión**, para medir la capacidad de un sistema para, dado un conjunto de ítems relevantes para un usuario, ordenarlos por importancia. En el 2009, Gunawardana y Shani han presentado un estudio sobre métricas basadas precisión, para su utilización en SR [51][53]. Las principales medidas de precisión utilizadas son: **Precisión**, **Recall** y **Medida F₁**:

La **Precisión** se define como la fracción de ítems relevantes del total del conjunto de ítems devueltos por el sistema; formalmente se define como:

$$precisión = \frac{|I \in \{\text{ítems relevantes}\}|}{|I|} \quad \text{Ecuación 24}$$

siendo I el conjunto de ítems recomendados por el sistema e $I \in \{\text{ítems relevantes}\}$ el conjunto de ítems que son considerados relevantes en los juicios de relevancia.

El **Recall**, o **cobertura**, se entiende como la fracción de los ítems relevantes recomendados por el sistema del total de ítems relevantes, formalmente se define como:

$$recall = \frac{|I \in \{\text{ítems relevantes}\}|}{|\{\text{ítems relevantes}\}|} \quad \text{Ecuación 25}$$

Por último, la **Medida F₁** trata de combinar la precisión y el *recall* en una medida única y se define formalmente como:

$$F_1 = 2 * \frac{precisión * recall}{precisión + recall} \quad \text{Ecuación 26}$$

Sobre estas medidas, Powers defiende que no siempre funcionan bien en el manejo de ejemplos negativos, ya que sólo tienen en cuenta los ejemplos positivos [110]. En este trabajo se definen 3 nuevas medidas, *markedness*, *informedness* y *correlación de Matthews* que extienden a la precisión, *recall* y medida F respectivamente.

Markedness combina la precisión y la precisión inversa de acuerdo a la siguiente fórmula:

$$\text{markedness} = \frac{|I \in \{\text{ítems relevantes}\}|}{|I|} + \frac{|\bar{I} \in \{\text{ítems no relevantes}\}|}{|\bar{I}|} \quad \text{Ecuación 27}$$

siendo I el conjunto de ítems recomendados por el sistema, \bar{I} el conjunto de ítems no recomendados e $\bar{I} \in \{\text{ítems no relevantes}\}$ el conjunto de ítems no relevantes de entre aquellos que no han sido recomendados (ver Figura 16).

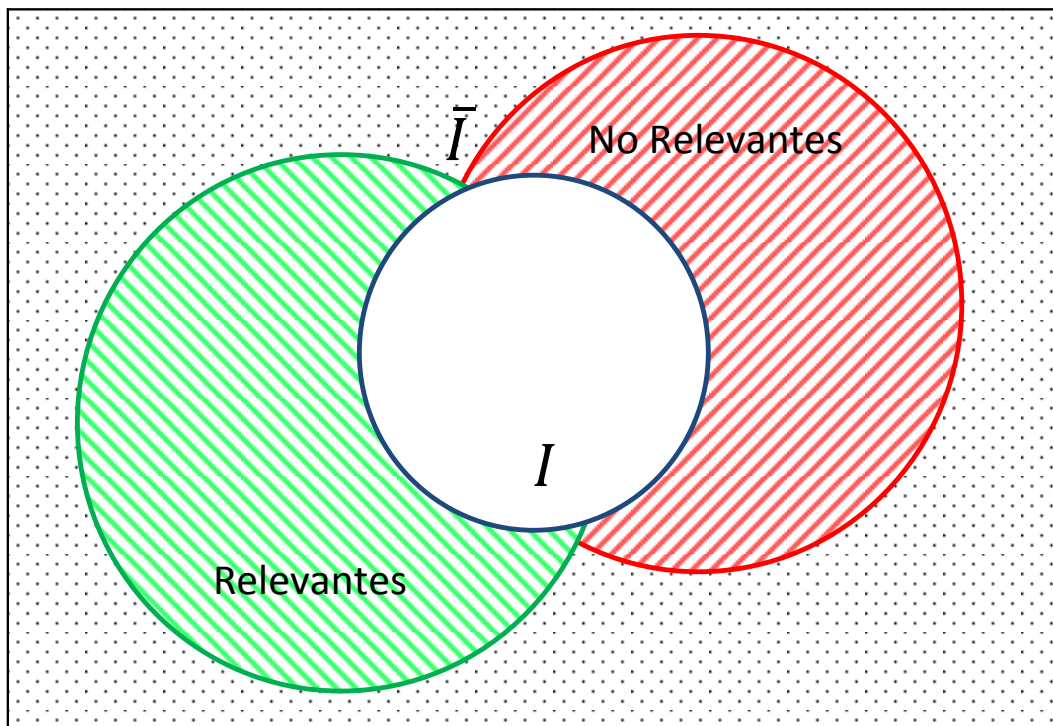


Figura 16 - Explicación Gráfica Markedness

Informedness se define como la combinación de *recall* y *recall* inverso de acuerdo a la siguiente fórmula:

$$\text{informedness} = \frac{|I \in \{\text{ítems relevantes}\}|}{|\{\text{ítems relevantes}\}|} + \frac{|\bar{I} \in \{\text{ítems no relevantes}\}|}{|\{\text{ítems no relevantes}\}|} \quad \text{Ecuación 28}$$

La **correlación de Matthews** combina las medidas de *informedness* y *markedness* de acuerdo a la siguiente fórmula:

$$\text{correlación de Matthews} = \sqrt{\text{informedness} * \text{markedness}} \quad \text{Ecuación 29}$$

6.1.3. Medidas de Satisfacción

Otras medidas de evaluación han sido propuestas para tratar de capturar otros criterios de **satisfacción**. Bellogín et al en [15] plantean el problema de utilizar únicamente medidas basadas en la precisión a la hora de evaluar SR. En su lugar abogan por la utilización de métricas que tengan también en cuenta la cobertura. De esta manera pretenden cubrir las limitaciones de las métricas basadas en precisión a la hora de tener en cuenta a usuarios de los cuales no se tiene mucha información. Esta idea es probada sobre varios tipos de recomendadores, presentes en el estado del arte, mejorando los resultados con la combinación de métricas (precisión y *recall*) que con métricas únicamente basadas en precisión.

Por otra parte, Adomavicius y Zhang presentan una medida llamada **estabilidad** [3], la cual refleja la consistencia de las recomendaciones de un sistema ante la llegada de nuevas puntuaciones que concuerdan con las estimaciones previas del sistema. Según se postula en este trabajo, la estabilidad de un sistema está estrechamente relacionada con la confianza y el grado de aceptación del sistema por parte de los usuarios. La estabilidad no necesariamente correla con la exactitud, pudiéndose dar el caso de que un sistema que sea perfectamente estable tenga muy poco rendimiento en términos de exactitud. Los propios Adomavicius y Zhang presentan en [4] una aproximación para mejorar la estabilidad de un sistema sin sacrificar su rendimiento en términos de exactitud. Relacionado con la estabilidad está la **robustez**, o capacidad de un sistema a no verse influenciado por valoraciones maliciosas, las cuales tratan de modificar el correcto funcionamiento del sistema y producir algún tipo de sesgo en los resultados [97].

Las tres medidas presentadas (estabilidad, satisfacción y robustez) anteriormente están enfocadas a medir la satisfacción del usuario relacionada con el rendimiento del sistema. La satisfacción de un usuario puede estar relacionada también con la usabilidad del sistema, y no solo con el rendimiento del mismo. A este respecto, en [60], Herlocker et al hacen una revisión de las métricas clásicas y defienden que los SR deberían tener en cuenta otras métricas que tengan en cuenta la usabilidad del sistema y no sólo su precisión. Para ello proponen varias métricas como: la **escalabilidad**, capacidad de un sistema para mantener su rendimiento a medida que crecen los datos; la capacidad de un sistema de **explicar** el motivo de las recomendaciones; o la **cobertura**, porcentaje de los contenidos que es recomendado.

6.1.4. Medidas de Diversidad y Novedad

Relacionada con la cobertura, está la **diversidad** de los objetos recomendados, que recoge lo diferentes que son entre sí estos objetos. Varios trabajos estudian cómo medir la diversidad: Fleder y Hosanagar estudian cómo medir este aspecto mediante el impacto en la diversidad de las ventas en un sistema de recomendación de uso comercial [47]; mientras que Lathia et al [81] tienen en cuenta un aspecto diferente a la hora de medir la diversidad, la diversidad temporal, para medir la habilidad de un sistema para evolucionar a lo largo del tiempo y adaptarse a los cambios.

Ziegler et al [150] proponen una medida para evaluar la diversidad en una lista de recomendaciones, similar *Maximal Marginal Relevance*: **Intra-List Similarity (ILS)**, definida en la Ecuación 30

$$ILS(R) = \frac{1}{2} \sum_{i \in R} \sum_{j \in R} sim(i, j) \quad \text{Ecuación 30}$$

siendo $sim(i, j)$ la función de similitud entre dos ítems en la lista de ítems recomendados. Destacar que esta métrica tiene la limitación de no tener en cuenta el orden de los ítems recomendados; factor que tiene mucha importancia en la utilidad del recomendador.

Vargas y Castells en [132] presentan un *framework* para la definición de métricas que unifiquen técnicas presentadas en el estado del arte en términos de diversidad y también de **novedad**. Este *framework* está fundamentado en tres relaciones entre usuarios y contenidos:

- **Descubrimiento:** Comprueba si un contenido era conocido, o no, previamente por un usuario
- **Elección:** Comprueba si un contenido ha sido “consumido” (usado, seleccionado, comprado, leído...), o no, previamente por un usuario.
- **Relevancia:** Comprueba si un contenido es considerado interesante por un usuario.

Siendo la probabilidad de elección de un contenido equivalente a la probabilidad conjunta de que haya sido descubierto y de que haya sido considerado relevante:

$$p(\text{elección}) \sim p(\text{descubierto}) * p(\text{rel}) \quad \text{Ecuación 31}$$

Basándose en estas relaciones, plantean una métrica genérica (Ecuación 32) que tenga en cuenta la diversidad y la novedad de un *ranking* de contenidos recomendados:

$$m(R|\theta) = C \sum_{i \in R} p(\text{elección}|i, u, R) nov(i|\theta) \quad \text{Ecuación 32}$$

donde $p(\text{elección}|i, u, R)$ es la probabilidad de que un usuario u elija un contenido i contenido en un *ranking* R , $nov(i|\theta)$ establece lo novedoso que es un contenido i , C es una constante de normalización y θ es una variable contextual genérica que permite considerar diferentes perspectivas a la hora de definir la novedad y la diversidad como: novedad basada en la popularidad de los contenidos o novedad basada en la distancia entre contenidos.

La ya citada novedad es otro de los conceptos importantes a la hora de evaluar el funcionamiento de un SR. Herlocker et al en [60] estudian también la originalidad de las recomendaciones de un sistema y proponen la utilización de medidas para capturar aspectos como la **novedad** y la **serendipia**. Ambos aspectos tratan sobre el concepto de la originalidad de una recomendación pero la serendipia hace referencia a un concepto más amplio. En [60] se ejemplifica la serendipia de la siguiente manera:

Considérese un sistema de recomendación que siempre recomiende a un usuario películas que han sido dirigidas por su director favorito. Si la película recomendada no ha sido vista o era desconocida para el usuario la recomendación sería novedosa, pero no tendría serendipia ya que el usuario tarde o temprano la podría haber descubierto por sí mismo. Un sistema que tuviese en cuenta la serendipia recomendará películas de nuevos directores, completamente desconocidas por el usuario.

Otro trabajo que estudia la utilización de medidas para establecer la serendipia es el de Murakami et al [98], donde presentan la “*unexpectedness*” para medir la diferencia entre la lista de recomendaciones generada por el sistema y una lista generada con los ítems habitualmente seleccionados por el usuario. Algunos aspectos a medir de estos conceptos propuestos son: ¿Cómo de satisfecho está el usuario con las nuevas recomendaciones? (observando, por ejemplo, el número de veces que el usuario vuelve a consultar esa recomendación) o ¿Cómo las recomendaciones se adaptan a los intereses del usuario a lo largo del tiempo?

Otros trabajos destacables que estudian cómo medir la novedad de las recomendaciones de un sistema son: el trabajo de Celma y Herrera[26], donde proponen un sistema híbrido para descubrir ítems novedosos; el trabajo de Shani y Chickering [125] o el trabajo de Ziegler et al [150]. Estos trabajos utilizan la popularidad de los contenidos para establecer como menos novedosos a aquellos contenidos que son más populares.

También para evaluar la novedad de una lista de ítems recomendados basándose en su popularidad, Zhou et al definen en [149] una métrica bautizada como **Mean Self-Information (MSI)** que mide la media de lo novedoso que es un ranking en base a los ítems que lo conforman:

$$MSI(R) = \frac{1}{|R|} \sum_{i \in R} \log \frac{|U|}{|\{u \in U \mid i \in u\}|} \quad \text{Ecuación 33}$$

donde R es el ranking de recomendaciones y U es el conjunto de usuarios del sistema, u es cada uno de los usuarios del conjunto U e i es un ítem perteneciente al conjunto de ítems visitados por el usuario u .

Por último citar el trabajo de Bellogín et al [14] en donde proponen sus propias métricas para evaluar la **diversidad** y la **novedad**, basadas en conceptos provenientes de la Teoría de la Información, como entropía o información mutua. Dado un usuario u , la diversidad de una lista de recomendaciones se define como:

$$div(R'_u) = - \sum_{i \in R'_u} p_{u,i} \log p_{u,i} \quad \text{Ecuación 34}$$

donde:

$$p_{u,i} = \frac{|\{R'_u \in R_u \mid i \in R'_u\}|}{|R_u|} \quad \text{Ecuación 35}$$

siendo R'_u una lista de recomendaciones perteneciente a un conjunto de recomendaciones R_u . para el usuario u .

La métrica para evaluar la novedad en una lista de recomendaciones propuesta en [14] se define como:

$$nov(R_u) = - \sum_{i \in R_u} p_i \log p_i \quad \text{Ecuación 36}$$

donde:

$$p_i = \frac{|\{u' \in U \mid i \in R_{u'}\}|}{|U|} \quad \text{Ecuación 37}$$

6.1.5. Medidas de Evaluación Online

Todas las medidas presentadas hasta este punto son medidas utilizadas o planteadas en la investigación académica y suelen estar relacionadas con **la evaluación offline** de los sistemas, en donde se utiliza un conjunto de datos de usuarios previamente coleccionado. Por el contrario, en el ámbito de los sistemas comerciales el tipo de evaluación que prima es la **evaluación online**, donde se evalúa el funcionamiento con datos de usuarios en tiempo real y, por tanto, el tipo de medidas de evaluación a utilizar es distinto. La distancia (*gap*) entre estos dos ámbitos es uno de los temas que actualmente más interés suscita [6]. Alguna de las medidas más comúnmente utilizadas por los SR comercial son: CTR (*click through rate*) [5] definida como el ratio entre las veces que se muestra una recomendación y las veces que el usuario accede a ella; ratio de conversión [131] definida como, dado contexto de recomendación de comunidades, la relación entre dos comunidades recomendadas y el contenidos de los registros de usuarios que pertenecen a esas comunidades; o el incremento de beneficios [16] al utilizar el recomendador de contenidos.

Otra medida centrada en la evaluación *online* es la **confianza** del usuario en las recomendaciones de un sistema [92] [36]. La confianza como medida de evaluación ha sido estudiada desde el comienzo del desarrollo de los SR, ya en 1998 Basu et al estudian cómo medir la confianza de los usuarios en su aproximación de recomendación basada en aspectos sociales [13]. Según Ricci et al [114], la confianza puede ser estudiada desde dos puntos de vista, la confianza en los otros usuarios del sistema de recomendación y la confianza en el funcionamiento del sistema de recomendación.

6.1.6. Comentarios finales

En la reflexión del capítulo anterior se expuso la importancia de las métricas de similitud para el funcionamiento de un sistema de recomendación a pesar de no ser la parte fundamental del mismo. Un caso similar ocurre con las medidas de evaluación, aunque estas medidas ni siquiera forman parte del sistema como tal son de una importancia capital a la hora de evaluar el rendimiento de un sistema, ya que, como defienden Gunawardana y Shani en [51], “*encontrar una medida de evaluación apropiada es a menudo crítica, ya que cada métrica puede favorecer a un algoritmo diferente*”. Este aspecto es tratado también por Schröder et al en [124], donde exponen que raramente se justifica la elección de una medida frente a otra; por ello, proponen una guía para ayudar a los autores a escoger una medida de evaluación en función del criterio que se desee tener en cuenta.

Las medidas más novedosas como la confianza, escalabilidad, estabilidad/robustez, novedad/serendipia o medidas de evaluación *online* son transversales al tipo de recomendador que se desee evaluar; siendo especialmente recomendables para nuevos tipos de SR, como son los basados en información social. Este tipo de medidas también pueden utilizarse para complementar las medidas clásicas (basadas en precisión y en error) en la evaluación de las aproximaciones de recomendación clásicas (Filtro colaborativo y Basadas en contenido).

Respecto a las medidas de evaluación clásicas, su utilización está claramente diferenciada. Las medidas basadas en error necesita de las puntuaciones numéricas de los usuarios sobre los contenidos para poder calcular el error cometido; por ello están estrechamente relacionadas con los recomendadores basados en filtro colaborativo, que suelen hacer uso de este tipo de puntuaciones. Dentro de estas medidas la utilización de RMSE frente a MAE es preferible cuando se quieran penalizar los errores más grandes frente a los de menor magnitud, como ya se ha expuesto con anterioridad.

Aunque en las aproximaciones basadas en contenido pueden aparecer puntuaciones, el uso de medidas basadas en error es menos recomendable que la utilización de medidas basadas en precisión; las medidas basadas en precisión proceden del campo de la recuperación de información y, puesto que las aproximaciones basadas en contenido se pueden ver como una aproximación de Recuperación de Información, parece más recomendable utilizar este tipo de medidas d esta área.

Este tipo de medidas son especialmente útiles para establecer la validez de una lista de resultados, que es la manera más común en que los SR basados en contenido ofrecen sus recomendaciones. Por el contrario las medidas basadas en error se centran en evaluar cada ítem de manera individual. Este tipo de medidas es especialmente recomendable también cuando, en lugar de disponer de puntuaciones, se disponga de información binaria (relevante/no relevante).

Por otra parte la utilización de medidas basadas en error favorece a sistemas que recomienden contenidos similares a los ya vistos por el usuario frente a aquellos que recomienden contenidos originales. Por ello se pueden ver las medidas basadas en error y las medidas de novedad y serendipia como opuestas. En función de en qué se quiera centrar el rendimiento del sistema (minimizar errores o maximizar recomendaciones originales) se deberá elegir entre una u otra.

Atención aparte requieren las medidas de evaluación online. Este tipo de medidas son aplicables a un contexto de ejecución muy determinado. Aunque este tipo de métricas han estado asociadas a los SR comerciales en producción comienza a surgir el interés por su estudio en el campo de la investigación. De nuevo aparece el problema de la dificultad de tener acceso a este tipo de datos en tiempo real. A este respecto, citar el CAMRa 2010⁴⁰ como el intento más destacable para investigación en SR con datos en tiempo real.

Más allá de las medidas, un aspecto clave a la hora de evaluar SR es la utilización de **datos de usuarios reales**. Ziegler et al [150] plantean la importancia de tener a usuarios reales para evaluar los sistemas en aspectos tales como la originalidad de las recomendaciones, ya que los datos generados de manera artificial no capturan fidedignamente todos los aspectos del comportamiento de usuarios reales (como, por ejemplo, los cambios de preferencias a lo largo del tiempo).

6.2. Evaluación centrada en el usuario

Mientras que el tipo de evaluación presentada anteriormente se basa en medir el funcionamiento del sistema, la evaluación centrada en el usuario se basa en medir la satisfacción del usuario en la utilización del sistema. En el área de los SR es especialmente importante este tipo de evaluación, ya que el éxito final del sistema no se basa en el buen rendimiento de este en términos de métricas sino en qué finalmente el usuario acceda a los contenidos recomendados. Tal y como señalan McNee et al en [93], en ocasiones la satisfacción del usuario no guarda correlación con una medida de rendimiento (precisión) muy alta.

⁴⁰ <http://www.dai-labor.de/camra2010/>

Por ello el interés en este tipo de evaluaciones cada vez es mayor en el ámbito de los SR, como demuestra la aparición de *Workshops* específicos del tema como el *Workshop on User-Centric Evaluation of Recommender Systems and Their Interfaces* (UCERSTI) [143].

La metodología para este tipo de evaluación es muy variable, dependiendo de qué características se quieran evaluar o a qué tipos de usuarios está enfocado el sistema (usuario profesional, usuario no experto,...). En [72] y [73], Knijnenburg et al presentan un *framework* para la evaluación centrada en el usuario que tiene en cuenta estos aspectos y que puede usarse como guía para futuras evaluaciones.

En [111] Pu y Chen recopilan una serie de trabajos que recogen evaluaciones enfocadas en los usuarios de diferentes SR y también proponen un modelo (*ResQue*) basado en un cuestionario para evaluar la satisfacción de los usuarios en el uso de SR, donde se puede encontrar información ampliada al respecto.

Algunas de las principales características relacionadas con la satisfacción del usuario como son: la velocidad de respuesta del sistema o su facilidad de uso (**utilidad**) [54]. Para evaluar estas características, entre otras, así como la satisfacción del usuario, los métodos más utilizados son:

- Utilización de cuestionarios para medir cualitativamente la satisfacción del usuario. A la hora de utilizar cuestionarios es importante ser neutral al definir las preguntas, de tal manera que la propia pregunta no sugiera una respuesta considerada “correcta” o “deseable” por el evaluador. Un trabajo comúnmente referido en la literatura de la creación de cuestionarios es el de Pfleeger y Kitchenham [107], en el cual se hace un repaso de los principios a seguir en este tipo de evaluaciones.
- Realización de tareas por parte del usuario, utilizando el sistema, para comprobar el rendimiento y la usabilidad del mismo.
- Comparación del sistema con otros sistemas similares mediante la definición de una tarea que el usuario tiene que completar con ambos sistemas.

Algunas de las medidas de evaluación utilizadas en este tipo de evaluación son:

- Tiempo que el usuario ha tardado en completar una tarea
- Número de *clicks* realizados por el usuario
- Número de páginas visitadas por el usuario
- Tiempo que el usuario ha estado en cada página
- Número de documentos relevantes encontrados por el usuario.

A pesar de que este tipo de información ofrece información más real acerca de la satisfacción del usuario con el sistema, presenta una serie de desventajas que ha hecho que su utilización no esté todo lo extendida que sería deseable [126]. La principal de estas desventajas es que esta evaluación es muy costosa de llevar a cabo debido a la necesidad de emplear a un número importante de usuarios. Otro aspecto a tener en cuenta es que los usuarios con los que llevar a cabo esta evaluación deben ser tan parecidos como sea posible a los usuarios finales del sistema, algo que no siempre es posible de encontrar.

6.3. Comentarios Finales

En el área de Sistemas de Recomendación, al igual que ocurre en otras áreas relacionadas con la Recuperación de Información y el Procesamiento del Lenguaje Natural, la evaluación representa una de las fases más importantes y complejas a la hora de desarrollar un sistema. Mediante la evaluación se busca tener una medida objetiva del rendimiento del sistema y tener un punto de comparación con otros sistemas similares.

Focalizándose en el área de SR, las técnicas de evaluación se dividen en evaluaciones experimentales o de laboratorio y evaluaciones utilizando las opiniones de usuarios. Las evaluaciones experimentales necesitan de la existencia de colecciones de prueba que posean juicios de relevancia, los cuales permitirán evaluar si los contenidos recomendados por un sistema se adaptan a lo que usuarios reales han marcado como relevante. Si bien para los SR basados en filtro colaborativo se han desarrollado un número significativo de colecciones de prueba, para los SR basados en contenido es complejo encontrar colecciones adecuadas. Por otra parte, la evaluación centrada en usuario a su vez es dificultosa ya que se necesita de la participación activa de un conjunto de usuarios. A pesar de esto este último tipo de evaluación cada vez está cobrando más importancia en el área de SR. No es de extrañar ya que, al fin y al cabo, los SR buscan ofrecer contenidos que satisfagan a los usuarios; por lo que, qué mejor que sean los propios usuarios los que puntúen el sistema.

Debido al problema de acceso a colecciones de prueba y a la cada vez más relevancia de la evaluación centrada en el usuario, la evaluación de SR basados en contenido mediante este último tipo se hace más recomendable y suscita cada vez más interés

Parte 2. Propuesta de trabajo

El sistema de recomendación planteado utiliza el contenido textual de los objetos susceptibles de ser recomendados. Más concretamente la utilización del texto de las noticias permite la aplicación de técnicas de procesamiento de lenguaje natural que permiten mejorar el conocimiento que se tiene sobre el contenido en cuestión. Por otro lado también ofrece la posibilidad de utilizar fuentes externas para añadir contenido para tener más información.

Mientras que los sistemas basados en filtro colaborativo presentan el problema del *cold start*, los sistemas basados en contenido permiten abordar este problema al utilizar el propio contenido de los objetos, sin necesitar de interacciones de usuarios previas sobre estos. Sin embargo, los SR basados en contenido también presentan desventajas. La más importante es el problema de la sobre especialización. Este problema se refiere a la recomendación de objetos muy similares a los ya consultados por el usuario. Pese a que esta recomendación no es errónea, (los contenidos son del interés del usuario) no es muy interesante para los usuarios, ya que estos objetos ya son conocidos sin necesidad del SR.

Para abordar el problema de la sobre especialización en este trabajo se propone la utilización de técnicas de divergencias del lenguaje para modelar tanto los contenidos susceptibles de ser recomendados como los perfiles de los usuarios. Estas técnicas, dado un conjunto de textos, extraen aquella terminología que mejor diferencia a un texto respecto al resto. Mediante la utilización de estas técnicas se pretende representar tanto a los contenidos como al perfil de los usuarios con aquellos términos que mejor les diferencian del resto. Con ello se espera que las recomendaciones no se centren únicamente en contenidos muy similares a los ya consultados por el usuario, sino también en otros contenidos que puedan resultar más originales, que también puedan ser interesantes para el usuario.

En el trabajo se estudia el uso de diferentes aproximaciones de recomendación. Estas aproximaciones tratan de cubrir los escenarios más frecuentes en los que la utilización de un SR puede ser de utilidad. Para llevar a cabo las recomendaciones se utilizará una aproximación de Recuperación de Información, de cara a encontrar nuevos contenidos relevantes para el usuario. El funcionamiento del SR se enmarca en el contexto de un sistema de búsqueda de contenidos, en el que el usuario, además de poder buscar contenidos de su interés, puede disfrutar de las recomendaciones ofrecidas por el sistema.

Para la experimentación se ha desarrollado un prototipo de búsqueda de contenidos, en el cual se integra el sistema de recomendación de contenidos. El prototipo le ofrece al usuario una interfaz de búsqueda, mediante la cual puede buscar a través de los recursos disponibles mientras que el sistema registra su actividad, generando un perfil de usuario. Este perfil será modelado utilizando la aproximación descrita y permitirá al SR ofrecer contenidos al usuario mientras interactúa con el sistema.

A lo largo de los siguientes capítulos se detalla la propuesta planteada en este trabajo, la experimentación llevada a cabo para comprobar su validez y por último la evaluación de

dicha experimentación, tanto para la aproximación de modelado como para las aproximaciones de recomendación.

Capítulo 7 - Descripción de la Propuesta

En este trabajo se presenta una aproximación de modelado basada en divergencias del lenguaje para su posterior utilización en un SR basado en contenidos. Para el desarrollo de dicha propuesta es necesaria una colección de pruebas, un método de representación de dicha colección, el modelado de los usuarios que hagan uso del sistema y una serie de aproximaciones de recomendación que utilicen estos recursos. Todo ello es detallado a lo largo de este capítulo.

7.1. Modelado de los contenidos

En la parte correspondiente al estado del arte se han planteado diferentes técnicas de representación textual, clasificadas en técnicas probabilísticas y no probabilísticas. Los modelos probabilísticos (PLSI, LDA) no se pueden adaptar fácilmente a nuestro trabajo, ya que debido al volumen de los datos (gran cantidad de términos asociados a cada documento y un vocabulario de la colección muy extenso) la representación sería muy difícil de construir, dado que la complejidad del modelo crece linealmente con el tamaño del corpus. Debido a ello, se ha decidido optar por una técnica no probabilística como **VSM**, ampliamente utilizada para representar información según el estado del arte de diferentes áreas. Sería interesante sin embargo poder combinar la sencillez de técnicas como VSM con las ventajas que ofrecen la utilización de técnicas probabilísticas.

VSM representa un contenido textual como un vector de términos con pesos asociados. Los pesos de los términos pueden ser establecidos de acuerdo a diferentes funciones de pesado. Dentro de estas funciones de pesado, TF-IDF es considerada casi un estándar *de facto* y ha sido aplicado en numerosas ocasiones en el campo de los SR basados en contenidos. Por otro lado cabe resaltar el problema de la **sobre-especialización** de las recomendaciones de los sistemas de este tipo.

7.1.1. Aproximación de modelado

Para abordar la sobre-especialización de las recomendaciones, así como para tratar de cubrir las limitaciones del uso de una técnica no probabilística como VSM, en este trabajo se ha optado por utilizar una técnica probabilística clásica de divergencia del lenguaje, como KLD, frente a TF-IDF. KLD se utiliza para extraer los términos más relevantes de un documento textual dentro de una colección. Se considera que un término es relevante respecto a otros para un documento dado si aparece frecuentemente en el documento y aparece raramente o no aparece en el resto de documentos de la colección. Para calcular la relevancia de cada término se usa KLD mostrada a continuación:

$$KLD_{pD,pC} = pD(t) \cdot \ln\left(\frac{pD(t)}{pC(t)}\right) \quad \text{Ecuación 38}$$

donde $pD(t)$ es la probabilidad de cada término t dentro del documento D (o lo que es lo mismo la frecuencia de t dividido por el total de términos del documento) y $pC(t)$ es la probabilidad del mismo término t en toda la colección C (frecuencia de t dividido por el total de términos en la colección). Aplicando la Ecuación 38 a un contenido textual es posible ordenar los términos en un ranking de acuerdo al peso otorgado a cada uno de ellos mediante KLD.

Mediante KLD se tiene a los contenidos representados por los términos que les hacen diferentes del resto de documentos de la colección. De esta manera se pretende capturar aquellos aspectos más originales de los contenidos que son de interés para el usuario. Con ello se espera poder encontrar otros contenidos, los cuales compartan estos aspectos originales, que puedan ser novedosos para el usuario.

Las ventajas de la utilización de KLD frente a TF-IDF han sido expuestas con anterioridad: en [20] se prueba de manera experimental que la utilización de KLD es preferible a la de TF-IDF para la categorización de textos (una tarea muy similar a la de recomendación, tratada en este trabajo) debido a que TF-IDF, y más concretamente IDF, solo tiene en cuenta el número de documentos en los que un término aparece. Por el contrario, KLD tiene en cuenta el contenido en sí estos documentos en los que el término aparece y no solo el número de documentos en los que aparece. También en el trabajo presentado en [128] se compara una técnica basada en divergencias del lenguaje similar a KLD (divergencia de Jensen-Shannon) con TF-IDF para el modelado de usuarios, obteniendo mejores resultados en términos de representatividad.

Resumiendo, de acuerdo a la técnica de representación propuesta en este trabajo, los contenidos de la colección se representan mediante un vector de términos, ponderando cada uno de ellos mediante el peso que KLD le otorga.

7.1.2. Almacenamiento de los modelos

La aproximación de modelado, basada en KLD, se aplica sobre la colección elegida para experimentación (ver sección 7.4). En la colección elegida, los contenidos están clasificados en 3 niveles: noticias, subcategorías y categorías. De manera independiente a cada uno de los 3 niveles, la aproximación de modelado se aplica sobre cada uno de los documentos; esto es, para cada documento se modelan los términos más representativos respecto al resto de documentos: para el caso de las noticias se modela cada noticia de acuerdo al resto de noticias.; para el caso de subcategorías se genera un único documento conteniendo todo el texto de las noticias pertenecientes a dicha subcategorías y modela dicho documento único respecto al resto de documentos de las otras subcategorías; y para el caso de las categorías se procede de manera análoga al del caso de las subcategorías.

De esta manera cada uno de los documentos queda modelado por un ranking de términos, ordenado de acuerdo a la importancia establecida por KLD. Para permitir su posterior utilización el modelo de cada uno de los documentos (noticias, subcategorías y categorías) es indexado, utilizando para ello Solr.

El índice generado tendrá una entrada por cada documento y para cada entrada se tendrá un campo con el descriptor del documento y un campo que almacene los términos modelados. Sin embargo, no todos los términos tienen la misma validez (depende del peso dado por KLD a cada uno, cuyos valores están entre 0 y 1). Para reflejar la diferencia de importancia se establece un peso igual a 1 al término con menos peso del ranking y se normalizan los demás peso de acuerdo a esto. Por ejemplo, dado el modelo:

Término	Peso
Término_1	0.4
Término_2	0.2
Término_3	0.1

La normalización de los pesos dará lugar al modelo:

Término	Peso
Término_1	4
Término_2	2
Término_3	1

Tras llevar a cabo esta normalización, en el índice se almacenará cada término el número de veces que indique el peso del modelo normalizado. Para el modelado anterior, la entrada del índice sería

IDENTIFICADOR: id_documento

KLD: Término_1 Término_1 Término_1 Término_1 Término_2 Término_2
Término_3

Aplicando este método de indexación a los diferentes niveles en los que están divididos los contenidos se obtienen los siguientes índices:

- **INDEX_DOC:** Este índice contiene una entrada por cada una de las 5360 noticias que conforman la colección. Para cada noticia se tiene el identificador de la misma junto con el modelo generado.
- **INDEX_SUBCAT:** Este índice contiene una entrada por cada una de las 182 subcategorías en la que están clasificados los contenidos de la colección. Para cada subcategoría se tiene el nombre de esta como identificador junto con el modelo generado.

- **INDEX_CAT:** Este índice contiene una entrada por cada una de las 19 categorías en las que se encuentran clasificados los contenidos. Para cada una de las categorías se tiene el nombre de la misma como identificador junto con el modelo generado.

7.2. Modelado de los usuarios

Para modelar la actividad previa de los usuarios se sigue una aproximación similar a la del modelado de los contenidos. La actividad previa de los usuarios viene definida por aquellos contenidos que previamente ha consultado; y más concretamente, por la información textual de dichos contenidos, o lo que es lo mismo una recopilación de los términos que aparecen en los contenidos que haya consultado.

Sin embargo, se puede dar el caso que un usuario haya accedido a un contenido pero no sea de interés para él, o que la relevancia que tienen para él los contenidos a los que accede no sea la misma. Por ello, el sistema desarrollado permite al usuario que establezca la importancia de los contenidos mediante una puntuación de 1 a 5, siendo 1 poco/nada relevante y 5 muy relevante. En caso de que el usuario no marque ninguna puntuación se considerará el contenido como no relevante (puntuación igual a 1).

Con todo esto en cuenta, se entiende el perfil de un usuario como la recopilación términos que forman parte de los contenidos visitados previamente por el usuario. Para cada término se tiene en cuenta la frecuencia de aparición en cada contenido y se multiplica esta frecuencia por la puntuación establecida por el usuario sobre dicho contenido. De esta manera se pretende tener en cuenta la relevancia del contenido para el usuario.

Dados estos perfiles, se plantea una aproximación de modelado que hará también uso de VSM y KLD para modelar los contenidos presentes en el perfil de usuario. Para ello, se tomará cada término del perfil de un usuario y se ponderará mediante KLD, teniendo en cuenta la probabilidad de que aparezca en los contenidos previamente consultados por el usuario (perfil del usuario) frente al resto de documentos que conforman la colección.

La aproximación es similar a la presentada por Shmueli-Scheuer et al en [128]; en este trabajo utilizan una técnica basada en divergencias del lenguaje para modelar la actividad previa de los usuarios. La diferencia radica en que, en lugar de comparar los perfiles de usuario contra el resto de documentos de la colección como se propone en este trabajo, en [128] se compara cada perfil de un usuario contra resto de perfiles de usuario. Esto hace que el modelo de un usuario pueda verse modificado por el hecho de que otro usuario, sin relación con él, consulte nuevos contenidos. Un ejemplo de esto se presenta a continuación:

Dado un usuario objetivo (usuario A), y otro usuario (usuario B) que consulta nuevos contenidos y por tanto modifica su perfil; entendiéndose por modificación que nuevos términos se incluirán en el perfil. Puesto modelar al usuario A se compara su perfil con el resto de perfiles, entre ellos el del usuario B; la modificación producida en el perfil del usuario B provocará que el modelo del usuario A varíe. Esto se produciría a pesar de que el usuario A no haya realizado ninguna nueva interacción ni sus preferencias hayan cambiado.

Que el modelo de un usuario, y extensión las recomendaciones, se modifiquen por la actividad de otros usuarios no es una característica muy deseable. Con nuestra aproximación, el perfil de cada usuario se compara con el resto de documentos de la colección, por lo que la acción de otros usuarios no producirá ninguna modificación en los modelos del resto.

7.3. Aproximaciones de recomendación

En este trabajo se presentan varios enfoques de recomendación. Los diferentes enfoques propuestos tratan de cubrir las diferentes situaciones en las que un SR puede ser útil para un usuario. Estos enfoques se basan en una aproximación de Recuperación de Información para buscar aquellos contenidos que van a ser recomendados a los usuarios.

Los diferentes enfoques planteados son: Recomendación utilizando únicamente los contenidos vistos por el usuario con anterioridad, Recomendación utilizando los contenidos consultados anteriormente y los nuevos que el usuario está consultando y Recomendación basada en un contenido específico. Todos estos enfoques son presentados a continuación.

7.3.1. Recomendación utilizando únicamente los contenidos vistos por el usuario con anterioridad

Este tipo de recomendación utiliza la información que ya se tiene del usuario, esto es, el contenido de las noticias que el usuario haya consultado previamente. En esta aproximación no se emplea ningún tipo de información adicional. Esta recomendación se lleva a cabo cuando un usuario accede al sistema, ya que no se tiene información actualizada de él, más allá de la disponible en su perfil. Con esto se le ofrece al usuario la posibilidad de consultar noticias que previsiblemente van a ser de su interés, en función de su comportamiento previo.

Un ejemplo de esta recomendación se muestra en la Figura 17. Como se puede ver, el sistema muestra, por un lado, una nube de etiquetas con los términos más representativos del perfil del usuario y por otro lado las noticias relacionadas con esta nube de etiquetas.

Preferencias del Usuario

milan robinho liga madrid

iker partido champions

real casillas max

Público.es - 2-0. El Real Madrid aprueba su primer examen difícil con buen juego ante el Milán
El Real Madrid vence al Milan con comodidad y encarrila el pase a octavos - 20minutos.es
El Bernabéu acoge el primer test serio con el Madrid-Milan, un duelo de gigantes - 20minutos.es
El Bernabéu acoge el primer test serio con el Madrid-Milan, un duelo de gigantes - 20minutos.es
Casillas: "Ronaldinho era intratable y temido en el Barça; ya no es el mismo" - 20minutos.es
Robinho: " Llego a Madrid enchufadisimo " - MARCA.com
FIFA.com - Casillas, el portero de más partidos
El Real Madrid vence al Milan con comodidad y encarrila el pase a octavos - 20minutos.es
La exaltación del colectivo en UEFA.COM
UEFA.com - UEFA Champions League 2011 - Real Madrid-Milan

Figura 17 – Recomendación utilizando el perfil de los usuarios

Dichas noticias se obtienen tomando el conjunto de términos del modelo y ejecutando con ellos una consulta contra el índice que almacena el modelado de la colección: INDEX_DOC (ver Sección 7.1.2). Esta consulta se forma concatenando cada uno de los términos del modelo. Para reflejar la importancia de cada término, éste se repetirá un número determinado de veces en la consulta; número que vendrá fijado por el peso que el modelado le haya otorgado: a mayor número de apariciones del término en la consulta más importancia tendrán los documentos en los cuales aparezca, de cara a ser devueltos por el sistema. Para ejemplificar, supóngase un modelo:

Término	Peso
Término_1	0.4
Término_2	0.2
Término_3	0.1

Para obtener noticias relevante para este modelado se ejecutará la consulta:

**Término_1 OR Término_1 OR Término_1 OR Término_1 OR Término_2 OR
Término_2 OR Término_3**

Resaltar que para concatenar los términos en la consultas se utiliza el operador lógico OR. Este operador buscará documentos en los que aparezca al menos un término de la consulta, teniendo más relevancia los documentos en los que más términos de la consulta aparezcan. Esto se ha decidido así ya que en caso de utilizar un operador AND la cobertura se reduciría drásticamente, pudiendo darse el supuesto de no ser devuelto ningún documento.

7.3.2. Recomendación utilizando los contenidos consultados anteriormente y los nuevos que el usuario está consultando

Este enfoque pretende aprovechar la información actualizada que el usuario proporciona en cada nueva interacción con el sistema. Dado un usuario que ha accedido al sistema y está consultando noticias, es de suponer que serán de interés para el usuario noticias relacionadas con las que está consultando en ese mismo momento. Basados en este enfoque se abordan dos supuestos, presentados a continuación.

7.3.2.1. Recomendación relacionada con una búsqueda de contenidos

Cuando un usuario realiza una búsqueda de noticias, el sistema recoge el contenido de la lista de resultados obtenidos y lo utiliza para complementar la información que se tiene en el perfil del usuario. De esta manera se intenta refinar el conocimiento que se tiene de un usuario, dada una necesidad de información especificada en la consulta. Un ejemplo práctico de este supuesto sería el siguiente:

Si un usuario interesado en noticias deportivas, y más concretamente noticias de fútbol, realiza una consulta MADRID probablemente esté interesado en noticias de fútbol que estén relacionadas con la ciudad de Madrid (Real Madrid, Atlético de Madrid, Partidos en Madrid, Eventos futbolísticos en Madrid, ...).

En la Figura 18 se muestra un ejemplo de este tipo de recomendación. Se puede ver a la izquierda de la imagen los resultados devueltos para la consulta *Madrid* y a la derecha las recomendaciones mostradas. La recomendación muestra una nube de términos recomendados y las noticias asociadas a esta nube. Se puede ver como para este usuario interesado en deportes, ante la consulta *Madrid*, los términos recomendados están mayoritariamente relacionados con el equipo de fútbol de la ciudad, el *Real Madrid: Mourinho*, su entrenador; *Bernabeu*, el estado en el que juega; o *Cristiano*, uno de sus jugadores. Esto es así a pesar de que entre los resultados devueltos aparezcan otras noticias como: declaraciones de la presidenta de la comunidad de Madrid o un desfile de moda producido en Madrid.

Para generar la nube de términos, se toman los términos del modelo cada una de las noticias devueltas y se agregan a un modelo común: **modelo_consulta_usuario**. Esta agregación se lleva a cabo tomando cada uno de los términos que aparecen en modelos individuales de cada noticia y añadiéndolos junto con su peso al **modelo_consulta_usuario**. Si algún término aparece repetido en varias noticias se suma el peso otorgado por cada uno de los modelos individuales de las noticias. Finalmente los términos mostrados en la nube son lo más representativos (los de más peso) en el **modelo_consulta_usuario**, reflejando el tamaño de los términos en la nube la relevancia de los mismos.

Búsqueda de Nuevas Noticias

madrid

El Milan ya está en Madrid - ABC.es ★★★★★
 El Milan ya ha aterrizado en Madrid con todas sus estrellas (Robinho, Ibra, Carvalho, Pedro León, Ozil, Khedira y Canales reconocen que han venido al Real Madrid

'Expediente X' Velickovic: " Honestamente, no sé por qué no he jugado " - MARCA.com ★★★★★
 del Real Madrid a Grecia para hablar sobre su situación marginal en la rotación de Ettore Altor Karanika, ex defensa central de Athletic de Bilbao y Real Madrid y en la actualidad miembro

La Fiscalía de Valencia denuncia a Cañizares por defraudar a Hacienda - ABC.es ★★★★★
 ex guardameta de Real Madrid y Valencia, por haber defraudado presuntamente a Hacienda

FIFA.com - Casillas: "Hemos pasado con nota alta" ★★★★★
 muy importante para los hombres de José Mourinho. Preguntado por el jugador del Real Madrid que destacaría para ver un poco cuál era la dimensión del Real Madrid y el examen se ha pasado con nota alta. Iker Casillas, portero del Real Madrid Por su parte, Gonzalo Higuain, delantero

FIFA.com - Regresan Albiol y Canales ★★★★★
 no ha jugado ningún partido oficial con el Real Madrid en lo que va de campaña. El joven delantero Sergio Canales, titular al inicio de la Liga en el Real Madrid (0-0 contra Mallorca).

Aguirre acusa al PSM de financiación ilegal - ABC.es ★★★★★
 La presidenta de la Comunidad de Madrid, ha acusado al Partido Socialista de Madrid les ha dado a ustedes Caja Madrid en el año 2007? 418. 764, más del doble». Por otra parte, que diga que el Gobierno de la Comunidad de Madrid ha financiado un acto del PP

Iker entra en el 'top ten' de la Champions - MARCA.com ★★★★★
 del Real Madrid ha superado a Luis Figo y a Kahn, que se retiraron con 103 partidos

Robinho: " Llego a Madrid enchufadísimo " - MARCA.com ★★★★★
 Superior de Diseño de Moda de Madrid (UPM), MADRID Prendas y accesorios para el hogar

Público.es - Aguirre se desmarca de la 'Gürtel' y señala otra vez a Escudero ★★★★★
 su etapa al frente del PP de Madrid de la a pesar de que los últimos informes policiales de su campaña electoral de 2003 los organizó la, ¿mantienen en el PP de Madrid del PP de Madrid respondió con otra cuestión: "En el año 2003, ¿quién era el responsable de las campañas

robinho	madrid	iker
cristiano	partido	real
seedorf	aguirre	olympiacos
oferta	higuain	casillas
mas	ave	milan
mourinho	ronaldo	reconoce
goles	galliani	equipo
estudio	club	bernabeu
portero		

Otras noticias relacionadas

Público.es - Aguirre presume de contratos transparentes
 FIFA.com - Beckham: "Lo mejor está por llegar"
 Casillas: «Hemos pasado el examen con nota alta»
 El Valencia amenaza el Camp Nou en UEFA.COM
 Público.es - La 'Gürtel' organizó todos los actos de Aguirre en la campaña de 2003
 Público.es - Aguirre se desmarca de la 'Gürtel' y señala otra vez a Escudero
 Robinho: " Llego a Madrid enchufadísimo " - MARCA.com
 UEFA.com - UEFA Champions League 2011 - Real Madrid-Milan
 Público.es - Granados se defiende: "No entiendo de contabilidad"
 La exaltación del colectivo en UEFA.COM

0 1 2 3 4 5 6 7 8 9


 Natural Language Processing and Information Retrieval Group at UNED nlp.uned.es

Figura 18 - Recomendación utilizando una búsqueda de contenidos

De nuevo, además de la nube de términos, se le muestra al usuario un conjunto de noticias relacionado con dicha nube. Para obtenerlas se sigue el mismo método que en el apartado previo (ver 7.3.1): se toman los términos de la nube como los consulta contra el índice que almacena la representación de la colección: INDEX_DOC.

7.3.2.2. Recomendación relacionada con un contenido específico

En este caso un usuario, tras haber realizado una búsqueda, ha accedido al contenido de una noticia en concreto. Con ello se entiende que el usuario desea ampliar información sobre dicha noticia, bien porque sea de su interés y quiere ampliar información o bien porque no tiene claro si es de su interés y quiere consultarla con más detalle. En ambos casos sería interesante ofrecerle al usuario información relacionada con la noticia. Para ello se toma la información asociada a la noticia y se lleva a cabo una recomendación a 3 niveles:

- *Recomendación utilizando únicamente el contenido de la noticia*
- *Recomendación utilizando el contenido de la subcategoría de la noticia*
- *Recomendación utilizando el contenido de la categoría de la noticia.*

En la Figura 19 se muestra la recomendación en los 3 niveles planteados. El usuario tras haber realizado una búsqueda ha accedido a una noticia que habla sobre un partido del *Real Madrid* contra el *Milan*. El contenido de la noticia se muestra a la izquierda de la imagen. El sistema entonces le muestra los modelos asociados a la noticia y a la subcategoría (*fútbol*) y categoría (*deporte*) de la misma, mostradas a la derecha de la imagen.

Mostrándole el modelo de la noticia se intenta ayudar al usuario a encontrar términos relevantes en la misma que le puedan ayudar a encontrar otras noticias relevantes. En el ejemplo mostrado en la Figura 19, el sistema le ofrece al usuario términos como: *Milan*, rival del *Real Madrid* en el partido del cual habla la noticia; *Seedorf*, jugador del *Milan*; *Ozil* o *Cristiano*, jugadores del Madrid; u *octavos*, fase del torneo en la que se ha producido el partido (octavos de final).

Por otra parte, al ampliar el rango de la recomendación a la subcategoría y la categoría de la noticia se espera ofrecer al usuario contenidos relacionados indirectamente con la noticia que, de otra manera, le hubiese resultado complejo encontrar. Para esta recomendación se toma el modelo de la subcategoría y la categoría respectivamente. Observando con más detalle la figura, se puede ver cómo estas recomendaciones ofrecen términos relacionados indirectamente con la noticia como: *Atlético*, *Barcelona*, *Bayern* o *Arsenal*, rivales del *Real Madrid* en otros partidos; *Champions*, competición jugada por el *Real Madrid*. También otros términos más genéricos que pueden ser relevantes para el usuario se muestran: *arbitro*, *balón*, *anoto*, *campo*, *área*, *choque*; todos ellos relacionados con el fútbol.

bupco
media

(2-0) Cristiano descose al Milan

El conjunto blanco se reivindicó con una victoria ante un 'grande' y lo hizo gracias a un inicio píetórico. Una falta en la frontal del área fue aprovechada por Cristiano Ronaldo para hacer el primero a los 13 minutos. Cuando los transalpinos no habían asimilado el mazazo, el '7' blanco realizó una gran jugada desde la izquierda y sirvió a Ozil, que puso el segundo tras rebotar en un partido de gran enjundia su mejor la mostrada ante Deportivo de la Coruña y Málaga, el bando local asedió la portería de un Amelia --titular por la lesión de Abbiati-- que cuando se quiso dar cuenta ya estaba recogiendo el esférico de la red en dos ocasiones. El extremo lusitano hizo gala de una de sus especialidades y aprovechó que la barrera lombarda se separó para hacer el primero. Sin tiempo para el respiro, realizó una de sus constantes galopadas por la banda, y dentro del área, puso en bandeja el gol al internacional alemán, que se ayudó en la espalda de Bonera para poner el de la tranquilidad. Con el trabajo hecho mucho antes de los esperados al inicio, los blancos bajaron el ritmo, aunque la presión que realizaba en la salida de balón de su rival, imposibilitó que los lombardos se sintieran cómodos en ningún momento. El equipo de Allegri tardó media hora en aparecer en el encuentro y lo hizo merced a una falta botada por Pirlo. El veterano centrocampista mandó el balón al delantero y avisó de una posible reacción visitante, que no se produjo, a pesar de que el ex madridista Seedorf pudo recortar distancias a diez del descanso. Cuando peor estaban los blancos, Cristiano volvió a aparecer para conducir un gran contragolpe desde su campo, que se encontró con los guantes del meta rival. Tras la reanudación, Higuaín pudo hacer el tercero, pero su flojo testarazo fue atrapado por Amelia. Con el partido más que sentenciado, el Real Madrid se dedicó a no pasar problemas en la zaga y a intentar aumentar la cuenta por medio de un --como siempre-- inconfiable Cristiano. En el minuto 71, Robson da Souza, 'Robinho', que se fue del conjunto de la capital para ser 'el mejor jugador del mundo' fue recibido con una sonora pitada por la parroquia merengue, que todavía no olvida esas declaraciones y que forzará su salida una vez comenzada la temporada. El habilidoso jugador carioca no pudo revolucionar el partido y aunque probó a Casillas en los minutos finales con un tiro lejano, no consiguió inquietar a su ex equipo, que acumula con este triunfo, más los sumados ante Ajax y Auxerre, un impecable inicio en la máxima competición continental. Así, el Real Madrid se coloca con nueve puntos en tres partidos y sigue como líder intratable del Grupo 'G'. La dinámica es la obsesión del madridismo --, afirmó el presidente florentino Píez a principio de la temporada. Parece que el entrenador y los jugadores lo saben y están haciendo todo lo posible para que el 'rey' de Europa vuelva a su trono. El Real Madrid está en órbita. ... Casillas, Arbeloa, Carvalho, Pepe, Marcelo, Xabi Alonso, Khedira, Cristiano Ronaldo, Ozil (Lass, min. 83), Di María (Granero, min. 87), e Higuaín (Benzema, min. 89). Amelia; Zambrotta, Nesta, Bonera, Antonini, Gattuso (Boateng, min. 59), Pirlo, Seedorf, Ronaldinho (Robinho, min. 71), Pato (Izazaga), min. 77) e Ibrahimovic. --: 1-0, min. 13. Cristiano Ronaldo, 2-0, min. 14. Ozil. --: Pedro Píezana (POR). Annoté con tarjeta amarilla a Bonera (min. 73). --: Santiago Bernabéu, Liendo. ROBINHO EN SU REGRESO. FICHA TÉCNICA. RESULTADO: REAL MADRID, 2 - MILAN, 0. (2-0, al descanso). EQUIPOS: REAL MADRID AC MILAN GOLES: ARBITRO ESTADIO

Noticia				
minutos	robinho	plido	madrid	mina
coru	lesi	cristiano	higua	real
amelia	seedorf	pirlo	milan	octavos
esf	ronaldo	doblegar	ozil	lombardo
blanco	bonera	aprovechar	descose	plet

SubCategoría				
centrocampista	anoto	agüero	cesc	campo
arsenal	bayern	champions	chelsea	brasileño
banquillo	area	casillas	barcelona	balon
b	atletico	campeon	azulgrana	choque
barça	bursaspor	bernabeu	cf	allegri

Categoría				
anoto	agüero	arsenal	balon	benfica
arbitro	atletico	barça	aficionados	bernabeu
brasileño				

Figura 19 - Recomendación por niveles

A este nivel también se utiliza la información previa que se tiene sobre el usuario, contenida en su perfil, y se utiliza junto con el contenido de la noticia a la cual se ha accedido para llevar a cabo la recomendación de otras noticias relacionadas. Con ello se desea añadir el conocimiento previo que se tiene del usuario, para recomendar noticias que, estando relacionadas con la noticia consultada, puedan ser de su interés.

Para ejecutar esta recomendación, se toman los términos del modelo del usuario y los términos del modelo de la noticia. Ambos conjuntos de términos se agregan en un modelo común: **modelo_noticia_usuario**. Tomando los términos del modelo_noticia_usuario se realiza una consulta contra el índice que almacena el modelado de las noticias: INDEX_DOC. La consulta es generada de la misma forma que la descrita en el apartado 7.3.1. En la Figura 20 se muestran un ejemplo de noticias recomendadas teniendo en cuenta la noticia consultada y el perfil del usuario. En ella se puede ver como el sistema recomienda otras noticias relacionadas con el partido del que habla la noticia consultada, reflejando diferentes puntos de vista.

Noticias Recomendadas

Niños frente a hombres

Público.es - Un gran triunfo sin más

Público.es - 2-0. El Real Madrid aprueba su primer examen difícil con buen juego ante el Milán

El Real Madrid vence al Milan con comodidad y encarrila el pase a octavos - 20minutos.es

Público.es - Un gran triunfo sin más

Once de gala en el Real Madrid - ABC.es

¿Carvalho, el fichaje del año? - MARCA.com

(2-0) Cristiano descose al Milan

El Real Madrid vence al Milan con comodidad y encarrila el pase a octavos - 20minutos.es

UEFA.com - UEFA Champions League 2011 - Real Madrid-Milan

Figura 20 - Recomendación basada en la noticia consultada

7.4. Colección para experimentación

Para tener un conjunto de contenidos con los cuales se pueda experimentar y evaluar tanto en el ámbito del modelado de usuarios como de la recomendación es necesario hacer uso de una colección para experimentación. En el apartado 5.2 se han mostrado diferentes colecciones, junto con sus características y limitaciones. En este apartado se expone el razonamiento llevado a cabo para decidir cuál de estas colecciones es la más propicia, de cara a su utilización, así como el trabajo realizado sobre dicha colección.

7.4.1. Elección de la colección

En el apartado 5.2 se han presentado diferentes colecciones susceptibles de poder ser utilizadas en este trabajo y se ha reflexionado acerca de las ventajas de uso de una u otra. En esta reflexión se concluye que las colecciones más interesantes para este trabajo son la **colección de *Delicious*** y la **colección de *Buscamedia*** (ver apartado 5.2.4).

Estudiando más en detalle las dos colecciones, la colección de *Buscamedia* era idónea por los contenidos que la conformaban; esto es, un conjunto de noticias ya preprocesadas y clasificadas en categorías y subcategorías. Los contenidos de colección de *Delicious* por el contrario no estaban presentes como tal; en su lugar la colección constaba de un conjunto de URLs en las cuales los contenidos estaban almacenados. Si bien es cierto que los contenidos podían ser descargados de estas URLs, este proceso no es sencillo, dando lugar a menudo a errores de descarga. Además, mientras que los contenidos de la colección de *Buscamedia* ya estaban preprocesados y formateados todos por igual, los contenidos descargados de la colección de *Delicious* tenían que ser preprocesados en orden de estandarizar su formato y eliminar información innecesaria (como por ejemplo las etiquetas HTML). Además la colección de *Delicious* tenía otro problema relacionado con las URLs, mientras que algunas de ellas sí que apuntaban a páginas de las cuales se podía extraer su contenido desde el HTML de las mismas, otras apuntaban a páginas cuyo contenido no estaba reflejado en el HTML, sino que era generado a través de alguna tecnología web como AJAX, JavaScript o FLASH.

Estos problemas asociados a la colección de *Delicious*, finalmente, ha motivado la decisión de elegir la colección de *Buscamedia*, aún a pesar de no tener información de perfiles de usuario. Para tratar de cubrir esta desventaja, se ha planteado una aproximación alternativa que no necesita de estos perfiles. Esta aproximación se detalla en los siguientes subapartados.

7.4.2. Generación de usuarios

Para llevar a cabo una evaluación de laboratorio de un sistema de recomendación es necesario disponer de los siguientes recursos:

- Colección de Contenidos que recopile un número suficiente de contenidos con los cuales se pueda interactuar.
- Colección de Interacciones de Usuario que contenga un registro de interacciones de un conjunto de usuarios sobre los contenidos presentes en la colección de contenidos.
- *Ground Truth* o de juicios de relevancia mediante el cual se tenga un conjunto de contenidos etiquetados de tal forma que establezcan si son de interés para un determinado usuario o no lo son.

La colección de contenidos ya está disponible (ver Apartado 7.4.1); sin embargo, no se dispone de un conjunto de Interacciones de Usuario. De igual modo, al no tener las interacciones de los usuarios no se dispone del conjunto de contenidos que son considerados relevantes por cada usuario. Para resolver este problema, tanto el de la falta de interacciones como de *ground truth*, se ha decidido simular este conjunto de interacciones sobre la colección, generando primero un conjunto de usuarios que las lleven a cabo. Este proceso es explicado a continuación.

7.4.2.1. Proceso de generación

Previamente al proceso de generación de los usuarios, se han definido 3 perfiles de usuario, en función de los contenidos que consulten, a fin de tratar de cubrir diferentes comportamientos de los mismos a la hora de consultar noticias:

- **Usuario general**

Pretende simular el comportamiento de un usuario genérico, que consulte noticias de cualquier temática, sin un interés particular en alguna de ellas. Este tipo de usuario es el más común; una persona que consulta noticias de manera transversal. Trasladando el ejemplo, se podría pensar en una persona que lee todas las secciones del periódico.

Para generar a los usuarios de este perfil, cada usuario consulta noticias del 60% de las categorías (12 categorías) dando lugar a un total de 120 documentos consultados (10 de cada sección).

- **Usuario interesado en una categoría**

Cada una de las categorías de la colección hace referencia a una temática genérica, cuyos contenidos están relacionados entre sí de alguna manera. Un ejemplo de categoría es deportes; en esta categoría se incluyen noticias de temáticas diferentes (e. g. baloncesto, fútbol, tenis) pero con un hilo conductor común.

Este perfil trata de simular un usuario interesado únicamente en una temática de este tipo. Siguiendo con el ejemplo anterior, es fácil imaginar un usuario interesado en deportes que consulte noticias de varias disciplinas (e. g. noticias de la NBA, noticias de La Liga, noticias de la Formula 1).

La metodología a seguir para generar los usuarios pertenecientes a este perfil es la siguiente: las noticias consultadas por cada usuario pertenecen a una única categoría y, dentro de esa categoría, cada usuario consultará el 50% de los documentos de la misma.

- **Usuario interesado en una subcategoría**

Este perfil trata de simular el comportamiento de un usuario cuyos intereses sean muy concretos y definidos; típicamente el perfil de un usuario técnico o experto. Para ello los usuarios pertenecientes a este perfil consultan solo noticias correspondientes a una subcategoría. Las subcategorías recogen noticias con una temática muy acotada y reducida, muy relacionadas entre sí.

Se puede pensar en los usuarios de este perfil como usuarios únicamente interesados en una temática concreta, bien por interés o bien por motivos de trabajo; por ejemplo, un profesional únicamente interesado en noticias relacionadas con la arqueología (categoría presente en la colección).

La metodología consiste en tomar una única subcategoría (imponiendo la condición de que tenga más de 10 documentos por motivos de representatividad) y consultar el 70% de documentos de esa subcategoría.

En la Tabla 13 se muestra un resumen de la metodología para generar cada uno de los perfiles:

Perfiles	Categorías Consultadas	Subcategorías Consultadas	Documentos Consultados
Usuario General	60% de las categorías	---	10 documentos por categoría
Usuario interesado en una categoría	1 categoría	---	50% de documentos de la categoría
Usuario interesado en una subcategoría	---	1 subcategoría	70% de documentos de la subcategoría

Tabla 13 - Perfiles para la generación de usuarios

De cada perfil, se han generado un total de 1000 usuarios. En las siguientes Figuras (Figura 21, Figura 22 y Figura 23) se puede ver un ejemplo de los términos asociados a un usuario concreto de cada uno de los tres perfiles. El tamaño de cada uno de los términos refleja la relevancia del término en el perfil, generado por la aplicación de la técnica de modelado propuesta (ver apartado 7.2).

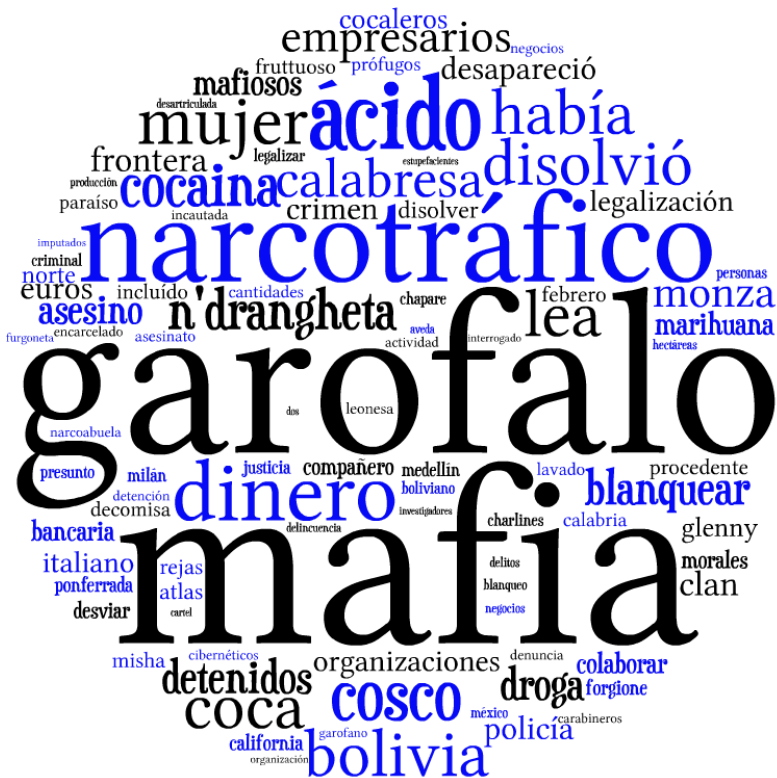


Figura 23 - Términos asociados a un usuario interesado en una subcategoría (crimen organizado)

Una vez finalizada la generación de usuarios se tiene un total de 3000 usuarios, divididos en tres perfiles, susceptibles de ser utilizados para investigación; Sin embargo, hay que tener en cuenta la limitación que supone el hecho de que los datos no se correspondan con usuarios reales, a la hora de poder extraer conclusiones válidas.

Estos datos presentan una serie de problemas que hacen que su utilización no sea muy adecuada. Estos problemas así como la decisión final de no utilizar estos usuarios es tratada más adelante en este trabajo (ver apartado 8.3.1.2).

7.4.2.2. Desarrollo del *Ground Truth*

Además del problema la inexistencia de datos de usuarios reales, también se tiene el problema de la falta de juicios de relevancia necesarios para la evaluación. Dado este problema, se propone la utilización de la propia colección para generar el conjunto de juicios de relevancia: puesto que se tiene el conjunto de contenidos visitados por un usuario, si un usuario ha accedido y consultado un determinado contenido se puede considerar éste como interesante para el usuario. Asumiendo esto como cierto, esta información puede ser tomada como *ground truth*, de la cual extraer los juicios de relevancia. Teniendo en cuenta esto, la metodología para llevar a cabo la experimentación sería la siguiente:

- Se toma de manera aleatoria un porcentaje del total de contenidos considerados interesantes por el usuario (aquellos a los que ha accedido) y se elimina del perfil del usuario. Puesto que los perfiles de los usuarios cuentan con un número de interacciones suficiente, se puede considerar que sus perfiles siguen expresando eficazmente sus preferencias, a pesar de la eliminación de estos contenidos. Este conjunto de contenidos eliminados será tomado como juicios de relevancia.
- Utilizando los perfiles de usuario modificados se le recomienda un conjunto de contenidos al usuario siguiendo la primera de las aproximaciones de recomendación (ver apartado 7.3.1).

7.4.3. Indexado de la colección

Para permitir la recuperación de información de los contenidos de la colección, a través del prototipo desarrollado, es necesario previamente indexar los contenidos de la misma. La indexación de contenidos es un paso indispensable en el área de Recuperación de Información. Mediante el indexado un conjunto de datos es almacenado en un formato que permite optimizar la búsqueda de documentos, en términos de velocidad y rendimiento.

La indexación, y la posterior recomendación de contenidos, es un proceso complejo y se requieren algoritmos y estructuras de datos especialmente diseñados para tal fin. Es por ello que, para este trabajo, se ha decidido hacer uso Solr, una plataforma para facilitar la indexación y recuperación de información de contenidos textuales.

Para indexar las noticias, se ha desarrollado un índice en el cual, para cada noticia de la colección, se tienen los siguientes campos:

- **Id:** Este campo almacena el identificador de la noticia. Este dato será único para cada noticia y será tomado como clave del índice para distinguir un contenido del otro.
- **Título:** Este campo almacena el título de la noticia.
- **Descripción:** Este campo almacena una breve descripción de la noticia, realizada por el propio autor de la noticia.
- **Keywords:** Este campo almacena un conjunto de palabras clave relacionadas con la noticia. Estas palabras clave son fijadas por el autor de la noticia.
- **Entradilla:** Este campo almacena la entrada de la noticia, que introduce el contenido de la misma.
- **Cuerpo:** Este campo almacena el contenido propiamente dicho de la noticia.
- **Categoría:** Este campo almacena la categoría de la noticia.
- **Subcategoría:** Este campo almacena la subcategoría de la noticia.

A la hora de ejecutar una consulta, es posible utilizar cualquiera de estos campos para realizar la búsqueda. Para ello se puede utilizar la sintaxis de Solr (similar a la de Google) para esto:

nombre_campo : consulta

En caso de no especificarse ningún campo, el campo del índice sobre el que se realizará la búsqueda por defecto será el campo *cuerpo*.

Se ha configurado el índice para que el operador por defecto sea *OR*. Este operador, aplicado sobre varios términos devolverá documentos en los que aparezca **al menos uno** de los términos. Se ha optado por este operador por defecto para maximizar la cobertura de los resultados devueltos. Si el usuario quiere realizar una búsqueda con el operador *AND*, deberá realizarlo de manera explícita, mediante la sintaxis de Solr:

termino_consulta_1 AND termino_consulta_2

o bien:

“ termino_consulta_1 termino_consulta_2 “

Por último, otro parámetro importante en la configuración del índice es el tipo de campo. Para este índice se han configurado dos tipos diferentes:

- **string:** Este tipo hace referencia a un campo de texto que no es analizado ni preprocesado. Es utilizado para guardar cadenas de texto utilizadas como claves; puesto que no se realiza ningún tipo de modificación de su contenido y solo se producirá coincidencia si los términos de la consulta son exactamente igual a los términos del campo. Por ejemplo:

*Dado un campo de este tipo conteniendo el texto: **economía y finanzas** y las consultas: **economía o finanzas o Economía y Finanzas**. No se produce una coincidencia con ninguna de ellas. Para que se produjese la coincidencia la consulta tendría que ser exactamente: **economía y finanzas***

Los campos a los que se aplica este tipo son: Id, Título, Keywords, Categoría y Subcategoría.

- **text_rev:** Es tipo hace referencia a un campo de texto analizado sin stemming. Sobre los contenidos almacenados en este campo se realiza el siguiente análisis: división en términos, teniendo en cuenta los espacios en blanco; eliminación de acentos; eliminación de palabras vacías; y conversión en minúsculas. Gracias a este análisis se producirá una coincidencia cada vez que un término de la consulta aparezca en uno de estos campos, sin tener en cuenta acentos o mayúsculas. Por ejemplo:

*Dado un campo de este tipo conteniendo el mismo texto: **economía y finanzas** y las consultas: **economía** o **finanzas** o **Economía y Finanzas** o **economía y finanzas**. Se producen coincidencias para todas ellas.*

Los campos a los que se aplica este tipo son: Descripción, Entradilla y Cuerpo.

Capítulo 8. Evaluación

En esta sección se explica la metodología de evaluación utilizada en este trabajo, se presentan los resultados obtenidos y finalmente se lleva a cabo un análisis de dichos resultados.

La evaluación se lleva a cabo tanto sobre la aproximación de modelado como sobre el propio funcionamiento del recomendador. Con ello se pretende poner en contexto el rendimiento del recomendador, teniendo en cuenta también el rendimiento del modelado a la hora de extraer la terminología más representativa de un contenido (documento, categoría de documentos, perfil de usuarios, etc...).

8.1. Evaluación de la Aproximación de Modelado

El modelado presentado en este trabajo basa su funcionamiento en extraer la terminología que mejor representa a un contenido textual, ya sea esta correspondiente a: noticias, categorías de noticias, subcategorías de noticias o los propios perfiles con la actividad de los usuarios. Por ello, la experimentación, y la posterior evaluación de los experimentos, se centra en comprobar la representatividad de los modelos generados a la hora de describir un contenido.

8.1.1. Configuración de la evaluación

Aunque el modelado presentado en este trabajo se utiliza sobre los documentos, categorías, subcategorías y perfiles de usuario, únicamente se va a tener en cuenta el modelado de categorías y subcategorías para esta experimentación. Esto es así porque, para llevar a cabo la evaluación, se necesita que los contenidos estén clasificados en categorías para las que se pueda generar un *ground truth*, con el cual comparar los resultados obtenidos.

No obstante, las conclusiones extraídas para categorías y subcategorías pueden extrapolarse al modelado de documentos y perfiles de usuario. El modelo de los documentos se puede equiparar al de subcategorías. En ambos casos, el contenido está muy centrado en una temática particular con una terminología muy concreta; al fin y al cabo las subcategorías no son más que una recopilación de documentos con un vocabulario muy similar, o prácticamente idéntico.

Por otra parte, el modelo de los perfiles de usuario es equiparable al de las categorías para aquellos usuarios sin preferencias específicas que consulten todo tipo de noticias: en ambos casos las temáticas son muy variadas y por extensión su terminología también lo será. Para aquellos usuarios que tengan unas preferencias más enfocadas (e. g. consulten solo noticias de fútbol), sus perfiles son equiparables a subcategorías con una temática y terminología mucho más reducida y concreta.

Teniendo este punto en cuenta, para la experimentación se toman los modelos

generados para las categorías y subcategorías (ver apartado 7.1). No todas las categorías y subcategorías han sido seleccionadas, del total de categorías se han eliminado la categoría *error* y la categoría *miscelánea*, ya que incluyen noticias que no han sido clasificadas correctamente debido a algún problema. De otra parte, de las subcategorías se han eliminado todas aquellas que no contengan un número mínimo de noticias (menos de 10 noticias), ya que su contenido no será realmente representativo de dicha subcategoría. Tras este proceso se tiene un total de 17 Categorías y 71 Subcategorías.

8.1.2. Metodología de Evaluación

La metodología de evaluación utilizada se basa en la desarrollada en el artículo de Shmueli-Scheuer et al [128]. En este trabajo se propone un modelado basado en KLD para la creación de perfiles de usuarios en un sistema de búsqueda y recomendación de noticias. Para evaluar el modelado, Shmueli-Scheuer et al proponen comparar los términos identificados como significativos para cada categoría de noticias con una bolsa de palabras que defina a dicha categoría, generada manualmente. La metodología empleada en este trabajo sigue un enfoque similar aunque con ligeras variaciones.

En primer lugar, para poder tener un punto de comparación, de cara a comprobar la validez de la terminología modelada, es necesario disponer de un *ground truth*, conteniendo la terminología que realmente sea representativa de cada categoría (y subcategoría). Para la generación de este *ground truth* se barajaron varias opciones:

- **WordNET:** Es una base de datos léxica que contiene más de 117.000 conceptos, así como las relaciones entre estos. Mediante WordNET es posible, dada una palabra obtener todos los sinónimos de esta. Debido a esta capacidad, WordNET se barajó como la primera opción para generar el *ground truth*: por cada categoría (y subcategoría) se buscarían los sinónimos del nombre de dicha categoría. Por ejemplo, para la categoría *deporte* se buscarían sus sinónimos en WordNET y se añadirían a su bolsa de palabras. Sin embargo, se vio que apenas existían términos relacionados con los nombres de las categorías (una media de 2 conceptos por nombre, y casi nunca más de 5); lo que hizo que se descartase como una alternativa viable.
- **ODP⁴¹:** Open Directory Project (ODP) es un proyecto para generar un directorio jerárquico del contenido de la web. ODP establece una serie de categorías y subcategorías en las cuales se clasifica su contenido. La idea detrás de la utilización de ODP es aprovechar esta clasificación para buscar contenidos relacionados con cada una de las categorías (y subcategorías de nuestra colección). A pesar de que en ODP sí se tiene gran cantidad de contenido referido a cada categoría, resulta complejo extraer terminología. Esto es debido a que el contenido son enlaces a diferentes páginas web por lo que habría que

⁴¹ <http://www.dmoz.org/>

descargar y preprocesar las páginas, además de llevar alguna técnica de extracción de terminología sobre dicho contenido. Por tanto, dada la complejidad de acceso a los contenidos y la necesidad de extracción de terminología, con la pérdida de rendimiento que en este proceso se puede dar, se ha decidido no utilizar este recurso.

- **Clasificación de IPTC:** International Press Telecommunications Council (IPTC) es un consorcio que agrupa a las más importantes agencias de prensa y comunicación. Este consorcio ofrece una clasificación mediante la que se puede categorizar noticias. Ésta es la misma clasificación que se ha utilizado para clasificar las noticias de este trabajo. De nuevo, la idea es aprovechar esta clasificación para generar el *ground truth*. Sin embargo, la utilización de la clasificación de IPTC no es posible debido a que cada categoría y subcategoría solo tiene asociada una breve descripción textual, de la cual no es posible extraer terminología alguna.

Ninguno de los recursos anteriores se adecuaba a los requerimientos de la evaluación planteada: o bien apenas tenían terminología asociada a las categorías o bien era muy complejo extraer la que contenían. Sin embargo, existe un recurso que sí que cumple con todos los requerimientos: *Delicious*. En *Delicious* se dispone de una colección de contenidos etiquetados, de manera social, por los usuarios de la plataforma, mediante una serie de etiquetas. De esta forma, dada una categoría o subcategoría, se pueden obtener otras etiquetas con las que los usuarios han etiquetado los contenidos de dicha categoría.

Una vez decidido que *Delicious* es el recurso a emplear, para cada una de las categorías y subcategorías se genera una bolsa de palabras relacionada, utilizando el nombre de la categoría como consulta contra *Delicious* para obtener etiquetas relacionadas. Serán estas bolsas de palabras el *ground truth* con el cual realizar la evaluación. A continuación se muestra un ejemplo de una categoría y su bolsa de palabras:

baloncesto baloncesto basket deportes nba acb europa olímpico real madrid manager euroleague euroliga kobe la lakers fiba ángeles marca

Con el *ground truth* ya generado, la evaluación de cada uno de los modelos de las categorías y subcategorías se hará buscando la primera aparición de alguno de los términos del *ground truth* en dicho modelo. Puesto que los modelos son un ranking, ordenado por la importancia de cada uno de los términos, cuanto antes se produzca la coincidencia en el ranking, mejor (más representativo) será el modelo. La métrica empleada para esta evaluación es **Match@K** [134], donde **K** es la posición del ranking donde se produce la primera coincidencia. Por ejemplo, dado el modelo, correspondiente a la categoría arte, cultura y espectáculos:

- 1 *película*
- 2 ***cine***
- 3 *matrix*
- 4 *disco*
- 5 *música*
- 6 *facebook*
- 7 *cantante*
- 8 *actor*
- 9 *artista*
- 10 *film*

y el *ground truth* correspondiente a la categoría:

arte, diseño, inspiración, fotografía, gráficos, foto, imágenes, web, color, creativo, cultura, libros, biblioteca, cine, ...

La primera coincidencia entre un término del modelo y del *ground truth* se produce en la posición 2 del ranking del modelo, en el término cine. Por ello, el valor Match@K para este modelo será de 2.

Para tener un punto de comparación, se han aplicado otras estrategias de modelado sobre la colección y se ha aplicado la misma evaluación sobre estos modelados. Las estrategias de modelado que han sido empleadas, además de KLD, son:

- **TF-IDF (Term Frequency – Inverse Document Frequency):** TF-IDF es una aproximación clásica de pesado de términos, utilizada comúnmente en el área de Recomendación de Contenidos.
- **MI (Mutual Information) [91]:** MI ha sido planteado en el ámbito de la recuperación de información como una métrica útil para la selección de características. En el ámbito que aquí nos ocupa, representación textual, MI es útil para seleccionar aquellos términos más representativos de una clase. Más concretamente, MI mide cuánto contribuye la presencia/ausencia de un término a la hora de clasificar un documento, que contenga dicho término, en una clase determinada.
- **χ^2 (Chi2) [91]:** χ^2 es otro método de selección de características similar a Mutual Information. Esta métrica es utilizada para establecer la independencia de dos eventos los cuales, en el caso que nos ocupa, se corresponden con, dado un documento, la ocurrencia de un término y la ocurrencia de una clase.

Se han seleccionado estas tres técnicas porque, por un lado TF-IDF es probablemente la técnica más utilizada en SR basados en contenido y, por otro lado, χ^2 y Mutual Information a pesar de no ser técnicas habituales en SR, son métodos extensamente utilizados a la hora de seleccionar características y extraer terminología.

8.1.3. Análisis de Relevancia Estadística

De cara a poder comparar los resultados obtenidos con KLD con los obtenidos con el resto de técnicas se llevará a cabo un análisis de relevancia estadística sobre los experimentos. De esta manera se podrá establecer si la diferencia entre los resultados es realmente relevante.

Los resultados obtenidos en esta evaluación son proporciones: el tanto por ciento de modelados en los cuales se ha producido una coincidencia para la posición K del modelo (o superiores). Se suponen los datos obtenidos son independientes entre sí (se obtienen de aplicar aproximaciones de modelado diferentes, no dependientes entre sí)

Para comparar los resultados de dos aproximaciones de modelado, y testear su relevancia estadística, se aplicará sobre cada valor de K un test de diferencia de proporciones mediante una prueba tipo χ^2 . En este tipo de pruebas compara las frecuencias observadas en ambas muestras con las que aparecerían bajo la hipótesis nula ($H_0 : p_{KLD} = p_X$). La hipótesis alternativa a probar en este caso sería que los resultados obtenidos con el modelado de KLD (p_{KLD}) es superior a los obtenidos por otro de los modelos (p_X).

La fiabilidad con la que se aceptaría la hipótesis nula es medida como la probabilidad de equivocarse en el rechazo de dicha hipótesis. Se rechazará la hipótesis nula si dicha probabilidad es superior a un umbral. El umbral para considerar si hay riesgo o no de equivocación es llamado *significatividad del contraste* o α . Se rechazará la hipótesis nula si el riesgo de equivocación al asumir la hipótesis nula, también conocido como **p-valor**, es superior a α . El umbral elegido para este trabajo ha sido $\alpha = 0.05$. En la se muestra una representación gráfica de este supuesto.

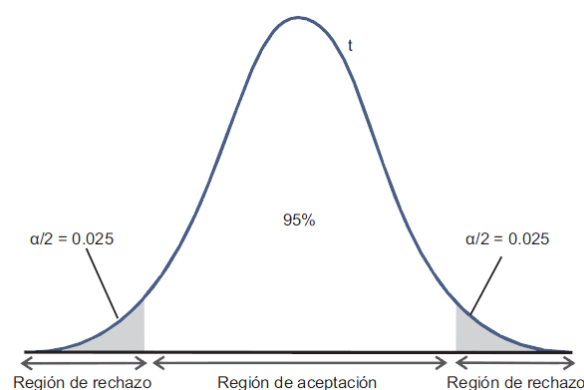


Figura 24 - Rangos de aceptación y rechazo

8.1.4. Resultados

Utilizando la metodología presentada se han evaluado los modelos de cada una de las categorías y subcategorías. En la Figura 25 se muestran los resultados de la evaluación teniendo en cuenta las categorías, mientras que en la Figura 26 se muestran los resultados teniendo en cuenta las subcategorías.

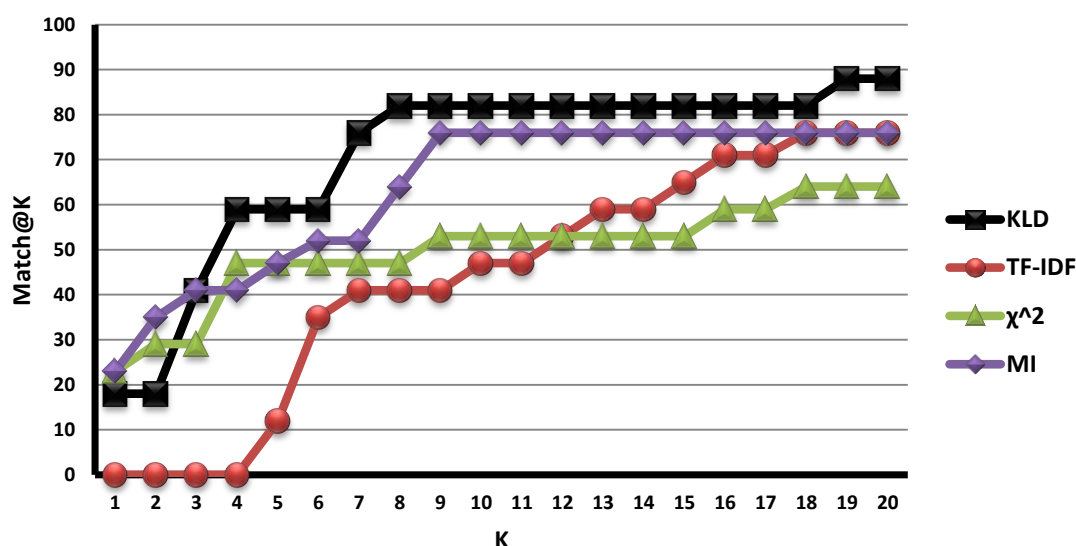


Figura 25 - Evaluación del modelado de las categorías

Las gráficas muestran, en el eje de ordenadas, el tanto por ciento de los modelos en los que se ha producido una coincidencia (**Match@K**) entre uno de sus términos y el *ground truth* en la **posición del ranking K** o superiores; posición mostrada en el eje de abscisas.

Tomando la primera de las gráficas, la de las categorías (Figura 25), se puede ver como el resultado del modelado KLD (marcado con cuadrados en la gráfica) está siempre por encima de TF-IDF (marcado con círculos en la gráfica). Con un número reducido de términos, en los modelos de TF-IDF apenas se producen coincidencias entre el *ground truth* y la terminología modelada, mientras que en los modelos KLD bastan 4 términos para que se produzcan coincidencias en un 60% de los modelos. No es hasta valores de K muy altos ($K > 15$) que el rendimiento de KLD y TF-IDF se aproximan. Sin embargo, del análisis de relevancia estadística se tiene que para estos valores de K superiores a 15, la diferencia entre los resultados de ambas aproximaciones no es significativa.

Fijándose en los resultados de χ^2 y Mutual Information, se observa que ofrecen valores superiores para valores de K iguales a 1 y 2, especialmente Mutual Information. Sin embargo, a partir del tercer término, KLD siempre ofrece valores superiores a ambas. No obstante, el análisis de relevancia estadística de estos datos establece que estas diferencias no son relevantes, puesto que su significancia es muy reducida (existiendo p-valores superiores a 0.25 para gran parte de los valores de K, incluso superiores a 0.5 para K igual a 1 y 2).

Esta diferencia en el comportamiento de los modelados se puede achacar al reducido número de categorías, 17. En este caso, a bajos valores de K se produce un tanto por ciento muy bajo de coincidencias (un 20% de las categorías se traduce en solo 3 categorías). Sin embargo, a medida que K es mayor, el número de categorías involucradas aumenta (50% con K igual a 4), reflejando el comportamiento real del modelado. Este hecho se confirma en el supuesto de las subcategorías, donde el número de éstas es mayor, 71. En este caso, desde un primer momento el modelado de KLD ofrece valores superiores al resto de aproximaciones.

El modelado de las categorías se podría asemejar al modelado de usuarios que no tengan unas preferencias específicas (e.g. usuarios interesados en noticias de todo tipo, o al menos en gran variedad de temáticas). Esto es así debido a que en ambos supuestos, modelos de usuarios y modelos de categorías, las temáticas y por ende la terminología son muy poco concretas y muy variadas. Este tipo de usuarios representa el mayor reto para un sistema de recomendación, ya que es complejo definir un patrón que se adapte a sus preferencias, más allá de recomendar cualquier tipo de contenido.

En este contexto, la aproximación propuesta es capaz de modelar con una exactitud relativamente alta los contenidos, de una manera mucho mayor que una técnica clásica del estado del arte como TF-IDF, y en general, salvo para casos de K menores que 2, también con una exactitud mayor que χ^2 y Mutual Information. Si bien es cierto que debido al reducido número de muestras (17 categorías), no se puede extraer ninguna conclusión significativa de estos datos.

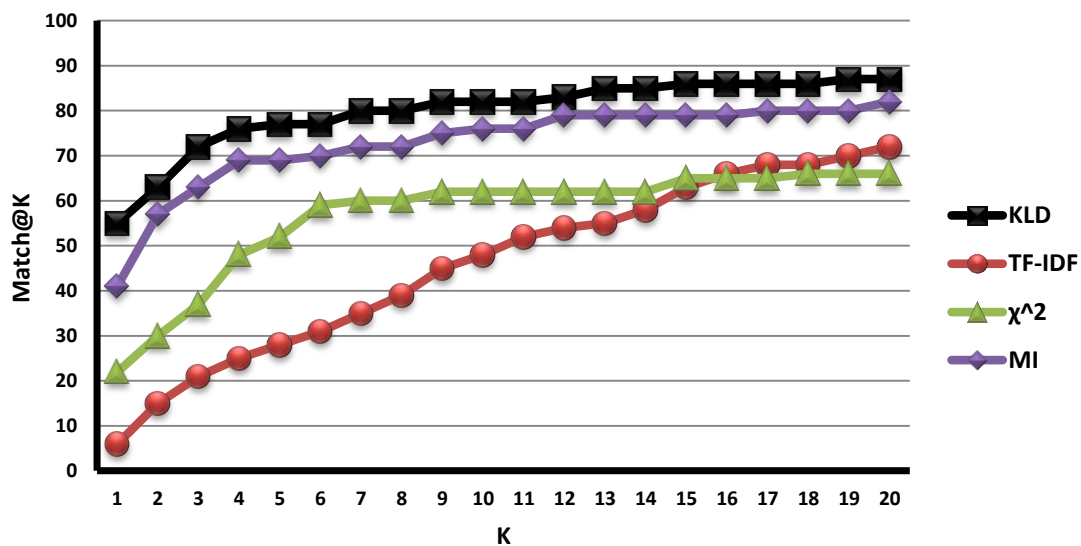


Figura 26 - Evaluación del modelado de la Subcategorías

En la segunda de las gráficas (Figura 26) se puede ver el resultado de la evaluación tomando las subcategorías como referencia. Este tipo de contenidos (subcategorías) se asemeja a perfiles de usuarios con unas preferencias más específicas, centradas en un número reducido de temáticas, debido a que la terminología en ambos casos es muy

específica y hay poca variación. Este escenario es más cercano a un contexto real de aplicación puesto que, raramente, un usuario estará interesado en consultar contenidos de todo tipo, sino que estará interesado en cierto tipo de contenidos (noticias de fútbol, noticias de economía española...).

Con estos contenidos, también el modelado KLD tiene siempre un rendimiento mejor que el modelado TF-IDF. Mientras que con TF-IDF se necesita un valor de K igual a 11 para que se produzcan coincidencias en al menos la mitad de los modelos, con KLD basta un valor de K igual a 1 para ello. Además, para este tipo de contenidos, el modelado de TF-IDF guarda una relación lineal entre K y número de coincidencias, frente al modelado de KLD en el que basta un K igual a 4 para que las coincidencias se establezcan en valores cercanos al 80%. A su vez, utilizando χ^2 y Mutual Information como punto de comparación, se puede observar que KLD también ofrece valores superiores a ambos.

Si bien es cierto que, mientras que para la comparación de KLD con TF-IDF y χ^2 , el análisis de relevancia establece que las diferencias son significativas, para cualquier valor de K; para la comparación de KLD con MI, el análisis de relevancia estadística establece que las diferencias producidas entre ambos valores no son significativas.

Un aspecto importante a destacar es el intervalo temporal entre las noticias de la colección y el *ground truth* utilizado. Mientras que la colección utilizada en este trabajo está extraída de noticias previas a 2011, el *ground truth* se ha generado con aportaciones de usuarios a Delicious de finales de 2012. Este intervalo temporal puede afectar a la evaluación, empeorando los resultados, dado un contexto tan dependiente del tiempo como las noticias: durante este intervalo han podido ocurrir eventos que incluyan nueva terminología (e.g. Terremoto de Japón, Juegos Olímpicos de Londres, Primavera Árabe, Movimiento 15-M,...).

8.2. Evaluación Extendida de la Aproximación de Modelado

La evaluación llevada a cabo en el apartado 8.1 trata de establecer qué tipo de modelado extrae la terminología que mejor representa a una categoría/subcategoría. Sin embargo, con la métrica utilizada (Match@K) solo se tiene en cuenta la primera coincidencia entre el modelo y el *ground truth* de *Delicious*, lo cual podría no reflejar la verdadera **representatividad** del modelo. Por ejemplo se podría dar el siguiente caso:

Dados dos modelos y sus correspondientes *ground truth*:

1	<i>jugador</i>	
2	<i>futbol</i>	
3	<i>balon</i>	
4	<i>portería</i>	
5	<i>jornada</i>	
6	<i>copa</i>	
7	<i>campeonato</i>	
8	<i>delantero</i>	<i>futbol, real, madrid, barcelona, fc, messi,</i>
9	<i>portero</i>	<i>cristiano, champions, league, gol, arbitro,</i>
10	<i>pelota</i>	<i>bernabeu,...</i>

1	<i>jugador</i>	
2	<i>balon</i>	
3	<i>fútbol</i>	
4	<i>real</i>	
5	<i>madrid</i>	
6	<i>barcelona</i>	
7	<i>messi</i>	
8	<i>victoria</i>	<i>futbol, real, madrid, barcelona, fc, messi,</i>
9	<i>champions</i>	<i>cristiano, champions, league, gol, arbitro,</i>
10	<i>españa</i>	<i>bernabeu,...</i>

El primero de los modelos tiene un **Match@2**, mientras que el segundo tiene un valor de **Match@3**. Según la evaluación del punto 8.1, el primero de los modelados sería mejor que el segundo; sin embargo, echando un vistazo a ambos modelos parece que el segundo es más deseable. En éste último modelo se producen más coincidencias, esto es, el modelo parece más representativo.

8.2.1. Metodología de Evaluación

Para abordar la problemática reflejada en el ejemplo expuesto, se hace necesario una nueva evaluación que tenga en cuenta todas las coincidencias del modelado, y no solo la primera. Una primera idea sería contar cuántas coincidencias hay y, cuanto mayor sea este número, más adecuado será el modelo. No es muy difícil darse cuenta de la limitación de este método: no todas las coincidencias tienen la misma importancia. Los dos aspectos que reflejan la importancia de una coincidencia son:

- **Ranking:** Cuanto más arriba esté la coincidencia en el modelo, más deseable será, ya que más importancia tendrá el término en el modelo. Este contexto es equiparable a los resultados devueltos por un sistema de búsqueda de contenidos en donde, cuanto más alto esté el documento devuelto en el ranking de resultados, más importancia tiene.
- **Peso:** El modelo generado además de ordenar los términos en un ranking otorga un peso a cada uno de los términos. Estos pesos reflejan la importancia de cada término como representativo de una clase, de acuerdo a la aproximación de modelado.

Teniendo todo esto en cuenta, se hace necesario tener una métrica que refleje los tres aspectos propuestos: el número de coincidencias, el ranking de cada una de las coincidencias y el peso de cada uno de los términos en los cuales se ha producido una coincidencia. Puesto que, como se ha comentado, el contexto es similar al de la recuperación de información se ha buscado una métrica proveniente de esta área. En RI, existe una métrica utilizada para medir la calidad de un ranking de resultados, teniendo en cuenta tanto el ranking como el peso de los mismos: **Discounted Cumulative Gain (DCG)**.

DCG es una métrica basada en la premisa de que cuanto mayor sea la importancia de un documento en una lista de resultados y más arriba esté en el ranking más importancia tendrá. Aplicada sobre el ranking total, DCG establece cuanta “**ganancia**” aporta a la calidad global del ranking la aparición de un documento con una determinada relevancia en una determinada posición del ranking. La fórmula de DCG se puede ver en la Ecuación 39:

$$DCG_p = rel_1 + \sum_{i=2}^p \frac{rel_i}{\log_2 i} \quad \text{Ecuación 39}$$

donde p es la posición del ranking en la que se quiere conocer el valor de DCG, i representa la posición de un documento en el ranking, y rel_i es la relevancia del documento en la posición del ranking i .

De acuerdo a la formula presentada, la importancia de cada uno de los documentos del ranking, viene marcada por su relevancia, reducida por su posición en el ranking (i.e. mientras más baja sea su posición en el ranking, mayor será el denominador y menor la importancia).

Una extensión de DCG es **nDCG (Normalized Discounted Cumulative Gain)**. Para poder comparar la calidad de varios ranking, DCG no es útil ya que su valor dependerá de los valores de relevancia establecidos para cada documento (i.e. un método que otorgue siempre mucha relevancia a todos los documentos hará que el ranking siempre tenga valores de DCG muy altos, independientemente de la calidad del mismo). Para resolver el problema se aplica una normalización del valor de DCG, dando lugar a nDCG. La formulación de nDCG se presenta en la

$$nDCG_p = \frac{DCG_p}{IDCG_p} \quad \text{Ecuación 40}$$

donde $IDCG_p$ es el valor DCG óptimo del ranking hasta la posición p , es decir que todos los documentos estén correctamente ordenados al menos hasta la posición p .

Gracias a nDCG se pueden tener la puntuación de diferentes rankings normalizada en un intervalo de 0 a 1, permitiendo de esta manera su comparación. Dadas sus características, una métrica similar a nDCG sería deseable para establecer la calidad de los modelos generados en este trabajo. Para ello, el ranking de términos que compone cada uno de los modelos se equipara al ranking de documentos devueltos por un sistema de RI; mientras que el peso otorgado a cada término de un modelo puede equipararse también a la relevancia de un documento en un sistema de RI.

Teniendo estas consideraciones en cuenta se define la siguiente métrica para establecer lo representativo que es un modelo de la categoría/subcategoría a la cual refiere:

$$nRep_p = \frac{Rep_p}{Rep_{Ideal_p}} \frac{w_{match_1} + \sum_{i=2}^p \frac{w_{match_i}}{\log_2 i}}{w_1 + \sum_{i=2}^p \frac{w_i}{\log_2 i}} \quad \text{Ecuación 41}$$

donde $nRep_p$ es la representatividad normalizada del modelo hasta la posición p del ranking y Rep_{Ideal_p} la representatividad ideal hasta la posición p , es decir, la representatividad en caso de que todos los términos del modelado coincidiesen con términos del *ground truth*. Para calcular $nRep_p$ se tendrá en cuenta el peso w_{match_i} de cada coincidencia en el modelado en la posición i . La Representatividad Ideal, Rep_{Ideal_p} , se calcula de igual modo pero teniendo en cuenta todos los términos del modelo, y no solo aquellos en los que se haya producido una coincidencia.

8.2.2. Análisis de Relevancia estadística

Sobre los datos obtenidos mediante esta evaluación también se va a realizar un análisis de relevancia estadística, permitiendo la comparación de los datos. A diferencia de la anterior, formada por proporciones, en esta evaluación los datos se corresponden a valores de un conjunto de muestras independientes.

La manera común de llevar a cabo un análisis estadístico sobre este tipo de datos es la utilización de la prueba t de Student; sin embargo, en este caso esta prueba no es aplicable puesto que los datos no siguen una distribución normal. Por ello el análisis de los datos se ha llevado a cabo mediante la realización de la **prueba U de Mann-Whitney**, la versión no paramétrica de la prueba t de Student. Para poder llevar a cabo esta prueba se debe tener un conjunto de observaciones independientes entre sí. Esta hipótesis se puede asumir como cierta ya que los resultados son obtenidos de la aplicación de técnicas de modelado diferentes, aplicadas de manera independiente.

Mediante esta prueba se compararán cada una de las aproximaciones de modelado frente a KLD, tomando como hipótesis nula que la distribución de los resultados de KLD es igual a la distribución de los datos de otro de los modelos ($H_0 : p_{KLD} = p_X$), y como hipótesis alternativa que la distribución de los resultados de KLD excede a los de la otra distribución.

De manera análoga a como ocurría en el análisis estadístico anterior (ver apartado 8.1.3), la hipótesis nula será rechazada, y los datos serán considerados como significantes, si el riesgo de rechazar dicha hipótesis (p-valor) es inferior a un umbral $\alpha = 0.05$.

8.2.3. Resultados

Utilizando la métrica **nRep** se han evaluado todos los modelos de las categorías y subcategorías generados para este trabajo. Para tener un punto de comparación se ha aplicado esta métrica también a los modelos generados mediante: TF-IDF, Mutual Information y χ^2 . Al igual que se ha hecho en la evaluación previa, el cálculo no se realiza sobre el modelo en general sino sobre los 25 primeros términos (**nRep₂₅**), que son los que concentran la mayor parte del peso del modelado.

Se han evaluado independientemente cada uno de los modelos, obteniendo un valor de **nRep₂₅** para cada uno de ellos. Estos valores se han agrupado por cada una de las aproximaciones de modelado (KLD, TF-IDF, MI y χ^2) tanto para las categorías como para las subcategorías.

Los resultados obtenidos para las categorías se muestran en la Figura 27, mientras que los de las subcategorías se muestran en la Figura 28. En ambos casos se han evaluado independientemente cada uno de los modelos, obteniendo un valor de **nRep₂₅** para cada

uno de ellos. Estos valores se han agrupado por cada una de las aproximaciones de modelado (KLD, TF-IDF, MI y χ^2), sumando cada uno de los valores de **nRep₂₅** individuales y normalizando de acuerdo al número total de categorías/subcategorías. Este valor es el reflejado como **nRep₂₅ Global** en las Figuras.

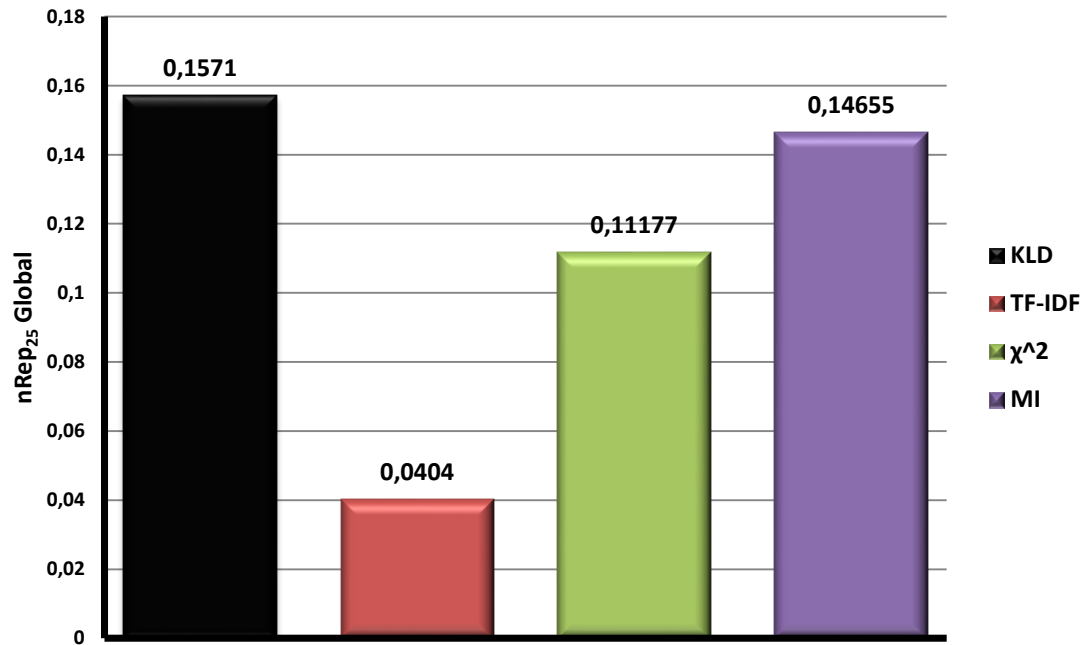


Figura 27 - Representatividad del Modelado de Categorías

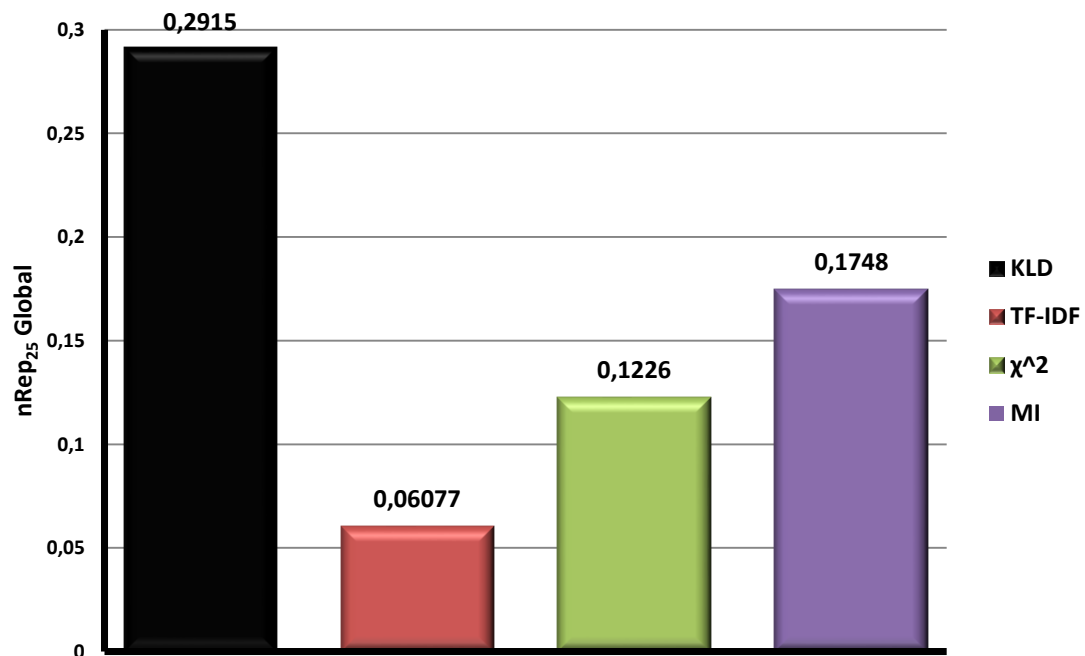


Figura 28 - Representatividad del Modelado de Subcategorías

Analizando los resultados para las Categorías (Figura 27), el mayor valor de **nRep₂₅ Global** es obtenido por el modelo KLD, aunque el modelo de *Mutual Information* obtiene resultados similares. De hecho mediante el análisis de relevancia estadística se comprueba como la significancia de esta diferencia es muy escasa (con un p-valor superior a 0.33). A pesar de ello es interesante destacar como aunque *Mutual Information* obtenía valores de Match@K superiores para valores de K pequeños (ver Figura 25), en esta evaluación el valor de Representatividad Normalizada de KLD es mayor que el de MI. De nuevo esto puede ser explicado debido al reducido número de muestras, que no permite extraer un patrón claro en los resultados.

En cuanto a los modelos de χ^2 , éstos obtienen unos valores menores de **nRep₂₅ Global**, aunque la relevancia esta diferencia de resultados sea mínima, de acuerdo al análisis de relevancia estadística (p-valor superior a 0.2). Sí que se puede tomar como relevante, de acuerdo a este análisis, la diferencia entre los resultados de KLD y TF-IDF para este caso, confirmando también los resultados obtenidos mediante Match@K (ver Figura 25).

Fijándose en las subcategorías (Figura 28), de nuevo es el modelo KLD el que obtiene mejores resultados de **nRep₂₅ Global**: un 66% mejor que *Mutual Information*, un 137% mejor que χ^2 y un 380% mejor que TF-IDF. Estos resultados son similares a los de Match@K (ver Figura 26), a diferencia de que en esta ocasión el análisis de relevancia estadística sí que marca como relevante la diferencia entre los resultados de KLD y MI.

Es destacable que, mientras los valores de Match@K del modelo KLD y del modelo de *Mutual Information* son muy similares para cualquier valor K, pudiendo ser considerados como iguales, el valor de **nRep₂₅ Global** de KLD es mucho mayor que el de *Mutual Information*. Esto se traduce en que, a pesar de que en ambos modelos la primera coincidencia se produce en posiciones del ranking similares, en el resto de términos del modelo se producen más coincidencias y de mayor relevancia con KLD que con *Mutual Information*.

8.3. Evaluación del Recomendador

Para llevar a cabo la evaluación se plantean dos aproximaciones, una evaluación experimental y una evaluación basada en usuario. Con esto se pretende cubrir el análisis del funcionamiento del sistema tanto en términos de rendimiento, a través de las medidas de evaluación comúnmente referidas en el estado del arte, como en términos de usabilidad mediante una evaluación llevada a cabo por usuarios del sistema.

8.3.1. Evaluación experimental

Para llevar a cabo esta evaluación se toma como base los resultados generados en la fase de experimentación, extraídos mediante la generación de perfiles de usuario (ver

Apartado ¡Error! No se encuentra el origen de la referencia.)

8.3.1.1. Metodología de Evaluación

Para llevar a cabo esta evaluación se hace uso tanto de los resultados como de los juicios de relevancia generados en la fase de experimentación. La metodología para llevar a cabo la evaluación se basa en comparar la lista de contenidos recomendados con el *ground truth* generado y aplicar alguna de las métricas utilizadas en recomendación para evaluar los resultados obtenidos.

Siguiendo esta metodología no tendría lógica emplear una medida basada en **precisión**, como se expone a continuación. En este marco se puede definir la precisión como los contenidos recomendados que son interesantes para el usuario, de entre el total de contenidos recomendados. Sin embargo, dada la naturaleza de los juicios de relevancia empleados, se puede dar el caso de que existan contenidos que, a pesar de ser relevantes, no hayan sido visitados con anterioridad por el usuario, con lo que serían clasificados como un fallo. En este supuesto, si aplicamos una medida de precisión, únicamente se sabrá cuándo el sistema ha acertado, pero no cuándo ha fallado, ya que si un contenido no está en los juicios de relevancia puede ser, o bien porque no sea relevante, o bien porque no haya sido visitado por el usuario.

Descartando las medidas de precisión, una medida que sí podría ser interesante es la **cobertura** o *recall*. Mediante la cobertura se mide, del total de contenidos que son considerados relevantes, cuántos son finalmente recomendados por el sistema. Este tipo de medida sí podría ser aplicada sin problemas al tipo de evaluación planteada. El problema es que esta medida, sin estar acompañada de la precisión, solo nos cuenta una parte de la historia y no nos permite hacernos una idea general del funcionamiento del sistema. Para dejar más claro este punto baste el siguiente ejemplo:

Un sistema de recomendación que considere todos los contenidos como relevantes (algo poco deseable) tendrá una cobertura máxima. Este tipo de sistemas es penalizado con la precisión, ya que producirá numerosos errores y su puntuación en este punto será muy baja.

Otro tipo de medidas que comúnmente son utilizadas en los SR de contenidos son las medidas **basadas en error**. Este tipo de medidas se utilizan cuando el interés de un usuario por un contenido es expresando mediante una puntuación no binaria ni unaria (i.e. un sistema de puntuación de 1 a 10). En este contexto las medidas basadas en error miden la diferencia entre la puntuación estimada por un sistema y la puntuación realmente dada por el usuario. Para llevar a cabo la evaluación, del conjunto de contenidos recomendados se toma solo aquellos que estén presentes también en los juicios de relevancia y se calcularía el error producido.

Esta aproximación tiene la desventaja de que no se tiene en cuenta todo el conjunto de contenidos recomendados por el sistema. En el caso extremo, es posible que en el conjunto de contenidos recomendados que coincidan con los juicios de relevancia se produzca un error muy bajo pero que en el resto de contenidos el error producido sea

muy alto, sin que éste quede reflejado en los resultados.

8.3.1.2. Justificación de uso

Como ya se ha comentado, en el apartado de experimentación y anteriormente en éste mismo apartado, esta aproximación de evaluación presenta, por un lado, el problema de no disponer de datos de usuarios reales y, por otro, los problemas a la hora de escoger una métrica que pueda establecer de manera precisa el funcionamiento del sistema (ver apartado anterior).

Debido a estos problemas, los resultados que se puedan extraer de esta evaluación no son válidos para la extracción de conclusiones acerca del funcionamiento de los métodos de recomendación planteados. Por ello, la evaluación de la parte correspondiente a la recomendación de contenidos se va a centrar únicamente en la evaluación centrada en los usuarios, presentada en la siguiente sección.

8.3.2. Evaluación centrada en el usuario

Mediante esta experimentación se pretende tener un conjunto de datos acerca de la actividad de usuarios reales en su utilización del sistema. Los datos son recopilados de manera implícita, registrando las interacciones del usuario con el sistema, y de manera explícita, solicitándole su opinión del sistema de manera directa a través de un cuestionario.

La evaluación está enfocada a la utilización del sistema por parte de **usuarios no experto**. A cada usuario se le plantean una serie de tareas que tendrá que completar utilizando el sistema. Para tener un punto de comparación, además de llevar a cabo las tareas con el sistema de recomendación, se les planteará también a los usuarios llevar a cabo también las tareas con un sistema de RI tradicional, sin recomendación de contenidos.

8.3.2.1. Configuración de la Evaluación

A cada usuario se le pide realizar un total de 10 tareas (ejecución de consultas), 3 de ellas serán realizadas con un sistema de RI obligatoriamente, otras 3 con el sistema de recomendación obligatoriamente y las 4 últimas con el sistema que escoja el usuario. El usuario deberá utilizar estos sistemas para encontrar información relevante acerca de cada una de las tareas; cuando él considere que ha encontrado suficiente información dará por finalizada la tarea y pasará a la siguiente. El tiempo máximo que se le ofrece al usuario para realizar cada tarea es de 5 minutos, en caso de superarlos la tarea se dará por concluida por parte del sistema.

A los usuarios se les ha dividido en 2 grupos, a uno de los grupos se le requerirá hacer las 3 primeras tareas con el Buscador y las 3 siguientes con el Recomendador, dejando las 4 últimas a su elección, mientras que al otro grupo se le requerirá que utilice primero el Recomendador y posteriormente el Buscador. Con esto se pretende asegurar que los

resultados obtenidos sean achacables al tipo de sistema utilizado en lugar de a la tarea (i.e. podría darse que debido a la dificultad de una tarea concreta, esta produzca muchas más búsquedas o puntuaciones más bajas por ejemplo, si esta tarea hubiese sido únicamente realizada por un tipo de sistema podrían aparecer sesgos en los resultados).

8.3.2.1.1. Tareas planteadas

Con estas tareas se pretende recopilar, de manera implícita, información acerca de cómo el usuario interacciona con el sistema. De manera general, para este tipo de evaluaciones, las tareas consisten en plantear una necesidad de información de un usuario y comprobar cómo se resuelve ésta con el sistema. Para este trabajo en particular se han planteado 3 tipos de necesidades de información que un usuario puede tener, las cuales un sistema de RI clásico puede tener dificultades de satisfacer. Se pretende ver si a través de la utilización de un sistema de recomendación de contenidos puede mejorar la satisfacción de los usuarios. Los 3 tipos propuestos se detallan a continuación:

- Se le plantea al usuario una necesidad de información genérica; por ejemplo: *busque información relacionada con el Mundial de Fútbol de Sudáfrica 2010.*

Mientras que un sistema de RI es bueno devolviendo información acerca de un evento concreto, le resulta mucho más complejo ofrecer información precisa cuando se necesitan contenidos relacionados con un evento genérico. En este punto la recomendación puede resultar útil al ofrecer contenidos relacionados con un contenido concreto, devuelto por el sistema de IR. La esperanza es que estos contenidos recomendados pertenezcan al evento genérico (en este caso el Mundial), del cual se necesita información.

- Se le plantea al usuario una necesidad de información acerca de una cadena de eventos; por ejemplo: *busque información sobre los partidos de España en el Mundial de Sudáfrica 2010.*

De igual modo que en el caso anterior, a un sistema de RI le resulta difícil devolver información que pertenezca a varios eventos, aunque estos tengan alguna relación entre sí. Se espera con la recomendación ofrecer contenidos relacionados, los cuales se espera que pertenezcan a otros eventos relacionados.

- Se le plantea al usuario una necesidad de información acerca de un único evento, pero de manera imprecisa; por ejemplo: *Goles importantes en la carrera de Iniesta.*

Con anterioridad se ha comentado que los sistemas de RI son relativamente precisos devolviendo información acerca de un evento en concreto. Sin embargo, hay ocasiones en las que dichos sistemas pueden tener problemas. Una de estas ocasiones es cuando la necesidad de información, bien es imprecisa, o bien está expresada de manera imprecisa. La idea es que el sistema de recomendación pueda ofrecer contenidos que resuelvan, al menos en parte, esta

imprecisión.

Las tareas/consultas propuestas que abarcan los 3 tipos de necesidades de información explicadas son las siguientes:

- Busque recursos acerca de las manifestaciones producidas en Francia por las reformas de Sarkozy
- Encuentre recursos relacionados con declaraciones de Esperanza Aguirre
- Encuentre recursos relevantes sobre las Cajas relacionadas con la crisis
- Encuentre recursos acerca de las películas del festival de Sitges 2010
- Encuentre recursos que tengan que ver con la Liga de Campeones
- Encuentre recursos que traten el tema de la burbuja inmobiliaria
- Busque recursos que traten el tema de la crisis financiera
- Busque recursos útiles para encontrar partidos ganados por el FC Barcelona
- Busque recursos útiles para identificar nombres de Jueces de la Audiencia Nacional
- Busque recurso útiles para identificar nombres de Ministros, o exministros españoles

8.3.2.1.2. Cuestionario

Además de la información recopilada anteriormente, también es importante conocer de primera mano la opinión del usuario acerca del funcionamiento del sistema. Para ello, una vez finalizadas las tareas, se pide al usuario que complete un cuestionario con preguntas relacionadas con el funcionamiento del sistema, así como con su experiencia en la utilización del mismo. En el cuestionario también se incide en el aspecto de si la utilización de un recomendador mejora la utilidad del sistema a la hora de resolver las tareas planteadas. A continuación se muestra el cuestionario que es dado a los usuarios del sistema (ver Figura 29):

***1. Valore la sencillez y comprensibilidad del Sistema de Búsqueda SIN Recomendación**

- Muy Complejo
- Complejo
- Sencillo
- Muy Sencillo

***2. Valore la utilidad del Sistema de Búsqueda SIN Recomendación a la hora de llevar a cabo las tareas propuestas**

- Muy Alta
- Alta
- Aceptable
- Baja
- Muy Baja

3. ¿Qué es lo que más le ha gustado del Sistema de Búsqueda SIN Recomendación?

4. ¿Y lo qué menos?

***5. Valore la sencillez y comprensibilidad del Sistema de Búsqueda CON Recomendación**

- Muy Complejo
- Complejo
- Sencillo
- Muy Sencillo

***6. Valor la utilidad del Sistema de Búsqueda CON Recomendación a la hora de llevar a cabo las tareas propuestas**

- Muy Alta
- Alta
- Aceptable
- Baja
- Muy Baja

7. ¿Qué es lo que más le ha gustado del Sistema de Búsqueda CON Recomendación?

8. ¿Y lo qué menos?

***9. ¿Le han resultado de ayuda las Recomendaciones del Sistema de Búsqueda con Recomendación a la hora de llevar a cabo las tareas propuestas?**

- Sí
- No
- No han supuesto una diferencia

***10. ¿Qué tipo de recomendación le ha resultado más útil? (Elija todas las que considere oportuno)**

- Recomendación tomando el perfil del usuario
- Recomendación tomando los resultados de búsqueda
- Recomendación tomando la noticia que se está consultando

Figura 29 - Cuestionario de Evaluación

8.3.2.2. Metodología de Evaluación

A continuación se detalla la metodología seguida para llevar a cabo la evaluación del recomendador centrada en el usuario. Para la evaluación se utilizan dos tipos de información: información obtenida de manera implícita, registrando la actividad del usuario, e información obtenida de manera explícita, preguntándole directamente al usuario.

8.3.2.2.1. Información recopilada mediante los logs

La información contenida en los logs almacena cada una de las interacciones que el usuario ha llevado a cabo con el sistema durante la realización de las tareas. Gracias a esta información se puede observar como el usuario interactúa con el sistema, tanto el sistema de RI, como el sistema de recomendación. La información recogida en los logs de usuario es la siguiente:

- Búsquedas realizadas por el usuario
- Documentos consultados
- Etiquetas consultadas
- Puntuación dada a los documentos
- Tiempo total dedicado a la tarea
- Navegación a través de las páginas de resultados
- Elección de uno u otro sistema para llevar a cabo las tareas

Con estos datos se pretende comprobar como varía el comportamiento de los usuarios en la realización de las tareas en función de si están utilizando el sistema con recomendación o no. Mediante esta información se pueden extraer patrones de comportamiento que reflejen en qué medida la utilización del recomendador ayuda a los usuarios.

8.3.2.2.2. Información recopilada mediante el cuestionario

El cuestionario está formado por un total de 10 preguntas, de las cuales 6 preguntas son obligatorias y 4 son opcionales. Las 4 preguntas opcionales son:

- **Pregunta 3:** ¿Qué es lo que más le ha gustado del Sistema de Búsqueda SIN Recomendación?
- **Pregunta 4:** ¿Y lo que menos?
- **Pregunta 7:** ¿Qué es lo que más le ha gustado del Sistema de Búsqueda CON Recomendación?
- **Pregunta 8:** ¿Y lo que menos?

Estas preguntas le permiten al usuario de poder expresar, de manera libre, aquellos comentarios que tenga sobre el funcionamiento del sistema. Para analizar estos campos se llevará un análisis cualitativo sobre los textos, destacando aquellos aspectos más interesantes o más comentados por los usuarios.

Estas 6 preguntas obligatorias requieren que el usuario elija la opción que más acertada considere, de un grupo de ellas, para cada uno de los temas planteados en las preguntas. Dentro de las preguntas obligatorias existen dos tipos de preguntas: preguntas con respuestas únicas y preguntas con respuestas múltiples. Las preguntas con respuestas únicas son:

- **Pregunta 1:** Valore la sencillez y comprensibilidad del Sistema de Búsqueda SIN Recomendación.
- **Pregunta 2:** Valore la utilidad del Sistema de Búsqueda SIN Recomendación a la hora de llevar a cabo las tareas propuestas.
- **Pregunta 5:** Valore la sencillez y comprensibilidad del Sistema de Búsqueda CON Recomendación.
- **Pregunta 6:** Valore la utilidad del Sistema de Búsqueda CON Recomendación a la hora de llevar a cabo las tareas propuestas.
- **Pregunta 9:** ¿Le han resultado de ayuda las Recomendaciones del Sistema de Búsqueda con Recomendación a la hora de llevar a cabo las tareas propuestas?

La pregunta con respuestas múltiples es:

- **Pregunta 10:** ¿Qué tipo de recomendación le ha resultado más útil? (Elija todas las que considere oportuno)

Estas 6 preguntas están enfocadas a conocer la opinión del usuario acerca del funcionamiento de ambos sistemas y la comparación entre ambos. Sobre estas respuestas se llevará a cabo un análisis cuantitativo de las respuestas de usuarios.

8.3.2.3. Resultados

8.3.2.3.1. Logs

Una vez finalizado el periodo de evaluación y recopilados los datos se tiene que 31 usuarios se han registrado para llevar a cabo la evaluación. A cada usuario se le ha dado una introducción acerca de lo que la evaluación persigue y se le ha ofrecido un manual para conocer cómo funciona tanto la evaluación como el propio sistema. Cabe remarcar que ninguno de los usuarios había tenido contacto previo con el sistema ni ha recibido ningún tipo de formación adicional.

Sin embargo, no todos los usuarios que se han registrado han llevado a cabo la evaluación al completo. Del total de 31 usuarios registrados, solamente 23 han llevado a cabo el proceso completo. Para evitar datos sesgados o erróneos, de cara a evaluar los resultados se han tenido únicamente en cuenta los logs de estos 23 usuarios que han finalizado el proceso.

Por tanto se tienen un total de 23 usuarios habiendo realizado cada uno de ellos 10 tareas, siendo el total de tareas realizadas de 230. El número total de interacciones llevadas a cabo por los usuarios son 2530, siendo de ellas: 369 búsquedas, 413 accesos a documentos, 796 ratings a documentos y 170 navegaciones entre páginas de resultados por citar algunas de las más importantes.

En la Tabla 14 se muestran los valores absolutos obtenidos con el Recomendador y con el Buscador para los tipos de información recogida en los logs de usuarios más interesantes.

	Recomendador	Buscador
Nº de usos del recomendador	140	90
Nº de interacciones con el recomendador	1448	1082
Nº de búsquedas realizadas	199	170
Nº de accesos a documento	227	186
Nº de accesos a documento tras ser recomendados	107	X
Nº de ratings de documentos	378	418
Nº de ratings de documentos con un 1	39	46
Nº de ratings de documentos con un 2	16	37
Nº de ratings de documentos con un 3	43	70
Nº de ratings de documentos con un 4	79	110
Nº de ratings de documentos con un 5	201	155
Nº de navegaciones realizadas	44	126
Nº de acceso a etiquetas	160	X

Tabla 14 - Análisis de los logs

El primer punto a destacar de la tabla es el número de veces que los usuarios han utilizado cada uno de los dos sistemas. Mientras que el Buscador ha sido utilizado 90 veces, es decir en 90 tareas, el Recomendador ha sido utilizado 140. Teniendo en cuenta que al menos cada usuario tenía que utilizar en 3 tareas cada sistema de manera obligatoria, el número de tareas en las cuales el usuario podía elegir entre un sistema u otro era de 4. Extrapolado esto al total de usuarios da lugar a que 69 de los usos tanto del Recomendador como del Buscador son obligatorios, siendo el resto decisión de los usuarios.

Teniendo esto en cuenta, de las 92 elecciones totales que los usuarios podían hacer (4 elecciones por usuario x 23 usuarios), en 21 ocasiones los usuarios han escogido el Buscador mientras que en 71 ocasiones el elegido ha sido el Recomendador. Estos datos dan una idea de la satisfacción de los usuarios con el sistema de recomendación, puesto que cuando se han visto instados a elegir un sistema han elegido el Recomendador en más del triple de ocasiones que el Buscador.

El resto de los datos vienen marcados por este mayor número de usos del Recomendador, ya que si se ha usado más éste es lógico que haya más interacciones de todos los tipos (búsquedas, ratings, acceso a documentos) que con el Buscador. Por ello, en lugar de tener en cuenta el número de interacción en bruto se tendrá en cuenta el número de interacciones por tarea. Para esto, en se dividirá cada uno de los números asociados a los tipos de interacción de cada sistema entre el total de tareas llevadas a cabo con dicho sistema. Los datos normalizados se muestran en la Tabla 15.

Atendiendo a los datos normalizados se puede ver que, en líneas generales, con el recomendador los usuarios tienen que realizar menos interacciones con el sistema para completar las tareas (10.34 interacciones/tarea con el Recomendador frente a 12.02 con el Buscador). Este dato puede achacarse a que gracias a las recomendaciones ofrecidas al usuario le resulta más fácil encontrar contenidos relacionados con las tareas.

Esta diferencia en el número de interacciones es aún mayor si se tiene en cuenta el número de búsquedas realizadas (1.42 búsquedas/tarea con el Recomendador frente a 1.89 con el Buscador) y especialmente atendiendo al número de navegaciones entre páginas de resultados (0.31 navegaciones/tarea con el Recomendador frente a 1.4 con el Buscador). Este último dato es especialmente significativo ya que indica que el usuario tiene que navegar mucho más a través de los resultados utilizando el Buscador que utilizando el Recomendador, probablemente debido a la utilidad de las recomendaciones a la hora de completar las tareas.

Atención aparte requieren los datos relativos a los ratings de los documentos con uno u otro sistema. Utilizando del Buscador los usuarios puntúan muchos más documentos que utilizando el Recomendador (4.65 ratings/tarea con el Buscador frente a 2.7 ratings/tarea con el Recomendador). Esto es achacable de nuevo a que los usuarios tienen que realizar muchas más interacciones con este sistema para completar las tareas y por lo tanto entran en contacto con mucho más documentos. Comparando las puntuaciones establecidas por el usuario, con el Recomendador el 53% de las puntuaciones establecidas son de 5 y el 83% son de al menos 4. Por otra parte, con el Buscador este porcentaje queda reducido a un 37% de puntuaciones igual a 5 y un 63% superiores a 4. Parece por tanto que, a pesar de puntuar menos documentos con el Recomendador, los usuarios encuentran más interesantes los documentos que ofrece el Recomendador que los que ofrece el Buscador.

Tomando únicamente los datos propios del Recomendador (Nº accesos a documentos recomendados y Nº de acceso a etiquetas) se observa que un 46% de los accesos a documentos mediante el Recomendador son accesos a documentos que han sido Recomendados. Parece ser pues que los usuarios encuentran satisfactorios los contenidos recomendados. Teniendo en cuenta el otro tipo de recomendación ofrecida por el sistema, las etiquetas, se puede ver que el número de accesos a etiquetas es equiparable al número de búsquedas realizadas, teniendo en cuenta que la primera interacción que se debe realizar en una tarea es casi obligatoriamente una búsqueda (1.42 búsquedas/tarea frente a 1.14 accesos a etiquetas/tarea).

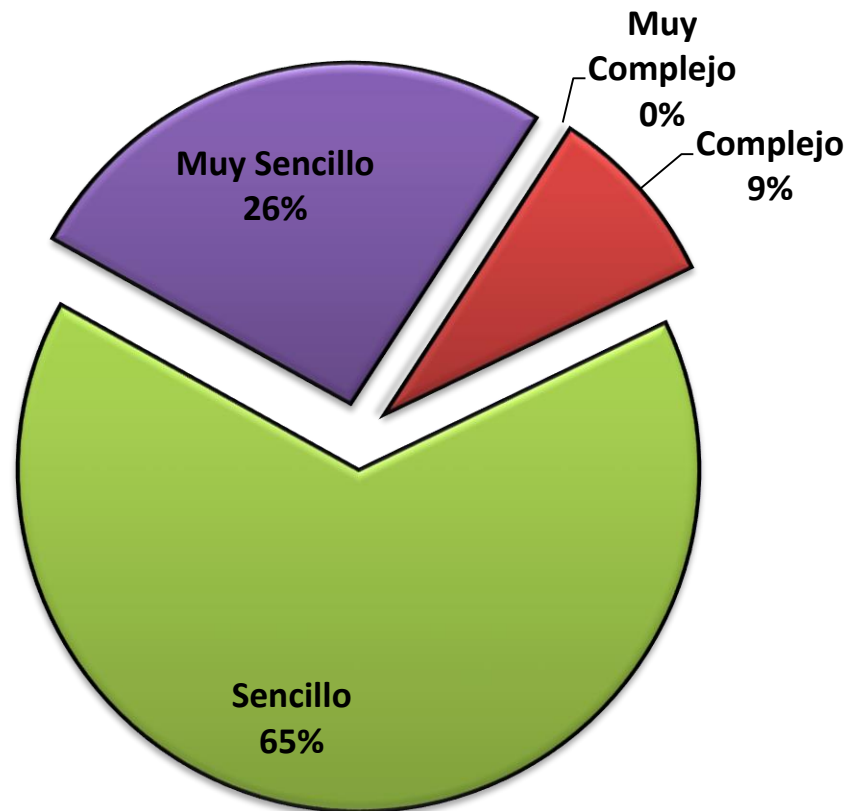
	Recomendador	Buscador
Nº de interacciones con el recomendador	10.34	12.02
Nº de búsquedas realizadas	1.42	1.89
Nº de accesos a documento	1.62	2.07
Nº de accesos a documento tras ser recomendados	0.76	X
Nº de ratings de documentos	2.7	4.65
Nº de ratings de documentos con un 1	0.27	0.51
Nº de ratings de documentos con un 2	0.11	0.41
Nº de ratings de documentos con un 3	0.31	0.78
Nº de ratings de documentos con un 4	0.56	1.22
Nº de ratings de documentos con un 5	1.44	1.72
Nº de navegaciones realizadas	0.31	1.4
Nº de acceso a etiquetas	1.14	X

Tabla 15 - Análisis de los logs normalizado por número de tareas

8.3.2.3.2. Cuestionario

A continuación se muestran los resultados obtenidos para cada una de las preguntas del cuestionario. En primer lugar se muestra el análisis cuantitativo de las 6 preguntas obligatorias:

Valore la sencillez y comprensibilidad del Sistema de Búsqueda SIN Recomendación



■ Muy Complejo ■ Complejo ■ Sencillo ■ Muy Sencillo

Figura 30 - Respuesta Pregunta 1

Valore la utilidad del Sistema de Búsqueda SIN Recomendación a la hora de llevar a cabo las tareas propuestas

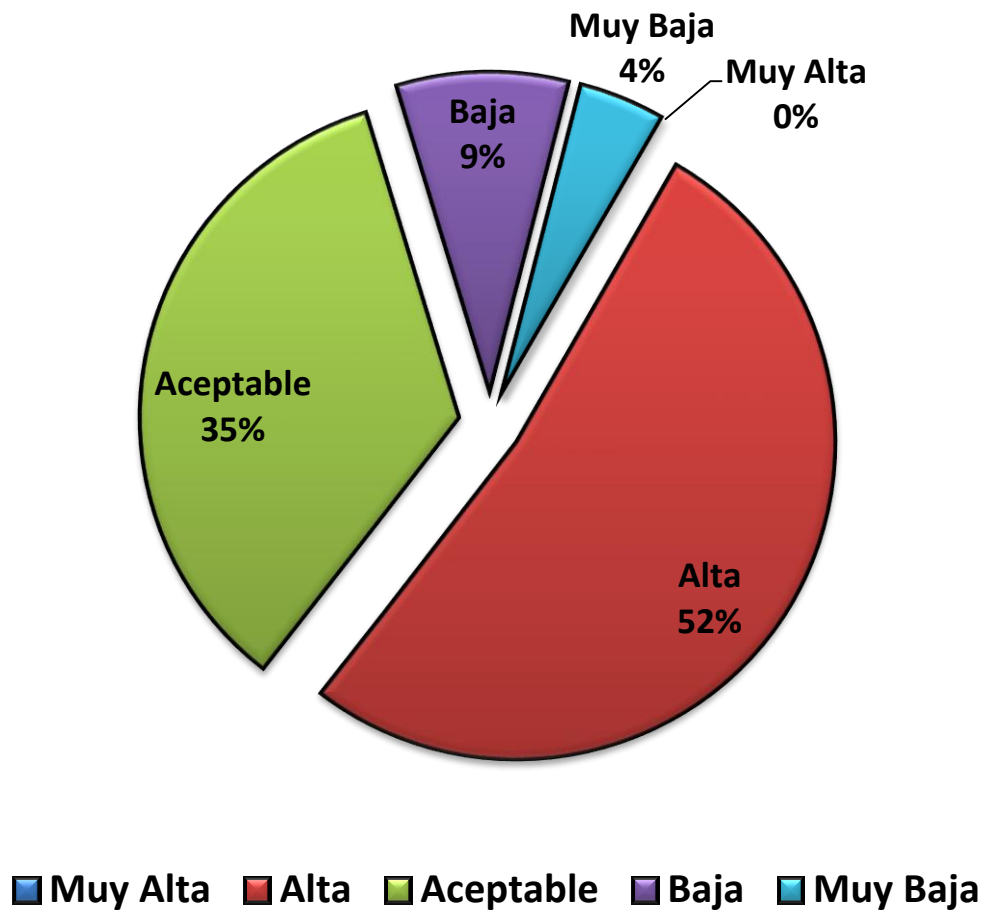
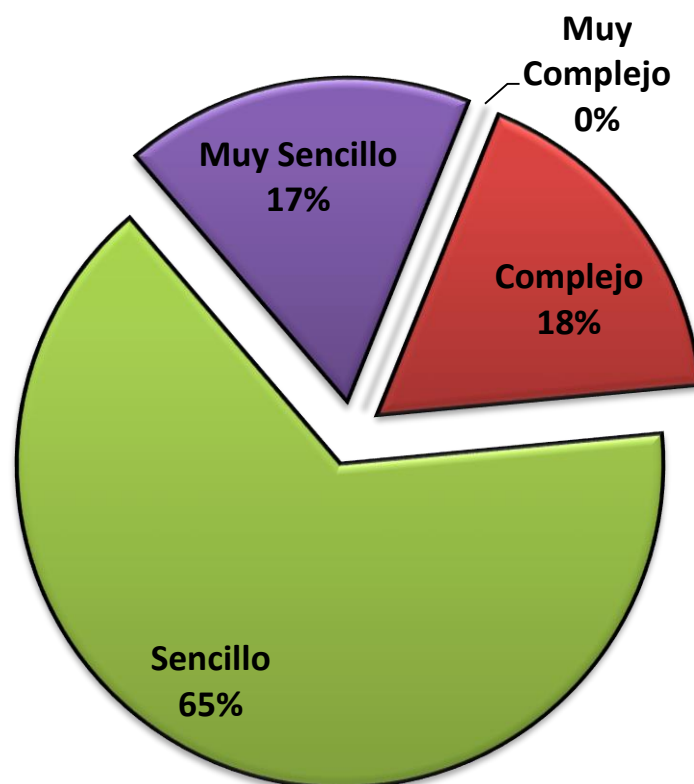


Figura 31 - Respuesta Pregunta 2

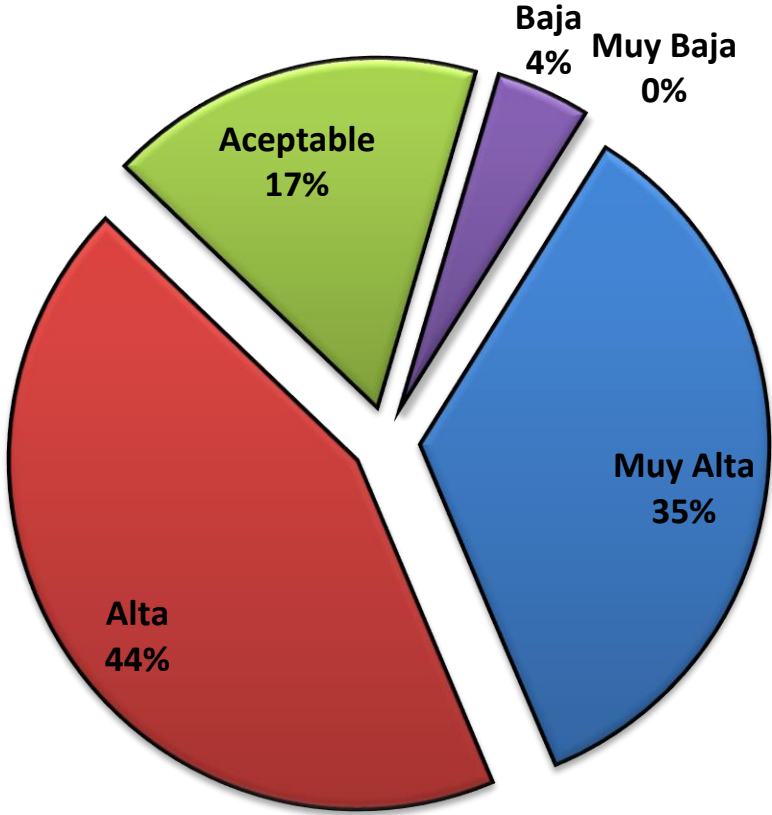
Valore la sencillez y comprensibilidad del Sistema de Búsqueda CON Recomendación



■ Muy Complejo ■ Complejo ■ Sencillo ■ Muy Sencillo

Figura 32 - Respuesta Pregunta 5

Valore la utilidad del Sistema de Búsqueda CON Recomendación a la hora de llevar a cabo las tareas propuestas



■ Muy Alta ■ Alta ■ Aceptable ■ Baja ■ Muy Baja

Figura 33 - Respuesta Pregunta 6

¿Le han resultado de ayuda las Recomendaciones del Sistema de Búsqueda con Recomendación a la hora de llevar a cabo las tareas propuestas?

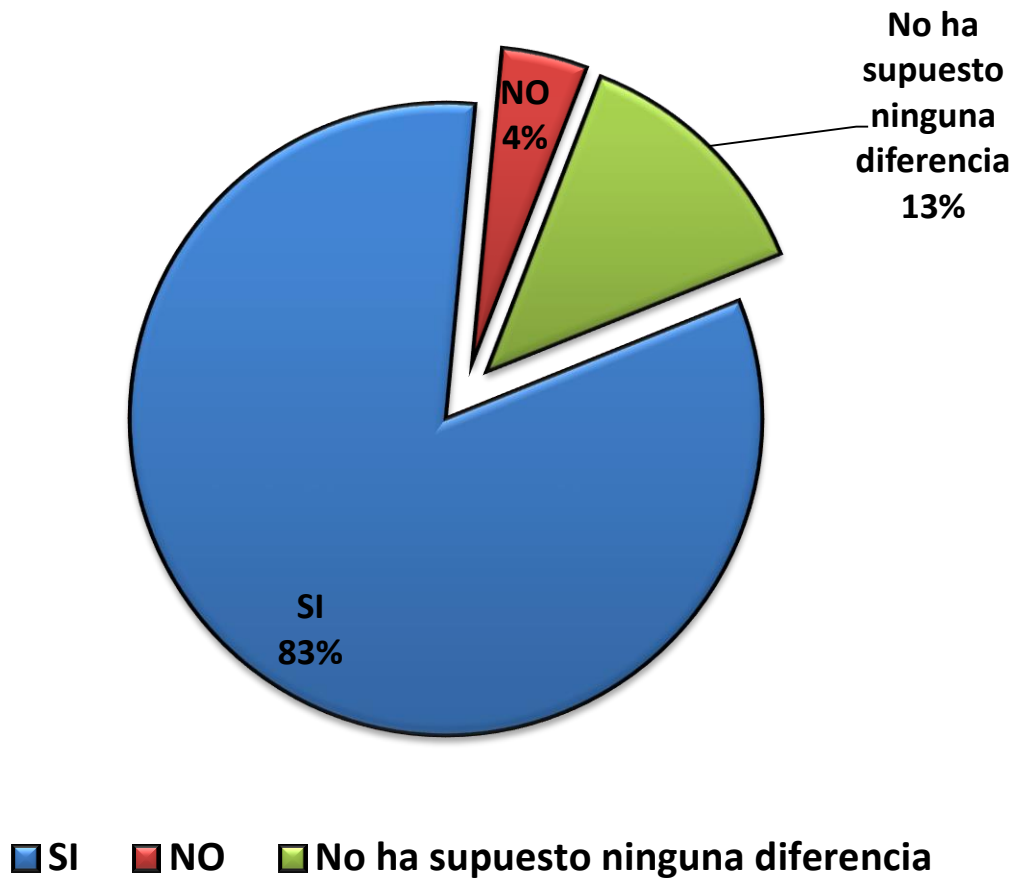


Figura 34 - Respuesta Pregunta 9

¿Qué tipo de recomendación le ha resultado más útil?

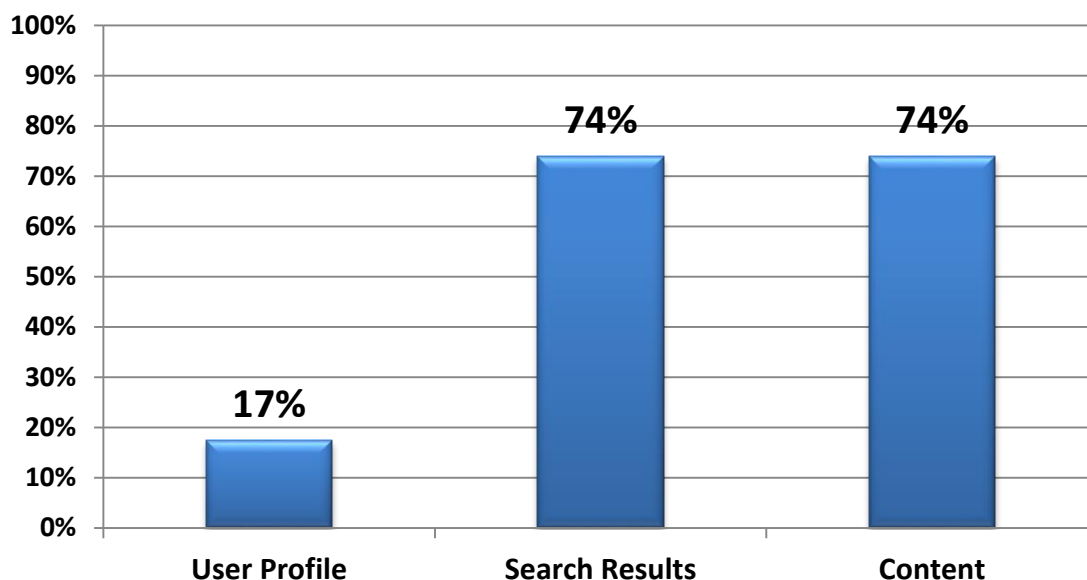


Figura 35 - Respuesta Pregunta 10

Tomando en consideración estas 6 preguntas obligatorias, se puede concluir que, en líneas generales, la satisfacción de los usuarios con el Recomendador es mayor que con el Buscador.

Observando las gráficas con más detalle vemos que, de acuerdo a las opiniones de los usuarios, el sistema con Recomendación ha resultado más útil para llevar a cabo las tareas. En la Figura 31 se puede ver que ningún usuario considera que la utilidad del Buscador sea Muy Alta, mientras que un 35% de los usuarios han considerado como Muy Alta la utilidad del Recomendador (ver Figura 33). Por otro lado mientras que un 13% de usuarios han afirmado que la utilidad del Buscador es Baja o Muy Baja, tan solo un 4% de usuarios han considerado como Baja la utilidad del Recomendador, y a ninguno de ellos le ha resultado Muy Baja.

La mayor utilidad del Recomendador frente al Buscador también es reflejada en la Figura 34. De acuerdo a esta Figura para un 83% de los usuarios los contenidos recomendados por el Recomendador le han resultado de ayuda, a la hora de completar las tareas. De los tres tipos de recomendaciones que se ofrecían a los usuarios: teniendo en cuenta el perfil del usuario, la búsqueda realizada o las noticias consultadas. Las más útiles para los usuarios son estas dos últimas; para ambas un 74% de los usuarios las han elegido como las más útiles.

Para la recomendación teniendo en cuenta al perfil del usuario sólo un 17% de los usuarios las han considerado como útiles. Esto es debido sin duda al tipo de evaluación que se le plantea al usuario. En esta evaluación se le obliga al usuario a buscar para cada tarea nuevos temas, lo que hace que el perfil generado teniendo en cuenta las interacciones previas no será de mucha utilidad. En un contexto de uso normal, donde el usuario solo busque temas relacionados con sus intereses, este tipo de recomendación sería mucho más útil.

Además en estas preguntas se ha consultado a los usuarios por otro aspecto, la complejidad de uso de ambos sistemas. Es posible que dado que el recomendador ofrece muchas más cosas que un sistema de búsqueda tradicional (a.k.a. *Google like*) los usuarios puedan verse abrumados, lo que afectaría a la satisfacción de los usuarios. A este respecto, el 91% de los usuarios considera el Buscador como Sencillo o Muy Sencillo de utilizar (ver Figura 30) frente al 82% de los usuarios que consideran Sencillo o Muy Sencillo el Recomendador. Si bien es cierto que, según los datos, los usuarios parece que consideren más sencillo de utilizar el Buscador, la diferencia respecto al Recomendador no es excesivamente grande y queda sobradamente compensada por la mayor utilidad del Recomendador frente al Buscador.

Teniendo en cuenta el otro tipo de preguntas, los resultados de las 4 preguntas opcionales se muestran a continuación. Hay que tener en cuenta que al no ser obligatorias, no todos los usuarios han contestado a las mismas.

En primer lugar, en la Tabla 16 y Tabla 17 se muestran las respuestas relativas al Buscador. Respecto a lo que más le ha gustado a los usuarios de este sistema (Tabla 16) lo que más destacan es la sencillez de uso y la rapidez del mismo y en menor medida la adaptación de los resultados a la búsqueda realizada. Centrándose en lo que menos les ha gustado a los usuarios (Tabla 17) no destaca ningún aspecto en concreto. Las quejas de los usuarios van desde aspectos relacionados con la usabilidad y la estética de la aplicación, la abundancia de información, la falta de configurabilidad o la dificultad de completar las tareas con este buscador.

¿Qué es lo que más le ha gustado del Sistema de Búsqueda SIN Recomendación?
Sencillez de uso
es muy rápido
utilidad
Es sencillo y permite evaluar.
la practicidad y sencillez
La facilidad para encontrar resultados
La facilidad de encontrar la información y temas relacionados
es rápido y sencillo de utilizar
La apariencia de las noticias es sencilla y se aprecian a golpe de vista.
La sencillez
Muy sencillo e intuitivo. Búsquedas certeras.
la sencillez y la posibilidad de explorar las noticias
La sencillez y que los primeros resultados estaban relacionados con mi búsqueda

Tabla 16 - Respuesta Pregunta 3

¿Y lo qué menos
Es complejo completar alguna de las tareas
espacio de lectura de la noticia pequeño
complejidad
No se pueden cambiar las evaluaciones.
aparecen noticias repetidas, y en algunas ocasiones el texto de dentro de la noticia no es completo
todo bien
Los colores y la estética de la página en general
Demasiada abundancia de información no relacionada con lo que se pide.
La configurabilidad
Al tratar de modificar una búsqueda en el campo de texto (por ejemplo para añadir un término más) no resulta sencillo. Aparecen seleccionadas automáticamente todos los términos, al menos desde el Firefox.

Tabla 17 - Respuesta Pregunta 4

Atendiendo a las respuestas relativas al Recomendador (Tabla 18 y Tabla 19), los usuarios destacan como aspecto positivo, casi de forma monográfica, la utilidad de las recomendaciones, ya sean mediante noticias o mediante etiquetas. Además reflejan otros aspectos como la sencillez de uso o lo certero de las búsquedas (ver Tabla 18). De nuevo, al igual que ocurre con el Buscador, las opiniones respecto a los aspectos negativos del recomendador (Tabla 19) no reflejan ninguna temática principal. En su lugar se citan temas como la complejidad, algún aspecto de usabilidad, o algún problema relacionado con las recomendaciones.

¿Qué es lo que más le ha gustado del Sistema de Búsqueda CON Recomendación?
Recomendaciones propuestas
es rápido y con muchas opciones
ayuda
Su facilidad de uso y sus buenas recomendaciones. Es muy intuitivo y sus recomendaciones facilitan la búsqueda.
Permite nuevas búsquedas sin necesidad de teclear, porque hay muchas recomendaciones que están relacionadas con los primeros criterios de búsqueda.
recomienda tags relacionados
Interesantes los resultados que ofrece
Las palabras (enlaces) que te sugiere para concretar más la búsqueda cuando es un tema en general
es rápido y sencillo como el anterior, las recomendaciones le otorgan un valor añadido, es más útil y completo
Su sencillez de búsqueda y lo concreto de los resultados.
Las noticias relacionadas, que ayudan a la búsqueda
Muy sencillo e intuitivo. Búsquedas certeras.
la opción de buscar por términos
No era necesario navegar por todas las páginas de resultados, ya que era posible hacer la navegación mediante las noticias relacionadas. Además te sugerían términos para afinar la búsqueda

Tabla 18 - Respuesta Pregunta 7

¿Y lo que menos?
espacio de lectura de la noticia pequeño
complejidad
No se pueden cambiar las evaluaciones.
Las propuestas si son acertadas es correcto, pero en el momento de seleccionar una, si esta selección no es fina, las nuevas recomendaciones te pierden el foco de tu búsqueda inicial. Se debería poder volver a la recomendación anterior.
se basa demasiado en las noticias exploradas más que en el tema
Al aparecer arriba, tienes que bajar para hacer la búsqueda y parece menos intuitivo
Nada.
Los conceptos relacionados muchas veces no son útiles para la búsqueda que se desea hacer
Al tratar de modificar una búsqueda en el campo de texto (por ejemplo para añadir un término más) no resulta sencillo. Aparecen seleccionadas automáticamente todos los términos, al menos desde el Firefox.
las noticias relacionadas con mis búsquedas no cambian mucho

Tabla 19 - Respuesta Pregunta 8

Capítulo 9. Conclusiones y Trabajo Futuro

En este trabajo se ha planteado una aproximación de modelado de contenidos y perfiles de usuario en un sistema de recomendación basado en contenido. Para ello, en este trabajo se ha propuesto un método de modelado que identifica aquellos términos de los contenidos que más los diferencian del resto de contenidos; esto es, los términos más originales. Para ello, como método de pesado se ha utilizado una técnica basada en divergencias del lenguaje: Kullback Leibler Divergence (KLD).

Serán estos modelos de los contenidos los que son utilizados posteriormente para encontrar y recomendar otros nuevos contenidos. Varias propuestas de recomendación, que hacen uso de este modelado, han sido también planteadas de cara a cubrir diferentes entornos en los que la recomendación de contenidos es susceptible de ser útil.

Para comprobar la validez del trabajo de modelización de contenidos propuesto se ha ejecutado una evaluación, siguiendo un enfoque experimental. Mediante esta evaluación, los modelos generados han sido comparados con un *ground truth* de referencia; el cual ha sido generado utilizando las anotaciones sociales que, sobre los contenidos, han establecido los usuarios de *Delicious*. Para medir los resultados obtenidos se han planteado dos marcos de evaluación. El primero de ellos hace uso de una medida conocida como *Match@K*, utilizada previamente para este tipo de evaluaciones. Sin embargo, se vio que esta métrica podía no capturar la representatividad real de los modelos generados; por ello, se planteó ampliar la metodología de evaluación, habiéndose definido una nueva métrica, basada en la métrica nDCG traída del campo de la RI, que se ha denominado como Representatividad Normalizada (*nRep*). Mediante esta métrica, dado un modelo de contenidos compuesto por un conjunto de terminología representativa del mismo, es posible establecer la representatividad del mismo, de acuerdo al *ground truth* utilizado.

Sobre la base de esta metodología de evaluación, la evaluación de las categorías mediante *Match@K* y *nRep* no permite establecer ninguna conclusión relevante más allá del mal funcionamiento de TF-IDF, a pesar de que los modelos KLD en general ofrecen mejores resultados, debido al reducido número de muestras (17) categorías. En cuanto a la evaluación de las subcategorías, destaca que la aproximación de modelado propuesta se ha demostrado mejor, tanto en términos de *Match@K* como de *nRep*, que otras técnicas de representación de información, comunes en el estado del arte: TF-IDF, *Mutual Information* y χ^2 , para prácticamente la totalidad de los resultados.

Los resultados de esta evaluación confirman como muy satisfactorio el funcionamiento de la aproximación propuesta en términos de representatividad. Es especialmente significativo destacar el mal funcionamiento de TF-IDF en este sentido ya que es la técnica que con más frecuencia se utiliza para representar contenidos en el ámbito de los SR

Para la evaluación de las aproximaciones recomendación, se ha llevado a cabo una evaluación centrada en el usuario. En esta evaluación se les ha facilitado a los usuarios el acceso a un prototipo de búsqueda y consulta de noticias incluyendo recomendación de contenidos. Tras la realización de un conjunto de tareas predefinidas, se les ha pedido a los usuarios que establezcan la utilidad del sistema desarrollado frente a un sistema de búsqueda de noticias tradicional. La evaluación desarrollada muestra que, tanto utilizando la información recopilada de manera implícita (a través de los logs de usuario) como la recopilada de manera explícita (a través de un consultorio posterior a ésta), la satisfacción de los usuarios con la utilización del sistema de recomendación es mucho mayor que con la utilización de un sistema de búsqueda tradicional.

Estos resultados parecen confirmar, por un lado, la validez del modelado propuesto a la hora de representar efectivamente a los contenidos y, por otro lado, la utilidad del sistema de recomendación propuesto, basado en una aproximación de Recomendación de Información.

El trabajo realizado deja varias líneas abiertas, que no han sido totalmente cubiertas, y otras líneas futuras susceptibles de ser investigadas. En primer lugar, el principal inconveniente que se ha tenido en este trabajo ha sido la falta de una colección de experimentación apropiada. En este trabajo se ha utilizado una colección la cual no está especialmente enfocada en recomendación (carece de interacciones de usuarios). A este respecto, sería deseable con una colección especialmente enfocada a recomendación. De esta manera se podría abordar otra de las líneas abiertas en este trabajo: la realización de una evaluación de laboratorio de las aproximaciones de recomendación. Actualmente la recomendación se ha evaluado únicamente mediante una evaluación centrada en el usuario, la cual, a pesar de obtener buenos resultados, debería ser complementada con la experimentación de laboratorio. En caso de no tener acceso a una colección que cubra todos los requisitos necesarios sería interesante estudiar la creación de una colección propia.

En cuanto a la evaluación de la parte de modelado, se ha comentado la existencia de un gap entre el *ground truth* generado y la colección de noticias; debido a la fecha de las noticias. Con respecto a esto se podría aplicar el mismo modelado sobre una colección más actual y ver como varían los resultados. Previsiblemente, en este caso los resultados deberían ser incluso mejores dada la reducción del gap temporal. Por otro lado, en cuanto a la evaluación de las aproximaciones de recomendación sería interesante llevar a cabo la comparación de las aproximaciones propuestas no solo con un sistema de

búsqueda sino también con otras aproximaciones de recomendación, presentes en el estado del arte.

En cuanto a las posibles líneas de investigación a las que puede dar lugar este trabajo, se pueden incluir todas ellas en una idea común: el enriquecimiento de los perfiles de los usuarios. En este trabajo, el modelado se ha aplicado únicamente sobre la información que se tenía de los usuarios dentro del propio sistema. Sin embargo, nada en el modelado propuesto impediría agregar información de otras fuentes, de datos de cara a ampliar la información de los usuarios.

Los métodos de enriquecimiento de los perfiles de usuario son muy variados y a lo largo del estado del arte se han propuesto muchas y muy variadas ideas. De todas ellas, la primera que podría ser llevada a cabo en este trabajo es la inclusión de información contextual sobre los usuarios (contexto espacial, temporal...). Es de suponer que las preferencias de los usuarios no siempre serán las mismas, dependiendo por ejemplo si está en casa o en el trabajo o de la hora del día. Este tipo de información sería fácilmente aplicable al modelado propuesto y a buen seguro ofrecería interesantes resultados de cara a mejorar la satisfacción del usuario.

Otro tipo de enriquecimiento de perfiles consiste en la agregación de información de diferentes plataformas en un modelo de usuario único. Actualmente solo se tiene la información del usuario dentro del sistema; sin embargo, si el usuario estuviera dispuesto se podría recolectar información suya en otras plataformas y así complementar el conocimiento que se tiene sobre él; incluyendo entre este tipo de plataformas las redes sociales.

Por último, puesto que el perfil del usuario es al fin y al cabo un texto, la aplicación de técnicas de procesamiento textual ayudaría a aumentar la información que se tiene sobre el usuario, así como facilitar la adquisición de conocimiento del mismo. Algunas de estas técnicas serían: Etiquetado POS, Expansión Semántica de la Información o Etiquetado morfológico de la Información.

Anexos

Anexo A: Instalación y Configuración de Herramientas

A.1: Apache Cassandra

Apache Cassandra es un proyecto para desarrollar un sistema de Base de Datos escalable, distribuida y de rápido acceso que se adapte mejor a aplicaciones actuales. La página web oficial del proyecto *Apache Cassandra* es:

<http://cassandra.apache.org/>

En esta página está disponible información acerca del proyecto y se pueden acceder a manuales, tutoriales, añadidos y descargas sobre *Cassandra*.

Cassandra ha sido desarrollada en un paquete auto contenido, para que no sea necesario ningún tipo de instalación para su funcionamiento. Basta con descomprimir el archivo descargado y ejecutar el script de arranque correspondiente al sistema operativo utilizado (estos scripts se encuentran en la caperta /bin del archivo una vez descargado).

No obstante antes de poder ejecutar la base de datos es necesario realizar unas operaciones de configuración previas. Estas operaciones están enfocadas a definir donde se va a desplegar *Cassandra* y en que directorios están los documentos necesarios. También se pueden configurar otras opciones para definir cómo se va a realizar el almacenado, frecuencia de backups, métodos de replicación de información entre nodos, etc... Para más información acerca de cómo configurar *Apache Cassandra* se puede consultar la documentación oficial presentada en:

<http://wiki.apache.org/Cassandra/GettingStarted> (principalmente la sección 2).

En nuestra instalación básica la base de datos corre sobre un solo nodo. Si se quieren aprovechar las ventajas de una base de datos distribuida consultar la página de configuración anterior para desplegar *Cassandra* en varios nodos.

Una vez finalizada la configuración *Cassandra* ya está lista para ser ejecutada y comenzar a almacenar información.

A.2: Solr

La última versión de *Solr* se puede descargar desde su página web:

<http://www.apache.org/dyn/closer.cgi/lucene/solr/>

Anexo B: Manual de Usuario del Recomendador

B.1: Introducción

Este documento pretende servir como manual para la utilización del prototipo desarrollado para la evaluación centrada en el usuario del activo de recomendación. Este activo dispone de dos interfaces, las cuales serán presentadas en este documento:

- Interfaz de Búsqueda (Buscador)
- Interfaz de Recomendación (Recomendador)

La primera de ellas consiste en una interfaz de recuperación de información básica, la cual le servirá al usuario como punto de comparación para la segunda de las interfaces, la interfaz de recomendación. Esta última ofrece al usuario, además de la funcionalidad de búsqueda, ofrece al usuario una serie de recomendaciones en función de su interacción con el sistema. Para la evaluación del prototipo se dispone de una colección de más de 5000 noticias de diversas temáticas.

El prototipo se encuentra disponible en la siguiente dirección:

<http://albali.lsi.uned.es/Recomendador>

B.2: Descripción de la Evaluación

La tarea de evaluación propuesta se basa en la búsqueda, por parte del usuario, de información relevante para un conjunto de consultas propuestas. Se han propuesto un total de 10 consultas, cubriendo diferentes necesidades de información que un usuario puede tener.

Para llevar a cabo la tarea, el usuario deberá utilizar la Interfaz de Búsqueda obligatoriamente para 3 consultas, la Interfaz de Recomendación para otras 3, también obligatoriamente, y podrá escoger entre una u otra para las 4 consultas restantes.

Una vez finalizada la evaluación, se solicitará al usuario la realización de un breve cuestionario donde indicará su opinión sobre su experiencia en la utilización del prototipo.

B.3: Utilización del Sistema

Alta de Usuario y Login

A fin de tener identificado al usuario durante la evaluación será necesario “logearse” en el sistema antes de proceder a su utilización. Aquellos nuevos usuarios que no estén dados de alta en la aplicación necesitarán registrarse en el sistema.

En la Figura 36 se puede ver la pantalla principal del prototipo. Para registrarse en el sistema, si no se está dado de alta en el sistema, el usuario deberá presionar el botón de registro:



“ Un buscador multimedia, multilingüe y multidominio ”

Sistema de Recomendación

Introduza su nombre de usuario y contraseña

User Name

Password

Login

Si no dispone de usuario puede Registrarse aquí

Registro



Natural Language Processing and
Information Retrieval Group at UNED

mlp.uned.es

Figura 36 - Pantalla de Acceso al Sistema

Una vez hecho esto, se accederá a la pantalla de registro (ver Figura 37). Esta pantalla es muy sencilla, únicamente le pedirá al usuario un nombre con el que acceder al prototipo y una contraseña. Una vez introducidos y dado al botón del registro, se le mostrará al usuario un mensaje confirmando su registro en el sistema (ver Figura 38). Con el alta realizada correctamente, el usuario ya podrá volver a la pantalla de Login, mediante el botón de Volver.



Figura 37 - Pantalla de Registro



Usuario: usuario_prueba_4 registrado correctamente

Figura 38 - Usuario registrado correctamente

Una vez en la pantalla de Login de nuevo, el usuario debe introducir los datos con los que se ha dado de alta, accediendo de esta manera al sistema (ver Figura 39)

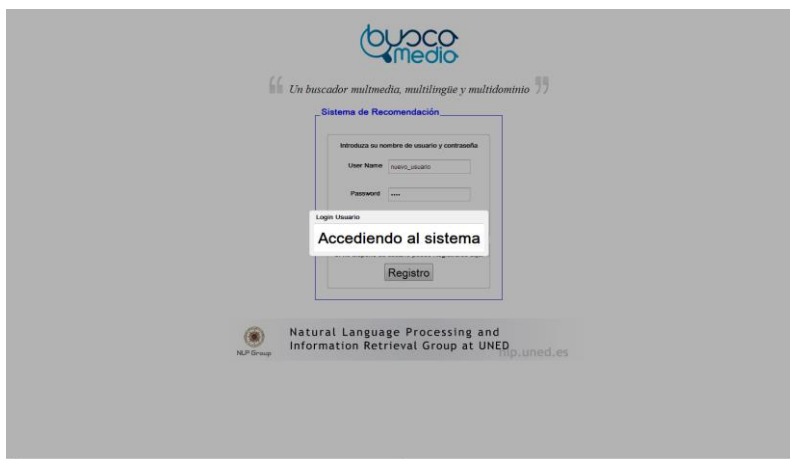


Figura 39 - Acceso al sistema

En todo momento el usuario podrá salir del sistema mediante el botón de Salir, mostrado en la Figura 40. En caso de que el usuario decida salir del sistema, tendría que volver a “loguearse” para poder finalizar la evaluación.



Figura 40 - Botón de Salir

Es **IMPORTANTE** resaltar que para navegar entre las diferentes pantallas del prototipo **NO SE DEBE UTILIZAR LOS BOTONES DE NAVEGACIÓN** (esto es flechas de atrás, adelante y actualizar del navegador), si no los botones que la interfaz ofrece para tal fin. En caso de utilizar los botones de navegación, la aplicación ofrecerá el aviso mostrado en la Figura 41.

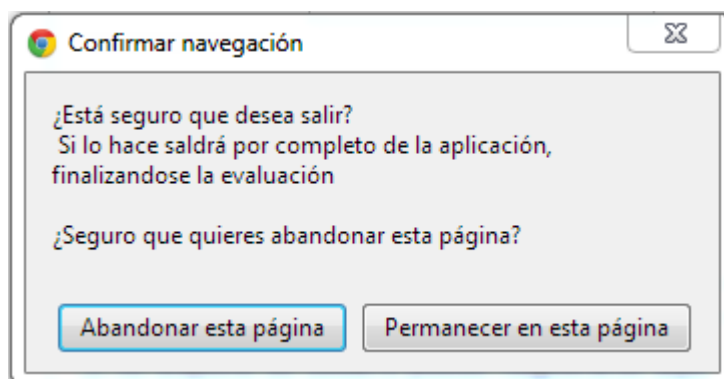


Figura 41 - Ventana de Confirmación

Si el usuario decide abandonar la página, el usuario saldrá de nuevo hasta la página del Login y tendrá que volver a “loguearse” para completar la evaluación.

Presentación de las tareas

La primera pantalla que verá el usuario cuando acceda al sistema le mostrará una recopilación de las tareas que debe llevar a cabo (ver Figura 42). El acceso a cada una de las tareas se realizará mediante los iconos (flecha azul para el Recomendador y flecha naranja para el Buscador) mostrados en esta página.



Figura 42 - Interfaz de acceso a las tareas

Dependiendo de la tarea, al usuario se le ofrece la posibilidad de utilizar el recomendador, utilizar el buscador o utilizar el que él mismo elija. Como se puede ver en la Figura 42, en este caso el usuario solo podrá utilizar el Buscador para completar las tres primeras tareas, el Recomendador para las 3 siguientes y podrá elegir entre ambos para realizar las 4 últimas. En el caso de estas cuatro últimas tareas el usuario deberá elegir aquella interfaz que más útil le haya resultado en las tareas anteriores.

Realización de la tarea

Una vez seleccionada la tarea que se quiere realizar se accederá a una de las dos interfaces, en función de la opción elegida (Recomendador o Buscador). Las Figura 43 y Figura 44 muestran respectivamente la Interfaz del Buscador y la Interfaz del Recomendador, cuando el usuario accede por primera vez al sistema.



Figura 43 - Interfaz del Recomendador (Primer Uso)



Figura 44 - Interfaz del Buscador

Como se puede ver, la principal diferencia entre ambas interfaces es la existencia de una parte dedicada a las preferencias del usuario en la interfaz de Recomendador. En esta parte se le mostrarán al usuario los términos más importantes en las noticias que ha consultado y recomendaciones de noticias referentes a estos términos. Sin embargo, en el primer acceso del usuario al sistema aún no se dispondrá de información suficiente para llevar a cabo la recomendación. Una vez que el usuario haya realizado interacciones suficientes, el Recomendador le comenzará a ofrecer contenidos (ver Figura 45).

Preferencias del Usuario

madrid
 as
 partido
 informe
 lesion
 ala
 campaña
 albiol
 aguirre escudero pp

Público.es - Aguirre presume de contratos transparentes
 Zapatero y Rajoy debatirán en televisión la lucha contra ETA
 Público.es - La 'Gürtel' organizó todos los actos de Aguirre en la campaña de 2003
 Público.es - Aguirre se desmarca de la 'Gürtel' y señala otra vez a Escudero
 El PP cierra filas con Aguirre ante el reto del 22-M - ABC.es
 El juez del 'caso Gürtel' reclama a 79 bancos información de 12 ex altos cargos del PP
 Aguirre disparó un 9% el gasto de las cuentas "austeras" de 2009
 Público.es - 'Lady Aguirre' falta al té de las tres
 Aguirre 'planta' a jóvenes conservadores que la invitaron a tomar el té en Londres | Madrid | elmundo.es
 Público.es - Granados se defiende: "No entiendo de contabilidad"

Búsqueda de Nuevas Noticias

Introduzca su búsqueda

Buscar



Figura 45 - Interfaz de Recomendación

El tiempo máximo para la realización de cada tarea es de 5 minutos; una vez concluido este tiempo la tarea se dará automáticamente por finalizada. Un contador con el tiempo que ha transcurrido desde el principio de cada tarea se puede consultar en la parte superior de la pantalla. No obstante, en cualquier momento el usuario puede dar por finalizada la tarea cuando considere que ya tiene suficiente información sobre la misma. Para ello en la parte superior de la pantalla se muestra un botón de finalizar tarea, el cuál devolverá al usuario a la pantalla con la lista de tareas a realizar para continuar con la siguiente (ver Figura 46).

Tareas

CONSULTAS	Recomendador	Buscador	Completada
Tarea 1: Busque recursos acerca de las manifestaciones producidas en Francia por las reformas de Sarkozy	➡		
Tarea 2: Encuentre recursos relacionados con declaraciones de Esperanza Aguirre	➡		
Tarea 3: Encuentre recursos acerca de las películas del festival de Sitges 2010	➡		
Tarea 4: Encuentre recursos que tengan que ver con la Liga de Campeones		➡	✓
Tarea 5: Busque recursos que traten el tema de la Burbuja Inmobiliaria		➡	
Tarea 6: Busque recursos que traten el tema de la crisis financiera		➡	
Tarea 7: Busque recursos útiles para encontrar partidos ganados por el FC Barcelona	➡	➡	
Tarea 8: Busque recursos útiles para identificar nombres de Jueces de la Audiencia Nacional	➡	➡	
Tarea 9: Busque recursos útiles para identificar nombres de Ministros, o Ex-Ministros españoles	➡	➡	
Tarea 10: Texto de la tarea	➡	➡	

Natural Language Processing and Information Retrieval Group at UNED
<http://ip.uned.es>

Figura 46 - Interfaz una vez completada la tarea

Es importante destacar que cuando el usuario considere que ha terminado con una tarea debe **utilizar el botón de finalizar tarea** y **NO EL BOTÓN DE SALIR**, ya que en caso contrario el usuario saldrá del sistema, teniendo que “loguearse” de nuevo, y la tarea quedará marcada como NO completa.

Búsqueda de Contenidos

Ambas interfaces (Buscador y Recomendador) ofrecen la posibilidad al usuario de buscar contenidos para completar las tareas. El proceso en ambas interfaces es la misma, el usuario introducirá una consulta y el sistema le devolverá una lista de resultados relacionados, paginados de 10 en 10.

La diferencia entre ambas interfaces radica en que, mientras en el Buscador únicamente se le mostrará la lista de resultados al usuario, en el Recomendador el sistema también le ofrecerá una recomendación que estará relacionada con la lista de resultados. Esta recomendación consiste en una nube de etiquetas con el conjunto de términos más significativos de la lista de resultados y una lista de noticias relacionadas con esos términos.

Además de la realización de consultas, otra forma que dispone el usuario para buscar información son los términos presentes en las nubes de etiquetas, mostradas en el Recomendador. Si el usuario hace click sobre cualquiera de estos términos, el sistema realizará una búsqueda utilizando el término como consulta.

Las interfaces de Búsqueda de Información mediante el Buscador y el Recomendador se muestran en las siguientes figuras (Figura 47 y Figura 48 respectivamente).

The screenshot displays the 'busca medio' search engine interface. At the top left is the logo 'busca medio'. In the top right corner, it shows 'Usuario: usuario_prueba_4' and a 'Salir' button. A task bar at the top center indicates 'Tarea 10: Texto de la tarea', a 'Finalizar Tarea' button, and 'Tiempo Transcurrido: 1:38 Tiempo Máximo: 5 min'. The main search area has a search box containing 'madrid' and a 'Buscar' button. Below the search box, a list of search results is shown, each with a title, source, and a star rating. The results include news about Real Madrid players and team performance from various sources like MARCA.com, ABC.es, and Público.es. At the bottom of the results list, there is a pagination control showing '0 1 2 3 4 5 6 7 8 9'.



Natural Language Processing and
Information Retrieval Group at UNED nlp.uned.es

Figura 47 - Búsqueda de Contenidos mediante el Buscador

Preferencias del Usuario

madrid
 as
 partido
 a
 leccion
 a
 informe
 alla
 campaña
 albiol
 ausente
 aguirre escudero pp

Público.es - Aguirre presume de contratos transparentes
 Zapatero y Rajoy debatirán en televisión la lucha contra ETA
 Público.es - La 'Gürtel' organizó todos los actos de Aguirre en la campaña de 2003
 Público.es - Aguirre se desmarca de la 'Gürtel' y señala otra vez a Escudero
 El PP cierra filas con Aguirre ante el reto del 22-M - ABC.es
 El juez del 'caso Gürtel' reclama a 79 bancos información de 12 ex altos cargos del PP
 Aguirre disparó un 9% el gasto de las cuentas "austeras" de 2009
 Público.es - 'Lady Aguirre' falta al té de las tres
 Aguirre 'planta' a jóvenes conservadores que la invitaron a tomar el té en Londres | Madrid | elmundo.es
 Público.es - Granados se defiende: "No entiendo de contabilidad"

Búsqueda de Nuevas Noticias

madrid

Buscar

- "Expediente X' Velickovic: " Honestamente, no sé por qué no he jugado " - MARCA.com** *****
 Novica Velickovic, el 'Expediente X' de la rotación del Real Madrid, aprovechó la visita del Real Madrid a Grecia para hablar sobre su situación marginal en la rotación de Ettore que el Real Madrid perdió contra el Olympiacos. Velickovic apenas jugó 5 minutos para fallar el único partido.
- Cristiano Ronaldo: "El 75 por ciento clasificación ya está hecha" - 20minutos.es** *****
 El jugador del Real Madrid, mostró su satisfacción al final del partido contra el AC Milan con tranquilidad. Por su parte, el portugués Jose Mourinho, entrenador del Real Madrid se mostró encantado con el Real Madrid. Mourinho dijo: "Fantásticos", reiteró: "Son auténticos jugadores 'top'".
- Este Madrid se gana el respeto - MARCA.com** *****
 Ganó y convenció el Real Madrid, que sabía que en Europa uno se gana el respeto a Amelia. El Madrid, que lo ha ganado todo en la fase de grupos, saca 5 puntos al Milan.
- El Milan ya está en Madrid - ABC.es** *****
 El Milan ya ha aterrizado en Madrid con todas sus estrellas (Robinho, Ibra, Iniesta, Pedro León, Ozil, Khedira y Canales reconocen que han venido al Real Madrid).
- Iker entra en el 'top ten' de la Champions - MARCA.com** *****
 del Real Madrid ha superado a Luis Figo y a Kahn, que se retiraron con 103 partidos en la Liga de Campeones: Raúl González (Real Madrid/Schalke 04) 132 Roberto Carlos (Real Madrid (Monaco)/Arsenal/Barça) 111 Clarence Seedorf (Ajax/Real Madrid/Milan) 111 Paolo Maldini (Milan).
- Aguirre acusa al PSM de financiación ilegal - ABC.es** *****
 La presidenta de la Comunidad de Madrid, Ibañeta, ha acusado al Partido Socialista de Madrid de haber dado a ustedes Caja Madrid en el año 2008? 418. 754, más del doble. Por otra parte, Aguirre ha dicho a la portavoz socialista en la Asamblea de Madrid, Maru Menéndez, que no le acepta.
- La Fiscalía de Valencia denuncia a Cañizares por defraudar a Hacienda - ABC.es** *****
 ex guardameta de Real Madrid y Valencia, por haber defraudado presuntamente a Hacienda también participó Aitor Karanka, ex defensa central de Athletic de Bilbao y Real Madrid.
- Robinho: " Llego a Madrid enchufadísimo " - MARCA.com** *****
 Robinho (Sao Vicente, 1984) no tenía intención de dejarse ver antes de volver a Madrid mis fuerzas para salir victorioso del Bernabéu, dijo Robinho. "¿Volver al Real Madrid? La vida Madrid. No te pierdas la entrevista completa a Robinho en tu diario MARCA.
- Público.es - Aguirre se desmarca de la 'Gürtel' y señala otra vez a Escudero** *****
 La presidenta de la Comunidad de Madrid, Esperanza Aguirre, insiste en desvincular su etapa al frente del PP de Madrid de la a pesar de que los últimos informes policiales de su campaña electoral de 2003 los organizó la "¿mantienen en el PP de Madrid

ronaldinho
 partido
 seedorf
 oferta
 mas
 milan
 reconoce
 estudio
 buena
 robinho
 cristiano
 aguirre
 casillas
 ave
 mourinho
 goles
 club
 madrid
 real
 olympiacos
 higuain
 ganar
 ronaldo
 equipo
 bernabeu

Otras noticias relacionadas

- Niños frente a hombres
- Público.es - Aguirre presume de contratos transparentes
- El Bernabéu acoge el primer test serio con el Madrid-Milan, un duelo de gigantes - 20minutos.es
- El Bernabéu acoge el primer test serio con el Madrid-Milan, un duelo de gigantes - 20minutos.es
- Cristiano Ronaldo: "El 75 por ciento clasificación ya está hecha" - 20minutos.es
- Robinho: " Llego a Madrid enchufadísimo " - MARCA.com
- Público.es - Aguirre se desmarca de la 'Gürtel' y señala otra vez a Escudero
- Mourinho: "España no conoce a Ibra"
- Allegri: «Deberemos jugar un partido técnicamente perfecto» - ABC.es
- La exaltación del colectivo en UEFA.COM

0 1 2 3 4 5 6 7 8 9



Figura 48 - Búsqueda de Contenidos mediante el Recomendador

Exploración de Contenidos

Tanto los contenidos devueltos por la Búsqueda de Contenidos como los contenidos recomendados, en el caso de la interfaz del Recomendador, están accesibles para que el usuario los pueda consultar y valorar si son de su interés o no para la tarea dada. La interfaz de Exploración de Contenidos se muestra en la Figura 49 para la Interfaz del Buscador y en la Figura 50 para la Interfaz del Recomendador:



Figura 49 - Exploración de Resultados mediante el Buscador

Mientras que en la Interfaz del Buscador únicamente se muestra al usuario el contenido del recurso seleccionado, en la Interfaz del Recomendador se le ofrece también un conjunto de recomendaciones relacionadas con el recurso. Estas recomendaciones consisten en: tres nubes de etiquetas conteniendo los términos más significativos del documento, la subcategoría del documento y la categoría del documento; y una lista de contenidos relacionados con los términos más significativos de la noticia y del perfil del usuario.

El Olympiacos castiga la fragilidad blanca - ABC.es

Si se han cumplido las palabras de Ettore Messina, a estas horas toda Europa sabrá ya que a su equipo le queda un mundo para poder aspirar a la Final Four. Y es que, según el técnico del Real Madrid, la mayor parte del continente iba a estar pendiente del choque entre su equipo y el Olympiacos, el que daba el pistoletazo oficial a una nueva edición de la Euroliga, y en el que se volvió a demostrar que al conjunto blanco le vienen grande este tipo de partidos. Superado en la zona de manera casi lastimosa, el Madrid aguantó el tipo gracias a su acierto exterior y al seso que Teodosic y Spanoulis demostraron durante la primera mitad. La falta de ritmo de Olympiacos — la liga en Grecia sigue parada por una huelga de jugadores — se notó en los primeros minutos, en los que el Madrid llegó a ir mandando por nueve puntos (21-30, min. 15). La entrada de Bourousis, que sólo falló dos lanzamientos en todo el partido, propició la remontada griega, con un parcial de 11-0 que le dio la vuelta al marcador (32-30, min. 16). Hasta ese momento, sólo Llull y Suárez daban muestras de estar dentro del partido, mientras Messina trataba de encontrar en el banquillo alguna solución a la avalancha griega. Probó con Fischer, indolente como siempre, y con Velickovic, que sigue con su crisis de identidad cada vez que se enfunda la elástica blanca. Con ese chollo en la pintura, Ivkovic no dudó en enviar a sus «hombres a la guerra», buscando una y otra vez el choque bajo los aros (51-42, min. 25). Messina colocó al equipo en zona y la apuesta le salió bien. A un minuto del último cuarto, Olympiacos mandaba sólo por uno (57-56) y el triunfo blanco parecía factible, pero fue entonces cuando llegó la enésima desconexión total del conjunto madridista. Un parcial de 15-0 — iniciado con un triple sobre la bocina de Keselj — dispuso cualquier atisbo de esperanza. El camino de la «Novena» empieza mal: Teodosic (15), Spanoulis (16), Nielsen (4), Papanikolaou (2) y Nesterovic (11) -quinteto inicial-; Bourousis (16) -quinteto inicial-; Lucas-Gordon (2), Keselj (3), Mavrokefalidis (9) y Erceg (-). Olympiacos Rodríguez (6), Llull (18), Suárez (12), Tomic (10) y Reyes (9) -quinteto inicial-; Fisher (4), Velickovic (-), Vidal (2), Garbajosa (-) y Tucker (5). Real Madrid: 17-17, 19-18, 26-21 y 20-10. Parciales: Lámónica (ITA), Belosevi (SRB) y Viator (FRA). Eliminado Nielsen por faltas personales por parte del Olympiacos. Árbitros: De la Paz y la Amistad. 9. 200 espectadores. Pabellón

Noticia

velickovic llull messina mina
muestra conjunto acertada suarez
olympiacos clave griego nielsen
bourousis keselj ettoe blanco
parcial equipo momento quinteto
spanoulis daba teodosic superar
fragil

SubCategoría

anoto buss badalona acreedores
estreno concurso euroliga cibona
ángel fisher franquicia griego
barcelona bourousis bryant gasol
ettoe acb cuarto equipo
baloncesto cancha basket barça
electronics

Categoría

anoto agüero arsenal balon
benfica arbitro atletico barça
aficionados bernabeu brasileño

Seleccionado: 0

Noticias Recomendadas

Público.es - 82-86. El Olympiacos vence al Real Madrid con un contundente último cuarto

'Tenemos todo para competir con el Olympiacos' | Baloncesto | elmundo.es

Olympiacos cuestiona el nivel del Real Madrid para optar a la Euroliga - MARCA.com

"Cuando juego, no pienso, todo sale solo"

El Olympiacos castiga la fragilidad blanca - ABC.es

Real Madrid y Olympiacos levantan el telón y reeditan el partido con el que se estrenó la Euroliga hace una década - MARCA.com

Demasiado Olympiacos para el Madrid

El Real Madrid se estrena con derrota en la Euroliga en casa del Olympiacos - 20minutos.es

Público.es - Bourousis retrata al Madrid

El Real Madrid se estrena con derrota en la Euroliga en casa del Olympiacos - 20minutos.es

Volver

Figura 50 - Exploración de Resultados mediante el Recomendador

Puntuación de un Contenido

Además de consultar los contenidos, el prototipo también permite al usuario puntuarlos en función de lo útiles que hayan sido para realizar la tarea. La puntuación de un recurso puede realizarse de dos maneras. La primera de ellas es en la pantalla de exploración de contenidos; en estas pantallas, debajo del contenido de la noticia, el usuario tiene un icono para puntuar el recurso de 0 a 5 (ver Figura 51 y Figura 52).

Seleccionado: 0

Figura 51 - Vista antes de realizar una puntuación



Has Seleccionado: 4

Figura 52 - Vista después de realizar una puntuación

Además de en la pantalla de exploración, el usuario también podrá puntuar los contenidos desde la pantalla de búsqueda de contenidos. En esta pantalla, junto a cada documento devuelto por una búsqueda, existe un icono que permite puntuar el recurso en función de su título y los *snippets* mostrados (ver Figura 53).



Figura 53 - Puntuación de Recursos en la Interfaz de Búsqueda

Finalizar Tarea

Como ya se ha dicho anteriormente, en ambas interfaces de Búsqueda de Contenidos existe un botón para finalizar la tarea cuando el usuario considere. Una vez que el usuario de por finalizada la tarea, el sistema le devolverá a la pantalla de selección de tarea (ver Figura 46). En esta pantalla, la tarea que acaba de ser realizada por el usuario aparecerá como completada y se le permitirá al usuario continuar con las siguientes tareas.

Finalización de la Evaluación

Una vez que el usuario haya considerado que ha finalizado todas las tareas, el prototipo le solicitará la realización de un breve cuestionario acerca de la satisfacción del usuario en la utilización del sistema (ver Figura 54). El cuestionario a realizar por el usuario se muestra en la Figura 55

busco
media

“ Un buscador multimedia, multilingüe y multidominio ”

Tareas

CONSULTAS	Recomendador	Buscador	Completada
Tarea 1: Busque recursos acerca de las manifestaciones producidas en Francia por las reformas de Sarkozy	→		✓
Tarea 2: Encuentre recursos relacionados con declaraciones de Esp	→		✓
Tarea 3: Encuentre recursos acerca de las películas del festival de S	→		✓
Tarea 4: Encuentre recursos que tengan que ver con la Liga de Ca		→	✓
Tarea 5: Busque recursos que traten el tema de la Burbuja Inmobiliaria		→	✓
Tarea 6: Busque recursos que traten el tema de la crisis financiera		→	✓
Tarea 7: Busque recursos útiles para encontrar partidos ganados por el FC Barcelona	→	→	✓
Tarea 8: Busque recursos útiles para identificar nombres de Jueces de la Audiencia Nacional	→	→	✓
Tarea 9: Busque recursos útiles para identificar nombres de Ministros, o Ex-Ministros españoles	→	→	✓
Tarea 10: Texto de la tarea	→	→	✓

Finalizar Evaluación

Ha finalizado la evaluación.

Por favor rellene el siguiente cuestionario

Realizar Cuestionario

Figura 54 - Vista de todas las tareas completadas

***1. Valore la sencillez y comprensibilidad del Sistema de Búsqueda SIN Recomendación**

- Muy Complejo
- Complejo
- Sencillo
- Muy Sencillo

***2. Valore la utilidad del Sistema de Búsqueda SIN Recomendación a la hora de llevar a cabo las tareas propuestas**

- Muy Alta
- Alta
- Aceptable
- Baja
- Muy Baja

3. ¿Qué es lo que más le ha gustado del Sistema de Búsqueda SIN Recomendación?

4. ¿Y lo qué menos?

***5. Valore la sencillez y comprensibilidad del Sistema de Búsqueda CON Recomendación**

- Muy Complejo
- Complejo
- Sencillo
- Muy Sencillo

***6. Valor la utilidad del Sistema de Búsqueda CON Recomendación a la hora de llevar a cabo las tareas propuestas**

- Muy Alta
- Alta
- Aceptable
- Baja
- Muy Baja

7. ¿Qué es lo que más le ha gustado del Sistema de Búsqueda CON Recomendación?

8. ¿Y lo qué menos?

***9. ¿Le han resultado de ayuda las Recomendaciones del Sistema de Búsqueda con Recomendación a la hora de llevar a cabo las tareas propuestas?**

- Sí
- No
- No han supuesto una diferencia

***10. ¿Qué tipo de recomendación le ha resultado más útil? (Elija todas las que considere oportuno)**

- Recomendación tomando el perfil del usuario
- Recomendación tomando los resultados de búsqueda
- Recomendación tomando la noticia que se está consultando

Listo

Figura 55 - Cuestionario

Referencias

- [1] Adomavicius, G. Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17 (6), pages 734–749. 2005.
- [2] Adomavicius, G. Tuzhilin, A.: Context-Aware Recommender Systems, pages 217–257, Springer, 2011.
- [3] Adomavicius, G. Zhang, J.: On the Stability of Recommendation Algorithms. In *Proceedings of the 4th ACM Conference on Recommender Systems (RecSys 2010)*. ACM New York, NY, USA, pages 47-54. 2010.
- [4] Adomavicius, G. Zhang, J.: Iterative Smoothing Technique for Improving the Stability of Recommender Systems. In *Proceedings of the Workshop on Recommendation Utility Evaluation (RUE 2012)*, pages 3-8, 2012.
- [5] Agarwal, D. Chen, B.-C. Elango, P.: Spatio-temporal models for estimating click-through rate. In *Proceedings of the 18th international conference on World Wide Web (WWW 2009)*, pages 21-30. 2009.
- [6] Amatriain, X. Castels, P. de Vries, A. Posse, C. Steck, H.: Proceedings of the Workshop on Recommendation Utility Evaluation: Beyond RMSE. In *Proceedings of the 6th ACM International Conference on Recommender Systems (RecSys 2012)*. 2012.
- [7] Antulov-Fantulin, N. Bošnjak, M. Šmuc, T. Jermol, M. Žnidaršič, M. Grčar, M. Keše, P. Lavrač, N.: ECML/PKDD 2011 - Discovery challenge: “VideoLectures.Net Recommender System Challenge”, <http://lis.irb.hr/challenge/>
- [8] Antulov-Fantulin, N. Bošnjak, M. Žnidaršič, M. Grčar, M. Morzy, M. Šmuc, T.: ECML/PKDD 2011 Discovery Challenge Overview, In *Proceedings of ECML-PKDD 2011 Discovery Challenge Workshop*, pages 7-20. 2011
- [9] Balabanovic, M. Shoham, Y.: Fab: Content-based, collaborative recommendation. *Communications of the ACM*, 40 (3), 66-72. 1997.
- [10] Baltrunas, L. Amatriain, X.: Towards time-dependent recommendation based on implicit feedback. In *Proceedings of the 3rd ACM conference on Recommender systems (RecSys 2009)*. ACM, New York, NY, USA, pages 423–424. 2009.
- [11] Barman, K. Dabber, O.: Local popularity based collaborative filters. In *Proceedings of the International Symposium on Information Theory (ISIT '10)*, 1668–1672. 2010.

- [12] Barnett, E.: Barack Obama signs up to *Pinterest*. Creado el 28 de Marzo de 2012. Recuperado el 20 de Abril de 2012. <http://www.telegraph.co.uk/technology/social-media/9170718/Barack-Obama-signs-up-to-Pinterest.html>.
- [13] Basu, C. Hirsh, H. Cohen, W. W.: Recommendation as classification: Using social and content-based information in recommendation. In *Proceedings of the 15th National Conference on Artificial Intelligence*, pages 714–720. 1998.
- [14] Bellogín, A. Cantador, I. Castells, P.: A study of heterogeneity in recommendations for a social music service. In *Proceedings of the 1st International Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec '10)*. ACM, New York, NY, USA, 1–8. 2010.
- [15] Bellogín, A. Cantador, I. Díez, F. Castells, P. Chavarriga, E.: An Empirical Comparison of Social, Collaborative Filtering, and Hybrid Recommenders. *ACM Transactions on Intelligent Systems and Technology, Special Issue on Context-Aware Movie Recommendations*. 2011.
- [16] Belluf, T. Xavier, L. Giglio, R.: Case study on the business value impact of personalized recommendations on a large online retailer. In *Proceedings of the sixth ACM conference on Recommender systems (RecSys 2012)*. ACM, New York, NY, USA, pages 277-280. 2012.
- [17] Ben-Shimon, D. Tsikinovsky, A. Rokach, L. Meisles, A. Shani, G. Naamani, L.: Recommender system from personal social networks. In *Proceedings of the 5th Atlantic Web Intelligence Conference (AWIC 2007)*, 47–55. 2007.
- [18] Berkovsky, S. Kuflik, T. Ricci, F.: Cross-representation mediation of user models. In *User Modeling and User-Adapted Interaction (19)*, pages 35–63. 2009.
- [19] Bettini, C. Brdiczka, O. Henriksen, K. Indulska, J. Nicklas, D. Ranganathan, A. Riboni, D.: A survey of context modeling and reasoning techniques. *Pervasive and Mobile Computing*. 6, pages 161-180. 2010.
- [20] Bigi, B.: Using Kullback-Leibler Distance for Text Categorization. In *Proceedings of the 25th European Conference on Information Retrieval (ECIR 2003)*, Fabrizio Sebastiani (Ed.). Springer-Verlag, Berlin, Heidelberg, pages 305-319. 2003.
- [21] Biswas, P. Robinson, P.: A brief survey on user modelling in HCI. In *Proceedings of the International Conference on Intelligent Human Computer Interaction (IHCI) 2010*. 2010.
- [22] Boim, R. Milo, T. Novgorodov, S.: Diversification and refinement in collaborative filtering recommender. In *CIKM*, pages 739-744. 2011.
- [23] Breese, J. Heckerman, D. Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In *Proceeding of 14th Conference on Uncertainty in Artificial Intelligence*, pages 43–52. 1998.

- [24] Candillier, L. Jack, K. Fessant, F. Meyer, F.: State-of-the-art recommender systems. *Chevalier, M., Julien, C., Soule-Dupuy, C. (eds.) Collaborative and Social Information Retrieval and Access: Techniques for Improved User Modeling*, IGI Global, USA, pages 1-22. 2009.
- [25] Cantador, I. Brusilovsky, P. Kuflik, T.: 2nd Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec 2011). In *Proceedings of the 5th ACM conference on Recommender Systems*. 2011.
- [26] Celma, O. Herrera, P.: A new Approach to evaluating novel recommendations. In *Proceeding of the 2nd ACM Conference on Recommendation Systems (RecSys 2008)*, pages 179-186. 2008.
- [27] CENIT BUSCAMEDIA: CENIT Buscamedia. Recuperado el 11 de Diciembre de 2012. <http://www.cenitbuscamedia.es/>
- [28] Charysam: *Cassandra Data Model*. Creado el 7 de Enero de 2011, Recuperado el 29 de Marzo de 2012. <http://charysam.wordpress.com/2011/01/07/Cassandra-data-model/>
- [29] Chin, D.: Empirical evaluation of user models and user-adapted systems. *User Modeling and User-Adapted Interaction 11*: 181-194. 2001.
- [30] Chin, D. Crosby, M.: Introduction to the Special Issue on Empirical Evaluation of User Models and User Modeling Systems. *User Modeling and User-Adapted Interaction 12*: 105-109. 2002.
- [31] Chu, W. Park, S. Beaupre, T. Motgi, N. Phadke, A. Chakraborty, S. Zachariah, J.: A case study of behavior-driven conjoint analysis on Yahoo!: Front page today module. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1097-1104, 2009.
- [32] Cilibrasi, R. Vitanyi, P.M.B.: The Google similarity distance. *IEEE Transactions on Knowledge and Data Engineering*, 19 (3), pages 370-383. 2007.
- [33] Cleger-Tamayo, S. Fernández-Luna, J.M. Huete, J.F.: On the use of Weighted Mean Absolute Error in Recommender Systems. In *Proceedings of the Workshop on Recommendation Utility Evaluation (RUE 2012)*, pages 24-26. 2012.
- [34] Cleger-Tamayo, S. Fernández-Luna, J.M. Huete, J.F.: A New Criteria for Selecting Neighborhood in Memory-Based Recommender Systems. In *Advances in Artificial Intelligence. Lecture Notes in Computer Science (7023)*, pages 423-432. 2011.
- [35] Constine, J.: *Pinterest Hits 10 Million U.S. Monthly Uniques Faster than any Standalone Site Ever*. Creado el 7 de Febrero de 2012. Recuperado el 20 de Abril de 2012. <http://techcrunch.com/2012/02/07/Pinterest-monthly-uniques/>

- [36] Cramer, H. Evers, V. Ramlal, S. Someren, M. Rutledge, L. Stash, N. Aroyo, L. Wielinga, B.: The effects of transparency on trust in and acceptance of a content-based art recommender. In *User Modeling and User-Adapted Interaction 18 (5)*: 455-496. 2008.
- [37] Cremonesi, P. Garzotto, F. Turrin, R.: User Effort vs. Accuracy in Rating-based Elicitation. In *proceedings of the 6th ACM conference on Recommender systems (RecSys 2012)*. ACM, New York, NY, USA, pages 27-34. 2012.
- [38] Das, A.S. Datar, M. Garg, A. Rajaram, S.: Google news personalization: scalable online collaborative filtering. In *Proceedings of the 16th World Wide Web Conference (WWW '07)*, pages 271-280. 2007
- [39] Deerwester, S. Dumais, S. T. Furnas, G. W. Landauer, T. K. Harshman, R.: Indexing by Latent Semantic Analysis. *Journal of the American Society For Information Science*, 41, pages 391-407. 1990.
- [40] De Luca, E.W. Plumbaum, T. Kunegis, J. Albayrak, S.: Multilingual Ontology-based Profile Enrichment. In *Proceedings of the 1st International Workshop on the Multilingual Semantic Web (MSW 2010)*, vol. 571, pages 41-42. 2010.
- [41] Diaz-Aviles, E. Drumond, L. Schmidt-Thieme, L. Nejdl, W.: Real-Time Top-N Recommendation in Social Streams. In *Proceedings of the 6th ACM conference on Recommender systems (RecSys 2012)*. ACM, New York, NY, USA, pages 59-66. 2012.
- [42] Degemmis, M. Lops, P. Semeraro, G.: A content-collaborative recommender that exploits WordNet-based user profiles for neighborhood formation. *User Modeling and User-Adapted Interaction*, 17, pages 217–255. 2007.
- [43] Deshpande, M. Karypis, G.: Item- based top-N recommendation algorithms. In *ACM Transactions on Information Systems*, 22(1), pages 143–177. 2004.
- [44] Devi, M.K.K. Venkatesh, P.: An Improved Collaborative Recommender System. In *First International Conference on Networks and Communications, 2009. NETCOM '09*. Pages 386-391. 2009.
- [45] Eckart, C. Young, G.: The approximation of one matrix by another of lower rank. *Psychometrika*, 1, pages 211-218. 1936.
- [46] Fischer, G.: User Modeling in Human–Computer Interaction. *User Modeling and User-Adapted Interaction 11, (1-2)*: 65-86. 2001.
- [47] Fleder, D.M. Hosanagar, K.: Recommender systems and their impact on sales diversity. In *EC '07: Proceedings of the 8th ACM Conference on Electronic Commerce*, ACM, New York, USA, pages 192-199. 2007.
- [48] Fröschl, C.: User Modeling and User Profiling in Adaptive E-learning Systems. Graz, Austria: Master Thesis (2005)
- [49] Griffiths, T. Steyvers, M.: Finding scientific topics. In *Proceedings of the National Academy of Sciences*, 101 (suppl. 1), pages 5228-5235. 2004.

- [50] Godoy, D. Schiaffino, S. Amandi, A.: Integrating user modeling approaches into a framework for recommender agents. *Internet Research*. 20, pages 29-54. 2010.
- [51] Gunawardana, J. Shani, G.: A Survey of Accuracy Evaluation Metrics of Recommendation Tasks. *Journal of Machine Learning Research* (10): 2935-2962. 2009.
- [52] Han, E.-H. S. Karypis, G.: Feature-based recommendation system. In *Proceedings of the 14th Conference of Information and Knowledge Management*, pages 446-452. 2005.
- [53] Hatcher, E. Gospodnetic, O. McCandless, M.: *Lucene in Action, 1st edition*, Manning publications Co. 2005.
- [54] Hu, R. Pu, P.: A comparative user study on rating vs. personality quiz based preference elicitation methods. In *Proceedings of the 13th International Conference on Intelligent User Interfaces (IUI '09)*, pages 367-372. 2009.
- [55] Jack, K. Hammerton, J. Harvey, D. Hoyt, J. J. Reichelt, J. Henning, V.: Mendeleev's Reply to the DataTEL Challenge. *Procedia Computer Science*, 1(2), pages 1-3. 2010.
- [56] Halpin, T. A.: *Information Modeling and Relational Databases: From Conceptual Analysis to Logical Design*. Morgan Kaufman. San Francisco, 2001.
- [57] He, L. Zahng, J. Zhuo, L. Shen, L.: Construction of user preference profile in a personalized image retrieval. In *Proceedings of the 10th International Conference on Neural Networks and Signal Processing*, pages 434-439. 2008.
- [58] Herlocker, J. Konstan, J. Riedl, J.: Explaining collaborative filtering recommendations. In *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*, pages 241-250, 2000.
- [59] Herlocker, J. Konstan, J. Riedl, J.: An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. *Information Retrieval*, 5(4), 287-310. 2002.
- [60] Herlocker, J. Konstan, J. Terveen, L. Riedl, J.: Evaluating collaborative filtering recommender systems. In *ACM Transactions on Information Systems*, 22 (1): 5-53. 2004.
- [61] Hofmann, T.: Probabilistic latent semantic indexing. *Proceedings of the 22nd annual international SIGIR Conference on Research and Development in Information Retrieval*, pages 50-57. 1999.
- [62] Iaquinta, L. de Gemmis, M. Lops, P. Semeraro, G. Filannino, M. Molino, P.: Introducing Serendipity in a Content-based Recommender System. In: *F. Xhafa, F. Herrera, A. Abraham, M. Köppen, J.M. Benitez (eds.) Proceedings of the Eighth International Conference on Hybrid Intelligent Systems HIS-2008*, pages 168-173. 2008.

- [63] Jack, K. Duclaye, F.: Etude de la pertinence de critères de recherche en recherche d'informations sur des données structurées. In *PeCUSI, INFORSID*, pages 285-297. 2007.
- [64] Jack, K. Duclaye, F.: Improving Explicit Preference Entry by Visualizing Data Similarities. In *Intelligent User Interfaces, International Workshop on Recommendation and Collaboration (ReColl)*. 2008.
- [65] Jamali, M. Ester, M.: A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the 4th ACM conference on Recommender systems (RecSys 2010)*. ACM, New York, NY, USA, pages 135–142. 2010.
- [66] Jannach, D. Zanker, M. Felfernig, A. Friedrich, G.: Collaborative Recommendation. In *Recommender Systems: An Introduction*. Cambridge University Press, pages 13-50. 2010.
- [67] Jannach, D. Zanker, M. Felfernig, A. Friedrich, G.: *Recommender Systems An Introduction*. Cambridge University Press. 2010.
- [68] Jessenitschnig, M. Zanker, M.: A generic user modeling component for hybrid recommendation strategies. In *Proceedings of the IEEE International Conference on e-Commerce Technology*, pages 337-344. 2009.
- [69] Kelleher, J. Bridge, D.: Rectree centroid : An accurate, scalable collaborative recommender. In *Proceedings of 14th Irish Conference on Artificial Intelligence and Cognitive Science*, pages 89–94. 2003.
- [70] Kim, H.N. Ha, I. Lee, K.S. Jo, G.S. El-Saddik, A.: Collaborative user modeling for enhanced content filtering in recommender systems. In *Decision Support Systems*. (51), pages 772-781 2011.
- [71] Kleinberg, J. Sandler, M.: Using mixture models for collaborative filtering. In *Proceedings of 36th ACM Symposium on Theory of Computing*, pages 569–578. ACM Press. 2004.
- [72] Knijnenburg, B.P. Willemsen, M.C. Gantner, Z. Soncu, H. Newell, C.: Explaining the user experience of recommender systems. In *User Modeling and User-Adapted Interaction* (22):4, pages 441-504. 2012.
- [73] Knijnenburg, B.P. Willemsen, M.C. Kobsa, A.: A Pragmatica Procedure to Support the User-Centric Evaluation of Recommender System. In *Proceedings of the 5th ACM Conference on Recommender Systems (RecSys 2011)*, pages 321-324. 2011.
- [74] Kobsa, A.: Generic User Modeling Systems. *User Modeling and User-Adapted Interaction* 11, (1-2): 49-63. 2001.

- [75] Konstas, I. Stathopoulos, V. Jose, J.M.: On Social Networks and Collaborative Recommendation. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval (SIGIR'09)*. ACM, New York, NY, USA, 195–202. 2009.
- [76] Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: *KDD'08: Proceeding of the 14th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pages 426–434. 2008.
- [77] Koren, Y. Bell, R.: Advances in collaborative filtering. In *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, P.B.Kantor, Eds. Springer US, Boston, MA, pages 145–186. 2011.
- [78] Lakiotaki, K. Matsanis, N. Tsoukiàs, A.: Multicriteria User Modeling in Recommender Systems. In *Intelligent Systems*, 26, pages 64-76. 2011.
- [79] Landauer, T.: On the computational basis of learning and cognition: Arguments from LSA. In Ross, N. (Ed.), *The psychology of learning and motivation*, 41, pages 1-63. 2002.
- [80] Lam, S. K. Frankowski, D. Riedl, J.: Do you trust your recommendations? An exploration of security and privacy issues in recommender systems. In *Proceedings of the International Conference on Emerging Trends in Information and Communication Security*. 2006.
- [81] Lathia, N. Hailes, S. Capra, L. Amatriain, X.: Temporal Diversity in Recommender Systems. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ACM New York, NY, USA, pages 210-217. 2010.
- [82] Lee, D. H. Brusilovsky, P.: Social networks and interest similarity: the case of CiteULike. In *Proceedings of the 21st ACM conference on Hypertext and hypermedia (HT '10)*. ACM, New York, NY, USA. 151–156. 2010.
- [83] Lin, W. Alvarez, S. Ruiz, C.: Efficient adaptive-support association rule mining for recommender systems. In *Data Mining and Knowledge Discovery*, 6 (1), pages 83–105. 2002.
- [84] Linden, G. Smith, B. York, J.: Amazon.com recommendations: Item-to-item collaborative filtering. In *IEEE Internet Computing*, 7(1), pages 76–80. 2003.
- [85] Liu, F. Lee, H.J.: Use of social network information to enhance collaborative filtering performance. *Expert Systems with Applications*, 37(7), 4772–4778. 2010.
- [86] Liu H., Salem B., Rauterberg M.: A survey on user profile modeling for personalized service delivery systems. In *Proceeding of IADIS International Conference on Interfaces and Human Computer Interaction 2009*, pages 45-51. 2009.

- [87] Loops, P. de Gemmis, M. Semeraro, G.: Content-Based Recommender Systems: State of the Art and Trends. In *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, P.B.Kantor, Eds. Springer US, Boston, MA, pages 73-106. 2011.
- [88] Ma, H. King, I. Lyu, M.R.: Learning to recommend with social trust ensemble. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval (SIGIR '09)*. ACM, New York, NY, USA, 203–210. 2009.
- [89] Ma, H. Yang, H. Lyu, M. R. King, I.: SoRec: social recommendation using probabilistic matrix factorization. In *Proceedings of the 17th ACM conference on Information and Knowledge Management (CIKM'08)*. ACM, New York, NY, 931–940. 2008.
- [90] Maes P. 2005. User modeling, Recommender Systems & Personalization. Retrieved from the following website on 9th, November 2007
- [91] Manning, C. Raghavan, P. Schtze, H.: *Introduction to Information Retrieval*, Cambridge University Press, New York. 2008.
- [92] Massa, P. Avesani, P.: Trust Metrics in Recommender Systems. In *Computing with Social Trust*. Golbeck, J, Eds. Springer London, pages 259-285. 2009.
- [93] McNee, S. Riedl, J. Konstan, J.: Being accurate is not enough: How accuracy metrics have hurt recommender systems. In *the Extended Abstracts of the 2006 ACM Conference on Human Factors in Computing Systems*, pages 1097-1101 2006.
- [94] McSherry, F. Mironov, I.: Differentially private recommender systems: building privacy into the Netflix prize contenders. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '09)*, pages 627-636. 2009.
- [95] Meyer, F. Fessant, F. Clérot, F. Gaussier, E.: Toward a New Protocol to Evaluate Recommender Systems. In *Proceedings of the Workshop on Recommendation Utility Evaluation (RUE 2012)*, pages 9-14. 2012.
- [96] Mobasher, B.: Contextual user modeling for recommendation. In *Keynote at the 2nd Workshop on Context-Aware Recommender Systems*. 2010.
- [97] Mobahser, B. Burke, R. Bhaumik, R. Williams, C.: Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness. In *ACM Transactions on Internet Technologies* 7 (4): 23. 2007.
- [98] Murakami, T. Mori, K. Orihara, R.: Metrics for evaluating the serendipity of recommendation lists. In *New Frontiers in Artificial Intelligence*, pages 40-46. 2008.

- [99] Nageswara Rao, K. Talwar, V.G.: Application Domain and Functional Classification of Recommender Systems - A Survey. In *Desidoc Journal of Library & Information Technology*, 28, pages 17-35. 2008.
- [100] Nguyen, A. Denos, N. Berrut, C.: Improving new user recommendations with rule-based induction on cold user data. In *Proceedings of the 2007 ACM conference on Recommender systems*, pages 121-128. 2007.
- [101] Nguyen, Q.N. Ricci, F.: Conversational case-based recommendations exploiting a structured casa model. In *Proceedings of the 9th European Conference on Advances in Case-Based Reasoning (ECCBR 2008)*, pages 400-414. 2008.
- [102] O'Donovan, J. Smyth, B.: Trust in recommender systems. In *Proceedings of the 10th international conference on Intelligent user interfaces (IUI '05)*. ACM, New York, NY, USA, 167-174. 2005.
- [103] Pazzani, M. J.: A framework for collaborative, content-based and demographic filtering. In *Artificial Intelligence Review*, 13 (5-6), pages 393-408. 1999.
- [104] Pazzani, M. Billsus, D.: Content-Based Recommendation Systems. *The Adaptive Web: Methods and Strategies of Web Personalization. Volume 4321 of LNCS*, pages 325-341. 2007.
- [105] Pennock, D. Horvitz, E. Lawrence, S. Giles, C. L.: Collaborative filtering by personality diagnosis: A hybrid memory-and model-based approach. In *Proceedings of 16th Conference on Uncertainty in Artificial Intelligence*, pages 473-480. 2000.
- [106] Perrault, C. R. Allen, J. F. Cohen, P. R.: Speech Acts as a Basis for Understanding Dialogue Coherence. *Department of Computer Science, University of Toronto, Canada, Report 78-5*. 1978
- [107] Pfleeger, S.L. Kitchenham, B.A.: Principles of survey research. *SIGSOFT Software Eng. Notes* 26 (6): 16-18. 2001.
- [108] Plumbaum, T. Lommatzsch, A. De Luca, E. W. Albayrak, S.: SERUM: Collecting Semantic User Behavior for Improved News Recommendations. In *Advances in User Modeling. Lecture Notes in Computer Science (7138)*, pages 176-180. 2012.
- [109] Popescu, A. Grefenstette, G.: Mining social media to create personalized recommendations for tourist visits. In *Proceedings of the 2nd International Conference on Computing for Geospatial Research & Applications (COM.Geo '11)*. ACM, New York, NY, USA, Article 37 , 6 pages. 2011.
- [110] Powers, D.M.: Evaluation: From precision, recall and f-factor to roc, informedness, markedness and correlation. *Technical Report, School of Informatics and Engineering, Flinders University, Adelaide, South Australia*. 2007.

- [111] Pu, P. Chen, L.: A User-Centric Evaluation Framework of Recommender Systems. In *Proceedings of the ACM RecSys 2010 Workshop on User-Centric Evaluation of Recommender Systems and Their Interfaces (UCERSTI)*, pages 14-21. 2010.
- [112] Puerta Melguizo, M.C. Boves, L. Deshpand, A. Ramos, O.M.: A proactive recommendation system for writing: helping without disrupting. In *Proceedings of the 14th European Conference on Cognitive ergonomics (ECCE 2007)*, pages 89-95. 2007.
- [113] Resnick, P. Iacovou, N. Suchak, M. Bergstrom, P. Riedl, J.: GroupLens: An open architecture for collaborative filtering of net news. In *Proceedings of ACM Conference on Computer Supported Cooperative Work*, pages 175-186. 1994.
- [114] Ricci, F. Rokach, L. Shapira, B.: Introduction to Recommender Systems Handbook. In *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, P.B.Kantor, Eds. Springer, US, Boston, MA, pages 1–35. 2011.
- [115] Ricci, F. Rokach, L. Shapira, B. Kantor, P.B. *Recommender Systems Handbook*. Eds. Springer, US, Boston, MA. 2011.
- [116] Rich, E.: User Modeling via Stereotypes. *Cognitive Science* 3: 329-354. 1979.
- [117] Robertson, S.E. Walker, S. Beaulieu, M.M. Gatford, M. Payne, A. Okapi at TREC-4. In *NIST Special Publication 500-236: The Fourth Textual Retrieval Conference (TREC-4)*, pagex 73-96. 1995.
- [118] Sae-Ueng, S. Pinyapong, S. Oginp, A. Kato, T.: Personalized shopping assistance service at ubiquitous shop space. In *Proceedings of the 22nd International Conference con Advanced Information Networking and Applications(AINAW 2008) – Workshops*, pages 838-843. 2008.
- [119] Said, A. Berkovsky, S. De Luca E. W.: Putting Things in Context: Challenge on Context-Aware Movie Recommendation. In *Proceedings of the RecSys'10 Challenge on Context-aware Movie Recommendation (CAMRa'10)*. ACM, New York, NY, USA, 2–6. 2010.
- [120] Said, A. De Luca, E.W. Albayrak, S.: Inferring Contextual User Profiles - Improving Recommender Performance, In *3rd International Workshop on Context-Aware Recommender Systems (CARS 2011)*. In conjunction with the 5th ACM Conference on Recommender Systems (RecSys 2011). 2011.
- [121] Salton, G. Wong, A. Yang, C.S.: A Vector Space Model for Automatic Indexing. In *Communications of the ACM (18)*, pages 613-620. 1975.
- [122] Sarwar, B. M. Karypis, G. Konstan, J. Riedl, J.: Item-based collaborative filtering recommendation algorithms. In *Proceedings of 10th International World Wide Web Conference*, pages 285-295. 2001.

- [123] Schein, A.I. Popescul, A. Ungar, L.H., Pennock, D.M.: Methods and metrics for cold-start recommendations. In *Proceedings of the 25th Annual international ACM SIGIR Conference on Research and Development in information Retrieval, Tampere, Finland. ACM Press, New York*, pages 253-260. 2002.
- [124] Schröder, G. Thiele, M. Lehner, W.: Setting Goals and Choosing Metrics for Recommender System Evaluations. In *Proceedings of the Second Workshop on User-Centric Evaluation of Recommender Systems and Their Interfaces (UCERSTI 2)*. 2011.
- [125] Shani, G. Chickering, D.M. Meek, C.: Mining recommendations from the web. In *Proceedings of the 2nd ACM Conference on Recommender Systems (RecSys 2008)*, pages 35-42. 2008.
- [126] Shani, G. Gunawardana, A.: Evaluating recommendation systems. In *Recommender systems handbook*, , F. Ricci, L. Rokach, B. Shapira, P.B.Kantor, Eds. Springer, US, Boston, MA, pages 257–298. 2011.
- [127] Shepitsen, A. Gemmell, J. Mobasher, B. Burke, R.: Personalized recommendation in social tagging systems using hierarchical clustering. In *Proceedings of the 2nd ACM Conference on Recommender Systems (RecSys 2008)*. ACM, New York, NY, USA, pages 259–266. 2008.
- [128] Shmueli-Scheuer, M. Roitman, H. Carmel, D. Mass, Y. Konopnicki, D.: Extracting User Profiles form Large Scale Data. In *Proceedings of the 2010 Workshop on Massive Data Analytics on the Cloud (MDAC 2010)*, pages 1-6. 2010.
- [129] Sinha, R.R. Swearingen, K.: Comparing recommendations made by online systems and friends. In *Proceedings of DELOS Workshop: Personalization and Recommender Systems in Digital Libraries*. 2001.
- [130] Siskos, Y. Grigoroudis, E. Matsatsinis, N.F.: UTA Methods. In *J. Figueira, S. Greco, M. Ehrgott (Eds.), State of the Art in Multiple Criteria Decision Analysis, Springer, Berlin (2005)*, pages. 297–343 (chapter 8). 2005.
- [131] Spertus, E. Sahami, M. Buyukkokten, O.: Evaluating similarity measures: A large-scale study in the orkut social network. In *Proceedings of 11th International Conference on Knowledge Discovery in Data Mining*, pages 678–684. New York: ACM Press. 2005.
- [132] Su, X. Khoshgoftaar, T.: A survey of collaborative filtering techniques. *Advances in Artificial. Intelligence*. 2009.
- [133] Tan, P. Steinbach, M. Kumar, V.: *Introduction to Data Mining*. Addison-Wesley. 2005.
- [134] Treeratpituk, P. Callan, J.: Automatically labeling hierarchical clusters. In *Proceedings of the 2006 International Conference on Digital Government Research (DG '06)*, pages 167-176. 2006.

- [135] Vargas, S. Castells, P.: Rank and Relevance in Novelty and Diversity Metrics for Recommender Systems. In *Proceedings of the 5th ACM Conference on Recommender Systems (RecSys 2011)*. ACM, New York, NY, USA, pages 109–116. 2011.
- [136] Vargas, S. Castells, P. Vallet, D.: Inter-Oriented Diversity in Recommender Systems. In *Proceedings of the 34th International ACM SIGIR Conference on Research Development in Information Retrieval (SIGIR '11)*, pages 1211-1212. 2011.
- [137] Villena-Román, J. Collada-Pérez, S. Lana-Serrano, S. González-Cristóbal, J. Hybrid Approach Combining Machine Learning and a Rule-Based Expert System for Text Categorization, In *Proceedings of the Twenty-Fourth International Florida Artificial Intelligence Research Society Conference*, 2011
- [138] Viviani, M. Bennani, N. Egyed-Zsigmond, E.: A Survey on User Modeling in Multi-application Environments. In *Third International Conference on Advances in Human-Oriented and Personalized Mechanisms, Technologies and Services*. pages 111-116. 2010.
- [139] Vozalis, M. Margaritis, K. G.: Enhancing collaborative filtering with demographic data: The case of item-based filtering. In *4th International Conference on Intelligent Systems Design and Applications*, pages 361–366. 2004.
- [140] Walter, F. E. Battiston, S. Schweitzer, F.: Personalised and dynamic trust in social networks. In *Proceedings of the 3rd ACM conference on Recommender systems (RecSys 2009)*. ACM, New York, NY, USA, pages 197–204. 2009.
- [141] Wang, J. de Vries, A. P. Reinders, M. J.: Unifying user-based and item-based collaborative filtering approaches by similarity fusion. In *Proceedings of the 29th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 501-508. 2006.
- [142] Wahlster, W. Kobsa, A.: Dialog-based User Models. In *Proceedings of the IEEE*, 74 (7). 1986.
- [143] Willemsen, M. Bollend, D. Ekstrand, M.: UCERTI 2: Second Workshop on User-Centric Evaluation of Recommender Systems and Their Interfaces. In *Proceedings of the 5th ACM Conference on Recommender Systems (RecSys 2011)*, New York, NY, USA, pages 395-396. 2011.
- [144] Yang, W. Hwang,S.: iTravel: A recommender system in mobile peer-to-peer environment. In *Journal of Systems and Software*. 2012.
- [145] Zhang, M.: Enhancing on Recommendation list diversity of recommender systems. In *Proceedings of the 3rd ACM Conference on Recommender Systems (RecSys 2009)*, pages 397-400. 2009

- [146] Zhang, M. Hurley, N.: Avoiding monotony: improving the diversity of recommendation lists. In *Proceedings of the 2008 ACM conference on Recommender systems (RecSys 2008)*, pages 123-130. 2008.
- [147] Zhang, M. Hurley, N.: Novelty and diversity in top-N recommendation – analysis and evaluation. In *ACM Transactions on Internet Technology 10 (4):14*. 2011.
- [148] Zhang, Y. Callan, J. Minka, T.: Novelty and redundancy detection in adaptive filtering. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*. 2002.
- [149] Zhou, T. McNee, S. M. Konstan, J. A. Lausen G.: Improving Recommendation List through Topic Diversification. In *Proceedings of the 14th International Conference on World Wide Web (WWW '05)*, pages 22-32. 2005.
- [150] Ziegler, C.N. McNee, S. Konstan, J. Lausen, G.: Improving recommendation lists through topic diversification. In *Proceedings of the 14th International World Wide Web Conference*, pages 22–32. 2005.