



UNIVERSITAT DE  
BARCELONA

## Integración de técnicas de personalización basadas en el comportamiento colaborativo del ser humano con interfaces de usuario inteligentes

David Contreras Aguilar

**ADVERTIMENT.** La consulta d'aquesta tesi queda condicionada a l'acceptació de les següents condicions d'ús: La difusió d'aquesta tesi per mitjà del servei TDX ([www.tdx.cat](http://www.tdx.cat)) i a través del Dipòsit Digital de la UB ([diposit.ub.edu](http://diposit.ub.edu)) ha estat autoritzada pels titulars dels drets de propietat intel·lectual únicament per a usos privats emmarcats en activitats d'investigació i docència. No s'autoritza la seva reproducció amb finalitats de lucre ni la seva difusió i posada a disposició des d'un lloc aliè al servei TDX ni al Dipòsit Digital de la UB. No s'autoritza la presentació del seu contingut en una finestra o marc aliè a TDX o al Dipòsit Digital de la UB (framing). Aquesta reserva de drets afecta tant al resum de presentació de la tesi com als seus continguts. En la utilització o cita de parts de la tesi és obligat indicar el nom de la persona autora.

**ADVERTENCIA.** La consulta de esta tesis queda condicionada a la aceptación de las siguientes condiciones de uso: La difusión de esta tesis por medio del servicio TDR ([www.tdx.cat](http://www.tdx.cat)) y a través del Repositorio Digital de la UB ([diposit.ub.edu](http://diposit.ub.edu)) ha sido autorizada por los titulares de los derechos de propiedad intelectual únicamente para usos privados enmarcados en actividades de investigación y docencia. No se autoriza su reproducción con finalidades de lucro ni su difusión y puesta a disposición desde un sitio ajeno al servicio TDR o al Repositorio Digital de la UB. No se autoriza la presentación de su contenido en una ventana o marco ajeno a TDR o al Repositorio Digital de la UB (framing). Esta reserva de derechos afecta tanto al resumen de presentación de la tesis como a sus contenidos. En la utilización o cita de partes de la tesis es obligado indicar el nombre de la persona autora.

**WARNING.** On having consulted this thesis you're accepting the following use conditions: Spreading this thesis by the TDX ([www.tdx.cat](http://www.tdx.cat)) service and by the UB Digital Repository ([diposit.ub.edu](http://diposit.ub.edu)) has been authorized by the titular of the intellectual property rights only for private uses placed in investigation and teaching activities. Reproduction with lucrative aims is not authorized nor its spreading and availability from a site foreign to the TDX service or to the UB Digital Repository. Introducing its content in a window or frame foreign to the TDX service or to the UB Digital Repository is not authorized (framing). Those rights affect to the presentation summary of the thesis as well as to its contents. In the using or citation of parts of the thesis it's obliged to indicate the name of the author.

# **Integración de técnicas de personalización basadas en el comportamiento colaborativo del ser humano con interfaces de usuario inteligentes**

DAVID CONTRERAS AGUILAR

Tesis presentada para optar al grado de Doctor por la  
Universitat de Barcelona en el programa de doctorado en  
*Ingeniería y Tecnologías Avanzadas*

bajo la dirección de

**Dra. Maria Salamó Llorente**  
Departamento de Matemáticas e Informática  
Universitat de Barcelona

Tutor: Dr. Oriol Pujol Vila



UNIVERSITAT DE  
BARCELONA

Abril de 2017



## AGRADECIMIENTOS

Quiero expresar mi agradecimiento a todos quienes han contribuido directa o indirectamente en el desarrollo de esta tesis.

A mi familia, por acompañarme en esta aventura que nos ha implicado alejarnos de nuestros seres queridos. A mi amada esposa Yubitza por ser mi compañera de vida y mi soporte incondicional tanto en los buenos como en los malos momentos, sin ti este logro no tendría sentido. A mis hijos David y Constanza por su comprensión, amor y apoyo durante esta etapa de nuestra familia.

A mi directora de tesis, Dra. Maria Salamó por su dedicación, enseñanza, comprensión y apoyo durante todos estos años de trabajo en conjunto. A los miembros del grupo de investigación WAI, en especial a Dra. Inma Rodríguez y Dra. Anna Puig por su colaboración y enseñanza en las actividades de publicación de los artículos científicos que ha generado este trabajo de investigación. A la Dra. Maite López por su apoyo para las asistencias a congresos y recursos adicionales para la realización de la tesis. También quiero agradecer a Pablo Almajano y Tomas Trescak por su apoyo en las labores iniciales de este trabajo, y a John Roberto por el material de apoyo entregado para la escritura de esta tesis.

Esta tesis ha sido financiada por una beca de la Comisión Nacional de Investigación y Tecnológica de Chile, en su programa Becas Chile.

DAVID CONTRERAS AGUILAR

Barcelona, 2017



## RESUMEN

Esta tesis está enfocada en la problemática asociada a la personalización de información para usuarios que interactúan con grandes volúmenes de información. En concreto, la tesis se centra en el estudio de los sistemas de recomendación conversacionales basados en críticas como herramientas para ayudar a los usuarios en la búsqueda de productos o servicios. Los sistemas de recomendación basados en críticas permiten recoger información asociada a las preferencias del usuario a través de un proceso conversacional, en el cual el usuario puede realizar una retroalimentación a las recomendaciones recibidas. La retroalimentación recibida desde el usuario, permite al recomendador adquirir conocimiento acerca de las preferencias que tiene el usuario respecto del producto que desea y generar un perfil asociado al usuario.

Este tipo de técnica de recomendación en forma de diálogo cíclico, que caracteriza a los sistemas de recomendación basados en críticas, ha demostrado resultados satisfactorios en estudios previos orientados principalmente al dominio de comercio electrónico. Sin embargo, cuando un usuario en el mundo real se encuentra en una tienda física, tiene la posibilidad de interactuar y comunicarse con otros compradores. Aún más, hoy en día existen entornos virtuales colaborativos, la mayoría en tres dimensiones, donde los usuarios se pueden comunicar con otras personas y sentirse inmersos en un mundo virtual.

Considerando todo lo anterior, esta tesis se focaliza en aprovechar el comportamiento colaborativo de las personas para mejorar las recomendaciones de productos o servicios. En concreto, el objetivo de la tesis es definir e integrar nuevas técnicas de personalización con interfaces de usuario inteligentes, basadas en el comportamiento colaborativo del ser humano y en la información recogida de las interacciones de las personas tanto con la interfaz como con otros usuarios.

La tesis se divide en dos partes. En la primera parte, describo las nuevas técnicas de recomendación que incorporan información colaborativa. En la segunda parte, presento nuevas interfaces de usuario inteligentes implementadas en diferentes entornos de visualización y que integran las técnicas de recomendación propuestas en esta tesis.

Las nuevas técnicas de recomendación que propongo integran, por un lado, opiniones de los usuarios acerca de los productos, por otro lado, integran recomendaciones históricas de otros usuarios que han utilizado el sistema de recomendación previamente y finalmente integran avances de estudios previos que han demostrado mejorar los procesos de recomendación. Adicionalmente, propongo otras técnicas basadas en información colaborativa instantánea que se puede obtener de parte del sistema de recomendación, como también de otros usuarios que están utilizando el sistema en el mismo momento.

Las interfaces de usuario inteligentes que propongo se basan en entornos de visualización en 2D y 3D. En primer lugar, propongo un *framework* basado en un entorno web en dos dimensiones que permite integrar técnicas basadas en información proveniente de opiniones y recomendaciones históricas de otros usuarios. En segundo lugar, propongo un *framework* basado en un entorno virtual colaborativo en tres dimensiones que permite integrar técnicas basadas en información colaborativa que se genera por la interacción de los usuarios dentro del entorno virtual. Es necesario destacar, que am-

Los *frameworks* son modulares y escalables, permitiendo integrar diferentes algoritmos de recomendación basados en críticas.

Para validar las técnicas de recomendación y las interfaces de usuario propuestas en esta tesis, defino dos metodologías de evaluación. En concreto, para la evaluación de las técnicas de recomendación utilizo un simulador que por medio de un “usuario artificial” permite evaluar el funcionamiento de las técnicas de recomendación simulando varias pruebas con usuarios reales de forma automática. Adicionalmente, para los *frameworks* que implementan las interfaces de usuario inteligentes, he realizado evaluaciones con usuarios reales.

El análisis de los resultados obtenidos mediante las evaluaciones con el simulador y con los usuarios reales, revela que tanto las nuevas técnicas propuestas, como las interfaces de usuario inteligentes obtienen resultados muy satisfactorios, los cuales superan significativamente a los resultados obtenidos con técnicas de recomendación de estudios previos. Es necesario destacar, que todos los resultados han sido corroborados por métodos estadísticos.

De acuerdo a los resultados obtenidos, puedo concluir que la incorporación de información colaborativa en nuevas técnicas de recomendación favorece la eficiencia y eficacia de los sistemas de recomendación basados en críticas. Además, las nuevas y novedosas interfaces de usuario inteligentes propuestas en esta tesis, representan un avance en el desarrollo de interfaces de usuario para sistemas de recomendación basados en críticas. Estas interfaces aparte de favorecer la integración de técnicas de recomendación colaborativa, mejoran la experiencia de los usuarios durante las sesiones de recomendación, lo cual es corroborado por los usuarios que participaron en los experimentos.

## ABSTRACT

This thesis is focused on the problems associated with personalization of information for users who interact with large volumes of information. In particular, the thesis focuses on the study of critique-based recommenders as tools for helping users in the search of products or services. The critique-based recommenders allow to collect information about user's preferences through a conversational process where the user makes critiques over the recommended product. This feedback allows the recommender to acquire knowledge about the user's preferences about the desired product and creates a user profile.

This type of recommendation technique based on a cyclical dialogue, which characterizes critical recommendation systems, has shown satisfactory results in previous studies oriented mainly to the domain of electronic commerce. However, when a real-world user is in a physical store, he has the ability to interact and communicate with other buyers. Moreover, nowadays there are collaborative virtual environments, most in 3D, where users can communicate with other people and feel immersed in a virtual world.

Considering all mentioned above, this thesis focuses on harnessing the collaborative behavior of people to improve recommendations for products or services. Specifically, the objective of the thesis is to define and to integrate new techniques of personalization with intelligent user interfaces, based on the collaborative behavior of the human being and on the information gathered from the interactions of people with the interface and with other users.

The thesis is divided into two parts. In the first part, I describe the new techniques of recommendation that incorporate collaborative information. In the second part, I present new intelligent user interfaces implemented in different visualization environments and that integrate the recommendation techniques proposed in this thesis.

My proposals of the new recommendation techniques integrate, on the one hand, user opinions about the products, on the other hand, integrate historical recommendations of other users that have used the recommendation system previously and finally integrate advances of previous studies that have shown improvements in the recommendation process. Additionally, I propose other techniques based on instant and on-line collaborative information that can be obtained from the recommendation system, as well as from other users that are using the system at the same time.

The intelligent user interfaces that I propose are based on 2D and 3D visualization environments. First, I propose a framework based on a 2D web environment that allows the integration of techniques based on information from historical opinions and recommendations of other users. Second, I propose a framework based on a 3D collaborative virtual environment that allows the integration of techniques based on collaborative information generated by the interaction of users within the virtual environment. It is worth noting, that both frameworks are modular and scalable, allowing to integrate different critique-based recommendation algorithms.

I define two evaluation methodologies to validate the recommendation techniques and user interfaces proposed in this thesis. Specifically, for the evaluation of the recommendation techniques, I run a simulator that uses an "artificial user" that allows the evaluation of the recommendation efficiency in the proposed techniques by simulating various tests with real users au-



tomatically. In addition, I have conducted evaluations with real users for evaluating the frameworks that implement the intelligent user interfaces.

The analysis of the results obtained through the simulator and real user evaluations, shows that both the proposed new techniques and the intelligent user interfaces obtain very satisfactory results, which significantly outperform the results obtained with recommendation techniques from previous studies. It is worth nothing that all the results have been corroborated by statistical methods.

According to the results, I can conclude that the integration of the collaborative information in new recommendation techniques improves the efficiency and efficacy of critique-based recommenders. In addition, the new and novel intelligent user interfaces proposed in this thesis represent an advance in the development of user interfaces for critique-based recommenders. These interfaces, besides facilitate the integration of collaborative recommendation techniques, improve the users' experience during the recommendation sessions, which is corroborated by the users who participated in the experiments.



# ÍNDICE GENERAL

Índice de cuadros	XIII
Índice de figuras	XIV
1 Introducción	1
Resumen . . . . .	1
1.1 Definición del ámbito y del problema . . . . .	1
1.1.1 Ámbito de trabajo . . . . .	2
1.1.2 Resumen de la propuesta . . . . .	3
1.2 Objetivos . . . . .	5
1.3 Estructura de la tesis . . . . .	5
1.4 Conclusiones . . . . .	7
2 Estado del arte: antecedentes y últimos avances	9
Resumen . . . . .	9
2.1 Introducción . . . . .	9
2.2 Sistemas Basados en Críticas: Antecedentes . . . . .	12
2.3 Mecanismos de retroalimentación . . . . .	14
2.3.1 Críticas Sugeridas por el Sistema . . . . .	15
2.3.2 Críticas Motivadas por el Usuario . . . . .	17
2.4 Entornos de Visualización . . . . .	18
2.4.1 Entornos de visualización en dos dimensiones . . . . .	19
2.4.2 Entornos de visualización en tres dimensiones . . . . .	20
2.5 Aspectos Colaborativos . . . . .	25
2.6 Eficiencia en la Recomendación . . . . .	26
2.7 Conclusiones . . . . .	29
3 Técnicas de Recomendación	31
Resumen . . . . .	31
3.1 Introducción . . . . .	31
3.2 Definiciones esenciales . . . . .	32
3.3 Base algorítmica de las propuestas . . . . .	34
3.4 Técnicas de recomendación basadas en críticas . . . . .	37
3.5 Incorporación de opiniones de usuarios en recomendaciones de productos . . . . .	38
3.5.1 Antecedentes . . . . .	39
3.5.2 Obtención de contenidos generados por los usuarios . . . . .	40
3.5.3 Uso de las opiniones <i>off-line</i> de usuarios para enriquecer las descripciones de los productos . . . . .	41
3.5.4 PerTecCB: Técnica de integración de características perceptivas y características técnicas . . . . .	41
3.5.5 PerIndCB: Técnica de integración de características perceptivas y características independientes . . . . .	42
3.5.6 Uso de las opiniones <i>off-line</i> de usuarios dentro del proceso de recomendación . . . . .	43
3.5.7 LwOpinion: Técnica de combinación de LW con contenidos generados por el usuario . . . . .	45
3.5.8 UserPerWeight: Técnica de ponderación de características basada en contenidos perceptivos generados por el usuario . . . . .	47

3.6	Integración de información histórica y opiniones en recomendaciones . . . . .	48
3.6.1	Antecedentes . . . . .	48
3.6.2	HOR: Técnica de integración de recomendaciones históricas y opiniones de usuarios . . . . .	49
3.6.3	Algoritmo HOR . . . . .	52
3.7	Integración de avances previos en recomendaciones basadas sesiones históricas . . . . .	56
3.8	Recomendaciones Conversacionales Colaborativas . . . . .	57
3.8.1	Antecedentes . . . . .	58
3.8.2	Estados del Proceso de Recomendación . . . . .	59
3.8.3	Algoritmo CCR . . . . .	61
3.9	Integración de información colaborativa <i>off-line</i> y <i>on-line</i> . . . . .	66
3.9.1	HO CR: Técnica de integración de información colaborativa <i>off-line</i> y <i>on-line</i> . . . . .	66
3.9.2	Algoritmo HO CR . . . . .	68
3.10	Conclusiones . . . . .	71
4	Entornos de Visualización . . . . .	73
	Resumen . . . . .	73
4.1	Introducción . . . . .	74
4.2	Interfaz de selección de productos . . . . .	74
4.3	e-WebCo: Recomendador Web para Comercio Electrónico . . . . .	76
4.3.1	Arquitectura Conceptual . . . . .	76
4.3.2	Objetos de Recomendación . . . . .	81
4.4	e-3DVirtualCo: Recomendador Virtual Colaborativo en Tres Dimensiones . . . . .	82
4.4.1	Arquitectura Conceptual e-3DVirtualCo . . . . .	83
4.4.2	Estados en el Proceso de Recomendación . . . . .	93
4.4.3	Objetos de Interacción en e-3DVirtualCo . . . . .	96
4.4.4	Ejemplo de interacción y recomendación colaborativa . . . . .	101
4.5	Conclusiones . . . . .	104
5	Evaluación . . . . .	107
	Resumen . . . . .	107
5.1	Introducción . . . . .	107
5.2	Evaluación de las técnicas de recomendación a través de un simulador . . . . .	108
5.2.1	Metodología de evaluación . . . . .	108
5.2.2	Resultados de la integración de opiniones en las bases de datos . . . . .	111
5.2.3	Resultados de integración de opiniones en proceso de recomendación . . . . .	113
5.2.4	Resultados de integración de sesiones históricas y opiniones en el proceso de recomendación . . . . .	116
5.2.5	Resultados de algoritmos de recomendaciones conversacionales colaborativos . . . . .	122
5.2.6	Resultados de integración de información colaborativa <i>off-line</i> y <i>on-line</i> . . . . .	128
5.3	Estudio con usuarios reales de las interfaces de usuario inteligentes . . . . .	133
5.3.1	Evaluación interfaz <b>e-WebCo</b> . . . . .	133
5.3.2	Evaluación interfaz colaborativa <b>e-3DVirtualCo</b> . . . . .	139
5.4	Conclusiones . . . . .	147

6	Conclusiones, contribuciones y trabajo futuro	149
6.1	Conclusiones . . . . .	149
6.1.1	Técnicas de Recomendación . . . . .	149
6.1.2	Interfaces de Usuario Inteligentes . . . . .	153
6.2	Contribuciones . . . . .	155
6.3	Trabajo futuro . . . . .	157
	Anexo	159
A	Documentos de Aplicación de Test e-WebCo	161
B	Documentos de Aplicación de Test e-3DVirtualCo	163

## ÍNDICE DE CUADROS

Cuadro 1	Resumen de técnicas de recomendación basadas en colaboración . . . . .	38
Cuadro 2	Asociación entre las características técnicas, perceptivas e independientes . . . . .	43
Cuadro 3	Proceso de generación de pesos perceptivos . . . . .	46
Cuadro 4	Descripción de las bases de datos . . . . .	109
Cuadro 5	Parámetros de pruebas en las bases de datos . . . . .	110
Cuadro 6	Resumen del proceso de creación de sesiones históricas	117
Cuadro 7	Configuración de las bases de datos de sesiones históricas para los experimentos . . . . .	117
Cuadro 8	Características de los usuarios reales en las pruebas de <b>e-WebCo</b> . . . . .	134
Cuadro 9	Descripción de las fases del protocolo de pruebas en <b>e-WebCo</b> . . . . .	135
Cuadro 10	Cuestionario para medir la eficiencia en la recomendación . . . . .	139
Cuadro 11	Cuestionario para medir la eficacia en la recomendación	139
Cuadro 12	Características de los usuarios reales en las pruebas de <b>e-3DVirtualCo</b> . . . . .	140
Cuadro 13	Tareas en los experimentos de <b>e-3DVirtualCo</b> . . . . .	140
Cuadro 14	Protocolo de evaluación de <b>e-3DVirtualCo</b> . . . . .	142
Cuadro 15	Cuestionario <i>post-test</i> para medir la satisfacción de los usuarios . . . . .	146

## ÍNDICE DE FIGURAS

Figura 1	Esquema general de personalización en interfaces de usuario inteligentes. . . . .	4
Figura 2	Interfaz gráfica del sistema de recomendación basado en críticas denominado <i>Entrée</i> . . . . .	12
Figura 3	Ciclo de recomendación en los sistemas recomendadores basados en críticas. . . . .	13
Figura 4	Interfaz gráfica utilizando críticas unitarias y críticas compuestas, <i>Critique Shop System</i> . . . . .	14
Figura 5	Ejemplos de recomendadores que utilizan críticas motivadas por el usuario. . . . .	18
Figura 6	Interfaz gráfica para la generación de críticas compuestas a través de la selección de productos. . . . .	19
Figura 7	Entornos de visualización del sistema <i>CATS</i> . . . . .	20
Figura 8	Interfaz de usuario del recomendador <i>MobyRek</i> . . . . .	21
Figura 9	Interfaz de usuario del recomendador <i>MapMobyRek</i> . . . . .	21
Figura 10	Entorno de visualización en un mundo virtual de tres dimensiones. . . . .	22
Figura 11	Ejemplos de entornos virtuales integrados con sistemas de recomendación. . . . .	23
Figura 12	Proceso de recomendación del algoritmo <i>Incremental Critiquing</i> . . . . .	35
Figura 13	Resumen gráfico de técnicas de recomendación propuestas. . . . .	37
Figura 14	Uso de opiniones <i>off-line</i> de usuario en la descripción de productos . . . . .	42
Figura 15	Generación de Pesos Perceptivos . . . . .	45
Figura 16	Diagrama conceptual de la técnica de recomendación HOR . . . . .	50
Figura 17	Estados de los usuarios en un proceso de recomendación conversacional colaborativo . . . . .	59
Figura 18	Ciclo de recomendación del Algoritmo CCR . . . . .	62
Figura 19	Diagrama conceptual de la técnica de recomendación HO CR . . . . .	67
Figura 20	Interfaz de recomendación con múltiples vistas: (a) vista local arriba a la izquierda , (b) vista actual arriba a la derecha, (c) vista global abajo y centrado . . . . .	75
Figura 21	Arquitectura Conceptual del Framework e-WebCo . . . . .	77
Figura 22	Interfaz de usuario inteligente para el objeto de recomendación en e-WebCo . . . . .	82
Figura 23	Arquitectura Conceptual del Framework <b>e-3DVirtualCo</b> . . . . .	83
Figura 24	Estados de los usuarios en una sesión de recomendación conversacional colaborativa . . . . .	93
Figura 25	Ejemplo de un <i>hallway</i> de teléfonos móviles inteligentes. . . . .	97
Figura 26	Interfaz del objeto URO . . . . .	97
Figura 27	Interfaz del objeto IMO . . . . .	98
Figura 28	Interfaz de los paneles de recomendación en tres dimensiones dentro de una sala . . . . .	99
Figura 29	Interfaz de usuario inteligente para el objeto de recomendación (RO) . . . . .	101

Figura 30	Escenas del entorno virtual en 3D (e-3DVirtualCo), que muestra las diferentes interacciones individuales y colaborativas que pueden realizar los usuarios . . . . .	103
Figura 31	Resultados de las propuestas <i>PerTecCB</i> y <i>PerIndCB</i> utilizando algoritmo de recomendación IC . . . . .	112
Figura 32	Evolución de la medida <i>ASL</i> para diferentes valores de $\gamma$ . . . . .	114
Figura 33	Resultados de las técnicas <i>LWOpinion</i> y <i>UserPerWeight</i> . . . . .	115
Figura 34	Resultados de <i>ASL</i> de las técnicas propuestas en comparación con las técnicas de IC y HGR en dominios de SMARTPHONE y RESTAURANT . . . . .	118
Figura 35	Beneficios de los algoritmos propuestos en comparación con el algoritmo IC . . . . .	119
Figura 36	Aplicación del test Nemenyi al ranking de <i>ASL</i> de los algoritmos evaluados . . . . .	121
Figura 37	Longitud promedio de la sesión ( <i>ASL</i> ) para las bases de datos CAMERA, PC, TRAVEL, SMARTPHONE y RESTAURANT, considerando diferentes niveles de dificultad en las consultas iniciales . . . . .	123
Figura 38	Beneficio obtenido por el algoritmo CCR en comparación con el algoritmo IC, para las bases de datos CAMERA, PC, TRAVEL, SMARTPHONE y RESTAURANT . . . . .	126
Figura 39	Test de Nemenyi sobre el ranking medio de los algoritmos, respecto a la medida <i>ASL</i> . . . . .	127
Figura 40	Comparación de <i>ASL</i> según niveles de preferencias de los usuarios para HOCR y CCR, utilizando las bases de datos de productos SMARTPHONE y RESTAURANT . . . . .	129
Figura 41	Resultados de <i>ASL</i> de la técnica propuesta HOCR en comparación con las técnicas IC, HORi y CCR en dominios de SMARTPHONE y RESTAURANT . . . . .	131
Figura 42	Beneficios del algoritmo HOCR en comparación con el algoritmo IC . . . . .	133
Figura 43	Pruebas con usuarios reales en <b>e-WebCo</b> . . . . .	136
Figura 44	Resultados de eficiencia y eficacia en la evaluación con usuarios reales . . . . .	137
Figura 45	Percepción de los usuarios acerca de la eficiencia y eficacia de los recomendadores propuestos . . . . .	138
Figura 46	Resultados de eficiencia en pruebas de usuarios reales para algoritmo CCR . . . . .	143
Figura 47	Calidad y análisis de las decisiones en las pruebas de usuarios reales de CCR . . . . .	145
Figura 48	Promedio de los valores de respuestas y desviación estándar para el cuestionario <i>post-test</i> . . . . .	147
Figura 49	Cuestionario post-test para medir la eficiencia en la recomendación . . . . .	161
Figura 50	Cuestionario post-test para medir la eficacia en la recomendación . . . . .	162
Figura 51	Cuestionario post-test para medir la satisfacción de los usuarios . . . . .	163
Figura 52	Cuestionario post-test para medir la satisfacción de los usuarios . . . . .	164





# 1 | INTRODUCCIÓN

## Resumen

En este capítulo presento el ámbito y la problemática inherente en la personalización de información para usuarios de servicios de Internet, que reciben habitualmente un gran volumen de datos, en ambientes tales como comercio electrónico, redes sociales u otros. Adicionalmente, en este capítulo se presentan los objetivos de la investigación y la estructura general de la tesis.

## 1.1 DEFINICIÓN DEL ÁMBITO Y DEL PROBLEMA

ESTA TESIS TRATA el problema de la personalización de información para usuarios de servicios que reciben grandes volúmenes de datos en entornos de trabajo individuales y colaborativos. Específicamente, la tesis se centra en los sistemas de recomendación (Ricci et al., 2011; Adomavicius and Tuzhilin, 2005) como una herramienta clave en la personalización de contenidos para usuarios. Este tipo de sistemas, conocidos también como **recomendadores**, se han implantado mayoritariamente en servicios de comercio electrónico en Internet con el fin de recomendar a los usuarios ítems, productos y/o servicios<sup>1</sup>. El objetivo principal de un recomendador es descubrir y sugerir la información correcta en el momento correcto. En este sentido, todos los sistemas de recomendación tienen en común que recopilan información de la persona que usa el sistema (el usuario) para conocer su perfil de preferencias y basarse en este perfil para realizar las recomendaciones oportunas.

Conocer el perfil del usuario es una ardua tarea que requiere de la participación del propio usuario. El problema surge cuando el usuario desea usar inmediatamente el recomendador y no dedicar tiempo a introducir datos solicitados por el sistema de recomendación (datos que permitirán al recomendador conocer las preferencias del usuario). Para mitigar el problema, en la literatura se han propuesto métodos de recomendación basados en recomendaciones previas realizadas al propio usuario o recomendaciones que se han realizado a otros usuarios que tengan un perfil similar al usuario actual (Koren and Bell, 2011). Sin embargo, en estos enfoques la recomendación no se beneficia de información que se puede obtener durante la interacción actual del usuario con el recomendador, como tampoco de las posibles interacciones y eventuales colaboraciones que el usuario pueda tener con otros usuarios en un espacio compartido, como lo es Internet o un mundo virtual.

Mi propuesta para mejorar la personalización a los usuarios, se centra en proponer técnicas de recomendación integradas en interfaces de usuario inteligentes, añadiendo como base para la recomendación: (1) la información implícita obtenida de la propia interacción del usuario con el recomendador; (2) la información colaborativa proveniente de la interacción con otros usua-

<sup>1</sup> Para simplificar, utilizaré en el resto del texto el concepto de producto para referirme a un ítem, artículo, servicio, noticia o cualquier otro contenido que se recomienda al usuario.

rios (colaboración *on-line*<sup>2</sup>); y (3) la información colaborativa histórica que es recogida de otros usuarios que han interactuado con el sistema (colaboración *off-line*<sup>3</sup>). La información recogida durante la recomendación permitirá alinear las futuras recomendaciones en la misma dirección de las preferencias del usuario registradas en su perfil. Por otro lado, las técnicas de recomendación que se integran en las interfaces inteligentes están focalizadas en espacios de trabajo individuales como también en nuevos espacios de trabajo colaborativos de múltiples usuarios.

#### 1.1.1 Ámbito de trabajo

La personalización de información en Internet involucra en diferente medida una serie de teorías y técnicas procedentes de varios campos de investigación. El marco teórico general de la personalización corresponde a los sistemas de recomendación. Sin embargo, la personalización se basa también en la interacción del usuario con el sistema mediante las interfaces de usuario inteligentes.

**SISTEMAS DE RECOMENDACIÓN (SR)** . Los sistemas de recomendación son técnicas que ofrecen a los usuarios sugerencias de productos de una manera personalizada. Es decir, cada usuario tiene una recomendación propia adecuada a sus preferencias y que puede ser distinta a los demás usuarios. Esta recomendación propia se basa en las preferencias que el usuario ha revelado de manera implícita o explícita al recomendador mediante la interfaz. Los sistemas de recomendación han demostrado ser un medio valioso para que los usuarios de Internet puedan hacer frente a los grandes volúmenes de información y se han convertido en una de las técnicas más populares y poderosas en el comercio electrónico (Ricci et al., 2011). Además, es necesario mencionar que los sistemas de recomendación también se asocian al ámbito del Aprendizaje Automático (AA), principalmente porque este tipo de sistemas utilizan métodos computacionales que pueden aprender y mejorar a partir de la interacción que se produce entre el usuario y el sistema.

**INTERFACES DE USUARIO INTELIGENTES (IUI)** . Las interfaces de usuario inteligentes integran el campo de la Interacción Persona Ordenador (IPO) y el campo de la Inteligencia Artificial (IA). El uso de distintos elementos de la IA dentro de la IPO ha dado lugar a que se acuñara el concepto de *Interfaz de Usuario Inteligente*. Muchos estudios se han centrado en entender el diálogo que se produce entre el usuario y el ordenador para modelar la presentación y despliegue de información que se realiza a los usuarios (Sullivan et al., 1994). En el ámbito de los recomendadores, el uso de una IUI cobra una gran relevancia, ya que uno de los aspectos más importantes en la recomendación corresponde a la forma en que el usuario proporciona información al recomendador a través de la interfaz de usuario. En este sentido, la evolución de la Web 2.0 y los continuos avances en tecnologías gráficas y de redes computacionales están produciendo en distintos ámbitos de Internet, entre ellos el comercio electrónico, una transición desde interfaces en

<sup>2</sup> El término *on-line* lo utilizo para referirme a las interacciones y colaboraciones que se realizan de manera instantánea entre los usuarios durante una sesión de recomendación.

<sup>3</sup> El término *off-line* lo utilizo para referirme a aquella colaboración que resulta de aprovechar información de otros usuarios en beneficio del usuario que está interactuando con el sistema, tales como opiniones, calificaciones de productos o preferencias respecto a productos.

2 dimensiones (2D) hacia experiencias interactivas en tres dimensiones (3D), donde los usuarios pueden compartir sus experiencias, colaborar y ayudar a otros en la toma de decisiones respecto a una compra específica. Un ejemplo claro de esta evolución la representan los Ambientes Virtuales Inteligentes (AVI), los cuales contemplan la integración entre la Inteligencia Artificial y los Ambientes Virtuales en tres dimensiones (Aylett and Luck, 2001). En concreto, un AVI combina técnicas y herramientas de inteligencia artificial con medios efectivos para su representación gráfica en tres dimensiones, favoreciendo la interacción de los usuarios.

#### 1.1.2 Resumen de la propuesta

Durante los últimos años, los estudios asociados a los sistemas recomendadores se han focalizado principalmente en mejorar la eficiencia de la recomendación y proponer distintas interfaces en dos dimensiones. Sin embargo, estos estudios se encuentran disgregados y en general no integran las diferentes técnicas propuestas por otros investigadores. Además, desde el punto de vista de las interfaces de usuario en las investigaciones previas, éstas solo se han focalizado en entornos web de dos dimensiones, sin hacer uso de las nuevas plataformas virtuales de tres dimensiones que favorecen la interacción del usuario y principalmente favorecen la colaboración que se puede producir en este tipo de entornos colaborativos.

La propuesta que planteo en esta tesis consiste en la definición de nuevas técnicas de recomendación que interactúen con interfaces de usuario inteligentes en entornos colaborativos. Esta propuesta implica (i) determinar nuevas técnicas de recomendación y (ii) integrar las técnicas de recomendación en interfaces inteligentes y colaborativas. Por lo tanto, el procedimiento que establezco para desarrollar la propuesta se compone de dos etapas, como se muestra en la Figura 1.

##### PRIMERA ETAPA

La primera etapa, descrita en la parte superior de la Figura 1, consiste en definir nuevas estrategias o técnicas de recomendación que incorporan por un lado información implícita que el usuario facilita al recomendador durante el diálogo que se produce en una sesión de recomendación, y por otro lado información colaborativa proveniente de otros usuarios. En primer lugar, se proponen técnicas de recomendación que integran nuevas fuentes de información colaborativa proveniente de opiniones de otros usuarios acerca de los productos (véase *PerTecCB*, *PerIndCB*, *UserPerWeight* y *LwOpinion* en la Figura 1). En segundo lugar, se definen nuevas técnicas que integran tanto opiniones de usuarios como también información histórica proveniente de sesiones de recomendación<sup>4</sup> efectuadas por otros usuarios (véase *HOR* en la Figura 1). En tercer lugar, se propone una técnica de recomendación que utiliza información colaborativa proveniente de la interacción inmediata (*on-line*) con otros usuarios (véase *CCR* en la Figura 1). Finalmente, se proponen mejoras para las nuevas técnicas y para técnicas existentes, mediante la aplicación de mecanismos de ponderación (véase *HORi* y *HGRi* en la Figura 1).

<sup>4</sup> Una sesión de recomendación comprende todo el proceso que se lleva a cabo desde el momento que el usuario inicia la interacción con el recomendador, hasta que compra un producto o abandona el proceso voluntariamente. En esta tesis utilizaré indistintamente los términos proceso de recomendación y sesión de recomendación.

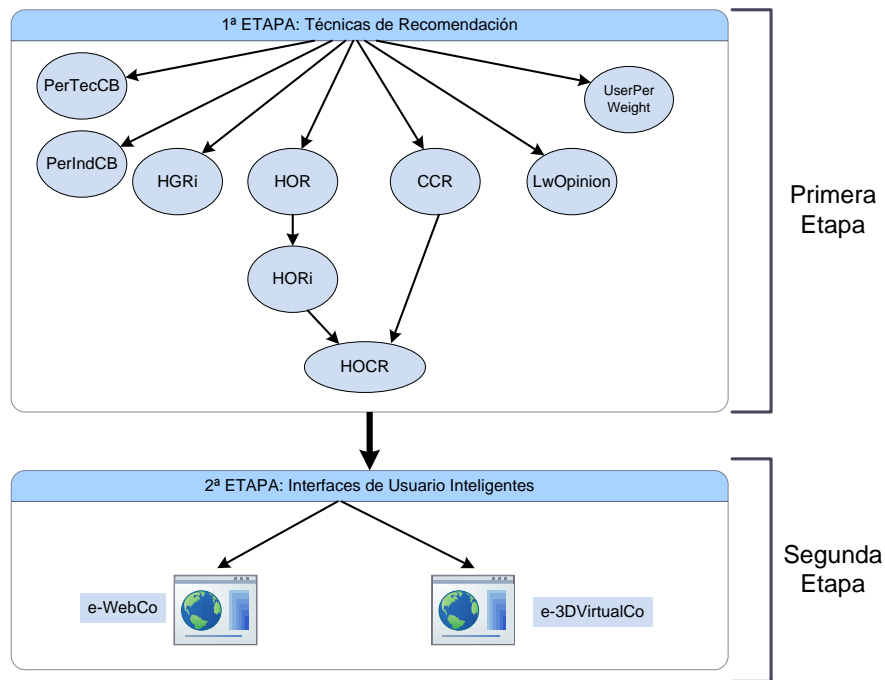


Figura 1: Esquema general de personalización en interfaces de usuario inteligentes.

El resultado de esta etapa es un conjunto de algoritmos de recomendación, que permiten hacer uso de la información obtenida a través de diferentes mecanismos y estrategias, con el fin de mejorar la personalización durante una recomendación. Finalmente, con el objetivo de unificar las técnicas propuestas, en esta etapa se define una única técnica de recomendación en la cual convergen las principales técnicas anteriores, denominada *HOCR*.

#### SEGUNDA ETAPA

La segunda etapa, descrita en la parte inferior de la Figura 1, consiste en proponer **interfaces de usuario inteligentes** implementadas en entornos de visualización, que favorecen en primer lugar la recogida de información de parte del usuario (mayoritariamente las preferencias de usuario), en segundo lugar la recomendación personalizada de productos al usuario y finalmente que potencien aspectos propios del ser humano, tales como la colaboración, comunicación e interacción entre las personas. En concreto, los entornos de visualización propuestos para las interfaces de usuario inteligentes se basan en los tradicionales ambientes web en dos dimensiones y en nuevos ambientes virtuales inteligentes en tres dimensiones, los cuales ofrecen a los usuarios sensaciones de inmersión dentro de un mundo alternativo y facilitan la interacción y colaboración con otros usuarios.

El resultado de esta etapa son dos interfaces de usuario inteligentes implementadas en plataformas de visualización de distinta naturaleza (véase las dos interfaces denominadas **e-WebCo** y **e-3DVirtualCo** en la Figura 1): (i) **e-WebCo** es una interfaz en un entorno web de dos dimensiones que integra los algoritmos de recomendación: HOR, HORi y HGRi; (ii) **e-3DVirtualCo** es una interfaz basada en un entorno

virtual colaborativo en tres dimensiones, que integra el algoritmo CCR y HOCR. En estas interfaces los sistemas de recomendación tienen el objetivo de aprovechar las características propias de cada entorno para mejorar la personalización en las recomendaciones, favorecer la interacción y colaboración entre los usuarios y finalmente mejorar la satisfacción del usuario en el uso del sistema de recomendación.

Cada etapa de la propuesta se presenta en un capítulo independiente: la primera etapa en el Capítulo 3 y la segunda etapa en el Capítulo 4. A continuación expongo los objetivos que guían el desarrollo de esta propuesta.

## 1.2 OBJETIVOS

El objetivo principal de esta tesis es integrar técnicas de personalización con interfaces de usuario inteligentes, basadas en el comportamiento colaborativo del ser humano y en la información recogida de las interacciones de las personas tanto con la interfaz como con otros usuarios. Para alcanzar este objetivo general he definido cuatro objetivos específicos:

- Definir nuevas técnicas de personalización basadas en las interacciones y retroalimentaciones por parte de los usuarios, y que incluyan también el comportamiento colaborativo del usuario con otras personas.
- Implementar las técnicas de personalización propuestas, por medio de nuevos algoritmos de recomendación.
- Integrar las técnicas de personalización con interfaces de usuario inteligentes en espacios de visualización individuales y colaborativos.
- Evaluar las técnicas de personalización y las interfaces de usuario inteligentes por medio de un simulador que automatiza el comportamiento de las personas y a través de evaluaciones con usuarios reales.

La tesis también deja dos entornos funcionando en la web: e-WebCo y e-3DVirtualCo.

## 1.3 ESTRUCTURA DE LA TESIS

El presente documento se estructura en seis capítulos que sintetizo a continuación.

### Capítulo 1 Introducción

En este capítulo presento el ámbito y el problema inherente en la personalización de información para usuarios que utilizan grandes volúmenes de datos provenientes de servicios de Internet, junto con los objetivos del trabajo. Adicionalmente, en este capítulo presento la estructura general de la tesis.

### Capítulo 2 Estado del arte: antecedentes y últimos avances

En este capítulo efectúo una revisión del estado del arte de los sistemas de recomendación. Esta revisión está orientada a describir los antecedentes que

representan los fundamentos en el estudio de los sistemas de recomendación, describir los diferentes tipos de sistemas de recomendación que existen actualmente (Sección 2.2), los mecanismos de retroalimentación que utilizan los sistemas de recomendación (Sección 2.3), las diferentes propuestas que existen para mejorar el desempeño durante la recomendación (Sección 2.6), los entornos de visualización mayormente utilizados (Sección 2.4) y los aspectos colaborativos que este tipo de sistemas de recomendación pueden disponer para el usuario (Sección 2.5). Finalmente, en la Sección 2.7 presento las conclusiones del capítulo.

### Capítulo 3 Técnicas de Recomendación

En este capítulo describo las nuevas propuestas de personalización, las cuales están basadas principalmente en la interacción y colaboración de los usuarios, tanto con el sistema de recomendación como entre los mismos usuarios. En primer lugar, presento las definiciones esenciales que se requieren para una correcta comprensión de las técnicas de personalización (Sección 3.2). En segundo lugar, describo los algoritmos de recomendación que sirvieron como base para las nuevas técnicas de personalización (Sección 3.3). En tercer lugar, describo en términos generales las técnicas de personalización propuestas (Sección 3.4). Posteriormente, describo en detalle las técnicas propuestas: incorporación de opiniones de usuarios en recomendaciones de productos (Sección 3.5), integración de recomendaciones histórica y opiniones de usuarios en la personalización de contenidos (Sección 3.6), integración de avances de estudios previos en recomendaciones basadas en sesiones históricas (Sección 3.7), recomendaciones conversacionales colaborativas (Sección 3.8), integración de información colaborativa *off-line* y *on-line* en recomendaciones (Sección 3.9). Finalmente, en la Sección 3.10 presento las conclusiones del capítulo.

### Capítulo 4 Entornos de Visualización

En este capítulo presento los entornos de visualización en los cuales se integran las técnicas de personalización descritas en el capítulo anterior. En concreto, propongo las interfaces de usuario inteligentes que permitirán a los usuarios interactuar con las técnicas implementadas en estas interfaces. Específicamente, las propuestas de interfaces de usuarios inteligentes presentadas en esta tesis, se categorizan en interfaces de dos dimensiones (Sección 4.3) e interfaces basadas en entornos virtuales inteligentes de tres dimensiones (Sección 4.4). Para cada interfaz inteligente se describen sus elementos principales, funcionalidades y fortalezas del entorno de visualización que favorecen el uso de las técnicas de personalización propuestas. Finalmente, en la Sección 4.5 presento las conclusiones del capítulo.

### Capítulo 5 Evaluación

En este capítulo presento la evaluación realizada tanto en las técnicas de personalización propuestas como en las interfaces de usuario inteligentes implementadas. En primer lugar, analizo el desempeño de las nuevas técnicas de recomendación, por medio de un simulador que emula el comportamiento de los usuarios (Sección 5.2). Posteriormente, en la Sección 5.3 presento los resultados obtenidos por medio de una evaluación con usuarios reales de

las técnicas de recomendación integradas en las interfaces de usuario inteligentes definidas en el Capítulo 4. Adicionalmente, para ambas evaluaciones (simulador y usuarios reales), analizo estadísticamente los resultados obtenidos. Finalmente, en la Sección 5.4 presento las conclusiones del capítulo.

Capítulo 6 Conclusiones, contribuciones y trabajo futuro

En este capítulo presento las conclusiones globales de este trabajo, las aportaciones del mismo y las líneas de trabajo futuro.

## 1.4 CONCLUSIONES

En este capítulo he presentado, en primer lugar, el ámbito y el problema que trata esta tesis. Para esto, he descrito el ámbito de trabajo y un resumen de mi propuesta. En segundo lugar, he presentado los objetivos del trabajo y finalmente, he presentado la estructura general de la tesis.





# 2 | ESTADO DEL ARTE: ANTECEDENTES Y ÚLTIMOS AVANCES

## Resumen

En este capítulo efectúo una revisión del estado del arte de los sistemas de recomendación. Esta revisión está orientada a definir los antecedentes que representan los fundamentos de los sistemas recomendadores, su historia y los últimos avances. En primer lugar, presento los antecedentes e historia general de los sistemas de recomendación que existen actualmente, entre los cuales destacan las técnicas de filtrado colaborativo, filtrado basado en contenidos y recomendadores basados en críticas. Posteriormente presento los últimos avances en sistemas de recomendación basados en críticas y me enfoco en analizar, categorizar y describir los sistemas de recomendación basados en críticas teniendo en cuenta: el mecanismo de retroalimentación que ofrecen para el usuario, el entorno de visualización que utilizan y los aspectos colaborativos que ofrecen. Finalmente, presento las conclusiones del capítulo.

## 2.1 INTRODUCCIÓN

**H**OY EN DÍA, los usuarios están más que nunca en busca de productos para comprar, de música para escuchar, de artículos y noticias para leer y cualquier tipo de contenido que se encuentre disponible en la web. Generalmente, el gran volumen y la diversidad de contenidos existentes en la web dificultan la búsqueda por parte de los usuarios. En este sentido, los sistemas de recomendación han surgido como una herramienta para ayudar a los usuarios en esta búsqueda, entregando sugerencias de ítems<sup>1</sup> en base a los requerimientos o preferencias específicas de cada usuario (Ricci et al., 2011; Konstan and Riedl, 2012). Estas sugerencias dan apoyo a los usuarios de la web en el proceso de toma de decisiones, tales como compras de productos, música para escuchar, o noticias para leer. El desarrollo de los sistemas de recomendación involucra múltiples disciplinas y/o ciencias tales como la Inteligencia Artificial, la Interacción Persona Ordenador, las Tecnologías de la Información, la Minería de Datos, la Estadística, los Sistemas de Soporte a Decisiones y Marketing, entre otras.

Un ejemplo de sitio de comercio electrónico que emplea un tipo específico de sistema de recomendación para personalizar contenidos para cada cliente es la popular web de Amazon.com<sup>2</sup>. Sin embargo, en la actualidad el uso de los sistemas de recomendación se ha extendido a variedad de sitios web tales como: TripAdvisor<sup>3</sup>, Youtube<sup>4</sup>, Netflix<sup>5</sup>, Yahoo<sup>6</sup>, entre otros.

Los sistemas de recomendación habitualmente se clasifican de acuerdo a las técnicas que utilizan para sugerir productos a los usuarios, las cuales van

<sup>1</sup> El término ítem es utilizado para describir el producto o servicio que el sistema de recomendación sugiere a los usuarios.

<sup>2</sup> <http://www.amazon.com>

<sup>3</sup> <http://www.tripadvisor.com>

<sup>4</sup> <http://www.youtube.com>

<sup>5</sup> <http://www.netflix.com>

<sup>6</sup> <http://www.yahoo.com>

desde técnicas de filtrado colaborativo (*Collaborative Filtering*, CF<sup>7</sup>) (Koren and Bell, 2011; Elahi et al., 2014) hasta técnicas de recomendación basadas en contenido (*Content-Based filtering*, CB<sup>8</sup>) (Lops et al., 2011; Pazzani and Billsus, 2007).

Por un lado, las técnicas de filtrado colaborativo (CF) comparan las preferencias del usuario actual con las valoraciones realizadas por otros usuarios a diferentes productos recomendados anteriormente. En base a esta comparación, estos sistemas recomiendan al usuario, aquellos productos que fueron del agrado de otros usuarios con preferencias similares al usuario actual. Estos recomendadores están basados principalmente en mecanismos de valoración de los productos por parte de los usuarios (por ejemplo por medio de *ratings*<sup>8</sup>). Además las técnicas CF permiten agregar nuevos productos al sistema de forma rápida e incremental, principalmente debido a que las recomendaciones no se basan en el contenido de los ítems. Sin desmedro de las fortalezas de las técnicas de CF, éstas pueden presentar dos tipos de problemas. El primer problema se produce porque en la mayoría de los casos se tienen evaluaciones de pocos productos (problema conocido en el ámbito de los sistemas de recomendación como escasez de datos, *data sparsity*). El segundo problema es un caso especial del primero, el cual se denomina problema del comienzo en frío (generalmente conocido como *cold start problem*). Este segundo problema se puede producir en un primer caso, cuando se quiere recomendar a un usuario que nunca ha evaluado un producto, o cuando se quiere recomendar un producto que no ha sido evaluado nunca por los usuarios. En ambos casos, no es posible obtener similitudes entre el usuario actual y los otros usuarios.

Por otro lado, las técnicas de recomendación Basadas en Contenido (CB) comparan las preferencias en las características del producto deseado por el usuario con las características de los productos disponibles para recomendar. En base a esta comparación, este tipo de sistemas recomiendan aquellos productos más similares a las características requeridas por el usuario. En las técnicas de filtrado CB, a diferencia de las técnicas CF, no se requieren *ratings* de otros usuarios, sino que solamente se focalizan en las preferencias sobre que características se desean del producto. Además, este tipo de técnica ofrece mayores posibilidades para explicar cada producto recomendado, porque se puede mostrar qué características son las que influyeron en la recomendación efectuada. Otra ventaja de las técnicas de CB, es que permiten recomendar productos nuevos que no han sido valorados por ningún usuario. Sin embargo, teniendo en cuenta que estos sistemas utilizan las características del producto como base para evaluar la similitud, se pueden presentar alguno de los problemas mencionados a continuación. El primer problema, denominado análisis de contenido limitado se presenta en algunos dominios en los cuales no se cuenta con un detalle uniforme y formal para las características de los productos (por ejemplo, cuando se recomiendan textos o documentos). El segundo problema se produce cuando el recomendador genera sugerencias muy similares a las que el usuario ya conoce (problema denominado como sobre-especialización). Un último problema que se puede presentar en este tipo de sistemas ocurre cuando se realizan sugerencias para nuevos usuarios, los cuales no habrán valorado productos anteriormente, y por lo tanto el recomendador no podrá realizar buenas recomendaciones. Es necesario mencionar que existen además técnicas híbridas que se basan

7 Utilizaré el acrónimo en inglés por ser éste el más ampliamente utilizado en la literatura científica.

8 Un *rating* corresponde a una valoración de un producto realizada por los usuarios, de acuerdo a un nivel establecido (numérico o simbólico).

en la combinación de técnicas de CF y CB, con el objetivo de intentar aprovechar las ventajas de una de las técnicas mencionadas para solventar las desventajas de la otra. Este tipo de recomendadores se denominan Sistemas de Recomendación Híbridos (Burke, 2002a).

La mayoría de los recomendadores, CF y CB, están basados en una estrategia denominada *single-shot*. La estrategia *single-shot* presenta a los usuarios un conjunto de recomendaciones en base a la descripción inicial definida por el usuario y, el proceso de recomendación generalmente finaliza. Por este motivo, los recomendadores *single-shot* reciben su nombre en referencia a la única, *single*, interacción del usuario con el sistema. En muchos casos, esta estrategia *single-shot* no es la más efectiva, ya que los usuarios no tienen en muchas ocasiones totalmente claras sus preferencias desde el inicio de la recomendación, sino hasta que ellos hayan interactuado un cierto tiempo con el sistema de recomendación y por lo tanto lleguen a entender el rango de alternativas de productos que pueden seleccionar dentro del gran abanico de posibilidades existentes. Sin desmedro de lo anterior, este tipo de estrategia generalmente es apropiada para sistemas recomendadores de productos simples, pero no es la más adecuada cuando hay que recomendar conjuntos de productos complejos<sup>9</sup> o dominios de productos calificados como *high-risk*<sup>10</sup>. Enfocado en este tipo de productos, los denominados Sistemas de Recomendación Conversacionales (Bridge et al., 2005), apoyan al usuario en la búsqueda en dominios de productos complejos y/o *high-risk* ya que involucran al usuario en un diálogo durante el proceso de recomendación. Este diálogo sirve para obtener incrementalmente una retroalimentación sobre las preferencias del usuario con el fin de refinar las recomendaciones.

Específicamente, los sistemas de recomendación conversacionales guían a los usuarios en la búsqueda de un producto deseado, usando un proceso de recomendación cíclico, el cual consiste en otorgar sugerencias de productos y recibir una retroalimentación por parte del usuario en cada ciclo. De esta forma, se produce un refinamiento de las preferencias del usuario en cada ciclo de recomendación. Los recomendadores conversacionales se fundamentan en el Razonamiento Basado en Casos (CBR<sup>11</sup>) (Bridge, 2002).

Actualmente, existen numerosas investigaciones centradas en recomendadores conversacionales, de las cuales un gran número se ha focalizado en una estrategia de retroalimentación denominada *critiquing* (Burke et al., 1997). La estrategia *critiquing* es un mecanismo de retroalimentación en el cual los usuarios proporcionan implícitamente información de sus preferencias al recomendador, por medio de críticas focalizadas sobre las características del producto recomendado. Los sistemas de recomendación conversacionales basados en *critiquing*, también se conocen habitualmente como Sistemas de Recomendación Basados en Críticas (Burke et al., 1997; Burke, 2002b; Pu and Faltings, 2004; Reilly et al., 2004; McGinty and Reilly, 2011; Pu et al., 2011; Chen and Pu, 2012). En la última década, lejos de disminuir el interés por los recomendadores basados en críticas, éste ha aumentado, principalmente porque este tipo de sistemas constituye una poderosa forma de ayudar a los usuarios cuando éstos no cuentan con una idea clara del producto deseado y el número de productos o ítems es demasiado amplio para poder tomar

<sup>9</sup> Un producto complejo, es aquel producto que posee una gran variedad de características que lo describen y que generalmente no existe una gran diferenciación entre un producto y otro. En estos casos el usuario se ve abrumado por toda la información.

<sup>10</sup> En dominios de productos *high-risk*, la tarea de buscar un producto determinado en un amplio conjunto de productos es intimidante para los usuarios, ya sea porque el usuario compra un producto por primera vez o porque éste no es un producto que se adquiera masivamente (por ejemplo la compra de un vehículo, un apartamento, etc).

<sup>11</sup> Las siglas del acrónimo provienen del término en inglés Case-Based Reasoning

una decisión. De hecho, los recomendadores basados en críticas permiten al usuario un mejor entendimiento de sus propias preferencias durante los ciclos de recomendación, ya que en cada ciclo el usuario refina sus preferencias de acuerdo a las recomendaciones que recibe y de esta forma en cada ciclo aprende como es el dominio de los productos. Considerando, que en este tipo de recomendadores la retroalimentación del usuario es de vital importancia, y tomando en cuenta el objetivo de mi investigación, en las siguientes secciones me enfocaré en la discusión y análisis de este tipo de sistemas. A continuación se presentan los principales antecedentes y fundamentos que sustentan los sistemas de recomendación basados en críticas.

## 2.2 SISTEMAS BASADOS EN CRÍTICAS: ANTECEDENTES

Los sistemas basados en críticas se fundamentan en la idea que para un usuario es más fácil criticar un producto recomendado, por ejemplo, indicando "me gusta este producto, pero más barato", que construir una consulta más formal y detallada en el recomendador (McSherry and Aha, 2007). Formalmente, una crítica se define como una preferencia del usuario sobre una característica específica de un producto (por ejemplo, "quiero un producto de otra marca").



Figura 2: Interfaz gráfica del sistema de recomendación basado en críticas denominado *Entrée*.

Uno de los primeros sistemas de recomendación basados en críticas fue el sistema *Entrée* (Burke et al., 1997). La Figura 2 muestra su interfaz de usuario. En la parte superior de la figura se muestra el producto recomendado, en este caso Restaurantes, y en la parte inferior se muestran varias críticas predefinidas por el sistema, las cuales afectan a una característica particular del producto. Algunas críticas que se pueden observar son: menor precio, restaurante tradicional, restaurante de ambiente tranquilo, etc. En base a la interacción del usuario con las críticas, el sistema recomendará un nuevo producto (restaurante) que satisfaga esas críticas.

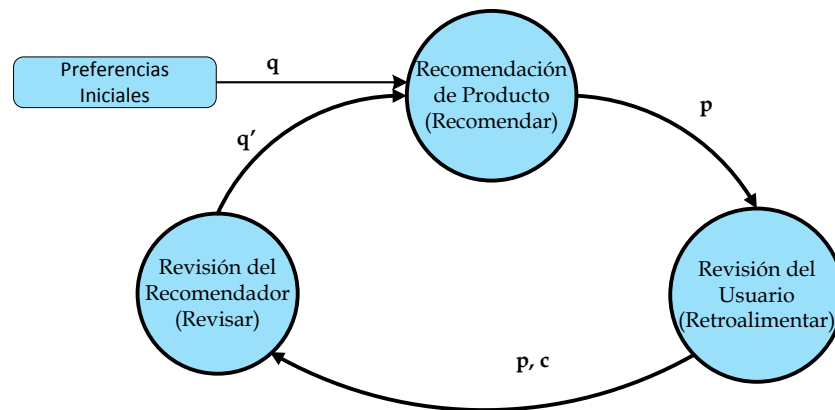


Figura 3: Ciclo de recomendación en los sistemas recomendadores basados en críticas.

Generalmente, un sistema recomendador basado en críticas tiene un proceso de recomendación de tres fases, tal como se describe en la Figura 3. En primer lugar el recomendador solicita al usuario que escoja un producto inicial como referencia de búsqueda o que defina algún valor específico para una o más características del producto. Esta información inicial se establece como la consulta inicial (*query*,  $q$ ) que el recomendador utilizará en la fase de recomendación y a partir de esta información, se inicia el proceso de recomendación según las siguientes fases:

1. **Recomendar.** El sistema de recomendación muestra uno o múltiples productos de acuerdo a las preferencias iniciales del usuario (consultas iniciales,  $q$ ) o en base al último producto recomendado ( $q'$ );
2. **Retroalimentar.** En este punto el usuario puede seleccionar un producto ( $p$ ) como su selección final y terminar el proceso de recomendación. Sin embargo, en la mayoría de los casos, un usuario quiere un producto similar al recomendado y tiene la necesidad de hacer alguna retroalimentación con algún pequeño cambio en alguna de sus características mediante una o varias críticas ( $c$ ) al producto recomendado.
3. **Revisar.** Una vez que la/s críticas son hechas, el sistema de recomendación actualiza las preferencias del usuario para el próximo ciclo de recomendación en base a las críticas realizadas por el usuario. En esta fase, el producto recomendado ( $p$ ) en el ciclo anterior pasa a ser el producto *query* ( $q'$ ) en la fase de recomendación.

En la literatura existen muchos algoritmos de recomendación basados en críticas que tienen como base este ciclo de recomendación. Así pues, en las siguientes secciones me centraré en analizar específicamente el estado de arte de los sistemas de recomendación basados en críticas de acuerdo a la siguiente categorización: el mecanismo de retroalimentación que ofrecen para el usuario, el entorno de visualización que utilizan y los aspectos colaborativos que ofrecen.

## 2.3 MECANISMOS DE RETROALIMENTACIÓN

En los sistemas de recomendación basados en críticas, los mecanismos de retroalimentación más comúnmente utilizados corresponden a: críticas unitarias (Burke et al., 1997) y críticas compuestas (Reilly et al., 2004). La Figura 4 muestra ambos tipos de críticas en el sistema de recomendación *Critique Shop* (Zhang and Pu, 2006).



Figura 4: Interfaz gráfica utilizando críticas unitarias y críticas compuestas, *Critique Shop System*.

Las críticas unitarias (Burke et al., 1997; Burke, 2002b; McCarthy et al., 2006c; Ricci and Nguyen, 2005; Ricci and Nguyen, 2007), corresponden a preferencias aplicadas sobre una sola característica del producto actualmente recomendado, por ejemplo una crítica unitaria sería "menor precio". Una crítica unitaria puede ser presentada en una interfaz de usuario como un botón asociado al valor de una característica, de tal forma que sea fácilmente seleccionado por el usuario (ver lado izquierdo de la Figura 4). Sin desmedro

de lo anterior, las críticas unitarias pueden no ser completamente eficientes cuando un usuario necesita definir sus preferencias sobre dos o más características del producto. Esta situación repercutirá en un aumento de los ciclos de interacción entre el usuario y el recomendador. Adicionalmente, las críticas unitarias eventualmente pueden limitar la capacidad del recomendador para acercarse al producto requerido por el usuario, ya que en ocasiones los usuarios por desconocimiento, podrían motivarse con criticar muchas veces una sola característica del producto actualmente recomendado, en desmedro de alguna otra característica que puede ser importante para alcanzar el producto que está buscando (véase un ejemplo de críticas unitarias en el lado izquierdo de la interfaz del recomendador *Critique Shop*, mostrado en la Figura 4).

De manera alternativa, existen las críticas compuestas (Reilly et al., 2004; Zhang and Pu, 2006; Reilly et al., 2007; Pu et al., 2008; McGinty and Reilly, 2011), las cuales son una combinación de múltiples críticas unitarias, de tal forma que cada crítica realizada afecta a múltiples características del producto recomendado. Un ejemplo de crítica compuesta puede ser: "menor precio y diferente marca". Este tipo de críticas puede mejorar la eficiencia del recomendador, ya que éstas le permiten obtener mayor información de las preferencias del usuario en un solo ciclo de recomendación. Las críticas compuestas pueden ser presentadas al usuario por medio de texto que describa las preferencias para el conjunto de críticas unitarias, pero también existe la posibilidad de representarlas mediante iconos que muestran visualmente la orientación de cada crítica unitaria como lo muestra la Figura 4 en la parte inferior derecha. Por ejemplo, para una crítica unitaria numérica el ícono muestra flechas positivas o negativas para orientar el incremento o decremento del valor de la característica del producto.

La mayoría de los sistemas de recomendación basados en críticas se decantan por definir la retroalimentación mediante críticas unitarias, dado que los usuarios prefieren manejar una crítica a la vez, lo cual incluso puede significar un menor esfuerzo cognitivo del usuario (Chen and Pu, 2006).

Adicionalmente, de acuerdo a las investigaciones realizadas en la última década, se han definido dos métodos para la generación de críticas: Críticas Sugeridas por el Sistema (*system-suggested critiquing*) y Críticas Motivadas por el Usuario (*user-motivated critiquing*).

### 2.3.1 Críticas Sugeridas por el Sistema

Las críticas sugeridas por el sistema, son críticas propuestas por el recomendador en las cuales el usuario no interviene directamente en su definición (Burke, 2002b; Reilly et al., 2004; Zhang and Pu, 2006; Pu et al., 2008; Zhang et al., 2008; Chen and Pu, 2009; Pu et al., 2011). En este método, el recomendador genera las críticas de acuerdo a su propio conocimiento del dominio del producto, a las preferencias del usuario actual o en base a los restantes productos disponibles para recomendar. Aunque, en este método el usuario no tiene la libertad para generar sus propias críticas, algunos usuarios lo prefieren por ser un método más intuitivo de usar y en algunos casos resulta más rápida la búsqueda cuando las críticas ofrecidas fueron las más relevantes para acotar el espacio de búsqueda<sup>12</sup>. Uno de los recomendadores más antiguo basado en críticas sugeridas por el sistema corresponde al sistema denominado *FindMe* (Burke et al., 1997; Burke, 2000; Burke, 2002b).

<sup>12</sup> Un espacio de búsqueda se refiere al conjunto completo de todos los productos disponibles para recomendar.



Este recomendador permite a los usuarios criticar el producto actualmente recomendado, seleccionando una de las críticas pre-definidas por el sistema (más barato, más grande, más bonito, etc.). Cuando un usuario selecciona una crítica, los productos disponibles para recomendar (generalmente conocidos como productos candidatos) son filtrados y se mantienen en la base de productos candidatos solo aquellos que satisfacen la crítica seleccionada por el usuario.

Durante los últimos años esta manera de generar las críticas ha sido abordado en muchas investigaciones con el fin de hacer más dinámica la generación de las críticas. En la literatura se han desarrollado principalmente dos propuestas.

- Críticas Dinámicas (en inglés se denomina *Dynamic Critiquing*) (Reilly et al., 2004; McCarthy et al., 2004; McCarthy et al., 2005b). Esta metodología busca eliminar las limitaciones que existían en los sistemas conversacionales basados en críticas previas, tales como *FindMe*, donde se trabaja con críticas predefinidas por el sistema, las cuales generalmente son de tipo unitarias. Teniendo esto en consideración nace una nueva estrategia para generar críticas compuestas dinámicas conocida por su nombre en inglés *dynamic compound critiques* (Reilly et al., 2004). Esta propuesta está basada en el uso del algoritmo denominado *APRIORI* (Agrawal et al., 1993; Agrawal et al., 1994), para descubrir conjuntos de frecuencia de valores distintos entre la actual recomendación y el resto de productos disponibles para recomendar. De esta forma, se establece un valor para cada posible crítica compuesta, de acuerdo a la cantidad de productos restantes que la satisfacen. El sistema de recomendación mostrará al usuario aquellas críticas que satisfacen un menor número de productos candidatos. Esta propuesta ha obtenido resultados muy satisfactorios en pruebas realizadas con usuarios reales (McCarthy et al., 2005b; McCarthy et al., 2005a). Posteriormente, esta propuesta ha sido mejorada en uno de los más conocidos sistemas de recomendación basados en críticas denominado *Incremental Critiquing, IC* (Reilly et al., 2005a). El recomendador IC mantiene un registro histórico de las críticas realizadas por el usuario en ciclos de recomendación anteriores, y por tanto, evita recomendar productos con valores de características que el usuario ha manifestado su desinterés a través de críticas previas, un mayor detalle del recomendador IC se presenta en la Sección 3.3.
- Críticas compuestas basadas en la teoría de Utilidades de Múltiples Atributos, conocida habitualmente por sus siglas en inglés MAUT (Multi-Attribute Utility Theory) (Zhang and Pu, 2006; Zhang et al., 2008). Un ejemplo de recomendador que implementa este tipo de críticas es el sistema *Critique Shop* que se muestra en la Figura 4. Considerando que *dynamic critiquing* no toma en cuenta las preferencias de un usuario en la generación de las críticas, Zhang et al. (2006), proponen modelar las preferencias de usuario en base a la teoría MAUT, la cual produce un puntaje para cada producto que representa su grado de satisfacción con las preferencias de usuario. De esta forma, el sistema de recomendación sugiere el producto con mayor puntaje y del resto de productos se generan las críticas compuestas de acuerdo a las diferencias que estos productos tengan en referencia al producto recomendado. Por ejemplo, una crítica podría ser del tipo "mismo marco del producto, pero con menor precio, menor tamaño y menor peso". Esta propuesta

basada en MAUT ha demostrado mejores resultados que *dynamic critiquing* en las evaluaciones realizadas con un simulador (Zhang and Pu, 2006).

- Técnica de Organización basada en Preferencias (Pu and Chen, 2006). Esta propuesta busca mejorar algunas carencias de los enfoques anteriores, donde solo es posible recomendar un producto al usuario y las críticas compuestas pueden contener muchas características que sobrecarguen de información al usuario. Específicamente, el recomendador genera críticas utilizando la teoría MAUT, pero adicionalmente utiliza un mecanismo de selección para las nuevas críticas compuestas basado en las preferencias del usuario de acuerdo a las críticas realizadas anteriormente. A continuación, por medio del algoritmo APRIORI se seleccionan las características más representativas y se establecen categorías de productos representados por las características compuestas generadas. Esta propuesta se ha evaluado en comparación con la técnica *dynamic critiquing* y la técnica basada en MAUT, obteniendo mejores resultados que las técnicas predecesoras (Chen and Pu, 2007b).

### 2.3.2 Críticas Motivadas por el Usuario

La mayoría de los sistemas de recomendación basados en críticas implementan críticas unitarias motivadas por el usuario (Chen and Pu, 2006). Este enfoque consiste en flexibilizar la generación de críticas permitiendo al usuario definir sus propias críticas. El recomendador *Apt Decision* (Shearin and Lieberman, 2001), es un ejemplo de recomendador que utiliza críticas motivadas por el usuario. En este caso, los usuarios son libres de criticar múltiples características de los productos (en este caso particular corresponden a apartamentos), ofrecidos por el recomendador (véase la Figura 5a).

Otra aproximación corresponde a un enfoque diferente denominado *Example Critiquing* (Chen and Pu, 2006), que permite al usuario un mayor control respecto a la restricción que quiere aplicar para una característica determinada del producto. Por ejemplo, si actualmente el producto recomendado tiene un precio de X euros, el usuario podría realizar una crítica del tipo “quiero un producto Y euros más barato que el producto recomendado”(véase la Figura 5b).

Otros autores, presentan propuestas que además de recomendar aquellos productos que más se ajusten a las preferencias del usuario, recomiendan productos que pueden ser del interés del usuario si éste agrega una nueva preferencia, sin realizar un nuevo ciclo de recomendación (Viappiani et al., 2006b; Viappiani et al., 2006a). Para la predicción de la nueva preferencia, estos autores utilizan un modelo basado en la eficiencia de Pareto (Luc, 2008).

En un trabajo previo, he participado en la elaboración de un algoritmo que facilita la selección directa de productos por parte del usuario, durante la recomendación, mediante diferentes vistas del espacio de búsqueda. En concreto, el sistema de recomendación posee una vista actual del producto, una vista local y una vista global. Precisamente, la selección de productos se realiza en la vista global de la interfaz de usuario mostrada en la Figura 6 (véase el área inferior de la figura). Cuando el usuario selecciona un producto desde esta vista global, se genera automáticamente una crítica compuesta motivada por la selección del usuario (Salamó et al., 2013).

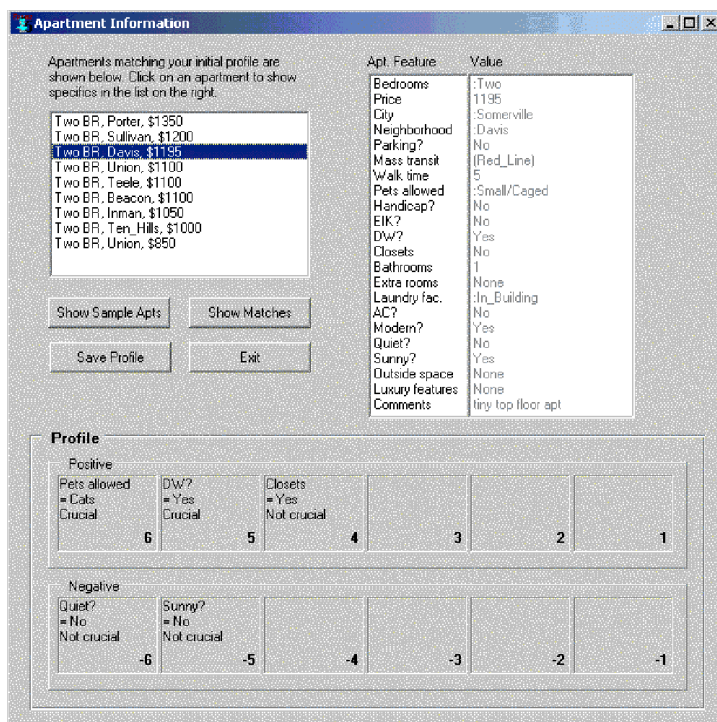
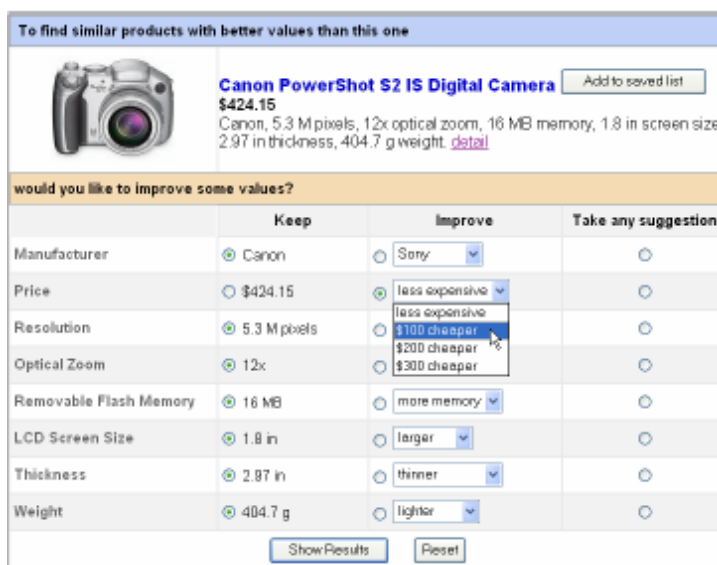
(a) Recomendador *Apt Decision*(b) Recomendador *Example Critiquing*

Figura 5: Ejemplos de recomendadores que utilizan críticas motivadas por el usuario.

## 2.4 ENTORNOS DE VISUALIZACIÓN

En esta sección presento los sistemas de recomendación basados en críticas categorizados de acuerdo a su entorno de visualización: Entornos de Visualización en dos dimensiones y en tres dimensiones.

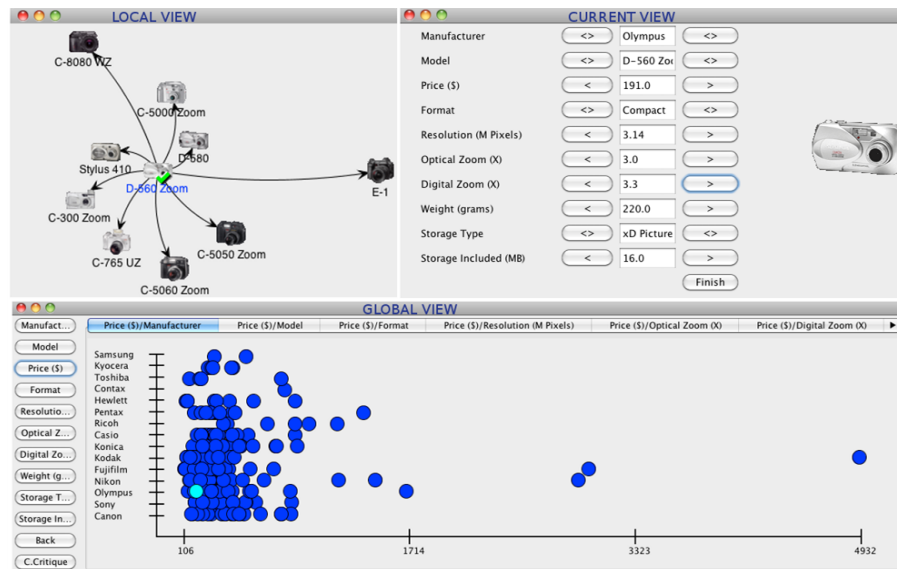
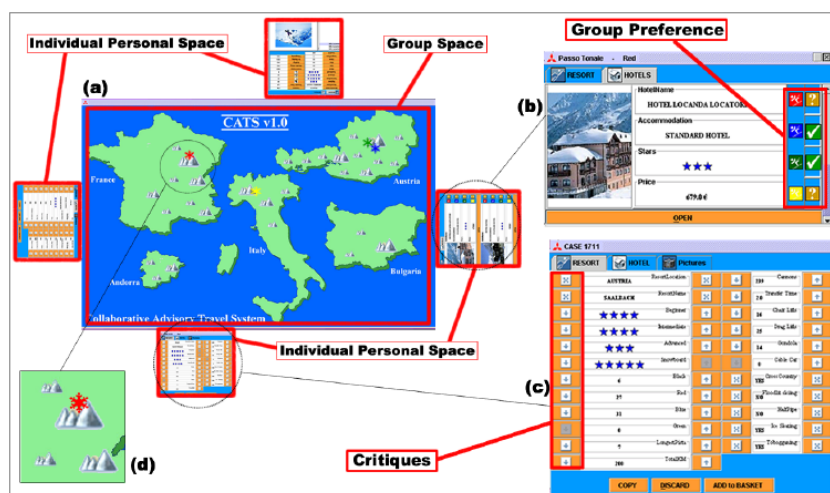


Figura 6: Interfaz gráfica para la generación de críticas compuestas a través de la selección de productos.

#### 2.4.1 Entornos de visualización en dos dimensiones

La mayoría de los sistemas de recomendación conversacionales utilizan plataformas web en dos dimensiones (Zhang and Pu, 2006; Pu et al., 2008; Chen and Pu, 2009; Salamó et al., 2013) para visualizar e interactuar con el usuario. Véanse los ejemplos de interfaces en dos dimensiones en las Figuras 2, 4, 5a, 5b) para recomendar productos a un solo usuario. Este tipo de plataformas son las más ampliamente usadas en sitios de comercio electrónico. Sin embargo, existen algunas propuestas que implementan recomendadores en otras plataformas de visualización, tal es el caso del recomendador para grupos conocido por su nombre en inglés *Collaborative Advisory Travel System (CATS)* (McCarthy et al., 2006c). Es importante destacar que en un sistema de recomendación para grupos todos los usuarios tienen un objetivo común y colaboran para alcanzar dicho objetivo. En el caso particular del sistema *CATS*, se cuenta con un entorno de visualización web en dos dimensiones (McCarthy et al., 2006b; McCarthy et al., 2006a) (véase la Figura 7a) y un entorno basado en una mesa interactiva que permite la utilización del sistema de recomendación por parte de múltiples usuarios a través de una pantalla táctil (McCarthy et al., 2006c) (véase la Figura 7b).

En los últimos años, con la masificación de los teléfonos móviles inteligentes, han surgido algunas propuestas que aprovechan las ventajas de estos dispositivos móviles para implementar nuevos entornos de visualización. La mayor parte de estas investigaciones utilizan técnicas de filtrado colaborativo y basados en contenidos (Kim et al., 2004; Ahn et al., 2006; Schifanella et al., 2008) o recomendaciones basadas en el contexto (*context-aware*) (Van Setten et al., 2004; Horozov et al., 2006; Unger et al., 2016; Horowitz et al., 2016). De estos enfoques, la única propuesta que utiliza un sistema de recomendación basado en críticas corresponde al sistema *MobyRek*, el cual realiza recomendaciones de restaurantes utilizando un teléfono móvil bajo una interfaz gráfica muy básica para la interacción de los usuarios (Ricci and Nguyen, 2005; Ricci and Nguyen, 2007) pero con algunas limitaciones de usabilidad



(a) Interfaz web del sistema CATS



(b) Interacción de los usuarios utilizando la mesa interactiva del sistema CATS

Figura 7: Entornos de visualización del sistema CATS

propias de la tecnología de los móviles disponibles en ese momento (véase la Figura 8).

Uno de los principales inconvenientes del recomendador *MobyRek*, corresponde a la baja percepción de proximidad en la ubicación del usuario respecto al restaurante recomendado. Una extensión de este recomendador es la interfaz denominada *MapMobyRek* (Averjanova et al., 2008) mostrado en la Figura 9, la cual está basada en mapas para solucionar el problema de proximidad en la versión anterior (*MobyRek*).

#### 2.4.2 Entornos de visualización en tres dimensiones

Hoy en día, con la evolución de la Web 2.0 y los avances en tecnologías gráficas y de redes, existe una transición desde la visualización en dos dimensiones hacia la representación de contenidos en tres dimensiones. Lo

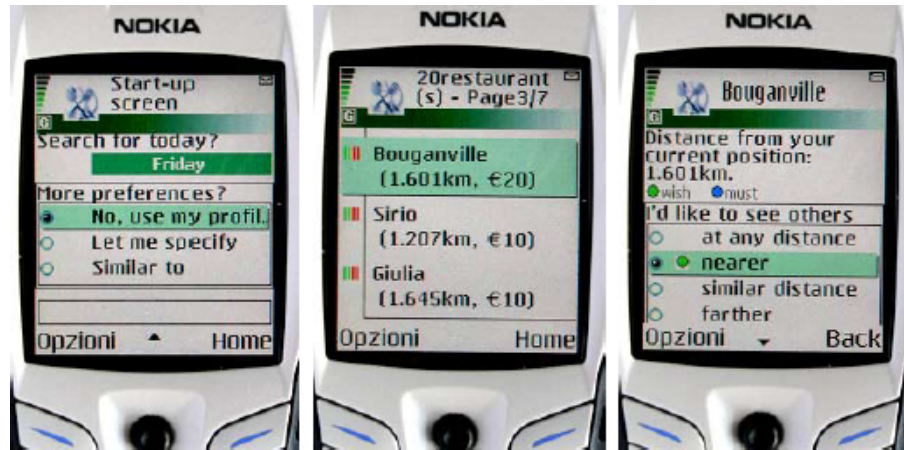


Figura 8: Interfaz de usuario del recomendador MobyRek.

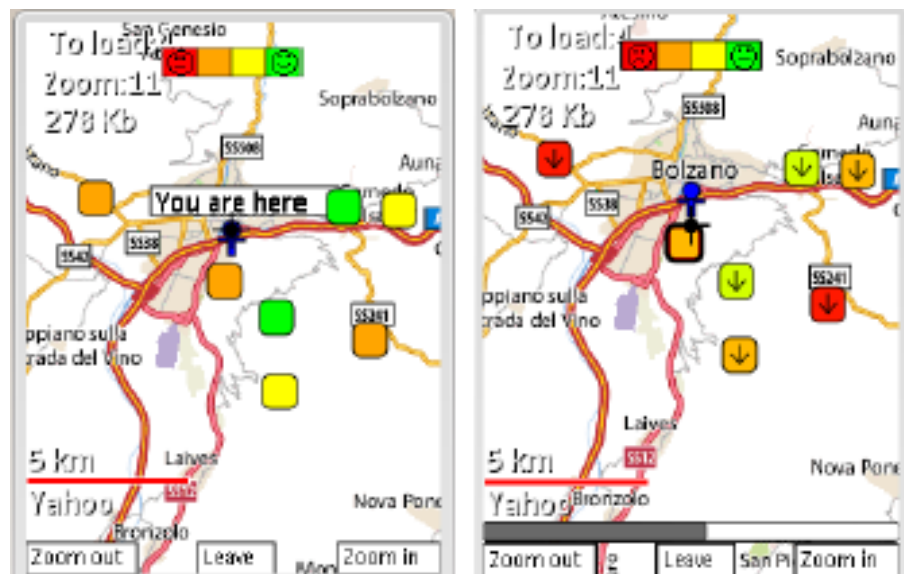


Figura 9: Interfaz de usuario del recomendador MapMobyRek.

anterior, podría beneficiar nuevos entornos de comercio electrónico donde los usuarios puedan compartir experiencias con otros usuarios para tomar la mejor decisión de compra (Trattner et al., 2014). En este contexto, existen en la actualidad entornos virtuales en tres dimensiones, que benefician la colaboración y comunicación entre usuarios. En estos entornos virtuales en tres dimensiones (comúnmente conocidos por su nombre en inglés, *3D Virtual Environments*), los usuarios son representados por avatares que se encuentran inmersos dentro de un mundo alternativo al mundo real, en dicho lugar pueden interactuar con otros usuarios a través de gestos, conversaciones tanto por voz como por texto (Lau et al., 2013; Bainbridge, 2010).

El objetivo inicial de los entornos virtuales en tres dimensiones dedicados al comercio electrónico fue generar reconocimiento de marcas específicas de productos e incrementar el tráfico de los usuarios hacia portales *e-commerce* de dos dimensiones. Sin embargo, en los últimos años los usuarios prefieren mantenerse en estos entornos virtuales y esperan completar su proceso de compra en el mismo entorno virtual (Messinger et al., 2009). La Figura 10

muestra un típico escenario de tiendas virtuales en un entorno virtual de tres dimensiones.



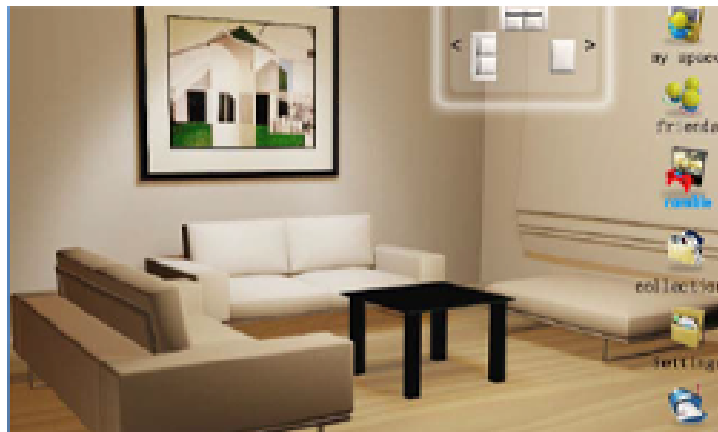
Figura 10: Entorno de visualización en un mundo virtual de tres dimensiones.

Teniendo en cuenta la evolución de 2D a 3D, han surgido algunas propuestas de recomendadores que utilizan entornos de visualización en tres dimensiones. A continuación presento las principales propuestas existentes de recomendadores en ambientes virtuales en tres dimensiones teniendo en cuenta diferentes enfoques: el dominio de aplicación, los métodos de recomendación y la plataforma de desarrollo.

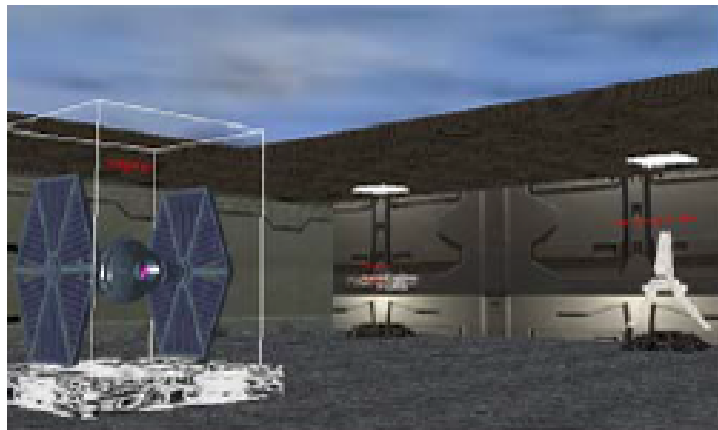
- Desde el punto de vista del dominio de las investigaciones previas, la mayoría de ellas corresponden a asistentes de venta virtuales para entornos de comercio electrónico. Estos agentes virtuales pueden proveer servicios personalizados que incrementen la confianza en el comercio electrónico por parte de los usuarios (Holzwarth et al., 2006). Una primera aproximación para este tipo de recomendadores lo presentan los autores Xu y Yu, quienes proponen una solución de un centro comercial virtual donde se recomiendan una lista de productos en base a cálculos de similitud entre las preferencias del usuario y los productos disponibles para recomendar (Xu and Yu, 2010). Recientemente, un nuevo estudio ha propuesto evaluar el nivel de las preferencias de los usuarios antes de realizar una compra, por medio de las emociones positivas que pueda emitir un usuario, las cuales son medidas a través de las señales extraídas mediante un Electroencefalograma (Guo and Elgendi, 2013) (véase un ejemplo del entorno virtual en la Figura 11a). En un dominio distinto, otros autores se focalizaron en recomendar objetos y lugares dentro de entornos virtuales en tres dimensiones, tales como entornos de realidad virtual y mundos virtuales (véase la Figura 11b) (Hu and Wang, 2010; Shah et al., 2010). El primero de estos trabajos (Hu and Wang, 2010), tiene como objetivo recomendar recursos 3D en un entorno de realidad virtual. Estas recomendaciones se realizan en base a la similitud entre los recursos disponibles y las preferencias del usuario. Además, el recomendador sugiere recursos que otros usuarios similares han seleccionado. En el segundo trabajo (Shah et al., 2010), los autores proponen recomendar lugares que el usuario puede visitar dentro de un mundo virtual en tres dimensiones. En es-



(a) Recomendaciones en tiendas virtuales



(b) Recomendaciones de lugares virtuales



(c) Recomendaciones en museos virtuales

Figura 11: Ejemplos de entornos virtuales integrados con sistemas de recomendación.

te trabajo, la recomendación se realiza teniendo en cuenta los *ratings* definidos por los propios usuarios. Los lugares recomendados serán aquellos con un *rating* más similar al *rating* que ha definido el usuario para los lugares visitados. En un dominio totalmente distinto, hay propuestas de recomendadores en el ámbito de la cultura, que ayudan



a los usuarios en la navegación dentro de museos y galerías virtuales en tres dimensiones (Oberlander et al., 2008; Bonis et al., 2009) (véase un ejemplo de este tipo de interfaz en la Figura 11c). Uno de estos trabajos utiliza mecanismos de lenguaje natural para entregar explicaciones de los objetos del museo que muestra al usuario (Oberlander et al., 2008). Este recomendador tiene la capacidad de refinar las explicaciones que realiza al usuario teniendo en cuenta los objetos que ha recomendado previamente y la ubicación física del avatar dentro del entorno virtual. Otro de los trabajos en el ámbito de museos corresponde a la propuesta denominada *PeVEP* (Bonis et al., 2009). En este entorno el recomendador recoge información implícita de las acciones que realiza el usuario dentro del mundo virtual, para generar un perfil de usuario. Con el perfil del usuario, el recomendador sugiere comunidades de otros usuarios con preferencias similares para que el usuario pueda interactuar.

- Desde el punto de vista de los métodos de recomendación, la mayoría de los trabajos previos utilizan técnicas de filtrado colaborativo (CF) (Oberlander et al., 2008; Guo and Elgendi, 2013) y técnicas de filtrado basado en contenido (CB) para realizar las recomendaciones (Bonis et al., 2009). Sin embargo, también se han empleado métodos de clasificación híbridos que utilizan técnicas de filtrado colaborativo, técnicas de filtrado basado en contenido y recomendaciones basadas en utilidad para realizar la recomendación (Hu and Wang, 2010). En particular en este trabajo, la técnica de filtrado basado en contenidos se utiliza para recomendar recursos virtuales disponibles comparando el perfil del usuario con el perfil de los recursos. Por otro lado, la técnica de filtrado colaborativo recomienda recursos virtuales de acuerdo a la similitud entre el perfil del usuario y el perfil de otros usuarios a los cuales se han recomendado recursos anteriormente. Finalmente, estos autores utilizan la técnica de recomendaciones basadas en utilidad para calcular la eficiencia que obtienen los usuarios cuando utilizan el sistema dentro del entorno de realidad virtual, con el objetivo de incluir esta medida dentro de los parámetros de recomendación del sistema. En el trabajo presentado por Xu y Yu, se utiliza la técnica de minería de datos denominada *K Nearest Neighbor* para calcular la similitud entre el usuario y los productos disponibles para recomendar (Xu and Yu, 2010). Además, este trabajo utiliza reglas de asociación para seleccionar, de los productos más similares, el producto a recomendar. En el caso del trabajo descrito en (Shah et al., 2010), la recomendación se realiza en base *clusters*<sup>13</sup> que segmentan las ubicaciones dentro del mundo virtual utilizando los *ratings* definidos por los usuarios, para finalmente realizar la recomendación comparando la similitud de los *ratings* del usuario con los *ratings* de los *clusters*.
- Desde el punto de vista de la plataforma de desarrollo, por un lado se han utilizado plataformas de desarrollo basadas en protocolo Second Life<sup>14</sup> (Oberlander et al., 2008; Bonis et al., 2009; Shah et al., 2010; Guo and Elgendi, 2013). Por otro lado, otras propuestas han sido

13 Un *cluster* está compuesto por objetos similares entre ellos y que tienen diferencias con otros objetos de *clusters* distintos.

14 Second Life es un entorno virtual *on-line* en tres dimensiones de afluencia masiva de usuarios, en el cual éstos pueden construir, interactuar y habitar su propio mundo virtual 3D.

desarrolladas en plataformas JAVA, específicamente en JAVA 3D con VRML<sup>15</sup> (Hu and Wang, 2010; Xu and Yu, 2010).

Aunque, de acuerdo a lo descrito anteriormente, existen a la fecha algunos enfoques que proponen la utilización de nuevos entornos de visualización integrados con sistemas de recomendación para facilitar la comunicación, interacción y colaboración entre usuarios, tales como los mundos virtuales, ninguna de estas investigaciones previas está basada en el uso e integración de recomendadores conversacionales (Bridge, 2002; Bridge et al., 2005).

## 2.5 ASPECTOS COLABORATIVOS

La colaboración y la cooperación son distintivos propios de la naturaleza humana (Sussman and Cloninger, 2011). En general, la colaboración se define como la realización conjunta de un trabajo o tarea para alcanzar un objetivo individual. Por otro lado, la cooperación se define como la acción de colaborar para alcanzar un objetivo común entre los cooperantes.

Algunos trabajos psicológico-sociales aplicados al marketing y últimamente al ámbito del *e-commerce*, concluyen que los consumidores necesitan algún tipo de ayuda poco formal para decidir el producto que comprarán, tales como las opiniones de otras personas que se transmiten de “boca en boca” (Burnkrant and Cousineau, 1975; Vachon, 2011). Estudios más recientes (Nielsen, 2014), concluyen que aproximadamente 84% de los consumidores confían en el “boca en boca”, cuando realizan sus compras. Además, desde el punto de vista de la teoría de juegos (Butnaru, 2009), las personas generalmente buscan su propio interés, sin embargo el comportamiento colaborativo del ser humano puede facilitar la búsqueda, por parte de los jugadores, de la mejor estrategia grupal para alcanzar los objetivos individuales.

La colaboración en los sistemas de recomendación se da de diferente manera según el tipo de recomendador. En este sentido es posible distinguir entre recomendadores basados en críticas para grupos de usuarios y para usuarios individuales.

- Los primeros se focalizan en grupos de usuarios que buscan un producto en común con preferencias individuales donde la colaboración y la cooperación es necesaria para llegar al consenso. En un sistema de recomendación para grupos se utiliza un perfil individual con las preferencias de cada usuario y un perfil grupal que corresponde a la combinación de los perfiles individuales. De esta manera, las sugerencias del recomendador tienen en cuenta tanto las preferencias personales de cada usuario, como las preferencias de todo el grupo. Un ejemplo de este tipo de sistema recomendador para grupos lo representa el recomendador *CATS* (McCarthy et al., 2006b; McCarthy et al., 2006c; McCarthy et al., 2006a), el cual está focalizado en la interacción *on-line* y sincronizada de múltiples usuarios por medio de una pantalla interactiva grupal que permite la colaboración y el *awareness*<sup>16</sup>. *CATS* provee sugerencias de productos que reúnen las preferencias de todos

<sup>15</sup> Virtual Reality Markup Language, es un lenguaje estándar para objetos o escenas en tres dimensiones que se transmiten a través de Internet (Bell et al., 1995).

<sup>16</sup> El término *awareness* en sistemas de recomendación para grupos, se utiliza para referirse al hecho de mantener informados en cada momento a los usuarios respecto al estado del resto de los integrantes del grupo.

los usuarios y permite conocer las preferencias entre usuarios que tienen un objetivo común. La colaboración se promueve por medio de sugerencias entre usuarios, ofreciendo así un entorno donde se puede realizar consenso respecto a las decisiones que se tomarán como grupo. Estas decisiones se basarán en las sugerencias que se reciban de parte del recomendador. Otro ejemplo de recomendador para grupos es el denominado *Travel Decision Forum* (Jameson et al., 2004). Este recomendador de vacaciones, permite realizar interacciones *off-line* para usuarios que no pueden estar conectados de manera síncrona en un momento determinado, es decir, los usuarios pueden interactuar con el recomendador y sus preferencias se guardan para que puedan ser visualizadas por el resto del grupo.

- En el caso de los recomendadores individuales, la orientación es hacia objetivos propios y no grupales. Este tipo de recomendadores también hacen uso de la colaboración para alcanzar sus objetivos, pero desde una perspectiva *off-line*. Lo anteriormente expuesto, significa que aunque no se utilice colaboración instantánea (*on-line*), estos sistemas hacen uso de información histórica proveniente de otros usuarios para recomendar productos a un usuario. En la literatura, existen algunos sistemas de recomendación individuales que utilizan información colaborativa *off-line* proveniente de valoraciones (*rating*) hechas por otros usuarios sobre los productos disponibles (Vasudevan and Chakraborti, 2014; Ricci and Nguyen, 2007; Pu et al., 2008; Zhang et al., 2008). Otro tipo de información colaborativa que denomino *off-line* es la proveniente de otros usuarios que han utilizado el sistema de recomendación con anterioridad. En particular, algunos autores utilizan información de este tipo que obtienen de sesiones de recomendaciones previas que han realizado otros usuarios. El objetivo de estas propuestas es aprovechar la información de las críticas realizadas por otros usuarios en sesiones previas para mejorar la eficiencia en la recomendación (McCarthy et al., 2010; Mandl and Felfernig, 2012; Salem and Hong, 2013; Salem et al., 2014). Este tipo de recomendadores se detallan en la Sección 2.6.

Aun cuando, los trabajos analizados en el presente apartado favorecen la colaboración entre usuarios, ninguno de ellos considera la posibilidad que usuarios individuales, que no se encuentran trabajando en grupo, puedan colaborar de forma instantánea con un objetivo individual (colaboración *on-line*). Solo el recomendador CATS lo permite, pero su objetivo son las recomendaciones grupales a un grupo que inherentemente ha de colaborar para llegar a un consenso. De esta forma, CATS solo permite colaborar por medio de las sugerencias que pueda hacer un usuario a otro respecto a un producto. Esta carencia cobra mayor importancia en los últimos años donde las redes sociales y entornos virtuales entregan al usuario mayores posibilidades de interacción y colaboración *on-line* con otros usuarios. Con respecto a esto, CATS tampoco permite una inmersión de los usuarios en un entorno colaborativo *on-line*, en el cual los usuarios puedan interactuar por gestos o mensajes de voz.

## 2.6 EFICIENCIA EN LA RECOMENDACIÓN

Recientemente, una gran parte de las investigaciones en los recomendadores basados en críticas se han centrado en mejorar la eficiencia del re-

comendador a través de diferentes mecanismos. Con el fin de mejorar la comprensión en el análisis que realizo de las mejoras en eficiencia de los recomendadores, he clasificado los trabajos previos en: (1) algoritmo base de *critiquing*; (2) mejoras en algoritmos bases de *critiquing*; (3) integración de contenidos generados por otros usuarios en los algoritmos de recomendación y (4) integración de información colaborativa obtenida de sesiones de recomendación históricas. A continuación detallo cada una de ellas.

1. El sistema *FindMe* fue el primer sistema basado en críticas, conocido como *standard critiquing* (Burke et al., 1997). Este sistema definió las bases de los sistemas de recomendación basados en críticas. Brevemente, *standard critiquing* recomienda un producto teniendo en cuenta aquellos productos candidatos que satisfacen la última crítica realizada por el usuario. Es decir, el recomendador no toma en cuenta el proceso completo de recomendación, ya que solo usa la última crítica para recomendar el siguiente producto, sin considerar todas las críticas que el usuario pudo haber realizado anteriormente. Esto implica que *standard critiquing* no está considerando todo el conocimiento que el usuario ha adquirido durante la sesión de recomendación. Posteriormente, se selecciona aquel producto que tenga mayor similitud con el producto actualmente recomendado. Uno de los inconvenientes de esta estrategia es que la similitud del producto actual con el producto que busca el usuario no necesariamente aumenta a medida que el usuario interactúa con el recomendador. Inclusive, cuando un usuario aplica una crítica, la nueva recomendación podría ser menos similar al producto requerido por el usuario, que el producto actual recomendado.
2. Posteriormente, han surgido nuevas propuestas para mejorar la eficiencia de *Standard Critiquing*. Tal es el caso del algoritmo conocido por su nombre en inglés *Incremental Critiquing* (IC) (Reilly et al., 2005a), el cual mejora significativamente la eficiencia en la recomendación con respecto a *Standard Critiquing*, por medio de un registro histórico de críticas realizadas por el usuario durante la sesión de recomendación. Concretamente, IC registra las críticas en un modelo de usuario, el cual influye en la determinación de las futuras recomendaciones, a diferencia de *Standard Critiquing* que solo utilizaba la última crítica. Al final de cada ciclo de recomendación, se agrega al modelo de usuario la crítica realizada por el usuario. De esta manera, en el momento de realizar una nueva recomendación, se tienen en cuenta tanto la similitud entre el producto candidato y el producto actualmente recomendado, como también la compatibilidad que puede tener un producto candidato con respecto al modelo de usuario. La compatibilidad mide la cantidad de críticas del modelo de usuario que son compatibles con el producto candidato en evaluación. Mayores detalles del enfoque de *Incremental Critiquing* se entregan en la Sección 3.3 de esta tesis. Otros estudios (Salamó et al., 2005; Salamó and Escalera, 2012), han presentado nuevas técnicas de ponderación de características de los productos (método tradicionalmente conocido por su nombre en inglés, *weighting*) y un conjunto de medidas de compatibilidad basadas en *Reinforcement Learning*<sup>17</sup> para mejorar el proceso de recomendación respecto al tradicional enfoque de *Incremental Critiquing*. Estas técnicas son definidas formalmente en la Sección 3.5.7 de esta tesis.

<sup>17</sup> Las técnicas de Reinforcement Learning permiten a un agente inteligente aprender a través de pruebas de ensayo y error en un ambiente dinámico (Kaelbling et al., 1996).

3. Recientemente, algunos investigadores han centrado sus estudios en la utilización de contenidos generados por usuarios<sup>18</sup> para mejorar la eficiencia de la recomendación. Concretamente, en redes sociales y sitios web de comercio electrónico, los usuarios expresan sus opiniones a través de comentarios en lenguaje natural o valoraciones de productos (*rating*). Específicamente, en la literatura existen estudios cuyo objetivo es analizar los comentarios de los productos que realizan los usuarios y poder extraer nuevas características perceptivas<sup>19</sup> de los productos que representan la percepción de los usuarios frente a un producto en particular (Dong et al., 2013b). En este trabajo los autores proponen utilizar técnicas de procesamiento de lenguaje natural para extraer la información desde las opiniones textuales de los usuarios y métodos estadísticos para convertir dicha información en las características perceptivas que más representen las opiniones emitidas por los usuarios. Otros estudios, intentan profundizar aún más en esta nueva forma de información y proponer mecanismos para extraer sentimientos desde las opiniones de los usuarios respecto a un producto (Dong et al., 2013c; Dong et al., 2013a). En particular, estos trabajos utilizan una versión de *opinion pattern mining* para evaluar sentimientos en las características obtenidas desde las opiniones de los usuarios. La técnica *opinion pattern mining* (Moghaddam and Ester, 2010), permite clasificar cada característica extraída como un sentimiento positivo, negativo o neutral, dependiendo si contiene términos (palabras) positivos, negativos o neutrales. Aunque, los trabajos descritos anteriormente buscan recomendar productos en base a la información que es posible extraer utilizando distintos mecanismos de minería de datos, éstos no consideran las preferencias del usuario durante el proceso de recomendación. Posteriormente, se ha presentado una propuesta que utiliza el conocimiento de los sentimientos de usuarios y estadísticas de compras en la descripción de los productos (Chen et al., 2014). Este último trabajo ha sido ampliado para aprovechar conocimiento adicional desde los comentarios realizados por los usuarios, introduciendo un nuevo concepto que los autores denominan *aspect-based sentiment analysis* (Chen et al., 2015). Este concepto permite identificar frases que representan un aspecto del producto y clasificarlas de acuerdo a las apariciones dentro de los comentarios de los usuarios. Finalmente, los productos candidatos son categorizados de acuerdo a los sentimientos y la medida del aspecto de las frases presentes en las opiniones de los usuarios. El uso de contenidos generados por usuarios también ha sido utilizado en estudios que buscan mejorar las explicaciones que se generan para que el usuario pueda entender una determinada recomendación (Muhammad et al., 2015).
4. En los últimos años, nuevos enfoques se han centrado en usar sesiones de críticas pasadas (históricas), realizadas por otros usuarios para mejorar la eficiencia en la recomendación. Por ejemplo, *Experience-Based Critiquing* (EBC) usa las experiencias de críticas realizadas por usuarios previos como una nueva fuente de conocimiento para realizar recomendaciones. En concreto, EBC utiliza como productos candidatos

<sup>18</sup> Los contenidos generados por el usuario corresponden a cualquier forma de contenido que es creado por el usuario en un servicio *on-line*, los cuales a menudo son realizados en sitios web sociales o de comercio electrónico.

<sup>19</sup> Las características perceptivas también se conocen como características no técnicas y habitualmente se asocian a un grupo de características técnicas. Las características técnicas son aquellas que describen un producto (por ejemplo, marca, modelo, precio, etc.)

para recomendar, aquellos productos que fueron finalmente seleccionados por otros usuarios (productos finales aceptados por el usuario en cada sesión histórica) y cuya sesión o secuencia de críticas realizadas es más similar a la secuencia de críticas realizada por el usuario actual. En EBC, una sesión de críticas pasadas es seleccionada si las críticas contenidas en dicha sesión se solapan suficientemente con las críticas que el usuario ha realizado en la sesión actual. Otro enfoque denominado *Nearest Neighbor Compatibility Critiquing* (Mandl and Felfernig, 2012) adopta el enfoque EBC, con la diferencia que en esta propuesta el producto candidato no necesariamente debe ser un producto final aceptado, sino que puede ser un producto recomendado intermedio en la sesión histórica. Otra propuesta llamada *History-Aware Critiquing-based Conversational Recommendation* (HAC) (Salem and Hong, 2013), considera la similitud entre los productos actualmente recomendados y los productos recomendados en las sesiones históricas, teniendo en cuenta además el solapamiento entre críticas utilizado en EBC. Una mejora de HAC corresponde a la propuesta denominada *History-Guided Recommendation* (HGR) (Salem et al., 2014). En HGR, la selección de sesiones similares se realiza teniendo en cuenta el producto recomendado y la crítica realizada como un conjunto indivisible. Es decir, en HGR una sesión pasada puede ser seleccionada si el par recomendación-crítica está suficientemente solapado con la sesión actual. De manera alternativa, existe una propuesta basada en sesiones históricas que utiliza críticas compuestas y que tiene como base la teoría de grafos para la identificación de sesiones históricas similares (Xie et al., 2014).

Si bien es cierto que hasta la fecha existen muchas propuestas para mejorar la eficiencia en los recomendadores, éstas se mantienen disgregadas en la literatura y no existe una propuesta que integre más de una alternativa presente en los últimos avances en el ámbito de los recomendadores basados en críticas.

## 2.7 CONCLUSIONES

En este capítulo he revisado los antecedentes que forman la base de los sistemas de recomendación, con un enfoque orientado en aquellos que utilizan las críticas como mecanismo de retroalimentación de parte del usuario. He descrito además las principales fortalezas con las que cuentan este tipo de sistemas respecto a los enfoques tradicionales de filtrado colaborativo y recomendadores basados en contenido.

Teniendo en cuenta la diversidad de enfoques propuestos en investigaciones previas, he categorizado el estudio del estado del arte en los aspectos que a mi juicio segmentan de mejor forma los trabajos previos. La principal clasificación de los recomendadores basados en críticas la representa el tipo de mecanismos de retroalimentación que utilizan. Lo anterior, se sustenta en que esta fase del proceso de recomendación es extremadamente importante desde el punto de vista del conocimiento acerca del perfil y las preferencias del usuario. Las siguientes tres clasificaciones (Entornos de Visualización, Aspectos Colaborativos y Eficiencia en la Recomendación) representan los lineamientos actuales y futuros en el campo de los sistemas de recomendación.

En referencia a los entornos de visualización y los aspectos colaborativos, la mayoría de las investigaciones previas son propuestas basadas en entornos en dos dimensiones, que generalmente no favorecen la colaboración por parte de los usuarios. Aun cuando, existen algunas aproximaciones que utilizan entornos que por naturaleza fomentan la colaboración e interacción entre usuarios, tales como los entornos virtuales en tres dimensiones, ninguno de ellos utiliza sistemas basados en críticas. Además, los trabajos previos solo se limitan a implementar métodos de recomendación de filtrado colaborativo (colaboración *off-line*) y basados en información histórica (colaboración *off-line*).

Respecto a la eficiencia en la recomendación, no existe actualmente un análisis del impacto que tendría la integración de varias investigaciones previas en una nueva propuesta que evite la disgregación actual de técnicas y algoritmos propuestos en la literatura.

Por el contrario, a las investigaciones previas, mi propuesta en primer lugar busca fomentar la integración de trabajos previos y además analizar nuevos mecanismos para mejorar la eficiencia de los recomendadores. En segundo lugar, mi propuesta está enfocada en aprovechar al máximo la interacción y colaboración entre usuarios en entornos *on-line* de tres dimensiones, con el fin de extraer conocimiento desde la información colaborativa *on-line* (proveniente del comportamiento de los usuarios en un entorno virtual y social) y *off-line* (proveniente de opiniones y sesiones históricas de otros usuarios) para mejorar la eficiencia en las recomendaciones.

## 3

## TÉCNICAS DE RECOMENDACIÓN

## Resumen

En este capítulo describo las nuevas propuestas de personalización, las cuales están basadas en incorporar el comportamiento colaborativo de las personas para mejorar las recomendaciones. Estas técnicas favorecen el uso de información colaborativa durante las sesiones de recomendación. En primer lugar, presento las definiciones esenciales que se requieren para una correcta comprensión de las técnicas de personalización propuestas. En segundo lugar, describo los algoritmos de recomendación que sirvieron como base para las nuevas técnicas de personalización. En tercer lugar, categorizo las técnicas de personalización propuestas y posteriormente, describo en detalle cada una de las técnicas propuestas. En concreto, en cada una de las técnicas propuestas presento los antecedentes, los fundamentos conceptuales, teóricos y matemáticos que la sustentan. Finalmente, presento las conclusiones del capítulo. Los resultados obtenidos a través de las técnicas propuestas se encuentran publicados en:

- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2013). Integrating a Conversational Recommender System within a 3D Collaborative Environment. In *Proceedings of the XV Conference of Spanish Association for Artificial Intelligence*, pages 89–98
- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2014). An approach to improve user experience with conversational recommenders through a 3D virtual environment. In *Proceedings of the XV International Conference on Human Computer Interaction*, pages 403–406
- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2015b). A 3D visual interface for critiquing-based recommenders: Architecture and interaction. *International Journal of Artificial Intelligence and Interactive Multimedia*, 3(3):pages 7–15
- Contreras, D. and Salamó, M. (2015). On the Use of User-generated Content in Critiquing Recommendation. In *Proceedings of the XVIII International Conference of the Catalan Association for Artificial Intelligence*, pages (195–204)
- Contreras, D. and Salamó, M. (2017). Integrating History and User Opinion for Conversational Recommendations. *ACM Transactions on Intelligent Systems and Technology*. Manuscript submitted for revision in July 2016. JCR IF=2.414

## 3.1 INTRODUCCIÓN

HABITUALMENTE, LAS PERSONAS EN SU VIDA COTIDIANA interactúan a diario con muchas otras personas. Esta situación no es diferente en lugares físicos de compras, donde las personas generalmente preguntan, conversan y colaboran para encontrar y adquirir el producto que sea de su mayor conveniencia. Esta interacción, en la mayoría de los casos, resulta muy enriquecedora, tal y como lo avalan estudios psicológicos y sociales asociados



principalmente al marketing y al comercio electrónico (Nielsen, 2014; Burnkrant and Cousineau, 1975; Vachon, 2011). Estos estudios concluyen que los clientes necesitan muchas veces ayuda en sus decisiones de compra. Esta ayuda proviene en muchos casos de las opiniones que se transmiten de boca en boca por parte de otros clientes (Burnkrant and Cousineau, 1975; Vachon, 2011). Con el objetivo de proveer al cliente de ayuda de otros usuarios, hoy en día, hay un crecimiento sostenido de sitios de comercio electrónico que basan su funcionamiento en aspectos sociales y colaborativos, donde los usuarios pueden contribuir con opiniones, comentarios y valoraciones acerca de un producto. Este nuevo tipo de información es comúnmente conocida por su nombre en inglés, *user-generated content* (contenidos generados por los usuarios). En concreto, en las redes sociales y sitios de comercio electrónico las personas expresan sus opiniones, ya sea en lenguaje natural o por medio de valoraciones de los productos, conocidas generalmente por el término en inglés de *ratings*.

Teniendo en cuenta los antecedentes previos, las técnicas de recomendación que propongo incorporan la naturaleza colaborativa del ser humano y su capacidad de compartir información, opiniones y sugerencias con el resto de las personas. Mi propuesta se sustenta tanto en estudios previos, como en la tendencia colaborativa de las principales plataformas de comunicación e interacción disponibles en Internet, tales como redes sociales, sitios de comercio electrónico, blogs, etc.

Específicamente, las técnicas que propongo utilizan dos mecanismos de captura de información colaborativa, *on-line* y *off-line*, con el objetivo de obtener un mayor conocimiento de las preferencias de los usuarios. La información *on-line* proviene del comportamiento de los usuarios en entornos virtuales y la información *off-line* proviene de opiniones y sesiones históricas de otros usuarios. Adicionalmente mi propuesta, busca fomentar la integración de trabajos previos y además analizar nuevos mecanismos para mejorar la eficiencia de los recomendadores.

El capítulo se encuentra estructurado de la siguiente forma: primero se presentan las definiciones esenciales para una correcta comprensión de las técnicas de recomendación propuestas; en segundo lugar presento los algoritmos a partir de los cuales se han desarrollado las técnicas propuestas; posteriormente presento en detalle cada una de las técnicas de recomendación propuestas; y finalmente presento las conclusiones del capítulo.

## 3.2 DEFINICIONES ESENCIALES

**Definición 1. Base de Datos de Productos (Case Base).** Defino la Base de Datos de Productos, CB, como un conjunto de productos disponibles para recomendar, descrito como  $CB = \{p_1, \dots, p_n\}$ , donde  $p_i$  es el  $i$ -ésimo producto y el conjunto de características que describe a cada producto está definido como  $F = \{f_1, \dots, f_m\}$ .

Adicionalmente, se define una característica  $f_s$  de un producto  $p_i$  como  $p_i^{f_s}$ .

Por ejemplo, en la base de productos CB de SMARTPHONE el producto  $p_{757}$  está definido por las siguientes características  $F = \{Sony, XperiaC, 153, 5.0, 4, 1024, 8, Android OS, v4.2.2, Quad-core 1.2 GHz, 300, 29, 7.8, 7.6, 7.6\}$ . Para este producto, las características  $p_i^{f_1}$  y  $p_i^{f_2}$  representan a la primera y segunda característica en el producto, es decir  $p_i^{f_1} = Sony$  y  $p_i^{f_2} = XperiaC$ .

**Definición 2. Base de Datos de Sesiones (Session Base).** Defino la Base de Datos de Sesiones históricas, SB, como un conjunto histórico de sesiones de recomendación de otros usuarios. Una sesión de recomendación comprende todo el proceso que se lleva a cabo desde el momento que el usuario inicia la interacción con el recomendador, hasta que compra un producto o abandona el proceso voluntariamente. Específicamente, esta base de datos se define como  $SB = \{s_1, \dots, s_l\}$ , donde  $s_i$  es una sesión definida como una secuencia de pares  $(r_i, c_i)$  compuestos por un producto recomendado y una crítica sobre el producto recomendado (véase la Definición 4). Además, cada sesión finaliza en un producto terminal, que corresponde al producto finalmente aceptado por el usuario de esa sesión, este producto es definido como  $term(s_i)$ .

Por ejemplo, para el dominio de productos SMARTPHONE una sesión de recomendación histórica podría corresponder a  $s_3 = \{\{Nokia Lumia1320, (price, <, \$220)\}, \{Nokia Asha501, (Manufacturer, <>, Nokia)\}, \{Samsung Galaxy Express, (Model, <>, Galaxy Express)\}, \{Samsung Galaxy SIII\}$ . Siendo el producto Samsung Galaxy SIII el producto finalmente seleccionado por el usuario,  $term(s_3)$ . Para simplificar, en el ejemplo nuestro solo la marca y modelo del producto recomendado, sin embargo, en una sesión real se guardan el producto con todas sus características.

**Definición 3. Base de Datos de Opiniones (Opinion Base).** Defino la Base de Datos de Opiniones, OB, como un conjunto de opiniones de usuarios. Concretamente, esta base se define como  $OB = \{o_1, \dots, o_n\}$ , donde  $o_i$  corresponde a la opinión de los usuarios respecto al  $i$ -ésimo producto. En mi propuesta, una opinión de usuario se describe como el promedio del *rating* que varios usuarios otorgan a un producto, sin embargo las opiniones pueden ser obtenidas desde los usuarios por medio de varios métodos comúnmente conocidos como contenidos generados por los usuarios o por su nombre en inglés *user-generated content*. Estos métodos pueden corresponder a *ratings* explícitos que los usuarios pueden realizar, comentarios de productos por parte de los usuarios, opiniones textuales en lenguaje natural o cualquier otro método que permita a los usuarios generar una valoración de un producto determinado.

Por ejemplo, en el dominio de SMARTPHONE las opiniones de los usuarios están asociadas a las características que el usuario percibe respecto a *design*, *performance* y *general features* del producto. En particular, para este dominio los usuarios expresan sus opiniones por medio de *ratings* explícitos para cada producto. Un ejemplo de las opiniones para el producto  $p_3$  podría estar definido por  $o_3 = \{5.2, 5.4, 6.0\}$ , donde por ejemplo, los usuarios han valorado en promedio con 5.2 el diseño del producto (*design*).

**Definición 4. Crítica.** Una crítica  $c_i$ , se representa como una tupla definida por  $(f_i, type_i, v_i)$ , donde  $f_i$  corresponde a una característica del producto recomendado  $r_i$ , por otro lado  $type_i$  es el tipo de la crítica  $c_i$  (generalmente del tipo menor, mayor, diferente representados por  $<$ ,  $>$ ,  $<>$ , respectivamente), y finalmente  $v_i$  es el valor actual de la característica  $f_i$ .

Un ejemplo de crítica en el dominio de SMARTPHONE puede ser  $c_1 = \{price, <, \$190\}$ , la cual indica que el usuario requiere un producto con un precio menor al precio del producto actualmente recomendado ( $\$190$ ).

**Definición 5. Modelo de Usuario.** Defino el modelo de usuario (U) para cada usuario, como un conjunto de pares compuesto por una crítica y un

producto recomendado  $(r_i, c_i)$ . Formalmente, el modelo de usuario se define como  $U = \{u_1, \dots, u_k\}$ , donde  $u_i = (r_i, c_i)$  es un par específico de crítica y producto recomendado. Así,  $r_i$  representa el producto recomendado y  $c_i$  representa la crítica aplicada sobre  $r_i$  (ver Definición 4). Todas las críticas generadas por el usuario representan sus preferencias durante el proceso de recomendación. De esta manera,  $u_1$  contiene la primera crítica en el modelo de usuario,  $u_2$  contiene la segunda y  $u_k$  la última crítica presente en el modelo.

Un ejemplo de un modelo de usuario en el dominio de SMARTPHONE puede ser  $U = \{\text{Nokia Lumia1210}, (\text{price}, <, \$190)\}, \{\text{Nokia 701}, (\text{Manufacturer}, <, \text{Nokia})\}, \{\text{Samsung Galaxy Trend}, (\text{Model}, <, \text{Galaxy Trend})\}$ .

### 3.3 BASE ALGORÍTMICA DE LAS PROPUESTAS

Considerando el ámbito de los sistemas de recomendación, mi foco de investigación se centra en los recomendadores basados en críticas (Burke et al., 1997; Burke, 2002b; Pu and Faltings, 2004; Reilly et al., 2004; McGinty and Reilly, 2011; Pu et al., 2011; Chen and Pu, 2012). En este sentido, existen en la literatura dos propuestas principales que representan las bases para el resto de trabajos que se han realizado sobre recomendadores basados en críticas.

La primera de ellas corresponde al algoritmo basado en críticas denominado *Standard Critiquing* (STD), que fué el primer trabajo asociado a algoritmos basados en críticas. Este algoritmo fue utilizado en el recomendador *Find Me* (Burke et al., 1997). El algoritmo *standard critiquing* tiene 3 fases dentro del proceso de recomendación como se muestra en la Figura 3 de la Sección 2.2. En la fase de recomendación, STD filtra los restantes productos a recomendar de la CB, eliminando aquéllos que no satisfacen la última crítica realizada por el usuario. Posteriormente, de los productos filtrados que se denominan productos candidatos de CB, realiza un ranking teniendo en cuenta la similitud de cada producto con el producto recomendado actualmente al usuario. El producto de mayor ranking es seleccionado para recomendarlo al usuario.

La segunda propuesta corresponde al algoritmo denominado *Incremental Critiquing* (IC) (Reilly et al., 2005a). El algoritmo IC representa actualmente el algoritmo base (*baseline*) de comparación para las nuevas propuestas que se han presentado en el ámbito de recomendadores basados en críticas. En este sentido, la eficiencia de IC respecto a STD representa una mejora bastante significativa. Por lo anteriormente expuesto, en las nuevas técnicas que propongo he considerado como base el algoritmo *Incremental Critiquing*.

Uno de los aspectos diferenciadores de IC frente a STD, es que IC toma en cuenta el histórico de recomendaciones efectuadas durante la sesión de recomendación. Es decir, el algoritmo IC almacena todas las críticas que ha realizado el usuario en un modelo de usuario,  $U$ . El modelo de usuario le permite a IC evaluar la siguiente recomendación considerando todas las críticas del usuario y no solo la última como en el caso del algoritmo STD. El algoritmo IC realiza el proceso de recomendación en cuatro fases, tal y como lo muestra la Figura 12:

1. **Recomendar.** En esta fase un nuevo producto es recomendado al usuario desde la base de productos disponibles. Por un lado, el producto puede ser el producto inicial que se muestra al usuario, en cuyo caso

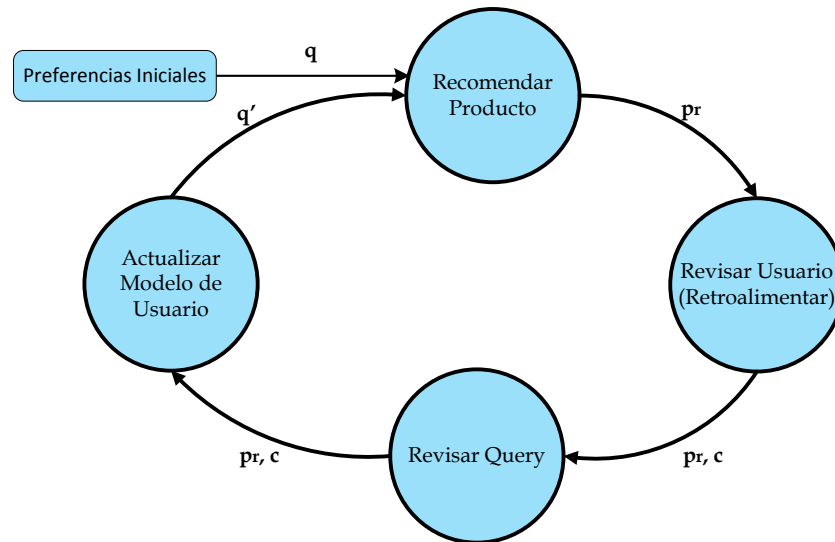


Figura 12: Proceso de recomendación del algoritmo *Incremental Critiquing*

el producto es seleccionado teniendo en cuenta las preferencias iniciales del usuario ( $q$ ). Por otro lado, cuando el producto no es inicial, se selecciona el siguiente producto a recomendar teniendo en cuenta las preferencias actuales del usuario ( $q'$ ).

2. **Revisar propuesta por parte del usuario o Retroalimentar.** En esta fase el usuario revisa el producto recomendado ( $p_r$ ) y puede aplicar un crítica ( $c$ ) sobre una característica determinada. Si el usuario acepta el producto recomendado, en esta fase puede terminar la sesión de recomendación.
3. **Revisar.** En esta fase la crítica ( $c$ ) es revisada por el sistema de recomendación y con esta crítica se define la nueva *query*  $q'$  que será utilizada en el siguiente ciclo de recomendación como las preferencias del usuario.
4. **Actualizar el Modelo de Usuario.** En esta fase el modelo de usuario es actualizado para incorporar la nueva crítica. Esta actualización implica eliminar aquellas críticas que estén presentes en el modelo de usuario actual y que contradigan la última crítica ( $c$ ) realizada por el usuario. Adicionalmente, una crítica en el modelo de usuario que haga referencia a la misma característica del producto expresada en la última crítica ( $c$ ), puede ser refinada de acuerdo al nuevo valor definido en la última crítica  $c$ .

El proceso de recomendación puede finalizar porque el usuario encuentra el producto que necesitaba o porque él explícitamente abandone el proceso.

Es necesario destacar que para recomendar el siguiente producto al usuario el algoritmo IC utiliza una calificación o medida denominada **Calidad** de recomendación (*Quality*,  $Q$ ) para establecer una prioridad entre los productos candidatos para la siguiente recomendación (véase la Fórmula 1).

$$Q(p_i, p_q, U) = \beta \cdot C_t^{P_i}(U) + (1 - \beta) \cdot S(p_i, p_q) \quad (1)$$

La medida de *Quality* se basa en dos términos claves: **Compatibilidad** ( $C_{p_i}^{p_i}(U)$ ) y **Similitud** ( $S(p_i, p_q)$ ). La constante  $\beta$  se utiliza para dar mayor prioridad a uno u otro término. En el caso de IC, de acuerdo a estudios realizados, se utiliza generalmente  $\beta = 0.75$  (Reilly et al., 2005a).

El primer término en la Fórmula 1, está asociado a la **Compatibilidad** (definida en la Fórmula 2) que existe entre el  $i$ -ésimo producto candidato ( $p_i$ ) para recomendar y el modelo de usuario ( $U$ ), es decir, esta medida indica la cantidad de críticas dentro del modelo de usuario que satisface el producto candidato en evaluación. Además, en esta fórmula 2 el término  $\delta$  es la función de satisfacción, y  $|U|$  es el número de críticas en el modelo de usuario,  $U$ . La función de satisfacción  $\delta$  devuelve 1 si el producto  $p_i$  satisface la crítica  $u_j$  o devuelve 0 si no la satisface (véase la Fórmula 3).

$$C_{p_i}(U) = \frac{\sum_{j=1}^{|U|} \delta(p_i, u_j)}{|U|} \quad (2)$$

$$\delta(p_i, u_j) = \begin{cases} 1 & \text{if } p_i \text{ satisfies } u_j \\ 0 & \text{if } p_i \text{ not satisfies } u_j \end{cases} \quad (3)$$

El segundo término en la Fórmula 1, está asociado a la **Similitud** (definida en la Fórmula 4) entre el producto actualmente recomendado ( $p_q$ ) y el  $i$ -ésimo producto candidato a recomendar ( $p_i$ ). Además, en esta fórmula 4 el término  $d(p_i^{f_s}, p_q^{f_s})$  corresponde a la distancia entre el producto candidato,  $p_i$ , y el actual producto recomendado al usuario  $p_q$  (*query*).

$$S(p_i, p_q) = \sum_{s=1}^{|F|} d(p_i^{f_s}, p_q^{f_s}) \quad (4)$$

La medida de distancia,  $d(p_i^{f_s}, p_q^{f_s})$ , depende de si la característica  $f_s$  es numérica o nominal. Esta definición es descrita en las Fórmulas 5, 6, y 7.

$$d(p_i^{f_s}, p_q^{f_s}) = \begin{cases} \text{num}(p_i^{f_s}, p_q^{f_s}) & \text{if } f_{t_s} \text{ is numeric} \\ \text{nom}(p_i^{f_s}, p_q^{f_s}) & \text{if } f_{t_s} \text{ is nominal} \end{cases} \quad (5)$$

$$\text{num}(p_i^{f_s}, p_q^{f_s}) = 1 - \frac{|p_i^{f_s} - p_q^{f_s}|}{\max(p_i^{f_s}, p_q^{f_s})} \quad (6)$$

$$\text{nom}(p_i^{f_s}, p_q^{f_s}) = \begin{cases} 1 & \text{if } f_s \text{ is nom. and } p_i^{f_s} = p_q^{f_s} \\ 0 & \text{if } f_s \text{ is nom. and } p_i^{f_s} \neq p_q^{f_s} \end{cases} \quad (7)$$

En la siguiente sección describo en detalle cada una de las técnicas de recomendación propuestas.

### 3.4 TÉCNICAS DE RECOMENDACIÓN BASADAS EN CRÍTICAS

En base a los antecedentes presentados en los apartados anteriores de esta tesis, las nuevas técnicas propuestas las he clasificado de acuerdo a la naturaleza colaborativa que sustenta a cada una de ellas. En concreto, aquellas técnicas que hacen uso de información colaborativa histórica que no es recogida en el momento mismo en que se realiza la interacción del usuario con el recomendador las denomino técnicas de recomendación basadas en colaboración *off-line* (véase el cuadro izquierdo en la Figura 13). Por el contrario, las técnicas que hacen uso de información colaborativa inmediata proveniente del recomendador como de otros usuarios que también están interactuando en ese momento con el sistema de recomendación, las denomino técnicas de recomendación basadas en colaboración *on-line* (véase el cuadro derecho en la Figura 13).

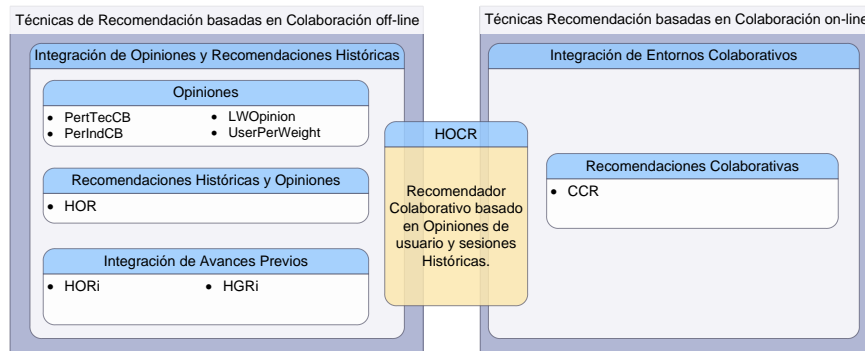


Figura 13: Resumen gráfico de técnicas de recomendación propuestas.

En primer lugar, las técnicas de recomendación basadas en colaboración *off-line* integran opiniones de los productos y recomendaciones históricas de otros usuarios que han utilizado el recomendador previamente. En concreto, tal y como se muestra en el lado izquierdo de la Figura 13, propongo integrar opiniones y recomendaciones históricas y además integrar avances de estudios previos que demuestran mejoras en los procesos de recomendación.

En segundo lugar, las técnicas de colaboración *on-line* están basadas en información colaborativa instantánea que se puede obtener de parte del recomendador como también de otros usuarios que están utilizando el sistema de recomendación en el mismo momento. De esta forma propongo integrar en el sistema de recomendación, la información colaborativa *on-line* que pueden facilitar algunos entornos que poseen una naturaleza colaborativa. Esta técnica de recomendación se muestra en el lado derecho de la Figura 13.

Adicionalmente, en el Cuadro 1 presento un breve resumen de cada una de las técnicas propuestas en esta tesis. En las siguientes secciones presento en detalle cada una de las nuevas técnicas propuestas, comenzando por las basadas en colaboración *off-line* y posteriormente las técnicas basadas en colaboración *on-line*.

Cuadro 1: Resumen de técnicas de recomendación basadas en colaboración

Nombre Técnica	Descripción
<ul style="list-style-type: none"> <li>• PerTecCB (Sección 3.5.4)</li> <li>• PerIndCB (Sección 3.5.5)</li> </ul>	Propongo integrar opiniones de usuario dentro de la base de productos con el fin de enriquecer su descripción con información colaborativa (opiniones) proveniente de otros usuarios.
<ul style="list-style-type: none"> <li>• LwOpinion (Sección 3.5.7)</li> <li>• UserPerWeight (Sección 3.5.8)</li> </ul>	Propongo integrar opiniones de usuario dentro del proceso de recomendación a través de técnicas de ponderación de características con el fin de priorizar aquellas características del producto que son más importantes para los usuarios.
<ul style="list-style-type: none"> <li>• HOR (Sección 3.6)</li> </ul>	Propongo integrar de forma conjunta información colaborativa proveniente de sesiones de recomendación históricas que otros usuarios realizaron anteriormente y las opiniones de otros usuarios acerca de los productos disponibles, con el fin de otorgar un soporte al usuario en la búsqueda de un producto determinado.
<ul style="list-style-type: none"> <li>• HORi (Sección 3.7)</li> <li>• HGRi (Sección 3.7)</li> </ul>	Propongo integrar avances de estudios previos en las propuestas de recomendación basadas en sesiones históricas, con el fin de aprovechar investigaciones previas que han demostrado mejoras en recomendaciones.
<ul style="list-style-type: none"> <li>• CCR (Sección 3.8)</li> </ul>	Propongo aprovechar la naturaleza colaborativa de las personas para facilitar la búsqueda de productos. En concreto, propongo un nuevo algoritmo de recomendación que integra la información que se puede generar en entornos colaborativos tales como entornos virtuales en tres dimensiones. Esta técnica tiene como objetivo utilizar la información colaborativa para mejorar la eficiencia en las recomendaciones.
<ul style="list-style-type: none"> <li>• HOCR (Sección 3.9)</li> </ul>	Propongo unificar los enfoques nombrados anteriormente, en una sola técnica que integre simultáneamente información colaborativa <i>off-line</i> y <i>on-line</i> y que además integre técnicas de investigaciones previas que permiten mejorar el proceso de recomendación.

### 3.5 INCORPORACIÓN DE OPINIONES DE USUARIOS EN RECOMENDACIONES DE PRODUCTOS

En esta sección explico mi propuesta para incorporar información colaborativa *off-line* proveniente de opiniones de usuarios en el proceso de recomendación. En primer lugar entrego los antecedentes previos del estudio realizado en este ámbito. En segundo lugar, explico la metodología utilizada para la obtención de las opiniones de usuarios. En tercer lugar, presento

dos propuestas para integrar las opiniones de usuarios dentro de la descripción de los productos. Finalmente, presento dos propuestas para integrar las opiniones de usuarios dentro del proceso de recomendación.

### 3.5.1 Antecedentes

En los últimos años ha crecido el interés por utilizar las valoraciones que los usuarios realizan de los productos (contenidos generados por el usuario) en los procesos de recomendación. La mayoría de las veces los contenidos generados por los usuarios, corresponden a opiniones que los usuarios emiten acerca de un producto, ya sea en forma de textos en lenguaje natural o por valoraciones de características de los productos conocidos como *ratings*. Aunque, existe mucho interés por estudios asociados a las valoraciones de los productos por parte de los usuarios, existen muy pocos estudios en el ámbito de los sistemas de recomendación basados en críticas. En este sentido, algunos trabajos se han enfocado en la utilización de los contenidos generados por usuarios como una forma de descripción de los productos durante la recomendación (Dong et al., 2013c; Dong et al., 2013a). Concretamente, en este estudio los autores proponen la utilización de las opiniones de usuarios como una alternativa de descripción de los productos respecto a las habituales características físicas del producto (Dong et al., 2013c). En su trabajo, ellos presentan una metodología para la extracción automática de características de los productos desde opiniones emitidas por los usuarios en lenguaje natural. Una vez extraídas las nuevas características del producto, se almacenan en una base de productos que será utilizada en el proceso de recomendación. Posteriormente, estos mismos autores presentaron una propuesta para analizar las opiniones de las personas en lenguaje natural, para descubrir y rescatar sentimientos y preferencias que se encuentren implícitas dentro de los comentarios emitidos por los usuarios (Dong et al., 2013a). Este último trabajo ha sido ampliado por otros investigadores para adquirir nuevos conocimientos sociales desde el análisis de los contenidos que pueda generar el usuario en sitios de comercio electrónico (Chen et al., 2014; Chen et al., 2015).

En un enfoque distinto, las opiniones de usuarios también se han utilizado para dar soporte a las personas cuando éstas necesitan expresar nuevas opiniones para productos (Dong et al., 2013b) o para mejorar las explicaciones que se entregan a los usuarios cuando se realiza una recomendación (Muhammad et al., 2015). La generación de explicaciones acerca de la recomendación de un producto busca entregar al usuario información más descriptiva respecto al tradicional listado de valores asociados a las características del producto recomendado. Es necesario mencionar que la generación de explicaciones representa por si sola todo un ámbito de estudio dentro de los sistemas de recomendación.

Todos los estudios previos han demostrado que la incorporación de contenidos generados por los usuarios en sistemas de recomendación conversacionales representan un enfoque que abre muchas posibilidades de mejoras en este tipo de sistemas. Sin embargo, ninguno de ellos lo han realizado sobre sistemas de recomendación basados en críticas y por otro lado, la mayoría busca representar los productos en base a descripciones que se puedan obtener del análisis de la nueva fuente de información que representan los contenidos generados por usuarios, específicamente las opiniones de las personas.



Mi propuesta consiste en incorporar los contenidos generados por usuarios en los recomendadores basados en críticas. Específicamente, propongo utilizar opiniones de los usuarios en forma de *ratings* que representan las valoraciones de los productos realizadas por los usuarios. Para lo cual no solo propongo la integración de los contenidos generados por los usuarios en las bases de productos, sino que también propongo integrar este tipo de información colaborativa *off-line* dentro del proceso de recomendación. Esta propuesta la divido en tres fases. Primero, establezco la metodología para la obtención de los contenidos generados por los usuarios desde sitios de comercio electrónico. Segundo, propongo enriquecer la base de descripción de los productos con la incorporación de los contenidos generados por los usuarios. La tercera fase, consiste en incorporar los contenidos generados por los usuarios dentro del proceso de recomendación.

### 3.5.2 Obtención de contenidos generados por los usuarios

Actualmente, uno de los ejemplos más comunes de contenidos generados por el usuario en sitios de comercio electrónico lo representan las opiniones de usuarios en forma de valoración de los productos (comúnmente conocido por su nombre en inglés, *ratings*). Los *ratings* de productos han sido ampliamente utilizados para entregar recomendaciones a los usuarios, basados principalmente en algoritmos de filtrado colaborativo (CF). Estas valoraciones de los productos, representan un tipo de información colaborativa *off-line* que puede ser utilizada por otros usuarios mientras realizan búsquedas de productos o servicios. Los *ratings* pueden aplicarse de tal forma que el usuario deja una valoración del producto como un todo, por ejemplo, marcando una cantidad de iconos (habitualmente iconos en forma de estrellas), que representan el nivel de satisfacción con el producto. Sin embargo, en la actualidad muchos sitios de comercio electrónico usan un nuevo tipo de *rating* que se aplica a un número reducido de características del producto, conocidas como características *perceptivas* o *cualitativas*. En este caso, la valoración va dirigida a las características perceptivas específicas tales como diseño, rendimiento, calidad de servicio, etc. Habitualmente, una característica perceptiva se asocia a un grupo de las características técnicas del producto, las cuales corresponden a las características originales que define el producto o servicio, por ejemplo, marca, modelo, precio, tamaño, etc. A modo de ejemplo, la característica perceptiva de rendimiento se asocia a RAM, CPU, etc., en una base de datos de PC's o smartphones.

En concreto, las características perceptivas facilitan la evaluación de un producto por parte de los usuarios, ya que éstos pueden captar a grandes rasgos si un producto satisface sus necesidades. Es por esto, que muchos sitios de comercio electrónico han incrementado el uso de este tipo de características, tales como: Amazon.com<sup>1</sup>, TripAdvisor<sup>2</sup>, GSMarena<sup>3</sup>, Ciao<sup>4</sup>, etc. Para ejemplificar, en el caso de TripAdvisor los usuarios pueden dar su opinión, a través de *ratings*, respecto a diferentes características perceptivas de un restaurante, tales como: servicios ofrecidos por el restaurante, comida, precio, entre otras características.

Para obtener estas características perceptivas desde sitios de comercio electrónico es posible aplicar cualquier mecanismo de reconocimiento de estruc-

---

1 <http://www.amazon.com>

2 <http://www.tripadvisor.com>

3 <http://www.gsmarena.com>

4 <http://www.ciao.com>

turas léxicas dentro de un texto, *parser*. En mi caso, para analizar y extraer las características perceptivas desde un sitio de comercio electrónico, he implementado un algoritmo que utiliza un mecanismo de *parser* basado en DOM (*Document Object Model*). Específicamente, en los datos de SMARTPHONE empleados para el análisis de las propuestas en esta tesis, he utilizado el sitio de teléfonos móviles GSMarena.

Adicionalmente, con el fin de obtener la correspondencia entre las características perceptivas y las características técnicas originales del producto, se debe realizar una asociación entre ambos tipos de características. Esta asociación generalmente se realiza teniendo en cuenta las percepciones de las personas y las relaciones entre las características que ellos mismos puedan generar. En mi caso particular la asociación entre las características perceptivas y técnicas, se realizó directamente por usuarios que participaron en una de las pruebas de evaluación realizadas durante el desarrollo de la tesis. En la Sección 3.5.6 describo en mayor detalle la utilización de la asociación establecida entre las características de los productos.

### 3.5.3 Uso de las opiniones *off-line* de usuarios para enriquecer las descripciones de los productos

La primera técnica que propongo para integrar información colaborativa *off-line* en recomendaciones de productos, consiste en enriquecer las descripciones de los productos a través de la incorporación de las opiniones de los usuarios. En concreto, propongo integrar características perceptivas dentro de las bases de datos de productos, tal y como lo muestra la Figura 14. En este enfoque, defino las opiniones de los usuarios acerca de un producto determinado como las características perceptivas. Para esto, las opiniones de los usuarios son extraídas como un conjunto de *ratings* que definen un conjunto de características perceptivas de los productos en sitios de comercio electrónico. Una vez que las opiniones son extraídas, éstas se almacenan en una base de productos que denomino,  $CB_p$ . En  $CB_p = \{p_1, \dots, p_n\}$  cada producto  $p_i$  es definido por un conjunto de características perceptivas,  $F_p = \{f_{p_1}, \dots, f_{p_r}\}$ . A su vez, cada característica perceptiva de un producto es el promedio del *rating* obtenido de las opiniones de todos los usuarios respecto a dicho producto en particular.

Adicionalmente, dispongo del conjunto original de productos, denominado  $CB_t = \{p_1, \dots, p_n\}$ , donde  $p_i$  es el  $i$ -ésimo producto. Para cada producto,  $p_i$ , en la base de datos de productos original,  $F_t = \{f_{t_1}, \dots, f_{t_m}\}$  representa el conjunto de características técnicas originales, tales como: precio, tamaño, color, etc.

Teniendo como base los dos conjuntos de datos  $CB_p$  y  $CB_t$ , propongo dos alternativas para integrarlas en dos nuevas bases de datos de productos denominadas  $CB_{pt}$  y  $CB_{pi}$  (véase la Figura 14). Estas nuevas bases de datos de productos son definidas por medio de las técnicas *PerTecCB* y *PerIndCB*, que describo en detalle en las siguientes secciones.

### 3.5.4 PerTecCB: Técnica de integración de características perceptivas y características técnicas

La técnica propuesta consiste en complementar la base de datos de productos originales con información colaborativa *off-line* para obtener una mejor descripción del espacio de búsqueda de productos. Esta idea considera que los usuarios de forma inherente tienden a categorizar los productos bajo va-

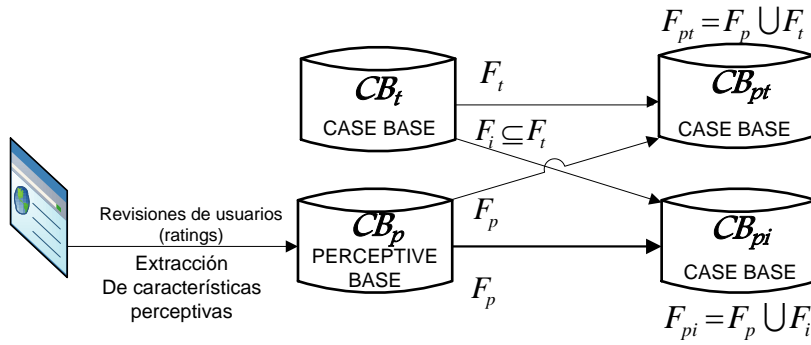


Figura 14: Uso de opiniones *off-line* de usuario en la descripción de productos

loraciones y por esto es importante agregar dentro de la descripción del producto las características perceptivas obtenidas de otros usuarios. Esta técnica la denominó **PerTecCB** y la base de datos que la sustenta la he nombrado  $CB_{pt} = \{p_1, \dots, p_n\}$ , donde cada conjunto de características que describen al producto,  $F_{pt} = F_p \cup F_t$ , está compuesto por la unión de las características perceptivas,  $F_p$ , y las características técnicas originales del producto,  $F_t$ .

Para clarificar, en el ejemplo 1 se describen todas las características de las bases de productos de telefonía móvil inteligente (SMARTPHONE) utilizadas en esta tesis.

### 3.5.5 PerIndCB: Técnica de integración de características perceptivas y características independientes

Esta técnica consiste en asumir que el conjunto de características perceptivas es suficiente para describir el espacio de búsqueda de productos. Sin desmedro de lo anterior, puede suceder que algunas de las características técnicas originales del producto no puedan ser incluidas dentro de las características perceptivas, como por ejemplo el precio del producto. Por esta razón, incluyo un conjunto de características que denominó características técnicas independientes,  $F_i$ .

Esta técnica la denominó **PerIndCB** y la base de datos que la sustenta la he nombrado  $CB_{pi} = \{p_1, \dots, p_n\}$ . En este caso la base,  $CB_{pi}$  describe cada producto,  $p_i$ , por medio de la unión de las características perceptivas  $F_p$ , y un conjunto de características técnicas que son independientes de las perceptivas,  $F_i \subseteq F_t$ . En el ejemplo 1 se pueden revisar en detalle la definición de las características técnicas independientes y el resto de las características que definen los productos de la base de datos SMARTPHONE.

- (1) El conjunto de las características técnicas originales de los productos se define como  $F_t = \{Manufacturer, Model, Weight, Size, Storage, RAM, Camera, OS, CPU, Price\}$  en la base de productos  $CB_t$ . Para este dominio en particular el conjunto de características perceptivas es definido como  $F_p = \{Design, Performance, General features\}$  en la base de productos  $CB_p$ . La característica perceptiva *Design* representa las características físicas del producto y que generalmente están a la vista de las personas. La característica *Performance* representa la eficiencia, desempeño y eficacia del producto. Finalmente, *General features* corresponde a cualquier otro servicio adicional que posea el producto. En este dominio en particular, cada característica es clasificada y ordenada en un ranking que va desde 1 a 10 puntos (1 es muy insatisfactorio y 10 es muy satisfactorio).

En el Cuadro 2 se muestra la asociación entre las características técnicas originales y las características perceptivas. Esta asociación fue realizada por usuarios reales durante una de las actividades de evaluación efectuadas en el desarrollo de la tesis. En el cuadro se puede observar que algunas características técnicas fueron establecidas como independientes, ya que los usuarios consideraron que éstas no eran representadas por ninguna de las características perceptivas. En concreto las características declaradas como independientes son  $F_i = \{Manufacturer, Model, Price\}$ .

Con este conjunto de características se construirán las dos nuevas bases de productos  $CB_{pt}$  y  $CB_{pi}$ , donde el conjunto de características para cada base de productos es descrita por:  $F_{pt} = \{Design, Performance, General features, Manufacturer, Model, Weight, Size, Storage, RAM, Camera, OS, CPU, Price\}$  y  $F_{pi} = \{Design, Performance, General features, Manufacturer, Model, Price\}$ , respectivamente.

Cuadro 2: Asociación entre las características técnicas, perceptivas e independientes

Característica Técnica	Característica Perceptiva o Independiente
$f_{t_1}$ Manufacturer	$f_{i_1}$ Independent
$f_{t_2}$ Model	$f_{i_2}$ Independent
$f_{t_3}$ Weight	$f_{p_1}$ Design
$f_{t_4}$ Size	$f_{p_1}$ Design
$f_{t_5}$ Storage	$f_{p_2}$ Performance
$f_{t_6}$ RAM	$f_{p_2}$ Performance
$f_{t_7}$ Camera	$f_{p_3}$ General feature
$f_{t_8}$ OS	$f_{p_3}$ General feature
$f_{t_9}$ CPU	$f_{p_2}$ Performance
$f_{t_{10}}$ Price	$f_{i_3}$ Independent

### 3.5.6 Uso de las opiniones *off-line* de usuarios dentro del proceso de recomendación

En la sección anterior he descrito una aproximación que permite utilizar las opiniones de usuarios dentro de las bases de datos de productos, con el objetivo de enriquecer las características técnicas de los productos. Específicamente, la propuesta considera la incorporación de características perceptivas de los usuarios acerca del producto.

En esta sección propongo incorporar la información colaborativa *off-line* proveniente de las opiniones de los usuarios dentro del proceso de recomendación. Concretamente, he tomado como base el algoritmo de recomendación *Incremental Critiquing* (Reilly et al., 2005a), teniendo en cuenta que es uno de los algoritmos basados en críticas más reconocidos y con mejores resultados asociados en términos de la eficiencia en la recomendación.

En términos generales, mi propuesta se centra en la fase de **recomendación** del proceso descrito anteriormente. Teniendo en cuenta que la recomendación de un nuevo producto en IC se centra en las medidas de compatibilidad y similitud que define la calidad de la recomendación, tal y como lo he descrito en la Fórmula 1, mis propuestas para incorporar opiniones *off-line* en el proceso de recomendación se centrarán en modificar estas medidas y por lo tanto mejorar la calidad de la recomendación.

En concreto, para la medida de compatibilidad propongo la utilización de dos técnicas de *Reinforcement Learning* (RL) denominadas *Exponential Monte Carlo* (EMC) y *Exponential Hit-Loss* (EHL). En estudios previos, el uso de técnicas de RL para mejorar la eficiencia de los recomendadores ha tenido resultados satisfactorios (Salamó et al., 2005; Salamó and Escalera, 2012). En general, las técnicas de RL aplicadas en procesos de recomendación, tienen en cuenta la crítica y el momento preciso en que se realiza. Con este tipo de técnica busco alinearme con el comportamiento de los usuarios, ya que éstos incrementan su conocimiento respecto al espacio de búsqueda a medida que avanzan en el proceso de recomendación. Es decir, en RL tendrán más importancia las críticas que se realizan posteriormente en el proceso de proceso de recomendación que las realizadas en los ciclos iniciales del proceso de recomendación.

La técnica de EMC define la compatibilidad como:

$$C_t^{p_i} = \begin{cases} C_{t-1}^{p_i} + \alpha \cdot (R_t^{p_i} + C_{t-1}^{p_i}) & \text{if } R_t^{p_i} = 1 \\ C_{t-1}^{p_i} - \alpha \cdot C_{t-1}^{p_i} & \text{if } R_t^{p_i} = 0 \end{cases} \quad (8)$$

donde  $C_{t-1}^{p_i}$  es la compatibilidad anterior de un producto candidato  $p_i$  en el instante  $t-1$ . El término  $R_t^{p_i}$  es la satisfacción del producto  $p_i$  en el instante  $t$ . Lo anterior, significa que  $R_t^{p_i} = 1$  si el producto candidato  $p_i$  satisface la actual crítica, o  $R_t^{p_i} = 0$  en cualquier otro caso. La constante  $\alpha$  es un parámetro para dar prioridad a uno de los términos de la fórmula. Como se puede observar en la Fórmula 8, la técnica de EMC tiene una tendencia exponencial en el cálculo de la compatibilidad, lo cual implica que la compatibilidad es más significativa cuando más críticas son satisfechas en los últimos ciclos de recomendación.

En el caso de EHL, la compatibilidad es definida como:

$$C_t^{p_i} = \begin{cases} C_{t-1}^{p_i} \cdot (1 + \alpha)^{(h_t+t)k}, & \text{if } R_t^{p_i} = 1 \\ C_{t-1}^{p_i} \cdot (1 - \alpha)^{(\ell_t+t)k}, & \text{if } R_t^{p_i} = 0 \end{cases} \quad (9)$$

donde  $h_t$  y  $\ell_t$  corresponden al número de veces que el producto candidato  $p_i$  ha satisfecho (*hit*) o no ha satisfecho (*loss*) las críticas en el instante  $t$ , respectivamente (cada producto tiene inicializados estos valores a cero en el tiempo  $t=0$ ), y  $k$  es un factor de regularización, el cual en mis propuestas tiene un valor de  $k = \frac{1}{2}$ . El factor de regularización  $k$  es utilizado para cambiar la influencia del factor exponencial dependiendo del objetivo de la aplicación y del tamaño del conjunto de datos, en términos de productos y características. La técnica EHL, al igual que EMC, tiene un comportamiento

exponencial el cual varía en base a la cantidad de *hits* y *losses* en la historia de cada producto y en el instante en que se produce.

Para la medida de similitud, propongo calcularla de la siguiente forma:

$$S(p_i, p_q) = \sum_{s=1}^{|F_t|} w(p_i^{f_{ts}}) \cdot d(p_i^{f_{ts}}, p_q^{f_{ts}}) \quad (10)$$

donde  $w(p_i^{f_{ts}})$  es el peso<sup>5</sup> asociado a la característica técnica del producto,  $f_{ts}$ , del producto candidato,  $p_i$ , y  $d(p_i^{f_{ts}}, p_q^{f_{ts}})$  es la distancia entre el producto candidato,  $p_i$ , y el actual producto recomendado al usuario  $p_q$  (*query*). La medida de distancia,  $d(p_i^{f_{ts}}, p_q^{f_{ts}})$ , depende de si la característica  $f_{ts}$  es numérica o nominal. Esta definición es descrita en las Fórmulas 5, 6, y 7.

Las técnicas de recomendación que propongo redefinen el término  $w(p_i^{f_{ts}})$  en el cálculo de la similitud (véase la Ecuación 10). Es decir, los valores obtenidos en los términos  $w_\gamma(p_i^{f_{ts}})$  y  $w_{per}(p_i^{f_{ts}})$  definidos en las Fórmulas 12 y 13 serán utilizados para representar el valor de  $w(p_i^{f_{ts}})$  en la Fórmula 10.

### 3.5.7 LwOpinion: Técnica de combinación de LW con contenidos generados por el usuario

Esta técnica de recomendación, que denomino **LWOpinion**, consiste en combinar el peso de las características perceptivas,  $wpn_{p_i}^{F_t}$  (véase el proceso de obtención de  $wpn_{p_i}^{F_t}$  en el Cuadro 3), con la medida de las preferencias del usuario obtenida mediante un mecanismo de ponderaciones de preferencias de usuarios conocido habitualmente por su nombre en inglés, *Local User Preference Weighting* (LW) (Salamó and Escalera, 2012). En general, en la literatura se ha demostrado que los mecanismos de ponderación mejoran la eficiencia de los sistemas de recomendación basados en críticas (Salamó et al., 2009; Salamó and Escalera, 2012).

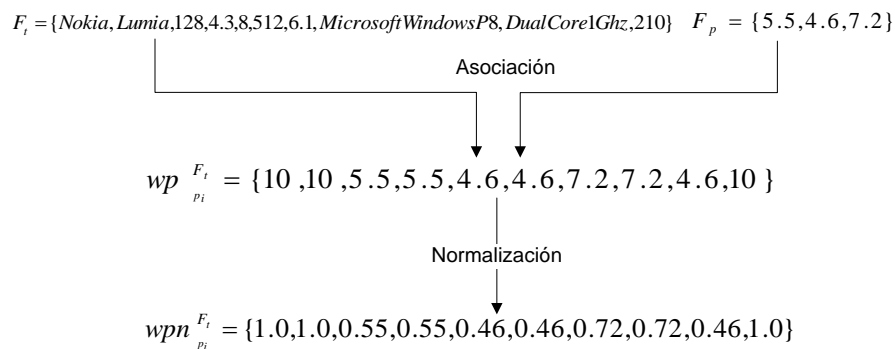


Figura 15: Generación de Pesos Perceptivos

En particular el mecanismo de LW tiene como objetivo priorizar la similitud de aquellas características del producto que no han sido criticadas durante una sesión de recomendación. Es decir, el mecanismo de LW hace uso de las preferencias manifestadas por el usuario a través de las críticas para

<sup>5</sup> Utilizo el término peso para referirme a la ponderación o importancia que tiene una característica determinada frente al resto de características del producto.

Cuadro 3: Proceso de generación de pesos perceptivos

Para la generación del peso,  $w(p_i^{f_{ts}})$  de cada característica técnica  $f_{ts}$  del producto  $p_i$ , utilizo las características perceptivas que se encuentran contenidas en la base de datos de productos,  $CB_p$ , descrita en la Sección 3.5.3. De esta forma, para cada producto se tiene un vector de características técnicas denominado  $F_t$  y un vector de características perceptivas denominado  $F_p$ , donde  $F_p$  contiene para cada característica perceptiva del producto, el promedio de todos los *ratings* de los usuarios en la característica específica. Estos dos vectores se pueden relacionar de acuerdo a la asociación establecida en el Cuadro 2, es decir, el valor de una característica técnica del producto en  $F_t$ , se puede relacionar con el valor de una característica perceptiva del producto en  $F_p$ . Considerando, la relación existente entre las características técnicas y perceptivas, defino un nuevo vector de pesos para cada producto,  $wp_{p_i}^{F_t} = \{wp_{p_i}^{f_{t_1}}, \dots, wp_{p_i}^{f_{t_m}}\}$ , donde cada valor  $wp_{p_i}^{f_{t_s}}$  corresponde al valor de la característica perceptiva  $f_{p_i}$  asociada a la característica técnica  $f_{t_s}$  del producto  $p_i$ . En el caso de las características independientes, se valoran con el valor máximo ya que considero que tienen una importancia mayor para el usuario. Una vez obtenido el vector de pesos  $wp_{p_i}^{F_t}$ , se procede a la normalización de éste utilizando un rango de valores  $[0,1]$ . Finalmente, se obtiene el vector de pesos normalizado  $wpn_{p_i}^{F_t} = \{wpn_{p_i}^{f_{t_1}}, \dots, wpn_{p_i}^{f_{t_m}}\}$  que será utilizado en las técnicas **LWOpinion** y **UserPerWeight**.

Para clarificar, en la Figura 15 muestro un ejemplo del proceso para productos del dominio de teléfonos móviles (*SMARTPHONE*). En este ejemplo, los vectores  $F_t$  y  $F_p$  corresponden a los vectores que contienen los valores de cada característica técnica y perceptiva del producto  $p_i$ , respectivamente.

En la Figura 15, se puede observar por ejemplo, que  $wp_{p_i}^{f_{t_5}}$  es el peso de la característica técnica  $f_{t_5}$  del producto  $p_i$ , la cual tiene un valor perceptivo de 4.6, porque la característica técnica *Storage*,  $f_{t_5}$ , está asociada a la característica perceptiva *Performance*,  $f_{p_2}$ . En el caso de las características técnicas independientes, se asigna el valor máximo del *rating*, por ejemplo, para la característica independiente *Manufacturer* ( $f_{t_1}$ ) se asigna en el vector de pesos el valor 10 ( $wp_{p_i}^{f_{t_1}} = 10$ ). Con el vector de pesos perceptivos calculado, se procede a normalizar, tal y como se muestran los valores de  $wpn_{p_i}^{F_t}$  en la Figura 15.

definir el peso asociado a cada característica del producto. El mecanismo LW se define de acuerdo a la Fórmula 11, donde  $|U^{f_s}|$  es el número de críticas realizadas por el usuario sobre la característica  $f_s$ , el término  $U_j^{f_s}$  corresponde a una crítica  $j$  sobre la característica  $f_s$  y el término  $\delta(p_i, U_j^{f_s})$  (véase el detalle en la Fórmula 3) es la medida de satisfacción que es calculada como el número medio de críticas satisfechas por el producto candidato,  $p_i$ .

$$LW(p_i^{f_{ts}}) = 1 - \frac{1}{2} \cdot \left( \frac{\sum_{j=1}^{|U^{f_{ts}}|} \delta(p_i, U_j^{f_{ts}})}{|U^{f_{ts}}|} \right) \quad (11)$$

La técnica *LWOpinion*, además de las preferencias del usuario, considera la incorporación de las opiniones *off-line* de otros usuarios en el cálculo de los pesos. De esta forma, propongo un enriquecimiento de las medidas previas utilizadas por LW al incluir los contenidos generados por el usuario (opiniones) en la definición de los pesos de cada característica.

Resumiendo, el objetivo de esta propuesta es evaluar el comportamiento de la recomendación teniendo en cuenta la importancia relativa tanto de las preferencias que tiene un usuario a lo largo de la sesión de recomendación, como las preferencias recogidas de las opiniones de otros usuarios.

$$w_{\gamma}(p_i^{f_{ts}}) = \gamma \cdot LW(p_i^{f_{ts}}) + (1 - \gamma) \cdot w_{pn}_{p_i}^{f_{ts}} \quad (12)$$

Concretamente, propongo definir el peso de cada característica técnica como  $w_{\gamma}$  (véase la Fórmula 12) donde  $LW(p_i^{f_{ts}})$  es el peso obtenido por LW sobre una característica técnica,  $f_{ts}$ , de un producto candidato,  $p_i$ , el término  $w_{pn}_{p_i}^{f_{ts}}$  es el peso perceptivo de la característica técnica,  $f_{ts}$ , de un producto candidato,  $p_i$  y que ha sido obtenido según el proceso descrito en el Cuadro 3.

Esta nueva medida de peso,  $w_{\gamma}(p_i^{f_{ts}})$ , sustituye la medida de peso  $w(p_i^{f_{ts}})$  de la Fórmula 10. Adicionalmente, el término  $\gamma$  es una constante que permite analizar tres posibles escenarios:

- En el primer escenario, asumo que las características perceptivas provenientes de las opiniones de usuarios tienen la mayor importancia. Para este caso por lo tanto, defino  $\gamma = 0$  para la propuesta denominada  $w_{\gamma-rat}$ .
- Contrario al primer escenario, en el segundo caso asumo que los pesos de las características que provienen de las preferencias de los usuarios, tienen mayor importancia. Por lo tanto, defino  $\gamma = 1$  para la propuesta denominada  $w_{\gamma-lw}$ .
- Finalmente, para el último escenario considero un valor medio entre las opiniones y las preferencias de los usuarios. Esta propuesta la denomino  $w_{\gamma-avg}$ . El valor medio corresponde a  $\gamma = 0.5$ .

### 3.5.8 UserPerWeight: Técnica de ponderación de características basada en contenidos perceptivos generados por el usuario

Esta técnica consiste en aprovechar el conocimiento colaborativo *off-line* que se puede obtener desde las opiniones de usuario para ajustar el peso de las preferencias de los usuarios. Además, la técnica propuesta se basa en la metodología de LW (véase la Fórmula 11), la cual prioriza la similitud para aquellas características que no han sido criticadas. Es decir, una característica técnica que no ha sido criticada asume un peso de 1.0, y su valor se decrementa cuando una crítica es satisfecha por el producto. Aun cuando, la metodología LW permite mejorar el rendimiento de un recomendador, ésta presenta una carencia, ya que la reducción en el peso e importancia de las características es igual para todas las características técnicas. Según mi opinión hay características que los usuarios establecen como más importantes que otras, por lo cual considero que si aplico una metodología similar a LW pero considerando las opiniones de otros usuarios, la técnica puede ayudar a mejorar el rendimiento del recomendador.



La técnica la denomino **UserPerWeight** y el término matemático lo he nombrado  $w_{per}$ , el cual es definido como  $w_{per}(p_i^{f_{ts}})$ . Este término sustituirá el término  $w(p_i^{f_{ts}})$  en la Fórmula 10.

$$w_{per}(p_i^{f_{ts}}) = 1 - \left( (1 - w_{pn}_{p_i^{f_{ts}}}) \cdot \left( \frac{\sum_{j=1}^{|U^{f_{ts}}|} \delta(p_i, U_j^{f_{ts}})}{|U^{f_{ts}}|} \right) \right) \quad (13)$$

La propuesta  $w_{per}$  se define en la Fórmula 13, donde  $|U^{f_{ts}}|$  es el número de críticas presentes en el modelo de usuario  $U$ , que han sido realizadas sobre la característica técnica  $f_{ts}$ . El término  $U_j^{f_{ts}}$  corresponde a una crítica  $j$  sobre la característica técnica  $f_{ts}$ , el término  $\delta(p_i, U_j^{f_{ts}})$  es la medida de satisfacción que es calculada como el número medio de críticas satisfechas por el producto candidato,  $p_i$  (véase la Fórmula 3), y el término  $w_{pn}_{p_i^{f_{ts}}}$  es el peso perceptivo, asociado a la característica técnica,  $f_{ts}$  del producto candidato,  $p_i$ . En esta técnica una característica que no ha sido criticada asume un valor de peso igual a 1.0 y se decrementa cada vez que una nueva crítica es satisfecha por el producto candidato, tomando en cuenta el peso perceptivo. Este decremento está limitado por el valor del peso perceptivo de la característica, cuyo valor mínimo es 0.1, por lo tanto el valor del peso nunca decrementa hasta un valor cero.

### 3.6 INTEGRACIÓN DE INFORMACIÓN HISTÓRICA Y OPINIONES EN RECOMENDACIONES

En esta sección explico la técnica que propongo para integrar recomendaciones históricas y opiniones de usuarios en procesos de recomendación que denomino *History and Opinion Recommender* (HOR). En primer lugar entrego los antecedentes previos del estudio realizado en este ámbito. En segundo lugar, presento los fundamentos de la técnica de personalización propuesta y finalmente, presento el algoritmo que la implementa.

#### 3.6.1 Antecedentes

En la sección anterior he presentado nuevos enfoques colaborativos que utilizan las opiniones de usuarios *off-line* como un nuevo conocimiento en recomendaciones basadas en críticas. De forma independiente, en la última década han surgido otros enfoques de recomendación basados en críticas que se han centrado en utilizar otro tipo de información colaborativa proveniente de sesiones<sup>6</sup> de críticas pasadas realizadas por otros usuarios (recomendaciones históricas).

El primer trabajo en el ámbito de recomendaciones históricas corresponde al recomendador basado en experiencias conocido por su nombre en inglés *Experience-Based Critiquing* (EBC) (McCarthy et al., 2010). El recomendador EBC mantiene una base de datos de sesiones de recomendación de otros usuarios. En cada sesión de recomendación se almacenan las críticas que realiza un usuario durante todos los ciclos de recomendación. Durante la sesión de recomendación, el recomendador busca encontrar las sesiones de

<sup>6</sup> Utilizo el término sesión para referirme a un proceso completo de recomendación desde el inicio hasta que el usuario selecciona un producto final y/o abandona el proceso.

recomendación más similares a la sesión actual y EBC selecciona como producto candidato para recomendar el producto final aceptado en una sesión histórica. Específicamente, en EBC una sesión histórica puede ser seleccionada si sus críticas se solapan con las críticas de la sesión actual en un número predefinido de críticas. Si en un ciclo de recomendación EBC no obtiene ninguna sesión histórica, recomienda un producto utilizando el algoritmo tradicional *standard critiquing*.

Posteriormente, otro enfoque que adopta la misma aproximación basada en sesiones históricas de EBC corresponde a la propuesta denominada *Nearest Neighbor Compatibility Critiquing* (Mandl and Felfernig, 2012). En esta propuesta, los autores utilizan una medida de compatibilidad que selecciona las sesiones históricas similares a la sesión actual, de la misma forma que EBC. Sin embargo, en esta propuesta la selección del producto candidato a recomendar, se realiza utilizando todos los productos recomendados en la sesión seleccionada. Es decir, a diferencia de EBC que recomienda el producto final de cada sesión, en esta propuesta se selecciona desde la sesión aquel producto que satisfaga la mayor cantidad de críticas realizadas por el usuario en la sesión seleccionada. En caso de no obtener una sesión similar a la actual, este enfoque al igual que EBC, recomienda un producto utilizando el algoritmo tradicional *standard critiquing*.

Otra propuesta que sigue la aproximación basada en sesiones históricas de EBC es la denominada *History-Aware Critiquing-based conversational recommendation* (HAC) (Salem and Hong, 2013). Sin embargo, este enfoque de recomendación agrega en las sesiones históricas, los productos recomendados en cada ciclo. De esta forma, HAC evalúa por separado tanto la cantidad de críticas actuales que se solapan con las críticas históricas, como la similitud que existe entre los productos actualmente recomendados y aquellos que están presentes en la sesión histórica. Además, en caso que no existan sesiones similares a la actual, el enfoque HAC utiliza el algoritmo tradicional IC para la nueva recomendación.

Los mismos autores de HAC, presentan posteriormente una mejora a su propuesta anterior denominada *History-Guided Recommendation*, HGR (Salem et al., 2014). Este enfoque de recomendación evalúa de manera conjunta el producto recomendado y la crítica realizada en un ciclo determinado para obtener las sesiones históricas que son más similares a la sesión actual. Es decir, en HGR una sesión de recomendación histórica puede ser seleccionada si sus pares de producto recomendado y crítica se solapan suficientemente con la sesión actual. Al igual que HAC, este enfoque utiliza el algoritmo IC para recomendar el nuevo producto en caso de no obtener sesiones similares a la actual.

### 3.6.2 HOR: Técnica de integración de recomendaciones históricas y opiniones de usuarios

En general, los enfoques descritos en la sección anterior han obtenido resultados satisfactorios. Sin embargo, considero que estos enfoques previos pueden ser mejorados integrando, además de la información de sesiones históricas, información colaborativa *off-line* proveniente de contenidos generados por los usuarios (opiniones de usuarios). De esta forma, propongo una nueva técnica de personalización denominada *History and Opinion Recommender* (HOR).

La técnica HOR, consiste en aprovechar el conocimiento colectivo que proviene tanto de las opiniones *off-line* que generan los usuarios acerca de pro-

ductos, como de las experiencias previas de recomendación de otros usuarios plasmadas en sesiones históricas de recomendación.

La Figura 16 muestra el diagrama conceptual de la técnica de recomendación HOR. El recomendador HOR mantiene una base de datos de opiniones, OB (véase la Definición 3), la cual es obtenida desde *rankings* o comentarios de productos que realizan los usuarios en sitios de comercio electrónico (véase el lado izquierdo de la Figura 16). La extracción y obtención de los contenidos generados (véase la letra a en la Figura 16) por los usuarios puede ser realizada por cualquiera de los métodos que existen actualmente, tales como los descritos en (Dong et al., 2013c; Dong et al., 2013a). Este paso nombrado por la letra (a) se hace como un pre-proceso al proceso de recomendación. Adicionalmente a las opiniones de usuarios, HOR mantiene una base de información correspondiente a sesiones históricas de recomendación, SB, la cual se muestra en la parte superior media de la Figura 16. Cada sesión histórica almacena para cada ciclo de recomendación el producto recomendado ( $r_i$ ) y la crítica aplicada a dicho producto ( $c_i$ ). Al final de cada sesión se almacena el producto aceptado por el usuario en la sesión ( $term(s_i)$ ), el cual podría ser un producto candidato para la recomendación actual si la sesión es seleccionada por el recomendador. Adicionalmente, HOR mantiene un modelo de usuario, U, que almacena el producto recomendado,  $r_i$ , y la crítica aplicada a ese producto para cada ciclo de la recomendación actual,  $c_i$ .

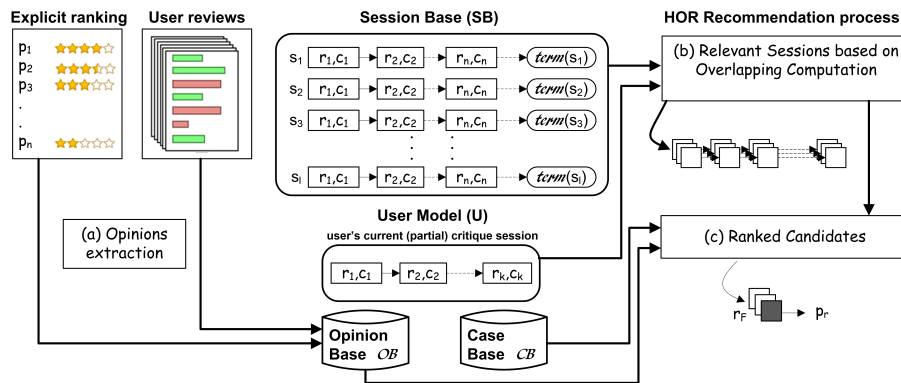


Figura 16: Diagrama conceptual de la técnica de recomendación HOR

El proceso de recomendación de HOR se muestra en el lado derecho de la Figura 16. Específicamente, el proceso en cada ciclo de recomendación, consiste en:

1. El recomendador identifica dentro de las sesiones históricas un conjunto de sesiones relevantes en base al cálculo de solapamiento que existe entre la sesión actual almacenada en U y cada sesión histórica almacenada en SB (véase la letra b en la Figura 16). Es decir, se calcula la cantidad de pares producto recomendado y crítica que están presentes en cada sesión histórica, SB y que también están presentes en la sesión actual, U. Con este cálculo se obtienen las sesiones relevantes, las cuales cumplen con tener como resultado del cálculo un valor mayor que cero. Mi propuesta para calcular el nivel de solapamiento se basa en la metodología descrita en (Salem et al., 2014) y que presento en las Fórmulas 14 y 16. En la Fórmula 14, calculo el número de pares de recomendación,  $\langle r_j, c_j \rangle$  (producto recomendado y crítica) en el modelo

de usuario  $U$ , que están también presentes en cada sesión histórica  $s_i$ . Para ello uso la medida  $\text{matchPair}$ , esta medida se calcula según la Fórmula 15, la cual devuelve 1 si en la sesión  $s_i$  se cumple que el producto recomendado  $r_i$  es igual al producto  $r_j$  que se encuentra en el modelo de usuario y además la crítica  $c_i$  aplicada sobre el producto  $r_i$  es igual a la crítica  $c_j$  aplicada al producto  $r_j$ . En caso de no cumplirse la igualdad mencionada anteriormente, la Fórmula 15 devuelve 0.

$$\text{OverlapPair}(U, s_i) = \sum_{\forall (r_i, c_i) \in U} \sum_{\forall (r_j, c_j) \in s_i} \text{matchPair}((r_i, c_i), (r_j, c_j)) \quad (14)$$

$$\text{matchPair}((r_i, c_i), (r_j, c_j)) = \begin{cases} 1 & \text{if } (r_i = r_j) \text{ and } (c_i = c_j), \\ 0 & \text{otherwise} \end{cases} \quad (15)$$

En caso que no se obtengan sesiones relevantes utilizando la Fórmula 14, es decir, que no existan pares de producto recomendado y crítica de la sesión actual que se solapen con sesiones históricas, se calcula el solapamiento por medio de la Fórmula 16. En esta fórmula, se calcula el número de críticas  $c_j$  del modelo de usuario  $U$  (sin tomar en cuenta los productos recomendados), que también están presentes en cada sesión histórica,  $s_i$ . En este caso, la medida  $\text{matchCritique}$  se calcula según la Fórmula 17. En concreto, la medida  $\text{matchCritique}$  devuelve 1 si en la sesión  $s_i$  se cumple que la crítica  $c_i$  es igual a la crítica  $c_j$  presente en el modelo de usuario  $U$ . En caso de no cumplirse la igualdad mencionada anteriormente, la Fórmula 17 devuelve 0.

$$\text{OverlapCritique}(U, s_i) = \sum_{\forall (c_i) \in U} \sum_{\forall (c_j) \in s_i} \text{matchCritique}(c_i, c_j) \quad (16)$$

$$\text{matchCritique}(c_i, c_j) = \begin{cases} 1 & \text{if } c_i = c_j, \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

2. El recomendador ordena los productos candidatos para la siguiente recomendación (véase la letra  $c$  en la Figura 16). Los productos candidatos para la siguiente recomendación en HOR corresponden a los productos aceptados por el usuario en cada sesión relevante obtenida de acuerdo a la selección descrita en los pasos anteriores,  $\text{term}(s_i)$ . Teniendo en cuenta que cada producto candidato puede estar presente en más de una sesión relevante, guardo estos candidatos en una lista denominada  $r_F$  que contiene los rankings finales de los productos candidatos para recomendar, y calculo una puntuación para cada uno de ellos. La puntuación de cada candidato se obtiene en base a la suma ponderada de dos valores: la puntuación de solapamiento que se obtuvo para cada sesión relevante en la cual el candidato está presente y el *rating* que tiene este producto candidato en la base de datos de opiniones,  $OB$  (véase la Fórmula 18). El producto candidato que obtiene la mayor puntuación,  $p_r$ , es recomendado al usuario. La información

completa del producto seleccionado para la recomendación se obtiene desde la base de datos, CB.

$$\text{RecScore}(U, s_i, o_j) = \alpha \cdot \text{OverlapScore}(U, s_i) + (1 - \alpha) \cdot o_j \quad (18)$$

La Fórmula 18 calcula un promedio entre la función *OverlapScore* (calculada usando la Fórmula 14 o la Fórmula 16) y el *rating* de las opiniones  $o_j \in \text{OB}$  del producto terminal  $p_j$  (este es el producto aceptado,  $p_j$  en  $s_i$ ,  $p_j = \text{term}(s_i)$ ). En este caso la constante  $\alpha$  permite controlar el peso relativo de la puntuación de solapamiento de los productos y la opiniones de usuarios. En la técnica HOR, de acuerdo a los experimentos que he realizado, los mejores resultados se obtienen para un valor de  $\alpha = 0.6$ .

### 3.6.3 Algoritmo HOR

En esta sección describo en detalle el algoritmo HOR que implementa la técnica de recomendación propuesta. En el Algoritmo 1 se muestra el proceso principal del HOR. El algoritmo recibe el producto inicial, denominado *query*,  $p_q$ , el cual puede estar basado en preferencias iniciales definidas explícitamente por el usuario o si se carece de dichas preferencias, este producto inicial puede ser un producto aleatoriamente seleccionado desde la base de datos de productos disponible, CB. Adicionalmente, el algoritmo recibe toda la información requerida para realizar las recomendaciones al usuario: la base de datos de productos (CB), la base de datos de sesiones históricas (SB), y la base de datos de opiniones (OB). Al inicio de la recomendación, no existen críticas generadas por el usuario, por lo tanto la última crítica realizada por el usuario,  $U_{c_q}$ , y el modelo de usuario,  $U$ , son inicializados con el valor inicial por defecto que se muestra en las líneas 3 y 4.

Una vez que el algoritmo HOR recibe la información inicial descrita en el párrafo anterior, éste inicia un ciclo que está compuesto por 4 fases (véase el Algoritmo 1, líneas 6-10): **Recommend Item**, **User Review**, **Query Revise** y **Update Model**. Este ciclo se repite hasta que el usuario acepta un producto recomendado o explícitamente abandona el proceso de recomendación. A continuación describo en detalle cada una de las fases del Algoritmo 1 en el Algoritmo 2.

```

Input:  $p_q$ : producto inicial, CB: base de productos, SB: base de sesiones, OB: base de opiniones
1 define HistoryOpinionRecommender ( $p_q, CB, SB, OB$ )
2 begin
3    $U_{c_q} \leftarrow \text{null}$  // current critique
4    $U \leftarrow \text{null}$  // user model
5   repeat
6      $p_r \leftarrow \text{RecommendItem}(p_q, CB, U, SB, OB)$  //  $p_r$ : product recommendation
7      $U_{c_q}, CB \leftarrow \text{UserReview}(p_r, CB)$ 
8      $p_q \leftarrow \text{QueryRevise}(p_r)$ 
9      $U \leftarrow \text{UpdateModel}(U, U_{c_q}, p_r)$ 
10    until  $\text{UserAccepts}(p_r)$  or  $\text{UserAbandon}()$ 
11 end

```

#### Algorithm 1: Algoritmo de HOR

La primera fase corresponde a la fase denominada **RecommendItem** (véase la línea 6 del Algoritmo 1 y las líneas 1-15 en Algoritmo 2). Esta fase es la encargada de realizar la recomendación de un nuevo producto  $p_r$  al usuario, seleccionando el producto desde la base de datos de productos CB. Esta recomendación se basa en: el producto actualmente recomendado  $p_q$ , las sesiones históricas de otros usuarios almacenadas en SB, las opiniones de otros usuarios almacenadas en OB, y la sesión actual de recomendación

que consiste en pares compuestos por el producto recomendado y la crítica aplicada a dicho producto (esta información se encuentra almacenada en el modelo de usuario  $U$ ).

```

1 define RecommendItem( $p_q, CB, U, SB, OB$ )
2 begin
3    $S^{REL} \leftarrow \text{RelevantSessions}(U, SB)$ 
4   if  $S^{REL} \neq \text{null}$  then
5      $S^{REL} \leftarrow \text{sort } s_i \in S^{REL} \text{ in decreasing } \text{RecScore}(U, s_i, o_j)$  //see Equation 18
6      $r_f \leftarrow \text{store } p_j = \text{term}(s_i)/s_i \in S^{REL}$ 
7      $p_r \leftarrow \text{getTopRanked}(r_f)$ 
8   else
9      $U_{c,q} \leftarrow \text{getLastCritique}(U)$ 
10     $CB' \leftarrow \{p \in CB \mid \delta(p, U_{c,q})\}$ 
11     $CB' \leftarrow \text{sort all } p_i \in CB' \text{ in decreasing } Q(p_i, p_q, U)$  //see Equation 1
12     $p_r \leftarrow \text{most quality product in } CB'$ 
13  end
14  return  $p_r$ 
15 end
16 define RelevantSessions( $U, SB$ )
17 begin
18    $S^{REL} \leftarrow \text{null}$ 
19    $S^{REL} \leftarrow \{s_i \in SB : \text{OverlapPair}(U, s_i) > 0\}$  //see Equation 14
20   if  $S^{REL} == \text{null}$  then
21      $S^{REL} \leftarrow \{s_i \in SB : \text{OverlapCritique}(U, s_i) > 0\}$  //see Equation 16
22   end
23   if  $S^{REL} \neq \text{null}$  then
24     for  $s_i \in S^{REL}$  do
25        $S^{REL} \leftarrow S^{REL} - \text{contradict}(\text{term}(s_i), U)$ 
26     end
27   end
28   return  $S^{REL}$ 
29 end
30 define UserReview( $p_r, CB$ )
31 begin
32    $U_{c,q} \leftarrow \text{user critique for some } f \in p_r$ 
33    $CB \leftarrow CB - p_r$ 
34   return  $U_{c,q}, CB$ 
35 end
36 define QueryRevise( $p_r$ )
37 begin
38    $p_q \leftarrow p_r$ 
39   return  $p_q$ 
40 end
41 define UpdateModel( $U, U_{c,q}, p_r$ )
42 begin
43    $U \leftarrow U - \text{contradict}(U, U_{c,q}, p_r)$ 
44    $U \leftarrow U - \text{refine}(U, U_{c,q}, p_r)$ 
45    $U \leftarrow U \cup \langle p_r, U_{c,q} \rangle$ 
46   return  $U$ 
47 end

```

Algorithm 2: Procedimientos de HOR

El procedimiento **RecommendItem** utiliza la sesión parcial de críticas realizadas por el usuario hasta el momento de la llamada al procedimiento,  $(r_1, c_1), \dots, (r_m, c_m)$ . Esta sesión se encuentra en el modelo de usuario  $U$  (véase la Definición 5). Con esta información de la sesión actual y con las sesiones históricas que se guardan en  $SB$ , el procedimiento identifica un conjunto de sesiones relevantes (*relevant sessions*),  $S^{REL}$ , como se muestra en las líneas 16-29 del procedimiento **RelevantSessions**. En concreto, una sesión relevante es una sesión histórica que se encuentra en  $SB$  y que al menos tiene algún solapamiento con la sesión actual almacenada en el modelo de usuario. Es decir, que el valor obtenido por la métrica de solapamiento utilizada (**overlapPair** o **overlapCritique**) es mayor que 0. En este sentido, HOR primero realiza una evaluación utilizando la métrica denominada **OverlapPair** (véase la Fórmula 14), la cual se basa en el cálculo del solapamiento de los pares producto recomendado y críticas que se encuentran en el modelo de usuario  $U$  y que también están presentes en alguna sesión histórica almacenada en  $SB$  (véase la línea 19 del Algoritmo 2). Si el proceso de puntuación obtenido a través de la función **OverlapPair** retorna un conjunto vacío de elementos (no hay sesiones relevantes), se procede a evaluar el solapamiento entre la sesión actual y las sesiones históricas por medio de la métrica denominada **OverlapCritique** (véase la Fórmula 16), la cual solo considera las críticas de la sesión actual que están presentes en las sesiones históricas (véase la línea 21). Cada sesión relevante termina con un producto terminal (producto acep-

tado por el usuario), el cual se formaliza como  $\text{term}(s_i)$ . Además, para cada sesión relevante se ha calculado su puntuación de solapamiento, por medio de la Fórmula 14 (si se obtuvieron sesiones relevantes desde la función **OverlapPair**, véase la línea 23) o por la Fórmula 16 (en cuyo caso la Fórmula utilizada corresponde a **OverlapCritique**, véase la línea 20). Considero como producto candidato a los productos terminales,  $\text{term}(s_i)$ , presentes en cada sesión relevante. Una vez obtenidas las sesiones relevantes, se procede a eliminar aquellos productos candidatos que son incompatibles con el modelo de usuario,  $U$  (véase las líneas 24-26). Para clarificar, un producto candidato es incompatible con el modelo de usuario, si el producto no satisface todas las críticas presentes en dicho modelo (véase el Ejemplo 2).

- (2) Considérese un dominio de teléfonos inteligentes, donde se tiene un modelo de usuario  $U$  que contiene los siguientes pares recomendación-crítica ( $\{\text{SamsungGalaxySIII}, (\text{price}, <, \$500)\}, \{\text{SamsungGalaxySII}, (\text{Manufacturer}, <>, \text{Samsung})\}, \{\text{SonyExperiaM5}, (\text{RAM}, >, 1024)\}$ ). Un ejemplo de producto candidato incompatible podría ser  $p_i = \{\text{Manufacturer} = \text{Samsung}, \text{Model} = \text{GalaxyS6}, \text{Ram} = 2048, \text{Weight} = 209, \text{Size} = 6.0, \text{Storage} = 32, \text{Resolution} = 20, \text{SO} = \text{AndroidOS}, v4.2, \text{CPU} = \text{Quad} - \text{core}2.2\text{GHz}, \text{Price} = 480\}$ . Este producto no satisface la crítica definida como  $(\text{Manufacturer}, <>, \text{Samsung})$  que se encuentra almacenada en el modelo de usuario  $U$ , por tanto se eliminaría de la lista de productos candidatos.

Una vez que se ha realizado el filtrado de las sesiones relevantes, el procedimiento **RelevantSessions** devuelve un conjunto de sesiones históricas relevantes,  $S^{\text{REL}}$ , como lo muestra la línea 3 del Algoritmo 2. Cuando el conjunto  $S^{\text{REL}}$  no está vacío, el algoritmo HOR recomienda un producto candidato desde el conjunto  $S^{\text{REL}}$  (véase las líneas 5-7). Específicamente, en la línea 5 las sesiones relevantes son ordenadas en orden decreciente respecto al valor de puntuación almacenado en  $\text{RecScore}$  (véase la Fórmula 18). Posteriormente, se extrae el producto terminal (producto candidato) desde cada sesión relevante y se agrega a la lista denominada  $r_F$ , que corresponde a una lista ordenada de productos candidatos para el siguiente ciclo de recomendación (véase la línea 6 del Algoritmo 2). Finalmente, el nuevo producto recomendado,  $p_r$ , será aquel producto que tenga la mejor puntuación en la lista  $r_F$  (véase la línea 7 del Algoritmo 2).

Cuando  $S^{\text{REL}}$  está vacío, es decir no hay sesiones relevantes y por lo tanto no hay productos candidatos para recomendar (véase la línea 8 del Algoritmo 2), propongo revertir el procedimiento al algoritmo de recomendación *incremental critiquing* (Reilly et al., 2005a), el cual solo usa el modelo de usuario  $U$  para encontrar el producto candidato a recomendar. Específicamente, primero se obtiene desde el modelo de usuario,  $U$ , la última crítica que ha realizado el usuario,  $U_{cq}$ , a través del procedimiento **getLastCritique** (véase la línea 9 del Algoritmo 2). Una vez que se obtiene la última crítica, se almacenan en  $CB'$  todos los productos que se encuentran en  $CB$  y que satisfacen la última crítica,  $U_{cq}$ , tal como se muestra en la línea 10 del Algoritmo 2. Si aún no se ha realizado ninguna crítica, todos los productos en  $CB$  son considerados ( $CB' \leftarrow CB$ ). Además, se utiliza la medida de satisfacción ( $\delta$ ), descrita en la Fórmula 3. Posteriormente, se ordenan de manera decreciente todos los productos candidatos desde  $CB'$ , teniendo como base la puntuación obtenida según la medida de calidad del producto (*quality score*),  $Q$  (véase la línea 11 del Algoritmo 2). La medida *quality* combina las medidas de compatibilidad (*compatibility*) y similitud (*similarity*) como lo he descrito

en la Fórmula 1 (véase la Sección 3.3 para mayores detalles de la medida *quality*). Finalmente, el nuevo producto recomendado  $p_r$  es el producto candidato con el valor más alto en la función de calidad, *quality* (véase la línea 12 del Algoritmo 2).

- (3) Considerando un dominio de cámaras fotográficas, en el cual se tiene un modelo de usuario que contiene los siguientes pares recomendación-crítica<sup>a</sup> ( $\{\text{Canon700D}, (\text{price}, <, \$700)\}, \{\text{Canon500D}, (\text{Manufacturer}, <>, \text{Canon})\}, \{\text{OlympusC} - 765, (\text{Model}, <>, \text{C} - 765)\}$ ) y el producto actualmente recomendado al usuario está definido por las siguientes características  $\{\text{Manufacturer} = \text{OlympusC} - 740, \text{Price} = \$750, \text{Format} = \text{SLR} - \text{Like}, \text{Resolution} = 3.14, \text{OpticalZoom} = 10, \text{DigitalZoom} = 3, \text{Weight} = 340, \text{StorageType} = \text{DPictureCard}, \text{Storage} = 16\}$ . Si el usuario realiza la crítica ( $\text{price}, >, \$750$ ) (esta crítica corresponde al término  $U_{c,q}$ ) sobre el actual producto recomendado  $\text{OlympusC} - 740$  ( $p_r$ ), el par recomendación-crítica que contiene la crítica ( $\text{price}, <, \$700$ ) se eliminará porque su crítica contradice la crítica actual,  $U_{c,q}$ . El nuevo modelo de usuario quedaría representado por  $\{\{\text{Canon500D}, (\text{Manufacturer}, <>, \text{Canon})\}, \{\text{OlympusC} - 765, (\text{Model}, <>, \text{C} - 765)\}, \{\text{OlympusC} - 740, (\text{price}, >, \$750)\}\}$

<sup>a</sup> Para simplificar el ejemplo, solamente mostramos una característica del producto recomendado  $p_i$ , sin embargo cada par contiene todas las características del producto.

En la segunda fase, **UserReview** (véase la línea 7 del Algoritmo 1 y las líneas 30-35 del Algoritmo 2), el usuario revisa el producto recomendado y realiza alguna crítica al producto. En esta fase, HOR realiza dos acciones: (1) HOR recibe el *feedback* (retroalimentación) desde el usuario en forma de una crítica como se muestra en la línea 32; y (2) HOR elimina el actual producto recomendado,  $p_r$ , desde la base de datos de productos, CB, tal como lo muestra la línea 33. Finalmente, esta fase retorna la base de datos de productos actualizada y la crítica realizada por el usuario al producto actual recomendado.

La tercera fase, **QueryRevise** (véase la línea 8 del Algoritmo 1 y las líneas 36-40 del Algoritmo 2), se focaliza en la revisión de la *query* del producto  $p_q$  para el siguiente ciclo. Básicamente, el producto actual recomendado,  $p_r$ , pasa a ser la nueva *query*,  $p_q$ .

Finalmente, la fase denominada **UpdateModel** (véase la línea 9 del Algoritmo 1 y las líneas 41-47 del Algoritmo 2) es la encargada de guardar tanto el producto recomendado,  $p_r$ , como la última crítica realizada por el usuario,  $U_{c,q}$ , en el modelo de usuario,  $U$ . Es necesario destacar que mantener el modelo de usuario no es tan simple como solo guardar una lista que contenga los productos recomendados y las críticas realizadas sobre esos productos. En este sentido, algunas críticas pueden generar inconsistencias con críticas realizadas previamente, ya que los usuarios pueden refinar sus requerimientos o necesidades durante el proceso de recomendación. Esta situación se debe a que los usuarios están aprendiendo acerca de los productos y como se distribuye el espacio de búsqueda durante la recomendación. Es esencial por tanto, actualizar el modelo de usuario agregando el último par recomendación-crítica ( $\{p_r, U_{c,q}\}$ ), pero solamente después de validar las críticas anteriormente realizadas para eliminar las inconsistencias que se pudieran generar. Este proceso implica eliminar todos los pares recomendación-crítica cuya crítica **contradice** la última crítica,  $U_{c,q}$  (véase la línea 43 del Algoritmo 2) y aquellos para los cuales la nueva crítica es un



**refinamiento** (véase la línea 44 del Algoritmo 2). El procedimiento *contradict* en la línea 43 elimina cualquier par recomendación-crítica donde la crítica contradice la crítica actual  $U_{c,q}$  (véase Ejemplo 3).

Adicionalmente, el procedimiento *refine* en la línea 44 reformula aquellos pares recomendación-crítica para los cuáles la nueva crítica es un refinamiento (véase Ejemplo 4).

- (4) Considerando el mismo dominio del ejemplo anterior, en el cual el modelo de usuario estaba definido por  $\{(\text{Canon700D}, (\text{price}, <, \$700)), (\text{Canon500D}, (\text{Manufacturer}, <>, \text{Canon})), (\text{OlympusC} - 765, (\text{Model}, <>, \text{C} - 765))\}$ , y para este ejemplo establezco el producto actual recomendado como  $\{\text{Manufacturer} = \text{OlympusD} - 560, \text{Price} = \$311, \text{Format} = \text{SLR} - \text{Like}, \text{Resolution} = 3.14, \text{OpticalZoom} = 3, \text{DigitalZoom} = 3, \text{Weight} = 220, \text{StorageType} = \text{DPictureCard}, \text{Storage} = 16\}$ . Un ejemplo de refinamiento sucedería si el usuario realiza la crítica  $(\text{price}, <, \$311)$ . En este caso, el par recomendación-crítica  $\{\text{Canon700D}, (\text{price}, <, \$700)\}$  será refinado a  $\{\text{OlympusD} - 560, (\text{price}, <, \$311)\}$  quedando el nuevo modelo de usuario definido por  $\{(\text{OlympusD} - 560, (\text{price}, <, \$311)), (\text{Canon500D}, (\text{Manufacturer}, <>, \text{Canon})), (\text{OlympusC} - 765, (\text{Model}, <>, \text{C} - 765))\}$ .

### 3.7 INTEGRACIÓN DE AVANCES PREVIOS EN RECOMENDACIONES BASADAS SESIONES HISTÓRICAS

Los sistemas de recomendación basados en críticas han experimentado una serie de avances gracias a todos los estudios previos realizados. Sin embargo, estos avances y mejoras actualmente se encuentran disgregados. En este sentido, no existe un análisis del impacto que pueda tener la integración de varias técnicas que han mejorado el algoritmo base IC en un solo algoritmo. Por lo anteriormente expuesto, considero que la técnica HOR, así como otros recomendadores basados en sesiones históricas (como es el caso de HGR) pueden mejorar su proceso de recomendación si se integran con avances previos en el ámbito de los recomendadores basados en críticas.

Actualmente, en la literatura se ha demostrado que hasta el momento el algoritmo de recomendación basado en sesiones históricas HGR, ha sido el mejor algoritmo de recomendación basado en críticas, lo cual también se demuestra en el análisis de algoritmos que realizo en el Capítulo 5. Por lo anterior, propongo integrar mi algoritmo propuesto HOR y el algoritmo HGR con enfoques previos de mejoras para algoritmos de recomendación basados en críticas. Estas nuevas técnicas las denomino HORi y HGRi.

Específicamente, me focalizo en la fase de recomendación denominada **RecommendItem** de ambos algoritmos HOR y HGR. En esta fase, cuando no se logran encontrar sesiones relevantes dentro de la base de sesiones históricas (véase la línea 9 en el Algoritmo 2), los recomendadores basados en sesiones históricas tradicionalmente revierten al algoritmo *standard critiquing* (Burke et al., 1997) o al algoritmo *incremental critiquing* (Reilly et al., 2005a). Por el contrario, yo propongo revertir a un algoritmo *incremental critiquing* mejorado, el cual específicamente modifica el uso de la medida denominada *quality*

(Q) de IC, que he descrito en la Fórmula 1. Mi propuesta, consiste en integrar enfoques que tengan en cuenta el aprendizaje que realiza el usuario durante una sesión de recomendación, para lo cual utilizo técnicas de *Reinforcement Learning* que permitan tener en cuenta el momento en que el usuario realiza una crítica. Adicionalmente, propongo integrar enfoques de ponderación que mejoren el efecto que produce una crítica sobre el producto recomendado para reducir la cantidad de ciclos de recomendación.

$$Q_{\text{HBRi}}(p_i, p_q, U) = \beta \cdot C_{\text{RL}_t}^{P_i}(U) + (1 - \beta) \cdot S_{\text{LW}}(p_i, p_q) \quad (19)$$

En concreto, la Fórmula 19 representa la medida de calidad de recomendación (*quality*) del algoritmo IC, con modificaciones en el cálculo de los términos  $C_{\text{RL}_t}^{P_i}(U)$  y  $S_{\text{LW}}(p_i, p_q)$ . Esta fórmula se aplica tanto para el algoritmo HORi como para el HGRi.

- Para calcular la compatibilidad,  $C_{\text{RL}_t}^{P_i}$ , utilizo el método de *Reinforcement Learning* denominado *Exponential Hit-Loss* definido en la Fórmula 9 de la Sección 3.5.6. En el método EHL (Salamó et al., 2005; Salamó et al., 2012), tendrán más importancia las críticas que se realizan en los últimos ciclos de recomendación. Este método es concordante con el comportamiento que generalmente tienen las personas durante la búsqueda de un producto, es decir, habitualmente al inicio de la recomendación el usuario realiza críticas sin tener mucha claridad o conocimiento respecto al producto que requiere. Por lo anterior, las críticas iniciales debieran ser menos importantes que las críticas que va realizando posteriormente, las cuales adquieren mayor importancia ya que el usuario tiene un mayor conocimiento del producto final deseado.
- Para calcular la similitud,  $S_{\text{LW}}(p_i, p_q)$ , utilizo la Fórmula 10 que he presentado en la Sección 3.5.6. En esta fórmula propongo utilizar el método conocido por su nombre en inglés *Local User Preference Weighting* (LW) (Salamó et al., 2005), que he descrito en detalle en la Sección 3.5.7. Este método prioriza la similitud de aquellas características que no han sido criticadas por el usuario, con el objetivo de otorgar nuevas recomendaciones que eviten que el usuario se mantenga criticando productos muy similares, lo cual podría llevar a que el usuario no logre visualizar alternativas que pueden resultar beneficiosas para su búsqueda. En concreto, este método consiste en definir el término  $w(p_i^{f_{ts}})$  que está presente en el cálculo de la similitud,  $S(p_i, p_q)$ . Específicamente, la Fórmula 11 en la Sección 3.5.7 describe en detalle el cálculo de la técnica LW.

Estas dos propuestas (HORi y HGRi) son las últimas propuestas de mi tesis enfocadas en usar información colaborativa *off-line* entre usuarios.

### 3.8 RECOMENDACIONES CONVERSACIONALES COLABORATIVAS

En esta sección presento una nueva técnica de recomendación basada en información colaborativa *on-line* entre usuarios que denomino *Collaborative Conversational Recommender* (CCR). En primer lugar, presento los antecedentes previos que sustentan la técnica propuesta. En segundo lugar, presento

brevemente los estados por los cuales transita el usuario en el proceso de recomendación. Finalmente, presento el algoritmo que implementa la técnica de recomendación propuesta.

### 3.8.1 Antecedentes

La mayoría de los recomendadores conversacionales que utilizan la técnica de *critiquing* como un mecanismo de retroalimentación por parte del usuario, se basan en mecanismos de recomendación individual. Es decir, en estos recomendadores el usuario solo puede interactuar con el sistema a través de una interfaz de usuario personal, sin la posibilidad de interactuar y colaborar con otros usuarios. Es el caso de los recomendadores *QuickShop* (Reilly et al., 2005a) y *Example Critiquing* (Pu et al., 2008), los cuales se han implementado en plataformas web de dos dimensiones o también el caso de *MobyRek* (Ricci and Nguyen, 2007) que utiliza una plataforma para teléfonos móviles.

Adicionalmente, muy pocos algoritmos que implementan sistemas de recomendación conversacionales se han enfocado en integrar información colaborativa (*off-line, on-line*), proveniente de otros usuarios, en sus procesos de recomendación. Solo algunos enfoques previos utilizan información *off-line* que proviene de opiniones de otros usuarios para realizar recomendaciones (Dong et al., 2013c; Dong et al., 2013a; Chen et al., 2014), u otros estudios que utilizan recomendaciones históricas *off-line* de otros usuarios como base para sus recomendaciones (McCarthy et al., 2010; Mandl and Felfernig, 2012; Salem and Hong, 2013).

Sin desmedro de lo anterior, ninguno de los enfoques previos utiliza información colaborativa *on-line*. Este tipo de información, se obtiene de la interacción in situ que se produce entre los usuarios en entornos colaborativos que en la actualidad son diversos, tales como redes sociales, entornos virtuales colaborativos u otros. Aunque, actualmente en la literatura existen algunas aproximaciones de recomendadores que utilizan entornos virtuales colaborativos (Arroyo et al., 2010; Shah et al., 2010; Guo and Elgendi, 2013; Hu and Wang, 2010; Xu and Yu, 2010), ninguno de ellos implementa algoritmos de recomendación basados en críticas, ya que en su mayoría corresponden a recomendadores tradicionales basados en filtrado colaborativo (CF) o filtrado basado en contenido (CB).

Mi propuesta, a diferencia del resto de estudios previos, considera la integración de información colaborativa *on-line* proveniente de las interacciones entre los usuarios en un entorno virtual colaborativo. En concreto, propongo un nuevo algoritmo de recomendación que denomino CCR (Collaborative Conversational Recommender), el cual aparte de utilizar las preferencias del usuario en sus recomendaciones, otorga al usuario la posibilidad de colaborar con otros usuarios en el momento de la recomendación. A diferencia de otros enfoques como los recomendadores para grupos (McCarthy et al., 2006c), los cuales permiten colaboración entre usuarios con un objetivo común, en esta propuesta el usuario siempre mantiene su producto objetivo individual y puede alcanzarlo con ayuda de otros usuarios. Resumiendo, el algoritmo CCR permite realizar recomendaciones individuales para un usuario y recomendaciones colaborativas para un grupo de usuarios que se encuentren interactuando entre ellos. A su vez, otorga al usuario la posibilidad de iniciar y finalizar colaboraciones con otros usuarios en el momento que lo desee.

## 3.8.2 Estados del Proceso de Recomendación

La técnica de recomendación CCR permite a los usuarios interactuar activamente con el recomendador y con otros usuarios. Esta interacción implica que el usuario transita por diferentes estados durante una sesión de recomendación, como se muestra en la Figura 17. Con el objetivo de facilitar el entendimiento del proceso de recomendación colaborativo, describo a continuación de manera breve los estados por los cuales transita un usuario durante una sesión de recomendación. Mayores detalles de cada estado y de las acciones que realiza el usuario durante la interacción en el recomendador, los entrego en el Capítulo 4, donde se explica en profundidad la interfaz de usuario inteligente que implementa la técnica de recomendación CCR.

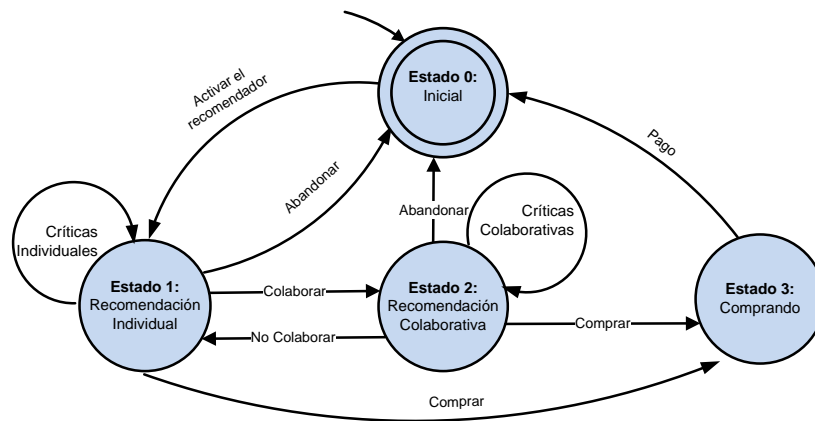


Figura 17: Estados de los usuarios en un proceso de recomendación conversacional colaborativo

**Estado 0: Inicial.** En este estado el usuario se encuentra conectado en un entorno virtual que permita la interacción y la colaboración con otros usuarios. Cuando el usuario inicia una sesión de recomendación, el recomendador CCR crea un modelo de usuario ( $U$ ) para el usuario y el usuario transita al **Estado 1**.

**Estado 1: Recomendación Individual.** En este estado, el recomendador retorna un producto para ser mostrado al usuario y el usuario puede realizar cuatro acciones.

1. *Críticas Individuales.* Corresponden a las críticas unitarias realizadas por el usuario, para las cuales el recomendador proporciona un nuevo producto recomendado y el modelo de usuario es actualizado para incluir la nueva crítica. Después de esta acción el usuario se mantiene en el mismo estado (**Estado 1**).
2. *Colaborar.* Esta acción ocurre cuando un usuario desea colaborar con otro usuario, es decir, cuando a un usuario le gusta el producto recomendado a otro usuario. El usuario que inicia la colaboración lo denominamos usuario colaborador (*guest user*,  $u_g$ ) y el otro usuario lo denominamos usuario líder (*host user*,  $u_h$ ). Cuando se realiza esta acción, tanto  $u_h$  como  $u_g$  transitan al **Estado 2**.

3. *Comprar*. Esta acción ocurre cuando el usuario ha encontrado un producto que le interesa comprar. Al realizar esta acción, el usuario transita al **Estado 3**.
4. *Abandonar*. Esta acción sucede cuando un usuario explícitamente finaliza el proceso de recomendación porque no encontró un producto adecuado. Por tanto, el usuario retorna al **Estado 0** donde puede comenzar una nueva recomendación.

**Estado 2:** Recomendación Colaborativa. Un usuario transita a este estado en dos situaciones: cuando el usuario inicia la colaboración (en este caso el usuario asume el rol de usuario colaborador,  $u_g$ ) o cuando otro usuario quiere colaborar con él (en este caso el usuario asume el rol de líder,  $u_h$ ). En un estado colaborativo el usuario líder es el único que puede interactuar con el recomendador. Lo anterior, se sustenta en el hecho que si los usuarios colaboradores realizaran críticas sin el consentimiento del usuario líder, estas críticas podrían corromper el modelo de usuario del líder. Asumo por tanto, que si un usuario está interesado en el producto que actualmente está siendo recomendado a otro usuario y se asocia a dicho usuario en un proceso de recomendación colaborativo deberá asumir las críticas que realiza el usuario líder.

Si un usuario colaborador,  $u_g$ , no está de acuerdo con la siguiente crítica que se realizará, él puede dejar el estado colaborativo cuando lo estime conveniente.

En el **Estado 2** se pueden realizar las siguientes cuatro acciones:

1. *Críticas Colaborativas*. El usuario líder puede realizar una crítica acordada según consenso con el resto de usuarios colaboradores, que denominamos críticas colaborativas. Como resultado de esta crítica realizada el usuario líder y todos los usuarios que están colaborando con él, actualizan sus modelos de usuarios y reciben un nuevo producto recomendado.
2. *No Colaborar*. En cualquier momento, un usuario colaborador es libre de dejar la colaboración y continuar solo el proceso de búsqueda del producto que requiere. En este caso el usuario retorna al **Estado 1**. Se debe tener en consideración que si el usuario líder no tiene más usuarios colaboradores, él también retorna al **Estado 1**.
3. *Comprar*. Esta acción es activada cuando el usuario líder ha encontrado un producto adecuado para comprarlo y finaliza el proceso de recomendación. Al realizar la acción el usuario líder transita al **Estado 3**. Si hubiera usuarios colaboradores, cada uno de ellos transitan al **Estado 1**. Si un usuario colaborador encuentra un producto adecuado, primero debe dejar el **Estado 2** de recomendación colaborativa.
4. *Abandonar*. Esta acción ocurre cuando el usuario líder explícitamente finaliza el proceso de recomendación porque no encuentra un producto adecuado. Después de realizar esta acción, el usuario retorna al **Estado 0**, donde el usuario puede iniciar una nueva recomendación. Cuando el usuario líder realiza esta acción, todos los usuarios colaboradores retornan al **Estado 1** (Recomendación Individual).

En el **Estado 2** un usuario líder puede interactuar con muchos usuarios colaboradores. Del mismo modo, el algoritmo CCR permite que un usuario

colaborador pueda colaborar con muchos usuarios al mismo tiempo, sin embargo como sucede en la realidad, es muy difícil que el usuario pueda seguir con atención más de dos tareas a la vez.

**Estado 3: Comprando.** En este estado el usuario paga el producto seleccionado. Después de esta acción, el usuario transita al **Estado 0** donde en cualquier momento puede iniciar un nuevo proceso de recomendación.

### 3.8.3 Algoritmo CCR

El algoritmo CCR integra información colaborativa *on-line* proveniente de las interacciones entre los usuarios en un entorno virtual colaborativo. Este algoritmo, realiza recomendaciones basadas en las preferencias de los usuarios, tal y como lo realiza el algoritmo IC, que es uno de los mejores algoritmos de recomendación basado en críticas (Reilly et al., 2005a). Sin embargo, el algoritmo CCR tiene la ventaja de integrar información colaborativa que se genera de las interacciones entre las personas que utilizan el recomendador.

Teniendo en consideración que el algoritmo CCR permite recoger información colaborativa in situ de las personas, esta técnica de recomendación puede ser utilizada en diversos entornos colaborativos y sociales que existen actualmente, tales como entornos virtuales de tres dimensiones, redes sociales, juegos en tres dimensiones u otros.

```

Input:  $p_q$ : product query, CB: case base
1 define CollaborativeConversationalRecommendation ( $p_q, CB$ )
2 begin
3    $U_{c_q} \leftarrow \text{null}$  // current critique
4    $U \leftarrow \text{null}$  // user model
5   feedback  $\leftarrow$  "Critiquing"
6   repeat
7      $p_r, CB \leftarrow \text{Recommend}(\text{feedback}, p_q, CB, U_{c_q}, U)$  //  $p_r$ : product recommendation
8      $U_{c_q}, p_r, CB, \text{feedback} \leftarrow \text{Review}(p_r, CB)$ 
9      $p_q \leftarrow \text{Revise}(p_r)$ 
10     $U \leftarrow \text{Remodel}(U, U_{c_q}, p_r, \text{feedback})$ 
11  until UserAccepts( $p_r$ ) or Abandon()
12 end

```

**Algorithm 3:** Algoritmo Collaborative Conversational Recommender (CCR)

En concreto, en esta sección describo el Algoritmo 3, el cual muestra la implementación del algoritmo CCR en un ciclo que está compuesto por cuatro fases (véase las líneas 6-11 del Algoritmo 3): **Recommend**, **Review**, **Revise** y **Remodel**. En la Figura 18, se pueden observar las cuatro fases del algoritmo CCR. En esta figura se puede observar que, a diferencia de los algoritmos tradicionales basados en críticas como el algoritmo IC, el algoritmo CCR puede recibir o entregar información colaborativa a los procesos de recomendación de otros usuarios colaboradores en la fase **Review**. Específicamente, cuando un usuario se encuentra colaborando con otros, y genera una nueva crítica (*feedback*), ésta se transmite a todo el resto de los usuarios colaboradores. A continuación describo en detalle cada una de las fases del Algoritmo 3 en el Algoritmo 4.

La primera fase (**Recommend**) se encarga de recomendar al usuario un nuevo producto,  $p_r$ , teniendo en cuenta el producto actualmente recomendado,  $p_q$ , y la retroalimentación realizada por el usuario durante la sesión de recomendación (véase la línea 7 del Algoritmo 3 y las líneas 1-10 del Algoritmo 4). Como se muestra en el procedimiento **RecommendProduct** (línea 4 y las líneas 42-48 del Algoritmo 4) hay dos tipos de retroalimentación: crítica individual (véase *individual critiquing* en la línea 3 del Algoritmo 4) y colaboración con un usuario host,  $u_h$  (véase *Collaboration* en la línea 6 del Algoritmo 4). En la retroalimentación de *individual critiquing* los usuarios

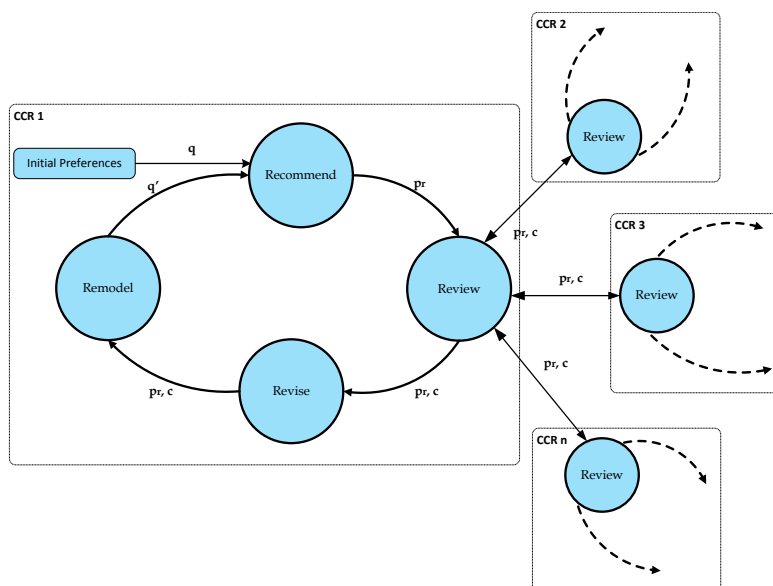


Figura 18: Ciclo de recomendación del Algoritmo CCR

aplican una preferencia dirigida sobre una característica del producto, y en la retroalimentación del tipo *collaboration* los usuarios que se interesan por un producto de otro usuario pasan a ser usuarios colaboradores invitados (es decir, usuarios *guests*),  $u_g$  y comienzan su colaboración con el usuario líder,  $u_h$  (véase los procedimientos **RecommendProduct** y **CollaborativeProduct** en el Algoritmo 4, líneas 42-48 y 49-56). El procedimiento **RecommendProduct** devuelve el nuevo producto recomendado,  $p_r$ , el cual se presenta al usuario. Este producto,  $p_r$ , es seleccionado desde un conjunto de productos relevantes,  $CB_r$ , el cual corresponde a un subconjunto de la base de datos de productos,  $CB$ . Un producto relevante satisface la última crítica realizada por el usuario (véase la línea 44 del Algoritmo 4). La función de satisfacción  $\delta$  es la misma utilizada en el algoritmo IC (véase la Fórmula 3), la cual devuelve 1 si el producto  $p_i$  satisface la crítica  $U_j$  o devuelve 0 si no la satisface. En el algoritmo CCR he utilizado la métrica de calidad (*quality*,  $Q$ ) para ordenar los productos,  $CB_r$  (véase la línea 45 del Algoritmo 4). La métrica de calidad  $Q$  está definida en la Fórmula 1, la cual se basa en la compatibilidad del producto candidato con el modelo de usuario y en la similitud entre el producto actual y el producto candidato. En la Fórmula 1 utilizo la constante  $\beta$  con el valor 0.75 para priorizar los productos candidatos que satisfacen la última crítica. Aunque hay muchas formas de combinar la similitud y compatibilidad, como se muestra en (Reilly et al., 2005a) los mejores resultados se han obtenido con el valor de  $\beta = 0.75$ .

El procedimiento **RecommendProduct** devuelve el producto,  $p_r$ , con la calidad más alta y este producto es mostrado al usuario como la nueva recomendación (véase la línea 46 del Algoritmo 4). El procedimiento **CollaborativeProduct** cambia el actual producto recomendado ( $p_r$ ) del usuario colaborador por el actual producto recomendado al usuario líder ( $p_q$ ) (véase la línea 51 del Algoritmo 4). Si  $p_q$  no existe en la base de datos de productos del usuario colaborador  $CB$ , este producto se agregará en la base de productos como lo muestran las líneas 52-54 del Algoritmo 4. Esta situación es normal y puede producirse cuando un usuario colaborador ha realizado críticas individuales previas que eliminaron un producto de su base de datos

```

Input:  $p_q$ : product query, CB: case base
1 define Recommend(feedback,  $p_q$ , CB,  $U_{c_q}$ ,  $U$ )
2 begin
3   if feedback is Critiquing then
4      $p_r \leftarrow$  RecommendProduct( $p_q$ , CB,  $U_{c_q}$ ,  $U$ )
5   else
6     //feedback is Collaboration with  $u_h$ 
7      $p_r, CB \leftarrow$  CollaborativeProduct(CB,  $p_q$ )
8   end
9   return  $p_r$ , CB
10 end
11 define Review( $p_r$ , CB)
12 begin
13   feedback  $\leftarrow$  ReceiveFeedback()
14   if feedback is Critiquing then
15      $U_{c_q} \leftarrow$  user critique for some  $f \in p_r$ 
16      $p_{new} \leftarrow p_r$ 
17   else
18      $p_{new} \leftarrow$  user likes a product from  $u_h$ 
19     if Collaboration is a new Collaborative Action then
20        $U_{c_q} \leftarrow$  null
21     else
22        $U_{c_q} \leftarrow$  user collaborative critique
23     end
24   end
25   CB  $\leftarrow$  CB -  $\{p_r\}$ 
26   return  $U_{c_q}$ ,  $p_{new}$ , CB, feedback
27 end
28 define Revise( $p_r$ )
29 begin
30    $p_q \leftarrow p_r$ 
31   return  $p_q$ 
32 end
33 define Remodel( $U$ ,  $U_{c_q}$ ,  $p_r$ , feedback)
34 begin
35    $U \leftarrow U -$  contradict( $U$ ,  $U_{c_q}$ ,  $p_r$ )
36    $U \leftarrow U -$  refine( $U$ ,  $U_{c_q}$ ,  $p_r$ )
37   if feedback is Critiquing then
38      $U \leftarrow U \cup (< p_r, U_{c_q} >)$ 
39   end
40   return  $U$ 
41 end
42 define RecommendProduct( $p_q$ , CB,  $U_{c_q}$ ,  $U$ )
43 begin
44    $CB_r \leftarrow \{p \in CB \mid \delta(p, U_{c_q})\}$ 
45    $CB_r \leftarrow$  sort all  $p_i \in CB_r$  in decreasing  $Q(p_i, p_q, U)$  // see Equation 1
46    $p_r \leftarrow$  most quality product in  $CB_r$ 
47   return  $p_r$ 
48 end
49 define CollaborativeProduct(CB,  $p_q$ )
50 begin
51    $p_r \leftarrow p_q$ 
52   if  $p_q \notin CB$  then
53     CB  $\leftarrow$  CB  $\cup \{p_q\}$ 
54   end
55   return  $p_r$ , CB
56 end
57 define ReceiveFeedback()
58 begin
59   feedback  $\leftarrow$  user activates a feedback action
60   return feedback
61 end

```

**Algorithm 4:** Procedimientos de CCR

de productos, pero este producto podría estar presente en la base de datos de productos del usuario líder.

En la segunda fase **Review** (véase las líneas 8 del Algoritmo 3 y las líneas 11-27 del Algoritmo 4) el usuario revisa el producto recomendado y efectúa alguna retroalimentación (*critiquing* o *collaboration*). Esta fase de revisión tiene dos pasos:

- Primero se recibe la retroalimentación por parte del usuario, como se muestra en la línea 13 del Algoritmo 4;



- En segundo lugar se elimina el producto actualmente recomendado,  $p_r$ , desde la base de datos de productos, CB, como se muestra en la línea 25 del Algoritmo 4.

Al eliminar el producto recomendado se evita el problema comúnmente conocido como *unreachability problem*<sup>7</sup>. En el paso de retroalimentación (véase la línea 13 del Algoritmo 4), por un lado el usuario puede realizar *Critiquing* y generar la nueva crítica denominada  $U_{c,q}$  en la línea 15 del Algoritmo 4. Por otro lado, en el caso de la retroalimentación del tipo *Collaboration* (véase la línea 18 del Algoritmo 4), ésta es realizada cuando un usuario inicia una colaboración con otro usuario, que denomino usuario líder, por medio de la acción Colaborar o cuando un usuario colaborador se mantiene colaborando con un usuario líder. En primer lugar, cuando el usuario inicia la colaboración, solo recibe el producto desde el usuario líder y la crítica toma un valor null (véase la línea 20 del Algoritmo 4). En segundo lugar, cuando el usuario ya se encuentra colaborando con un usuario líder, recibe el producto  $p_r$  desde  $u_h$  y la crítica colaborativa que ha consensuado con el usuario  $u_h$ , ésta última pasa a ser la crítica  $U_{c,q}$  del usuario colaborador (véase la línea 22 del Algoritmo 4).

La tercera fase está focalizada en la revisión (**Revise**) del producto *query*,  $p_q$ , para el siguiente ciclo de recomendación y el actual producto recomendado,  $p_r$ , se define como el nuevo producto *query*,  $p_q$ , véase las líneas 9 en el Algoritmo 3 y las líneas 28-32 en el Algoritmo 4).

Finalmente, la última fase (**Remodel**) del ciclo de CCR remodela el modelo de usuario (U) (véase la línea 10 del Algoritmo 3 y las líneas 33-41 del Algoritmo 4). El algoritmo CCR, al igual que el algoritmo IC y el algoritmo HOR, asume que los usuarios pueden refinar sus requerimientos durante el proceso de recomendación porque el usuario va adquiriendo mayor conocimiento de los productos durante el proceso y de esta forma pueden eventualmente cambiar sus preferencias sobre las características del producto (Reilly et al., 2005a). Sin embargo, en el caso del algoritmo CCR esta fase es más compleja que en el algoritmo IC o en el algoritmo HOR, por la naturaleza colaborativa del algoritmo. Específicamente, cuando los usuarios se encuentran colaborando, es necesario actualizar el modelo de usuario del usuario líder ( $u_h$ ) y también el modelo de usuario de todos los colaboradores ( $u_g$ ), teniendo en cuenta tanto la última crítica realizada ( $U_{c,q}$ ) de manera colaborativa, como también el producto recomendado colaborativamente ( $p_r$ ). En particular, el algoritmo CCR utiliza dos tipos de feedback: críticas (individuales o colaborativas) y la acción de iniciar una colaboración como usuario  $u_g$  de un usuario  $u_h$ , que automáticamente es como seleccionar el producto recomendado actualmente al usuario  $u_h$  (acción colaborativa). Por lo anterior, es necesario tener en cuenta ambos tipos de feedback en el momento de realizar el remodelado del modelo de usuario U.

- En primer lugar, el usuario puede realizar críticas individuales, en cuyo caso el procedimiento **Remodel** solo utiliza la crítica realizada,  $U_{c,q}$ , en el llamado al procedimiento **contradict** que permite eliminar las críticas contradictorias y en el procedimiento **refine** para refinar aquellas existentes en el modelo de usuario de la misma forma que en el algoritmo IC (véase las líneas 35 y 36 del Algoritmo 4).

<sup>7</sup> Este problema sucede cuando un usuario después de realizar varios ciclos de recomendación vuelve al producto inicialmente recomendado o una recomendación anterior que el usuario ya había criticado y por ende había descartado. Esto sucede como consecuencia de no eliminar las recomendaciones previas.

- En segundo lugar, el usuario puede estar interactuando colaborativamente con otro usuario. En esta situación, el usuario puede asumir el rol de usuario líder ( $u_h$ ) o usuario colaborador ( $u_g$ ). Cuando el usuario corresponde a  $u_h$ , el procedimiento **Remodel** funciona de la misma forma que en interacciones individuales, es decir, solo se utiliza la crítica ( $U_{c,q}$ ) consensuada con los colaboradores para realizar el remodelado del modelo de usuario del usuario líder,  $U$ . Sin embargo, en el caso del o los usuarios colaboradores  $u_g$ , éstos reciben la nueva crítica  $U_{c,q}$  que se ha consensuado con el líder  $u_h$  y además reciben el nuevo producto recomendado al usuario líder  $p_r$  (véase la línea 33 en el Algoritmo 4). En este caso el procedimiento **Remodel** hace el llamado a las funciones **contradict** y **refine** con los dos valores  $p_r$  y  $U_{c,q}$ . Así, para el producto recibido colaborativamente ( $p_r$ ), la función **contradict** elimina las críticas que contradicen las características del nuevo producto recomendado,  $p_r$ , y la función **refine**, refina las críticas existentes en el modelo de usuario, teniendo como base el nuevo producto  $p_r$ . En el caso de la crítica colaborativa  $U_{c,q}$ , las funciones **contradict** y **refine** utilizan la crítica  $U_{c,q}$  para eliminar las críticas del modelo de usuario que sean contradictorias y refina las críticas de acuerdo a la nueva crítica.

Para clarificar el concepto de eliminación de críticas contradictorias en el algoritmo CCR véase el Ejemplo 5. Del mismo modo para el concepto de refinar críticas previas véase el Ejemplo 6.

- (5) Teniendo un proceso de recomendación colaborativa en un dominio de cámaras fotográficas donde un usuario inicia una colaboración ( $u_g$ ) realizando una acción de colaboración con otro usuario ( $u_h$ ). Si el modelo de usuario de  $u_g$  contiene una crítica definida como [manufacturer, ≠, Sony] y el producto del usuario  $u_h$  que es del agrado del usuario  $u_g$  denominado  $p_r$  es una cámara Sony, la crítica [manufacturer, ≠, Sony] es eliminada desde el modelo de usuario de  $u_g$  porque ésta contradice (*contradicts*) el actual producto recomendado al usuario  $u_h$  (véase la línea 43). Posteriormente,  $u_h$  realiza una crítica [weight, >, 100] (es decir  $U_{c,q} \neq \text{null}$ ) y el modelo de usuario,  $U$ , de  $u_g$  contiene una crítica [weight < 80]. Esto significa, que la crítica [weight, <, 80] contradice la nueva crítica y por esto, esta crítica es eliminada del modelo de usuario de  $u_g$  (véase la línea 35 del Algoritmo 4). La última crítica  $U_{c,q}$  es también agregada al modelo de usuario  $U$  del usuario  $u_g$  y  $u_h$  (véase la línea 38 del Algoritmo 4). Para el caso del usuario líder  $u_h$ , las críticas en su modelo de usuario que sean contradictorias a las nuevas críticas, también son eliminadas.

- (6) Un ejemplo de la acción de refinar críticas puede ser si el modelo de usuario de  $u_g$  contiene una crítica definida como [price, <, \$1500] y el usuario  $u_h$  realiza la crítica [price, <, \$1000]. Esta nueva crítica hará que la crítica en el modelo de usuario de  $u_g$ , [price, <, \$1500], sea refinada a la crítica [price, <, \$1000] (véase la línea 36 del Algoritmo 4).

El algoritmo CCR termina el proceso de recomendación si el usuario encuentra un producto de su agrado o cuando el usuario explícitamente abandona el proceso (véase la línea 11 del Algoritmo 3). Es necesario mencionar que el proceso de recomendación basa las recomendaciones en el modelo de usuario, el cual se actualiza incrementalmente en cada ciclo con el objetivo de mantener un modelo consistente.

### 3.9 INTEGRACIÓN DE INFORMACIÓN COLABORATIVA *off-line* Y *on-line*

En los apartados anteriores, he propuesto nuevas técnicas de recomendaciones basadas en críticas que integran, por separado, información colaborativa *off-line* e información colaborativa *on-line*. En particular, los dos enfoques principales de estudio en esta tesis han sido, por un lado, las técnicas que integran opiniones y sesiones históricas de otros usuarios (técnicas de recomendación HOR y HORi) y por otro lado, la técnica que permite colaboración e interacción in situ entre los usuarios (CCR).

Teniendo en cuenta lo expuesto anteriormente, en esta sección propongo unificar los dos enfoques de estudio en una sola técnica que permita integrar simultáneamente información colaborativa *off-line* e información colaborativa *on-line*. En concreto, propongo una nueva técnica denominada *History and Opinion Collaborative Recommender* (HOCR), la cual en su proceso de recomendación permite integrar información colaborativa proveniente de sesiones históricas y opiniones de otros usuarios (*off-line*), junto con información proveniente de las interacciones y colaboraciones instantáneas que puedan tener los usuarios dentro de entornos colaborativos (*on-line*).

En esta sección entrego el detalle de la técnica propuesta denominada HOCR. En primer lugar realizo una descripción conceptual de la técnica de personalización propuesta y posteriormente, presento el algoritmo que la implementa.

#### 3.9.1 HOCR: Técnica de integración de información colaborativa *off-line* y *on-line*

La Figura 19 muestra el diagrama conceptual de la técnica de recomendación HOCR. En particular, en esta figura ejemplifico el escenario con  $n$  usuarios utilizando el recomendador. Es decir, cada usuario $_i$  tiene un recomendador HOCR $_i$  (véase la parte central de la figura), un modelo de usuario  $U_i$  y una base de productos disponibles para recomendar  $CB_i$  (véase la parte derecha de la figura).

En primer lugar, en la Figura 19 se puede observar que la técnica de recomendación HOCR, al igual que la técnica de recomendación HOR (véase la Sección 3.6), mantiene información colaborativa *off-line* (véase el lado izquierdo de la Figura 19), en la base de datos de sesiones históricas, SB (véase la Definición 2) y en la base de datos de opiniones, OB (véase la Definición 3).

Por otro lado, la Figura 19 muestra en su lado derecho los elementos de información que los usuarios comparten en procesos de recomendación colaborativa *on-line*. En particular, para la información colaborativa *on-line*, tal y como en la técnica de recomendación CCR (véase la Sección 3.8), el algoritmo HOCR permite que los usuarios colaboren y compartan críticas del modelo de usuario  $U_i$  o los productos recomendados que son obtenidos de la base de datos de productos disponibles de cada usuario,  $CB_i$ .

Adicionalmente, la Figura 19 muestra que el algoritmo de recomendación HOCR permite retroalimentaciones colaborativas entre usuarios (véase la flecha nombrada por *collaborative feedback*). Esta característica heredada del algoritmo CCR, permite que las críticas colaborativas y las acciones para iniciar una colaboración sean compartidas entre los distintos recomendadores y así, posteriormente se puedan actualizar información en el ámbito de cada usuario ( $U$  y  $CB$ ).

En HO CR, el manejo de las sesiones históricas en SB y las opiniones en OB se realiza de la misma forma que en HOR (véase la Sección 3.6.2).

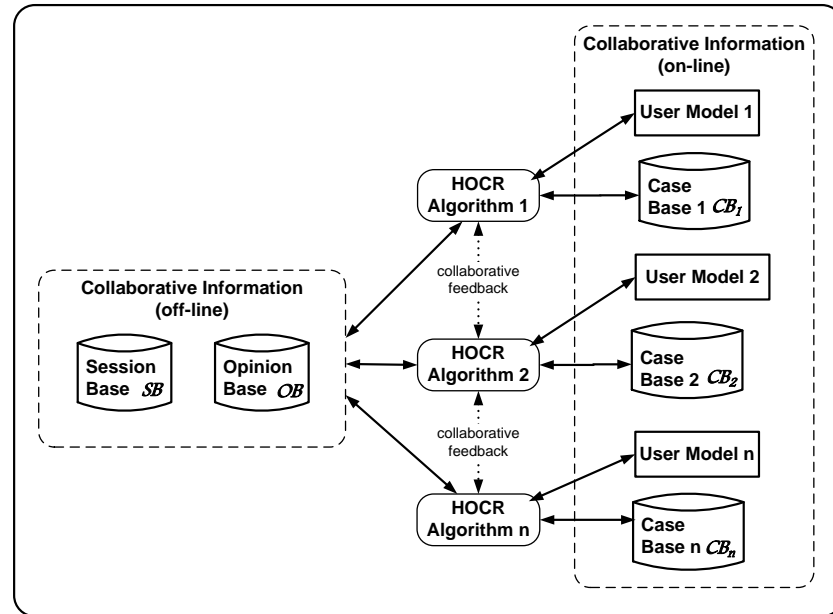


Figura 19: Diagrama conceptual de la técnica de recomendación HO CR

El proceso de recomendación de la técnica de recomendación HO CR consiste en:

1. El recomendador puede realizar recomendaciones individuales y colaborativas. En las recomendaciones colaborativas, la técnica HO CR sigue la misma metodología definida para la técnica CCR. Por ejemplo, en la técnica de recomendación HO CR los usuarios que colaboran, al igual que en CCR, asumen roles de usuario líder  $u_h$  y usuarios colaboradores  $u_g$ .
2. En la técnica HO CR, las recomendaciones se realizan utilizando información colaborativa *off-line* proveniente de las sesiones históricas y de las opiniones de otros usuarios. Este tipo de recomendación se realiza de la misma forma que en la técnica de recomendación HOR, es decir, identificando sesiones históricas relevantes por medio de las Fórmulas 14 y 16. La metodología utilizada es la misma descrita en la Sección 3.6.2. En caso de no identificar sesiones históricas, la técnica HO CR, revierte al algoritmo *incremental critiquing* modificado, que he descrito en la Sección 3.7.
3. La técnica HO CR también permite realizar recomendaciones colaborativas a un grupo de usuarios que se encuentren colaborando en un entorno que favorezca este tipo de interacción. Específicamente, la técnica HO CR permite que un usuario pueda iniciar una colaboración con otro usuario, que automáticamente es como seleccionar el producto recomendado al otro usuario y mantenerse colaborando con éste durante los ciclos de recomendación. Para esto, HO CR define los mismos estados por los cuales transitan los usuarios durante una sesión

de recomendación (véase los estados descritos en la Sección 3.8.2 y en la Figura 17).

4. Finalmente, la técnica HOOCR permite todas las acciones individuales y colaborativas definidas en la técnica CCR (por ejemplo, *críticas individuales*, *críticas colaborativas*, *iniciar colaboración*, etc.). Para mayores detalles véase las acciones descritas en la Sección 3.8.2.

### 3.9.2 Algoritmo HOOCR

En esta sección describo el algoritmo HOOCR que implementa la técnica de recomendación propuesta. Para simplificar, y teniendo en cuenta que el algoritmo HOOCR se basa en el algoritmo HOR y en el algoritmo CCR, los procedimientos o funciones que ya estén descrito y/o definidos en las secciones anteriores, solo serán referenciados en esta sección.

En el Algoritmo 5 se muestra el proceso principal del HOOCR. El algoritmo recibe el producto inicial, denominado *query*,  $p_q$ , el cual puede estar basado en preferencias iniciales definidas explícitamente por el usuario o si se carece de dichas preferencias, este producto inicial puede ser un producto aleatoriamente seleccionado desde la base de datos de productos disponible, CB. Adicionalmente, el algoritmo recibe toda la información requerida para realizar las recomendaciones al usuario: la base de datos de productos (CB), la base de datos de sesiones históricas (SB), y la base de datos de opiniones (OB). Al inicio de la recomendación, no existen críticas generadas por el usuario, por lo tanto la última crítica realizada por el usuario,  $U_{c_q}$ , y el modelo de usuario,  $U$ , son inicializados con el valor inicial por defecto que se muestra en las líneas 3 y 4. Por defecto, al inicio del algoritmo defino la retroalimentación (*feedback*) con el valor de “Critiquing” (véase la línea 5 del Algoritmo 5).

El algoritmo HOOCR está compuesto por 4 fases (véase el Algoritmo 5, líneas 6-11): **Recommend**, **Review**, **Revise** y **Remodel**. Este ciclo se repite hasta que el usuario acepta un producto recomendado o explícitamente abandona el proceso de recomendación. A continuación describo en detalle cada una de las fases del Algoritmo 5 en el Algoritmo 6.

```

Input:  $p_q$ : producto inicial, CB: base de productos, SB: base de sesiones, OB: base de opiniones
1 define HistoryOpinionConversationalRecommender ( $p_q$ , CB, SB, OB)
2 begin
3    $U_{c_q} \leftarrow \text{null}$  // current critique
4    $U \leftarrow \text{null}$  // user model
5    $\text{feedback} \leftarrow \text{“Critiquing”}$ 
6   repeat
7      $p_r, CB \leftarrow \text{Recommend}(\text{feedback}, p_q, CB, U, SB, OB)$  //  $p_r$ : product recommendation
8      $U_{c_q}, p_r, CB, \text{feedback} \leftarrow \text{Review}(p_r, CB)$ 
9      $p_q \leftarrow \text{Revise}(p_r)$ 
10     $U \leftarrow \text{ReModel}(U, U_{c_q}, p_r, \text{feedback})$ 
11  until UserAccepts( $p_r$ ) or UserAbandon()
12 end

```

#### Algorithm 5: Algoritmo de HOOCR

La primera fase corresponde a la fase denominada **Recommend** (véase la línea 7 del Algoritmo 5 y las líneas 1-20 en el Algoritmo 6). Esta fase es la encargada de realizar la recomendación de un nuevo producto  $p_r$  al usuario, seleccionando el producto desde la base de datos de productos CB. El algoritmo HOOCR utiliza dos tipos de retroalimentación: crítica individual (véase *Critiquing* en la línea 3 del Algoritmo 6) y colaboración con un usuario host,  $u_h$  (véase *Collaboration* en la línea 16 del Algoritmo 6).

Si la retroalimentación es una crítica individual (línea 3 del Algoritmo 6, la recomendación de un nuevo producto se basa en: el producto actualmente recomendado  $p_q$ , las sesiones históricas de otros usuarios almacenadas en SB, las opiniones de otros usuarios almacenadas en OB, y la sesión actual de recomendación que consiste en pares compuestos por el producto recomendado y la crítica aplicada a dicho producto (esta información se encuentra almacenada en el modelo de usuario U). Esta recomendación se efectúa siguiendo la misma metodología definida para el algoritmo HOR (véase la Sección 3.6.3). De manera resumida, HO CR selecciona las sesiones relevantes desde la base de datos de sesiones, SB, por medio del procedimiento **RelevantSessions** que he descrito anteriormente en el Algoritmo 2. Si existen sesiones relevantes, éstas se ordenan de acuerdo a la medida **RecScore** definida en la Fórmula 18. En caso que no existan sesiones relevantes, se obtiene el producto para recomendar,  $p_r$ , utilizando como medida de calidad la descrita en la Fórmula 19, que corresponde al algoritmo IC modificado. Esta medida también la he utilizado en las técnicas de recomendación HORi y HGRi.

Si la retroalimentación es una acción de colaboración (línea 16 del Algoritmo 6), la recomendación se obtiene por medio del procedimiento **CollaborativeProduct** (véase las líneas 66-73 del Algoritmo 6). Este procedimiento cambia el actual producto recomendado ( $p_r$ ) del usuario colaborador por el producto recomendado al usuario líder,  $p_q$  (véase la línea 68 del Algoritmo 6). Si  $p_q$  no existe en la base de datos de productos del usuario colaborador CB, este producto se agregará en la base de productos como lo muestra la línea 70 del Algoritmo 6.

En la segunda fase **Review** (véase las líneas 8 del Algoritmo 5 y las líneas 35-51 del Algoritmo 6) el usuario revisa el producto recomendado y efectúa alguna retroalimentación (*critiquing* o *collaboration*). Esta fase de revisión tiene dos pasos:

- Primero se recibe la retroalimentación por parte del usuario, como se muestra en la línea 37 del Algoritmo 6;
- En segundo lugar se elimina el producto actualmente recomendado,  $p_r$ , desde la base de datos de productos, CB, como se muestra en la línea 49 del Algoritmo 6.

Cuando el usuario realiza una retroalimentación (véase la línea 37 del Algoritmo 6), ésta puede corresponder a *Critiquing* y de esta forma se genera la nueva crítica denominada  $U_{c,q}$  en la línea 39 del Algoritmo 6. Por otra parte, la retroalimentación también puede corresponder a *Collaboration* (véase la línea 42 del Algoritmo 6), en cuyo caso ésta es realizada cuando un usuario inicia una colaboración con otro usuario, que denomino usuario líder, por medio de la acción Colaborar o cuando un usuario colaborador se mantiene colaborando con un usuario líder. En primer lugar, cuando el usuario inicia la colaboración, solo recibe el producto desde el usuario líder y la crítica toma un valor null (véase la línea 44 del Algoritmo 6). En segundo lugar, cuando el usuario ya se encuentra colaborando con un usuario líder, recibe el producto  $p_r$  desde  $u_h$  y la crítica colaborativa que ha consensuado con el usuario  $u_h$ , ésta última pasa a ser la crítica  $U_{c,q}$  del usuario colaborador (véase la línea 46 del Algoritmo 6).

La tercera fase, denominada **Revise** (véase la línea 9 del Algoritmo 5 y las líneas 52-56 del Algoritmo 6), se focaliza en la revisión de la *query* del producto  $p_q$  para el siguiente ciclo. Esta fase no sufre modificaciones respecto a HOR y CCR.

```

1 define Recommend(feedback, pq, CB, U, SB, OB)
2 begin
3   if feedback is Critiquing then
4     SREL ← RelevantSessions(U, SB)
5     if SREL ≠ null then
6       SREL ← sort si ∈ SREL in decreasing RecScore(U, si, oj) //see Equation 18
7       rF ← store pj = term(si)/si ∈ SREL
8       pr ← getTopRanked(rF)
9     else
10      Ucq ← getLastCritique(U)
11      CB' ← {p ∈ CB | δ(p, Ucq)}
12      CB' ← sort all pi ∈ CB' in decreasing QHORi(pi, pq, U) //see Equation 19
13      pr ← most quality product in CB'
14    end
15  else
16    //feedback is Collaboration with uh
17    pr, CB ← CollaborativeProduct(CB, pq)
18  end
19  return pr, CB
20 end
21 define RelevantSessions (U, SB)
22 begin
23   SREL ← null
24   SREL ← {si ∈ SB : OverlapPair(U, si) > 0} //see Equation 14
25   if SREL == null then
26     SREL ← {si ∈ SB : OverlapCritique(U, si) > 0} //see Equation 16
27   end
28   if SREL ≠ null then
29     for si ∈ SREL do
30       SREL ← SREL - contradict (term(si), U)
31     end
32   end
33   return SREL
34 end
35 define Review (pr, CB)
36 begin
37   feedback ← ReceiveFeedback()
38   if feedback is Critiquing then
39     Ucq ← user critique for some
40     f ∈ pr
41     pnew ← pr
42   else
43     pnew ← user likes a product from
44     uh
45     if Collaboration is a new Collaborative
46     Action then
47       Ucq ← null
48     else
49       Ucq ← user collaborative
50       critique
51     end
52   end
53   CB ← CB - pr
54   return Ucq, pnew, CB, feedback
55 end
56 end
57 define ReModel (U, Ucq, pr, feedback)
58 begin
59   U ← U - contradict (U, Ucq, pr)
60   U ← U - refine (U, Ucq, pr)
61   if feedback is Critiquing then
62     U ← U ∪ (< pr, Ucq >)
63   end
64   return U
65 end
66 define CollaborativeProduct (CB, pq)
67 begin
68   pr ← pq
69   if pq ∉ CB then
70     CB ← CB ∪ {pq}
71   end
72   return pr, CB
73 end
74 define ReceiveFeedback ()
75 begin
76   feedback ← user activates a feedback
77   action
78   return feedback
79 end

```

Algorithm 6: Procedimientos de HO CR

Finalmente, la fase denominada **Remodel** (véase la línea 10 del Algoritmo 5 y las líneas 57-65 del Algoritmo 6) es la encargada de guardar tanto el producto recomendado,  $p_r$ , como la última crítica realizada por el usuario,  $U_{cq}$ , en el modelo de usuario,  $U$ . En el algoritmo HO CR se utiliza la misma metodología aplicada en HOR y CCR para mantener el modelo de usuario. Es decir, en HO CR antes de guardar la última crítica en el modelo de usuario, se utilizan las funciones **contradict** y **refine** que he descrito previamente en el algoritmo CCR, para eliminar o modificar críticas incompatibles con la última crítica  $U_{cq}$  o con el producto seleccionado de otro usuario en caso de recomendaciones colaborativas. Una vez realizada la validación del modelo de usuario, si el *feedback* es una crítica, se procede a guardar la crítica

y el producto recomendado en el modelo de usuario (véase la línea 62 del Algoritmo 6).

La técnica HOOCR corresponde a la última técnica de recomendación que propongo en esta tesis, la cual integra información colaborativa *off-line* y *on-line*.

### 3.10 CONCLUSIONES

En este capítulo he presentado las nuevas técnicas de recomendación y personalización que propongo con el objetivo de mejorar la eficiencia en la recomendación y facilitar la experiencia del usuario en el uso de un sistema de recomendación.

Las técnicas que he propuesto tienen como base fundamental la naturaleza colaborativa del ser humano. Adicionalmente, mis propuestas buscan aprovechar los estudios e investigaciones previas, que han demostrado resultados satisfactorios, con el objetivo de integrarlas en las nuevas técnicas de recomendación propuestas.

En concreto, las técnicas propuestas integran información colaborativa que puede provenir de información histórica asociada a recomendaciones previas realizadas a otros usuarios, como de las opiniones que se puedan obtener de manera *off-line*, o por otro lado, a través de las propias interacciones que se generen entre los usuarios durante la recomendación (*on-line*). Brevemente, las técnicas que he descrito en este capítulo son:

- **PerTecCB, PerIndCB.** Integración de opiniones de usuario dentro de la base de productos.
- **LwOpinion, UserPerWeight.** Integración de opiniones de usuario dentro del proceso de recomendación a través de técnicas de ponderación de características.
- **HOR.** Integración de información colaborativa proveniente de sesiones de recomendación que otros usuarios realizaron anteriormente y de las opiniones que tienen otros usuarios respecto a los productos disponibles.
- **HORi, HGRi.** Integración de avances de estudios previos en las propuestas de recomendación basadas en sesiones históricas.
- **CCR.** Algoritmo que integrado en un *framework* colaborativo permite la interacción y colaboración entre los usuarios, aprovechando de esta forma la naturaleza colaborativa de las personas para facilitar la búsqueda de un producto.

Para cada una de las técnicas propuestas he presentado los antecedentes, los fundamentos conceptuales, teóricos y matemáticos que la sustentan. Adicionalmente, he presentado una técnica que integra las principales técnicas propuestas en esta tesis, que he denominado HOOCR.

- **HOOCR.** Esta técnica está basada por un lado, en sesiones de recomendación históricas y en opiniones de otros usuarios (colaboración *off-line*), y por otro lado, en la interacción colaborativa que se genera en entornos virtuales de tres dimensiones (colaboración *on-line*).



Resumiendo, las nuevas técnicas de recomendación que he propuesto en esta tesis representan una gran contribución en los estudios asociados a la colaboración que pueden realizar las personas en sistemas de recomendación basados en críticas. En primer lugar, para las técnicas de recomendación *off-line* se han propuesto nuevas técnicas que integran este tipo de información tanto a nivel de bases de datos como del proceso de recomendación, además he integrado técnicas previas que han demostrado obtener buenos resultados en el mejoramiento de las recomendaciones. En segundo lugar, he propuesto la primera técnica de recomendación que integra información colaborativa *on-line*, ya que ninguno de los estudios previos en recomendadores basados en críticas ha realizado integración de este tipo información en sus procesos de recomendación. En tercer lugar, todas mis propuestas buscan integrar estudios previos que se encuentran disgregados y que representan una alternativa real para mejorar la eficiencia de las recomendaciones. Finalmente, he integrado las principales técnicas propuestas en una sola que reúne los dos tipos de información colaborativa (*off-line, on-line*) y que integra estudios previos que han demostrado mejoras en los procesos de recomendación basados en críticas.

En el siguiente capítulo presento los entornos de visualización desarrollados para la implementación de las técnicas propuestas.

## 4

## ENTORNOS DE VISUALIZACIÓN

## Resumen

En este capítulo presento los entornos de visualización y las interfaces de usuario inteligentes en las cuales se integran las técnicas de personalización descritas en el capítulo anterior. En concreto, propongo las interfaces de usuario inteligentes para distintos entornos de visualización, que permitirán a los usuarios interactuar con las técnicas descritas en el capítulo anterior. Para cada propuesta de entorno de visualización y la correspondiente interfaz de usuario inteligente se describe su arquitectura, las funcionalidades y los elementos de interacción que favorecen el uso, por parte de los usuarios, de las nuevas técnicas de personalización propuestas en esta tesis. Los resultados obtenidos durante el diseño, desarrollo e implementación de estas propuestas se encuentran publicados en:

- Salamó, M., Puig, A., and Contreras, D. (2013). An approach to couple two forms of feedback in conversational recommenders. In *Proceedings of the XVI International Conference of the Catalan Association for Artificial Intelligenc*, pages 273–282
- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2013). Integrating a Conversational Recommender System within a 3D Collaborative Environment. In *Proceedings of the XV Conference of Spanish Association for Artificial Intelligence*, pages 89–98
- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2014). An approach to improve user experience with conversational recommenders through a 3D virtual environment. In *Proceedings of the XV International Conference on Human Computer Interaction*, pages 403–406
- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2015b). A 3D visual interface for critiquing-based recommenders: Architecture and interaction. *International Journal of Artificial Intelligence and Interactive Multimedia*, 3(3):pages 7–15
- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2015c). Supporting Users Experience in a 3D eCommerce Environment. In *Proceedings of the XVI International Conference on Human Computer Interaction*, pages 383–386
- Contreras, D. and Salamó, M. (2017). Integrating History and User Opinion for Conversational Recommendations. *ACM Transactions on Intelligent Systems and Technology*. Manuscript submitted for revision in July 2016. JCR IF=2.414
- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2017). Implementation and Evaluation of a Collaborative Conversational Recommender in a 3D Virtual World. *IEEE Consumer Electronics Magazine. Special Issue Advanced Interaction and Virtual/Augmented Reality*. Manuscript submitted for revision in December 2016

## 4.1 INTRODUCCIÓN

EL OBJETIVO DE ESTE CAPÍTULO es presentar y describir en detalle los entornos de visualización utilizados para integrar las técnicas de recomendación presentadas en el capítulo anterior. En concreto, en este capítulo presento tres interfaces de usuario inteligentes en entornos de dos y tres dimensiones, que permiten a los usuarios interactuar con las técnicas de recomendación propuestas. En el ámbito de los sistemas de recomendación basados en críticas, la mayoría de los estudios previos han implementado recomendadores en entornos de dos dimensiones (Burke et al., 1997; Burke, 2002b; Zhang and Pu, 2006; Pu et al., 2008; Chen and Pu, 2009). Teniendo esto en cuenta, mis primeras dos aproximaciones de interfaz de usuario inteligente para recomendaciones, se basaron en entornos de visualización en dos dimensiones (Secciones 4.2 y 4.3).

Considerando las limitaciones colaborativas de las interfaces en dos dimensiones, hoy en día surgen nuevos entornos que facilitan la interacción, la comunicación y la colaboración entre los usuarios. Estos nuevos entornos corresponden a plataformas inmersivas en tres dimensiones, conocidos habitualmente como entornos virtuales inteligentes o mundos virtuales en tres dimensiones.

Por lo anterior, propongo adicionalmente un entorno virtual colaborativo en tres dimensiones que facilita la interacción con el recomendador y con otros usuarios, como así mismo favorece la colaboración entre los usuarios presentes en el entorno virtual (Sección 4.4).

## 4.2 INTERFAZ DE SELECCIÓN DE PRODUCTOS

En esta sección describo una interfaz en un entorno de dos dimensiones, que ofrece al usuario distintas vistas para la interacción con el recomendador. Para esta interfaz, he propuesto una modificación al algoritmo tradicional de recomendación IC (Reilly et al., 2005a). Específicamente, propongo una nueva forma de retroalimentación por parte del usuario que complemente las tradicionales críticas a los productos. En concreto, propongo otorgar al usuario la posibilidad de seleccionar un producto distinto al producto recomendado. Esta nueva forma de retroalimentación establece una nueva técnica de recomendación, ya que el algoritmo de recomendación no recibe una crítica del producto, sino un nuevo producto seleccionado por el usuario. En esta técnica de recomendación, cuando el usuario selecciona un producto por la interfaz propuesta, el modelo de usuario se debe actualizar de acuerdo al producto seleccionado. Es decir, se deben eliminar las inconsistencias que el producto seleccionado tenga con respecto a las críticas que se encuentran almacenadas en el modelo de usuario.

A continuación, entrego una breve descripción de la interfaz utilizada para integrar la nueva propuesta de selección de productos por parte del usuario.

La interfaz de usuario (véase la Figura 20) es una aplicación que ofrece tres paneles de interacción al usuario, con el objetivo de ampliar el conocimiento del usuario respecto al espacio de búsqueda y otorgar al usuario la posibilidad de seleccionar un producto específico para avanzar dentro del espacio de búsqueda. Específicamente, los paneles disponibles son: un panel de vista actual, un panel de vista local y un panel de vista global.

El panel de vista actual tiene las mismas características de una interfaz tradicional de *critiquing* (véase la Figura 20 (b)). Esta vista permite al usua-

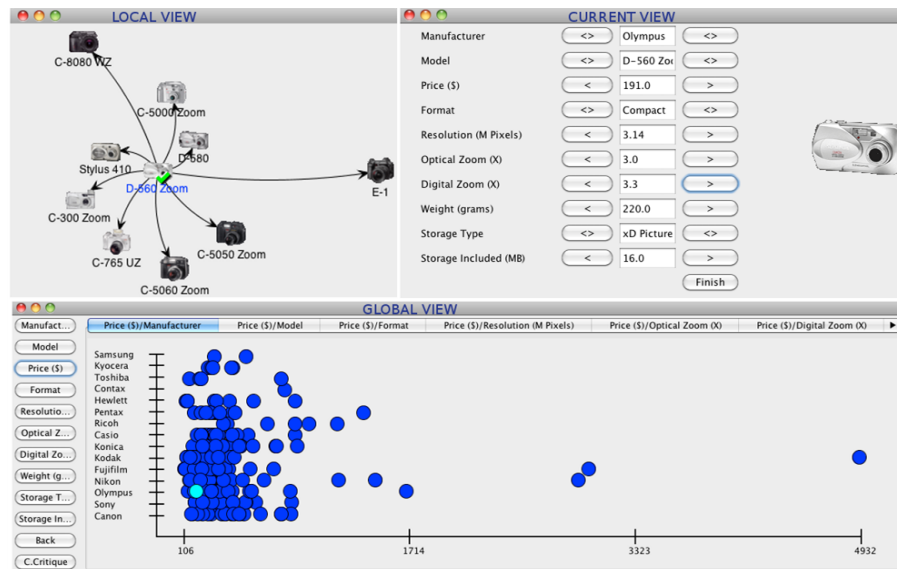


Figura 20: Interfaz de recomendación con múltiples vistas: (a) vista local arriba a la izquierda, (b) vista actual arriba a la derecha, (c) vista global abajo y centrado

rio ver el producto actualmente recomendado con el valor de cada una de sus características. Además, la vista actual utiliza un mecanismo de críticas unitarias a través de los botones que se encuentran en cada característica. De esta manera, el usuario puede visualizar detalles del producto, puede realizar una crítica y también puede terminar la sesión de recomendación.

El panel de vista local (véase la Figura 20 (a)) permite al usuario moverse entre los productos similares al producto recomendado. De esta manera, los usuarios pueden explorar y seleccionar directamente un producto candidato que se encuentre cerca del producto recomendado. Esta vista hace uso de la función de calidad ( $Q$ ) definida en la Fórmula 1 para establecer la relación de cercanía entre el producto candidato y el producto recomendado. Como se puede deducir, es en esta vista donde se requiere del nuevo mecanismo de retroalimentación que permita al usuario seleccionar un producto distinto al producto recomendado. Es importante destacar, que una vez seleccionado el producto por parte del usuario, se debe actualizar el modelo de usuario para eliminar cualquier inconsistencia entre las críticas guardadas en el modelo de usuario y el producto seleccionado.

El panel de vista global presenta un gráfico de dispersión que muestra valores para dos características seleccionadas del producto recomendado (véase la Figura 20 (c)). Este tipo de gráficos ha sido utilizado previamente en otros trabajos de recomendaciones conversacionales (Pu and Faltings, 2000; Averjanova et al., 2008; Zhang et al., 2008). Estos trabajos previos utilizan este tipo de gráficos para mejorar la visualización del espacio de búsqueda por parte del usuario y de esta forma otorgarle un mejor conocimiento de cómo están distribuidos los productos y hacia dónde va cada preferencia. Adicionalmente, en la vista global cuando el usuario selecciona un valor desde el gráfico, los ejes del gráfico dan una nueva crítica compuesta por dos características del producto. Con esta crítica compuesta se eliminan las inconsistencias en el modelo de usuario y se almacenan las críticas presentes en la crítica compuesta.

En general, el objetivo de esta interfaz de usuario es ampliar el conocimiento del usuario acerca de los productos disponibles. En el gráfico del panel de vista global, los ejes horizontales y verticales pueden ser modificados utilizando las pestañas superiores y los botones del lado derecho. Adicionalmente, a la vista actual y local, el panel de vista global permite a los usuarios realizar "saltos" sin restricciones en el espacio completo de productos. Por lo anterior, esta vista es apropiada para localizar, con un mínimo ingreso de información de parte del usuario, sub-espacios de productos que satisfagan un valor de una característica predefinida. Debo mencionar, que aparte de la interfaz de usuario inteligente, esta propuesta implicó modificar el algoritmo de recomendación para ofrecer al usuario un nuevo mecanismo de *feedback* que le permitiera seleccionar un producto directamente desde la base de datos de productos, sin necesidad de generar una crítica.

### 4.3 E-WEBCO: RECOMENDADOR WEB PARA COMERCIO ELECTRÓNICO

En esta sección propongo un *framework* multilingüe en dos dimensiones denominado *e-WebCo*, el cual permite integrar distintos algoritmos o técnicas de recomendación y además permite utilizar distintos dominios de productos. En particular, el *framework* ha sido diseñado para integrar principalmente algoritmos de recomendación que utilicen información colaborativa *off-line* proveniente de sesiones históricas de recomendaciones y de opiniones de otros usuarios. Para este trabajo en concreto he integrado el algoritmo IC y las técnicas de recomendación basadas en sesiones históricas y opiniones (HOR, HGR, HOR-I, y HGR-I), que he propuesto en la Sección 3.6.2. En primer lugar describo la arquitectura conceptual del *framework* y posteriormente describo los objetos de interacción que el *framework*, a través de una interfaz de usuario inteligente, ofrece a los usuarios.

#### 4.3.1 Arquitectura Conceptual

La arquitectura conceptual de *e-WebCo* está basada en un entorno web en dos dimensiones. Esta arquitectura sigue el modelo cliente servidor que favorece la modularidad del recomendador. La Figura 21 muestra la arquitectura conceptual de mi propuesta, la cual está dividida en tres capas: la Capa Cliente (*Space Client*), la Capa Servidor (*Space Server*) y la Capa del Recomendador (*Recommender Space*).

##### 4.3.1.1 Capa Cliente

La Capa Cliente (*Space Client*) se encarga de gestionar la interacción con el usuario. Es decir, esta capa se encarga de capturar las interacciones del usuario con la interfaz, recoger y mostrar las recomendaciones. Esta capa permite conexiones concurrentes de muchos usuarios, tal como lo muestra la parte superior de la Figura 21. Sin embargo, este *framework* no permite a los usuarios interactuar entre ellos. En la Capa Cliente los usuarios interactúan con los objetos de recomendación, los cuales son independientes para cada usuario conectado.

Cada objeto de recomendación (*Recommendation Object*, RO), es un objeto que presenta un producto recomendado en la interfaz. Este objeto contie-

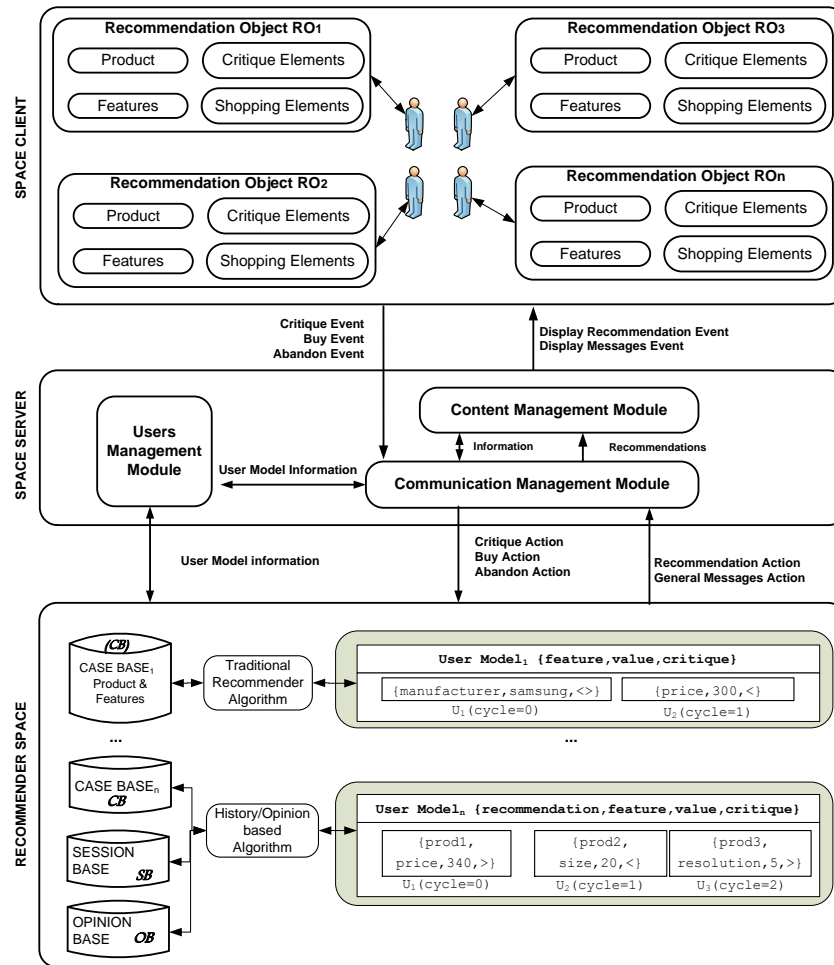


Figura 21: Arquitectura Conceptual del Framework e-WebCo

ne información del producto recomendado al usuario (*product*) y el detalle de sus características (*features*). Además incluye elementos que permiten la interacción del usuario con el recomendador (*Critique Elements* y *Shopping Elements*). Por un lado, los elementos denominados *Critique Elements* permiten al usuario realizar críticas sobre el producto recomendado. Por ejemplo, en el dominio de teléfonos móviles inteligentes (SMARTPHONE), la información del producto incluye una imagen del producto, las características que lo describen (por ejemplo, precio-1000\$) y los elementos interactivos disponibles para realizar críticas (mayor que o menor que un valor numérico o diferente si la característica es un valor nominal). Un ejemplo de característica numérica en la base de datos SMARTPHONE puede ser el precio, y una característica nominal puede ser el modelo del teléfono. Por otro lado, los elementos definidos como *Shopping Elements* permiten al usuario realizar dos acciones: comprar el producto actualmente recomendado o abandonar la sesión de recomendación.

Las interacciones de los usuarios en la Capa Cliente generan tres tipos de eventos: los eventos de críticas (*Critique Event*), los eventos de compra de producto (*Buy Event*) y los eventos para abandonar una sesión de recomendación (*Abandon Event*). Estos eventos son enviados a la Capa Servidor

(*Server Space*), como se muestra en la flecha que va desde la Capa Cliente a la Capa Servidor en la Figura 21. Por un lado, el evento de crítica ocurre cuando el usuario realiza una crítica a través de un elemento de crítica (*Critique Elements*), en el objeto de recomendación, RO. Por otro lado, el evento de compra (*Buy Event*) ocurre cuando el usuario compra un producto de su agrado. Finalmente, el evento de abandono (*Abandon Event*) ocurre cuando el usuario finaliza la interacción con el recomendador porque no encuentra ningún producto acorde con sus requerimientos.

Adicionalmente, la Capa Cliente recibe desde la Capa Servidor el evento que muestra una nueva recomendación denominado *Display Recommendation Event* (véase la flecha que va desde la Capa Servidor a la Capa Cliente en la Figura 21). En concreto, este evento se encarga de mostrar un nuevo producto recomendado en el objeto de recomendación (RO).

#### 4.3.1.2 Capa Servidor

La Capa Servidor (*Space Server*) que se muestra en el recuadro del medio de la Figura 21, actúa como un controlador y comunicador entre el usuario y el recomendador. En la Figura 21, se muestran los eventos entre la Capa Cliente (*Space Client*) y la Capa Servidor, como también las acciones entre la Capa Servidor y la Capa del Recomendador (*Recommender Space*). Esta capa tiene tres componentes, el primer componente es el Módulo de Administración de Usuarios (*Users Management Module*), el cual guarda y administra la información de los usuarios tales como su identificación y los estados por los cuales transita durante una sesión de recomendación. El segundo componente es el Módulo de Administración de Contenidos (*Content Management Module*), el cual se encarga de actualizar la visualización de información en el objeto de recomendación durante cada ciclo de recomendación (como por ejemplo, la imagen del producto recomendado y sus características). El tercer componente corresponde al Módulo de Administración de Comunicaciones (*Communication Management Module*), el cual asocia a cada evento de la Capa Cliente una acción de recomendación en la Capa del Recomendador, y en forma inversa, asocia acciones de recomendación (véase la flecha *Recommendation Action* que va desde la Capa del Recomendador hasta la Capa Servidor en la Figura 21) con eventos del tipo *Display Recommendation Event*, a través del módulo *Content Management Module*.

A continuación describo los eventos que el usuario activa en la Capa Cliente (véase los eventos en la flecha que va desde la Capa Cliente a la Capa Servidor en la Figura 21) y que se traducen en acciones enviadas a la Capa del Recomendador (véase las acciones *Critique Action*, *Buy Action* y *Abandon Action* en la flecha que va desde la Capa Servidor a la Capa del Recomendador en la Figura 21).

1. El evento de crítica (*Critique Event*) se convierte a la acción de crítica (*Critique Action*) como se describe en la Fórmula 20. Esta acción contiene el usuario que realiza la crítica (*userId*), el producto recomendado (*productId*), el objeto de recomendación donde se realiza la crítica (*recObjectId*), la característica sobre la cual se realiza la crítica (*featureId*, por ejemplo el precio, *price*), el tipo de crítica (*typeCritique*, por ejemplo, más grande que,  $\uparrow$  o menor que,  $\downarrow$ ), y el valor actual de la característica criticada (*critiqueValue*).

$$\text{CritiqueAction}(\text{userId}, \text{recObjectId}, \text{productId}, \text{featureId}, \\ \text{typeCritique}, \text{critiqueValue}) \quad (20)$$

Por ejemplo, la acción  $\text{critiqueAction}(\text{user1}, \text{LGG3}, \text{price}, \downarrow, 300)$ , significa que el usuario  $\text{user1}$  envía una crítica acerca del producto  $\text{LGG3}$  para obtener una nueva recomendación con un precio menor al precio del producto actual, 300\$. Además, esta crítica será guardada en el modelo del usuario (U).

2. El evento de compra de un producto (*Buy Event*) se asocia a la acción de compra denominada *Buy Action*, la cual se describe en la Fórmula 21. Esta acción contiene el usuario que realiza la acción de compra ( $\text{userId}$ ), el objeto de recomendación donde se realiza la acción y el producto seleccionado por el usuario ( $\text{productSelectedId}$ ).

$$\text{BuyAction}(\text{userId}, \text{recObjectId}, \text{productSelectedId}) \quad (21)$$

Por ejemplo, la acción  $\text{BuyAction}(\text{user1}, \text{LGG3})$  significa que el usuario  $\text{user1}$  ha seleccionado el producto  $\text{LGG3}$  como su producto requerido. De esta forma la sesión de recomendación finaliza.

3. El evento de abandonar la sesión de recomendación (*Abandon Event*) se asocia a la acción de finalización denominada *Abandon Action*, la cual se describe en la Fórmula 22. Esta acción contiene el usuario que realiza la acción de abandono de la sesión de recomendación ( $\text{userId}$ ), y el objeto de recomendación en el cual el usuario finaliza la sesión de recomendación ( $\text{recObjectId}$ ).

$$\text{AbandonAction}(\text{userId}, \text{recObjectId}) \quad (22)$$

Por ejemplo, la acción  $\text{AbandonAction}(\text{user1}, \text{Rec01})$  significa que el usuario  $\text{user1}$  ha abandonado la sesión de recomendación en el recomendador  $\text{Rec01}$ . De esta forma la sesión de recomendación finaliza.

A continuación describo las acciones que van desde la Capa del Recomendador (*Recommender Space*) hacia la Capa Servidor (*Server Space*). Específicamente, estas acciones corresponden a las acciones de recomendación y las acciones para envío de mensajes con información general para el usuario (véase la flecha que va desde la Capa del Recomendador a la Capa Servidor en la Figura 21). Además, describo en este apartado, como estas acciones son enviadas a la Capa Cliente (*Space Client*).

1. La acción de recomendación (*Recommendation Action*) se transforma dentro del Módulo de Administración de Comunicaciones al evento para mostrar la recomendación al usuario (*Display Recommendation Event*) descrito previamente en la Sección 4.3.1.1. Como se muestra en la Fórmula 23, esta acción contiene el usuario que realizó la crítica,  $\text{userId}$ , el objeto de recomendación donde el usuario hizo la crítica,  $\text{recObjectId}$ , el nuevo producto recomendado,  $\text{productId}$ , y una lista de los valores para cada característica del nuevo producto recomendado,  $\text{featureValues}$ . Posteriormente, desde el Módulo de Administración de Comunicaciones, el evento es enviado al Módulo de Administración de Contenidos, el cual se encarga de actualizar la visualización en el RO de la imagen del producto y los valores de las características del producto recomendado activando el evento *Display Recommendation*



*Event* (véase la flecha que va desde la Capa Servidor a la Capa Cliente en la Figura 21).

$$\text{RecommendationAction}(\text{userId}, \text{recObjectId}, \text{productId}, \text{featureValues}) \quad (23)$$

- La acción de envío de mensajes (*General Messages Action*) se transforma dentro del Módulo de Administración de Comunicaciones, al evento que sirve para mostrar cualquier tipo de información al usuario por medio de mensajes que denomino *General Messages Event*. En particular, este tipo de mensajes sirven para entregar información al usuario cuando inicia una sesión de recomendación, cuando finaliza una sesión o en general cualquier tipo de notificación que la Capa del Recomendador deba notificar al usuario en la Capa Cliente. Como se muestra en la Fórmula 24, esta acción contiene el usuario que es destinatario del mensaje, *userId* y el cuerpo del mensaje, *textMessage*. Posteriormente, desde el Módulo de Administración de Comunicaciones, el evento es enviado al Módulo de Administración de Contenidos, el cual se encarga de mostrar el mensaje al usuario en el RO por medio del evento *Display Messages Event* (véase la flecha que va desde la Capa Servidor a la Capa Cliente en la Figura 21).

$$\text{MessagesAction}(\text{userId}, \text{textMessage}) \quad (24)$$

#### 4.3.1.3 Capa del Recomendador

La Capa del Recomendador (*Recommender Space*) que se muestra en el recuadro inferior de la Figura 21, está compuesta por tres componentes: las bases de datos de productos, el algoritmo de recomendación y el modelo de usuario definido para cada usuario.

El primer componente corresponde a las bases de datos que el algoritmo de recomendación utiliza durante las sesiones de recomendación. En primer lugar, cualquier tipo de algoritmo de recomendación utiliza la base de datos de productos CB (véase la Definición 1 en la Sección 3.2), la cual contiene el conjunto de productos que están disponibles para recomendar a cada usuario en cada ciclo de recomendación. Adicionalmente, los algoritmos de recomendación que integran información colaborativa *off-line*, tales como HOR, HGR, HORi y HGRi, utilizan bases de datos adicionales para almacenar este tipo de información. En concreto, para algoritmos basados en información colaborativa *off-line*, la Capa del Recomendador tiene una base de datos de sesiones históricas SB, descrita en la Definición 2, y una base de datos de opiniones de otros usuarios, OB, descrita en la Definición 3 (véase los dos tipos de bases de datos en el lado izquierdo de la Capa del Recomendador en la Figura 21).

El segundo componente corresponde al algoritmo de recomendación, el cual puede corresponder a un algoritmo tradicional de recomendación basado en críticas, como por ejemplo el algoritmo IC o cualquier otro algoritmo de recomendación basado en información colaborativa *off-line* tales como HOR, HGR, HORi, HGRi u otro. Es necesario destacar, que el *framework* propuesto permite la integración de diferentes algoritmos de recomendación basados en críticas (véase los dos tipos de recomendadores en los recuadros del medio de la Capa del Recomendador en la Figura 21).

Finalmente, el tercer componente corresponde al modelo de usuario,  $U$  (véase la Definición 5 en la Sección 3.2). Este modelo es usado por el recomendador en cada ciclo para almacenar las preferencias del usuario. Este modelo contendrá la información que establezca cada algoritmo específico, por ejemplo en el caso del algoritmo IC, que corresponde a un algoritmo tradicional basado en críticas, solo se almacenan las críticas realizadas por el usuario (véase el recuadro  $UserModel_1$  que se muestra como ejemplo en la Figura 21), sin embargo en un algoritmo basado en información colaborativa *off-line* como el HOR, se guarda la crítica y el producto recomendado, tal como lo he descrito en la Sección 3.6 (véase el recuadro  $UserModel_n$  que se muestra como ejemplo en la Figura 21).

#### 4.3.2 Objetos de Recomendación

La interfaz de usuario de **e-WebCo** se muestra en la Figura 22. Esta interfaz está compuesta por los siguientes elementos visuales:

- a. Corresponde a una imagen del producto actualmente recomendado al usuario. Esta imagen puede ser un vídeo, imagen, o una imagen en 3D.
- b. Corresponde a iconos visuales que representan a las características del producto. He utilizado iconos visuales, ya que se ha demostrado en estudios previos que los usuarios valoran positivamente esta representación frente a una descripción textual (Zhang et al., 2008).
- c. Implementa a través de botones, los elementos para realizar críticas (*Critique Event*) denominados *Critique Elements* en la arquitectura conceptual que he descrito en la Sección 4.3.1. Estos botones permiten al usuario realizar las críticas sobre la característica correspondiente a través de uno (<>) o dos (+,-) botones. Este modelo sigue la metodología utilizada en estudios previos de interfaces de usuario que implementan algoritmos basados en críticas (Burke, 2002b; Reilly et al., 2005b; Pu et al., 2008). En particular, el botón (<>) significa que el usuario necesita otro producto con un valor distinto en la característica nominal seleccionada. Los botones (+,-) significan que el usuario necesita un producto con un valor mayor o menor en la característica numérica seleccionada. En la parte superior de cada botón se muestra el valor actual de la característica para el producto recomendado.
- d. Implementa a través de botones, los elementos para finalizar una sesión de recomendación. Una sesión puede finalizar porque el usuario encontró el producto de su agrado y presiona el botón *Buy it!* (*Buy Event*), o puede finalizar porque el usuario no desea continuar con la sesión de recomendación presionando el botón *Abandon* (*Abandon Event*).

Cuando el usuario interactúa con la interfaz activa los siguientes eventos en la Capa Cliente: evento de criticar una característica del producto recomendado, por ejemplo cuando un usuario manifiesta "Quiero una teléfono de distinto modelo"; compra de un producto sugerido por el recomendador; o abandono de la sesión de recomendación. Como lo he descrito en la Sección 4.3.1, todos estos eventos son trasladados a las acciones de recomendación que llegan a la Capa del Recomendador, situada en la parte inferior de la Figura 21. En particular, cuando un usuario realiza una crítica, el módulo de recomendación filtra y selecciona el siguiente producto que se recomendará al usuario. Cuando la Capa del Recomendador retorna la acción de recomendación al RO (*Recommendation Action*), se actualiza la visualización

del nuevo producto recomendado. Es decir, se muestra la nueva imagen y todos los valores de las características del producto.

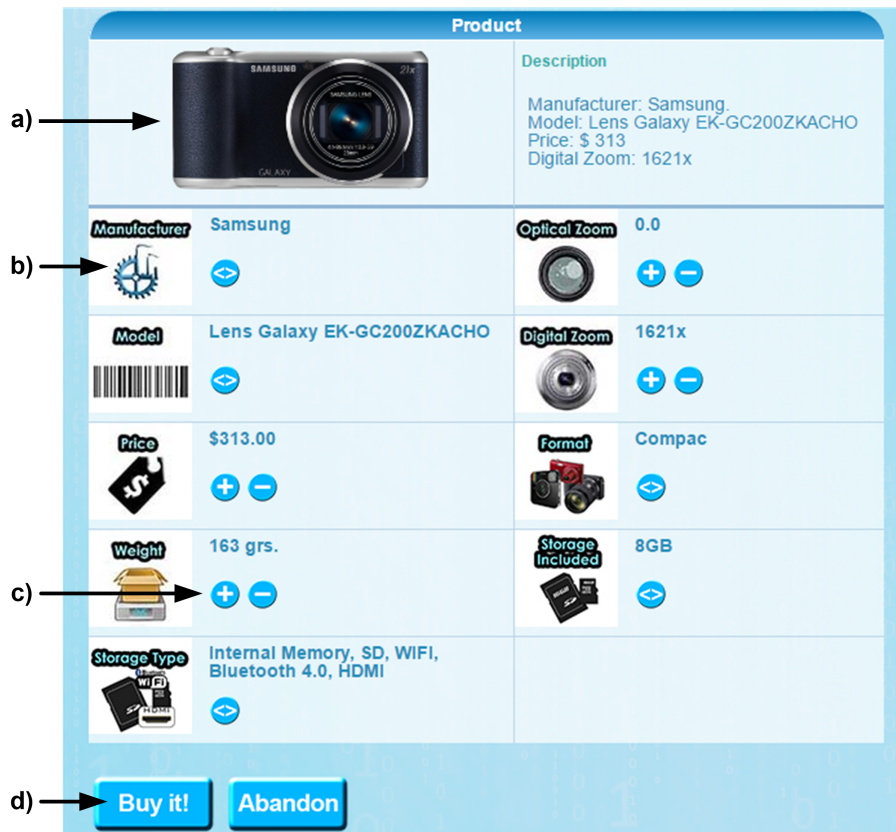


Figura 22: Interfaz de usuario inteligente para el objeto de recomendación en e-WebCo

#### 4.4 E-3DVIRTUALCO: RECOMENDADOR VIRTUAL COLABORATIVO EN TRES DIMENSIONES

En esta sección propongo un *framework* denominado **e-3DVirtualCo**, el cual permite integrar técnicas de recomendación basadas en información colaborativa *on-line*. Este *framework* permite el uso de distintos dominios de productos y ofrece un entorno inmersivo en tres dimensiones que permite la interacción y colaboración in situ entre los usuarios conectados al entorno de visualización. Además en este entorno, a diferencia de la interfaces de recomendación en entornos de visualización en 2D, varios usuarios pueden estar conectados en el mismo momento, lo cual facilita la interacción, comunicación y eventuales colaboraciones entre ellos. Lo anterior, se produce ya que en este entorno los usuarios pueden percibir la presencia de otros usuarios, pueden visualizar lo que hacen los otros usuarios e incluso pueden interactuar con los otros usuarios por medio de los mecanismos de comunicación que ofrece el entorno en tres dimensiones (tales como chat de texto, chat de audio, gestos, etc). Adicionalmente, **e-3DVirtualCo** ofrece una interfaz de usuario inteligente que permite a los usuarios interactuar con el recomendador y con otros usuarios por medio de varios objetos interactivos.

A continuación describo en detalle el *framework* propuesto, para lo cual, en primer lugar explico la arquitectura conceptual que lo sustenta, luego describo los objetos de recomendación que ofrece al usuario la interfaz de usuario inteligente del *framework* y finalmente ejemplifico el proceso de recomendación y los estados por los cuales transitan los usuarios en una sesión de recomendación.

#### 4.4.1 Arquitectura Conceptual e-3DVirtualCo

Generalmente, la arquitectura de un entorno virtual en 3D está basada en el clásico modelo cliente-servidor. En la Figura 23, presento la arquitectura conceptual del *framework* propuesto, siguiendo este modelo clásico. El *framework* se divide en tres capas: la Capa Cliente (*3D Collaborative Space Client*), la Capa Servidor (*3D Collaborative Space Server*) y la Capa del Recomendador (*Collaborative Recommender Space*).

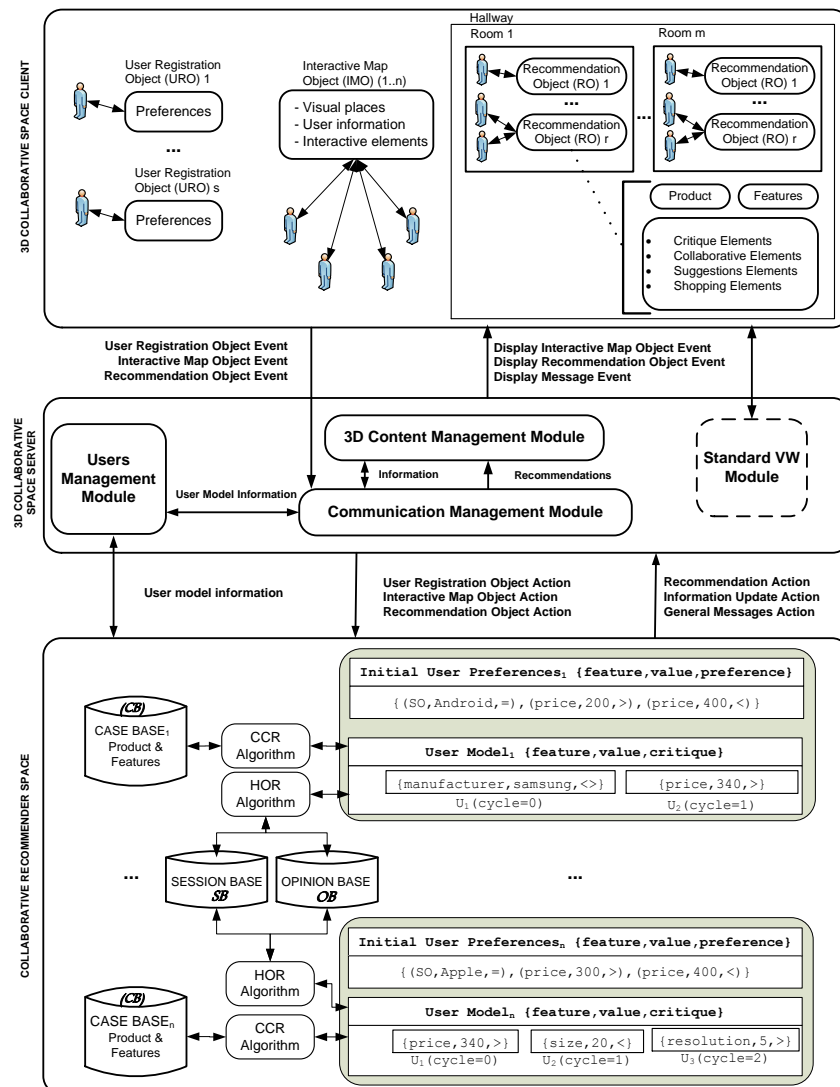


Figura 23: Arquitectura Conceptual del Framework e-3DVirtualCo

#### 4.4.1.1 *Capa Cliente*

El recuadro superior de la Figura 23 muestra la Capa Cliente (*3D Collaborative Space Client*). Esta capa es un espacio virtual en 3D, donde los usuarios pueden sentirse inmersos en un mundo virtual, en el cual pueden interactuar y colaborar con múltiples usuarios a la vez (véase la representación de múltiples usuarios conectados en la Capa Cliente de la Figura 23). En el entorno virtual, los usuarios son representados como avatares y pueden moverse libremente e interactuar con otros usuarios por medio de los mecanismos de comunicación que ofrece el entorno, tales como el chat textual, el chat de audio, o los gestos. Estas características propias de un entorno virtual colaborativo en 3D permiten emular un escenario real de comercio. Adicionalmente, el entorno ofrece a los usuarios todas las funcionalidades que posee un entorno virtual en tres dimensiones, tales como: caminar, correr, volar, tele transportación a otras ubicaciones, visualización en primera persona, vista panorámica del entorno, entre otros.

Cuando el usuario se conecta en la Capa Cliente del *framework*, debe registrarse por medio del objeto denominado *User Registration Object*, URO (véase el lado izquierdo de la Capa Cliente en la Figura 23). Este objeto permite al usuario registrar sus preferencias iniciales, las cuales consisten específicamente en dos o tres características que mejor definen el dominio del producto. Con estas preferencias iniciales, se define un perfil básico del usuario y es posible recomendar al usuario las salas (*rooms*) u objetos de recomendación (RO) en donde los usuarios busquen productos similares a las preferencias ingresadas. En base a las preferencias introducidas por el usuario en el objeto URO, es posible recomendar usuarios o productos afines por medio del objeto denominado *Interactive Map Object*, IMO (véase el centro de la Capa Cliente en la Figura 23). Un objeto IMO entrega al usuario un mapa que muestra todas las salas y los usuarios conectados dentro de una sección de ventas, es decir, los objetos IMO muestran secciones enfocadas a un dominio de productos en particular (por ejemplo, SMARTPHONES). EL IMO dispone de elementos de interacción para que el usuario realice consultas asociadas a los perfiles de los usuarios conectados o productos actualmente recomendados a otros usuarios.

Adicionalmente, el espacio dispuesto en la Capa Cliente, contiene varias secciones de ventas focalizadas en un producto en particular, que denomino *Hallways* (por ejemplo, *hallway* de cámaras fotográficas o *hallway* de smartphones). Para simplificar en la Figura 23 solo muestro un *hallway*, pero en el entorno virtual pueden existir muchos *hallways*. Cada *hallway* posee uno o más objetos IMO para orientar a los usuarios dentro de su sección de ventas. El *hallway* está compuesto por un número fijo  $m$  de salas de ventas (nombradas como *rooms* en el lado derecho de la Figura 23). Cada sala contiene un número fijo  $r$  de objetos de recomendación, nombrados como RO (Recommender Object). Un RO (véase el lado derecho de la Capa Cliente en la Figura 23) puede ser representado por un panel de tres dimensiones que muestra información del producto recomendado y sus características. Además, el RO facilita al usuario la interacción con el recomendador a través de los elementos para realizar críticas (*Critique Elements*), de los elementos colaborativos (*Collaborative Elements*), de los elementos que ofrecen sugerencias al usuario (*Suggestions Elements*) y de los elementos de compra (*Shopping Elements*). A continuación describo cada uno de estos elementos.

- Los Elementos para Criticar (*Critique Elements*) permiten a los usuarios realizar las críticas sobre las características del producto recomendado.

- Los Elementos Colaborativos (*Collaborative Elements*) facilitan la colaboración entre los usuarios (por ejemplo, para iniciar o finalizar una colaboración entre usuarios). Por la propia naturaleza multi-usuario de los entornos virtuales en 3D, los usuarios pueden captar la presencia de otros usuarios, y como sucede en salas de ventas reales, ellos pueden ver los productos que son recomendados a otros usuarios. Si un usuario está interesado en el producto que se recomienda a otro usuario, el usuario interesado en el producto puede iniciar la colaboración utilizando los elementos colaborativos (*Collaborative Elements*) en el recomendador (RO) como lo muestra la Figura 23 en su lado derecho.
- Los Elementos de Sugerencias (*Suggestions Elements*) permiten al usuario interactuar con sugerencias que ofrece el recomendador, tanto de productos como de otros usuarios conectados y con los cuales el usuario actual podría eventualmente colaborar.
- Finalmente, los Elementos de Compra (*Shopping Elements*) permiten al usuario comenzar o finalizar una sesión de recomendación, ya sea porque se activa un RO, se compra un producto o se abandona la sesión de recomendación.

#### 4.4.1.2 Capa Servidor

En la parte media de la Figura 23 se encuentra la Capa Servidor (*3D Collaborative Space Server*), la cual se encarga de la comunicación entre el usuario y la Capa del Recomendador (*Collaborative Recommender Space*). Básicamente, la Capa Servidor tiene tres componentes. El primer componente es el Módulo de Administración de Comunicaciones (*Communication Management Module*), el cual relaciona los eventos activados por el usuario en algún objeto de la Capa Cliente (véase la flecha de eventos que va desde la Capa Cliente a la Capa Servidor en la Figura 23) con las acciones que se transmiten a la Capa del Recomendador (véase la flecha de acciones que va desde la Capa Servidor a la Capa del Recomendador en la Figura 23). El segundo componente es el Módulo de Administración de Usuarios (*Users Management Module*), el cual guarda y gestiona la información de identificación de los usuarios, la información del estado de los usuarios, y la información asociada al RO que está usando el usuario. Además, el Módulo de Administración de Usuarios almacena para cada usuario una lista de los usuarios que están colaborando con él. Si el usuario se encuentra colaborando con otros usuarios, y éste realiza una acción, el módulo de administración de usuarios envía un mensaje a la Capa del Recomendador para actualizar el modelo de usuario de cada uno de los usuarios que se encuentran colaborando con él. El tercer componente es el Módulo de Administración de Contenidos (*3D Content Management Module*), el cual es responsable de actualizar la visualización de los objetos presentes en el espacio cliente en tres dimensiones (por ejemplo, la imagen y características del producto recomendado en un RO o la información mostrada en un objeto IMO).

La comunicación entre las capas del *framework*, representa uno de los aspectos importantes durante el proceso de recomendación, ya que en esta comunicación por un lado, se transmite a la Capa del Recomendador la información generada por los eventos activados por los usuarios en los objetos de interacción de la Capa Cliente (URO, IMO o RO). Por otro lado, se comunican desde la Capa Servidor a la Capa Cliente, las acciones generadas por la

Capa del Recomendador. Para clarificar el paso de mensajes entre una capa y otra del *framework*, a continuación describo el protocolo de comunicación utilizado por el Módulo de Administración de Comunicaciones para este paso de mensajes. Esta descripción, la clasifico de acuerdo al objeto de interacción donde se produce el evento: eventos en el objeto de registro de usuarios URO (*User Registration Object Event*), eventos en el objeto de información IMO (*Interactive Map Object Event*) y eventos en el objeto de recomendación RO (*Recommendation Object Event*) (véase los eventos nombrados en la flecha que va desde la Capa Cliente a la Capa Servidor en la Figura 23).

- Eventos en Objetos de Registro de Usuarios (*User Registration Object Event*). Este evento ocurre cuando un usuario se conecta al entorno **3D-VirtualCo** y realiza el registro de sus preferencias iniciales en un objeto URO. Una vez que el usuario realiza este registro, queda habilitado para poder utilizar un recomendador (RO). El evento de registro de usuario se corresponde con una acción de registro definida por la Fórmula 25, la cual contiene: el usuario que se registra en el objeto URO (`userId`), y una lista de las preferencias ingresadas por el usuario (`userPreferences`).

`RegisterAction(userId, userPreferences)` (25)

En la Capa Servidor, el Módulo de Administración de Comunicaciones *Communication Management Module* entrega la información al Módulo de Administración de Usuarios *Users Management Module*, el cual realiza el registro de las preferencias en la base de preferencias iniciales del usuario UIP, definida en la Capa del Recomendador (véase el recuadro nombrado como *Initial User Preferences* en la parte derecha de la Capa del Recomendador en la Figura 23).

- Eventos en Mapas Interactivos (*Interactive Map Event*). Estos eventos ocurren cuando un usuario accede a información de localización, sugerencias de ubicaciones o perfiles de otros usuarios por medio del objeto IMO. A continuación describo los eventos que se producen en el objeto IMO.
  - ★ Eventos de información de ubicaciones y usuarios. Estos eventos se corresponden con una acción de solicitud de información de usuarios conectados y sus ubicaciones dentro del *hallway* donde está ubicado el objeto IMO según la Fórmula 26, la cual contiene: el usuario que solicita la información a través del objeto IMO (`userId`), y la identificación del objeto IMO en el cual el usuario está realizando la consulta (`imoObjectId`).

`requestIMOLocationInformationAction(userId,  
imoObjectId)` (26)

- ★ Eventos de sugerencias de usuarios o productos. Estos eventos se corresponden con una acción de solicitud de sugerencias de productos y/o usuarios afines, según la Fórmula 27, la cual contiene: el usuario que solicita la información a través del objeto IMO (`userId`), las preferencias iniciales del usuario que realiza la

solicitud (`initialPreferences`) y la identificación del objeto IMO en el cual el usuario está realizando la consulta (`imoObjectId`).

```
requestIMOSuggestionAction(userId,
                             initialPreferences, imoObjectId) (27)
```

La solicitud de información de ambos eventos es enviada a la Capa del Recomendador por medio del Módulo de Administración de Comunicaciones en la Capa Servidor. La Capa del Recomendador, será la encargada de generar la información requerida por el usuario, la cual será enviada por medio de la acción de información de ubicaciones de usuarios y la acción de sugerencias de productos o usuarios afines.

- Eventos en objetos de recomendación (*Recommendation Object Event*). Estos eventos ocurren cuando un usuario realiza acciones en un RO. Las acciones de los usuarios que activan estos eventos pueden corresponder a: una crítica del usuario sobre una característica del producto recomendado, una acción de colaboración del usuario cuando se interesa por un producto de otro usuario, la compra de un producto, la acción de abandonar la sesión de recomendación. Adicionalmente, de manera implícita en un RO existen dos eventos que se activan sin intervención del usuario y que corresponden a los eventos para solicitar a la Capa del Recomendador información de usuarios y productos afines que se podrían sugerir, aparte del producto recomendado, al usuario que está interactuando con el RO.

A continuación, describo cada uno de los eventos transmitidos desde la Capa Cliente a la Capa Servidor.

- ★ Evento de crítica (*Critique Event*): el evento de crítica se asocia a una acción de crítica como lo muestra la Fórmula 28, la cual contiene: el usuario que realiza la crítica (`userId`), el RO donde se realizó la crítica (`recObjectId`), el producto recomendado (`productId`), la característica criticada (`featureId`), el tipo de crítica (`<>`, `↑` o `↓`) y el valor actual de la característica criticada (`critiqueValue`).

```
CritiqueAction(userId, recObjectId, productId,
                featureId, typeCritique, critiqueValue) (28)
```

Por ejemplo, `critiqueAction(user1, rec2, SamsungGalaxySIII, price, ↑, 340)` describe que el usuario con nombre `user1`, que está utilizando el objeto de recomendación `rec2`, envía una crítica sobre el actual producto recomendado, Samsung Galaxy SIII, para obtener un nuevo producto que cumpla con la condición que el precio sea más alto que 340. Posteriormente, en la Capa del Recomendador, esta crítica es almacenada en el modelo de usuario (véase la representación de los modelos de usuarios en la parte derecha de la Capa del Recomendador en la Figura 23). Como resultado de recibir el evento, el recomendador se activa, genera una nueva recomendación y envía una acción de recomendación (*Recommendation Action*) de vuelta a la Capa Servidor (véase la



flecha nombrada como *Recommendation Action* y que va desde la Capa del Recomendador a la Capa Servidor).

- \* Evento colaborativo (*Collaborative Event*): los eventos colaborativos se asocian a acciones colaborativas como se describe en la Fórmula 29, la cual contiene la acción de colaboración (*actionId*) que puede ser *Colaborar* o *Dejar de Colaborar*, el usuario que realiza la acción de colaboración (*userId*, que corresponde al usuario colaborador,  $u_g$ ), el RO donde se realiza la acción (*recObjectId*, que corresponde al RO del usuario que después de esta acción asumirá el rol de usuario líder,  $u_h$ ) y el producto actualmente recomendado en el RO del usuario  $u_h$  (*productId*). Por un lado, la acción *Colaborar* la realiza un usuario cuando le interesa el producto de otro usuario y comienza a colaborar con éste. Por otro lado, la acción denominada *Dejar de Colaborar* la realiza el usuario que inició la colaboración, cuando ya no le interesa continuar colaborando con el otro usuario y decide continuar la sesión de recomendación individualmente.

$$\text{CollaborationAction}(\text{actionId}, \text{userId}, \text{recObjectId}, \text{productId}) \quad (29)$$

La acción de colaboración se envía a la Capa del Recomendador para actualizar el estado de los usuarios ( $u_h$  y  $u_g$ ) y actualizar el modelo de usuario de  $u_g$ . Una vez actualizada la información, la Capa del Recomendador genera la acción de Actualización de Información (*Information Update Action*) que se enviará de vuelta a la Capa Cliente a través del Módulo de Administración de Comunicaciones en la Capa Servidor.

- \* Evento de compra (*Buy Event*): el evento de compra se corresponde con la acción de compra descrita en la Fórmula 30, la cual contiene el usuario que realiza la acción (*userId*), el objeto RO donde fue realizada la acción (*recObjectId*) y el producto comprado por el usuario (*productId*).

$$\text{BuyAction}(\text{userId}, \text{recObjectId}, \text{productId}) \quad (30)$$

La acción de compra, libera el objeto de recomendación (RO) utilizado por el usuario, y el usuario puede continuar navegando libremente por el entorno virtual y de igual forma, puede iniciar otra sesión de recomendación en el objeto de recomendación que él desee.

- \* Evento abandonar una sesión de recomendación (*Abandon Event*): el evento abandonar en el proceso de recomendación se corresponde con las acción abandonar descrita en la Fórmula 31, la cual contiene el usuario que realiza la acción (*userId*) y el objeto RO donde fue realizada la acción (*recObjectId*).

$$\text{AbandonAction}(\text{userId}, \text{recObjectId}) \quad (31)$$

La acción de abandonar una sesión de recomendación al igual que la acción de compra de un producto, libera el objeto de recomendación (RO) utilizado por el usuario, y el usuario puede continuar navegando libremente por el entorno virtual y de igual forma, puede iniciar otra sesión de recomendación en el objeto de recomendación que él desee.

- ★ Evento de sugerencias de usuarios o productos. Estos eventos se corresponden con una acción implícita de solicitud de sugerencias de productos y/o usuarios afines según la Fórmula 32. Nótese que este evento no lo genera explícitamente el usuario, sino que es generado automáticamente por el RO. En concreto este evento se activa en cada ciclo de recomendación, cuando la Capa del Recomendador genera una Acción de Recomendación (*Recommendation Action*). Es decir, este evento permite otorgar al usuario sugerencias adicionales al producto recomendado en la Acción de Recomendación. La acción descrita en la Fórmula 32 contiene: el usuario que está interactuando con el objeto RO que activa el evento (*userId*), el modelo de usuario que contiene los productos recomendados al usuario *userModel* en la actual sesión de recomendación y la identificación del objeto RO en el cual el usuario está realizando la sesión de recomendación (*recObjectId*).

$$\text{requestROSuggestionAction}(\text{userId}, \text{userModel}, \text{recObjectId}) \quad (32)$$

A continuación describo las Acciones que se generan en la Capa del Recomendador y que responden a los eventos generados en la Capa Cliente que he descrito anteriormente:

- Acción de información de ubicaciones y usuarios (*Location Information Action*): esta acción se produce en la Capa del Recomendador y es la respuesta a un evento de localización e información de usuarios en un objeto IMO (*Interactive Map Object Event*) en la Capa Cliente. Esta acción está descrita en la Fórmula 33, la cual contiene el objeto IMO sobre el cual se está solicitando la información (*imoObjectId*), el usuario que está consultando información en el objeto IMO (*userId*), una lista con información de los productos actualmente recomendados en el *hallway* donde se ubica el objeto IMO (*productsInformation*) y una lista con información de los usuarios que actualmente están interactuando con un RO en el *hallway* del objeto IMO (*usersInformation*). Posteriormente, la acción de información es enviada a la Capa del Servidor, el cual será el encargado de activar el evento (*Display Interactive Map Object Event*) en la Capa Cliente (véase la flecha que va desde la Capa Servidor a la Capa Cliente en la Figura 23), mediante el Módulo de Administración de Contenidos 3D (*3D Content Management Module*). Con este evento se actualiza la información en el objeto IMO. Específicamente, se mostrará la ubicación de los usuarios, clasificándolos entre aquellos que están conectados en el *hallway* sin interactuar con un objeto de recomendación, los que están interactuando individualmente con un RO y los que están interactuando colaborativamente con un RO. Para cada una de estas clasificaciones, se utilizan distintos colores que permitan al usuario diferenciar entre unos y otros usuarios.

$$\text{IMOInformationAction}(\text{imoObjectId}, \text{userId}, \\ \text{productsInformation}, \text{usersInformation}) \quad (33)$$

- Acción de sugerencias en objetos IMO (*IMO Suggestion Action*): esta acción se produce en la Capa del Recomendador y es la respuesta a un evento de solicitud explícita del usuario respecto a usuarios o productos afines dentro del *hallway* (*Interactive Map Object Event*). En primer lugar, los usuarios afines corresponden a aquellos usuarios ( $us_q$ ) que tienen preferencias iniciales ( $pr_q$ ) similares a las preferencias iniciales ( $pr_s$ ) del usuario que realiza la solicitud de sugerencias ( $us_i$ ). Para el cálculo de la similitud aplico la sumatoria de las distancias  $d(us_i^{pr_s}, us_q^{pr_s})$  entre cada una de las preferencias del usuario actual ( $us_i$ ) y el candidato a ser usuario afín ( $us_q$ ) según la Fórmula 34. En segundo lugar, los productos afines también son evaluados de acuerdo a las preferencias iniciales del usuario que realiza la solicitud de información. En concreto, se evalúa la similitud entre las preferencias iniciales del usuario y las características de los productos que actualmente se están recomendando a los usuarios que se encuentran interactuando con un RO. La acción de sugerencias en objetos IMO, está descrita en la Fórmula 35, la cual contiene el usuario que hizo la solicitud de información en el objeto IMO ( $userId$ ), el objeto IMO sobre el cual se está realizando la acción ( $imoObjectId$ ), una lista de usuarios afines obtenida de acuerdo al cálculo de la similitud entre las preferencias iniciales del usuario y las preferencias de los usuarios que se encuentran dentro del *hallway* donde está ubicado el objeto IMO ( $affineUsers$ ) y una lista de productos afines obtenida de acuerdo al cálculo de similitud entre las preferencias iniciales del usuario y los productos actualmente recomendados a los usuarios que se encuentra dentro del *hallway* donde está ubicado el objeto IMO ( $affineProducts$ ). Posteriormente, el evento de sugerencias es enviado al Módulo de Administración de Contenidos 3D, el cual será el encargado de actualizar las listas de productos afines y usuarios afines en el objeto IMO, mediante el evento *Display Interactive Map Object Event*.

$$\text{Sim}(us_i, us_q) = \sum_{s=1}^{|IUP|} \cdot d(us_i^{pr_s}, us_q^{pr_s}) \quad (34)$$

$$\text{IMOSuggestionAction}(\text{userId}, \text{imoObjectId}, \\ \text{affineUsers}, \text{affineProducts}) \quad (35)$$

- Acción de recomendación (*Recommendation Action*): esta acción se produce en la Capa del Recomendador y es la respuesta al evento de recomendación en la Capa Cliente (*Recommendation Object Event*). La acción de recomendación está descrita en la Fórmula 36, la cual contiene el usuario que realizó la crítica que produce la recomendación ( $userId$ ), el nuevo producto recomendado ( $productId$ ) y una lista de características del producto recomendado ( $featureValues$ ). Posteriormente,

la acción de recomendación es enviada a la Capa Cliente mediante el Módulo de Administración de Contenidos 3D (*3D Content Management Module*) que se encuentra en la Capa Servidor. Este módulo será el encargado de activar el evento *Display Recommendation Object Event* para actualizar la visualización del producto en el correspondiente objeto RO en la Capa Cliente. Específicamente, se mostrará la imagen y los valores de cada característica del nuevo producto recomendado.

RecommendationAction(userId, productId, featureValues) (36)

- Acción de sugerencias en objetos RO (*RO Suggestion Actions*): este evento se produce en la Capa del Recomendador y es la respuesta a un evento de solicitud implícita de información de usuarios o productos afines que se activa en un RO (*Recommendation Object Event*). Esta acción está descrita en la Fórmula 37, la cual contiene el usuario que hizo la solicitud de información en el objeto RO (userId), el objeto de recomendación en el cual el usuario está interactuando (recObjectId), una lista de usuarios afines (affineUsers) que se obtiene por medio del cálculo de la similitud entre el modelo de usuario de userId y el modelo de usuario del resto de usuarios que se encuentran realizando una sesión de recomendación y una lista de productos afines (affineProducts) que se obtiene por medio del algoritmo HOR que se integra en el *framework e-3DVirtualCo*, el cual hace uso de información colaborativa *off-line* proveniente de la base de datos de sesiones históricas (SB) y de la base de datos de opiniones de otros usuarios (OB). Es decir, los productos afines se obtienen evaluando las sesiones históricas similares a las sesiones de recomendación del usuario actual y además teniendo en cuenta las opiniones de otros usuarios respecto a los productos disponibles para recomendar (véase la descripción de la técnica HOR en la Sección 3.6). Posteriormente, esta acción de sugerencias es enviada al Módulo de Administración de Contenidos 3D en la Capa Servidor, el cual será el encargado de activar el evento *Display Recommendation Object Event* para actualizar las listas de productos afines y usuarios afines en el objeto RO.

ROSuggestionAction(userId, affineUsers, affineProducts) (37)

- La acción de envío de mensajes (*General Messages Action*) se transforma dentro del Módulo de Administración de Comunicaciones, al evento que sirve para mostrar cualquier tipo de información al usuario por medio de mensajes que denomino *General Messages Event*. En particular, este tipo de mensajes sirven para entregar información al usuario cuando inicia una sesión de recomendación, cuando finaliza una sesión o en general cualquier tipo de notificación que la Capa del Recomendador deba notificar al usuario en la Capa Cliente. Como se muestra en la Fórmula 38, esta acción contiene el usuario que es destinatario del mensaje, userId y el cuerpo del mensaje, textMessage. Posteriormente, desde el Módulo de Administración de Comunicaciones, el evento es enviado al Módulo de Administración de Contenidos en tres dimensiones, el cual se encarga de mostrar el mensaje al usuario en el objeto de interacción URO, IMO o RO, por medio del evento *Display Messages*

*Event* (véase la flecha que va desde la Capa Servidor a la Capa Cliente en la Figura 23).

MessagesAction(userId, textMessage) (38)

#### 4.4.1.3 Capa del Recomendador

En la parte inferior de la Figura 23, se muestra la Capa del Recomendador (*Collaborative Recommender Space*), la cual está compuesta por el algoritmo de recomendación basado en información colaborativa *on-line* (*Collaborative Conversational Recommender*, CCR), el algoritmo de sugerencias basadas en información colaborativa *off-line* (*History and Opinion Recommender*, HOR), el conjunto de bases de datos de productos, CB (véase la Definición 1 en la Sección 3.2), la bases de datos con información colaborativa *off-line* SB y OB (véase la Definición 2 y 3 en la Sección 3.2), el modelo que almacena las preferencias iniciales del usuario IUP y el modelo de usuario, U (véase la Definición 5 en la Sección 3.2) de cada usuario en el entorno virtual. Para cada usuario que utiliza el recomendador se crea un modelo de usuario y se asigna una copia de la base de datos de productos. El algoritmo CCR es un módulo que ejecuta el proceso de recomendación, controla el acceso a los datos y actualiza el modelo de usuario durante la sesión de recomendación. Es necesario destacar, que el *framework* propuesto posee un diseño modular y escalable que permite el crecimiento en sus tres capas, por ejemplo en la Capa del Recomendador es posible integrar nuevos algoritmos de recomendación sin afectar el resto de las capas. En particular, el algoritmo **HO**CR, que implementa la última técnica que he propuesto en el capítulo anterior, puede ser integrado fácilmente en la Capa del Recomendador sin realizar cambios de estructura en el *framework*.

Para un mejor entendimiento del proceso completo de recomendación, a continuación presento un resumen de los principales pasos que se debieran realizar en un ciclo de recomendación. Para simplificar, me enfocaré principalmente en un proceso de recomendación individual, ya que el proceso de recomendación colaborativo lo explico en detalle en la siguiente sección (véase la Sección 4.4.2).

En primer lugar, cuando un usuario se conecta en el entorno virtual (3D *Collaborative Space Client*), debe registrar sus preferencias iniciales en un objeto URO. Con este paso, el usuario queda habilitado para iniciar una sesión de recomendación en un objeto RO libre. En segundo lugar, el usuario puede utilizar un objeto IMO, perteneciente a un *hallway*, para ubicarse dentro del entorno y acceder a información del resto de usuarios conectados. En el objeto IMO el usuario puede encontrar usuarios o productos afines y trasladarse a la sala donde se encuentre el usuario similar o el producto que le interese. Una vez en el lugar el usuario, puede interactuar individualmente con un RO libre o también cuando lo desee podría colaborar con algún otro usuario. Formalmente, el proceso de recomendación individual comienza cuando el usuario ya se encuentra registrado en el espacio cliente e inicia un objeto de recomendación RO. Cuando el usuario realiza una crítica sobre una característica del producto actualmente recomendado, la crítica se envía a la Capa del Recomendador por medio del Módulo de Administración de Comunicaciones de la Capa Servidor. Una vez que el algoritmo CCR ha recibido la crítica en la Capa del Recomendador, éste actualiza el Modelo de Usuario (U) con las nuevas preferencias del usuario y selecciona desde la base de datos de productos CB, el siguiente producto que se recomendará. Posteriormente, usando el Módulo de Administración de Comunicaciones y el Módulo de

Administración de Contenidos 3D, la Capa del Recomendador envía la nueva recomendación que se visualizará en el Objeto de Recomendación (RO) del usuario que realizó la crítica. Adicionalmente, los usuarios son libres de colaborar en cualquier momento utilizando los Elementos Colaborativos y el proceso de recomendación se lleva a cabo de manera similar.

A continuación describo en detalle el proceso de recomendación incluyendo los estados individuales y colaborativos por los cuales transita un usuario, durante una sesión de recomendación.

#### 4.4.2 Estados en el Proceso de Recomendación

Durante una sesión de recomendación, los usuarios realizan retroalimentaciones de información al recomendador por medio de los elementos interactivos disponibles en los objetos de recomendación (RO). Como consecuencia de esta interacción, los usuarios transitan por diferentes estados durante la sesión de recomendación, como se muestra en la Figura 24. A continuación, describo en detalle cada uno de los estados que componen una sesión de recomendación.

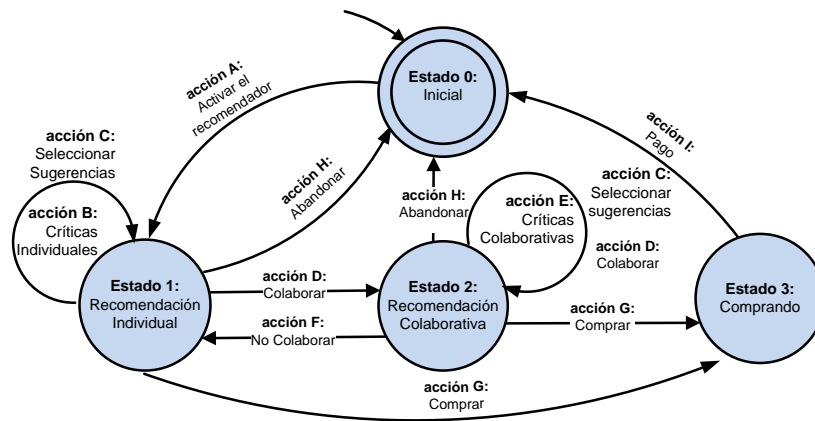


Figura 24: Estados de los usuarios en una sesión de recomendación conversacional colaborativa

**Estado 0:** Inicial. En este estado el usuario se encuentra conectado y registrado en el entorno virtual, es decir, el usuario se encuentra habilitado para interactuar con los objetos disponibles en el espacio cliente 3D (capa *3D Collaborative Space Client*). Adicionalmente, el usuario puede interactuar con el resto de usuarios conectados en el entorno virtual. Cuando el usuario inicia una sesión de recomendación por medio de la **acción A** (activar el recomendador) utilizando un Elemento de Compra en un objeto RO disponible. En este momento el recomendador crea un modelo de usuario (U) para el usuario y el nuevo estado para el usuario será el **Estado 1** (Recomendación Individual).

**Estado 1:** Recomendación Individual. En este estado, el recomendador retorna un producto para ser mostrado en el objeto RO utilizado por el usuario. El usuario puede realizar cuatro acciones en este estado.

1. **Críticas Individuales (acción B).** Esta acción corresponde a las críticas unitarias realizadas por el usuario a través de los Elementos de Críticas disponibles en el RO. Para cada evento de crítica (*Critique Event*), el

recomendador proporciona un nuevo producto recomendado generando una acción de recomendación y el modelo de usuario es actualizado para incluir la nueva crítica. Esta acción mantiene al usuario en el mismo estado (Estado 1).

2. *Seleccionar una Sugerencia (acción C)*. Esta acción ocurre cuando el usuario selecciona una de las sugerencias que el recomendador entrega en los Elementos de Sugerencias del RO. Si el usuario selecciona un producto distinto al recomendado, se actualiza su modelo de usuario y se mantiene en el **Estado 1**. En el caso que el usuario seleccione un usuario sugerido para colaborar, el usuario es trasladado al panel del usuario sugerido sin cambiar su estado, para que pueda optar por colaborar o volver a su panel inicial. En función de la selección del usuario, éste pasará al Estado 2 o continuará en el Estado 1.
3. *Colaborar (acción D)*, es activada por medio de un Elemento Colaborativo en un RO que está siendo utilizado por otro usuario (véase el lado derecho en la Capa Cliente de la Figura 23). Esta acción ocurre cuando un usuario desea colaborar con otro usuario, es decir cuando a un usuario le gusta el producto recomendado y mostrado en el RO de otro usuario (*Collaborative Event*). El usuario que inicia la colaboración lo denominamos usuario colaborador (*guest user*,  $u_g$ ) y el otro usuario lo denominamos usuario líder (*host user*,  $u_h$ ). Después de esta acción, tanto  $u_h$  como  $u_g$  transitan al **Estado 2** (Recomendación Colaborativa).
4. *Comprar (acción G)*. Esta acción ocurre cuando el usuario ha encontrado un producto que le interesa comprar. Esta acción se activa por medio de un Elemento de Compra en un RO (*Buy Event*), como se muestra en el lado derecho de la Capa Cliente en la Figura 23. Después de esta acción, el usuario transita al **Estado 3** (Comprando).
5. *Abandonar (acción H)*. Esta acción es activada por medio de un Elemento de Compra en un RO (*Abandon Event*) y sucede cuando un usuario explícitamente finaliza la sesión de recomendación porque no encontró un producto adecuado. Al realizar esta acción el usuario retorna al **Estado 0** (Inicial) donde posteriormente puede comenzar una nueva recomendación.

**Estado 2:** Recomendación Colaborativa. Un usuario transita a este estado en dos situaciones: cuando el usuario inicia la colaboración (en este caso el usuario asume el rol de usuario colaborador,  $u_g$ ) o cuando otro usuario quiere colaborar con él (en este caso el usuario asume el rol de líder,  $u_h$ ). En un estado colaborativo el usuario líder es el único que puede interactuar con el objeto RO. Lo anterior, se sustenta en el hecho que si los usuarios colaboradores realizaran críticas sin el consentimiento del usuario líder, estas críticas podrían corromper el modelo de usuario del líder. Asumo por tanto, que si un usuario está interesado en el producto que actualmente está siendo recomendado a otro usuario y se asocia a dicho usuario en una sesión de recomendación colaborativa deberá asumir las críticas que realiza el usuario líder. Sin desmedro de lo anterior, el *framework* y el entorno virtual colaborativo ofrecen los mecanismos de comunicación necesarios para que los usuarios puedan llegar a un consenso respecto a las críticas que se efectuarán (por ejemplo, utilizando el chat textual o el chat de voz).

Si un usuario colaborador,  $u_g$ , no está de acuerdo con la siguiente crítica que se realizará, el usuario puede dejar el estado colaborativo usando

el Elemento Colaborativo denominado *Dejar de Colaborar* cuando lo estime conveniente y así volverá a **Estado 1**.

En el **Estado 2** se pueden realizar las siguientes cuatro acciones:

1. *Seleccionar una Sugerencia (acción C)*. Esta acción ocurre cuando el usuario líder  $u_h$  selecciona una de las sugerencias que el recomendador entrega en los Elementos de Sugerencias del RO. Si el usuario  $u_h$  selecciona un producto distinto al recomendado, se actualiza su modelo de usuario y el de sus colaboradores ( $u_g$ ). Esta acción mantiene al usuario  $u_h$  y los colaboradores  $u_g$  en el **Estado 2**. En el caso que el usuario seleccione un usuario sugerido para colaborar,  $u_h$  es trasladado (tele transportado) al panel del usuario sugerido sin cambiar su estado, para que pueda optar por colaborar o volver al panel que estaba utilizando antes de ser trasladado.
2. *Colaborar (acción D)*, es activada por medio de un Elemento Colaborativo en un RO que está siendo utilizado por otro usuario (véase el lado derecho en la Capa Cliente de la Figura 23). En este estado, la acción ocurre cuando el usuario líder  $u_h$  desea colaborar con otro usuario que denominaré  $u_{h_{new}}$ . En este caso, los usuarios colaboradores de  $u_h$  pasan a ser colaboradores del nuevo usuario líder  $u_{h_{new}}$  y el usuario que inicia la colaboración ( $u_h$ ) pasa a ser un usuario colaborador  $u_g$ , del nuevo usuario líder. Después de esta acción, si el nuevo  $u_h$  se encontraba en una recomendación individual (**Estado 1**), éste transita al **Estado 2** (Recomendación Colaborativa). El resto de usuarios  $u_g$  se mantienen en el **Estado 2**.
3. *Críticas Colaborativas (acción E)*. Esta acción es realizada por el usuario líder. Específicamente, el usuario líder realiza la crítica acordada según consenso con el resto de los usuarios colaboradores. Como resultado de esta acción el usuario líder y todos los usuarios que están colaborando con él, actualizan sus modelos de usuarios y reciben un nuevo producto recomendado, el cual se muestra en el correspondiente RO. La actualización del modelo de usuario del usuario líder y de los usuarios colaboradores, es comunicada por el Módulo de Administración de Comunicaciones al recomendador de cada uno de los usuarios para que realice la actualización del correspondiente modelo de usuario. Posterior a esta acción, el o los usuarios se mantienen en el **Estado 2**.
4. *Dejar de Colaborar (acción F)*. En cualquier momento, un usuario colaborador es libre de dejar la colaboración y continuar solo el proceso de búsqueda del producto para comprar. Para esto, el usuario activa el Elemento Colaborativo *Dejar de Colaborar* en el RO del usuario líder con el cual estaba colaborando y retorna al **Estado 1** (Recomendación Individual). Se debe tener en consideración que si el usuario líder no tiene más usuarios colaboradores, él también retorna al **Estado 1**.
5. *Comprar (acción G)*. Esta acción es activada cuando el usuario líder ha encontrado un producto adecuado para comprarlo y finaliza la sesión de recomendación. Al realizar la acción el usuario líder transita al **Estado 3** (Comprando). Si hubieran usuarios colaboradores, cada uno de ellos transitan al **Estado 1** (Recomendación Individual). Si un usuario colaborador encuentra un producto adecuado, primero debe dejar el **Estado 2** de recomendación colaborativa por medio de la **acción E** y posteriormente, en su objeto RO inicial (el cual tiene el mismo producto que el usuario líder), puede realizar la acción de compra.



6. *Abandonar (acción H)*. Esta acción es activada por medio de un Elemento de Compra en un RO. La acción ocurre cuando el usuario líder explícitamente finaliza la sesión de recomendación porque no encuentra un producto adecuado. Después de realizar esta acción, el usuario retorna al **Estado 0** (Inicial), donde el usuario puede iniciar una nueva recomendación. Cuando el usuario líder realiza esta acción, todos los usuarios colaboradores retornan al **Estado 1** (Recomendación Individual).

En el **Estado 2** un usuario líder puede interactuar con muchos usuarios colaboradores. Del mismo modo, el *framework* propuesto permite que un usuario colaborador pueda colaborar con otros usuarios al mismo tiempo en diferentes RO's, sin embargo como sucede en la realidad es muy difícil que el usuario pueda seguir con atención más de dos tareas a la vez.

**Estado 3: Comprando.** En este estado el usuario paga el producto seleccionado. Después de esta acción (**acción I**), el usuario transita al **Estado 0** donde en cualquier momento puede iniciar una nueva sesión de recomendación en un RO libre.

En la siguiente sección describo en profundidad los objetos gráficos interactivos de la interfaz de usuario inteligente, que implementan los objetos conceptuales pertenecientes a la Capa Cliente del *framework*.

#### 4.4.3 Objetos de Interacción en e-3DVirtualCo

En esta sección presento en detalle todos los objetos de interacción de la interfaz virtual en tres dimensiones, que implementan los objetos conceptuales de la Capa Cliente del *framework* (3D *Space Client*). En particular, la interfaz de usuario que describo corresponde al dominio de comercio electrónico<sup>1</sup>, en el cual los usuarios se pueden conectar y navegar libremente dentro de los diferentes espacios habilitados.

Como he mencionado en la Sección 4.4.1, el entorno de comercio electrónico que propongo está distribuido en *hallways*. La Figura 25 muestra un ejemplo de un *hallway* de teléfonos inteligentes (SMARTPHONE). Cada *hallway* está compuesto por salas de ventas y en cada sala existe un número limitado de paneles de recomendación (objetos RO). El usuario se conecta al entorno virtual en tres dimensiones por medio de un usuario y contraseña. En este entorno los usuarios son representados como avatares y pueden moverse libremente e interactuar con otros usuarios por medio de los mecanismos de comunicación que ofrece el entorno virtual, tales como el chat textual, el chat de audio o los gestos.

La interfaz de usuario inteligente implementa dentro del entorno virtual en tres dimensiones los objetos conceptuales descritos en la Sección 4.4.1: URO (*User Registration Object*), IMO (*Interactive Map Object*) y RO (*Recommendation Object*).

Los objetos de interacción propuestos permiten a los usuarios recoger información del sistema de recomendación como también de otros usuarios. Estos objetos guían al usuario en cada fase de su proceso de compra de un producto: cuando el usuario se conecta y se mueve dentro del entorno de comercio electrónico puede utilizar los objetos URO e IMO; y cuando se encuentra en el proceso de búsqueda del producto puede utilizar el objeto RO

<sup>1</sup> Para simplificar, en toda la sección se utilizará un solo tipo de producto (SMARTPHONE). Sin embargo, como se mencionó en la Sección 4.4.1, el *framework* permite utilizar cualquier tipo de producto.

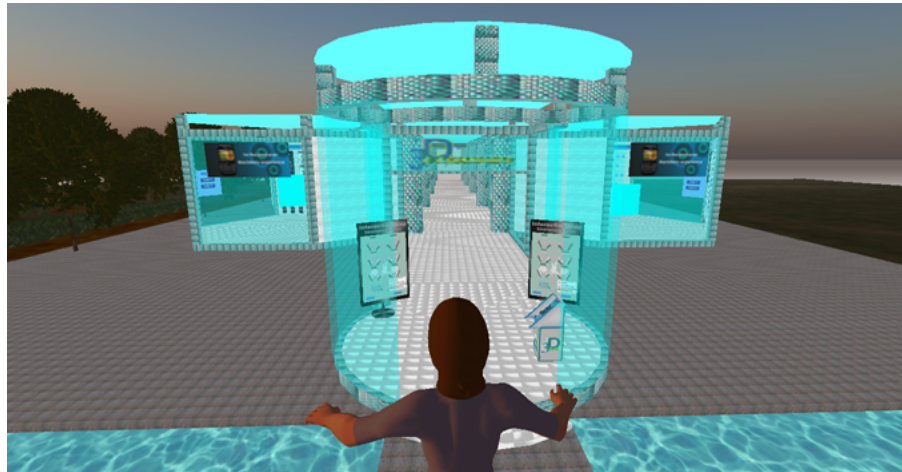


Figura 25: Ejemplo de un *hallway* de teléfonos móviles inteligentes.

y si es necesario también puede utilizar el objeto IMO. A continuación entrego una descripción detallada de los objetos de interacción que se ofrecen al usuario dentro del entorno de tres dimensiones y posteriormente, presento una ejemplificación del proceso de recomendación y de las interacciones de los usuarios que son soportadas por el recomendador dentro de la interfaz propuesta.

El objeto conceptual URO, es representado dentro del entorno virtual de tres dimensiones por una máquina de tipo terminal como se muestra en la Figura 26. Esta máquina implementa la metáfora de la comunicación individual que se realiza entre una persona y un cajero automático, que en este caso permite la comunicación entre el usuario y el recomendador.



Figura 26: Interfaz del objeto URO

Los iconos en la máquina representan las dos preferencias más representativas del usuario respecto al producto que está buscando. Estas preferencias

permiten al recomendador establecer un perfil inicial del usuario (por ejemplo, en la figura se muestran las preferencias del sistema operativo y del precio en un dominio de teléfonos móviles inteligentes). Dependiendo del dominio del producto, los iconos utilizados harán referencia a la característica del producto que se utilizará como preferencia inicial (por ejemplo, en la Figura 26 los iconos asociados al Sistema Operativo del SMARTPHONE SON representados por el logo de cada sistema operativo). Una vez que el usuario selecciona una de las opciones para cada preferencia, ésta se destaca para que el usuario pueda revisar y confirmar su elección.

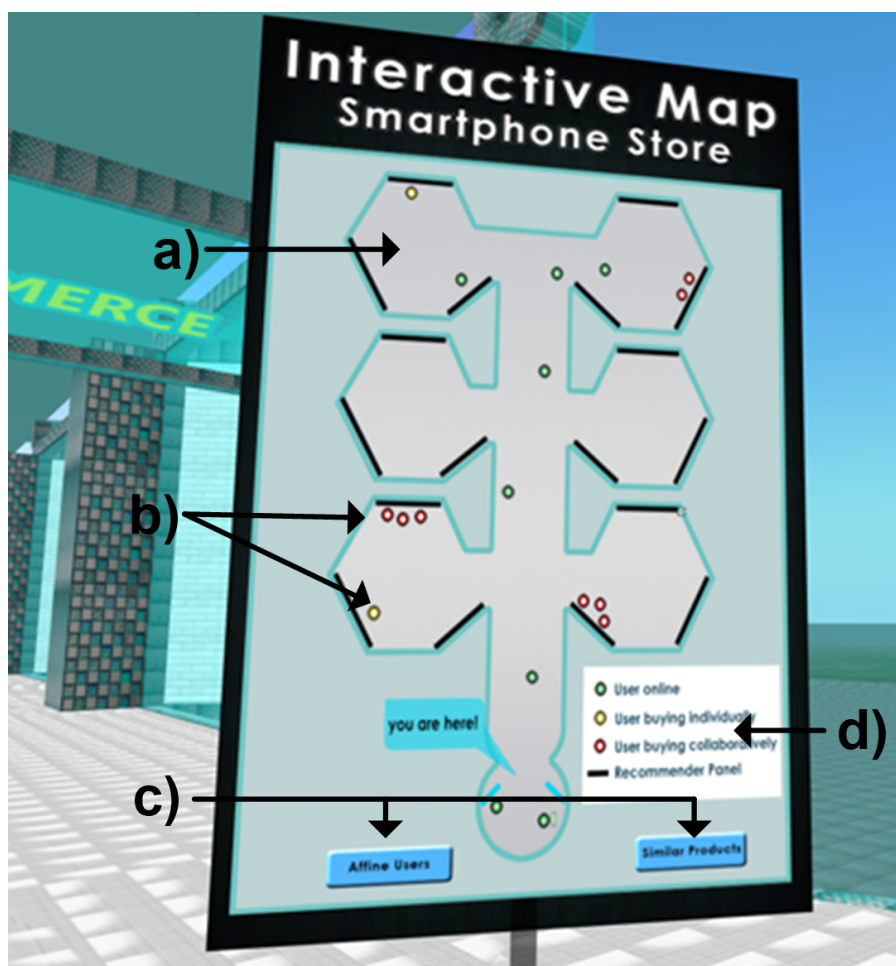


Figura 27: Interfaz del objeto IMO

El objeto conceptual IMO, se representa en el entorno virtual por un panel en tres dimensiones como lo muestra la Figura 27. El objeto IMO, muestra todas las salas del hallway donde se encuentra ubicado y los usuarios que se encuentran actualmente conectados y aquellos que están interactuando con un objeto RO. Este objeto muestra al usuario información de orientación de acuerdo a la numeración descrita en la Figura 27, donde:

- Muestra la ubicación física de las salas dentro del entorno en tres dimensiones. En concreto, la Figura 27 muestra 6 salas (*Rooms*), cada una de ellas con 3 objetos de recomendación (*Recommendation Objects*), entre los cuales algunos se encuentran activos y otros inactivos.
- Muestra donde se encuentran el resto de los usuarios conectados, estableciendo estados para cada uno de ellos, donde los círculos verdes re-

presentan a los usuarios conectados que no han iniciado ninguna sesión de recomendación (véase el **Estado Inicial** en la Figura 24), los círculos amarillos representan a los usuarios que se encuentran conectados e interactuando individualmente con un RO (véase el **Estado Recomendación Individual** en la Figura 24) y un círculo rojo representa a un usuario que se encuentra interactuando con uno o más usuarios de manera colaborativa en un RO (véase el **Estado Recomendación Colaborativa** en la Figura 24). Por ejemplo, en la Figura 27 en b) se pueden observar 3 usuarios colaborando en un RO y otro usuario interactuando individualmente.

- c. Corresponde a los botones que el usuario puede utilizar para recibir sugerencias de parte del recomendador en referencia a las salas donde podría encontrar otros usuarios o productos que sean similares a sus preferencias. Los usuarios pueden obtener información de un usuario en particular presionando el botón denominado "Usuarios Afines" (*Affine Users*). La información de los otros usuarios se muestra en un cuadro emergente de mensaje de texto que entrega además la similitud entre sus preferencias. Si el usuario se interesa por el usuario sugerido, puede seleccionarlo y realizar tele transportación a la ubicación donde se encuentra el usuario similar. Lo mismo puede hacer para productos similares.
- d. Corresponde a la leyenda y ayuda del mapa para el usuario.

El objeto conceptual de recomendación RO, se representa por medio de un panel en tres dimensiones en cada sala, como se muestra en la Figura 28.



Figura 28: Interfaz de los paneles de recomendación en tres dimensiones dentro de una sala

Cada panel de recomendación en 3D, está compuesto por los siguientes elementos visuales (véase la numeración definida en la Figura 29):

- a. Corresponde a iconos visuales que representan a las características del producto. Los iconos visuales facilitan al usuario el uso de la interfaz, tal y como se ha demostrado en estudios previos (Zhang et al., 2008).
- b. Implementa a través de botones, los elementos para realizar críticas denominados *Critique Elements* en la arquitectura conceptual que he descrito en la Sección 4.4.1. Estos botones permiten al usuario realizar las críticas sobre la característica correspondiente a través de uno (<>) o

dos (+,-) botones. El botón (<>) significa que el usuario necesita otro producto con un valor distinto en la característica nominal seleccionada. Los botones (+,-) significan que el usuario necesita un producto con un valor mayor o menor en la característica numérica seleccionada. En la parte superior de cada botón se muestra el valor actual de la característica para el producto recomendado.

- c. Corresponde a una representación gráfica del producto actualmente recomendado al usuario. Por ejemplo, puede ser una imagen, un vídeo o un objeto en tres dimensiones.
- d. Corresponden a botones que implementan los elementos colaborativos (*Collaborative Elements* en la arquitectura conceptual que he descrito en la Sección 4.4.1). Los botones que sirven para colaborar corresponden a los botones *I like* (Acción Colaborar) y *I leave* (Acción Dejar de Colaborar). Si en un panel de recomendación existen varios usuarios colaborando, se mostrará sobre el botón *I like*, el nombre de cada uno de los usuarios colaboradores.
- e. Corresponden a botones que implementan los elementos para realizar la compra denominados *Shopping Elements* en la arquitectura conceptual que he descrito en la Sección 4.4.1. Las acciones de compra que el usuario puede realizar son: activar un panel de recomendación (*Activate*), comprar un producto (*Buy it*) o abandonar la sesión de recomendación (*Abandon*).
- f. Esta sección muestra al usuario actual sugerencias de otros usuarios que tienen preferencias similares a las preferencias iniciales del usuario y con los cuales podría ser beneficioso colaborar (véase la descripción conceptual de la *Acción de Sugerencias en objetos RO* en la Sección 4.4.1.2).
- g. Esta sección muestra al usuario actual sugerencias de productos comprados por otros usuarios con preferencias similares. Estas sugerencias están basadas en información histórica de sesiones de recomendación previas y en opiniones de otros usuarios (véase la descripción de la *Acción de Sugerencias en objetos RO* en la Sección 4.4.1.2 y la descripción de la acción *Seleccionar una Sugerencia* en la definición de los Estados del Proceso de Recomendación en la Sección 3.8.2). Estas sugerencias de productos y las sugerencias de usuarios del ítem anterior, implementan los elementos de sugerencias denominados *Suggestions Elements* en la arquitectura conceptual descrita en la Sección 4.4.1.

El usuario interactúa con el panel de recomendación de acuerdo a las siguientes acciones: criticar una característica del producto recomendado, por ejemplo "Quiero una teléfono de menor precio"; seleccionar (comprar) un producto sugerido por el recomendador; seleccionar la opción de colaborar con otro usuario; o abandonar la sesión de recomendación. Todas estas acciones son trasladadas a los eventos de recomendación que llegan al módulo de recomendadores (*Collaborative Recommender Space* en la Figura 23). Este módulo utiliza el algoritmo de recomendación para seleccionar el siguiente producto que recomendará al usuario. Cuando el módulo de recomendación envía las acciones de recomendación al panel de recomendación, se actualiza la visualización del nuevo producto recomendado. Es decir, se muestra la nueva imagen y todos los valores de las características del producto. Adicionalmente, al momento de realizar una recomendación, el módulo de recomendación también envía las acciones de sugerencias de productos y usuarios que se muestran en el área de sugerencias del panel de recomendación en 3D.



Figura 29: Interfaz de usuario inteligente para el objeto de recomendación (RO)

Cuando una nueva recomendación es enviada desde la Capa Servidor hacia la Capa Cliente, el objeto IMO es también actualizado automáticamente con información del producto actualmente recomendado en cada panel de recomendación.

En la siguiente sección presento un ejemplo de funcionamiento del entorno, poniendo énfasis en los aspectos colaborativos que los usuarios pueden realizar cuando interactúan con el recomendador y con otros usuarios.

#### 4.4.4 Ejemplo de interacción y recomendación colaborativa

En esta sección presento un ejemplo de funcionamiento de la interfaz de usuario en el entorno virtual en tres dimensiones. Para simplificar y mejorar el entendimiento, describo una sesión de colaboración entre 3 usuarios con el objetivo de mostrar en detalle los estados por los cuales transita cada uno de ellos, durante una sesión de recomendación.

Cuando los usuarios se conectan al entorno virtual de comercio electrónico, en primer lugar deben registrar sus preferencias iniciales en el objeto URO. Posteriormente, ellos pueden moverse libremente por todo el espacio virtual disponible (esta situación corresponde al **Estado Inicial** en la Figura 24). Además, en este estado los usuarios pueden obtener información de otros usuarios utilizando los objetos IMO disponibles en la entrada de cada *hallway*. El usuario puede optar por iniciar una sesión de recomendación en un panel de recomendación que se encuentre libre, utilizando el botón *Activate* (véase la letra (e) en la Figura 29) del panel seleccionado.

La Figura 30 muestra una interacción hipotética de tres usuarios que denominamos *Tester 1*, *Tester 2* y *Tester 3*. Estos tres usuarios se encuentran interactuando con tres paneles de recomendación de manera individual (véase el **Estado de Recomendación Individual** en la Figura 24). En este estado, cada

usuario visualiza distintos productos recomendados, tal como se muestra en la Figura 30a, donde cada usuario tiene un producto recomendado diferente en su panel.

En este estado los usuarios pueden continuar realizando críticas de manera individual hasta encontrar el producto que sea de su agrado. Por el contrario, los usuarios también pueden buscar otro usuario que tenga un producto que lo atraiga para iniciar una colaboración. En este sentido, en la Figura 30b se muestra que al usuario denominado *Tester 2* le gusta el producto que se muestra actualmente en el panel de recomendación del usuario denominado *Tester 3*, y por tanto, decide iniciar acciones de colaboración presionando el botón *I like* en el panel de recomendación del usuario *Tester 3*.

Después de realizar esta acción, los dos usuarios cambian al estado Recomendación Colaborativa (**Estado 2 de Recomendación Colaborativa** en la Figura 24), en el cual el usuario *Tester 3* asume el rol de usuario líder,  $u_h$ , y el usuario *Tester 2* asume el rol de usuario colaborador,  $u_g$ . Como se muestra en la Figura 30b, el nombre del usuario colaborador, *Tester 2*, se muestra encima de los botones colaborativos del panel del usuario *Tester 3*. Cuando el usuario *Tester 2* inicia la colaboración con el usuario *Tester 3*, el panel de recomendación del usuario *Tester 2* se deshabilita y no puede ser usado por otros usuarios. En este panel, se continuará mostrando cada uno de los productos que se recomienden en el panel del usuario *Tester 3*. La Figura 30b, además muestra como los usuarios utilizan el chat textual para comunicarse y para decidir la siguiente crítica que realizarán (véase la esquina inferior izquierda de la Figura 30b). Es necesario mencionar que adicionalmente al chat textual, los usuarios pueden utilizar el chat de voz disponible en el entorno virtual. Una vez que los usuarios llegan a un acuerdo respecto a la siguiente crítica (en el caso del ejemplo, ellos acuerdan que necesitan un teléfono móvil más barato),  $u_h$  efectúa la crítica y el nuevo producto recomendado se muestra en ambos paneles de recomendación RO's, tanto en el de  $u_g$  como  $u_h$ .

En la Figura 30c se muestra que el usuario *Tester 3* notifica al usuario *Tester 2* que desea colaborar con el usuario *Tester 1*. En esta situación, el usuario *Tester 2* puede optar por presionar el botón *I leave* para dejar de colaborar con el usuario *Tester 3* y regresar a su panel original cambiando su estado a **Estado de Recomendación Individual**. Sin embargo, en el ejemplo se muestra la situación contraria, donde el usuario *Tester 2* desea continuar colaborando con el usuario *Tester 3* y por tanto, no presiona el botón *I leave*.

Cuando el usuario *Tester 3* presiona el botón *I like* en el panel del usuario *Tester 1*, el usuario *Tester 2* de forma automática se convierte en usuario colaborador ( $u_g$ ) del usuario *Tester 1* y es transportado a la ubicación del panel del usuario *Tester 1*, como se muestra en la Figura 30c. A partir de este momento, el usuario *Tester 1* es el usuario líder ( $u_h$ ) y los usuarios *Tester 2* y *Tester 3* son los usuarios colaboradores ( $u_g$ ). En la misma Figura 30c se muestra que los tres usuarios se comunican por medio del chat textual con el objetivo de alcanzar un consenso sobre la siguiente crítica que aplicarán. Cuando el usuario líder aplica la crítica consensuada en su panel, el nuevo producto recomendado se muestra en los paneles de todos los usuarios colaboradores, sin embargo, solo el panel del usuario *Tester 1* se encuentra activo y habilitado. Adicionalmente, la crítica aplicada actualiza el modelo de usuario del usuario líder y de todos los usuarios colaboradores.

Suponiendo que antes de realizar una nueva crítica, el usuario *Tester 2* presiona el botón *I leave* en el panel del usuario *Tester 1*. La Figura 30d muestra esta situación y el estado en que se encuentran los usuarios una vez realizada



(a) Tres usuarios se encuentran actualmente interactuando con un panel de recomendación de manera individual: Tester 1, Tester 2 y Tester 3



(b) El usuario *Tester 2* se interesa por el producto mostrado en el panel del usuario *Tester 3* y presiona el botón *I like*. En este momento los usuarios *Tester 2* y *Tester 3* están colaborando. Para ello, los usuarios usan mensajes de chat o mensajes de voz para decidir la siguiente crítica que realizarán



(c) El usuario *Tester 3* decide colaborar con el usuario *Tester 1*, teniendo al usuario *Tester 2* como usuario colaborador, u.g. En este momento, todos los usuarios están colaborando y ellos usan mensajes de chat o mensajes de voz para decidir la siguiente crítica que ejecutarán



(d) El usuario *Tester 2* decide presionar el botón *I leave*

Figura 30: Escenas del entorno virtual en 3D (e-3DVirtualCo), que muestra las diferentes interacciones individuales y colaborativas que pueden realizar los usuarios



la acción descrita anteriormente. En la figura se puede observar que el usuario *Tester 2* vuelve a su panel, el cual se activa automáticamente para que el usuario realice las acciones que estime convenientes. En el panel se muestra el último producto recomendado en el panel del usuario *Tester 1* antes que el usuario *Tester 2* presionara el botón *I leave*. La figura además muestra que el panel del usuario *Tester 1* muestra solo un usuario colaborador, el usuario *Tester 3*.

Finalmente, el usuario *Tester 1* y el usuario *Tester 3* continúan colaborando y realizando críticas de manera conjunta (**Estado de Recomendación Colaborativa** en la Figura 24), y por su lado, el usuario *Tester 2* continúa en el estado de recomendación individual (Estado 1 en la Figura 24).

## 4.5 CONCLUSIONES

En este capítulo he presentado los entornos de visualización y las interfaces de usuario inteligentes que implementan las nuevas técnicas de personalización propuestas en esta tesis, describiendo para cada una de ellas su arquitectura conceptual, funcionalidades y objetos interactivos que la componen. Las interfaces propuestas se orientan principalmente al ámbito del comercio electrónico, sin embargo las propuestas permiten el uso de las interfaces en otros dominios, tales como espacios culturales o de diversión.

En primer lugar, he propuesto una primera aproximación de una interfaz de usuario que otorga al usuario una nueva forma de retroalimentación distinta a las tradicionales recomendaciones basadas en críticas de estudios previos. Esta nueva forma de retroalimentación consiste en otorgar al usuario la posibilidad de realizar una selección directa de un producto distinto al recomendado de forma complementaria a las tradicionales críticas de recomendadores conversacionales.

Posteriormente, he propuesto un nuevo *framework* modular y multilingüe basado en un entorno de visualización web en dos dimensiones que he denominado **e-WebCo**. Este *framework* permite la integración de recomendadores tradicionales, tales como IC, sin embargo la principal contribución de este entorno es la integración de nuevas técnicas de recomendación basadas en colaboración *off-line* mediante sesiones históricas y opiniones (HOR, HORi, HGR y HGRi). Es necesario mencionar, que de todas las investigaciones previas orientadas a recomendadores basados en sesiones históricas (tales como EBC, HAC, HGR), ninguna de ellas se ha implementado en interfaces de usuario, ya que todos los experimentos han sido realizados por medio de simuladores. Es decir, el *framework e-WebCo* es la primera interfaz de usuario inteligente que integra este tipo de recomendadores basados en sesiones históricas. Adicionalmente, la modularidad del *framework* posibilita la integración de nuevos algoritmos de recomendación.

Aunque, las interfaces web en dos dimensiones han sido ampliamente utilizadas en recomendadores conversacionales basados en críticas, éstas no disponen de mecanismos que permitan a los usuarios interactuar y colaborar *on-line* entre ellos. Adicionalmente, este tipo de interfaces están limitadas a interacciones individuales de los usuarios que utilizan el recomendador.

Por lo anterior, he propuesto un novedoso *framework* denominado **e-3D-VirtualCo**, el cual está basado en un entorno virtual colaborativo en tres dimensiones. Este *framework* es la primera aproximación en la literatura que aborda un recomendador conversacional en un entorno virtual en tres dimensiones.

Las principales fortalezas del *framework e-3DVirtualCo* son:

- Posee mecanismos de interacción propios del entorno virtual (chat textual, chat de voz, gestos, movimientos, diferentes vistas, etc.) que favorecen la percepción de inmersión del usuario en un entorno virtual y la comunicación con el resto de usuarios conectados en el entorno virtual;
- Dispone de objetos de interacción y recomendación que favorecen la colaboración entre los usuarios y ayudan al usuario durante toda su interacción con el entorno, desde el momento de la conexión y registro del usuario, hasta el momento de la compra del producto o hasta cuando el usuario voluntariamente abandone el entorno de comercio electrónico;
- Permite la integración de algoritmos de recomendación basados en información colaborativa *off-line* y *on-line*;
- Posee un robusto protocolo para eventos y acciones que se producen entre las capas que componen el *framework*.

Resumiendo, mis dos propuestas basadas en entornos de visualización en dos dimensiones (**e-WebCo**) y en tres dimensiones (**e-3DVirtualCo**) representan un avance en entornos de visualización para sistemas de recomendación basados en críticas, ya que cada una de ellas aborda nuevos enfoques que no han sido tratados en investigaciones previas y que abren nuevas líneas de investigación a futuro.



# 5 | EVALUACIÓN

## Resumen

En este capítulo presento la evaluación realizada a las técnicas de personalización propuestas y a las interfaces de usuario inteligentes que integran dichas técnicas de recomendación. En primer lugar, presento los resultados obtenidos del análisis de las nuevas técnicas de recomendación propuestas, la cual fue realizada por medio de una aplicación que simula el comportamiento de los usuarios. Posteriormente, presento los resultados obtenidos por medio de evaluaciones con usuarios reales, de las técnicas de recomendación integradas en las interfaces de usuario inteligentes que propongo en esta tesis. Finalmente entrego las conclusiones del capítulo.

## 5.1 INTRODUCCIÓN

EL OBJETIVO DE ESTE CAPÍTULO es analizar los resultados obtenidos en la evaluación de las nuevas técnicas de recomendación y de las interfaces de usuario inteligentes que integran dichas técnicas. Esta evaluación fue realizada en las siguientes dos fases:

- La primera fase, consistió en evaluar la eficiencia de los algoritmos que implementan cada una de las técnicas de recomendación propuestas en el Capítulo 3. Para ello, he simulado el comportamiento de las personas a través de un simulador<sup>1</sup> que me permite evaluar automáticamente diversos algoritmos a un bajo coste de ejecución.
- La segunda fase, consistió en evaluar la eficiencia, eficacia y usabilidad de las interfaces inteligentes que integran las técnicas de recomendación. Específicamente se han evaluado las interfaces propuestas en el Capítulo 4, las cuales he denominado **e-WebCo** y **e-3DVirtualCo**. Estas evaluaciones las realicé con usuarios reales.

El contenido del capítulo se divide en base a las dos fases mencionadas anteriormente. Para cada una de las fases se presenta: la metodología de evaluación aplicada y el análisis de los resultados obtenidos de la evaluación. Adicionalmente, he aplicado métodos estadísticos para cada una de las evaluaciones.

<sup>1</sup> El simulador consiste en una aplicación que permite ejecutar masivamente sesiones de recomendación que simulan la interacción de los usuarios con el recomendador. Esto permite tener resultados de uso de múltiples algoritmos de recomendación por parte de los usuarios (a los usuarios simulados los denomino "usuarios artificiales"), situación que es difícil de realizar cuando se requiere evaluar una gran cantidad de algoritmos con usuarios reales.

## 5.2 EVALUACIÓN DE LAS TÉCNICAS DE RECOMENDACIÓN A TRAVÉS DE UN SIMULADOR

En esta sección presento la evaluación que he realizado por medio de un simulador a las técnicas propuestas en el Capítulo 3. En primer lugar, describo la metodología utilizada para la evaluación. Posteriormente, presento y analizo los resultados obtenidos para cada una de las técnicas propuestas.

### 5.2.1 Metodología de evaluación

En esta sección describo, en primer lugar, las bases de datos de productos o servicios<sup>2</sup> que he utilizado para las evaluaciones y posteriormente, describo la metodología de simulación que he utilizado para emular las sesiones de recomendación.

#### 5.2.1.1 Bases de Datos

Mis experimentos han sido realizados utilizando cinco dominios de productos, los cuales corresponden a las siguientes bases de datos: PC, CAMERA, TRAVEL, RESTAURANT y SMARTPHONE. Las bases de datos de PC, CAMERA y TRAVEL provienen de la University College Dublin (UCD). Estas bases de datos han sido utilizadas anteriormente en el análisis de algoritmos de recomendación basados en críticas tales como *Incremental Critiquing* (Reilly et al., 2005a), el recomendador *Easy Shop* (Zhang and Pu, 2006) y estudios de otras propuestas de mejoras al proceso de recomendación (Salamó et al., 2005). Por otra parte, la base de datos RESTAURANT ha sido cedida por el Dr. Yasser Salem y se ha usado en estudios previos (McCarthy et al., 2010; Salem and Hong, 2013; Salem et al., 2014). Finalmente, la base de datos de SMARTPHONE fue completamente construida durante el desarrollo de la tesis con información de productos desde la web GSMarena<sup>3</sup>.

En el Cuadro 4 se entrega el detalle de las bases de datos que representan cada uno de los dominios de los productos. En el cuadro se puede observar que existen diversos tamaños de bases de datos, desde 120 productos en la base de datos PC hasta 9945 productos en la base de datos RESTAURANT. Para cada base de datos, se cuenta con los siguientes atributos: el nombre de la base de datos, la cantidad de productos que contiene la base de datos, los atributos nominales y los atributos numéricos que describen el producto. Debo mencionar, que dependiendo de la técnica que se evalúe se utilizan unas u otras bases de datos, ya que no todas cumplen con todos los requisitos de atributos<sup>4</sup> requeridos por cada una de las técnicas de recomendación. Por ejemplo, para las bases de datos PC, CAMERA y TRAVEL no se cuentan con opiniones de usuario en sus atributos, por esto no pueden utilizarse en la evaluación de técnicas que integran información colaborativa (opiniones).

#### 5.2.1.2 Metodología de Simulación

En mis experimentos he simulado el comportamiento de los algoritmos de recomendación por medio de una metodología de evaluación *off-line*. Este tipo de metodología resulta muy apropiada para comparar un amplio rango

<sup>2</sup> Para simplificar, utilizaré el término producto para referirme indistintamente a producto o servicio.

<sup>3</sup> <http://www.gsmarena.com>

<sup>4</sup> En este capítulo emplearé el término atributo para referirme a las características de los productos o servicios presentes en una base de datos.

Cuadro 4: Descripción de las bases de datos

Base de Datos	Productos	Atributos Nominales	Atributos Numéricos
PC	120	Manufacturer, Processor Type, Type	Processor Speed, Monitor, Memory, Drive Capacity, Price
CAMERA	210	Manufacturer, Model, Format, Storage Type, Storage Included	Resolution, Optical Zoom, Digital Zoom, Weight, Price
TRAVEL	1024	HolidayType, Region, Transportation, Season, Accommodation, Hotel	Price, NumberOfPersons, Duration
RESTAURANT	9945	Restaurant Name, Cuisines, Location, After-hours, Bar Scene, Breakfast/Brunch, Buffet, Business, Cheap Eats, Child-friendly, Delivery, Dessert, Dinner, Entertaining clients, Groups, Local cuisine, Lunch Spot, Outdoor seating, Private Dining, Reservations, Romantic, Special Occasion Dining, Street Food, Takeout, View	Price equal or more than, Price equal or less than, Average Overall rate, Recommended Percentages, Atmosphere rate, food rate, Service rate, Value rate, number of reviews, Excellent, Very good, Average, Poor, Terrible
SMARTPHONE	1722	Manufacturer, Model, SO, CPU	Weight, Size, Storage, Ram, Resolution, Price, Design, General Features, Performance

de algoritmos a un coste asequible (Shani and Gunawardana, 2011), a diferencia de los experimentos con usuarios reales, para los cuales es difícil realizar muchas tareas de evaluación con diferentes algoritmos.

Específicamente, he utilizado una metodología denominada *leave-one-out*, la cual ha sido utilizada en varios trabajos previos de *critiquing* (Reilly et al., 2005a; Salamó et al., 2005; Salamó and Escalera, 2012). Esta metodología consiste en seleccionar aleatoriamente un subconjunto de productos desde la **base de datos original** y utilizar cada uno de ellos como un **caso de prueba**. El número de casos de prueba para cada base de datos se describe en la segunda columna del Cuadro 5. Cada **caso de prueba** seleccionado se elimina temporalmente de la **base de datos original** para ser utilizado en dos sentidos.

En primer lugar, el **caso de prueba** sirve como base para generar un conjunto de **consultas iniciales** que simulan las preferencias iniciales del usuario. Cada **consulta inicial** se genera seleccionando aleatoriamente un subconjunto de atributos del producto (véase la columna “consultas iniciales” en el

Cuadro 5). De esta manera, el conjunto de **consultas iniciales** representan las consultas o preferencias iniciales sobre un producto, que provee el usuario la primera vez que interactúa con el recomendador (por ejemplo, una consulta inicial podría ser precio menor que 500 euros y marca distinta de Samsung). Debo mencionar, que en el caso de las **bases de datos originales** de PC y de CAMERA se tienen muy pocos **casos de prueba** (120 en el caso de base de datos PC y 210 en el caso de la base de datos CAMERA), por lo tanto en estas bases de datos, he usado algunos **casos de prueba** más de una vez. Por ejemplo, para la base de datos PC un **caso de prueba** se tuvo que seleccionar más de una vez y de esta forma se generaron dos **consultas iniciales** diferentes con distintas preferencias sobre el producto. En el resto de las bases de datos habían suficientes casos en la **base de datos original**, que permitieron seleccionar 300 **casos de prueba** y generar una **consulta inicial** para cada uno de ellos.

En segundo lugar, el **caso de prueba** se utiliza para seleccionar, desde la **base de datos original** el caso más similar al **caso de prueba**. Este caso representa el **producto objetivo** para los experimentos. Es decir, el producto que el “usuario artificial” del simulador alcanza después de una serie de críticas generadas de forma aleatoria.

Cuadro 5: Parámetros de pruebas en las bases de datos

Base de Datos	Casos de Pruebas	Consultas Iniciales
PC	120	240
CAMERA	210	300
TRAVEL	300	300
SMARTPHONE	300	300
RESTAURANT	300	300

En cada ciclo de recomendación, se generan un conjunto de críticas que son compatibles con el **producto objetivo**. De este conjunto de críticas compatibles, el “usuario artificial” selecciona una de forma aleatoria.

Por lo tanto, después de una serie de ciclos de recomendación se logrará alcanzar el **producto objetivo**. El proceso completo desde que se inicia la interacción del usuario con el recomendador hasta que alcanza el producto objetivo se denomina *sesión de recomendación*.

Para la simulación de las preferencias iniciales del “usuario artificial”, he generado tres tipos de **consultas iniciales** denominadas *hard*, *moderate* y *easy*. Una consulta del tipo *hard*, representa un usuario con muy poco conocimiento del producto que quiere comprar, es decir, una consulta del tipo *hard* solo contiene la preferencia de 1 atributo del producto. Por el contrario, una consulta del tipo *easy* representa un usuario con un alto conocimiento del producto que quiere comprar, en cuyo caso este tipo de consulta contiene las preferencias de 5 atributos del producto. Finalmente, una consulta *easy* contiene las preferencias de 3 del producto. Para clarificar, en el Ejemplo 7 muestro ejemplos de consultas iniciales en dos tipos de bases de datos.

- (7)
- En la base de datos TRAVEL, un ejemplo de consulta del tipo *hard* podría ser [Region = Crete], una consulta inicial del tipo *moderate* podría ser [numberofpersons = 3, transportation = Plane, season = June] y finalmente una consulta de tipo *easy* podría ser [holidaytype = Active, region = Crete, duration = 21, season = June, accomodation = FourStars].
  - En la base de datos PC, un ejemplo de consulta del tipo *hard* podrá ser [Manufacturer = Sony], una consulta del tipo *moderate* podría ser [sizedisplay = 22, storage = 500, procesortype = AMDDuron] y finalmente una consulta de tipo *easy* podría ser [manufacturer = Sony, storage = 500, RAM = 8, price = 800, procesortype = IntelPentium].

Cada **consulta inicial** representa una sesión de recomendación con un **producto objetivo** seleccionado desde los **casos de pruebas**. La ejecución de las sesiones de recomendación las he realizado cien veces por cada **consulta inicial** para obtener un promedio de los ciclos de recomendación necesarios para alcanzar el **producto objetivo**. El promedio de los ciclos de recomendación se conoce habitualmente por su nombre en inglés *Average Session Length*, *ASL*. Con el cálculo del valor medio de los ciclos intento evitar resultados que pueden resultar del azar al utilizar selecciones aleatorias de las críticas que se aplican durante la simulación. La sesión de recomendación simulada termina cuando el **producto objetivo** es recomendado.

En las siguientes secciones detallo la evaluación realizada en cada una de las técnicas propuestas en el Capítulo 3.

### 5.2.2 Resultados de la integración de opiniones en las bases de datos

Las técnicas de recomendación que he propuesto para integrar opiniones en la base de datos, corresponden a las propuestas denominadas *PerTecCB* y *PerIndCB* que he descrito en la secciones 3.5.4 y 3.5.5, respectivamente. En el Cuadro 1 se puede observar un resumen de las mismas.

En esta sección analizo los resultados obtenidos por estas técnicas en los experimentos realizados. En particular, para estos experimentos he utilizado la base de datos SMARTPHONE<sup>5</sup>. En esta base de datos cada producto es descrito por 10 características técnicas y 3 características perceptivas provenientes de opiniones de los usuarios (*Design*, *Performance* y *General features*). Es necesario recordar que una característica técnica es una característica que define algún aspecto físico del producto (por ejemplo, el modelo o el precio de un smartphone). Por el contrario, una característica perceptiva puede estar asociada a más de una característica técnica y generalmente representa una opinión de los usuarios respecto a varias características técnicas (por ejemplo, la característica perceptiva diseño de un smartphone, se puede asociar al tamaño, peso, color, etc.). En particular, en la base de datos SMARTPHONE cada característica perceptiva es el valor promedio del ranking de todas las opiniones realizadas por los usuarios acerca de dicha característica.

En la evaluación analizo el impacto que produce, en la eficiencia de un recomendador tradicional basado en críticas (IC), el hecho de integrar información *off-line* colaborativa proveniente de opiniones de usuarios. En concreto, la Figura 31 muestra los resultados de comparar tres escenarios: (1) el algoritmo IC utilizando la base de datos original que solo contiene características

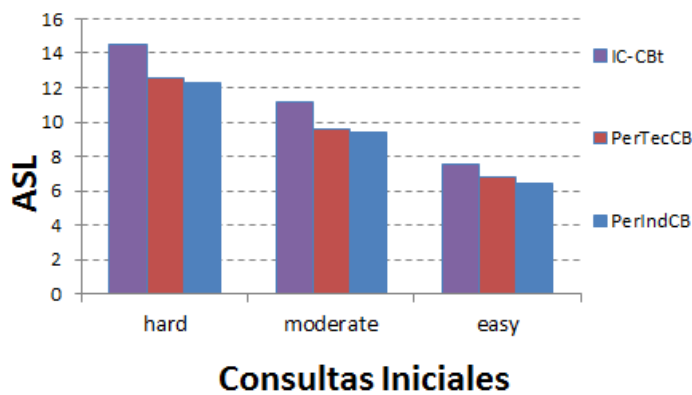
<sup>5</sup> Las bases de datos PC, CAMERA y TRAVEL no contienen opiniones de usuarios, por lo que no fue posible utilizarlas para estas evaluaciones.



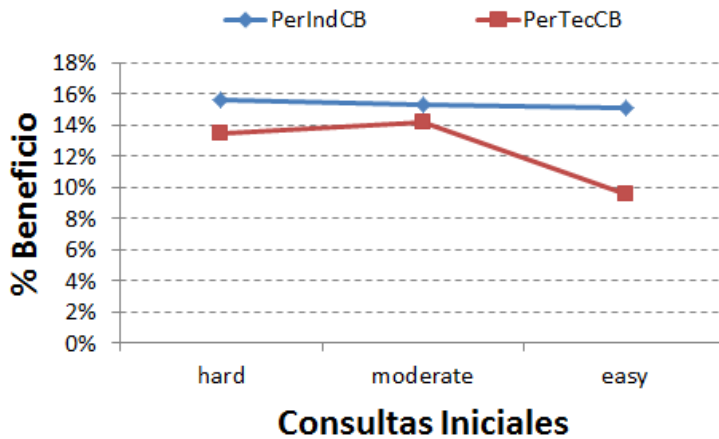
técnicas ( $CB_t$ ), (2) la propuesta *PerTecCB* que utiliza una base de datos de productos compuesta por características técnicas y características perceptivas del producto ( $CB_{pt}$ ) y (3) la propuesta *PerIndCB* que utiliza una base de datos de productos descritos por características perceptivas y tres características independientes ( $CB_{pi}$ ). Para mayores detalles, véase la Sección 3.5.3, en la cual he descrito cada una de estas técnicas.

En particular, en la Figura 31a presento el promedio de ciclos de recomendación (*ASL*) que tarda cada una de las técnicas propuestas en alcanzar el producto objetivo, de acuerdo a los tres tipos de consultas iniciales definidas (*hard*, *moderate* y *easy*).

En esta figura, se puede observar que la cantidad de ciclos disminuye desde las consultas del tipo *hard* hasta las consultas *easy*. Además, para cada tipo de consulta inicial, las técnicas propuestas reducen la cantidad de ciclos, respecto al algoritmo IC utilizando la base de datos original  $CB_t$ . Especí-



(a) ASL para la técnicas IC –  $CB_t$ , *PerTecCB*, *PerIndCB*



(b) Beneficios obtenidos por las técnicas *PerTecCB* y *PerIndCB* con respecto a IC usando  $CB_t$

Figura 31: Resultados de las propuestas *PerTecCB* y *PerIndCB* utilizando algoritmo de recomendación IC

ficamente, la Figura 31a muestra que para consultas *hard* IC usando  $CB_t$  alcanza 14.5 ciclos, mientras que la propuesta *PerTecCB* alcanza 12.6 ciclos y la propuesta *PerIndCB* reduce los ciclos a 12.3. En el caso de consultas

*moderate* IC – CB<sub>t</sub> alcanza 11.1 ciclos, mientras que la propuesta *PerTecCB* alcanza 9.6 ciclos y la propuesta *PerIndCB* reduce los ciclos a 9.4. Finalmente, para las consultas *easy* IC – CB<sub>t</sub> alcanza 7.5 ciclos, mientras que la propuesta *PerTecCB* alcanza 6.8 ciclos y la propuesta *PerIndCB* reduce los ciclos a 6.4.

En la Figura 31b presento el beneficio de usar cada una de las técnicas propuestas (*PerTecCB* y *PerIndCB*), con respecto a usar IC con la base de datos original CB<sub>t</sub>. El cálculo del porcentaje de beneficio lo realizo de acuerdo la Fórmula 39, donde  $x$  e  $y$  corresponden a la medida *ASL* de las propuestas *PerTecCB* y *PerIndCB*, en comparación con IC usando la base de datos CB<sub>t</sub>.

$$\text{Beneficio}(y, x) = \left(1 - \frac{y}{x}\right) \cdot 100 \quad (39)$$

La Figura 31b muestra además que ambas propuestas, *PerTecCB* y *PerIndCB* incrementan el beneficio sobre IC usando la base de datos original CB<sub>t</sub>. En concreto, la nuevas representaciones de bases de datos de productos permiten una reducción en la longitud de las sesiones de recomendación. En concreto, *PerIndCB* obtiene los mejores resultados, con beneficios que van desde un 15.1 % a un 15.6 % (véase la línea azul con rombos en la Figura 31b). En el caso de la propuesta *PerTecCB* se obtienen beneficios que van desde un 9.5 % a un 14.2 % (véase la línea roja con cuadrados en la Figura 31b).

Los resultados obtenidos en la evaluación de las técnicas *PerTecCB* y *PerIndCB*, demuestran que la incorporación de opiniones de usuarios en las bases de datos de productos, mejora la eficiencia de los algoritmos de recomendación basados en críticas. Aunque los resultados obtenidos por las técnicas propuestas han sido bastante satisfactorios, al aplicar el test estadístico ANOVA (Fisher, 1956; Sheskin, 2003) he obtenido un valor de  $p = 0.04713 < 0.05$ , lo cual significa que las diferencias con el algoritmo IC usando la base de datos original CB<sub>t</sub>, son estadísticamente significativas.

Resumiendo, los mejores beneficios se obtienen con la propuesta de base de datos *PerIndCB* que corresponde a la unión de las características perceptivas (opiniones *off-line*) y tres características independientes (véase la definición de la técnica en la Sección 3.5.5). Aunque, esta propuesta define una base de datos que contiene mayor cantidad de características que la base de datos original, hecho que podría llevar a pensar que los ciclos de recomendación aumentarían, se produce el efecto contrario y se reduce la cantidad de ciclos con respecto a una técnica IC utilizando una base de datos que contiene una menor cantidad de características (CB<sub>t</sub> solo contiene las características técnicas del producto).

### 5.2.3 Resultados de integración de opiniones en proceso de recomendación

En esta sección analizo los resultados que produce la integración de opiniones dentro del proceso de recomendación basado en críticas. En concreto, analizo los resultados obtenidos por los algoritmos que implementan las técnicas *LWOpinion* y *UserPerWeight*, las cuales he descrito en la Sección 3.5.7 y Sección 3.5.8, respectivamente. La evaluación contempla la comparación de los algoritmos propuestos con respecto al algoritmo IC. Dado que el objetivo es analizar el impacto en el proceso de recomendación, para estos experimentos he utilizado la base de datos original CB<sub>t</sub> de SMARTPHONE que contiene sólo características técnicas.

En primer lugar, considerando que la técnica *LWOpinion* depende del valor que se asigne a la constante  $\gamma$  (véase la Fórmula 12), a continuación analizo el comportamiento de la técnica para distintos valores de la constante  $\gamma$ . El

valor más representativo de  $\gamma$  será utilizado en el resto de los experimentos de esta sección.

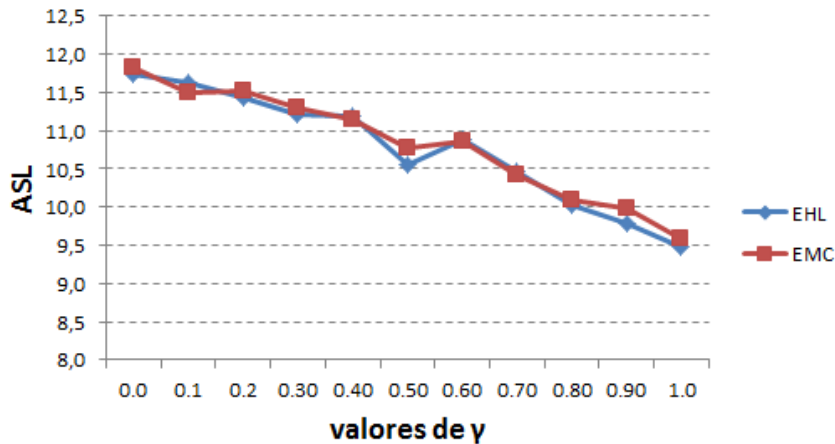


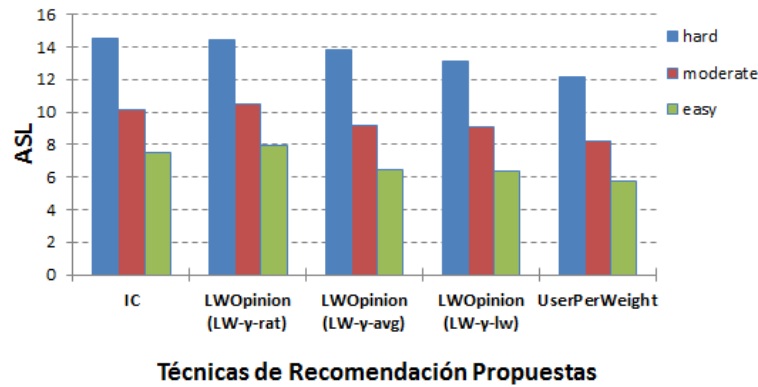
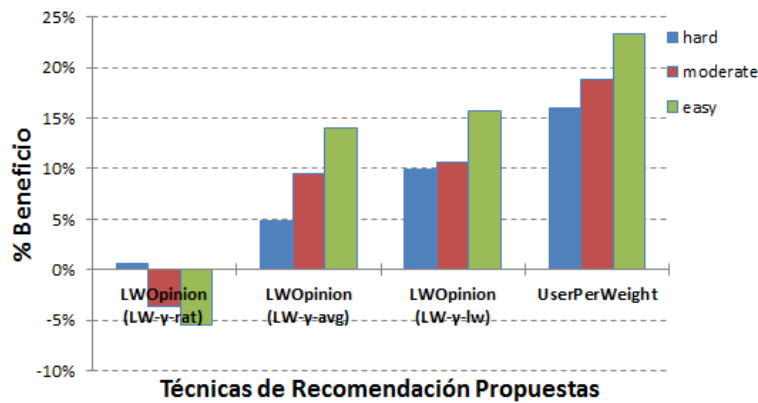
Figura 32: Evolución de la medida ASL para diferentes valores de  $\gamma$

La Figura 32 muestra la cantidad de ciclos de recomendación (ASL) para cada valor de la constante  $\gamma$  (desde un valor de  $\gamma = 0$  a un valor de  $\gamma = 1$ ). De manera adicional, esta figura muestra el impacto de utilizar un algoritmo que integra técnicas de *Reinforcement Learning* EMC y EHL descritas en la Sección 3.5.6 (véase la Fórmula 8 y la Fórmula 9).

Como se muestra en la Figura 32, los peores resultados se obtienen cuando  $\gamma = 0$ , es decir cuando en la Fórmula 12 sólo se consideran los pesos perceptivos que se extraen de las opiniones de los usuarios (en forma de *ratings*). Estos resultados sugieren que una sesión de recomendación, aparte de la información colaborativa que pueda tener, también necesita priorizar productos de acuerdo a las preferencias del usuario obtenidas durante las interacciones conversacionales que realiza el usuario con el recomendador. Por otro lado, los mejores resultados se obtienen para el valor de constante  $\gamma = 1$ , lo cual es concordante con el algoritmo IC el cual privilegia las preferencias de los usuarios. Por lo anterior, defino tres escenarios de evaluación para la técnica *LWOpinion*:  $LW_{\gamma-rat}$ , donde la constante  $\gamma$  tiene un valor de 0,  $LW_{\gamma-avg}$  que corresponde a un valor de  $\gamma$  igual a 0.5 (media) y  $LW_{\gamma-lw}$  que corresponde a un valor de 1 para la constante  $\gamma$ . Con respecto al tipo de compatibilidad utilizada (EHL o EMC), la Figura 32 muestra que no hay una diferencia mayor, por lo cual he utilizado el promedio entre ambas técnicas en los experimentos de esta sección.

A continuación presento los resultados de la evaluación para la técnica *LWOpinion* en sus tres variantes ( $LW_{\gamma-rat}$ ,  $LW_{\gamma-avg}$ ,  $LW_{\gamma-lw}$ ) y para la técnica *UserPerWeight* (véase la Figura 33).

La Figura 33a muestra el promedio de ciclos que ha realizado cada algoritmo durante los experimentos. En concreto, en la figura se puede observar que el algoritmo IC es un algoritmo que realiza una gran cantidad de ciclos para alcanzar el producto objetivo, obteniendo 14.5 ciclos para consultas *hard*, 10.1 para consultas *moderate* y 7.5 ciclos para consultas *easy*. Solo la técnica  $LW_{\gamma-rat}$  en los tres tipos de consultas iniciales, presenta mayor cantidad de ciclos que IC. El resto de las técnicas obtienen buenos resultados, por ejemplo la técnica  $LW_{\gamma-avg}$  alcanza 13.4 ciclos en *hard*, 9.2 ciclos en *moderate* y 6.5 ciclos en *easy*. En el caso de la técnica  $LW_{\gamma-lw}$  se obtienen 13.1 ciclos en consultas *hard*, 9.1 ciclos en consultas *moderate* y 6.4 ciclos en *easy*. Finalmen-

(a) ASL para el algoritmo IC, la técnica *LWOpinion* y la técnica *UserPerWeight*(b) Beneficio obtenido por las técnicas *LWOpinion* y *UserPerWeight* en comparación con el algoritmo ICFigura 33: Resultados de las técnicas *LWOpinion* y *UserPerWeight*

te, la técnica con mejores resultados es *UserPerWeight*, la cual alcanza 12.2 ciclos en consultas *hard*, 8.2 en consultas *moderate* y 5.8 en consultas *easy*.

La Figura 33b muestra el beneficio promedio de los algoritmos que implementan las técnicas propuestas en comparación con el algoritmo IC. La evaluación se ha realizado considerando los distintos niveles de consultas iniciales. En la figura se puede observar que los mayores beneficios se logran con la técnica *UserPerWeight*, la cual obtiene un rango de beneficios que va desde 16.0% hasta un 23.4%. Los resultados de la técnica *LWOpinion* también son bastante satisfactorios para sus variaciones de  $LW_{\gamma-lw}$  que alcanza un beneficio entre un 10.0% y un 15.7%, y para la variante  $LW_{\gamma-avg}$  que obtiene un beneficio entre 4.8% y 14.0%. Sin embargo, en la variante de *LWOpinion* denominada  $LW_{\gamma-rat}$  los resultados no son satisfactorios para las consultas iniciales *moderate* y las consultas *easy*. Estos resultados indican que durante la sesión de recomendación, el sistema de recomendación debiera aprender acerca de las preferencias del usuario y utilizar dichas preferencias para generar mejores recomendaciones.

Los resultados obtenidos por las técnicas *LWOpinion* y *UserPerWeight* demuestran que es posible reducir las sesiones de recomendación, por medio de la integración de opiniones dentro del proceso de recomendación. Es decir, se puede mejorar la eficiencia de un algoritmo de recomendación basado en críticas, en mi caso el algoritmo IC, modificando el peso o importancia

de las características técnicas de los productos por medio de la integración de opiniones. Adicionalmente, he aplicado el test estadístico ANOVA (Fisher, 1956; Sheskin, 2003) obteniendo un valor de  $p = 0.0416 < 0.05$ , lo cual significa que existen diferencias significativas entre los algoritmos evaluados. Aplicando, el test correctivo de Bonferroni (Bland and Altman, 1995) obtengo que ambas técnicas *LWOpinion* y *UserPerWeight* son significativamente superiores a IC.

#### 5.2.4 Resultados de integración de sesiones históricas y opiniones en el proceso de recomendación

En esta sección presento los resultados de los experimentos que he realizado para evaluar la eficiencia de los algoritmos que implementan las técnicas de recomendación que integran información colaborativa *off-line*, sesiones históricas y opiniones de los usuarios acerca de los productos: algoritmo HOR, algoritmo HORi y algoritmo HGRi. Estas técnicas fueron descritas en detalle en la Sección 3.6.

Específicamente, en la evaluación he comparado la eficiencia de mis propuestas con el algoritmo tradicional basado en críticas IC. Adicionalmente, incorporo dentro de la evaluación el algoritmo basado en sesiones históricas denominado History-Guided Recommender (HGR), con la finalidad de realizar la comparación con un algoritmo que también incorpora sesiones históricas. Por lo tanto, los cinco algoritmos de recomendación evaluados en esta sección corresponden a: (1) algoritmo Incremental Critiquing (IC); (2) algoritmo HGR; (3) Integración de técnicas previas con el algoritmo HGR, obteniendo el nuevo algoritmo HGRi; (4) algoritmo *History and Opinion Recommender* (HOR); y (5) la integración de técnicas previas en el algoritmo HOR, obteniendo el nuevo algoritmo HORi. Para estos experimentos he utilizado las bases de datos SMARTPHONE y RESTAURANT, dado que son las únicas de las que dispongo de información colaborativa *off-line*.

En particular, para la evaluación de las técnicas de recomendación basadas en sesiones históricas, he adoptado una metodología similar a la planteada en trabajos previos, principalmente en la generación de las sesiones históricas de críticas basadas en un comportamiento de usuarios racionales (McCarthy et al., 2010; Salem and Hong, 2013; Salem et al., 2014). Para clarificar, en el Cuadro 6 describo brevemente el proceso de creación de las sesiones históricas. Específicamente, he generado cinco bases de datos de sesiones de diferentes tamaños para cada dominio (véase la columna *Bases de Sesiones* en el Cuadro 7). Adicionalmente, he dividido las dos bases de datos en subconjuntos de la base de datos original, con el objetivo de evaluar los algoritmos de recomendación en espacios de búsqueda de diferentes tamaños (véase la columna *Espacios de Búsqueda* en el Cuadro 7).

En estos experimentos, también utilizo la medida denominada *ASL* para evaluar la eficiencia de los algoritmos y además evalúo el beneficio de los algoritmos con respecto al algoritmo base IC.

En la Figura 34 presento varios gráficos con los resultados del análisis realizado de la eficiencia de cada algoritmo para alcanzar el producto objetivo, teniendo en cuenta los diferentes tamaños de las bases de sesiones históricas y en relación al tamaño de los espacios de búsqueda. En estos gráficos, cada línea muestra el promedio de la medida *ASL* obtenido por cada algoritmo, para las consultas iniciales definidas (*hard*, *moderate* y *easy*). Como se puede observar en los gráficos de la Figura 34, el algoritmo IC muestra los mismos resultados para distintos tamaños de bases de sesiones, ya que este algorit-

**Cuadro 6:** Resumen del proceso de creación de sesiones históricas

El proceso de creación de las sesiones históricas lo he implementado utilizando algoritmos estándar de recomendaciones basados en críticas en el simulador que he explicado en la Sección 5.2.1.2. Específicamente, el proceso consiste en seleccionar aleatoriamente un producto objetivo desde la base de datos de productos disponibles. Con este producto objetivo, automáticamente he creado una consulta inicial con 3 a 5 características aleatoriamente seleccionadas desde el producto objetivo.

El algoritmo de recomendación utiliza la consulta inicial para presentar al “usuario artificial” el primer producto recomendado. Desde esta recomendación, el “usuario artificial” aplica aleatoriamente una de las crítica generadas por el simulador y que son compatibles con el producto objetivo. Este proceso de retroalimentación entre el “usuario artificial” y el algoritmo de recomendación se repite hasta que la sesión de recomendación finaliza. Cada sesión termina cuando se alcanza el producto objetivo. Este proceso se repite de acuerdo a la cantidad de sesiones que se deseen tener en la base de datos de sesiones históricas.

**Cuadro 7:** Configuración de las bases de datos de sesiones históricas para los experimentos

Base de Productos	Bases de Sesiones	Espacios de Búsqueda
SMARTPHONE	500, 1000, 3000, 5000 y 10000	1000, 1300, y 1721
RESTAURANT	500, 1000, 3000, 5000 y 10000	6000, 8000, y 9945

mo no utiliza sesiones históricas de otros usuarios para definir la siguiente recomendación.

En las Figuras 34a, 34c y 34e, muestro los resultados utilizando 1721, 1300 y 1000 smartphones como base de productos, respectivamente. En particular, en la Figura 34a, que corresponde a la base de productos más grande, el algoritmo IC obtiene un *ASL* de 14.3 mientras que HGR, HGRi, HOR y HORi obtienen los mejores resultados cuando se incrementa el tamaño de la base de sesiones históricas (es decir, la base de 10000 sesiones históricas), obteniendo en la base de 10000 sesiones un *ASL* de 10.5, 9.6, 9.2 y 8.6, respectivamente. Adicionalmente, en las Figuras 34c y 34e, donde he utilizado una base de datos de 1300 y 1000 smartphones, los mejores resultados también se obtuvieron en las bases de sesiones históricas más grandes. Por lo anterior, se puede concluir que mientras más grande es la base de sesiones históricas, más opciones tiene el recomendador para encontrar una sesión histórica que sea similar a la sesión actual. Específicamente, en la Figura 34c el algoritmo IC alcanza 13.2 ciclos, mientras que los algoritmos HGR, HGRi, HOR y HORi alcanzan un *ASL* de 10.4, 9.1, 8.9 y 8.7, respectivamente. Finalmente, en la Figura 34e donde se muestran los resultados obtenidos utilizando la base de datos de productos más pequeña, la cual solo contiene 1000 smartphones, la medida de *ASL* de IC obtenida es 12.1 y para el resto de algoritmos es un *ASL* de 8.9 en el caso de HGR, 7.9 para el algoritmo HGRi, 7.8 para el algoritmo HOR y 7.6 en el caso de HORi.

En las Figuras 34b, 34d y 34f muestro los resultados utilizando 9945, 8000, y 6000 RESTAURANTS como base de productos, respectivamente. El espacio de búsqueda en este dominio es más grande que la base de productos de SMARTPHONE (1721). En la base de datos de RESTAURANT, el algoritmo IC

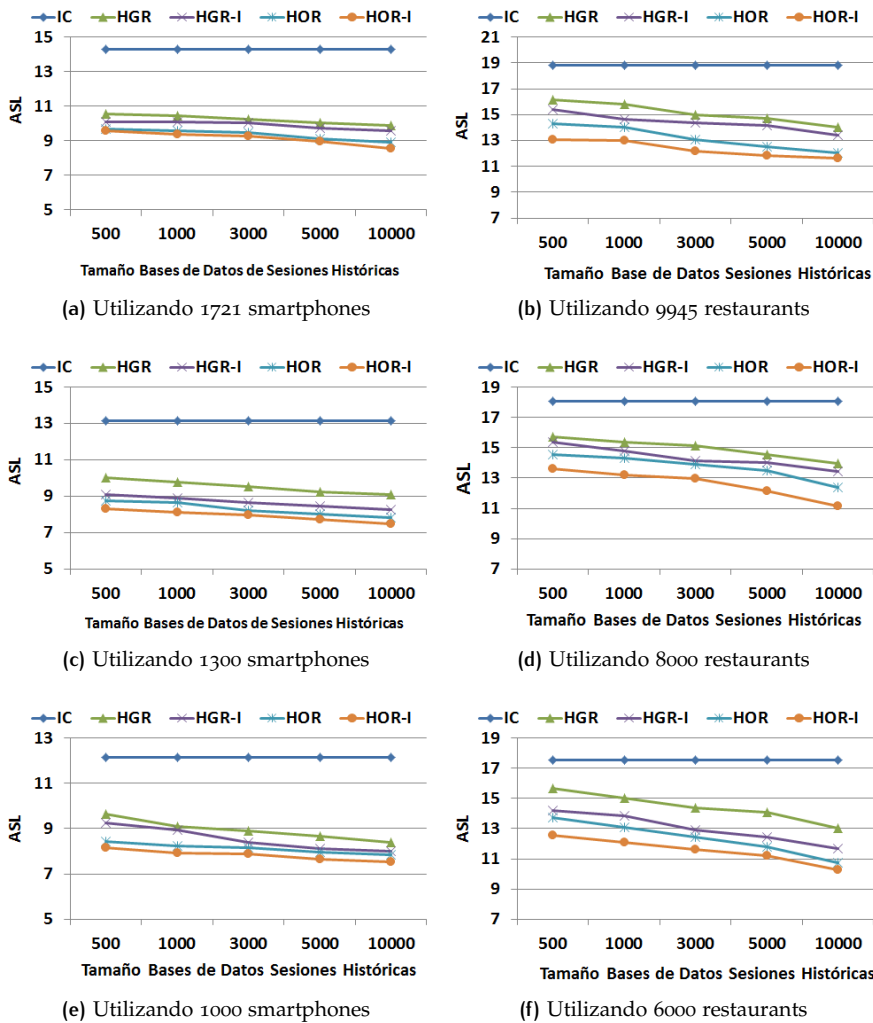


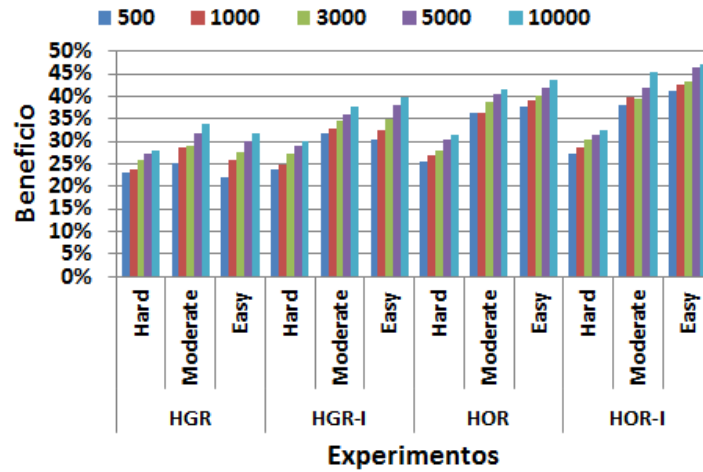
Figura 34: Resultados de ASL de las técnicas propuestas en comparación con las técnicas de IC y HGR en dominios de SMARTPHONE y RESTAURANT

obtiene un ASL de 18.8, 18.1 y 17.6 para cada tamaño de base de productos (9945, 8000 y 6000), respectivamente. En las bases de datos de RESTAURANT, al igual que en SMARTPHONE, los mejores resultados se obtienen para bases de sesiones históricas más grandes (10000 sesiones). Como se puede observar en la Figura 34b, los algoritmos HGR, HGRi, HOR y HORi alcanzan para las bases de 10000 sesiones un promedio de ASL de 14.0, 13.6, 12.1 y 11.7, respectivamente. Estos resultados son mejores que los obtenidos por el algoritmo IC.

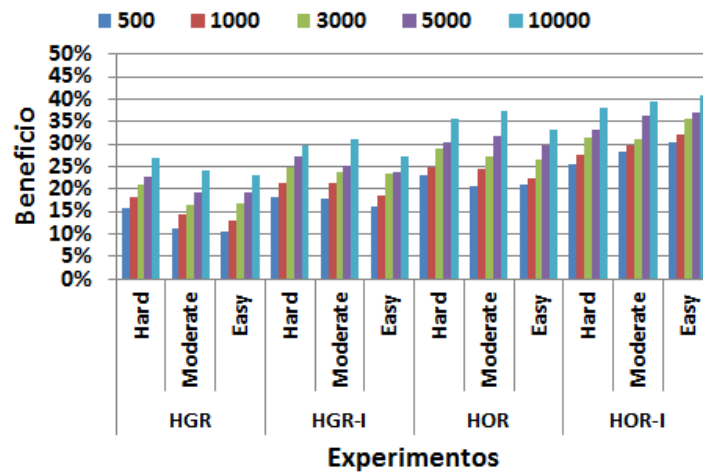
Adicionalmente, en la Figura 34d, se muestran los resultados para la base de datos de productos de tamaño medio (8000 RESTAURANT) donde los algoritmos HGR, HGRi, HOR y HORi obtienen un promedio de ASL de 13.9, 13.4, 12.4 y 11.5, respectivamente. Finalmente, en la Figura 34f, se muestran los resultados asociados a la base de productos más pequeña. En este caso, para la base de 10000 sesiones se obtiene un ASL de 13.0 ciclos para el algoritmo HGR, 11.8 para HGRi, 10.7 para HOR es y 10.7 para HORi.

Analizando los resultados en ambos dominios de productos, se puede observar que la curva de los algoritmos propuestos (es decir, el algoritmo

HOR y los algoritmos mejorados HORi y HGRi) son menos dependientes del tamaño de la base de datos de sesiones históricas, situación contraria a lo que sucede con el algoritmo HGR, el cual es muy dependiente del tamaño de la base de sesiones históricas. Además, todas mis propuestas reducen en gran medida el *ASL* para bases de datos históricas de gran tamaño.



(a) Base de Datos SMARTPHONE



(b) Base de Datos RESTAURANT

Figura 35: Beneficios de los algoritmos propuestos en comparación con el algoritmo IC

A continuación, me centraré en analizar el beneficio que se obtiene al utilizar algoritmos basados en sesiones históricas, en comparación con el algoritmo base IC. La Figura 35 muestra el beneficio promedio de los algoritmos propuestos en comparación con el algoritmo IC, usando diferentes bases de datos de sesiones históricas (500 a 10000 sesiones para ambas bases de datos SMARTPHONE y RESTAURANT) y utilizando los diferentes tipos de consultas iniciales (*hard*, *moderate* y *easy*). Primero que todo, analizando ambos conjuntos de datos, se puede observar que el beneficio depende tanto del tamaño de la base de sesiones (es decir, el beneficio es más grande para bases de sesiones históricas más grandes), como también del tipo de consulta, don-



de en la mayoría de los casos las consultas del tipo *easy* obtienen mayores beneficios.

Analizando cada uno de los algoritmos en detalle, como se puede observar en la Figura 35a, los beneficios más grandes, en comparación con IC, se obtienen por el algoritmo HORi con un beneficio que va desde un 27.3 % a un 47.3 %. Además, HOR obtiene un beneficio en el rango de 25.5 % a 43.5 %, mientras que el beneficio obtenido por HGRi está en el rango de 23.8 % a 39.7 %. Por otro lado, utilizando la base de datos RESTAURANT, se han obtenido resultados similares a la base de datos SMARTPHONE. En la Figura 35b, se puede observar que el algoritmo HOR obtiene buenos resultados cuyo beneficio se encuentra en el rango de 23.2 % a 37.3 %. También es necesario destacar que el beneficio en la eficiencia de recomendación también se mejora cuando he realizado la integración de técnicas previas, como es el caso de HGRi, el cual obtiene un porcentaje de beneficios sobre IC en el rango de 16.2 % a 31.0 %, mientras que el algoritmo original HGR solo obtiene un beneficio entre 10.3 % y 27.0 %. Esta misma situación sucede en el algoritmo HORi, el cual obtiene un beneficio en el rango de 25.4 % a 40.8 % que es mejor al resultado del algoritmo HOR original.

Resumiendo, los resultados muestran que el beneficio en la eficiencia de la recomendación de todas mis propuestas (el algoritmo HOR y los algoritmos que integran técnicas previas, HORi y HGRi) es mayor que el beneficio obtenido por IC y el algoritmo HGR. Esto se demuestra en los dos dominios de evaluación utilizados: SMARTPHONE y RESTAURANT.

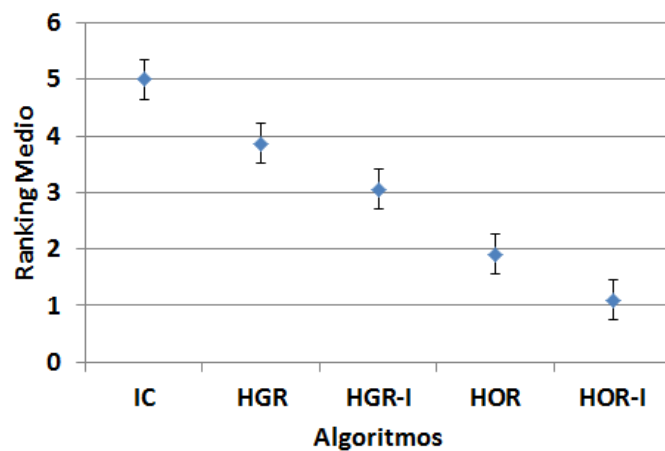
De manera adicional, para demostrar que la eficiencia de mis propuestas es significativamente mejor que el algoritmo base IC, he aplicado el método estadístico denominado test de Friedman<sup>6</sup> y un post-test denominado Nemenyi<sup>7</sup> (Friedman, 1940) en los resultados obtenidos en ambos conjuntos de datos. Estos test, corresponden a procedimientos especializados para evaluar si existen diferencias significativas entre múltiples medidas (Demšar, 2006), de esta forma permiten también controlar problemas que se producen en otros tipos de test cuando se trabaja con múltiples hipótesis (más de 5 o igual a 5), como sucede en particular en el método T-TEST. Por lo anterior, y teniendo en cuenta que en mi caso la evaluación la realizo sobre múltiples algoritmos de recomendación, el uso del test de Friedman y su correspondiente test *post-hoc* Nemenyi es lo más apropiado.

En primer lugar, la evaluación considera  $k = 5$  algoritmos (IC, HGR, HOR, HORi y HGRi) y  $N = 45$  diferentes experimentos para cada test. La cantidad de experimentos corresponde a los tres tamaños diferentes que he usado para cada base de datos, las tres diferentes consultas iniciales (*hard*, *moderate*, *easy*) y los cinco tamaños diferentes de bases de sesiones históricas. De esta manera, se obtienen los 45 experimentos,  $3 \times 3 \times 5 = 45$ . Adicionalmente, he calculado un ranking medio ( $r$ ) para cada algoritmo considerando todos los experimentos. En concreto, para cada experimento he clasificado los algoritmos siguiendo la metodología descrita en (Friedman, 1937; Friedman, 1940). De esta forma, el algoritmo que obtiene el mejor rendimiento (menor cantidad de ciclos o *ASL* menor) es clasificado con el número 1, el segundo mejor algoritmo se clasifica como 2, y así sucesivamente. Posteriormente, se calcula el ranking de cada algoritmo a través del promedio obtenido por éste en todos los experimentos.

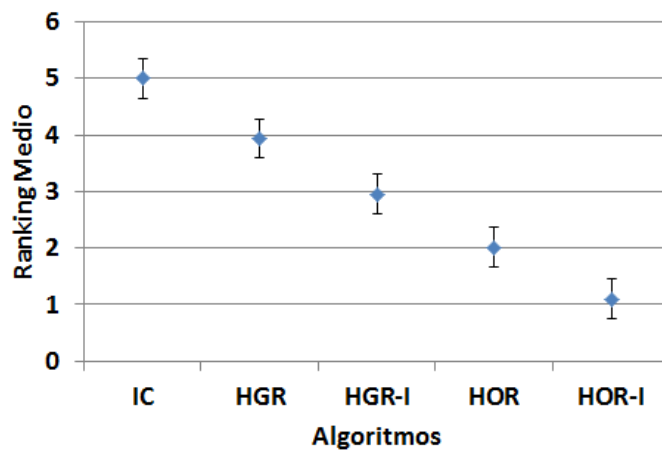
<sup>6</sup> Es un test estadístico no paramétrico, utilizado para detectar diferencias entre experimentos de múltiples algoritmos.

<sup>7</sup> Es usado después de un test estadístico de múltiples algoritmos (por ejemplo, test de Friedman). Es un test de rango *post-hoc* usado para identificar que algoritmos poseen una diferencia significativa respecto a otro, en una comparación de múltiples algoritmos.

A continuación, aplico el test de Friedman y Nemenyi para analizar si la diferencia entre los algoritmos es estadísticamente significativa. En este test,  $F_F$  se distribuye de acuerdo a la distribución F con  $(5 - 1) = 4$  y  $(5 - 1) \cdot (45 - 1) = 176$  grados de libertad. El valor crítico para  $F(4, 176)$  es igual a 2.42, considerando un nivel de criticidad de 0.05, es decir con un nivel de confianza de 95.0%. En mi caso, para la comparación de la eficiencia he obtenido los valores de  $X_F = 145.65$  y  $F_F = 186.58$  para el ranking de eficiencia de los algoritmos. Dado que el valor de  $F_F$  que he obtenido es mayor que el valor teórico 2.42, procedo a rechazar la hipótesis nula y por ende concluyo que existen diferencias estadísticamente significativas entre mis propuestas y el algoritmo base. A continuación, procedo a detectar que algoritmos son significativamente diferentes en su medida de *ASL*. Para esto, utilizo el test de Nemenyi *two-tailed*, en el cual comparo cinco algoritmos con un valor crítico de  $\alpha = 0.05$ . Con este valor crítico obtengo una medida  $q_{0.05} = 2.569$  y un valor de diferencia crítica de  $CD = 0.699$ .



(a) Base de Datos SMARTPHONE



(b) Base de Datos RESTAURANT

Figura 36: Aplicación del test Nemenyi al ranking de *ASL* de los algoritmos evaluados

Los resultados del test de Nemenyi se muestran en las Figuras 36a y 36b, para las bases de datos SMARTPHONE y RESTAURANT, respectivamente. En

estos gráficos, los diamantes representan el ranking medio obtenido por cada algoritmo y las líneas verticales que traspasan los diamantes representan la diferencia crítica, CD. Básicamente, se puede entender que la eficiencia de dos algoritmos es significativamente diferente si sus líneas verticales no se solapan. Por ejemplo, se puede observar en ambos dominios que todos los algoritmos son significativamente mejores que el algoritmo IC. De hecho, el mejor algoritmo (es decir, aquel que tiene un promedio de ranking menor) es el algoritmo HOR<sub>i</sub>, el cual es mejor que el resto. Además, se puede observar que los algoritmos propuestos (HGR<sub>i</sub>, HOR, y HOR<sub>i</sub>) son significativamente mejores que HGR.

En resumen, puedo concluir que: (1) el algoritmo HOR es significativamente mejor que el algoritmo IC y HGR; y (2) la integración de técnicas previas en algoritmos existentes (HGR<sub>i</sub> y HOR<sub>i</sub>), mejoran significativamente la eficiencia respecto a las técnicas previas tradicionales (IC y HGR).

#### 5.2.5 Resultados de algoritmos de recomendaciones conversacionales colaborativos

En esta sección presento los resultados obtenidos en la evaluación del algoritmo de recomendación colaborativo CCR. La evaluación se ha focalizado principalmente en evaluar la eficiencia de la recomendación, por medio de la cantidad de ciclos definidos por la medida *ASL*. Estos experimentos los he realizado utilizando cinco dominios de productos: PC, CAMERA, TRAVEL, SMARTPHONE y RESTAURANT. Con respecto a la simulación del comportamiento colaborativo de los usuarios, he simulado las acciones de colaboración descritas en la Sección 3.8.3 (véase la línea 17 en el Algoritmo 3), por medio de la selección aleatoria de un producto desde un subconjunto de la base de datos de productos para recomendar. Este subconjunto contiene los productos que son más compatibles con las preferencias del usuario (en este caso más compatibles con el producto objetivo). Además, este subconjunto de productos se acota a aquellos productos con una proximidad, al producto objetivo del usuario, entre un 95 % y un 60 %. Este rango de proximidad lo denominé nivel de preferencias. Para clarificar, por un lado un nivel de preferencias de 95 % significa que los productos en el subconjunto serán aquellos que se encuentran dentro del 5 % más próximos al producto objetivo del usuario y se descarta el 95 % restante de los productos más alejados del objetivo. Por otro lado, un nivel de preferencias del 60 % implica que los productos en el subconjunto se encuentran dentro del 40 % más próximo al producto objetivo del usuario, descartando el 60 % restante de productos más alejados del objetivo. Lo anterior, me permite simular el comportamiento de un usuario  $u_g$ , el cual puede seleccionar desde otro usuario  $u_h$ , un producto que se encuentra en el rango de proximidad entre 5 % hasta un 40 % con respecto a su producto objetivo. La idea detrás de la medida que denominé nivel de preferencias, es analizar el nivel máximo de tolerancia para la colaboración que permite el algoritmo colaborativo. Este nivel de tolerancia implica por ejemplo, que un usuario no va a escoger colaborar con alguien que esté un 50 % lejos de sus preferencias.

A continuación presento los resultados obtenidos respecto a la evaluación de la eficiencia en la recomendación y la evaluación del beneficio del algoritmo propuesto con respecto al algoritmo base IC.

La Figura 37 presenta los gráficos asociados a los resultados obtenidos con las distintas bases de datos de productos. En cada gráfico se muestra el *ASL* obtenido por el algoritmo IC y por el algoritmo CCR considerando

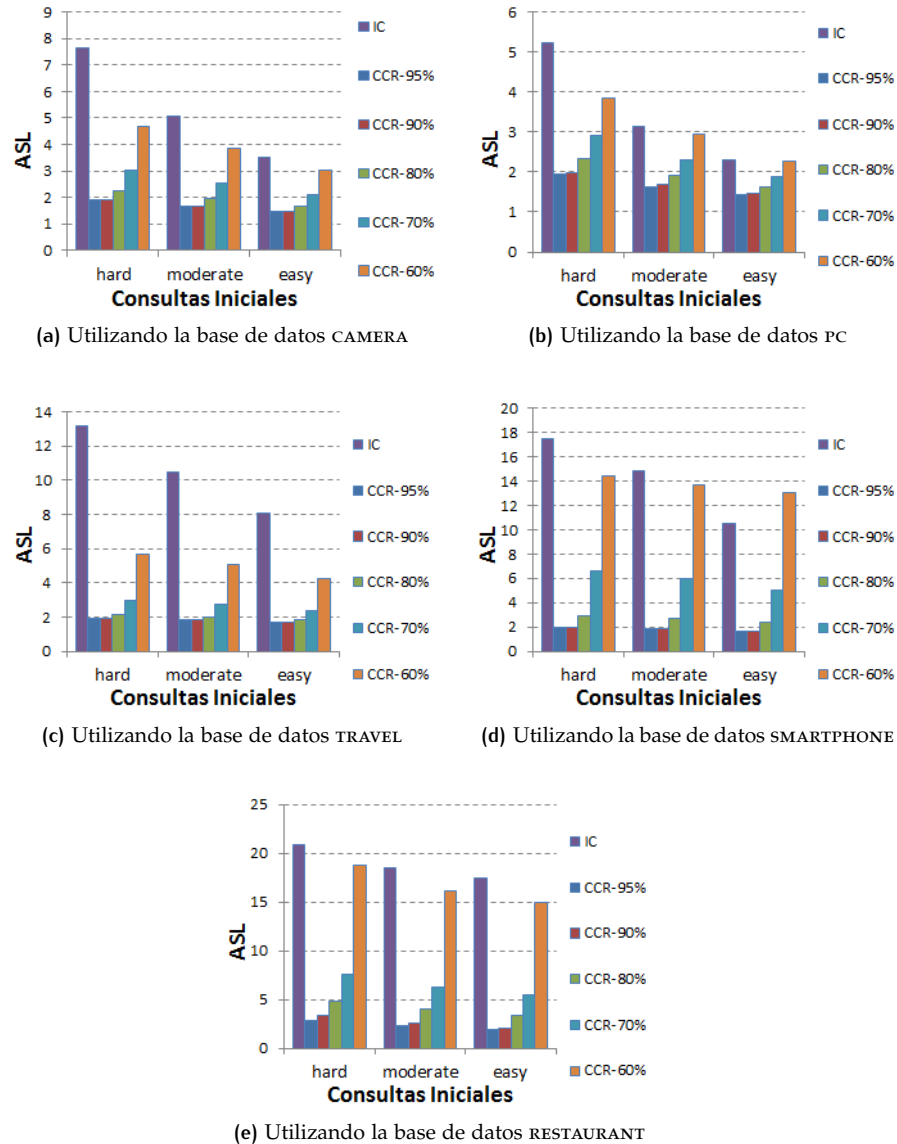


Figura 37: Longitud promedio de la sesión (ASL) para las bases de datos CAMERA, PC, TRAVEL, SMARTPHONE y RESTAURANT, considerando diferentes niveles de dificultad en las consultas iniciales

los diferentes niveles de preferencias (desde 95.0% a un 60.0%) y considerando además diferentes niveles de dificultad en las consultas iniciales (*hard*, *moderate*, y *easy*).

En particular, la Figura 37a, muestra los resultados obtenidos con la base de datos CAMERA, donde el algoritmo IC presenta un ASL de 7.6 ciclos para las consultas del tipo *hard*, 5.1 para consultas *moderate* y 3.5 para consultas *easy*. En el caso del algoritmo CCR con nivel de preferencia del 95.0%, se reducen bastante los ciclos de recomendación obteniendo los ASL 1.9, 1.7 y 1.5 para las consultas *hard*, *moderate* y *easy*, respectivamente. Lo mismo sucede con el resto de niveles de preferencia, donde se obtienen para consultas *hard* una cantidad de ciclos que van desde 1.9 hasta 4.5, considerando niveles de preferencia desde 90.0% hasta un 60.0%, respectivamente. En el caso de las

consultas *moderate* con niveles de preferencia de usuario desde el 90.0% hasta el 60.0%, los resultados van desde 1.7 hasta 3.8 ciclos. Finalmente, para las consultas *easy* con niveles de preferencia de usuario de un 90.0% hasta un 60.0%, la cantidad de ciclos oscila entre 1.5 hasta 3.0.

La Figura 37b, muestra los resultados obtenidos con la base de datos PC, en la cual el algoritmo IC presenta un *ASL* de 5.2 ciclos para las consultas del tipo *hard*, 3.1 para consultas *moderate* y 2.3 para consultas *easy*. En el caso del algoritmo CCR con nivel de preferencia del 95.0%, se reducen los ciclos de recomendación y se obtiene una cantidad de ciclos de 1.9, 1.6 y 1.4 para las consultas *hard*, *moderate* y *easy*, respectivamente. Esto mismo, acontece con el resto de los niveles de preferencia de usuario, donde se obtienen para consultas *hard* ciclos que van desde 1.9 hasta 3.8, considerando niveles de preferencia desde 90.0% hasta un 60.0%, respectivamente. Para el caso de las consultas *moderate*, y teniendo en cuenta niveles de preferencia de usuario que van desde un 90.0% hasta un 60.0%, la cantidad de ciclos se mueve desde 1.7 hasta 2.9 ciclos. Finalmente, para las consultas *easy* con niveles de preferencia de usuario de un 90.0% hasta un 60.0%, la cantidad de ciclos oscila entre 1.4 hasta 2.3.

En el caso de la base de datos TRAVEL, la Figura 37c muestra que el algoritmo IC presenta los tres tipos de consultas –*hard*, *moderate* y *easy*– un *ASL* de 13.2, 10.5 y 8.1, respectivamente. Sin embargo, el algoritmo CCR muestra una alta eficiencia aún en bases de datos más grandes, ya que para niveles de preferencia de usuario del 95.0% se obtiene una cantidad de ciclos de 1.9, 1.8 y 1.6 para las consultas *hard*, *moderate* y *easy*, respectivamente. Esta mejora en la eficiencia del algoritmo, se repite en el resto de los niveles de preferencia de usuario, donde se obtienen para consultas *hard* una cantidad de ciclos que van desde 1.9 hasta 5.7, considerando niveles de preferencia desde 90.0% hasta un 60.0%, respectivamente. Para el caso de las consultas *moderate*, y teniendo en cuenta niveles de preferencia de usuario que van desde un 90.0% hasta un 60.0%, la cantidad de ciclos se mueve desde 1.8 hasta 5.9 ciclos. En el caso de las consultas *easy* con niveles de preferencia de usuario entre un 90.0% y un 60.0%, la cantidad de ciclos obtenida se encuentre entre 1.7 hasta 4.3.

En el caso de la base de datos SMARTPHONE los resultados son similares a los anteriores. En este caso, tal como muestra la Figura 37d, el algoritmo IC alcanza un *ASL* de 17.5, 14.8 y 10.5 para consultas *hard*, *moderate* y *easy*, respectivamente. Por otro lado, el algoritmo CCR obtiene mejores resultados en todos los tipos de consultas iniciales y para todos los niveles de preferencias de usuarios. En el caso de consultas del tipo *hard*, y para niveles de preferencias de usuario que van de un 95.0% a un 60.0%, el algoritmo CCR alcanza ciclos que van desde 1.9 hasta 14.4, respectivamente. Para las consultas *moderate*, la cantidad de ciclos para CCR-95% es 1.8 y alcanza el valor más alto para CCR-60% con 13.7 ciclos. En el caso de consultas *easy*, el mejor rendimiento de CCR respecto a IC se mantiene desde un nivel de preferencias de usuario de 95.0% (1.6 ciclos) hasta el 70.0% (5.0 ciclos). Sólo para el nivel de preferencias de usuario de 60.0% la cantidad de ciclos es muy cercana a la eficiencia de IC, ya que se obtienen 10.2 ciclos.

Finalmente, para el dominio de RESTAURANT, he utilizado la base de datos con mayor cantidad de productos (9945 productos). En este caso, la Figura 37e muestra que el algoritmo IC alcanza un *ASL* de 20.9, 18.5 y 17.5 para consultas *hard*, *moderate* y *easy*, respectivamente. Por otro lado, el algoritmo CCR obtiene mejores resultados en todos los experimentos realizados con esta base de datos. En el caso de consultas del tipo *hard*, y para niveles de

preferencias de usuario que van de un 95.0 % a un 60.0 %, el algoritmo CCR alcanza ciclos que van desde 2.8 hasta 18.7, respectivamente. Para las consultas *moderate*, la cantidad de ciclos para CCR-95 % es 2.2 y alcanza el valor más alto para CCR-60 % con 16.2 ciclos. En el caso de consultas *easy*, el rendimiento de CCR es superior a IC desde el nivel de preferencias de usuario de 95.0 % (1.9 ciclos) hasta el 60.0 % (15.1 ciclos).

Los mejores resultados del algoritmo CCR se obtienen para las consultas iniciales de baja dificultad (*easy*), sin embargo, para el resto de consultas también CCR es mejor que IC. Una observación interesante es que en el caso de CCR el *ASL* aumenta, para todas las bases de datos, cuando las preferencias de usuario son más diferentes, es decir, cuando el nivel de preferencias disminuye. A mayor nivel de preferencias, mayor reducción del *ASL* en una recomendación. Es decir, es mejor colaborar con usuarios que tengan productos muy similares a mis preferencias, aunque el sistema es estable y mantiene una reducción en *ASL* con preferencias menos parecidas.

La Figura 38 presenta los resultados obtenidos para cada base de datos, a través de los experimentos realizados y teniendo en cuenta el beneficio en la reducción de ciclos de recomendación que se obtiene por el algoritmo CCR respecto al algoritmo IC. La Figura 38 muestra que el beneficio desciende cuando la similitud en las preferencias del usuario  $u_g$ , respecto al producto que tiene otro usuario, se decrementa. Estos resultados establecen que es beneficioso colaborar cuando existe un mínimo nivel de preferencias similares entre los usuarios que pretenden colaborar.

Específicamente, el máximo beneficio del algoritmo CCR sobre el algoritmo IC se obtiene para las consultas iniciales del tipo *hard*, en las cuales se obtienen los siguientes beneficios: un 75.2 % para la base de datos CAMERA (véase la Figura 38a), 63.0 % para la base de datos PC (véase la Figura 38b), 85.3 % para la base de datos TRAVEL (véase la Figura 38c), un 88.8 % para la base de datos SMARTPHONE (véase la Figura 38d) y un 86.6 % para la base de datos RESTAURANT (véase la Figura 38e) con un nivel de similitud en las preferencias de 95 %. La figura además muestra que para un nivel de preferencias de 60 %, el beneficio es mucho menor, porque mientras más diferencias existan entre el producto actualmente recomendado al usuario líder ( $u_h$ ) y el producto objetivo del usuario colaborador ( $u_g$ ), más número de ciclos necesitará para alcanzar el producto objetivo el usuario  $u_g$ .

Adicionalmente, se puede observar que el tipo de consulta inicial tiene diferentes efectos en cada base de datos. En este sentido, las diferencias de beneficios entre los tipos de consultas son mayores en las base de datos más pequeñas (PC y CAMERA)—tal y como se puede observar en la Figura 38a y en la Figura 38b— mientras que esas diferencias son menores en las bases de datos más grandes (TRAVEL, SMARTPHONE y RESTAURANT), como se puede observar en las Figuras 38c, 38d y 38e. Por ejemplo, las consultas del tipo *easy* en las bases de datos PC o CAMERA, necesitan una menor cantidad de ciclos para alcanzar el producto objetivo que en una consulta del tipo *hard*, la cual requiere una mayor cantidad de ciclos. Por otro lado, las bases de datos TRAVEL, SMARTPHONE y RESTAURANT, que tienen un número mayor de productos, requieren más ciclos para alcanzar el objetivo, independiente del tipo de consulta inicial. Por esta razón, hay menos diferencias en el beneficio obtenido en los diferentes tipos de consultas iniciales, para las bases de datos más grandes.

Si bien, los resultados obtenidos demuestran que el algoritmo CCR obtiene mejores resultados que el algoritmo IC en diferentes escenarios (distintas bases de datos y distintos tipos de consultas iniciales), es necesario evaluar

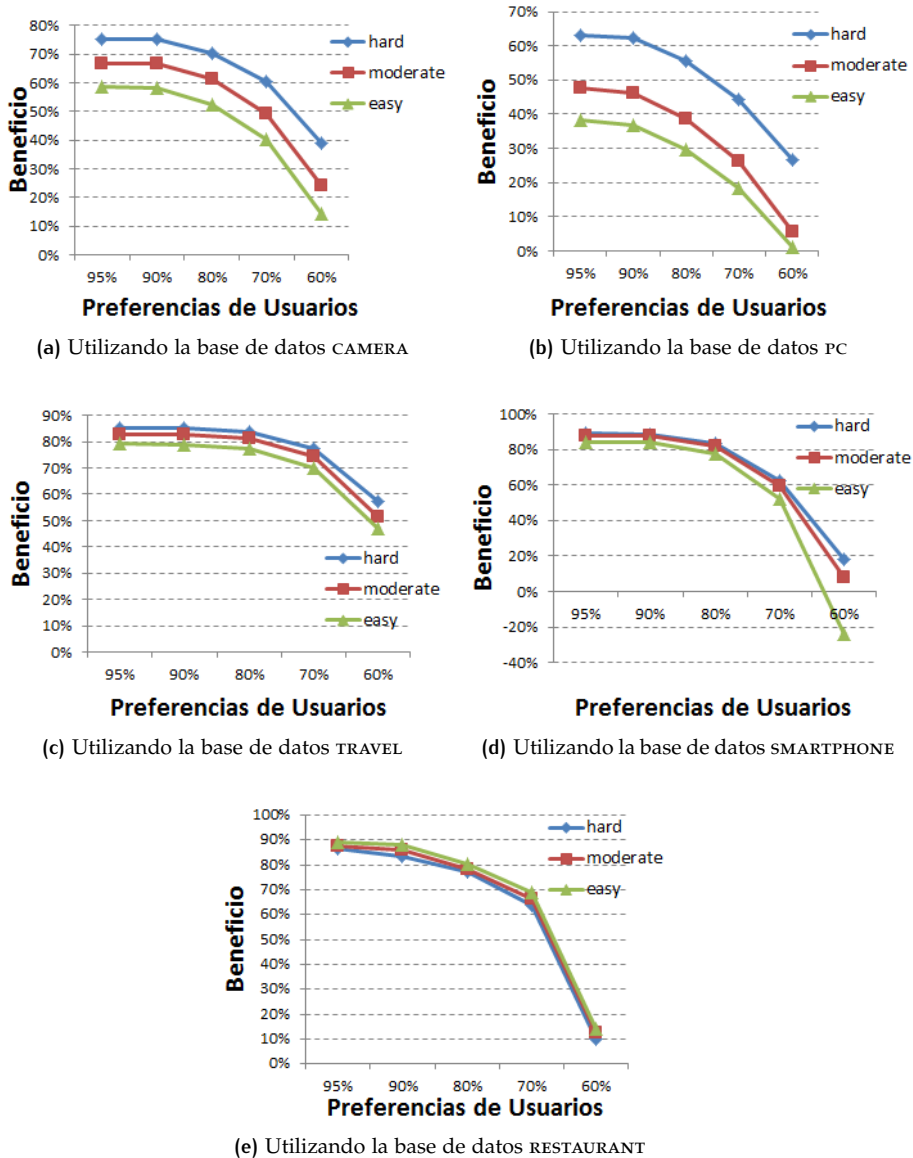


Figura 38: Beneficio obtenido por el algoritmo CCR en comparación con el algoritmo IC, para las bases de datos CAMERA, PC, TRAVEL, SMARTPHONE y RESTAURANT

si la reducción en el número de ciclos es significativamente menor en el algoritmo CCR, respecto a IC. Por lo anterior, he analizado estadísticamente los resultados por medio de los test de Friedman y Nemenyi.

Para realizar el análisis estadístico, primero he calculado el ranking medio de cada algoritmo ( $r$ ), considerando todos los experimentos ( $k$  algoritmos y  $N$  diferentes experimentos para cada prueba). En mi caso, los algoritmos comparados son: IC, CCR-95, CCR-90, CCR-80, CCR-70 y CCR-60. Los algoritmos "CCR-n", me permiten evaluar distintos niveles de similitud en las preferencias de los usuarios colaboradores. Además, los experimentos dependen del número de consultas iniciales (*hard*, *moderate* y *easy*) y del número de bases de datos (en esta caso PC, CAMERA, TRAVEL, SMARTPHONE y RESTAURANT). El ranking de los algoritmos lo he realizado definiendo con el ranking

1 al algoritmo que obtuvo mejor rendimiento (menor número de ciclos, *ASL*), el segundo mejor algoritmo lo he clasificado como 2 y así sucesivamente. De esta forma, el ranking medio para cada algoritmo es el promedio obtenido entre todos los experimentos realizados.

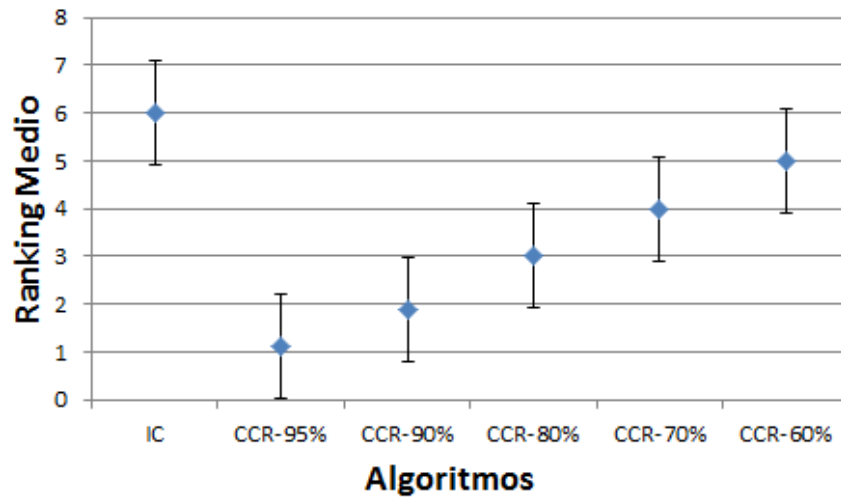


Figura 39: Test de Nemenyi sobre el ranking medio de los algoritmos, respecto a la medida *ASL*

Después de calcular el ranking medio, he aplicado el test de Friedman. Para esto, he configurado el test considerando ( $k = 6$ ) algoritmos y ( $N = 15$ ) experimentos (tres diferentes consultas iniciales sobre cinco diferentes bases de datos). Bajo esta configuración,  $F_F$  se distribuye de acuerdo a la distribución F con  $(6 - 1) = 5$  y  $(6 - 1) \cdot (15 - 1) = 70$  grados de libertad. El valor crítico de  $F(5, 70)$  es igual a 2.34 con un nivel de confiabilidad de 0.05. En la comparación de la eficiencia de los algoritmos, he obtenido los valores de  $X_F = 17.32$  y  $F_F = 1347.1$ . Dado que el valor  $F_F$  es mayor que 2.34, he podido rechazar la hipótesis nula y concluir que existen diferencias significativas en la eficiencia de los algoritmos.

Una vez que he validado que los resultados poseen diferencias significativas, he aplicado el test de Nemenyi para identificar los algoritmos que poseen dichas diferencias significativas. En mi caso, teniendo en cuenta que comparo seis algoritmos con un valor de confiabilidad de  $\alpha = 0.05$ ,  $q_{0.05} = 2.850$  para el test de Nemeny *two-tailed*, he obtenido un valor de diferencia crítica de  $CD = 2.18$ . Los resultados analizados anteriormente, se muestran en la Figura 39.

En la Figura 39, los diamantes representan el ranking medio de cada algoritmo. Las líneas verticales que traspasan los diamantes indican la "diferencia crítica". Aplicando el test de Nemenyi, se tendría que la eficiencia de dos algoritmos es significativamente diferente si sus correspondientes ranking medios difieren al menos en el valor de diferencia crítica, es decir, dos algoritmos son significativamente diferentes si sus líneas verticales no se solapan. La Figura 39 muestra que el algoritmo CCR con un nivel de similitud de preferencias entre un 95 % y un 80 % obtiene el mejor ranking medio con valores de 1.13, 1.88 y 3.00, respectivamente. Además, la diferencia es significativamente mejor que IC (que obtiene un ranking medio de 6). Debo destacar que CCR-70 y CCR-60 (con un nivel de preferencias de usuario de 70 % y 60 %) son mejores que IC pero no muestran una diferencia significativa. Por



lo anterior, considero que incluso con un nivel de preferencias bajo, el algoritmo CCR obtiene buenos resultados y cuando el nivel de preferencias es mayor (es decir, hay una mayor similitud entre los usuarios colaboradores), el beneficio es significativamente mayor.

Resumiendo, en mis experimentos he simulado el comportamiento colaborativos de los usuarios. En este sentido, un usuario sin mucho conocimiento del producto que requiere, debiera reducir los ciclos de búsqueda del producto, si él colabora con otro usuario que tenga preferencias similares. Sin embargo, los resultados de los experimentos muestran que es necesario un mínimo nivel de similitud entre las preferencias de los usuarios para que la colaboración sea beneficiosa. Aunque, mi análisis muestra que en el algoritmo CCR, el mínimo nivel de similitud en las preferencias debiera ser cercano a un 80 % para obtener diferencias significativas con los otros algoritmos, el usuario puede también obtener beneficios incluso con niveles cercanos al 70 % y 60 %.

#### 5.2.6 Resultados de integración de información colaborativa *off-line* y *on-line*

En esta sección presento los resultados de los experimentos que he realizado para evaluar la eficiencia del algoritmo HOOCR, el cual integra los dos tipos de información colaborativa que he analizado en esta tesis: información colaborativa *off-line* e información colaborativa *on-line*. Esta técnica la he descrito en detalle en la Sección 3.9.

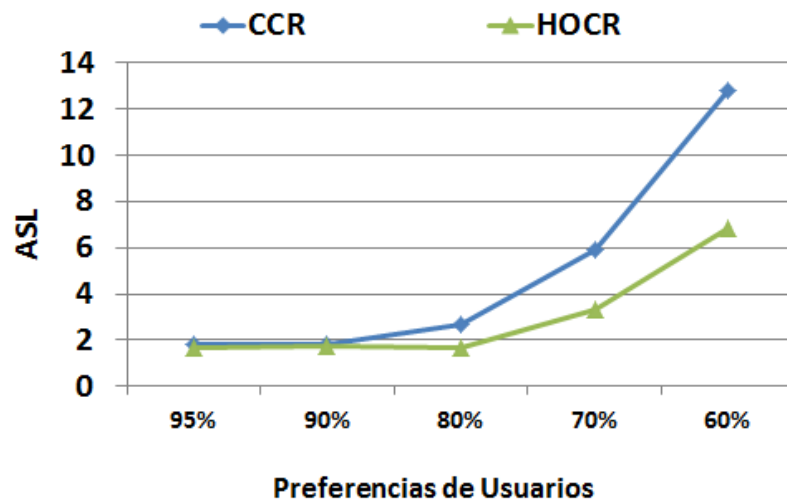
Específicamente, en esta evaluación he comparado mi propuesta con el algoritmo tradicional basado en críticas IC. Sin embargo, dado que esta técnica integra información proveniente de opiniones y sesiones históricas, tal y como la técnica HOR e integra información *on-line* como el algoritmo CCR, creo necesario incluir dentro de la comparación estos algoritmos. En el caso particular del algoritmo HOR, incluyo en la comparación la propuesta que obtuvo mejores resultados, el algoritmo HORi. Por lo tanto, los cuatro algoritmos de recomendación evaluados en esta sección corresponden a: (1) algoritmo Incremental Critiquing (IC); (2) el algoritmo mejorado basado en opiniones y sesiones históricas HORi; (3) el algoritmo de recomendaciones conversacionales colaborativas (CCR); y (4) el algoritmo que integra información colaborativa *off-line* y *on-line* (HOOCR).

Para estos experimentos he utilizado las bases de datos SMARTPHONE y RESTAURANT, dado que son las únicas de las que dispongo de información colaborativa *off-line*.

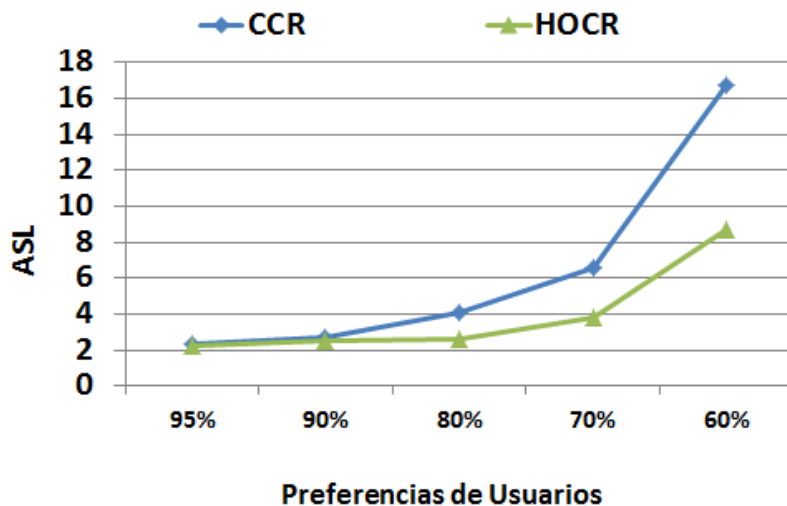
Un punto importante en la evaluación del algoritmo HOOCR, al igual que en el algoritmo CCR, corresponde a los niveles de preferencias que existan entre los usuarios cuando se realizan acciones de colaboración. Por lo anterior, previo a realizar las comparativas entre HOOCR y el resto de algoritmos mencionados anteriormente, he analizado los resultados que se obtienen al comparar el comportamiento del algoritmo HOOCR frente al algoritmo CCR para distintos niveles de preferencias. Este análisis lo realizo con el objetivo de simplificar la visualización de resultados y realizar la comparativa en base a un solo nivel de preferencias.

En concreto, he analizado los resultados obtenidos para los niveles de preferencias comprendidos entre 95 % y 60 %, tal y como lo hice anteriormente con el algoritmo CCR (véase la Sección 5.2.5). La Figura 40 muestra los resultados obtenidos por ambos algoritmos para los distintos niveles de preferencias en los dominios de SMARTPHONE y RESTAURANT. En esta figura se

puede observar que para niveles de preferencia de 90 % y 95 % los resultados son muy similares para ambos algoritmos. Sin embargo, a partir del nivel de preferencias de 80 % el algoritmo HOCR reduce la cantidad de ciclos con respecto al algoritmo CCR. En particular, para el dominio de SMARTPHONE (véase la Figura 40a), el algoritmo CCR alcanza un total de 2.6 ciclos y el algoritmo HOCR reduce los ciclos a 1.7 ciclos. Por otro lado, para el dominio de RESTAURANT (véase la Figura 40b), el algoritmo HOCR reduce la cantidad de ciclos obtenidos por CCR (4.12 ciclos) a 2.65 ciclos. Adicionalmente, para el nivel de preferencias mayores se mantiene la tendencia en la reducción de ciclos por parte de HOCR. Por ejemplo, para el nivel de preferencias de 60 %, el algoritmo HOCR reduce los ciclos desde 12.78 ciclos hasta 6.84 en el caso del dominio de SMARTPHONE (véase la Figura 40a) y desde 16.69 hasta 8.48 en el caso del dominio de RESTAURANT (véase la Figura 40b).



(a) Base de datos de smartphones



(b) Base de datos de restaurants

Figura 40: Comparación de *ASL* según niveles de preferencias de los usuarios para HOCR y CCR, utilizando las bases de datos de productos SMARTPHONE y RESTAURANT

Estos resultados, me llevan a concluir que el algoritmo HO CR tiene un comportamiento similar al algoritmo CCR cuando los usuarios tienen preferencias muy similares en el producto que están buscando (es decir, en el rango entre 95 % y 90 %). Sin embargo, cuando las preferencias de los usuarios difieren más allá del 80 %, el algoritmo HO CR puede mejorar el rendimiento de CCR por medio de la información colaborativa *off-line* que utiliza en la recomendación.

Con el objetivo de realizar una comparación más cercana a la realidad, utilizaré un nivel de preferencias de usuario promedio entre el valor donde se inician las diferencias entre HO CR y CCR (80 %) y el valor más alto evaluado (60 %). Por lo anterior, para las comparaciones de HO CR con el resto de las técnicas (IC, HORi, y CCR) utilizaré un nivel de preferencias de 70 %.

Para la evaluación de HO CR, he adoptado la misma metodología que he aplicado en la evaluación de los algoritmos basados en opiniones y sesiones de recomendación históricas (véase la Sección 5.2.6 y el Cuadro 6 donde se describe la metodología utilizada para la generación de las sesiones históricas). En concreto, en estos experimentos he utilizado las cinco bases de datos de sesiones de diferentes tamaños para cada dominio (véase la columna *Bases de Sesiones* en el Cuadro 7). Adicionalmente, he dividido las dos bases de datos en subconjuntos de la base de datos original, con el objetivo de evaluar los algoritmos de recomendación en espacios de búsqueda de diferentes tamaños (véase la columna *Espacios de Búsqueda* en el Cuadro 7). Al igual que en todos los experimentos previos, también utilizo la medida denominada *ASL* para evaluar la eficiencia de los algoritmos y además evalúo el beneficio de los algoritmos con respecto al algoritmo base IC.

En la Figura 41 presento varios gráficos con los resultados del análisis realizado de la eficiencia de cada algoritmo evaluado para alcanzar el producto objetivo, teniendo en cuenta los diferentes tamaños de las bases de sesiones históricas y en relación al tamaño de los espacios de búsqueda. En estos gráficos, cada línea muestra el promedio de la medida *ASL* obtenido por cada algoritmo, para las consultas iniciales definidas (*hard*, *moderate* y *easy*). Como se puede observar en los gráficos de la Figura 41, los algoritmos IC y CCR muestran los mismos resultados para distintos tamaños de bases de sesiones, ya que estos algoritmos no utilizan sesiones históricas de otros usuarios para definir la siguiente recomendación.

En las Figuras 41a, 41c y 41e, muestro los resultados utilizando 1721, 1300 y 1000 smartphones como base de productos, respectivamente.

En particular, en la Figura 41a, el algoritmo IC y el algoritmo CCR obtienen para todos los tamaños de las bases de datos de sesiones históricas un *ASL* de 14.3 y 5.9, respectivamente. Por otro lado, para el algoritmo HORi el rango de ciclos va desde 9.6 ciclos en la base de datos de 500 sesiones hasta 8.6 ciclos en la base de datos de 10000 sesiones. Finalmente, el algoritmo HO CR obtiene los mejores resultados con un rango de ciclos desde 3.3 ciclos para la base de datos de 500 sesiones hasta 2.4 ciclos en la base de datos de 10000 sesiones.

En el caso de las bases de datos de productos de menor tamaño (1300 y 1000 smartphones), los resultados también son mejores para HO CR. Por ejemplo, para la base de datos de 1300 smartphones, la Figura 41c muestra que para 10000 sesiones históricas la medida *ASL* para IC, HORi, CCR y HO CR corresponden a 13.2, 7.46, 4.82 y 2.16, respectivamente. En el caso de la base de datos de productos más pequeña (1000 smartphones) con 10000 sesiones el algoritmo IC alcanza 12.1 ciclos, el algoritmo HORi obtiene 7.5

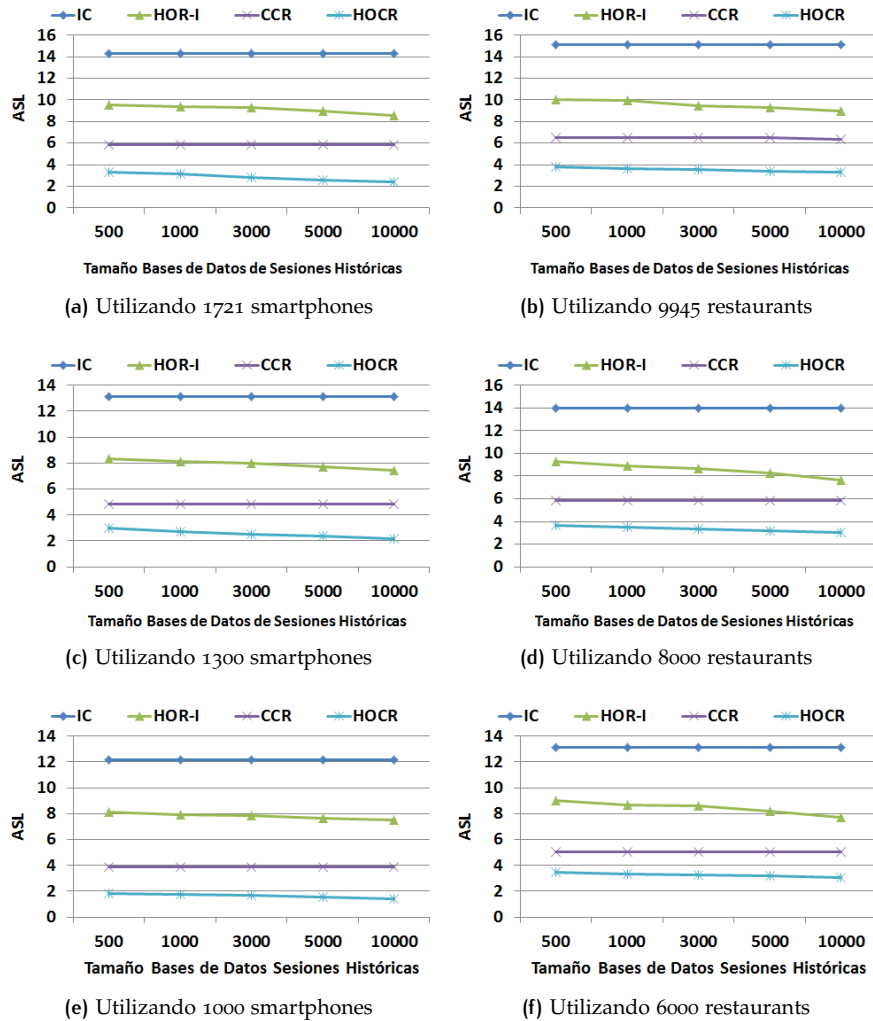


Figura 41: Resultados de *ASL* de la técnica propuesta HO CR en comparación con las técnicas IC, HORi y CCR en dominios de SMARTPHONE y RESTAURANT

ciclos, el algoritmo CCR alcanza 3.9 ciclos y el algoritmo HO CR reduce los ciclos a 1.4 (véase la Figura 41e).

Para el dominio de restaurants, en las Figuras 41b, 41d y 41f muestro los resultados utilizando 9945, 8000, y 6000 RESTAURANTS como base de datos de productos, respectivamente. En la base de datos de RESTAURANT, el algoritmo IC obtiene un *ASL* de 18.8, 18.1 y 17.6 para cada tamaño de base de productos (9945, 8000 y 6000), respectivamente. En el caso del algoritmo CCR, los resultados obtenidos para la medida *ASL* son de 6.5, 5.8 y 5.1 para cada tamaño de base de productos (9945, 8000 y 6000), respectivamente. Por otra parte, los resultados del algoritmo HORi varían dependiendo del tamaño de la base de datos de sesiones. En primer lugar, para la base de datos más grande (véase la Figura 41b) los ciclos van desde 10.1 para 500 sesiones históricas, hasta 8.9 ciclos para 10000 sesiones históricas. En segundo lugar, para la base de datos de 8000 productos (véase la Figura 41d) los ciclos van desde 9.3 para 500 sesiones históricas, hasta 7.6 ciclos para 10000 sesiones históricas. Finalmente, en la Figura 41f que corresponde a la base de datos más pequeña, el rango de la cantidad de ciclos de HORi se mueve entre 8.9

y 7.7 ciclos. En la base de datos de RESTAURANT, el algoritmo HO CR también obtiene los mejores resultados en comparación con los algoritmos IC, HORi y CCR. En concreto, para la base de datos más grande (9945 restaurants) el algoritmo HO CR obtiene un rango de ciclos que va desde 3.7 para 500 sesiones históricas a 3.3 ciclos cuando se utilizan 10000 sesiones (véase la Figura 41b). Para la base de datos de 8000 restaurants, el rango de ciclos se mueve entre 3.6 (500 sesiones históricas) y 3.0 ciclos (10000 sesiones históricas). Finalmente, para la base de datos más pequeña (6000 restaurants), los ciclos alcanzados por HO CR tienen un rango de 3.4 ciclos (500 sesiones históricas) a 2.9 ciclos (10000 sesiones históricas).

Analizando los resultados en ambos dominios de productos, puedo concluir:

- El algoritmo HO CR al igual que el algoritmo HORi, obtiene los mejores resultados cuando se tiene mayor cantidad de información en las bases de datos de sesiones históricas (información *off-line*), pero al igual que HORi, no es exclusivamente dependiente de dicho tamaño, ya que complementa la información de sesiones históricas con información proveniente de las opiniones de otros usuarios. Esta característica la diferencia de otros enfoques orientados a sesiones históricas, como es el caso de HGR que depende completamente del tamaño de las base de datos de sesiones históricas, para reducir los ciclos de recomendación.
- El algoritmo HO CR muestra en los resultados un comportamiento similar al algoritmo HORi, pero mejora los resultados al integrar la información colaborativa *on-line* que proviene de la técnica CCR. Es por esto, que se puede observar que en todos los experimentos la técnica HO CR supera tanto a HORi como a CCR.

A continuación, me centraré en analizar el beneficio que se obtiene al utilizar el algoritmo HO CR, en comparación con el algoritmo base IC. La Figura ?? muestra el beneficio promedio del algoritmo propuesto en comparación con el algoritmo IC, usando diferentes bases de datos de sesiones históricas (500 a 10000 sesiones para ambas bases de datos SMARTPHONE y RESTAURANT) y utilizando los diferentes tipos de consultas iniciales (*hard*, *moderate* y *easy*). Primero que todo, se puede observar que el beneficio depende tanto del tamaño de la base de sesiones (es decir, el beneficio es más grande para bases de sesiones históricas más grandes, como también sucedía en el algoritmo HORi), como también del tipo de consulta, donde en la mayoría de los casos las consultas del tipo *hard* obtienen mayores beneficios (es decir, este comportamiento es similar al algoritmo CCR).

En concreto, la Figura 42 muestra que el beneficio del algoritmo HO CR en la base de datos SMARTPHONE obtiene un beneficio que va desde un 78.7% a un 81.3%, desde un 81.8% a un 85.7% y desde un 73.7% hasta un 82.8% para consultas iniciales del tipo *hard*, *moderate* y *easy*, respectivamente. Por otro lado, utilizando la base de datos RESTAURANT, se han obtenido resultados similares a la base de datos SMARTPHONE. En la Figura 42, se puede observar que el algoritmo HO CR obtiene un beneficio en el rango de 78.7% a 81.3% para las consultas iniciales del tipo *hard*. En el caso de las consultas del tipo *moderate* el rango de beneficio se mueve entre un 71.3% y un 75.5%. Finalmente, para las consultas del tipo *easy* el rango del beneficio de HO CR en comparación con el algoritmo IC va desde un 69.2% a un 73.6%.

Teniendo en cuenta los beneficios obtenidos por los algoritmos HORi (véase la Figura 35) y el algoritmo CCR (véase la Figura 38), puedo concluir que el beneficio en la eficiencia de la recomendación del algoritmo HO CR es mayor

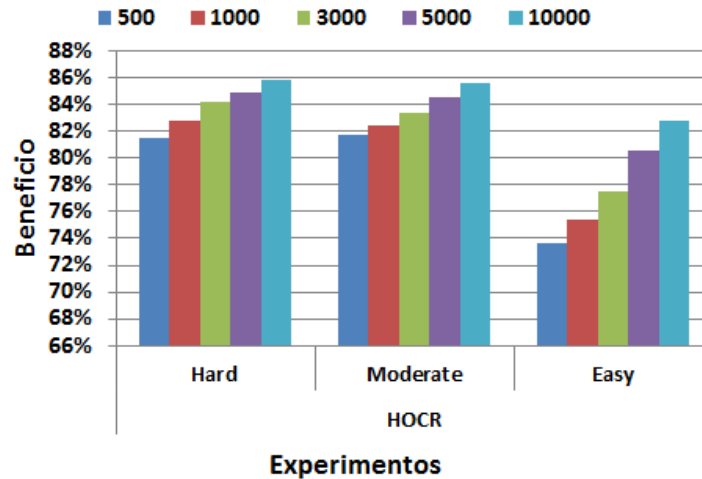


Figura 42: Beneficios del algoritmo HOCR en comparación con el algoritmo IC

que las técnicas previas de recomendación HORi y CCR. Esto se demuestra en los dos dominios de evaluación utilizados: SMARTPHONE y RESTAURANT.

Adicionalmente, con el objetivo de demostrar que la eficiencia del algoritmo HOCR es significativamente mejor que el algoritmo base IC y que el resto de algoritmos evaluados en esta sección (HORi y CCR), he aplicado el método estadístico ANOVA (Fisher, 1956; Sheskin, 2003) en los resultados obtenidos en ambos conjuntos de datos.

Al aplicar ANOVA sobre los resultados obtenidos para las dos bases de datos (SMARTPHONE y RESTAURANT), obtengo un valor significativo de  $p = 2.2e^{-16} < 0.001$ , que verifica que existen diferencias significativas entre los algoritmos evaluados. Posteriormente, aplico el test de Bonferroni (Bland and Altman, 1995) para validar si las medias de los algoritmos son estadísticamente significativas. Al aplicar el test de Bonferroni obtengo un valor de  $p = 2.4e^{-5} < 0.001$  entre HOCR y CCR;  $p = 2e^{-16} < 0.001$  entre HOCR y los algoritmos IC y HORi. En resumen, puedo concluir que el algoritmo HOCR es significativamente mejor que los algoritmos IC, CCR y HORi.

### 5.3 ESTUDIO CON USUARIOS REALES DE LAS INTERFACES DE USUARIO INTELIGENTES

En esta sección presento la evaluación de las interfaces de usuario inteligentes que integran las técnicas de recomendación propuestas. En concreto, la evaluación la he realizado con usuarios reales sobre las dos interfaces propuestas en esta tesis: **e-WebCo** (véase la Sección 4.3) y **e-3DVirtualCo** (véase la Sección 4.4). Para cada una de las interfaces describo: (1) la metodología utilizada para la evaluación y (2) el análisis de los resultados obtenidos a través de los experimentos realizados con los usuarios reales.

#### 5.3.1 Evaluación interfaz e-WebCo

En esta sección presento los resultados obtenidos de la evaluación de la interfaz inteligente de dos dimensiones **e-WebCo**. Para estos experimentos he utilizado los algoritmos de recomendación basados en sesiones históri-

cas y opiniones. Sin desmedro de lo anterior, es necesario recordar que esta interfaz de usuario permite integrar cualquier tipo de recomendador conversacional.

En los experimentos realizados, se evalúa la eficiencia y la eficacia del algoritmo de recomendación integrado en la interfaz de usuario. Específicamente, en esta interfaz he evaluado los parámetros mencionados anteriormente para las propuestas HORi y HGRi<sup>8</sup>, en comparación con el algoritmo tradicional IC. Adicionalmente, se evalúa la satisfacción de los usuarios y la percepción de uso de la interfaz **e-WebCo**.

#### 5.3.1.1 e-WebCo: Metodología

En los experimentos he utilizado una evaluación sumativa, la cual es adecuada para prototipos finales que incorporan la mayor parte de las funcionalidades requeridas, y se focaliza en obtener información cualitativa y cuantitativa de parte de los usuarios reales (Bowman et al., 2002). En concreto, para realizar las pruebas se agregaron a las funcionalidades propias del recomendador instrucciones, interfaces de ayuda y cuestionarios que permiten a los participantes interactuar remotamente con la interfaz sin necesidad de tener un instructor o guía durante el experimento. Por lo anterior, la interfaz web cuenta con una sección de ayuda que explica completamente la interfaz del recomendador, la forma de interactuar con él y las tareas definidas para la evaluación.

**Cuadro 8:** Características de los usuarios reales en las pruebas de **e-WebCo**

Rango de edad	#Usuarios	#Hombres	#Mujeres	Experiencia en Recomendadores
20-29	22	14	8	Medio-alto
30-39	7	4	3	Medio-bajo
40-49	6	4	2	Bajo

El experimento fue diseñado como un test *between-subject*. Es decir, un participante solo evalúa alguno de los algoritmos disponibles (no todos) y para cada algoritmo el participante completa un cuestionario al final del experimento *post-test*. En el experimento participaron 35 personas, con diversas características, tales como edad, género y habilidades informáticas (véase el Cuadro 8). Con el fin de motivar la participación por parte de los usuarios, utilicé la base de datos SMARTPHONE, ya que es una de las bases de datos más actualizadas que dispongo. El protocolo de evaluación lo describo en detalle en el Cuadro 9.

Después de realizar el test, he recogido los datos desde los registros del recomendador y desde los cuestionarios. A continuación describo en detalle los resultados obtenidos del análisis de estos datos.

#### 5.3.1.2 e-WebCo: Resultados de los experimentos con usuarios reales

En esta sección presento los resultados obtenidos respecto a eficiencia, eficacia y satisfacción de usuario para los algoritmos IC, HGRi y HORi. En este caso, como en el resto de los experimentos, el algoritmo IC es mi algoritmo base de comparación.

<sup>8</sup> La evaluación con usuarios reales se realizó con los algoritmos que obtuvieron los mejores resultados en las pruebas realizadas con el simulador.

**Cuadro 9:** Descripción de las fases del protocolo de pruebas en **e-WebCo**

1. *Pre-test:* En esta fase se solicita a los participantes que ingresen información personal no privada y sus experiencias previas con sistemas de recomendación.
2. *Entrenamiento:* En esta fase los participantes realizan tareas de entrenamiento, donde ellos deben encontrar un producto objetivo predefinido, el cual es seleccionado aleatoriamente desde la base de datos. El usuario tiene la libertad de poder repetir varias veces la prueba de entrenamiento.
3. *Test de Eficiencia:* En esta fase los usuarios realizan dos tareas con un producto objetivo predefinido, con el objetivo de evaluar la eficiencia de la recomendación del algoritmo evaluado. Esta forma de evaluar con un objetivo predefinido es habitual en evaluaciones de algoritmos basados en *critiquing* (Chen and Pu, 2007a; Chen and Pu, 2008). Para esta fase, utilizo como unidad de medida de eficiencia la medida *ASL*. Para implementar esta fase, la aplicación **e-WebCo** selecciona aleatoriamente un algoritmo (IC, HGRi o HORi) para cada tarea que realiza el usuario.
4. *Test de Eficacia:* En esta fase los usuarios realizan una tarea en la cual no existe un producto objetivo. El propósito de esta tarea es evaluar la eficacia de los recomendadores HGRi y HORi, por medio de la medida reconocida habitualmente por su nombre en inglés *decision accuracy*. Para medir la *decision accuracy*, he aplicado una metodología basada en trabajos previos de evaluación de eficacia en recomendadores conversacionales (Chen and Pu, 2006) y que también se ha aplicado en otros campos de estudios como el marketing (Häubl and Trifts, 2000). En particular, la medida de *decision accuracy* se obtiene de la comparación entre el producto seleccionado al final del ciclo de recomendación (véase la Figura 43a) y el producto que el usuario puede seleccionar si tiene en frente la lista completa de los productos disponibles en la base de datos (véase la Figura 43b). De esta forma, un algoritmo con alta eficacia (*decision accuracy*), debiera tener un alto grado de coincidencia entre el producto seleccionado usando el recomendador y el producto seleccionado por el usuario desde una lista completa de la base de datos. Específicamente, en mis experimentos he definido la eficacia como la similitud entre el producto finalmente seleccionado con el recomendador y el producto seleccionado desde la lista completa de productos. Dado que la base de datos SMARTPHONE es demasiado grande (1722 productos), he seleccionado un subconjunto de 90 productos para mejorar la visualización de la lista completa de productos. La selección de los productos que conforman la nueva base de datos reducida se hizo tomando en cuenta el atributo *manufacturer*, ya que es uno de los más representativos del producto. Para esto, seleccioné aleatoriamente una cantidad proporcional de productos para cada marca de smartphone. Por ejemplo, para la marca Samsung (413 productos), que es cercano al 24.0% de los datos originales (1722 productos) se seleccionaron aleatoriamente 21 productos de esta marca.
5. *Satisfacción del Usuario:* Para cada tarea realizada en la fase test de eficiencia, los participantes contestaron un cuestionario, el cual consiste de 5 preguntas (véase Cuadro 10). Para cada pregunta se utiliza una escala de siete puntos, donde 1 corresponde al concepto "muy en desacuerdo" y 7 corresponde a "muy de acuerdo". Adicionalmente, para la fase de test de eficacia, los participantes contestaron un cuestionario consistente en 7 preguntas (véase Cuadro 11).



La Figura 44a muestra que el *ASL* en ambos algoritmos, HGRi (18.04) y HORi (16.08) es menor que en el algoritmo IC (22.94). Adicionalmente, la línea azul de esta figura muestra el beneficio de ambos algoritmos (HGRi y HORi) sobre el algoritmo IC. En concreto, HGRi alcanza un beneficio sobre IC de 21.3 % y en HORi el beneficio es 29.9 %. Estos resultados confirman los resultados obtenidos por el simulador y que he descrito en la Sección 5.2.4. La figura muestra además que los mejores resultados para la eficiencia del algoritmo los obtiene el algoritmo HORi.

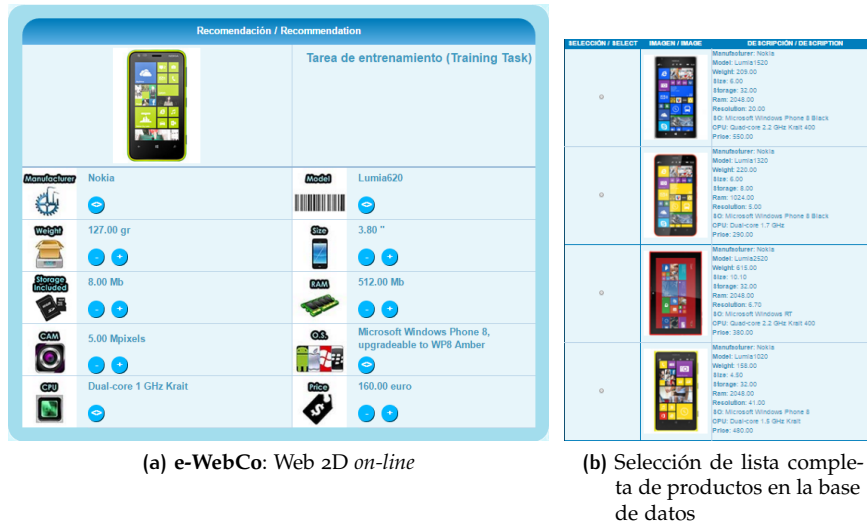
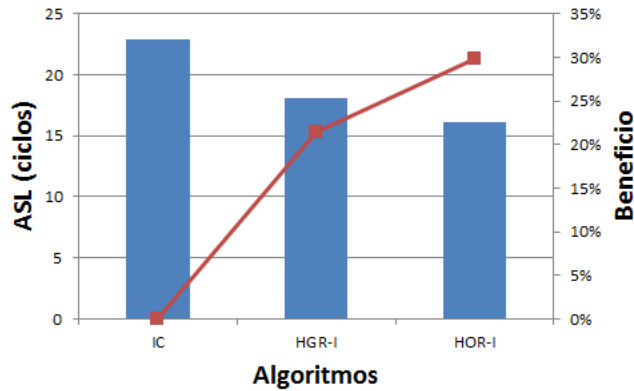


Figura 43: Pruebas con usuarios reales en e-WebCo

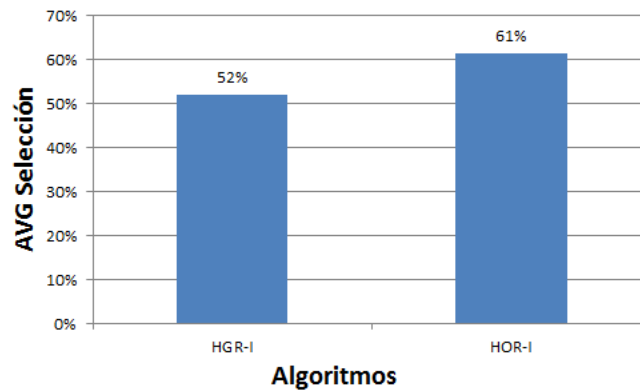
Adicionalmente, para validar si existe una diferencia significativa en la eficiencia de los algoritmos, he aplicado el método estadístico ANOVA (Sheskin, 2003; Sheskin, 2003). En particular, he aplicado ANOVA en tres algoritmos,  $k = 3$ , con  $k - 1 = 2$  grados de libertad. Los resultados de ANOVA muestran que las diferencias son significativas entre los algoritmos, donde *p-value* es 0.02453 que es menor que el valor crítico  $\alpha = 0.05$ . De forma complementaria, aplico el test de Bonferroni (Bland and Altman, 1995) para demostrar que la eficiencia de HORi y HGRi son significativamente mejores que IC por separado. Al aplicar el test de Bonferroni obtengo un *p-value* de 0.047 entre IC y HGRi y 0.028 entre IC y HORi. Estos valores obtenidos por Bonferroni confirman que tanto HGRi como HORi son significativamente mejores que IC.

La Figura 44b muestra los resultados de la evaluación de la eficacia de la recomendación (*decision accuracy*), donde HORi obtiene un nivel de eficacia mayor que HGRi, 61.0 % y 52.0 %, respectivamente. Esta medida significa que en promedio, en el recomendador HORi el producto final aceptado por el usuario es 61.0 % similar a la selección final realizada desde la lista completa de productos.

Para recoger las medidas asociadas a la satisfacción del usuario, he diseñado dos cuestionarios *post-test*, que permiten evaluar la percepción de los usuarios en dos escenarios diferentes: (1) cuando los usuarios interactúan con el recomendador para alcanzar un producto objetivo predeterminado (véanse las preguntas en el Cuadro 10); (2) cuando los usuario interactúan con el recomendador para alcanzar un producto deseado y sin un producto objetivo predefinido (véanse las preguntas en el Cuadro 11).



(a) Eficiencia en la recomendación



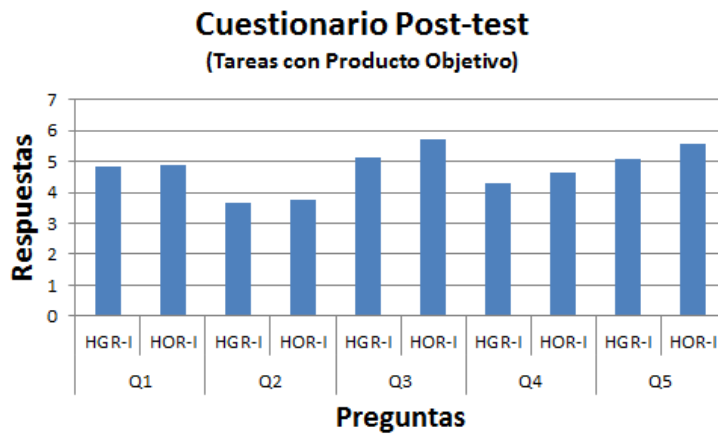
(b) Eficacia en la recomendación

Figura 44: Resultados de eficiencia y eficacia en la evaluación con usuarios reales

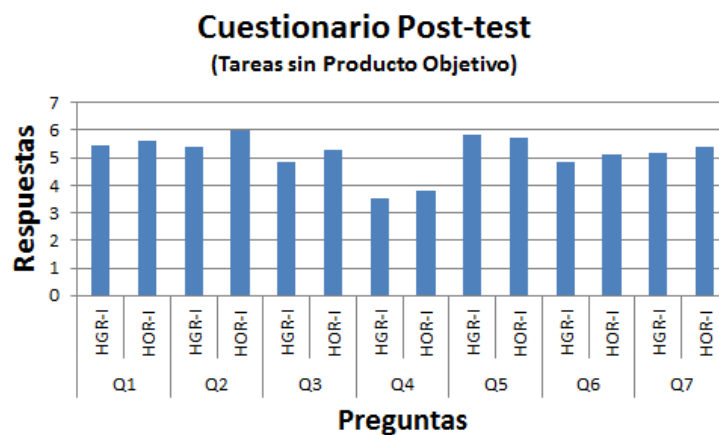
Los resultados obtenidos se muestran en las Figuras 45a y 45b. Es necesario destacar que, en promedio, todas las respuestas en ambas figuras son mayores que 3.54 en una escala de 1 a 7.

La Figura 45a muestra que los participantes evalúan positivamente la eficiencia de las propuestas (Q1 y Q2). En particular, ellos evalúan Q1 con un promedio de 4.81 para HGRi y 4.88 para HORi. Adicionalmente, las respuestas a Q2, que obtiene un promedio de 3.65 para HGRi y 3.75 para HORi, muestran que mis propuestas entregan recomendaciones adecuadas a los usuarios en periodos de tiempo corto. Por otro lado, los participantes evalúan positivamente la privacidad del recomendador respecto a su información (Q3) donde se obtienen valores de 5.13 en HGRi y 5.70 en HORi. Finalmente, las respuestas a la pregunta Q4 (que obtiene un promedio de 4.31 para HGRi y 4.63 para HORi) y a la pregunta Q5 (que obtiene un promedio de 5.06 para HGRi y 5.57 para HORi), muestran un alto nivel de satisfacción de parte de los usuarios respecto al uso del recomendador.

La Figura 45b muestra que los resultados respecto a la eficacia de los recomendadores propuestos son bastante satisfactorios, como se puede observar en las respuestas a la pregunta Q1 (que obtiene un valor de 5.45 para HGRi y 5.64 para HORi) y Q2 (con un valor de 5.36 para HGRi y 6.0 para HORi). Adicionalmente, la Figura 45b muestra un alto nivel de satisfacción en las respuestas que los usuarios entregan a las preguntas Q3 y Q4. Específicamente,



(a) Valor promedio de respuestas para cuestionario de eficiencia



(b) Valor promedio de respuestas para cuestionario de eficacia

Figura 45: Percepción de los usuarios acerca de la eficiencia y eficacia de los recomendadores propuestos

el promedio de Q3 alcanza un valor de 4.82 para HGRi y 5.27 para HORi. En el caso de Q4, HGRi obtiene un promedio de 3.54 y HORi un promedio de 3.82, lo cual significa que los participantes perciben que estos algoritmos proveen recomendaciones adecuadas en un periodo de tiempo corto. Aún más, los participantes evalúan positivamente la privacidad de su información en el recomendador (Q5), obteniendo un 5.82 en el caso de HGRi y un 5.73 en el caso de HORi. Finalmente, las respuestas a Q6 y Q7 muestran que los usuarios tienen una percepción bastante positiva acerca del uso del recomendador. En concreto, Q6 obtiene un promedio de 4.85 para HGRi y 5.09 para HORi, y en el caso de Q7 obtiene un promedio de 5.18 para HGRi y 5.36 HORi.

En resumen, las pruebas con usuarios reales de las interfaces web en dos dimensiones propuestas – las cuales integran algoritmos de recomendación basados en sesiones históricas y opiniones – muestran resultados satisfactorios cuando se analiza el comportamiento y percepción de los usuarios. En concreto, mis propuestas reducen los ciclos de recomendación (*ASL*) y muestran un alto nivel de eficacia en la recomendación. Además, las respuestas de

Cuadro 10: Cuestionario para medir la eficiencia en la recomendación

Número Pregunta	Pregunta
Q1	En general, he percibido que el recomendador tomaba en cuenta mis preferencias.
Q2	Me tomó poco tiempo hasta que el sistema me entregara una recomendación adecuada.
Q3	Tengo confianza que el sistema resguarda mi privacidad.
Q4	El uso del sistema ha sido una experiencia agradable.
Q5	Utilizaría este recomendador para comprar productos en el futuro.

Cuadro 11: Cuestionario para medir la eficacia en la recomendación

Número Pregunta	Pregunta
Q1	Me han gustado los productos que me ha recomendado el sistema.
Q2	Creo que el producto finalmente seleccionado ha sido la mejor opción para mí.
Q3	Cada uno de los productos recomendados fue relevante para mí.
Q4	Me tomó poco tiempo hasta que el sistema me entregara una recomendación adecuada.
Q5	Tengo confianza que el sistema resguarda mi privacidad.
Q6	El uso del sistema ha sido una experiencia agradable.
Q7	Utilizaría este recomendador para comprar productos en el futuro.

los usuarios en los cuestionarios, describen una percepción positiva de los usuarios respecto a la eficiencia, eficacia y satisfacción del usuario en el momento de utilizar la interfaz propuesta y los algoritmos de recomendación que ésta integra.

### 5.3.2 Evaluación interfaz colaborativa e-3DVirtualCo

En esta sección presento los resultados que he obtenido de la evaluación de la interfaz virtual inteligente de tres dimensiones que he denominado **e-3DVirtualCo**. En los experimentos realizados, se evalúa la eficiencia y la eficacia del algoritmo de recomendación (CCR) que se ha integrado en la interfaz de usuario. Adicionalmente, los experimentos diseñados, permiten evaluar la percepción de los usuarios respecto a las recomendaciones recibidas y respecto a la satisfacción de los mismos en el uso del entorno virtual en tres dimensiones. Para la evaluación, al igual que en la interfaz de dos dimensiones, se compara la eficiencia y eficacia del algoritmo colaborativo CCR respecto a un algoritmo no colaborativo, como lo es el algoritmo IC.

Mi objetivo en estos experimentos, es demostrar con usuarios reales, el efecto positivo de la colaboración *on-line*, con el propósito de mejorar la experiencia de compra de los usuarios en espacios de comercio electrónico. A continuación describo la metodología aplicada en estos experimentos.

## 5.3.2.1 e-3DVirtualCo: Metodología

En los experimentos he utilizado una evaluación sumativa (Bowman et al., 2002). Los experimentos fueron realizados utilizando la base de datos SMARTPHONE. En los experimentos participaron 20 personas, de diversas características tales como edad, género, habilidades computacionales y experiencia en ambientes virtuales de tres dimensiones (véase el detalle de las características de los participantes en el Cuadro 12).

Cuadro 12: Características de los usuarios reales en las pruebas de e-3DVirtualCo

Rango Edad	#Usuarios	#Hombres	#Mujeres	Experiencia en 3D VW
20-29	11	9	2	Medio-alto
30-39	5	4	1	Medio-bajo
40-49	4	2	2	Bajo

Los experimentos fueron conducidos por un moderador y un observador. Al momento de realizar las pruebas, el moderador introdujo a los usuarios la forma de realizar los experimentos, le entregó el material necesario, tales como la descripción de las tareas a realizar, un formulario de consentimiento para el manejo de datos, y un cuestionario de satisfacción del usuario. El observador cumple la función de tomar notas durante los experimentos. El equipamiento para realizar las pruebas consistió en tres ordenadores: el servidor del entorno virtual en tres dimensiones, y dos ordenadores clientes que se conectaron al servidor.

Cuadro 13: Tareas en los experimentos de e-3DVirtualCo

Tarea	Tipo	Productos Objetivos
Tarea 1	Individual	Con producto objetivo
Tarea 2	Colaborativa	Objetivos similares entre usuarios
Tarea 3	Colaborativa	Objetivos no similares entre usuarios
Tarea 4	Colaborativa	Sin producto objetivo
Tarea 5	Individual	Sin producto objetivo

En los experimentos los usuarios realizaron las siguientes 5 tareas (véase el resumen de tareas en el Cuadro 13):

- **Tarea 1**, es una tarea individual utilizando el algoritmo IC con un producto objetivo definido previamente al usuario;
- **Tarea 2**, es una tarea de recomendación colaborativa, en la cual el moderador le indica a los usuarios que pueden decidir libremente si desean colaborar con otros usuarios (algoritmo CCR) o directamente no colaborar. En esta tarea los usuarios tienen asignados productos objetivos que son similares.
- **Tarea 3**, es una tarea de recomendación colaborativa donde los usuarios no tienen productos objetivos similares. He diseñado las **Tarea 2** y **Tarea 3** con el objetivo de confirmar que la colaboración favorece la eficiencia del recomendador inclusive cuando los productos objetivos no son demasiado similares. Para esto, me baso en los resultados obtenidos con el simulador para niveles de similitud en preferencias

de usuarios menores a 80.0%. Debo aclarar, que tanto para la **Tarea 2** como para la **Tarea 3** los participantes no conocen el producto objetivo del resto de los usuarios. Ellos debieran percibir la similitud o diferencia en las preferencias, cuando visualizan los productos que se recomiendan al otro usuario.

- **Tarea 4**, es una tarea de recomendación colaborativa, donde nuevamente el moderador le indica a los usuarios que son libres de colaborar con otros usuarios si lo desean. En esta tarea los usuarios no tienen un producto objetivo predefinido, y buscan el producto que es de su agrado en su panel de recomendación correspondiente.
- **Tarea 5**, es una tarea de recomendación individual, donde el usuario no tiene un producto objetivo al igual que en la **Tarea 4**. He diseñado la **Tarea 4** y la **Tarea 5** para medir calidad de las decisiones con el recomendador (*decision accuracy*) y confirmar que esta medida se incrementa cuando los usuarios colaboran. Para evaluar la medida *decision accuracy*, he aplicado una metodología similar a la utilizada en la evaluación de **e-WebCo** (véase la descripción del Test de Eficacia en el Cuadro 9) y que también se ha utilizado en estudios previos de recomendadores conversacionales (Chen and Pu, 2006). En concreto, para estos experimentos mido la eficacia de la recomendación por la fracción de participantes que no cambian su selección del producto finalmente escogido en el recomendador, por una mejor opción cuando se les muestra la base de datos completa de productos. Para simplificar la evaluación de los usuarios, al igual que en la evaluación de la interfaz **e-WebCo**, he utilizado el subconjunto de 90 productos de la base de datos SMARTPHONE.

El protocolo empleado en estos experimentos consiste en 4 fases, las cuales se detallan en el Cuadro 14.

Después de realizados los experimentos, el grupo de evaluación recoge los datos obtenidos y los cuestionarios. Posteriormente, se analizan los datos y se extrae la información que concierne a los objetivos de la evaluación. En la siguiente sección describo en detalle el análisis de los resultados obtenidos a través de los experimentos.

#### 5.3.2.2 *e-3DVirtualCo: Resultados de los experimentos con usuarios reales*

La evaluación de la interfaz de recomendación **e-3DVirtualCo** la he realizado en base a los criterios de eficiencia, eficacia y satisfacción del usuario. A continuación, presento los resultados obtenidos de acuerdo a los criterios establecidos.

##### 5.3.2.2.1 **Análisis de eficiencia**

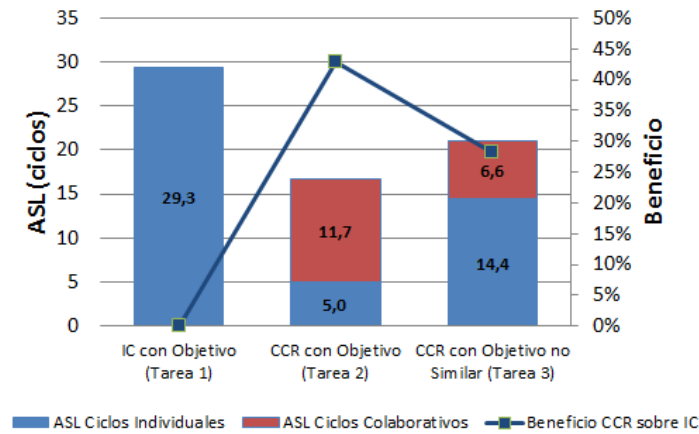
En estos experimentos he utilizado la medida *ASL* para evaluar la eficiencia de la recomendación. El gráfico de barras en la Figura 46 muestra la eficiencia de la recomendación en términos de ciclos de interacción individuales de los usuarios y de ciclos en los cuales los usuarios se mantuvieron colaborando en cada tarea de los experimentos. El gráfico de la Figura 46a muestra todas las tareas en las cuales los usuarios tenían un producto objetivo predefinido (Tarea 1, Tarea 2 y Tarea 3). En el caso de la Figura 46b, ésta muestra las tareas para las cuales no se tenía un producto objetivo predefinido (Tarea 4 y Tarea 5). Para las tareas con producto objetivo predefinido

Cuadro 14: Protocolo de evaluación de e-3DVirtualCo

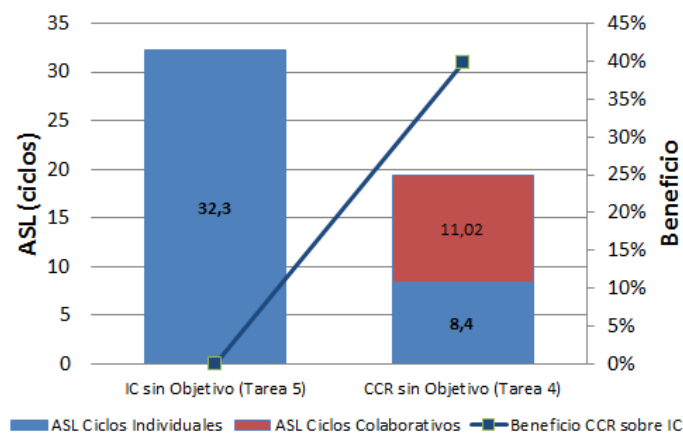
1. *Pre-test*: En esta fase el moderador da la bienvenida a los usuarios, explica brevemente los objetivos del experimento y pregunta a los usuarios acerca de las experiencias que puedan tener con entornos virtuales en tres dimensiones y sistemas de recomendación.
2. *Entrenamiento*: En esta etapa los usuarios realizan las tareas de entrenamiento que consisten en ejecutar varias acciones dentro del entorno virtual en tres dimensiones, tales como buscar un producto usando el recomendador y colaborar con otros usuarios. Esta fase es guiada por el moderador, el cual explica los elementos visuales interactivos del entorno, los elementos de interacción del panel de recomendación y le explica la forma de funcionamiento del panel.
3. *Test*: En esta fase los usuarios, sin la ayuda del moderador, llevan a cabo las 5 tareas descritas previamente. Esta fase fue diseñada como un test *within-subject*, es decir, el mismo grupo de participantes realiza las 5 tareas definidas. Las 5 tareas se rotan para evitar efectos de aprendizaje. En particular, para este experimento los participantes fueron divididos en dos grupos de 10 personas y cada grupo realizó las tareas en orden diferente. En concreto, el primer grupo realizó las tareas en el siguiente orden: Tarea 1, Tarea 2, Tarea 3, Tarea 4 y Tarea 5. Para el segundo grupo, el orden de las tareas cambió de la siguiente forma: Tarea 3, Tarea 2, Tarea 1, Tarea 5, y Tarea 4.
4. *Cuestionario Post-test*: El moderador entrega a los usuarios un cuestionario que mide la percepción del usuario respecto a la eficiencia y eficacia del recomendador y además mide la satisfacción del usuario en el uso del sistema de recomendación. El cuestionario consiste en 10 preguntas que los usuarios deben responder en una escala de siete puntos, donde 1 corresponde a "completamente en desacuerdo" y 7 corresponde a "completamente de acuerdo".

(véase la Figura 46a), la cantidad de ciclos disminuye cuando los usuarios colaboran (Tarea 2 y Tarea 3). Una sesión de recomendación colaborativa con productos objetivo similares (Tarea 2) genera la menor cantidad de ciclos (16.7), mientras que cuando se tienen productos objetivos no similares (Tarea 3), la cantidad de ciclos aumenta a 21.1. Por el contrario, cuando los usuarios trabajan individualmente se incrementa la cantidad de ciclos (algoritmo IC), obteniendo un promedio de 29.3 ciclos. Adicionalmente, al analizar los ciclos colaborativos (ciclos en que los usuarios interactúan colaborativamente en el recomendador) en la Tarea 2 y Tarea 3 de la Figura 46a, se puede concluir que éstos se reducen desde 11.7 ciclos que muestra la Tarea 2 hasta 6.6 ciclos en la Tarea 3. Lo anterior, se podría explicar por el hecho que en la Tarea 3, en la cual los usuarios tenían productos objetivos no similares, los usuarios no encontraron similitudes en los productos que se recomendaban y prefirieron interactuar individualmente con el recomendador en la mayor parte de los ciclos.

Para analizar si las diferencias entre los resultados del algoritmo CCR (Tarea 2 y Tarea 3) en comparación con el algoritmo IC (Tarea 1) son estadísticamente significativas, he aplicado el método estadístico ANOVA (Sheskin, 2003; Sheskin, 2003). En particular, he aplicado ANOVA en las tres tareas (algoritmo individual IC, algoritmo colaborativo CCR usando objetivos similares y algoritmo colaborativo con objetivos no similares),  $k = 3$  con  $k - 1 = 2$  grados de libertad. Los resultados del análisis ANOVA muestran que las diferencias son significativas entre los algoritmos, donde  $p\text{-value} = 3.236e^{-6}$ ,



(a) Tareas con producto objetivo



(b) Tareas sin producto objetivo

Figura 46: Resultados de eficiencia en pruebas de usuarios reales para algoritmo CCR

que es un valor menor que el valor crítico  $\alpha = 0.05$ . Una vez confirmado que las diferencias son significativas entre los algoritmos, he procedido a aplicar el test de Bonferroni (Bland and Altman, 1995), el cual muestra que la eficiencia de CCR en la Tarea 2 y la Tarea 3 son significativamente mejores que IC en la Tarea 1, de forma separada. En concreto, he obtenido un p-value de  $2.4e^{-6}$  entre IC (Tarea 1) y CCR en la Tarea 2. Por otro lado, he obtenido un valor de p-value = 0.00081 entre IC (Tarea 1) y CCR en la Tarea 3.

En la Figura 46b presento los resultados de la Tarea 4 y la Tarea 5. En estas tareas el ASL también disminuye desde 32.3 ciclos para usuarios que interactúan individualmente con el recomendador (algoritmo IC en la Tarea 5) a 19.5 ciclos en la Tarea 4, donde los usuarios colaboran. Es importante destacar que el número de ciclos durante los cuales los usuarios estuvieron colaborando y el número de ciclos individuales están más balanceados en la Tarea 4 que en la Tarea 2 y en la Tarea 3. Esto puede significar que los usuarios inician la sesión de recomendación con una idea más o menos clara, pero a medida que transcurre el proceso de recomendación descubren que obtienen mejores resultados si colaboran con otros usuarios o porque duran-



te la sesión se han desenfocado de su producto objetivo final y un producto de otro usuario les ha llamado la atención.

En la Figura 46b se muestra la comparación solamente de dos algoritmos, por lo tanto en este caso he aplicado el método estadístico T-TEST para evaluar estadísticamente las diferencias entre estos algoritmos. Al aplicar el método T-TEST los resultados indican que las diferencias son significativas entre los algoritmos. En concreto, se obtiene un valor p-value de  $2.5e^{-5}$  (que es un valor menor al valor crítico de  $\alpha = 0.05$ ) entre el algoritmo CCR en la Tarea 4 y el algoritmo IC en la Tarea 5.

Además, los gráficos contenidos en la Figura 46 muestran una línea azul que representa el beneficio en porcentaje del algoritmo CCR en comparación con el algoritmo IC, tanto para las pruebas realizadas con y sin producto objetivo. En la Figura 46a, la recomendación colaborativa alcanza un beneficio sobre IC de 42.9% cuando los usuarios tienen productos objetivos similares. Este beneficio es de 28.2% para usuarios con productos objetivos no similares. Por otro lado, en la Figura 46b, la recomendación colaborativa (Tarea 4) también alcanza un beneficio sobre IC (Tarea 5) de 39.8%. Estos resultados obtenidos en las pruebas con usuarios reales, se encuentran alineados con los obtenidos en las pruebas con el simulador en la Sección 5.2.5.

#### 5.3.2.2.2 Eficacia de la recomendación

Para evaluar la eficacia de la recomendación (*decision accuracy*), he seguido la metodología aplicada en estudios previos de recomendadores conversacionales (Chen and Pu, 2006). En concreto, mido la eficacia de la recomendación por la fracción de participantes que no cambian su selección del producto finalmente escogido en el recomendador, por una mejor opción cuando se les muestra la base de datos completa de productos. En particular, los usuarios realizaron dos tareas que me permitieron evaluar la eficacia del recomendador: la Tarea 4 y la Tarea 5. Los resultados obtenidos para la medida de eficacia se muestran en la Figura 47.

En los experimentos realizados, he obtenido una eficacia del recomendador relativamente más alta en el algoritmo CCR que en el algoritmo IC. Específicamente, se obtuvo un 70% para CCR y un 50% para IC, tal y como lo muestra la Figura 47a. Este resultado significa que en el caso de CCR un 70% de los usuarios mantiene el mismo producto seleccionado al final del proceso de recomendación en el recomendador, cuando ellos pueden revisar la base de datos completa de productos y seleccionar un producto libremente. El restante 30% de los usuarios, cambia el producto del recomendador por uno diferente que está presente en la lista completa de productos. En el caso de IC el 50% de los usuarios cambió de producto.

Además, con el objetivo de corroborar la efectividad de la decisión de **e-3DVirtualCo**, he analizado el comportamiento de los usuarios durante el proceso de recomendación del algoritmo CCR en la Tarea 4. De esta manera, la Figura 47b muestra el porcentaje de ciclos individuales y colaborativos que realizaron en las dos situaciones: aquellos usuarios que mantuvieron su decisión y aquellos que la cambiaron. Específicamente, cuando los usuarios mantuvieron el mismo producto (es decir, aquellos con una calidad de la decisión alta) la mayoría de los ciclos que duró la sesión de recomendación, estuvieron colaborando (65.0% de los ciclos) y solamente un 35.0% de los ciclos interactuaron individualmente con el recomendador. En contraste a esto, cuando los usuarios seleccionaron un producto diferente, ellos estuvieron mayor tiempo interactuando individualmente con el recomendador (75.0% de los ciclos) que colaborando con otros usuarios (25.0% de los ciclos). Estos

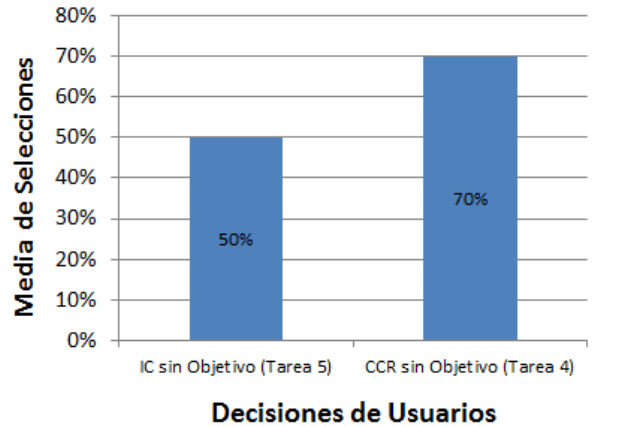
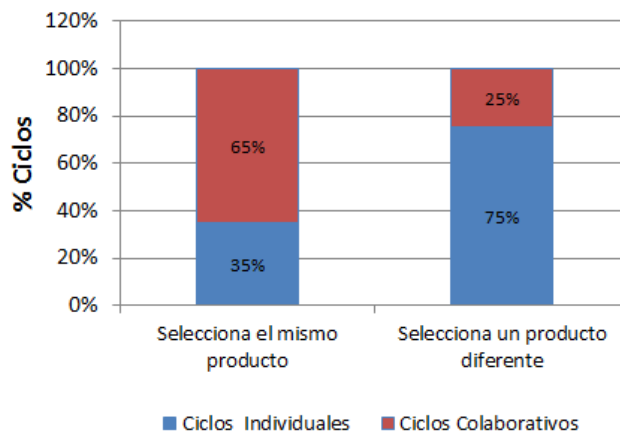
(a) Calidad de la decisión (*Decision Accuracy*)(b) Análisis de las Decisiones (*Decision Analysis*)

Figura 47: Calidad y análisis de las decisiones en las pruebas de usuarios reales de CCR

resultados muestran la efectividad de mi propuesta, la cual ayuda de forma colaborativa a los usuarios para encontrar sus productos.

### 5.3.2.2.3 Satisfacción de los usuarios

Para medir la satisfacción del usuario en el momento de utilizar la interfaz propuesta, he diseñado el cuestionario *post-test* que se presenta en el Cuadro 15. La Figura 48 presenta los datos recogidos de las respuestas de los usuarios en un gráfico de barras, en el cual el promedio de todas las respuestas son mayores que 4.95 en una escala de 7 puntos.

Sobre todo, los participantes encontraron que es fácil aprender a interactuar con el recomendador y con otros usuarios. En este sentido, las respuestas a las preguntas Q1 y Q2 muestran un promedio de 5.95 y 5.60, respectivamente. Además, las respuestas a la pregunta Q9 muestran un alto nivel de satisfacción de los usuarios respecto al uso del recomendador en un entorno virtual en tres dimensiones (5.80).

Cuadro 15: Cuestionario *post-test* para medir la satisfacción de los usuarios

Número Pregunta	Pregunta
Q1	Ha sido fácil aprender a interactuar con el recomendador en el entorno virtual en 3D.
Q2	Ha sido fácil aprender a interactuar con otros usuarios en la interfaz del recomendador en el entorno virtual en 3D.
Q3	Encontré más provechosa y entretenida la búsqueda de un producto, cuando colaboré con otro usuario.
Q4	Siento que me ha tomado menos tiempo encontrar un producto objetivo, cuando he colaborado con otro usuario.
Q5	Encontré útil la interfaz en 3D para colaborar con otros usuarios.
Q6	Creo que los productos que compré en cada sesión fueron los mejores productos para mí.
Q7	Encontré útil la interfaz en 3D para comprar productos con un sistema de recomendación.
Q8	El sistema de recomendación me ayudó a cumplir las tareas, él tiene todas las funciones que yo esperaba.
Q9	Sobre todo, el sistema fue fácil de usar.
Q10	Me gustaría utilizar este sistema para comprar productos en el futuro.

Con respecto a la eficiencia percibida por parte de los usuarios, éstos han evaluado satisfactoriamente este aspecto. Durante el test, los usuarios percibieron que el proceso de recomendación fue más corto cuando ellos colaboraron con otros usuarios, lo cual se refleja en el valor de 4.95 para la pregunta Q4).

En referencia al soporte a la decisión que otorga el recomendador, el promedio de las respuestas a Q8 (5.55) denotan que el recomendador representa una ayuda para los usuarios en el proceso de compra. Principalmente, los usuarios quedaron satisfechos con el producto finalmente comprado, tal y como lo muestra el puntaje alcanzado por Q6 (5.9).

Finalmente, el promedio de 5.60 y 5.5 de las respuestas a las preguntas Q5 y Q7 respectivamente, significan que los usuarios evaluaron positivamente mi propuesta de potenciar la colaboración entre usuarios por medio de la integración de un sistema de recomendación en un entorno virtual en tres dimensiones. Adicionalmente, durante el test, los usuarios se sintieron cómodos cuando colaboraron con otros usuarios (Q3 obtuvo un valor de 5.1). Además, cuando los usuarios fueron consultados por su intención de volver a utilizar el sistema de recomendación para futuras tareas similares a las planteadas en el experimento (Q10), los usuarios respondieron con un promedio de 5.7, lo cual significa que ellos tuvieron una percepción positiva respecto al uso del recomendador.

Resumiendo, la evaluación de la interfaz de usuario en tres dimensiones (**e-3DVirtualCo**) me ha permitido detectar que los usuarios valoran positivamente la usabilidad y facilidades de uso que ofrece la interfaz propuesta. Sin embargo, las conclusiones más importantes de la evaluación tienen que ver con los aspectos colaborativos que se disponen en la nueva interfaz. En este sentido, los usuarios mayoritariamente reconocen que la interfaz de usua-

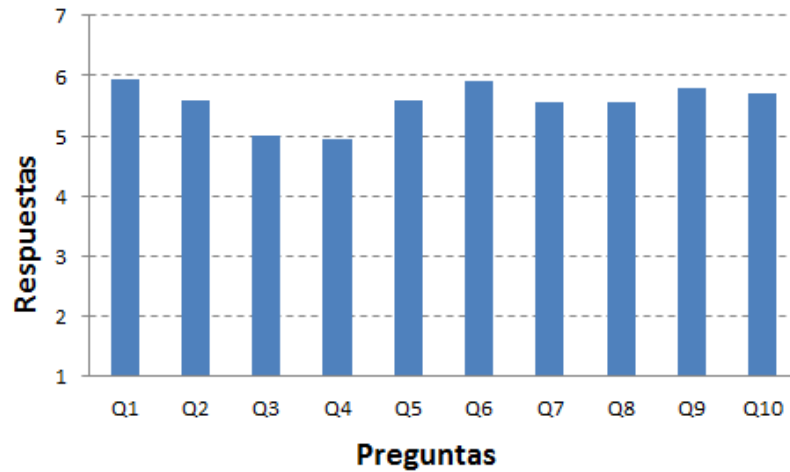


Figura 48: Promedio de los valores de respuestas y desviación estándar para el cuestionario *post-test*

rio y el sistema de recomendación integrado en dicha interfaz favorecen y promueven la colaboración durante la búsqueda de sus productos.

## 5.4 CONCLUSIONES

En este capítulo he presentado la evaluación realizada de las nuevas técnicas de recomendación propuestas en esta tesis y la evaluación realizada a las interfaces de usuario inteligentes que integran las técnicas propuestas.

En primer lugar, he realizado la evaluación de las siguientes técnicas: (1) técnicas que integran información colaborativa *off-line* (opiniones de usuarios acerca de los productos) en las base de datos (*PerIndCB* y *PerTecCB*) y aquellas que integran dicha información en el proceso de recomendación (*LWOpinion* y *UserPerWeight*); (2) técnicas que integran información colaborativa *off-line* proveniente de opiniones de otros usuarios y proveniente de sesiones históricas de recomendación que han realizado otros usuarios (*HOR*, *HORi*, *HGRi*); (3) técnicas que integran información colaborativa *on-line* proveniente de la interacción con otros usuarios (*CCR*).

En general, los resultados han sido bastante satisfactorios. Para las técnicas definidas como *PerIndCB* y *PerTecCB* se obtuvieron beneficios entre un 9.5% y un 15.6%, con respecto al algoritmo base de comparación IC. En este caso, los mejores resultados los obtuvo la técnica *PerIndCB*. Para las técnicas *LWOpinion* y *UserPerWeight* los beneficios obtenidos se encuentran en el orden de 5.5% y 29.7%, siendo la técnica con mejores resultados *UserPerWeight*. En la evaluación de las técnicas *HOR*, *HORi* y *HGRi*, el beneficio se encuentra entre un 18.7% y un 47.3%. En este caso el algoritmo *HORi*, es el que obtiene mejores resultados. Finalmente, al evaluar la técnica colaborativa *CCR*, esta obtiene beneficios que van desde un 40.1% hasta un 88.8% para preferencias de usuarios similares entre un 70.0% y un 95.0%. En resumen, los mejores resultados de los algoritmos que implementan las nuevas técnicas de recomendación, son obtenidos por la técnica *CCR*. Esto me permite concluir que

la colaboración *on-line* entre usuarios favorece la eficiencia de un sistema de recomendación basado en críticas.

En segundo lugar, he evaluado las interfaces que integran las nuevas técnicas de recomendación: **e-WebCo** y **e-3DVirtualCo**. Esta evaluación contempla evaluar la eficiencia y eficacia del recomendador en la interfaz propuesta. Adicionalmente, he evaluado la percepción de los usuarios en el uso del sistema de recomendación por parte de los usuarios.

En el caso de la interfaz **e-WebCo** se han integrado las técnicas *HORi* y *HGRi*. Los resultados obtenidos muestran que las técnicas en esta interfaz alcanzan un beneficio en la eficiencia con respecto al algoritmo IC de 21.34 % para *HGRi* y 29.87 % para *HORi*. En referencia a la eficacia del algoritmo de recomendación, los usuarios perciben que el algoritmo *HGRi* tiene un 52 % de eficacia frente a un 61 % de eficacia de *HORi*. Respecto a la percepción de los usuarios acerca del uso del sistema de recomendación (eficiencia y eficacia medidas en una escala de 1 a 7), los resultados son bastante satisfactorios obteniendo un promedio de puntuación entre 3.35 y un 6.0 en el caso de *HORi* y entre un 3.45 y un 5.82 para *HGRi*.

Para la interfaz **e-3DVirtualCo** que integra la técnica CCR, los resultados son altamente satisfactorios. Específicamente, en referencia a la eficiencia del recomendador, cuando el usuario colabora con otros usuarios que tienen preferencias similares, se obtiene un beneficio de 42.9 %. Este beneficio es un poco menor, cuando los usuarios no tienen preferencias similares, en cuyo caso el beneficio baja a un 39.8 %. Con respecto a la eficacia del recomendador, la técnica CCR obtiene un porcentaje de 68 % que es superior al algoritmo IC, el cual solo alcanza una eficacia del 47 %. Finalmente, con respecto a la percepción de los usuarios en el uso del recomendador dentro del entorno virtual, la puntuación de los usuarios estuvo entre un 4.95 y un 5.95 en una escala de 1 a 7.

En resumen, los resultados obtenidos de la evaluación de las técnicas propuestas y de las interfaces que integran las propuestas, me permiten concluir que la integración de información colaborativa (*off-line* y *on-line*), favorecen la eficiencia, eficacia de los sistemas de recomendación conversacionales, como así mismo las interfaces inteligentes que integran las nuevas técnicas ayudan y mejoran la experiencia de los usuarios durante las sesiones de recomendación.

# 6 | CONCLUSIONES, CONTRIBUCIONES Y TRABAJO FUTURO

EN ESTE CAPÍTULO recojo las conclusiones del trabajo de investigación realizado durante el desarrollo de esta tesis. El capítulo lo separo en tres apartados diferentes: conclusiones (Sección 6.1), contribuciones (Sección 6.2) y trabajo futuro (Sección 6.3).

## 6.1 CONCLUSIONES

En esta tesis he tratado el problema de la personalización de información para usuarios de servicios que reciben grandes volúmenes de datos en entornos de trabajo individuales y colaborativos. Específicamente, me he centrado en la personalización de información por medio de sistemas de recomendación conversacionales basados en críticas. Este tipo de sistemas guían a los usuarios en la búsqueda de un producto, usando un proceso de recomendación basado en un diálogo cíclico, el cual consiste en otorgar sugerencias de productos y recibir retroalimentación de parte del usuario, en forma de críticas focalizadas en características del producto. De esta forma, se produce un refinamiento de las preferencias del usuario en cada ciclo de recomendación.

Para mejorar la personalización de información de los usuarios en los sistemas de recomendación, he propuesto un enfoque basado en aprovechar la naturaleza colaborativa de las personas. En particular, este enfoque se basa en utilizar información colaborativa *off-line* e información colaborativa *on-line* en la definición de nuevas técnicas de recomendación. Además, teniendo en cuenta que en la actualidad existen entornos de visualización que favorecen la interacción y colaboración entre los usuarios, he propuesto integrar las nuevas técnicas de personalización en interfaces de usuario inteligentes bajo diferentes entornos de visualización.

Por lo anterior, la primera etapa de mi trabajo de investigación consistió en definir nuevas estrategias o técnicas de recomendación que integrasen información colaborativa *off-line* y *on-line* (véase el Capítulo 3). La segunda etapa de mi trabajo de investigación consistió en diseñar e implementar nuevas interfaces de usuario inteligentes que integrasen las técnicas de recomendación definidas en la primera etapa (véase el Capítulo 4).

A continuación describo para cada etapa los objetivos que me he planteado en este trabajo de investigación, las propuestas que he desarrollado y los resultados obtenidos.

### 6.1.1 Técnicas de Recomendación

Los objetivos definidos en esta tesis, que se asocian a esta primera etapa se detallan a continuación:

- Definir nuevas técnicas de personalización basadas en las interacciones y retroalimentaciones por parte de los usuarios, y que incluyan también el comportamiento colaborativo del usuario con otras personas.

- Implementar las técnicas de personalización propuestas, por medio de nuevos algoritmos de recomendación.
- Evaluar las técnicas de personalización por medio de un simulador que automatiza el comportamiento de las personas.

Para alcanzar estos objetivos he propuesto nuevas técnicas de recomendación basadas en la naturaleza colaborativa de las personas. Además, las propuestas que he desarrollado buscan integrar los estudios e investigaciones previas que han demostrado resultados satisfactorios y que en su mayoría se encuentran disgregadas. Las técnicas propuestas tienen como objetivo mejorar la eficiencia en la recomendación de productos o servicios, en comparación con las técnicas de recomendación basadas en críticas actualmente disponibles de estudios previos en este ámbito, tales como el algoritmo *Incremental Critiquing* (IC). En particular, he definido el algoritmo IC como la base de comparación para todas mis propuestas en los experimentos de evaluación realizados. En general, para todos los experimentos se ha evaluado la eficiencia de la recomendación de las nuevas técnicas a través del promedio de ciclos de una sesión de recomendación, medida habitualmente conocida por sus siglas en inglés *ASL*.

Las técnicas propuestas las he clasificado de acuerdo a la naturaleza de la información utilizada. Es decir, por un lado he propuesto técnicas de recomendación que integran solamente información colaborativa *off-line* y por otro lado, he propuesto técnicas de recomendación que integran información colaborativa *on-line*. Finalmente, he propuesto una técnica de recomendación que integra ambos tipos de información colaborativa (*off-line* y *on-line*).

En primer lugar, para las técnicas que utilizan información colaborativa *off-line* he integrado información proveniente de las opiniones que otros usuarios han realizado sobre los productos que se recomiendan, e información proveniente de sesiones de recomendación previas que otros usuarios han realizado. Específicamente, las nuevas técnicas de recomendación que he propuesto y los resultados obtenidos, corresponden a:

- **PerTecCB, PerIndCB.** Estas dos nuevas técnicas integran opiniones de los usuarios dentro de la base de datos de productos (Contreras and Salamó, 2015). Específicamente, he integrado las opiniones de los usuarios como nuevas características para cada producto, las cuales he denominado como características perceptivas. Una característica perceptiva describe, a través de las opiniones de los usuarios, un conjunto de características técnicas del producto. El objetivo de estas técnicas es enriquecer la base de descripción de los productos con la incorporación de las opiniones generadas por los usuarios.

Los resultados obtenidos en la evaluación de las técnicas han sido muy satisfactorios y se ha logrado reducir los ciclos de recomendación en comparación con el algoritmo base IC, utilizando una base de datos que solo contiene características técnicas de los productos, obteniendo beneficios por encima de 9.5%. Adicionalmente, de las técnicas propuestas (**PerTecCB** y **PerIndCB**), los mejores resultados los he obtenido al utilizar la técnica **PerIndCB** alcanzando hasta un 15.6% de beneficio. Esta técnica integra opiniones y solo un sub-conjunto de características técnicas denominadas características independientes (es decir, características técnicas que no se asocian a las características perceptivas sobre las cuales los usuarios han manifestado su opinión).

- **LWOpinion, UserPerWeight.** Estas dos técnicas integran opiniones de los usuarios dentro del proceso de recomendación como mecanismo de ponderación de características en la función de similitud (Contreras and Salamó, 2015). En primer lugar, la técnica **LWOpinion** consiste en combinar el peso definido para las características perceptivas, con la medida de las preferencias del usuario obtenidas por el mecanismo de ponderación conocido habitualmente como *Local User Preference Weighting*, *LW*. En segundo lugar, la técnica **UserPerWeight** la he formulado tomando como base el mecanismo *LW*, pero integrando los pesos de las características perceptivas dentro del cálculo de la ponderación de las características del producto.

La utilización de opiniones como un mecanismo de ponderación de las características de los productos ha revelado mejoras significativas en la eficiencia de la recomendación. Estos resultados demuestran que, aparte de los mecanismos de recomendación tradicionales basados en las preferencias de los usuarios, el uso de opiniones de usuarios, como un mecanismo de ponderación de características de los productos, representa una alternativa eficiente dentro del proceso de recomendación. En particular, los mejores resultados se han logrado cuando se integraron técnicas de ponderación previas focalizadas en las preferencias de los usuarios y la ponderación basada en las opiniones de otros usuarios (**UserPerWeight**). Específicamente, la técnica **UserPerWeight** obtuvo un beneficio sobre 16% en comparación con IC y la técnica **LWOpinion** alcanza un beneficio mayor a 4.8%. Lo anterior, me permite concluir que las recomendaciones deben tener en cuenta las opiniones de otros usuarios respecto a los productos a recomendar, pero también deben considerar las preferencias que tiene el usuario respecto a los productos disponibles.

- **HOR.** Esta nueva técnica utiliza información colaborativa proveniente de sesiones de recomendación que otros usuarios realizaron anteriormente y de las opiniones que tienen otros usuarios respecto a los productos disponibles (Contreras and Salamó, 2017). Específicamente, esta técnica consiste en aprovechar el conocimiento colectivo que proviene tanto de las opiniones *off-line* que generan los usuarios acerca de productos, como de las experiencias previas de recomendación de otros usuarios plasmadas en sesiones históricas de recomendación.

Los resultados obtenidos por la técnica **HOR** han sido muy satisfactorios y muestran una significativa reducción de ciclos de recomendación en comparación con IC. En particular, HOR muestra un beneficio promedio sobre IC que alcanza hasta un 43%. Además, HOR también obtiene mejores resultados que la técnica previa HGR, lo cual implica que mi propuesta también es más eficiente que la mejor técnica previa basada en sesiones históricas. Por lo anterior, puedo concluir que el usuario obtiene mejores recomendaciones cuando se tienen en cuenta en conjunto sus preferencias acerca de los productos que se están recomendando, la experiencia previa de otros usuarios durante una sesión de recomendación y las opiniones de otros usuarios respecto a los productos que se recomiendan.

- **HORi, HGRi.** Integración de avances de estudios previos en técnicas de recomendación basadas en sesiones históricas (Contreras and Salamó, 2017). Estos avances previos consisten, en primer lugar, en integrar mecanismos que tengan en cuenta el aprendizaje que realiza el usuario



durante una sesión de recomendación por medio de técnicas de *Reinforcement Learning* que permitan tener en cuenta el momento en que el usuario realiza una crítica. En segundo lugar, he propuesto integrar enfoques de ponderación de características que mejoren el efecto que produce una crítica sobre el producto recomendado al usuario, con el objetivo de reducir la cantidad de ciclos de recomendación. En particular, estos avances se utilizan en una técnica basada en sesiones históricas cuando no es posible encontrar sesiones similares de otros usuarios que permitan recomendar un producto. En este caso, a diferencia de los estudios previos que utilizan el algoritmo IC para recomendar, yo propongo utilizar el algoritmo IC con la integración de los mecanismos mencionados anteriormente.

Los resultados obtenidos de los experimentos realizados demuestran que las técnicas HORi y HGRi reducen considerablemente la cantidad de ciclos de recomendación con respecto a las técnicas HOR y HGR, respectivamente. Del mismo modo, estas técnicas muestran un alto beneficio en comparación con el algoritmo base IC, donde HORi obtiene beneficios de hasta un 45 % y el algoritmo HGRi alcanza beneficios del orden de 40 % en comparación con IC. Por lo anterior, puedo concluir que la integración de sesiones históricas y estudios previos en una nueva técnica de recomendación, ofrece sesiones de recomendación más cortas para los usuarios.

En segundo lugar, he propuesto nuevas técnicas que utilizan información colaborativa *on-line*. En estas técnicas, he integrado información proveniente de la colaboración in situ que se puede producir entre las personas en entornos colaborativos inmersivos. Específicamente, las nuevas técnicas de recomendación que he propuesto, corresponden a:

- **CCR.** La técnica CCR está integrada en un *framework* que permite a los usuarios realizar sesiones de recomendación individuales y colaborativas con otros usuarios, aprovechando de esta forma la naturaleza colaborativa de las personas para facilitar la búsqueda de productos (Contreras et al., 2013; Contreras et al., 2014). En concreto, el algoritmo que implementa la técnica permite a los usuarios iniciar o finalizar colaboraciones con otros usuarios en el momento que encuentren un producto que les agrada y que sea actualmente recomendado a otro usuario. Es necesario mencionar, que esta técnica es la primera aproximación en la literatura que integra un entorno colaborativo con un algoritmo de recomendación basado en críticas.

Los experimentos realizados con la técnica **CCR** han arrojado resultados muy satisfactorios, los cuales reducen notablemente los ciclos de recomendación, siendo significativamente mejores que el algoritmo base IC y que el resto de las técnicas enumeradas en los puntos anteriores. Para ejemplificar, la técnica CCR obtiene beneficios sobre IC que van desde un 20 % en dominios de pocos productos hasta un 88 % en dominios de muchos productos. Lo anterior, me permitió concluir que la interacción y colaboración *on-line* que pueda ocurrir entre los usuarios es muy importante para reducir las sesiones de recomendación de los usuarios. Por lo anterior, puedo deducir que los actuales entornos de visualización colaborativos que permiten la interacción entre usuarios, tales como los entornos virtuales en tres dimensiones, favorecen la incorporación de técnicas de recomendación que integran información colaborativa *on-line* (**CCR**).

- **HOCR.** La técnica **HOCR** integra ambos tipos de información colaborativa (*off-line* y *on-line*). Con esta técnica busco aprovechar las fortalezas de las técnicas basadas en sesiones históricas y opiniones (HOR) y la técnica basada en información colaborativa de usuarios en entornos colaborativos (CCR).

Los experimentos realizados con el algoritmo que implementa la técnica HOCR reduce significativamente los ciclos de recomendación en comparación con el algoritmo IC, obteniendo beneficios entre 70% y 85%. Por lo tanto, concluyo que la información colaborativa que generan las personas, ya sea por medio de opiniones, registro de sesiones de recomendación y por la propia interacción in situ (*on-line*) con el resto de usuarios, mejoran considerablemente la eficiencia de las recomendaciones de productos o servicios.

En resumen, puedo concluir que he cumplido a cabalidad con los objetivos planteados en esta primera etapa de mi trabajo de investigación. Además, puedo concluir que de acuerdo al análisis de los resultados obtenidos en mis experimentos, todas las técnicas que he propuesto en esta tesis incrementan la eficiencia en las recomendaciones de productos en comparación con el algoritmo base IC. Adicionalmente, los resultados obtenidos por las nuevas técnicas son significativamente mejores que el algoritmo IC. Lo anterior, lo he corroborado por métodos estadísticos ampliamente reconocidos, tales como ANOVA test, Friedman test, T-Test, entre otros.

#### 6.1.2 Interfaces de Usuario Inteligentes

Los objetivos definidos en esta tesis, que se asocian a esta segunda etapa se detallan a continuación:

- Integrar las técnicas de personalización con interfaces de usuario inteligentes en espacios de visualización individuales y colaborativos.
- Evaluar las interfaces de usuario inteligentes a través de evaluaciones con usuarios reales.

Para lograr los objetivos planteados, he diseñado e implementado nuevas interfaces de usuario inteligentes, que integran técnicas de recomendación basadas en información colaborativa *off-line* y *on-line*. En concreto, he presentado tres interfaces de usuario inteligentes:

- En primer lugar, he presentado una primera aproximación que otorga al usuario, aparte de las críticas, la posibilidad de seleccionar un producto directamente desde los productos disponibles para recomendar. Además, esta interfaz ofrece al usuario distintas vistas del espacio de búsqueda de productos. En concreto, la interfaz ofrece: (1) un panel de vista actual que permite al usuario ver las características del producto recomendado y realizar críticas sobre ese producto; (2) un panel de vista local que permite al usuario moverse entre productos similares al producto recomendado; (3) una vista global que permite al usuario tener una visión global del espacio de búsqueda y de esta forma adquirir un mayor conocimiento de cómo están distribuidos los productos dentro del espacio de búsqueda (Salamó et al., 2013).
- En segundo lugar, he propuesto un *framework* que implementa una interfaz de usuario bajo un entorno de visualización web en dos dimen-

siones, denominado **e-WebCo** (Contreras and Salamó, 2017). Este *framework* permite integrar técnicas basadas en información colaborativa *off-line*. En particular, para este trabajo de investigación se han integrado las técnicas: IC, HGRi, y HORi. Además, el *framework* es modular, multilingüe y permite implementar distintos dominios de productos (por ejemplo, cámaras fotográficas, teléfonos, restaurantes, etc.).

Es necesario mencionar, que el *framework* **e-WebCo** es la primera interfaz de usuario inteligente que integra técnicas de recomendación basadas en sesiones históricas, ya que los estudios previos de este tipo de técnicas, tales como EBC, HAC y HGR, solo han realizado pruebas con simuladores sin llegar a implementar interfaces de usuario.

- Finalmente, con el objetivo de aprovechar las características colaborativas de los nuevos entornos en tres dimensiones que existen actualmente, he propuesto un *framework* que denomino **e-3DVirtualCo** (Contreras et al., 2014; Contreras et al., 2015c; Contreras et al., 2015b; Contreras et al., 2017). Este *framework* permite el uso de distintos dominios de productos y ofrece un entorno inmersivo en tres dimensiones que permite la interacción y colaboración in situ entre los usuarios conectados al entorno virtual. Adicionalmente, el *framework* permite integrar técnicas basadas en información colaborativa *on-line* y técnicas basadas en información colaborativa *off-line*. En concreto, para este trabajo de investigación se ha integrado la técnica **CCR** y la técnica **HOR**, sin embargo la característica modular y escalable del *framework* permite integrar cualquier otro tipo de recomendador basado en críticas como el algoritmo HOCR. Adicionalmente, en este entorno a diferencia de los entornos de visualización en 2D, pueden conectarse varios usuarios en el mismo momento. De esta forma, los usuarios pueden percibir la presencia de otros usuarios, pueden comunicarse y pueden colaborar durante las sesiones de recomendación.

Debo destacar que el *framework* **e-3DVirtualCo** es el primer estudio que integra un sistema de recomendación conversacional con un entorno virtual colaborativo en tres dimensiones.

La evaluación de las interfaces de usuario inteligentes, la he realizado con usuarios reales y en base a tres aspectos: (1) la eficiencia de la técnica de recomendación integrada en el *framework* propuesto, (2) la eficacia de la técnica de recomendación integrada, y (2) la percepción de los usuarios en el uso de la interfaz de usuario (usabilidad). En general, el análisis de los resultados obtenidos por **e-WebCo** y **e-3DVirtualCo**, ha revelado que estas interfaces favorecen la eficiencia y eficacia de las técnicas de recomendación basadas en información colaborativa.

En particular, en el *framework* **e-WebCo** he integrado los algoritmos IC, HGRi y HORi. En primer lugar, en referencia a la eficiencia de los algoritmos, los resultados han sido muy satisfactorios con beneficios de HGRi y HORi sobre el algoritmo IC de 21,9 % y 29,9 %, respectivamente. Con respecto a la eficacia de la recomendación, HGRi obtiene un 52 % y HORi un 61 %.

Por otro lado, en el caso del *framework* **e-3DVirtualCo** los resultados de eficiencia del algoritmo CCR integrado en el *framework* son notablemente mejores que el algoritmo IC, obteniendo beneficios por sobre el 30 % en comparación con IC. En relación a la eficacia de la recomendación, ésta alcanza un porcentaje de 70 %.

Además, de acuerdo a la percepción de los usuarios en el uso de ambas interfaces, puedo concluir que las interfaces propuestas favorecen la búsqueda

da de productos y que los usuarios valoran en promedio con más de 3.8 y 4.9 puntos en una escala de 7 puntos, cada algoritmo integrado en **e-WebCo** y **e3DVirtualCo**, respectivamente.

En particular, el *framework* **e-3DVirtualCo**, aparte de favorecer la búsqueda de productos por medio de las técnicas de recomendación propuestas, ofrece un entorno de naturaleza colaborativa donde las personas se sienten inmersas y pueden interactuar y colaborar con otras personas. En este sentido, la valoración de los usuarios respecto a la experiencia de uso del *framework* propuesto y respecto intenciones de uso futuro alcanza una puntuación por sobre 5.7 puntos en la misma escala de 7 puntos.

En resumen, puedo concluir que he cumplido a cabalidad con los objetivos planteados en esta segunda etapa de mi trabajo de investigación. Adicionalmente, debo destacar que los *frameworks* propuestos favorecen la integración de técnicas de recomendación basadas en información colaborativa *off-line* y *on-line*. Lo anterior, es corroborado por los resultados obtenidos de los experimentos que he realizado con usuarios reales.

## 6.2 CONTRIBUCIONES

Esta tesis contribuye a mejorar la personalización de información para usuarios que utilizan grandes volúmenes de datos, teniendo como base de todas las propuestas incorporar el comportamiento colaborativo de las personas. En particular la tesis aporta nuevas técnicas de recomendación de productos o servicios basadas en información colaborativa; aporta nuevos algoritmos de recomendación; contribuye a proporcionar nuevas interfaces de usuario inteligentes en diferentes entornos de visualización. A continuación, describo en detalle cada una de las principales contribuciones o aportaciones de esta tesis:

1. Implementación de un algoritmo que utiliza un mecanismo de *parser* basado en DOM (*Document Object Model*) para analizar y extraer características perceptivas desde un sitio de comercio electrónico.
2. Definición de mecanismos para enriquecer las bases de datos de productos, por medio de información colaborativa proveniente de opiniones de las personas acerca de los productos disponibles para recomendar. Técnicas de recomendación **PerTecCB** y **PerIndCB**.
3. Incremento de la eficiencia en las recomendaciones por medio de la integración de opiniones dentro del proceso de recomendación. Técnicas de recomendación **LWOpinion** y **UserPerWeight**.
4. Definición e implementación de una nueva técnica de recomendación denominada **HOR** que integra información colaborativa proveniente de sesiones históricas y de opiniones de otros usuarios (información colaborativa *off-line*).
5. Definición e implementación de nuevas técnicas de recomendación que integran avances previos en algoritmos de recomendación basados en sesiones históricas, denominadas **HGRi** y **HORi**.
6. Definición e implementación de una nueva técnica de recomendación denominada **CCR** basada en información colaborativa *on-line* proveniente de la interacción in situ de los usuarios en entornos colaborativos.

7. Integración conjunta de información colaborativa *off-line* e información colaborativa *on-line* en una nueva técnica de recomendación denominada **HOCR**.
8. Una propuesta de un *framework* en un entorno web en dos dimensiones denominado e-WebCo, el cual integra técnicas de recomendación basadas en información colaborativa *off-line*.
9. Una propuesta de un *framework* en un entorno virtual inteligente en tres dimensiones denominado **e-3DVirtualCo**, el cual permite la integración de técnicas de recomendación basadas en información colaborativa *on-line* y *off-line*.

Adicionalmente, parte de los logros obtenidos a lo largo de la tesis han sido publicados en congresos y en revistas con proyección internacional. Paralelamente, al trabajo de esta tesis y considerando los conocimientos que he adquirido, he apoyado otros trabajos en los cuales he participado como autor y coautor en publicaciones del ámbito de sistemas de recomendación para grupos (Pascual et al., 2014; Contreras et al., 2015a) y de sistemas de recomendación móviles (Horowitz et al., 2016; Horowitz et al., 2017). A continuación se muestra el listado completo de publicaciones:

- Salamó, M., Puig, A., and Contreras, D. (2013). An approach to couple two forms of feedback in conversational recommenders. In *Proceedings of the XVI International Conference of the Catalan Association for Artificial Intelligence*, pages 273–282
- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2013). Integrating a Conversational Recommender System within a 3D Collaborative Environment. In *Proceedings of the XV Conference of Spanish Association for Artificial Intelligence*, pages 89–98
- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2014). An approach to improve user experience with conversational recommenders through a 3D virtual environment. In *Proceedings of the XV International Conference on Human Computer Interaction*, pages 403–406
- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2015b). A 3D visual interface for critiquing-based recommenders: Architecture and interaction. *International Journal of Artificial Intelligence and Interactive Multimedia*, 3(3):pages 7–15
- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2015c). Supporting Users Experience in a 3D eCommerce Environment. In *Proceedings of the XVI International Conference on Human Computer Interaction*, pages 383–386
- Contreras, D. and Salamó, M. (2015). On the Use of User-generated Content in Critiquing Recommendation. In *Proceedings of the XVIII International Conference of the Catalan Association for Artificial Intelligence*, pages (195–204)
- Contreras, D. and Salamó, M. (2017). Integrating History and User Opinion for Conversational Recommendations. *ACM Transactions on Intelligent Systems and Technology*. Manuscript submitted for revision in July 2016. JCR IF=2.414

- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2017). Implementation and Evaluation of a Collaborative Conversational Recommender in a 3D Virtual World. *IEEE Consumer Electronics Magazine. Special Issue Advanced Interaction and Virtual/Augmented Reality*. Manuscript submitted for revision in December 2016
- Pascual, J., Contreras, D., and Salamó, M. (2014). Analysis of a collaborative advisory channel for group recommendation. In *CCIA*, pages 116–125
- Contreras, D., Salamó, M., and Pascual, J. (2015a). A web-based environment to support online and collaborative group recommendation scenarios. *Applied Artificial Intelligence*. JCR IF=0.563, 29(5):480–499
- Horowitz, D., Contreras, D., and Salamó, M. (2016). EventAware: A Context-Aware Tag-based Mobile Recommender System for Events. In *Proceedings of the XIX International Conference of the Catalan Association for Artificial Intelligence*, pages (195–204)
- Horowitz, D., Contreras, D., and Salamó, M. (2017). EventAware: A Mobile Recommender System for Events. *Pattern Recognition Letters*. Manuscript submitted for revision in February 2017. JCR IF=1.586

### 6.3 TRABAJO FUTURO

El trabajo futuro se dirige en cuatro ámbitos principales, de los cuales los dos primeros corresponden a líneas directas de investigación que quedan abiertas del trabajo actual y los dos últimos se refieren a nuevas líneas de investigación que nacen de este trabajo de tesis.

En primer lugar, considero importante definir de nuevos métodos para evaluar sesiones históricas que tengan en cuenta el momento en que el usuario realiza las críticas. En las técnicas basadas en sesiones históricas actuales, se mide la similitud entre una sesión histórica y la sesión actual del usuario contando la cantidad de apariciones de un elemento en ambas sesiones. Sin embargo, estudios previos han demostrado que el usuario adquiere más conocimiento a medida que avanza en la sesión de recomendación. Por esto, considero que se puede evaluar la posibilidad de aplicar técnicas de *Reinforcement Learning* que tienen en cuenta el momento en que se realiza cada crítica. Estas técnicas podrían otorgar mayor importancia a los elementos más cercanos al final de la sesión cuando realice la evaluación. Así, la similitud entre las sesiones sería mucho más cercana a situaciones de la vida real.

En segundo lugar, en esta tesis he aplicado mecanismos de ponderación de características que han obtenido resultados muy satisfactorios. Por lo anterior, creo que la integración de mecanismos de aprendizaje automático para ponderar características o eventualmente para segmentar el espacio de búsqueda de productos, también puede generar buenos resultados en las técnicas de recomendación propuestas que han resultado de esta tesis.

En tercer lugar, un aspecto importante en los procesos de recomendación lo representa la persuasión que puede ejercer el recomendador para que el usuario adquiera un producto. En esta tesis, he definido algunos mecanismos de sugerencias de productos y servicios que son un punto inicial de persuasión al usuario. Estos mecanismos de sugerencias los debo completar y hacer una evaluación profunda con usuarios reales. Con este tipo de evaluación podré refinar los mecanismos de sugerencias que dispongo en estos

momentos o definir nuevos mecanismos de persuasión. Algunos enfoques que pueden ser interesantes de evaluar son:

- Evaluar nuevas formas de comunicación entre las personas y el recomendador (por ejemplo a través de lenguaje natural).
- Integrar en las técnicas de recomendación explicaciones de cómo fue generada la recomendación realizada.
- Incluir en el entorno virtual **e-3DVirtualCo**, agentes (avatares) que interactúen con el usuario como un mecanismo adicional de interacción en los paneles de recomendación actuales, con el objetivo de establecer una relación más cercana a la realidad.
- Integrar en el entorno virtual en tres dimensiones, dentro de lo posible, bases de datos con modelos de los productos en tres dimensiones.

Finalmente, considero importante evaluar nuevos dominios de aplicación para el entorno virtual colaborativo **e-3DVirtualCo**. Actualmente, el *framework* **e-3DVirtualCo** está focalizado en el dominio de comercio electrónico, pero la arquitectura del framework permite ampliarlo a otros dominios. En este sentido, considero importante realizar pruebas en otros dominios que actualmente son de mucho interés, tal y como es el caso de entornos educativos que permitan a los estudiantes interactuar colaborativamente.

# Anexo







## DOCUMENTOS DE APLICACIÓN DE TEST E-WEBCO

Post-Experiment 01 - Tarea con 1 producto objetivo / Post-Test 01 - 1 Target Task

Español
English

Este cuestionario nos entrega tu apreciación respecto a la tarea que acabas de finalizar. Tus respuestas nos ayudarán a definir aspectos del sistema y de nuestra investigación que debemos mejorar o potenciar.

Al momento de contestar cada pregunta, agradeceremos puedas tener en cuenta sólo las acciones que realizaste durante la última tarea realizada.

En cada pregunta debes indicar el grado de acuerdo o desacuerdo con la afirmación, seleccionando tu preferencia. El grado MUY EN DESACUERDO corresponde al número 1 y el grado MUY DE ACUERDO corresponde al número 7.

**Preguntas / Questions**

1- En general, he percibido que el recomendador seguía las preferencias que indiqué durante la tarea

Muy en desacuerdo  1  2  3  4  5  6  7 Muy de acuerdo

2- El uso del sistema es una experiencia agradable.

Muy en desacuerdo  1  2  3  4  5  6  7 Muy de acuerdo

3- El sistema tarda demasiado en entregar recomendaciones adecuadas.

Muy en desacuerdo  1  2  3  4  5  6  7 Muy de acuerdo

4- Confío en que el sistema respeta mi privacidad.

Muy en desacuerdo  1  2  3  4  5  6  7 Muy de acuerdo

5- Volvería a utilizar este Sistema de Recomendación en el futuro, para comprar otros productos.

Muy en desacuerdo  1  2  3  4  5  6  7 Muy de acuerdo

Comentarios adicionales. (Qué te ha gustado, que faltó, ideas o sugerencias).

Figura 49: Cuestionario post-test para medir la eficiencia en la recomendación

**Post-Experimento - Tarea sin producto objetivo / Post-Test - Task Without Target**

Español
English

Este cuestionario nos entrega tu apreciación respecto a la tarea que acabas de finalizar. Tus respuestas nos ayudarán a definir aspectos del sistema y de nuestra investigación que debemos mejorar o potenciar.

Al momento de contestar cada pregunta, agradeceremos puedas tener en cuenta sólo las acciones que realizaste durante la última tarea realizada.

En cada pregunta debes indicar el grado de acuerdo o desacuerdo con la afirmación, seleccionando tu preferencia. El grado MUY EN DESACUERDO corresponde al número 1 y el grado MUY DE ACUERDO corresponde al número 7.

**Preguntas / Questions**

1- Me gustaron los productos recomendados por el sistema.

Muy en desacuerdo  1  2  3  4  5  6  7 Muy de acuerdo

2- Cada producto recomendado resultó relevante para mí.

Muy en desacuerdo  1  2  3  4  5  6  7 Muy de acuerdo

3- Creo que el producto finalmente seleccionado fue la mejor opción que yo podía haber conseguido.

Muy en desacuerdo  1  2  3  4  5  6  7 Muy de acuerdo

4- El uso del sistema es una experiencia agradable.

Muy en desacuerdo  1  2  3  4  5  6  7 Muy de acuerdo

5- El sistema tarda demasiado en entregar recomendaciones adecuadas.

Muy en desacuerdo  1  2  3  4  5  6  7 Muy de acuerdo

6- Confío en que el sistema respeta mi privacidad.

Muy en desacuerdo  1  2  3  4  5  6  7 Muy de acuerdo

7- Volvería a utilizar este sistema de recomendación en el futuro para comprar otros productos.

Muy en desacuerdo  1  2  3  4  5  6  7 Muy de acuerdo

Comentarios adicionales. (Qué te ha gustado, que faltó, ideas o sugerencias).

Figura 50: Cuestionario post-test para medir la eficacia en la recomendación

## B

DOCUMENTOS DE APLICACIÓN  
DE TEST E-3DVIRTUALCO**ENCUESTA DE SATISFACCIÓN**

*Nombre:*.....

Este cuestionario nos entrega tu apreciación respecto al sistema que acabas de utilizar. Tus respuestas nos ayudarán respecto a que aspectos del sistema y de nuestra investigación debemos mejorar o potenciar.

Agradeceremos puedas tener en cuenta todas las acciones que realizaste durante la tarea, al momento de contestar cada pregunta.

En cada pregunta debes indicar el grado de acuerdo o desacuerdo con la afirmación, encerrando en un círculo tu preferencia. El grado MUY EN DESACUERDO corresponde al número 1 y el grado MUY DE ACUERDO corresponde al número 7.

1.- Ha sido fácil aprender a interactuar con el recomendador en el entorno virtual en 3D.

Muy en Desacuerdo	1	2	3	4	5	6	7	Muy de Acuerdo
----------------------	---	---	---	---	---	---	---	-------------------

2.- Ha sido fácil aprender a interactuar con otros usuarios en la interfaz del recomendador en el entorno virtual en 3D.

Muy en Desacuerdo	1	2	3	4	5	6	7	Muy de Acuerdo
----------------------	---	---	---	---	---	---	---	-------------------

3.- Encontré más provechosa y entretenida la búsqueda de un producto, cuando colaboré con otro usuario.

Muy en Desacuerdo	1	2	3	4	5	6	7	Muy de Acuerdo
----------------------	---	---	---	---	---	---	---	-------------------

4.- Siento que me ha tomado menos tiempo encontrar un producto objetivo, cuando he colaborado con otro usuario.

Muy en Desacuerdo	1	2	3	4	5	6	7	Muy de Acuerdo
----------------------	---	---	---	---	---	---	---	-------------------

5.- Me ha resultado útil la interfaz del entorno 3D para colaborar con otros usuarios y completar la tarea.

Muy en Desacuerdo	1	2	3	4	5	6	7	Muy de Acuerdo
----------------------	---	---	---	---	---	---	---	-------------------

Figura 51: Cuestionario post-test para medir la satisfacción de los usuarios

6.- Creo que los productos que compré en cada sesión fueron los mejores productos para mí.

Muy en Desacuerdo    1    2    3    4    5    6    7    Muy de Acuerdo

7.- Encontré útil la interfaz en 3D para comprar productos con un sistema de recomendación.

Muy en Desacuerdo    1    2    3    4    5    6    7    Muy de Acuerdo

8.- El sistema de recomendación me ayudó a cumplir las tareas, él tiene todas las funciones que yo esperaba.

Muy en Desacuerdo    1    2    3    4    5    6    7    Muy de Acuerdo

9.- En general, el sistema fue fácil de usar.

Muy en Desacuerdo    1    2    3    4    5    6    7    Muy de Acuerdo

10.- Me gustaría utilizar este sistema para comprar productos en el futuro.

Muy en Desacuerdo    1    2    3    4    5    6    7    Muy de Acuerdo

Comentarios adicionales (qué te ha gustado, qué has echado en falta, ideas o sugerencias...):

Figura 52: Cuestionario post-test para medir la satisfacción de los usuarios

## BIBLIOGRAFÍA

- Adomavicius, G. and Tuzhilin, A. (2005). 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):734–749.
- Agrawal, R., Imieliński, T., and Swami, A. (1993). Mining association rules between sets of items in large databases. *SIGMOD Rec.*, 22(2):207–216.
- Agrawal, R., Srikant, R., et al. (1994). Fast algorithms for mining association rules. In *Proceedings 20th International Conferences very large data bases, VLDB*, volume 1215, pages 487–499.
- Ahn, H., Kim, K., and Han, I. (2006). Mobile advertisement recommender system using collaborative filtering: Mar-cf.
- Arroyo, A., Serradilla, F., and Calvo, O. (2010). Modeling spatial-temporal context information in virtual worlds. In *Proceedings of the 23rd International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems - Volume Part I*, pages 437–447. Springer-Verlag.
- Averjanova, O., Ricci, F., and Nguyen, Q. N. (2008). Map-based interaction with a conversational mobile recommender system. In *Mobile Ubiquitous Computing, Systems, Services and Technologies, 2008. UBICOMM'08. The Second International Conference on*, pages 212–218. IEEE.
- Aylett, R. and Luck, M. (2001). Intelligent virtual environments.
- Bainbridge, W. S. (2010). Virtual worlds as cultural models. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 1(1):1–21.
- Bell, G., Parisi, A., and Pesce, M. (1995). The virtual reality modeling language.
- Bland, J. M. and Altman, D. G. (1995). Multiple significance tests: the bonferroni method. *Bmj*, 310(6973):170.
- Bonis, B., Stamos, J., Vosinakis, S., Andreou, I., and Panayiotopoulos, T. (2009). A platform for virtual museums with personalized content. *Multi-media tools and applications*, pages 139–159.
- Bowman, D., Gabbard, J., and Hix, D. (2002). A Survey of Usability Evaluation in Virtual Environments : Classification and Comparison of Methods 1 Introduction and motivation 2 Distinctive characteristics of VE evaluation. *Presence: Teleoperators and Virtual Environments*, 11(4):404–424.
- Bridge, D., Göker, M. H., McGinty, L., and Smyth, B. (2005). Case-based recommender systems. *Knowledge Engineering Review*, 20(3):315–320.
- Bridge, D. G. (2002). Towards conversational recommender systems: A dialogue grammar approach. In *ECCBR Workshops*, pages 9–22.
- Burke, R. (2000). A Case-Based Reasoning Approach to Collaborative Filtering. In Blanzieri, E. and Portinale, L., editors, *Proceedings of the Fifth European Conference on Case-Based Reasoning, EWCBR '00*, pages 370–379. Springer. Trento, Italy.
- Burke, R. (2002a). Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12(4):331–370.

- Burke, R. (2002b). Interactive critiquing for catalog navigation in e-commerce. *Artificial Intelligence Review*, 18(3-4):245–267.
- Burke, R. D., Hammond, K. J., and Yound, B. (1997). The findme approach to assisted browsing. *IEEE Expert*, 12(4):32–40.
- Burnkrant, R. E. and Cousineau, A. (1975). Informational and normative social influence in buyer behavior. *Journal of Consumer Research*, pages 206–215.
- Butnaru, C. (2009). Social psychology and marketing: The consumption game. understanding marketing and consumer behavior through game theory. *Review of Economic and Business Studies (REBS)*, pages 165–185.
- Chen, L. and Pu, P. (2006). Evaluating critiquing-based recommender agents. In *Proceedings of the National Conference on Artificial Intelligence*, volume 21, pages 157–162.
- Chen, L. and Pu, P. (2007a). The evaluation of a hybrid critiquing system with preference-based recommendations organization. In *Proceedings of the 2007 ACM conference on Recommender systems*, pages 169–172. ACM.
- Chen, L. and Pu, P. (2007b). Preference-based organization interfaces: aiding user critiques in recommender systems. In *User Modeling 2007*, pages 77–86. Springer.
- Chen, L. and Pu, P. (2008). A cross-cultural user evaluation of product recommender interfaces. In *Proceedings of the 2008 ACM conference on Recommender systems*, pages 75–82. ACM.
- Chen, L. and Pu, P. (2009). Interaction design guidelines on critiquing-based recommender systems. *User Modeling and User-Adapted Interaction*, 19(3):167–206.
- Chen, L. and Pu, P. (2012). Critiquing-based recommenders: survey and emerging trends. *User Modeling and User-Adapted Interaction*, 22(1-2):125–150.
- Chen, Y., Ferrer, X., Wiratunga, N., and Plaza, E. (2014). Sentiment and preference guided social recommendation. In *Case-Based Reasoning Research and Development*, volume 8765 of *Lecture Notes in Computer Science*, pages 79–94. Springer International Publishing.
- Chen, Y. Y., Ferrer, X., Wiratunga, N., and Plaza, E. (2015). Aspect selection for social recommender systems. In *Case-Based Reasoning Research and Development*, pages 60–72. Springer.
- Contreras, D. and Salamó, M. (2015). On the Use of User-generated Content in Critiquing Recommendation. In *Proceedings of the XVIII International Conference of the Catalan Association for Artificial Intelligence*, pages (195–204).
- Contreras, D. and Salamó, M. (2017). Integrating History and User Opinion for Conversational Recommendations. *ACM Transactions on Intelligent Systems and Technology*. Manuscript submitted for revision in July 2016. JCR IF=2.414.
- Contreras, D., Salamó, M., and Pascual, J. (2015a). A web-based environment to support online and collaborative group recommendation scenarios. *Applied Artificial Intelligence*. JCR IF=0.563, 29(5):480–499.

- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2013). Integrating a Conversational Recommender System within a 3D Collaborative Environment. In *Proceedings of the XV Conference of Spanish Association for Artificial Intelligence*, pages 89–98.
- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2014). An approach to improve user experience with conversational recommenders through a 3D virtual environment. In *Proceedings of the XV International Conference on Human Computer Interaction*, pages 403–406.
- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2015b). A 3D visual interface for critiquing-based recommenders: Architecture and interaction. *International Journal of Artificial Intelligence and Interactive Multimedia*, 3(3):pages 7–15.
- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2015c). Supporting Users Experience in a 3D eCommerce Environment. In *Proceedings of the XVI International Conference on Human Computer Interaction*, pages 383–386.
- Contreras, D., Salamó, M., Rodríguez, I., and Puig, A. (2017). Implementation and Evaluation of a Collaborative Conversational Recommender in a 3D Virtual World. *IEEE Consumer Electronics Magazine. Special Issue Advanced Interaction and Virtual/Augmented Reality*. Manuscript submitted for revision in December 2016.
- Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30.
- Dong, R., O'Mahony, M., Schaal, M., McCarthy, K., and Smyth, B. (2013a). Sentimental product recommendation. In *Proceedings of the 7th ACM Conference on Recommender Systems*, pages 411–414. ACM.
- Dong, R., Schaal, M., O'Mahony, M., McCarthy, K., and Smyth, B. (2013b). Mining features and sentiment from review experiences. In *Case-Based Reasoning Research and Development*, pages 59–73. Springer.
- Dong, R., Schaal, M., O'Mahony, M., McCarthy, K., and Smyth, B. (2013c). Opinionated Product Recommendation. In *Case-Based Reasoning Research and Development*, volume 7969 of LNCS, pages 44–58. Springer.
- Elahi, M., Ricci, F., and Rubens, N. (2014). Active learning strategies for rating elicitation in collaborative filtering: A system-wide perspective. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(1):13:1–13:33.
- Fisher, R. A. (1956). Statistical methods and scientific inference.
- Friedman, M. (1937). The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association*, pages 675–701.
- Friedman, M. (1940). A comparison of alternative tests of significance for the problem of  $m$  rankings. *The Annals of Mathematical Statistics*, 11(1):86–92.
- Guo, G. and Elgendi, M. (2013). A New Recommender System for 3D E-Commerce: An EEG Based Approach. *Journal of Advanced Management Science*, 1(1).
- Häubl, G. and Trifts, V. (2000). Consumer decision making in online shopping environments: The effects of interactive decision aids. *Marketing science*, 19(1):4–21.



- Holzwarth, M., Janiszewski, C., and Neumann, M. M. (2006). The influence of avatars on online consumer shopping behavior. *Journal of marketing*, 70(4):19–36.
- Horowitz, D., Contreras, D., and Salamó, M. (2016). EventAware: A Context-Aware Tag-based Mobile Recommender System for Events. In *Proceedings of the XIX International Conference of the Catalan Association for Artificial Intelligence*, pages (195–204).
- Horowitz, D., Contreras, D., and Salamó, M. (2017). EventAware: A Mobile Recommender System for Events. *Pattern Recognition Letters*. Manuscript submitted for revision in February 2017. JCR IF=1.586.
- Horozov, T., Narasimhan, N., and Vasudevan, V. (2006). Using location for personalized poi recommendations in mobile environments. In *International Symposium on Applications and the Internet (SAINT'06)*, pages 6–pp. IEEE.
- Hu, X. and Wang, K. (2010). Personalized Recommendation for Virtual Reality. *2010 International Conference on Multimedia Technology*, pages 1–5.
- Jameson, A., Baldes, S., and Kleinbauer, T. (2004). Two methods for enhancing mutual awareness in a group recommender system. In *Proceedings of the working conference on Advanced visual interfaces*, pages 447–449. ACM.
- Kaelbling, L. P., Littman, M. L., and Moore, A. W. (1996). Reinforcement learning: A survey. *Journal of artificial intelligence research*, 4:237–285.
- Kim, C. Y., Lee, J. K., Cho, Y. H., and Kim, D. H. (2004). Viscors: A visual-content recommender for the mobile web. *IEEE Intelligent Systems*, 19(6):32–39.
- Konstan, J. A. and Riedl, J. (2012). Recommender systems: from algorithms to user experience. *User Modeling and User-Adapted Interaction*, 22(1-2):101–123.
- Koren, Y. and Bell, R. (2011). Advances in collaborative filtering. In *Recommender Systems Handbook*, pages 145–186. Springer.
- Lau, H., Kan, C., and Lau, K. (2013). How consumers shop in virtual reality? how it works? *Advances in Economics and Business*, 1(1):28–38.
- Lops, P., De Gemmis, M., and Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. In *Recommender Systems Handbook*, pages 73–105. Springer.
- Luc, D. T. (2008). *Pareto Optimality*, pages 481–515. Springer New York, New York, NY.
- Mandl, M. and Felfernig, A. (2012). Improving the performance of unit critiquing. In *User Modeling, Adaptation, and Personalization*, volume 7379, pages 176–187. Springer.
- McCarthy, K., McGinty, L., Smyth, B., and Reilly, J. (2005a). On the evaluation of dynamic critiquing: A large-scale user study. In *AAAI*, pages 535–540.
- McCarthy, K., McGinty, L., Smyth, B., and Salamó, M. (2006a). Social interaction in the cats group recommender. In *Workshop on the social navigation and community based adaptation technologies*.

- McCarthy, K., Reilly, J., McGinty, L., and Smyth, B. (2004). On the dynamic generation of compound critiques in conversational recommender systems. In *Adaptive Hypermedia and Adaptive Web-Based Systems*, volume 3137 of *Lecture Notes in Computer Science*, pages 176–184. Springer.
- McCarthy, K., Reilly, J., McGinty, L., and Smyth, B. (2005b). Experiments in dynamic critiquing. In *Proceedings of the 10th international conference on Intelligent user interfaces*, pages 175–182. ACM.
- McCarthy, K., Salamó, M., Coyle, L., McGinty, L., Smyth, B., and Nixon, P. (2006b). Cats: A synchronous approach to collaborative group recommendation. In *Proceedings of the FLAIRS 2006 Conference*, volume 2006, pages 86–91.
- McCarthy, K., Salamó, M., Coyle, L., McGinty, L., Smyth, B., and Nixon, P. (2006c). Group recommender systems: A critiquing based approach. In *Proceedings of the 11th International Conference on Intelligent User Interfaces, IUI '06*, pages 267–269. ACM Press.
- McCarthy, K., Salem, Y., and Smyth, B. (2010). Experience-based critiquing: Reusing critiquing experiences to improve conversational recommendation. In *Proceedings of the International Conference on Case Base Reasoning*, pages 480–494. Springer.
- McGinty, L. and Reilly, J. (2011). On the evolution of critiquing recommenders. In *Recommender Systems Handbook*, pages 419–453. Springer.
- McSherry, D. and Aha, D. W. (2007). The Ins and Outs of Critiquing. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 962–967.
- Messinger, P. R., Stroulia, E., Lyons, K., Bone, M., Niu, R. H., Smirnov, K., and Perelgut, S. (2009). Virtual worlds—past, present, and future: New directions in social computing. *Decision Support Systems*, 47(3):204–228.
- Moghaddam, S. and Ester, M. (2010). Opinion digger: an unsupervised opinion miner from unstructured product reviews. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1825–1828. ACM.
- Muhammad, K., Lawlor, A., Rafter, R., and Smyth, B. (2015). Great explanations: Opinionated explanations for recommendations. In *Proceedings 23rd International Conference on Case Base Reasoning*, pages 244–258. Springer International Publishing.
- Nielsen (2014). Nielsen study. <http://www.nielsen.com/uk/en.html>.
- Oberlander, J., Karakatsiotis, G., Isard, A., and Androutsopoulos, I. (2008). Building an adaptive museum gallery in second life. *Proceedings of Museums and the Web, Montreal, Quebec, Canada*.
- Pascual, J., Contreras, D., and Salamó, M. (2014). Analysis of a collaborative advisory channel for group recommendation. In *CCIA*, pages 116–125.
- Pazzani, M. J. and Billsus, D. (2007). *The Adaptive Web: Methods and Strategies of Web Personalization*, chapter Content-Based Recommendation Systems, pages 325–341.

- Pu, P. and Chen, L. (2006). Trust building with explanation interfaces. In *Proceedings of International Conference on Intelligent User Interfaces (IUI)*.
- Pu, P., Chen, L., and Kumar, P. (2008). Evaluating product search and recommender systems for e-commerce environments. *Electronic Commerce Research*, 8(1-2):1–27.
- Pu, P. and Faltings, B. (2000). Enriching buyers' experiences: the smartclient approach. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 289–296. ACM.
- Pu, P. and Faltings, B. (2004). Decision Tradeoff Using Example-Critiquing and Constraint Programming. *Constraints*, 9(4):289–310.
- Pu, P., Faltings, B., Chen, L., Zhang, J., and Viappiani, P. (2011). Usability guidelines for product recommenders based on example critiquing research. In *Recommender Systems Handbook*. Springer.
- Reilly, J., McCarthy, K., McGinty, L., and Smyth, B. (2004). Dynamic critiquing. In *Advances in Case-Based Reasoning*, volume 3155 of *Lecture Notes in Computer Science*, pages 763–777. Springer.
- Reilly, J., McCarthy, K., McGinty, L., and Smyth, B. (2005a). Incremental critiquing. *Knowledge-Based Systems*, 18(4-5):143–151.
- Reilly, J., Smyth, B., McGinty, L., and McCarthy, K. (2005b). Critiquing with confidence. In *Case-Based Reasoning Research and Development*, volume 3620 of *Lecture Notes in Computer Science*, pages 436–450. Springer.
- Reilly, J., Zhang, J., McGinty, L., Pu, P., and Smyth, B. (2007). A comparison of two compound critiquing systems. In *Proceedings of the 12th International Conference on Intelligent User Interfaces*, pages 317–320, USA. ACM.
- Ricci, F. and Nguyen, Q. N. (2005). Critique-based mobile recommender systems. *OEGAI Journal*, pages 1–7.
- Ricci, F. and Nguyen, Q. N. (2007). Acquiring and Revising Preferences in a Critique-Based Mobile Recommender System. *IEEE Intelligent Systems*, 22(3):22–29.
- Ricci, F., Rokach, L., Shapira, B., and Kantor, P. B., editors (2011). *Recommender Systems Handbook*. Springer.
- Salamó, M. and Escalera, S. (2012). Increasing retrieval quality in conversational recommenders. *IEEE Transactions on Knowledge and Data Engineering*, 24(10):1–14.
- Salamó, M., McCarthy, K., and Smyth, B. (2012). Generating recommendations for consensus negotiation in group personalization services. *Personal Ubiquitous Computing*, 16(5):597–610.
- Salamó, M., Puig, A., and Contreras, D. (2013). An approach to couple two forms of feedback in conversational recommenders. In *Proceedings of the XVI International Conference of the Catalan Association for Artificial Intelligence*, pages 273–282.
- Salamó, M., Reilly, J., McGinty, L., and Smyth, B. (2005). Improving Incremental Critiquing. In *Proceedings of the 16th Artificial Intelligence and Cognitive Science*, pages 379–388.

- Salamó, M., S., and Radeva, P. (2009). Quality enhancement based on reinforcement learning and feature weighting for a critiquing-based recommender. In *Case-Based Reasoning Research and Development, 8th International Conference on Case-Based Reasoning*, pages 298–312.
- Salamó, M., Smyth, B., and McCarthy, K. (2005). Reducing critiquing repetition in conversational recommendation. In *Proceedings IX Workshop on Multi-agent Information Retrieval and Recommender Systems at the International Joint Conference on Artificial Intelligence*, pages 55–61.
- Salem, Y. and Hong, J. (2013). History-aware critiquing-based conversational recommendation. In *Proceedings of the 22Nd International Conference on WWW Companion*, pages 63–64, Switzerland.
- Salem, Y., Hong, J., and Liu, W. (2014). History-guided conversational recommendation. In *Proceedings of the 23rd International Conference on WWW Companion*, pages 999–1004.
- Schifanella, R., Panisson, A., Gena, C., and Ruffo, G. (2008). Mobhinter: epidemic collaborative filtering and self-organization in mobile ad-hoc networks. In *Proceedings of the 2008 ACM conference on Recommender systems*, pages 27–34. ACM.
- Shah, F., Bell, P., and Sukthankar, G. (2010). A destination recommendation system for virtual worlds. In *FLAIRS Conference*. AAAI Press.
- Shani, G. and Gunawardana, A. (2011). Evaluating recommendation systems. In *Recommender Systems Handbook*, pages 257–297. Springer.
- Shearin, S. and Lieberman, H. (2001). Intelligent profiling by example. *Proceedings of the 6th international conference on Intelligent user interfaces - IUI '01*, pages 145–151.
- Sheskin, D. J. (2003). *Handbook of parametric and nonparametric statistical procedures*. crc Press.
- Sullivan, J., Sullivan, J. W., Tyler, S. W., et al. (1994). *Intelligent user interfaces*.
- Sussman, R. W. and Cloninger, C. R. (2011). *Origins of Altruism and Cooperation*. Springer.
- Trattner, C., Parra, D., Eberhard, L., and Wen, X. (2014). Who will trade with whom?: Predicting buyer-seller interactions in online trading platforms through social networks. In *Proceedings of the 23rd International Conference on World Wide Web*, pages 387–388, New York, NY, USA. ACM.
- Unger, M., Bar, A., Shapira, B., and Rokach, L. (2016). Towards latent context-aware recommendation systems. *Knowledge-Based Systems*, 104:165–178.
- Vachon, F. (2011). Can online decision aids support non-cognitive web shopping approaches? *International Journal of Business and Management*, pages 16–27.
- Van Setten, M., Pokraev, S., and Koolwaaij, J. (2004). Context-aware recommendations in the mobile tourist application compass. In *International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, pages 235–244. Springer.

- Vasudevan, S. R. and Chakraborti, S. (2014). Mining user trails in critiquing based recommenders. In *Proceedings of the 23rd International Conference on World Wide Web, WWW '14 Companion*, pages 777–780, New York, NY, USA. ACM.
- Viappiani, P., Faltings, B., and Pu, P. (2006a). Evaluating preference-based search tools: A tale of two approaches. In *Proceedings of the 21st National Conference on Artificial Intelligence - Volume 1, AAAI'06*, pages 205–211. AAAI Press.
- Viappiani, P., Faltings, B., and Pu, P. (2006b). Preference-based search using example-critiquing with suggestions. *Journal Artificial Intelligence Research*, 27:465–503.
- Xie, H., Chen, L., and Wang, F. (2014). Collaborative compound critiquing. In *User Modeling, Adaptation, and Personalization*, volume 8538, pages 254–265. Springer.
- Xu, B. and Yu, Y. (2010). A Personalized Assistant in 3D Virtual Shopping Environment. *2010 Second International Conference on Intelligent Human-Machine Systems and Cybernetics*, pages 266–269.
- Zhang, J., Jones, N., and Pu, P. (2008). A Visual Interface for Critiquing-based Recommender Systems. *Proceedings of the 9th ACM conference on Electronic commerce*, pages 230–239.
- Zhang, J. and Pu, P. (2006). A comparative study of compound critique generation in conversational recommender systems. In *Adaptive Hypermedia and Adaptive Web-Based Systems*, volume 4018 of *Lecture Notes in Computer Science*, pages 234–243. Springer.