
Impacto de los factores y organizaciones sociales en los procesos de recomendación para grupos



Memoria de Proyecto Fin de Master en Sistemas
Inteligentes

Lara Quijano Sánchez

Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática
Universidad Complutense de Madrid

Dirigido por:

Belén Díaz Agudo

Codirigido por:

Juan A. Recio García

Curso 2009-2010

Documento maquetado con T_EX^S v.1.0+.

Impacto de los factores y organizaciones sociales en los procesos de recomendación para grupos

Memoria que presenta el proyecto fin de Master
Ingeniería del Software e Inteligencia Artificial

Versión 1.0+

**Departamento de Ingeniería del Software e Inteligencia
Artificial**

**Facultad de Informática
Universidad Complutense de Madrid**

**Dirigido por:
Belén Díaz Agudo
Codirigido por:
Juan A. Recio García**

Curso 2009-2010

Copyright © Lara Quijano Sánchez

Agradecimientos

Quiero agradecer el esfuerzo, el trabajo y el tiempo dedicado a mis directores de proyecto Belén Díaz Agudo y Juan A. Recio García. Muchas gracias por haberme ayudado y guiado a lo largo de todo este proyecto.

También quiero dar las gracias a todos mis amigos y a mi familia por todo el tiempo que han gastado en contestar a todas los test y debates a los que les he sometido y por haberme servido de “conejiillos de indias” en los experimentos realizados para este proyecto.

Muchas gracias a todos!!

Resumen

There is a proliferation of recommender systems that cope with the challenge of addressing recommendations for groups of users in different domains. The work presented in here involves the improvement of current group recommendation techniques by introducing three novel factors: the personality of every individual in the group, the structure of the group itself and the experience obtained through previous recommendations. In this project we present our theories for making recommendations to groups of people connected through social network structures. Our method consists on making recommendations to groups using existing techniques of collaborative filtering, taking into account the group personality composition and the social connections between the individuals of the group. We have performed a case study with real users in the movie recommendation domain where we identify the set of social factors that influence in the trust between users and how they impact in the recommendation process. We have also studied the different group personality compositions, and how its distribution, the size of the group or the trust among users can influence in the recommendation. We have analyzed this factors not only with real data but also with synthetic data. To be able to evaluate the reaction of the users towards the recommendations and to avoid future repetitions of the recommendations we have developed a system with memory where we store the solutions proposed by the recommender and we make an analysis of the users happiness through time. Finally we did a variation of our recommender system with distributed models and argumentation using JCOLIBRI and *D²ISCO*.

KeyWords: Group Recommendation, Personality, Trust, Social Network, Experience, Distributed Models and Argumentation.

Actualmente hay una proliferación de sistemas recomendadores que realizan recomendaciones para grupos de usuarios en diferentes dominios. En este trabajo se propone una mejora de las técnicas de recomendación para grupos existentes a través de la inclusión de tres nuevos factores: la personalidad de cada individuo del grupo, la estructura del grupo en sí y la experiencia obtenida con recomendaciones previas. El proyecto expone nuestras teorías sobre recomendaciones a grupos de personas conectadas a través de redes sociales. Nuestro método consiste en la realización de recomendaciones a grupos usando técnicas existentes de filtrado colaborativo, teniendo en cuenta la composición de la personalidad del grupo y las conexiones sociales entre sus miembros. Hemos realizado un caso de estudio tanto con usuarios reales como con datos sintéticos en el dominio de las películas. En él hemos identificado la colección de factores sociales que influyen en la confianza entre individuos y cómo pueden tener una repercusión en el proceso de recomendación. También hemos realizado estudios sobre la influencia en las recomendaciones de las diferentes composiciones de personalidad que puede tener un grupo, el tamaño y la distribución de la confianza entre los componentes. Para evaluar la reacción de los usuarios hacia las recomendaciones y evitar repeticiones en los productos recomendados hemos realizado un sistema con memoria en el que se almacenan las soluciones propuestas por el recomendador y se realiza un análisis de la felicidad de los usuarios en el tiempo. Finalmente hemos realizado una variación de nuestro recomendador usando modelos distribuidos y argumentación utilizando JCOLIBRI y *D²ISCO*.

Palabras Clave: Recomendaciones a grupos, Personalidad, Confianza, Redes Sociales, Experiencia, Modelos distribuidos y Argumentación.

Índice

Agradecimientos	v
Resumen	vii
1. Introducción	1
1.1. Objetivos del proyecto	2
1.2. Estructura de la memoria	3
En el próximo capítulo	4
2. Estado del Arte	5
2.1. Sistemas Recomendadores	5
2.1.1. Sistemas de Recomendación para individuos	5
2.1.2. Sistemas Recomendadores para grupos	8
2.1.3. Implementación de Sistemas Recomendadores	11
2.2. Factores sociales en los sistemas de recomendación	15
2.2.1. Redes Sociales	15
2.2.2. Modelos de Confianza	17
En el próximo capítulo	18
3. Propuesta de Trabajo: GRUPITO	19
3.1. Introducción a GRUPITO (Group Recommendations Using Personality Investigation and Trust Organizations)	19
3.2. Recomendaciones basadas en personalidad	21
3.3. Recomendaciones basadas en factores Sociales	24
3.4. Inclusión de los factores de personalidad y modelado social en las recomendaciones para grupos	26
3.4.1. Ratings basados en la personalidad	27
3.4.2. Ratings basados en la delegación	27
3.4.3. Ratings basados en la influencia	28
3.5. Recomendaciones con memoria	28
3.6. Modelos distribuidos y argumentación	32

3.6.1. Configuración de recomendadores para grupos con modelos distribuidos y argumentación	34
3.7. Reutilización de la metodología presentada a través de plantillas de jCOLIBRI	39
En el próximo capítulo	42
4. Caso de estudio: Recomendador para películas	43
4.1. Introducción	43
4.2. Experimento con datos sintéticos	43
4.2.1. Resultados del experimento	47
4.3. Experimento con datos reales	51
4.3.1. Resultados del experimento	56
4.4. Experimento con datos reales y memoria	58
4.4.1. Resultados del experimento	60
En el próximo capítulo	62
5. Conclusiones y Trabajo Futuro	63
5.1. Conclusiones	63
5.2. Trabajo futuro	65
Bibliografía	67

Índice de figuras

2.1. Ejemplos de sistemas CBR en jCOLIBRI	13
2.2. Ejemplos de recomendadores individuales en jCOLIBRI	14
3.1. Arquitectura del modelo teórico	20
3.2. Ejemplo de preguntas realizadas en el test TKI	22
3.3. Tipos de personalidad TKI	23
3.4. Ciclo CBR	29
3.5. Ejemplo del proceso de recomendación con memoria	32
3.6. Estructura de red social	33
3.7. a) Topología "todos con todos" b) Topología en forma de red social	34
3.8. Propagación de la Query	35
3.9. Argumentación	35
3.10. Primer paso: configuración de los agentes	36
3.11. Segundo paso: configuración de las queries y el recomendador individual	37
3.12. Comparación de los resultados obtenidos utilizando los tres métodos propuestos.	38
3.13. Plantilla general para la construcción de sistemas recomendadores grupales	40
3.14. Plantilla específica para la construcción de nuestro método de sistemas recomendadores grupales	41
4.1. Resultados Globales	47
4.2. Resultados con la estrategia Least Misery en función del tamaño del grupo.	48
4.3. Resultados con la estrategia Least Misery en función de la distribución de la personalidad.	48
4.4. Resultados con la estrategia Minimizing Penalization en función del tamaño del grupo.	49
4.5. Resultados con la estrategia Minimizing Penalization en función de la distribución de la personalidad.	49

4.6. Resultados con la estrategia Average Satisfaction en función del tamaño del grupo.	50
4.7. Resultados con la estrategia Average Satisfaction en función de la distribución de la personalidad.	50
4.8. Muestra del test de 50 películas que rellenaron nuestros usuarios en un evento de Facebook.	52
4.9. Muestra de la cartelera de cine que valoraron nuestros usuarios.	53
4.10. Tasa de aciertos para los 4 sistemas recomendadores con nuestros dos enfoques	57
4.11. Tasa de aciertos para cada tamaño de grupo con los Ratings basados en la delegación	58
4.12. Tasa de aciertos para cada tamaño de grupo con los Ratings basados en la influencia	59
4.13. Importancia de los factores de confianza	60
4.14. Recomendaciones a lo largo del tiempo	61
4.15. Felicidad a lo largo del tiempo	61

Índice de Tablas

3.1. Ejemplo de coeficientes CMW de un usuario	23
3.2. Coeficientes usados para calcular el CMW	24

Capítulo 1

Introducción

Los sistemas de recomendación para grupos de usuarios son actualmente una línea de investigación muy prometedora dentro del área de recomendadores. Los sistemas de recomendación clásicos (i.e: individuales) han demostrado su importancia y repercusión dentro de la industria al ser uno de los pilares centrales del comercio on-line. Por ejemplo, sitios web tan populares como Amazon¹ utilizan estas técnicas para guiar al usuario en la adquisición de productos. Una vez que este tipo de tecnología está alcanzando cierta madurez, se presenta el objetivo de ampliar dichas técnicas a la recomendación de productos a un determinado grupo de usuarios. Las posibilidades de esta nueva línea de investigación son muy amplias y cuentan con dominios de aplicación inmediatos, principalmente recomendaciones de ocio para grupos de personas: teatro, cine, restaurantes, música, etc (McCarthy y Anagnost, 1998; Masthoff y Gatt, 2006; O'Connor et al., 2001). Otro elemento que hace muy interesante dicha línea de investigación es su relación y aplicabilidad en las redes sociales omnipresentes en la web, ya que los recomendadores grupales pueden utilizar las redes sociales como plataforma de ejecución y/o aprovechar la información implícita existente en estas redes sobre las relaciones entre usuarios para mejorar su rendimiento.

En nuestra aproximación existen dos factores que destacan dentro de la estructura de un grupo, el papel individual de cada componente del grupo, que depende de la personalidad de cada individuo y las relaciones entre los componentes del grupo. En este proyecto describimos una propuesta para realizar recomendaciones a grupos conectados a través de redes sociales. En él, se propone una arquitectura CBR distribuida, donde la recomendación final se verá influenciada por la personalidad de cada componente y la forma en la que están conectados a través de sus relaciones sociales, básicamente de amistad, que se definen en la red social de la que forman parte. Proponemos este tipo de arquitectura basándonos en que los sistemas recomendadores, que son el elemento principal de estudio en este trabajo, son una de las ramas

¹<http://www.amazon.com/>

de mayor aplicación de los sistemas CBR. Por tanto, nuestra aproximación consiste en realizar recomendaciones a grupos usando técnicas existentes de filtrado colaborativo (Konstan et al., 1997), teniendo en cuenta la composición de la personalidad del grupo y las conexiones sociales entre los miembros individuales del grupo. Probaremos este método en el dominio de las recomendaciones de películas. Aunque hemos realizado nuestros experimentos en este dominio en particular, nuestro método se podría trasladar fácilmente a otro dominio en el que los productos a recomendar estén valorados con ratings por los usuarios, ya sea música, restaurantes, viajes, etc.

Para realizar este proyecto partimos de resultados previos preliminares del grupo GAIA² incluidos en su plataforma jCOLIBRI (Recio-García et al., 2008). Cabe destacar que el desarrollo de este tipo de técnicas conlleva una complejidad mucho mayor que los sistemas de recomendación clásicos ya que se deben coordinar y maximizar las preferencias de los distintos usuarios objeto de la recomendación.

El principal mecanismo para alcanzar este objetivo es la colaboración entre los usuarios involucrados. Obviamente en un sistema software no serán los usuarios quienes colaboren para llegar a una decisión sino que cada uno llevará asociado un agente con conocimiento sobre sus preferencias y que realizará la tarea en cuestión. De esta forma, el usuario sólo tiene que reflejar sus preferencias en el agente que le representa dentro del proceso de negociación (qué película ver, qué restaurante ir a cenar, etc). El agente será el encargado de realizar el proceso de elección del producto, liberando al usuario de dicha tarea. Este sistema multi-agente consiste en una arquitectura de sistemas colaborativos y de argumentación como la que se utiliza en *D²ISCO* (González-Sanz et al., 2009), herramienta implementada con jCOLIBRI en la cual nos basaremos para implementar nuestras teorías.

Por lo tanto, el objetivo de este proyecto es realizar una investigación sobre los sistemas de recomendación grupal existentes y su mejora mediante la adición de dos componentes totalmente novedosas en el área: las relaciones sociales del usuario y su perfil de personalidad. Además se ha guardado un historial de todas las recomendaciones realizadas a cada grupo para poder realizar recomendaciones basadas en casos previos, es decir tener en cuenta la experiencia del recomendador, esta técnica está basada en la de los recomendadores basados en casos.

1.1. Objetivos del proyecto

Los objetivos concretos del proyecto se desglosan en los siguientes puntos:

- Estudiar y clasificar en un marco teórico las técnicas de recomendación

²Group for Artificial Intelligence Applications - <http://gaia.fdi.ucm.es/>

para grupos.

- Estudiar y desarrollar técnicas de recomendación para grupos que exploren las posibilidades de las redes sociales.
- Implementar el marco teórico obtenido y ejemplificar la extensión realizada con un sistema de recomendación grupal que utilice las técnicas desarrolladas.
- Validar nuestra aproximación y comprobar la mejora al usar este método sobre datos reales tomados de redes sociales.
- Explorar técnicas de agrupar a los usuarios en los diferentes subconjuntos posibles que se proponen en el proceso de recomendación. Y explorar los diferentes resultados al variar la estructura de los grupos y la presencia de distintos tipos de personalidades en ellos.
- Explorar técnicas de almacenamiento de recomendaciones previas. Y crear un sistema con memoria que asegure un nivel homogéneo de aceptación de las recomendaciones en todos los usuarios de un grupo.
- Estudiar el valor de la experiencia en las recomendaciones realizadas.
- Incorporar los métodos a JCOLIBRI, realizar una plantilla para recomendadores grupales según nuestro método que permita la reutilización de los resultados de este trabajo.

1.2. Estructura de la memoria

Esta memoria está organizada de la forma que se expone a continuación. Primero se describe el marco teórico de las recomendaciones en general y de las recomendaciones para grupos en particular. Una vez se ha expuesto en qué consisten las recomendaciones grupales y un análisis general de los trabajos previos relacionados con el tema de la personalidad, de los componentes de un grupo, y de las redes sociales, se realiza un estudio sobre cómo mejorarlos. Finalmente se expone un caso de estudio donde se aplican las nuevas técnicas estudiadas y una evaluación sobre la eficiencia de éstas. Cada uno de los temas aquí expuestos se estructuran en los capítulos del siguiente modo:

Capítulo 2. Estado del arte. Este capítulo presenta una visión general de los sistemas recomendadores individuales y para grupos. Después se explican las herramientas disponibles para desarrollar dichos recomendadores en JCOLIBRI y una introducción a *D²ISCO* cuya función detallada se explicará en los próximos capítulos. Por último se explica la importancia de las

redes sociales en los últimos años y cómo de ellas se puede extraer información útil para los sistemas recomendadores para grupos.

Capítulo 3. Propuesta de Trabajo. En este capítulo se explica nuestro método de recomendación para grupos, teniendo en cuenta la personalidad de cada individuo y las relaciones sociales entre los componentes del grupo. También se explica cómo realizar recomendaciones satisfactorias para todos los usuarios a lo largo del tiempo, presentando un recomendador con memoria. Luego se expone la idea de cómo trasladar el recomendador utilizado a una arquitectura de modelos distribuidos y de argumentación basada en *D²ISCO*. Y finalmente se explican los pasos que se deben seguir para reutilizar nuestro método ya sea con otros datos o con otros dominios siguiendo a través de una plantilla.

Capítulo 4. Caso de estudio: Recomendador para películas. En este capítulo detallamos 3 experimentos distintos que hemos realizado en el dominio de la recomendación de películas. El primero con datos sintéticos en el que analizamos la influencia que tienen el tamaño y la distribución de personalidad y las relaciones sociales entre los miembros del grupo. Un segundo experimento con datos reales en el que demostramos el funcionamiento de nuestro método de confianza basado en redes sociales. Y un tercer experimento en el que probamos las ventajas de utilizar un recomendador con memoria.

Capítulo 5. Conclusiones y Trabajo Futuro. Finalmente exponemos las conclusiones que hemos obtenido en la realización de este proyecto y comentamos las líneas de investigación que seguiremos en el futuro.

Resumen

En este capítulo hemos introducido el ámbito de este proyecto y las líneas generales que hemos tomado. Así como los objetivos que se pretendían lograr y la estructuración de esta memoria. A continuación explicaremos los conceptos en los que se basa este trabajo, las técnicas básicas de recomendación, las herramientas que existen para implementar recomendadores, en qué consisten las redes sociales y cómo se pueden utilizar para obtener modelos de confianza.

Capítulo 2

Estado del Arte

2.1. Sistemas Recomendadores

Los sistemas recomendadores nacen con el propósito de facilitar la toma de decisiones en temas/dominios en los que las posibilidades de elección son muchas y variadas. Actúan sugiriéndonos buenos productos y/o servicios bien sea para comprar algo o para consumir. En la actualidad podemos encontrar recomendadores para todo tipo de productos: viajes, libros, películas, restaurantes, coches y un largo etcétera. Sin duda, un campo de gran aplicación de estos sistemas son las actividades relacionadas con el ocio. Algunas de estas actividades son típicamente realizadas en grupo en vez de en forma individual, luego tiene sentido realizar no sólo recomendaciones individuales sino recomendaciones a todo un grupo de personas que vaya a realizar alguna actividad conjuntamente. De ahí que existan estos dos tipos de recomendadores.

A continuación explicamos en qué consisten los sistemas recomendadores, los tipos que hay, sus diferencias y las herramientas que existen para implementarlos.

2.1.1. Sistemas de Recomendación para individuos

Existen dos grandes familias de recomendadores para individuos en base a la fuente de conocimiento: *basados en contenido*, aquellos que realizan la recomendación en base a la descripción de los elementos (Pazzani y Billsus, 2007), y *colaborativos* aquellos que utilizan información de los usuarios junto con las valoraciones que los usuarios hacen de los elementos (Bridge et al., 2005). Entre estos dos extremos, existen también múltiples aproximaciones híbridas que mezclan ingredientes de recomendación colaborativa y por contenidos.

2.1.1.1. Sistemas Recomendadores colaborativos

Los recomendadores colaborativos son aquellos que no necesitan información sobre las características del producto, ya que utilizan en su lugar las valoraciones de otros usuarios a dichos productos. Los recomendadores colaborativos normalmente se basan en el coeficiente de Pearson para realizar las estimaciones en la posible valoración de un producto por parte de un usuario. Por lo tanto, su mayor inconveniente radica en la necesidad de adquirir valoraciones para los elementos de los usuarios. Para resolver este problema, frecuentemente se utilizan las técnicas *Long tail* (Anderson, 2007), que se refieren a la propiedad estadística de que una gran parte de la población se mantiene en una cola de distribución de probabilidad como la observada en una distribución de Gauss. Esta técnica ha sido utilizada por Amazon¹ y Netflix² entre otras aplicaciones comerciales. El número de valoraciones necesarias se reduce si se cuenta con perfiles de usuario, en cuyo caso el recomendador utilizará esta información para hacer recomendaciones al usuario actual en función de lo que otros usuarios con un perfil similar hayan valorado. Esto permite hacer recomendaciones al usuario actual de elementos que no están en su perfil pero sí en el de usuarios parecidos a él. En contraposición a los recomendadores colaborativos, encontramos los recomendadores basados en contenidos, que pasamos a describir a continuación.

2.1.1.2. Sistemas Recomendadores basados en contenidos

Los sistemas recomendadores basados en contenido utilizan descripciones de los productos a recomendar y proporcionan un conjunto de soluciones formado por los productos cuyas descripciones más se ajusten a la consulta que realice el usuario. Los recomendadores *basados en casos* son un tipo especial de recomendadores basados en contenidos en los que cada caso que se recupera representa una recomendación anterior de un producto similar al que el usuario desea. Tienen un origen común con las técnicas de razonamiento basado en casos (CBR, del inglés Case-based Reasoning). Los sistemas CBR (Leake, 1996) cuentan con una base de casos que se compone de problemas resueltos anteriores, junto con la solución tomada. De este modo los nuevos problemas se resuelven adaptando soluciones pasadas, usadas para resolver problemas similares. Este tipo de recomendadores se pueden clasificar atendiendo a las siguientes características (nótese que algunas de estas características también pueden encontrarse en los recomendadores colaborativos):

- Quién toma la iniciativa. Podemos distinguir dos características en base a quién lleva la iniciativa en la recomendación. Así, podemos tener un

¹<http://www.amazon.com/>

²<http://www.netflix.com/>

recomendador reactivo, donde es el usuario quien lleva la iniciativa realizando una consulta al sistema. Por otro lado está el recomendador proactivo, donde el que lleva la iniciativa es el recomendador, realizando una propuesta inicial al usuario basada en el historial pasado del usuario, o en valoraciones asociadas a los elementos, o en cualquier otra estrategia previamente seleccionada.

- **Dinámica de recomendación.** También aquí distinguimos dos características: *single-shot* y *conversacional*. *Single-shot* son aquellos en los que sólo se muestra un conjunto de elementos recomendados al usuario y este tiene la oportunidad de elegir uno o descartarlos. Si la recomendación no agradara al usuario, este debería empezar de nuevo para obtener nuevos elementos. Los recomendadores *conversacionales* son aquellos en los que la recomendación se entiende como un proceso iterativo, en el que el usuario puede ir refinando sus requisitos hasta obtener un elemento adecuado para él. Existen dos estrategias de conversación. *Navegación-por-propuesta* y *navegación-por-pregunta* (Shimazu, 2001, 2002). En la primera, un conjunto de elementos es mostrado al usuario, a partir del cual éste podrá refinar sus requisitos. En la *navegación-por-preguntas* el sistema recoge los requisitos del usuario a partir de un conjunto de preguntas cuidadosamente seleccionadas.
- **Capacidad de personalización.** Es decir, si en el proceso de recuperación intervienen o no las características/preferencias/necesidades del usuario. Los perfiles de usuario pueden contener información sobre el historial de navegación, las preferencias, las necesidades del usuario, o lo que se crea conveniente. La capacidad de personalización está relacionada con cómo se maneja toda esta información, es decir, si es tenida en cuenta o no a la hora de realizar las recomendaciones. También hay que tener en cuenta las técnicas utilizadas para mantener actualizados los perfiles y la persistencia de los mismos.
- **Determinación de la calidad de los elementos.** Recomendadores que siguen una aproximación tradicional de la similitud entre elementos, o recomendadores que apuestan por innovar en la similitud introduciendo una medida de calidad en los elementos (McGinty y Smyth, 2003). Esta medida está relacionada con la diversidad de los elementos recuperados en la recomendación. Se define la diversidad del conjunto recuperado como la disimilitud existente entre cada par de elementos del conjunto. Es decir, un elemento mejorará su calidad cuanto más disimilar sea a los ya recuperados, siempre y cuando siga manteniendo la similitud con la consulta.

A continuación explicamos las diferencias entre los recomendadores individuales y los que se estudian en este proyecto, los recomendadores grupales.

2.1.2. Sistemas Recomendadores para grupos

Los sistemas de recomendación se han ocupado tradicionalmente de hacer recomendaciones de elementos a usuarios individuales. Ha sido recientemente cuando se ha empezado a trabajar en el desarrollo de técnicas que permitan proponer recomendaciones a grupos de usuarios simultáneamente (Jameson y Smyth, 2007). Este tipo de sistemas plantea problemas específicos a las técnicas de recomendación, como son la necesidad de adquirir las preferencias del grupo, ayudar al grupo en el proceso de toma de decisiones de cuál es la mejor opción y explicar al grupo las razones de una recomendación. La mayoría de los recomendadores de grupos utilizan métodos casi indistinguibles de los métodos de adquisición de información que se aplican en los sistemas recomendadores para individuos. Básicamente se pueden dividir en:

- Adquisición de preferencias sin especificación explícita. Muchos sistemas recomendadores no requieren que sus usuarios especifiquen explícitamente sus preferencias. El sistema puede funcionar con información adquirida implícitamente sobre los usuarios. Un ejemplo de ello es Let's Browse (Lieberman et al., 1999) que se trata de un sistema que recomienda páginas web a grupos de dos o más personas que están navegando en la red juntos.
- Especificación de preferencias explícita. Otros tipos de recomendadores para grupos, sí que requieren una especificación explícita de las preferencias de los usuarios. Por ejemplo en Travel Decision Forum (Jameson et al., 2004) que se trata de un sistema que ayuda a un grupo de usuarios a acordar unas determinadas características de unas vacaciones que planeen juntos. Otro sistema recomendador menos explícito sobre la especificación de preferencias es PolyLens (O'Connor et al., 2001) se trata de un sistema que recomienda películas a grupos de usuarios. Es una extensión del sistema MovieLens (Schafer et al., 2007), basado en técnicas de filtrado colaborativo, donde los usuarios no describen explícitamente sus preferencias en el dominio de las películas, sino que puntúan las películas individualmente en una escala de 1 a 5 estrellas. Este sistema hace recomendaciones a grupos agregando las preferencias de los miembros del grupo con la estrategia de minimizar el malestar, basándose en la hipótesis de que la felicidad del grupo será igual a la del menos feliz de sus miembros.

Una vez que el sistema ha adquirido el conocimiento necesario sobre los usuarios, puede adaptar la especificación de preferencias a los requerimientos de la recomendación para el grupo. Ejemplos de esta adaptación serían los sistemas que se centran en las preferencias negativas, como en Adaptive Radio (Chao et al., 2005) o los sistemas que comparten información sobre las preferencias especificadas, como podemos ver en Travel Decision Forum

(Jameson et al., 2004) o en CATS (McCarthy et al., 2006). Los primeros sólo tienen sentido si el procedimiento que se está utilizando para generar las recomendaciones es diseñado principalmente para evitar que el producto seleccionado sea especialmente contrario a los gustos de cualquier miembro del grupo.

En cuanto a los otros sistemas, se puede dar el caso que en un sistema recomendador para grupos interese que cada miembro pueda conocer las preferencias de los otros miembros del grupo. Por ejemplo para aprender de otros miembros del grupo, para ahorrar tiempo a la hora de especificar sus propias preferencias, para asimilar los motivos de los demás componentes y así llegar a un consenso más fácilmente o para poder prever actitudes y comportamientos de los otros componentes del grupo. Fácilmente podemos ver cómo de este enfoque surge un problema de manipulación. Por ejemplo si una persona no quiere que salga un producto en particular, podría calificarlo como “odiado” y asegurarse así que nunca salga recomendado, esta situación ocurría en MusicFX (McCarthy y Anagnost, 1998).

Dependiendo del tamaño y la homogeneidad del grupo, puede ser difícil encontrar una recomendación que sea adecuada para cada miembro del grupo de manera individual (Jameson, 2004). En la mayoría de los casos, el recomendador debe escoger aquella opción que satisfaga al mayor número de usuarios del grupo, de acuerdo con sus preferencias individuales. Por tanto, se necesita algún tipo de método de agregación para combinar la información sobre las preferencias individuales de los usuarios de forma que el sistema pueda obtener la recomendación idónea para el grupo en sí. Existen tres aproximaciones básicas para resolver este problema: (a) mezclar las recomendaciones que se harían por separado a cada uno de los miembros del grupo. Es un método simple de agregación donde se unen las soluciones del sistema recomendador para cada componente del grupo en una única lista. Este método fue considerado por PolyLens (O’Connor et al., 2001); (b) agregar las valoraciones para cada usuario. Para cada producto candidato y por cada miembro del grupo, el sistema predice cómo ese componente evaluaría dicho producto, y devuelve una colección de candidatos que tengan las valoraciones previstas más altas. Un ejemplo de esto es el Pocket RestaurantFinder (McCarthy, 2002); y (c) construir un modelo de las preferencias del grupo. En este enfoque el sistema usa la información sobre las preferencias individuales de los componentes del grupo para construir un modelo de preferencias para el grupo en sí. Let’s Browse (Lieberman et al., 1999) es un ejemplo de este tipo de sistemas.

2.1.2.1. Funciones de agregación

Una vez se ha elegido un enfoque general, hay que decidir qué procedimiento computacional se va a usar para la agregación. Hay varios objetivos

que deben tenerse en cuenta a la hora de elegir uno, como la satisfacción total, la comprensibilidad, el grado de igualdad, esto variará dependiendo de la situación dada:

- Maximizar la satisfacción media. En esta función de agregación se computa una media de la satisfacción predicha para cada miembro del grupo y se usa como base de la selección de productos candidatos. La función a utilizar sería:

$$R_i = \text{average}(r_{ij}) = 1/n * \sum_{k=1}^n r_{ij} \quad (2.1)$$

Donde r_{ij} es la valoración de cada usuario $u_i, i \in [1..n]$, para un producto p_j . R_i es el rating final que obtiene el producto p_j para el grupo.

- Minimizar la miseria. Aunque la satisfacción media sea alta, si una solución deja a un componente del grupo especialmente disconforme, se puede considerar como una situación indeseada. En PolyLens (O'Connor et al., 2001) la minimización de la miseria es el único criterio que se aplica. Es posible tener en cuenta este factor como una restricción que debe cumplir cada solución, por ejemplo, considerando que el rating más bajo debe ser siempre superior a un cierto umbral.
- Minimizar la penalización, asegurar un grado de justicia. En esta función se considera una solución que satisfaga a cada persona de igual modo. Esta situación es en general preferible a una solución que satisfaga a un cierto número de componentes a expensas de otros. La función a utilizar sería:

$$R_i = \text{average}(r_{ij}) - \omega * \text{desviaciónEstandar}(r_{ij}) \quad (2.2)$$

Donde r_{ij} es la valoración estimada para cada usuario u_i para un producto p_j , ω es el peso que refleja la importancia del grado de justicia y R_i es el rating final que obtiene el producto p_j para el grupo.

Las estrategias de agregación más populares han sido criticadas porque combinan las valoraciones de los usuarios siempre de la misma forma, sin tener en cuenta las interacciones concretas entre los miembros de un grupo determinado (Chen et al., 2008). Para resolver este problema se han propuesto, por ejemplo, el uso de algoritmos genéticos para determinar el peso óptimo de las valoraciones individuales, aunque esto tiene el inconveniente de que se ha de disponer de valoraciones previas del grupo para otros productos.

En algunos trabajos recientes se han comenzado a incluir consideraciones sociales en la relación de los miembros del grupo a la hora de hacer las recomendaciones (Masthoff y Gatt, 2006). Se ha utilizado la idea de combinar la satisfacción individual con el contagio emocional aplicada a la recomendación de videoclips a un grupo de usuarios. Se considera que la elección

del mejor clip a escuchar a continuación viene determinada en parte por el último clip seleccionado anteriormente. Esta evolución se computa como una función de satisfacción que tiene en cuenta el estado afectivo del individuo. El estado de un individuo influye a su vez en el estado afectivo de los demás miembros del grupo, produciendo así el contagio afectivo que se toma en consideración durante el proceso de recomendación.

Otro factor que proporciona una variación en la agregación de preferencias es la influencia de los distintos perfiles de personalidad de cada usuario involucrado en el proceso (Recio-García et al., 2009). En este trabajo como veremos más adelante, se realizan recomendaciones a grupos, teniendo en cuenta las diferentes personalidades de los miembros del grupo. Donde en la recomendación final, las preferencias de cada individuo tendrán diferentes pesos dependiendo de la manera en que cada miembro del grupo reaccionaría en una situación conflictiva.

Una vez el sistema realiza la recomendación es natural pensar que los componentes del grupo a recomendar deseen saber en cierto modo cómo se llegó a la recomendación, y en particular cuán atractiva es dicha recomendación para cada uno de ellos como individuos. Por ello muchos sistemas recomendadores acompañan cada solución con un análisis de la aceptabilidad predicha. Un ejemplo de este tipo de sistemas es Let's Browse (Lieberman et al., 1999). Esto puede variar de un simple índice de la confianza del sistema a una visualización compleja de los pros y contras de una solución. A la hora de tomar la decisión final no existe ninguna garantía de que se vayan a tomar ninguna de las recomendaciones realizadas, no importa cuán apropiada o convincente sean las recomendaciones o la explicación del sistema. Con las recomendaciones a grupos se prevén debates extensivos y negociaciones. Para ello algunos sistemas tienden a no proporcionar una única decisión final, sino un medio para argumentar entre los miembros del grupo y llegar a un consenso como se explica en Jameson et al. (2004). Veremos más adelante que este tipo de aproximación es similar a la deliberación que hace *D²ISCO* (González-Sanz et al., 2009), herramienta que utilizamos en este proyecto junto con jCOLIBRI y que a continuación explicamos.

2.1.3. Implementación de Sistemas Recomendadores

En este proyecto hemos utilizado la herramienta jCOLIBRI y una de sus extensiones *D²ISCO*. Aunque jCOLIBRI no es la única plataforma para desarrollar sistemas de recomendación en particular o sistemas CBR en general, es una de las principales plataformas de desarrollo. Como exponente de la repercusión, tanto nacional como internacional, lograda por esta iniciativa, jCOLIBRI ha conseguido la cifra de 8000 descargas, distribuidas en casi 100 países. Cuenta además, con el apoyo de varios grupos internacionales de investigación que contribuyen a su desarrollo.

2.1.3.1. jCOLIBRI

jCOLIBRI es una plataforma de desarrollo de sistemas CBR en Java desarrollada por el grupo GAIA³ de la Facultad de Informática de la Universidad Complutense de Madrid. La primera versión de la plataforma fue desarrollada como prototipo de LISP, evolucionando después a su versión Java (jCOLIBRI). jCOLIBRI tiene dos versiones principales:

- jCOLIBRI versión 1: jCOLIBRI 1 es la primera versión de la plataforma de desarrollo. Incluye una completa interfaz gráfica de usuario que guía al usuario en el diseño de un sistema CBR. Esta versión es recomendada para usuarios no programadores que quieran crear un sistema CBR sin programar nada de código.
- jCOLIBRI versión 2: jCOLIBRI 2 es una nueva implementación con una nueva y más clara arquitectura dividida en dos capas: una orientada a desarrolladores y otra orientada a diseñadores. El nuevo diseño permite a los desarrolladores incluir las características de jCOLIBRI en sus aplicaciones CBR.

La capa de desarrollo de jCOLIBRI 2 contiene los componentes básicos de una plataforma de desarrollo. El objetivo de esta capa es que sea utilizada por los programadores dentro de las propias aplicaciones, sin incluir ninguna clase de herramientas gráficas. Incluye las herramientas necesarias para resolver la mayoría de los problemas identificados en la versión previa, de forma que se presenta como una herramienta fundamental para el desarrollo de cualquier tipo de sistema CBR. La capa de diseño de jCOLIBRI 2 incluye multitud de herramientas con interfaces gráficas de usuario. Esta capa abstrae todo el funcionamiento de la aplicación en el manejo del sistema CBR, ayudando al usuario a diseñar un sistema CBR completo a través de un proceso de configuración sin necesidad de conocimientos previos de programación o del lenguaje específico. Algunas de las características que incluye jCOLIBRI 2 son:

- Herramientas para el manejo de casos textuales como recuperación textual con Apache Lucene, clustering textual con Carrot2, medidas de similitud con OpenNLP y GATE.
- Herramientas para la visualización, clasificación y mantenimiento de bases de casos, así como diversos conectores que permiten importar y exportar bases de casos desde diferentes formatos, utilizando Hibernate y JavaBeans.
- Medidas de similitud basadas en ontologías gracias al uso de OntoBridge.

³Group for Artificial Intelligence Applications - <http://gaia.fdi.ucm.es/>

Test	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13	#14	#15	#16
Data Base connector	✓	✓	✓	✓	✓			✓								
Plain Text connector						✓	✓	✓						✓	✓	
Ontology connector										✓						
Custom connectors													✓			✓
KNN	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Enumerated Attributes		✓	✓	✓	✓											
User Defined Types		✓	✓	✓	✓											
Compound Attributes			✓	✓	✓											
Reuse methods			✓	✓	✓											
Revise methods			✓	✓	✓											
Retain methods			✓	✓	✓	✓										
Cases with solution				✓	✓	✓	✓		✓	✓				✓	✓	✓
Attributes mapped to an ontology					✓					✓						
Ontological similarity functions					✓					✓	✓					
Evaluation of CBR systems								✓						✓	✓	✓
Visualization of case bases									✓							✓
Textual CBR methods												✓	✓			✓
Textual similarity methods													✓			✓
Lucene similarity method													✓			
Accuracy Evaluation														✓		
Maintenance Algorithms							✓									✓
Maintenance Evaluation															✓	✓
Classification															✓	✓

Figura 2.1: Ejemplos de sistemas CBR en jCOLIBRI

- Integración de Tomcat y acceso a Wordnet (Miller y Fellbaum, 2007).
- Dispone de 16 ejemplos de sistemas CBR totalmente desarrollados y documentados, con las características que se muestran en la figura 2.1. Sirven como referencia al usuario a la hora de construir un sistema CBR utilizando jCOLIBRI y le muestran el uso de cada una de las características desarrolladas en la plataforma para facilitar su comprensión y su posterior uso.

jCOLIBRI permite construir sistemas CBR de los tipos:

- **CBR de carácter general:** incluye método que facilitan la construcción de sistemas CBR de propósito general como recuperación filtrada de casos, uso de la diversidad para la selección y recuperación de casos, métodos para obtención de consultas de forma gráfica y para la visualización de resultados, etc.
- **CBR textual:** Con los métodos comentados anteriormente para la recuperación, clustering o medidas de similitud textual, permite la construcción de sistemas CBR textuales de forma sencilla.
- **Sistemas recomendadores:** Dispone de los métodos necesarios para la construcción de recomendadores colaborativos, basados en casos,

Feature	Test	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13	#14
Template Type	Single Shot recommender	✓						✓						✓	✓
	Conversational A		✓			✓	✓			✓	✓	✓			
	Conversational B			✓					✓					✓	
Navigation Type	Navigation by Asking				✓	✓			✓						
	Navigation by Proposing						✓		✓	✓	✓	✓		✓	
Preference elicitation	Form Filling	✓	✓	✓			✓								✓
	Asking for an attribute				✓	✓			✓						
	Profile							✓						✓	
Retrieval	Filtering			✓	✓		✓		✓	✓	✓	✓		✓	✓
	NN Scoring	✓	✓			✓	✓	✓	✓	✓	✓	✓		✓	
	Collaborative													✓	
	Cases Selection	✓	✓			✓	✓	✓	✓	✓	✓	✓	✓	✓	
	Tabu List													✓	
	Create Complex Query from Critiques or selected Case						✓		✓	✓	✓	✓		✓	
Feature	Test	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13	#14

Figura 2.2: Ejemplos de recomendadores individuales en jCOLIBRI

navegación por preguntas y navegación por propuestas, single-shot, conversacional, así como la obtención de consultas a través de perfiles de usuario, consultas de productos en concreto, de propiedades o medidas de similitud local.etc. Además incluye 14 plantillas de recomendadores individuales, son ejemplos totalmente implementados con diferentes características para facilitar su comprensión y posterior uso, podemos ver las características de cada uno de estos ejemplos en la figura 2.2.

En resumen, jCOLIBRI es una herramienta para la construcción de sistemas CBR en general, y en particular se puede utilizar para construir sistemas recomendadores que son una versión simplificada de los sistemas CBR y el tema de interés de este proyecto. Destacamos la opción que esta plataforma ofrece para el prototipado de sistemas de recomendación por medio de plantillas (Recio-García et al., 2008), en este trabajo se ampliará la base de casos de estas plantillas con la construcción de una que permita prototipar sistemas recomendadores para grupos según nuestro método como veremos en los próximos capítulos.

2.1.3.2. D^2ISCO

Se trata de una plataforma para diseñar e implementar sistemas de Razonamiento Basado en Casos (CBR) deliberativos y colaborativos. Utilizando D^2ISCO (González-Sanz et al., 2009) se pueden diseñar e implementar sistemas CBR distribuidos donde cada nodo colabora, argumenta y contraargumenta sus resultados con otros nodos para mejorar el rendimiento de la respuesta global del sistema. D^2ISCO se ha integrado como parte de

jCOLIBRI2 (Díaz-Agudo et al., 2007).

Para probar su funcionamiento se utilizó un caso de estudio de un sistema de recomendación musical y se presentaron los resultados de un experimento donde se midió la precisión del sistema utilizando una versión difusa del sistema de argumentación AMAL (Plaza y Ontañón, 2006).

Esencialmente esta plataforma consiste en la creación de un marco práctico para la caracterización de los sistemas CBR distribuidos. Es una extensión de jCOLIBRI para la creación de sistemas CBR distribuidos, en el se implementa una modificación del protocolo de argumentación AMAL para su aplicación sobre todo tipo de casos con la única restricción de que estén valorados. Para la generación de argumentaciones dentro del protocolo de razonamiento multi-agente AMAL, utiliza un sistema de razonamiento fuzzy. Además incluye la creación de un sistema de aprendizaje del modelo de confianza para sistemas CBR distribuidos. En *D²ISCO* se realiza una demostración de forma práctica de los siguientes aspectos:

- La topología de red influye de forma significativa en los resultados obtenidos por los sistemas CBR distribuidos.
- Las redes sociales aportan una topología de red así como un modelo de confianza que mejoran los resultados obtenidos con otras topologías.
- La utilización de argumentaciones durante los procesos de razonamiento mejora los resultados obtenidos por los sistemas que la emplean.

2.2. Factores sociales en los sistemas de recomendación

El servicio que ofrecen las redes sociales se centra en construir y reflejar las relaciones sociales entre personas que por ejemplo comparten intereses comunes y/o actividades. Esencialmente consisten en la representación de cada usuario, a menudo mediante un perfil, sus conexiones sociales y una variedad de servicios adicionales.

2.2.1. Redes Sociales

Las redes sociales proporcionan una medida de confianza entre los diversos usuarios que forman parte de ellas y nos aportan toda una estructura de red construida entre ellos. Un enlace entre dos usuarios de la red social simboliza una afinidad entre estos dentro de la temática de la red. Redes sociales como Facebook⁴, MySpace⁵ o Tuenti⁶ (entre otras muchas), tienen

⁴<http://www.facebook.com>

⁵<http://www.myspace.com/>

⁶<http://www.tuenti.com>

como objetivo el intercambio de información entre sus usuarios. La temática de las redes sociales es amplia, variando desde la laboral al intercambio de fotos y música. La expansión de las redes sociales tiene su auge en los últimos años coincidiendo con la expansión de Internet, donde han cobrado una gran importancia. Los usuarios de las redes sociales buscan un lugar donde encontrar gente similar a ellos dentro de la temática de la red y con quien poder compartir sus ideas.

La investigación multidisciplinar ha mostrado que las redes sociales operan en muchos niveles. En su forma más simple, una red social es un mapa de todos los lazos relevantes entre todos los nodos estudiados. Se habla en este caso de redes “sociocéntricas” o “completas”. Otra opción es identificar la red que envuelve a una persona (en los diferentes contextos sociales en los que interactúa); en este caso se habla de “red personal”. El análisis de redes sociales se está utilizando para una enorme cantidad de temas punteros donde se ha aprovechado la información contenida en las redes sociales para construir sistemas con distintos propósitos:

- La evolución de las redes sociales a veces puede ser simulada por el uso de modelos basados en agentes, proporcionando información sobre la interacción entre las normas de comunicación, propagación de rumores y la estructura social (Yahja y Carley, 2005).
- La teoría de Difusión de innovaciones (Rogers, 1983) explora las redes sociales y su rol en la influencia de la difusión de nuevas ideas y prácticas. El cambio en los agentes y en la opinión del líder a menudo tienen un papel más importante en el estímulo a la adopción de innovaciones, a pesar de que también intervienen factores inherentes a las innovaciones.
- Un estudio ha descubierto que la felicidad tiende a correlacionarse en redes sociales (Fowler y Christakis, 2008). Cuando una persona es feliz, los amigos cercanos tienen una probabilidad un 25 por ciento mayor de ser también felices. Además, las personas en el centro de una red social tienden a ser más felices en el futuro que aquellos situados en la periferia. En las redes estudiadas se observaron tanto a grupos de personas felices como a grupos de personas infelices, con un alcance de tres grados de separación: se asoció felicidad de una persona con el nivel de felicidad de los amigos de los amigos de sus amigos.
- En McDonald (2003) se muestran diferentes sistemas que han utilizado las redes sociales como mecanismo para recomendar personas con las que colaborar o cómo utilizar visualizaciones de redes sociales para poder aprovechar las diferentes colaboraciones que surgen entre usuarios de un mismo entorno laboral.

A continuación explicaremos las ventajas de utilizar las redes sociales para crear modelos de confianza.

2.2.2. Modelos de Confianza

Las investigaciones actuales indican que las personas tienden a confiar más en recomendaciones que provienen de gente en la que confían (amigos) que en recomendaciones basados en valoraciones anónimas (Sinha y Swearingen, 2001). Este factor es incluso más importante cuando se realizan recomendaciones donde los usuarios tienen que decidir un producto para todo el grupo. Esta clase de recomendaciones generalmente sigue un proceso de argumentación donde cada usuario defiende sus preferencias y rebate las opiniones de otros. Aquí, la confianza entre usuarios es el factor principal ya que éstos deben cambiar su opinión y alcanzar una decisión común.

Existe un gran historial de trabajo sobre la generación de modelos de confianza. Sin embargo, el alzamiento de la web colaborativa actual (Web 2.0) ha estimulado la idea de una web de confianza (WOT, Web of Trust) (Golbeck, 2006b; O'Donovan y Smyth, 2005; Victor et al., 2008). La WOT representa la confianza entre usuarios, usando un modelado de la red online. Hay algunos enfoques específicos que utilizan una red social personalizada para recomendar elementos. Un ejemplo de esto es FilmTrust (Golbeck, 2006b) que utiliza este tipo de redes de confianza entre usuarios en el ámbito de las preferencias cinematográficas. Sin embargo algunas de estas redes de confianza son bastante difíciles de obtener, pues requieren preguntas directas a los usuarios para crear la red y esto puede ocasionar rechazo.

Otro enfoque prometedor es la recolección del conocimiento sobre la confianza de redes sociales existentes como Facebook o Twitter. Estas redes contienen información implícita que se podría usar para mejorar el proceso de recomendación. Esta opción es la principal ventaja de ser completamente transparente a los usuarios. Los usuarios no necesitan proporcionar información implícita sobre su confianza en otros usuarios, así que este conocimiento se extrae implícitamente de la interacción diaria con la red social. Sin embargo este método tiene un claro inconveniente y es que cada usuario del grupo a recomendar tiene que pertenecer a una red social. No obstante, la creciente popularidad de esta clase de aplicaciones web minimiza este riesgo. Es más, está siendo cada vez más común la creación de eventos (como ir al cine) a través de redes sociales luego se podrían incluir en este tipo de sitios web técnicas de recomendación para grupos.

Trabajos recientes indican que los usuarios prefieren la clase de sistema que depende en la confianza de las redes sociales, ya que suelen preferir recomendaciones de personas a las que conocen y confían (Sinha y Swearingen, 2001). El trabajo en Golbeck (2006a) presenta un estudio sobre cómo inferir las relaciones de confianza en redes sociales. El principal problema computacional de la confianza es determinar cuánto debería confiar una per-

sona en otra. Ciertamente, las inferencias sobre la confianza no serán tan precisas como una valoración directa. Pero en este trabajo, Golbeck presenta un algoritmo para inferir la confianza en red con sistemas de valoraciones continuos llamado TidalTrust, que mejora la eficiencia hasta ahora adquirida en un 10 %.

Resumen

En este capítulo hemos dado una visión general de los sistemas recomendadores y los tipos que hay, individuales colaborativos, individuales basados en contenido y grupales. También hemos explicado las herramientas existentes para construir los diferentes tipos de recomendadores. Finalmente hemos realizado una introducción a las redes sociales, su uso y sus aplicaciones. En el siguiente capítulo explicaremos nuestra propuesta de trabajo: las recomendaciones basadas en personalidad y en factores sociales, los sistemas recomendadores con memoria y una arquitectura por medio de modelos distribuidos y de argumentación.

Capítulo 3

Propuesta de Trabajo: GRUPITO

3.1. Introducción a GRUPITO (Group Recommendations Using Personality Investigation and Trust Organizations)

Los objetivos de este proyecto se centran en el estudio de los factores sociales en sistemas de recomendación, para así reflejar las interacciones que realmente hay entre los usuarios cuando se desarrolla un proceso de recomendación o negociación a la hora de realizar una actividad social. Además, se ha incluido un segundo factor que valore la influencia de los distintos perfiles de personalidad de cada usuario involucrado en el proceso (Recio-García et al., 2009).

Al realizar este trabajo, se pretende mejorar las técnicas existentes que típicamente se usaban a la hora de recomendar distintos tipos de productos a grupos (Jameson y Smyth, 2007). Una de las principales tendencias cuando se realizan recomendaciones es usar agregación de ratings¹ para individuos. Sin embargo, como vimos en el capítulo anterior, la estrategia de agregación ha sido criticada por algunos autores ya que los ratings tienden a combinarse siempre del mismo modo sin tener en cuenta las relaciones de los miembros del grupo con los otros miembros. Para solucionar este problema nosotros incluimos las interacciones, lo que significa que, realizamos las recomendaciones usando información personal sobre el comportamiento en situaciones de conflicto para cada uno de los miembros del grupo, e información sobre cómo interactuarán entre ellos.

El proceso de recomendación tendrá en cuenta la personalidad de cada individuo, es decir, la forma en la que reaccionará cuando el sistema recomienda una propuesta diferente a la que él quería, si son abiertos de mente o no, co-

¹Siendo un rating la valoración o calificación de un producto.

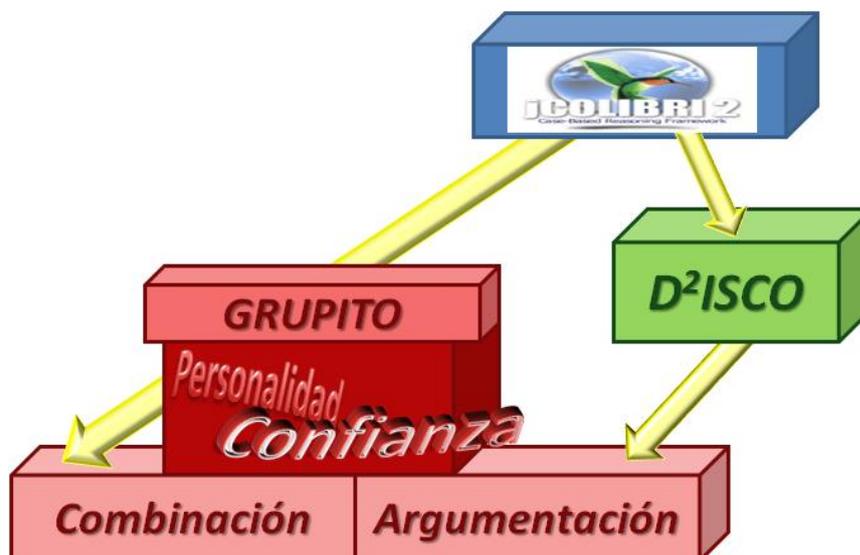


Figura 3.1: Arquitectura del modelo teórico

laboran en la toma de la decisión, etc. Y la confianza entre los componentes del grupo, quien se fiará de la opinión de quien, con quien coinciden en las decisiones normalmente, etc. De esta forma se pretende reflejar todavía más fielmente los procesos reales de recomendación y negociación sociales llevados a cabo por individuos reales.

En particular se investigará un caso de estudio en el que se intentará imitar el proceso real de argumentación cuando un grupo está decidiendo qué película ir a ver. El objetivo es comprobar si la personalidad de cada integrante y la estructura del grupo influyen en la fiabilidad del recomendador. En el sistema el grupo está conectado con enlaces sociales, luego los usuarios sólo intercambiarán recomendaciones con otros usuarios con los que tengan un enlace social. Realizar esto cuando se trata de grupos es realmente algo natural ya que la gente tiende a realizar esta clase de actividades con sus amigos o familia. Además normalmente van juntos en los mismos grupos. En la Figura 3.1 podemos ver la arquitectura de nuestra propuesta y una visión global de lo que se explicará a lo largo de este capítulo.

Aunque hemos aplicado nuestro método sólo al dominio de las películas, nuestra aproximación es general, y se podría trasladar fácilmente a otro dominio en el que los productos a recomendar estén valorados igualmente con ratings por los usuarios, ya sea música, restaurantes, viajes, etc. Para resolver la situación conflictiva según nuestro método, necesitaríamos los mismos elementos aquí usados, la personalidad de cada individuo, que se podría obtener con el test TKI (Thomas y Kilmann, 1974) que refleja el modo de reaccionar en situaciones de toma de decisiones en grupo (nosotros proponemos este test ya que se centraba específicamente en el manejo de conflictos y era re-

lativamente sencillo de hacer, sin embargo se podría usar cualquier otro test que reflejara la personalidad de cada individuo en situaciones conflictivas), la confianza en las opiniones y decisiones de los otros componentes del grupo, y cómo combinar estos factores.

3.2. Recomendaciones basadas en personalidad

La mayoría de los trabajos anteriores sobre recomendaciones grupales consideran las preferencias de cada miembro del grupo con el mismo grado de importancia intentan satisfacer las predilecciones de cada uno, como en McCarthy y Anagnost (1998); McCarthy (2002) o Lieberman et al. (1999). Sin embargo, hay que considerar que no todos los grupos son iguales, estos tienen diferentes características, como el tamaño o pueden estar formados por personas con preferencias similares o antagonistas. Cuando nos enfrentamos a una situación en la que las inquietudes de las personas son incompatibles surge una *situación conflictiva*. Los sistemas recomendadores para grupos existentes generalmente resuelven este tipo de conflicto, intentando maximizar las preferencias del mayor número de miembros del grupo. Pero, es bien sabido que la satisfacción general del grupo no es siempre la agregación de las satisfacciones de sus miembros pues cada persona tiene diferentes expectativas y comportamientos en situaciones de conflicto, por lo que estas se deben tener en cuenta.

En esta sección presentamos un método para las recomendaciones grupales que distingue entre diferentes tipos de individuos en un grupo. La investigación consiste en caracterizar a la gente usando el test Thomas-Kilmann Conflict Mode Instrument (TKI) (Thomas y Kilmann, 1974). El TKI es un test diseñado para la medición del comportamiento humano en situaciones conflictivas. Es un instrumento puntero en los estudios de determinación de conflictos. También es usado por asesores de Recursos Humanos y Desarrollo Organizativo para facilitar el aprendizaje sobre cómo los distintos estilos de manejar el conflicto afectan a dinámicas personales y grupales. El TKI se encarga de construir un perfil de usuario a través de 30 preguntas de respuesta múltiple. Podemos ver ejemplos de las preguntas que se realizaron a los usuarios en la Figura 3.2. El test proporciona puntuaciones para cinco tipos diferentes de personalidad, ver Figura 3.3. Estas representan la tendencia de cada persona al enfrentarse a situaciones conflictivas. Las puntuaciones se normalizan para obtener percentiles usando una muestra de 8000 personas. Este test se eligió, debido a que se centra en la gestión y manejo de conflictos, es sencillo, pues puede hacerse en 15 minutos y los resultados se integran fácilmente en su enfoque.

El TKI describe el comportamiento de una persona en situaciones conflictivas por medio de dos dimensiones básicas: autoritarismo y cooperacionismo. Estas dos dimensiones de comportamiento se pueden utilizar para

Test de Personalidad

Pregunta 1

- En ocasiones he dejado a los otros la responsabilidad de encontrar la solución a los problemas.
- En vez de negociar sobre los aspectos en los que hemos estado de acuerdo, he preferido aclarar bien los aspectos en los que si hemos estado de acuerdo

Pregunta 2

- Dependiendo de la situación, algunas veces he luchado por los resultados y en otras, más me ha importado mantener la buena relación.
- He intentado tomar en cuenta tanto mis inquietudes como también las de la otra parte.

Pregunta 3

- Generalmente he estado muy decidido a lograr mis objetivos.
- A veces he intentado calmar los sentimientos del otro y mantener nuestra relación.

Figura 3.2: Ejemplo de preguntas realizadas en el test TKI

definir cinco tipos distintos de personalidad existentes en situaciones conflictivas: competitiva, colaborativa, evasiva, complaciente y comprometida. El método, que titulamos como *Personality Aware Recommendation to Groups*, tiene en cuenta estos cinco tipos de personalidades.

La propuesta consiste en recomendar productos a un grupo usando la información personal sobre el modo de comportamiento conflictivo de cada miembro del grupo. El autoritarismo penaliza negativamente las diferencias entre las selecciones preferidas de otros miembros (pues esas selecciones no satisfacen sus propias inquietudes), mientras que el cooperacionismo recompensa las diferencias con las selecciones preferidas por otros miembros (ya que no es su selección pero es suficientemente buena para los otros componentes y por tanto para el grupo).

Una vez que cada usuario u_i ha completado el test TKI, se calcula un *peso del tipo de conflicto* (CMW_i) (*Conflict Mode Weight*), este valor representará el comportamiento del usuario predominante de acuerdo con su evaluación del TKI. Para calcular el valor CMW_i realizamos los siguientes pasos: miramos el porcentaje que cada usuario tiene de personalidad competitiva, colaborativa, evasiva, complaciente y comprometida. Estos factores están reflejados en algunas preguntas concretas del test, más concretamente hay 12 preguntas por cada categoría, por ejemplo para calcular cuán de *competitiva* es una persona, miramos cuántas de las preguntas 3, 8, 10, 17, 25 y 28 tienen marcada la primera opción y cuantas de las preguntas 6, 9, 13, 14, 16 y 22 tienen marcada la segunda opción, una vez que hemos contabilizado cuántas respuestas reflejan una personalidad *competitiva* dividimos entre 12 y obtenemos el porcentaje de personalidad de esta categoría. Repetimos este procedimiento para las categorías restantes y sus preguntas respectivas, al final tenemos 5 valores distintos de porcentajes de tipos de personalidad. El siguiente paso es reducir estos cinco tipos de personalidad a dos, *Autoritaris-*

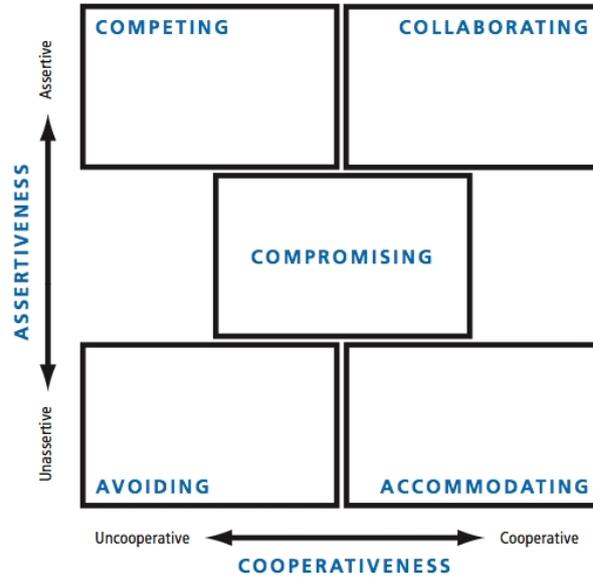


Figura 3.3: Tipos de personalidad TKI

Modalidad TKI	Porcentaje	Calificación
Competitiva	16.7 %	Bajo
Colaborativa	25.0 %	Bajo
Complaciente	75.0 %	Alto
Evasiva	83.3 %	Alto
Comprometida	58.3 %	No relevante

Tabla 3.1: Ejemplo de coeficientes CMW de un usuario

mo y *Cooperacionismo*, para ello utilizamos los valores de la tabla 3.2. Los coeficientes de la tabla se obtuvieron por medio de una evaluación experimental y se fueron ajustando a medida que hicimos experimentos de forma que reflejasen del mejor modo la personalidad de los usuarios. Consideramos que un tipo de personalidad tiene un coeficiente alto si supera el 73 % y un coeficiente bajo si es inferior al 27 %, si está entre estos dos límites consideramos que no tiene un valor relevante para esa categoría. Para cada una de las opciones, *Autoritarismo* y *Cooperacionismo*, sumamos los valores indicados en la tabla de las 5 categorías. Para cada una de las 5 categorías observamos si su valor es mayor o menor a esos límites y las calificamos como *Alto*, *Bajo* o *No relevante* en función de esa clasificación sumaremos un valor o otro. Por ejemplo, vemos cómo se calculan los valores de *Autoritarismo* y *Cooperacionismo* para un usuario en particular u_x que tenga los porcentajes de personalidad que muestra la tabla 3.1:

$$Autoritarismo(u_x) = -0,075 + (-0,075) + 0 + (-0,375) + NoRelevante = -0,525 \quad (3.1)$$

Modalidad TKI	Autoritarismo		Cooperacionismo	
	Alto	Bajo	Alto	Bajo
Competitiva	0.375	-0.075	-0.15	0
Colaborativa	0.375	-0.075	0.375	-0.075
Complaciente	0	0	0	0
Evasiva	-0.375	0.075	-0.375	0.075
Comprometida	-0.15	0	0.375	-0.075

Tabla 3.2: Coeficientes usados para calcular el CMW

$$Cooperacionismo(u_x) = 0 + (-0,075) + 0 + (-0,375) + NoRelevante = -0,45 \quad (3.2)$$

Finalmente, el factor $CMW_i \in [0, 1]$ se calcula utilizando la siguiente ecuación:

$$CMW_i = \frac{1 + Autoritarismo(u_i) - Cooperacionismo(u_i)}{2} \quad (3.3)$$

Valor que resultaría para nuestro usuario en particular u_x igual a 0.4625. Se puede ver que los usuarios que tengan un carácter predominantemente autoritario tendrán un $CMW_i > 0,5$ mientras que los usuarios cuyo carácter predominante sea el cooperativo obtendrán un $CMW_i < 0,5$.

Los experimentos que realizamos, determinaron que la composición de la personalidad en el grupo influía ligeramente en la eficacia de la recomendación para el grupo. Esta mejoría sólo era alcanzada para cierto tipo de grupos, como fue comprobado tras compararlo con diferentes algoritmos de recomendación grupal.

En este proyecto, ampliamos esa propuesta con la confianza social, es decir, teniendo en cuenta la estructura del grupo y las conexiones sociales entre los diferentes individuos. La forma en la que incluiremos este nuevo factor se explicará a continuación.

3.3. Recomendaciones basadas en factores Sociales

En esta sección se explica cómo hemos estudiado y tenido en cuenta los factores de personalidad y confianza en el proyecto. Sabíamos que la estimación de un valor de confianza entre dos personas variaría de red social en red social, que hay diversas formas de obtenerlo y que no sería igual de preciso que una valoración directa. Sin embargo tras realizar estudios sobre dos redes sociales bastante populares, como vimos en la Sección 2.2.2 y después de estudiar diversas propuestas como Gilbert y Karahalios (2009)

o Golbeck (2006a) confirmamos que la mejor forma de valorar los factores de confianza es la que aquí se presenta.

Después de la revisión bibliográfica hemos recopilado 10 factores distintos para realizar el cálculo de la confianza entre usuarios de un determinado grupo, que son:

- f_1 : Distancia en la red social (amigo, amigo de amigo, ...).
- f_2 : Numero de amigos en común.
- f_3 : Intensidad de la relación (cómo de a menudo se escriben en los muros o por mensajes privados).
- f_4 : Intimidad de la relación (qué clase de mensajes se intercambian, cariñosos, amistosos, profesionales, ...).
- f_5 : Duración (desde cuando se conocen, cuándo se comunicaron por primera vez, cuándo fue la última comunicación, ...).
- f_6 : Servicios recíprocos (links intercambiados en el muro, aplicaciones comunes, ...).
- f_7 : Variable estructural (grupos en común, redes que coincidan, intereses, y datos del perfil).
- f_8 : Distancia social (política, educacional, religiosa, geográfica, ...).
- f_9 : Status (familia, mejor amigo, pareja, ...).
- f_{10} : Fotos (número de fotos que aparecen juntos o que han comentado).

(3.4)

Algunos de estos factores se pueden obtener fácilmente de las redes sociales. Por ejemplo, Facebook proporciona la información requerida por los factores 2, 9 o 10. Sin embargo, hay otros factores que requieren un análisis extra de la red social. Más específicamente, hay algunos factores (como la intimidad) que requieren un procesado de los mensajes intercambiados. En estos casos, se necesitan técnicas de Extracción de Información para extraer palabras clave que identifiquen la naturaleza del texto.

Una vez que hemos elegido los distintos factores que están involucrados en el cálculo de la confianza social, existe el problema de cómo combinarlos. Los factores previamente descritos pueden tener impactos distintos en el proceso de recomendación, así que hemos optado por combinarlos usando una media ponderada:

$$t_{ij} = \sum_{k=1}^{10} \alpha_k \cdot f_k(u_i, u_j) \quad (3.5)$$

Hemos medido la importancia de cada factor α_k usando un enfoque experimental. En el caso de estudio que se presenta en la Sección 4.3 se muestra cómo se han calibrado estos factores para maximizar la eficiencia del recomendador.

3.4. Inclusión de los factores de personalidad y modelado social en las recomendaciones para grupos

En esta sección vamos a explicar nuestro método para obtener la valoración de cada usuario u_i para cada producto m , utilizando un enfoque basado sólo en la personalidad, con un enfoque basado en delegación y con un enfoque basado en influencia.

Nuestro método de recomendación para grupos está basado en los enfoques clásicos de agregación de preferencias. Estos enfoques (Masthoff y Gatt, 2006; O'Connor et al., 2001) combinan los ratings individuales de cada usuario para tratar de satisfacer al mayor número de miembros del grupo posible. Por tanto el bloque básico de construcción de nuestro recomendador grupal consiste en un recomendador individual que calcula las preferencias de un determinado usuario.

El recomendador individual implementa un algoritmo de filtrado colaborativo usando la función de Correlación de Pearson. Más concretamente, implementamos el algoritmo que se propone en Kelleher y Bridge (2004). El resultado del recomendador individual será un rating estimado $r_{i,m}$ para un usuario u_i y una producto dado m . A continuación, combinamos los ratings que el recomendador individual ha predicho para obtener la recomendación para el grupo. Una forma bastante común de combinar los ratings individuales $ir_{i,m}$ es la estrategia de agregación promedio (average aggregation), que fue presentada en la sección 2.1.2.1, ecuación 2.1:

$$average(G, m) = \sum_{u_i \in G} ir_{i,m} \quad (3.6)$$

Donde G es un grupo de usuarios dado al que u_i pertenece. Esta función proporciona un valor de rating que representa la preferencia del grupo para un producto dado m . Con estos ratings el recomendador grupal propone la colección de elementos que posean la mayor satisfacción media predicha para cada grupo de usuarios. Nos referiremos a esta colección como *gf*: las *favoritas del grupo*.

La estrategia de agregación promedio (average aggregation) se ha utilizado satisfactoriamente en muchos recomendadores como Intrigue (Ardisson et al., 2003), Travel Decision Forum (Jameson, 2004) o YuTv (Yu et al., 2006), por esto la elegimos como nuestra estrategia base.

Nuestra propuesta de recomendación emplea la función básica de promedio pero utilizando ratings de cada individuo modificados para incluir la personalidad y la confianza.

$$average'(G, m) = \sum_{u_i \in G} ir'_{i,m} \quad (3.7)$$

La mejoría en la eficiencia del recomendador grupal con respecto del recomendador base antes explicado va a depender de la implementación de los nuevos valores $ir'_{i,m}$. A continuación proponemos varios enfoques para calcular el $ir'_{i,m}$. Las fórmulas que proponemos consisten en combinar los ratings individuales (predichos por el recomendador individual) con los factores de confianza y personalidad.

3.4.1. Ratings basados en la personalidad

En este apartado calculamos ratings en los que sólo se tiene en cuenta la personalidad. Lo que observamos son las diferencias de personalidad entre pares de individuos del grupo. Se basa en una variación de la satisfacción media presentada en un trabajo anterior (Recio-García et al., 2009). Esta estrategia refleja que las personas con carácter más autoritario tendrán más influencia en la satisfacción media que las que tengan caracteres más cooperativos. Este factor se calcula como la distancia de los valores CMW en la función de diferencia de personalidad $pd(u_i, u_j)$. Recordamos que un valor alto de CMW representa una personalidad fuerte, mientras que uno bajo representa una personalidad afable.

$$pbr(u_i, m) = \frac{1}{|G|} \sum_{u_j \in G \wedge j \neq i} (ir_{i,m} + pd(u_i, u_j)) \quad (3.8)$$

donde $pd(u_i, u_j) = (CMW_j - CMW_i) * \alpha$, el valor α se ha seleccionado experimentalmente (tiene un valor de 5.0) y se utiliza para modificar el impacto de las diferencias de personalidad en el rating modificado.

3.4.2. Ratings basados en la delegación

El enfoque basado en delegación sigue las ideas presentadas en Golbeck (2006a), donde el rating individual de una persona está basado en los ratings dados por otros usuarios. La idea detrás de este enfoque es que la opinión del usuario se crea basándose en las opiniones de sus amigos. Este promedio de opiniones se calibra dependiendo del nivel de confianza con cada amigo. Adicionalmente, la personalidad de cada amigo también se tiene en cuenta, modificando así la opinión base. El rating basado en delegación (delegation-based rating) intenta simular el siguiente comportamiento: cuando se decide qué elemento elegir entre un grupo de usuarios, preguntaremos a la gente con

la que tengamos más confianza. Por tanto, también tendremos en cuenta su personalidad, y le daremos más importancia a unas preferencias que a otras.

El rating basado en delegación se calcula del siguiente modo:

$$dbr(u_i, m) = \frac{\sum_{u_j \in G \wedge j \neq i} (t_{ij} \cdot (ir_{j,m} + CMW_j))}{\sum_{u_j \in G \wedge j \neq i} t_{ij}} \quad (3.9)$$

donde t_{ij} es la confianza existente entre el usuario i y el j .

3.4.3. Ratings basados en la influencia

El enfoque de los ratings basados en influencia simula la influencia que tiene cada amigo en una determinada persona. En vez de crear una nueva preferencia, parte de la suposición de que el usuario modificará sus preferencias por un elemento dado, dependiendo de las preferencias de sus amigos por ese mismo. Por ejemplo, si nuestro rating para un elemento es de 3 y nuestro amigo tiene un rating para el mismo elemento de 5, podríamos modificar nuestro rating a 4. Dependiendo de la confianza con este amigo en concreto decidiremos el nivel de variación de nuestro rating (i.e. daremos 3.5 si es baja, o 4.5 si la confianza es alta). Además la variación del rating también dependerá de nuestra propia personalidad. Si tenemos una personalidad muy fuerte no estaremos dispuestos a cambiar nuestra valoración, pero si en cambio tenemos una personalidad débil, fácilmente nos podrán influenciar las opiniones de otros. Combinando estos dos factores obtenemos la siguiente formula:

$$ibr(u_i, m) = ir_{i,m} + (1 - CMW_i) \cdot \frac{\sum_{u_j \in G \wedge j \neq i} (t_{ij} \cdot (ir_{j,m} - ir_{i,m}))}{|G|} \quad (3.10)$$

donde t_{ij} es la confianza existente entre el usuario i y el j y $|G|$ es el número de componentes del grupo.

3.5. Recomendaciones con memoria

Tener recomendaciones con memoria significa ser capaces de crear un sistema que recuerde cuáles han sido las recomendaciones anteriores para un grupo determinado. Creemos que éste es un paso necesario cuando se pretende proporcionar toda una colección de recomendaciones. Hasta ahora, nos hemos centrado en la situación en la que el sistema hará recomendaciones sólo una vez. Pero frecuentemente un grupo esperará que el sistema le proporcione una muestra mayor de decisiones. Cada una de estas decisiones se podría tratar individualmente, sin embargo, para evitar una insatisfacción global en el grupo se debería tener en cuenta la necesidad de la existencia

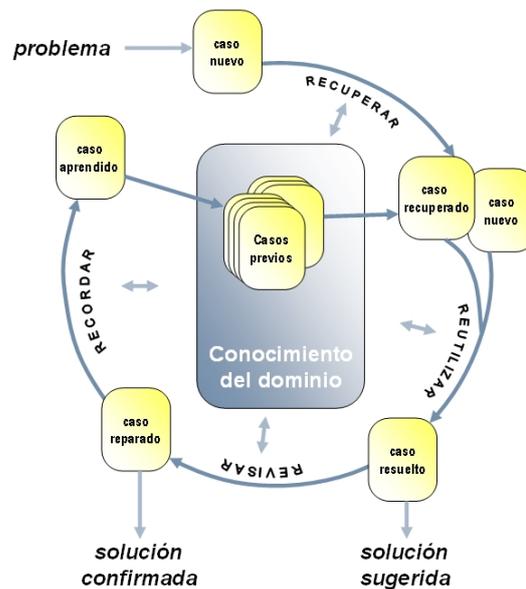


Figura 3.4: Ciclo CBR

de un determinado grado de justicia. Por ejemplo, se puede dar el caso en el que una recomendación resulte muy buena para el grupo en general pero a costa de esa recomendación puede que un usuario quede especialmente insatisfecho. Por ello sería deseable que las recomendaciones futuras favoreciesen a ese componente del grupo en concreto para que alcance un nivel apropiado de satisfacción.

En nuestro estudio hemos tenido en cuenta la historia de los resultados que se van obteniendo. Este factor se ha tenido en cuenta en otros proyectos, por ejemplo en FlyTrap (Crossen et al., 2002), que se trata de un sistema de selección de música, se tienen en cuenta las selecciones anteriores para que a la hora de elegir la siguiente canción no haya por ejemplo cambios bruscos de género. Nosotros hemos creado un sistema con memoria de las recomendaciones previas realizadas al grupo, de este modo si un miembro ha tenido que aceptar una propuesta en la que no estaba muy interesado, ahora tendrá algún tipo de preferencia para la próxima recomendación, es decir, el peso de su opinión será mayor la próxima vez. Estos pesos también estarán influenciados por la personalidad de cada componente del grupo (por ejemplo, alguien con una personalidad muy conflictiva si ha sido perjudicado una vez habrá que compensarle inmediatamente para la próxima, sin embargo una persona con una personalidad muy afable no tendrá inconveniente en ceder varias veces).

El proceso que hemos seguido es similar a las técnicas de Razonamiento Basado en Casos (Aamodt y Plaza, 1994), podemos ver el proceso del ciclo CBR en la figura 3.4. Cada recomendación realizada se guardará, lo que

sería en el ciclo CBR la etapa de retener. De este modo adquirimos experiencia, que nos es útil para la resolución de futuras recomendaciones, pues sabemos qué productos se le han recomendado al grupo pudiendo así evitar repeticiones, también sabemos el estado de felicidad de cada miembro del grupo con esa recomendación, permitiéndonos corregir el factor de felicidad en futuras recomendaciones. Cuando se realice la siguiente recomendación revisaremos la situación previa, lo que en el ciclo CBR sería la etapa de recordar. Una vez obtengamos dicha información realizaremos de nuevo la recomendación, pero teniendo en cuenta lo que hemos recordado (los productos que ya hemos recomendado y cuán de contentos están cada uno de los miembros del grupo), esto equivaldría a la etapa CBR de reutilizar. Por último modificaremos la recomendación para que no se repitan productos previamente recomendados y para asegurar un cierto grado de justicia a la hora de favorecer las preferencias de cada usuario, esta última etapa, la de revisar, sería la que cerraría el ciclo CBR.

Las siguientes fórmulas reflejan estos conceptos, la primera aplicada al enfoque basado en delegación y la segunda al enfoque basado en influencia:

$$dbrm(u_i, m) = \frac{\sum_{j \in G \wedge j \neq i} (t_{ij} \cdot (ir_{j,m} + CMW_j) + \alpha \cdot CMW_j \cdot (1 - d_j))}{\sum_{j \in G \wedge j \neq i} t_{ij}} \quad (3.11)$$

$$ibrm(u_i, m) = ir_{i,m} + (1 - CMW_i) \cdot \frac{\sum_{j \in G \wedge j \neq i} (t_{ij} \cdot (ir_{j,m} - ir_{i,m}) + \alpha \cdot CMW_j \cdot (1 - d_j))}{|G|} \quad (3.12)$$

Donde α (que tiene un valor de 1.0) se ha seleccionado experimentalmente y se utiliza para modificar el impacto del uso de memoria en el rating modificado, d_j es el nivel de descontento de un usuario j y $|G|$ es el número de componentes del grupo G . Recordamos que t_{ij} es la confianza existente entre el usuario i y el j y que $ir_{i,m}$ es el rating individual del usuario i sobre el producto m .

Vemos cómo se tendrían en cuenta todos estos factores con un ejemplo de un grupo real. Por ejemplo Peter, John y Mary van juntos al cine por primera vez. Inicialmente todos tienen el mismo nivel de descontento, es decir $d_j = 1$ para todos ellos (el valor de descontento d_j está tomado de forma que un usuario sumamente contento con las recomendaciones tendrá este valor de medición de su felicidad siempre a 1, sin embargo cuanto más le desagraden las recomendaciones, ese valor irá disminuyendo hasta llegar a 0 en el peor caso). Como es la primera vez que van al cine juntos no tienen historia luego ésta no se tiene en cuenta esta vez. Al tener todos el nivel de felicidad a 1, la

segunda parte de las ecuaciones explicadas arriba, 3.11 y 3.12, queda anulada, es decir, $\alpha \cdot CMW_j \cdot (1 - d_j) = 0$. Luego el proceso de recomendación es el mismo que aplicábamos anteriormente cuando no teníamos en cuenta ni la felicidad ni la historia, es decir los procesos originales de ratings basados en delegación, Sección 3.4.2, y ratings basados en influencia, Sección 3.4.3. A continuación obtendrán una recomendación, por ejemplo podrían obtener que la película que deberían ir a ver es *Amovie*. En este momento se actualizan los valores de descontento d_j . Para actualizar estos valores estudiamos la valoración de la película recomendada según el recomendador individual para cada usuario. Si individualmente esa película no sería recomendada al usuario el valor de descontento será muy bajo, si en cambio sería una de las películas que se le recomendarían al usuario individualmente el valor d_j seguirá siendo cercano a 1. Pongamos que Peter, que tiene una personalidad muy fuerte ha salido especialmente insatisfecho, John que tiene una personalidad afable también ha salido perjudicado con la recomendación, mientras que Mary se ha salido con la suya y ha visto la película que ella quería, luego está muy contenta y su valor de descontento sigue a 1. La siguiente vez que vayan al cine, la opinión de Mary sobre una película estará únicamente influenciada por la confianza que tienen sus amigos en ella y por su personalidad, lo que se refleja por ejemplo en el método basado en delegación con $t_{ij} \cdot (ir_{j,m} + CMW_j)$. El mismo proceso ocurrirá con las opiniones de Peter y John, sin embargo como ellos salieron perjudicados la última vez, la valoración de su opinión tendrá un incremento ya que se le está sumando su nivel de descontento, en función de su personalidad, $\alpha \cdot CMW_j \cdot (1 - d_j)$. Además en el caso particular de Peter, que como hemos dicho antes tiene una personalidad muy fuerte, esa incremento de valoración será mayor en su caso que en el de John, ya que el valor del *peso del tipo de conflicto* (CMW_j), que como recordamos es mayor cuanto más conflictiva sea una persona y menor cuanto más afable sea, es más alto. Finalmente se obtiene la recomendación de la nueva película que deberían ir a ver, como ahora tenemos memoria de qué películas han ido a ver ya juntos, en el caso de que el recomendador propusiese una película que ya han visto, véase la película *Amovie*, la descartaría automáticamente y propondría la siguiente mejor película que ofrece el recomendador grupal. A continuación se volverían a actualizar los niveles de descontento y la base de datos de películas vistas y el proceso volvería a ser el mismo la siguiente vez que quisieran ir al cine juntos. Podemos ver un esquema de este proceso de recomendaciones con memoria en la figura 3.5.

En el siguiente apartado proponemos una forma de implementar y llevar a cabo estas teorías con una arquitectura multi-agente.



Figura 3.5: Ejemplo del proceso de recomendación con memoria

3.6. Modelos distribuidos y argumentación

Hasta ahora hemos descrito tres aspectos generales, los relacionados con la personalidad, Sección 3.2 los relacionados con las redes sociales y la confianza entre los miembros del grupo, Sección 3.3 y los relacionados con la memoria, Sección 3.5. El segundo aspecto novedoso de nuestro enfoque es la inclusión de una arquitectura del grupo que imita las conexiones de redes sociales en el modelo de recomendación. El objetivo principal es mejorar las recomendaciones teniendo en cuenta, la topología de la red, la composición de la personalidad del grupo y la confianza entre los componentes del grupo. Para realizar esto usaremos sistemas multiagentes con topología de red social, donde cada agente representa a un miembro del grupo.

Cada uno de los agentes representando a cada usuario defenderá sus mejores intereses argumentando con los otros agentes a los que está conectado, y la recomendación final dependerá de la personalidad de cada uno y de cuanto confíen entre ellos. Eso significa que las preferencias de cada individuo tendrán diferentes pesos dependiendo de la manera en que cada miembro del grupo reaccionaría en una situación conflictiva. De esta manera consideramos el proceso de recomendación en un grupo teniendo en cuenta una topología de red basada en una red social. Como hemos mencionado anteriormente en la Sección 2.2.1, las redes sociales están siendo uno de los temas más impor-

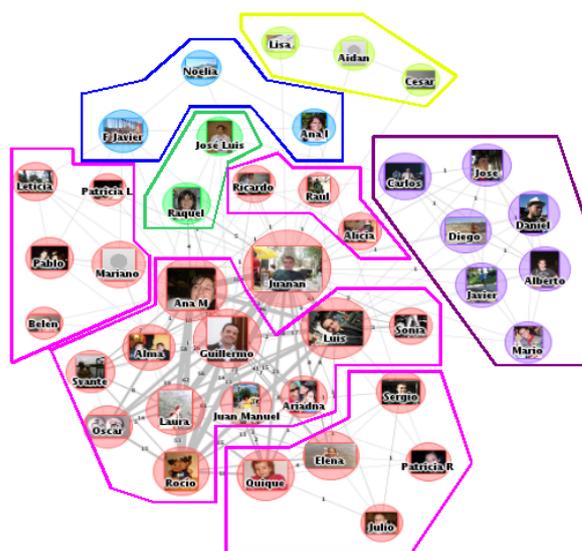


Figura 3.6: Estructura de red social

tantes en los últimos años, con redes como Facebook², MySpace³ o Tuenti⁴. Estas representan una estructura más realista de las relaciones entre los miembros de un grupo. Por ejemplo la Figura 3.6 muestra una representación de una red social real en Facebook. Por tanto nuestra hipótesis inicial es que esta nueva organización de la estructura del grupo afectará y mejorará el resultado de la recomendación. Esta mejora se deberá principalmente a que la red social aporta una estructura más realista a la organización de los agentes, es decir, es más similar a la manera en las que las argumentaciones tendrían lugar si se tratase de un grupo real que está discutiendo sobre la actividad a realizar. Para un grupo de amigos dado, utilizaremos una red social de donde extraeremos la información relativa a la organización del grupo, y donde cada nodo representará a un individuo, y cada conexión representará la relación real con otro individuo al que está conectado.

Cada nodo discutirá sus opiniones con sus vecinos en la red. Cuando dos nodos no están conectados significa que las personas a las que representan no se conocen entre ellas o que no son cercanas. Esta estructura refleja las discusiones cara a cara donde nunca se daría una discusión entre ellos ni tendrían que argumentar al no conocerse o tener confianza suficiente. Por ejemplo si alguien llevase a su pareja y el resto del grupo no le conociese, este nuevo integrante en el grupo no tendría que argumentar con gente que no ha conocido antes, por tanto cuando nosotros representamos las relaciones sólo conectaremos a esta persona con las personas que conoce. Sin

²<http://www.facebook.com>

³<http://www.myspace.com/>

⁴<http://www.tuenti.com>

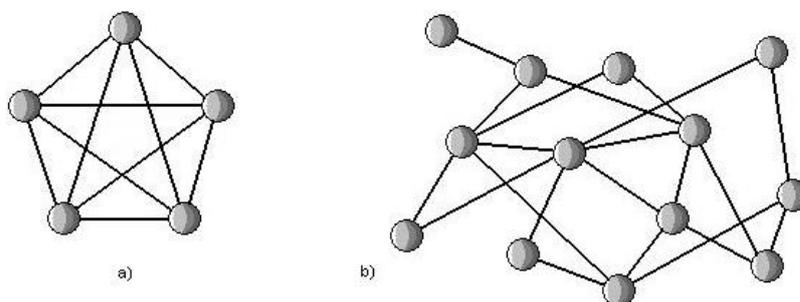


Figura 3.7: a) Topología "todos con todos" b) Topología en forma de red social

embargo si estuviéramos tratando con una topología “todos con todos”, cada agente debatiría con todos los demás, sin tener en cuenta si el otro agente está representando a una persona que conoce o no. La Figura 3.7 muestra las diferencias entre las dos alternativas de topología de red que hemos comentado, la topología “todos con todos” y la topología en forma de red social. Para el proceso de argumentación se usará un sistema multi-agente, de esta manera cada agente representa a un componente del grupo, y discutirá/ defenderá sus intereses, de este modo usamos la estructura de nodos colaborativos para el sistema CBR, que es el que usa D^2ISCO , y además la topología de red que seleccionamos es de tipo social. El protocolo de razonamiento comenzará cuando un agente genere una consulta a uno de los agentes con los que está conectado, como representa la Figura 3.8. Cada agente proporcionará una recomendación local para el problema en cuestión según sus propias preferencias. En cada ronda de discusión cada agente intentará rebatir la solución que alguno de sus agentes vecinos haya hecho. Estos contraejemplos que se van rebatiendo tendrán diferentes pesos, por lo tanto serán tomados más o menos en cuenta en la decisión final, dependiendo de qué agente provengan, pues al representar a una persona en concreto debe quedar reflejada el tipo de personalidad que tenga esa persona, y el peso de sus opiniones en una discusión en el grupo. Este proceso de argumentación hasta llegar a la respuesta final queda reflejado en la Figura 3.9.

3.6.1. Configuración de recomendadores para grupos con modelos distribuidos y argumentación

Para poder realizar recomendaciones a grupos utilizando la arquitectura de sistemas multi-agente explicada en el apartado anterior utilizamos la herramienta D^2ISCO , que ya introducimos en la Sección 2.1.3.2, de ella utilizaremos la distribución y arquitectura de los agentes, que tienen una topología en forma de red social y los métodos de argumentación entre agentes. Hemos realizado diferentes configuraciones a las que presentaba la

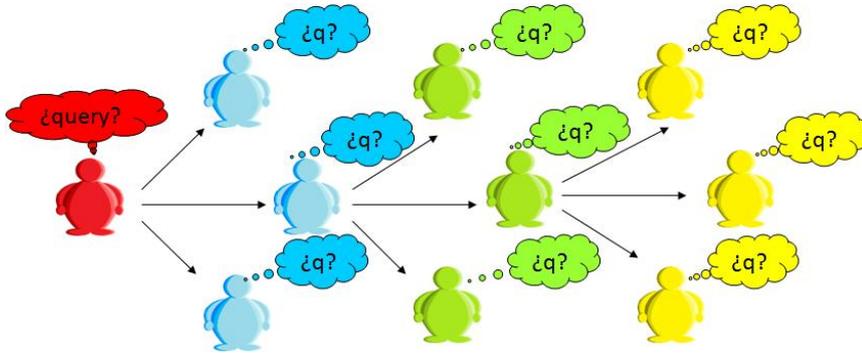


Figura 3.8: Propagación de la Query

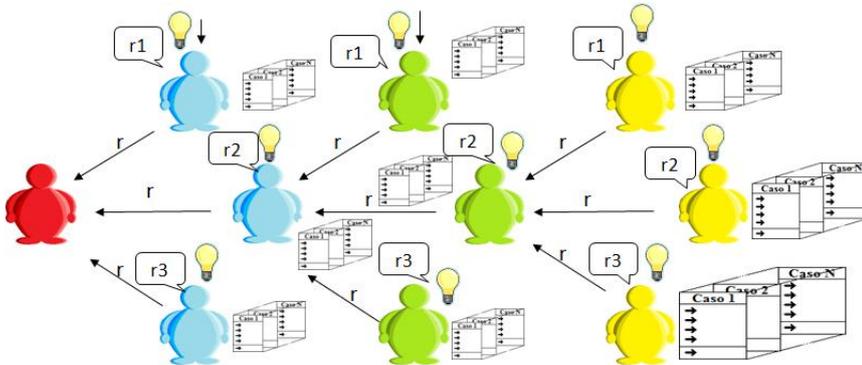


Figura 3.9: Argumentación

aplicación inicialmente para poder tener en cuenta los valores personalidad y confianza, como hemos explicado a lo largo de este Capítulo. Para realizar estos cambios necesarios hemos realizado los siguientes pasos: (1) Configuración de los agentes y configuración de una base de datos, (2) configuración de queries a lanzar, (3) implementación de un recomendador individual y (4) inclusión en el sistema de argumentación de *D²ISCO*.

Nosotros hemos utilizado los modelos distribuidos y la argumentación para el caso concreto de las recomendaciones de películas para grupos de personas, teniendo en cuenta la personalidad de cada individuo, las conexiones de los componentes en una red social y la confianza mutua. A continuación explicamos el proceso que hemos seguido. Cabe destacar que aunque aquí lo aplicamos a nuestro dominio particular y con un recomendador individual de nuestra elección, es un proceso válido para cualquier otro dominio en el que los productos estén valorados en ratings u otro tipo de recomendador individual. Lo primero que realizamos es la configuración de los agentes, la Figura 3.10 muestra los pasos iniciales a seguir para poder configurar la aplicación y reutilizar nuestro método. Cada usuario del sistema está representado por un agente. Cada agente deberá tener la siguiente información: identificación,

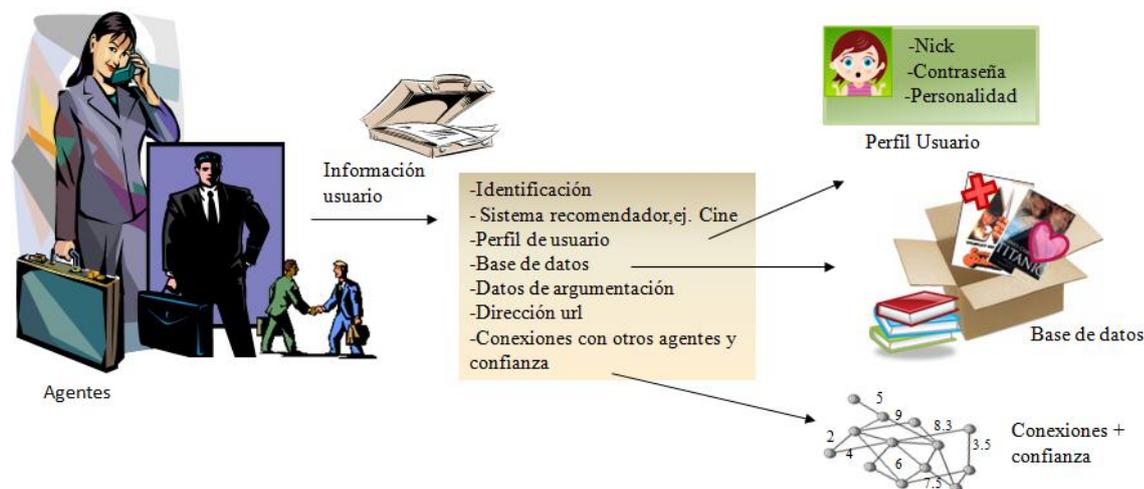


Figura 3.10: Primer paso: configuración de los agentes

sistema recomendador que va a utilizar, perfil del usuario, base de datos del usuario, datos de argumentación, una dirección url, links con otros agentes y confianza entre los agentes a los que está conectado.

El perfil de cada usuario contiene su nick, su contraseña y un número que califique su personalidad (el CMW). La base de datos de cada usuario consiste en una colección de productos (en nuestro caso particular, de películas) que contiene las descripciones de cada producto y la valoración del usuario. En cuanto a los datos de argumentación, se introducen valores de configuración del proceso de argumentación que realiza D^2ISCO , como el máximo de ciclos, el valor mínimo de contraejemplos, el valor mínimo en una defensa...etc

A continuación se configuran las queries que se van a lanzar, es decir los productos (y sus descripciones) que se les va a pedir valorar a los usuarios grupalmente. En nuestro caso las queries representan una cartelera de cine, luego obtendremos la valoración grupal de cada una de las posibles películas a ir a ver y finalmente se seleccionará como producto a recomendar al grupo la película que obtenga una valoración mayor.

Para finalizar la configuración inicial debemos implementar un recomendador individual. En nuestro caso particular hemos realizado un recomendador basado en contenido, como los explicados en la Sección 2.1.1.2. En los que se compara la descripción de la query con las descripciones de los productos alojados en la base de datos del usuario en particular que recibe la query y se califica la query con la calificación que tenga aquel producto que mayor similitud tenga, así como la medida de similitud. El esquema de cómo finalizar la configuración inicial está reflejado en la Figura 3.11.

Un pequeño problema que encontramos en el sistema de argumentación de D^2ISCO era la limitación de su funcionamiento cuando existían ciclos

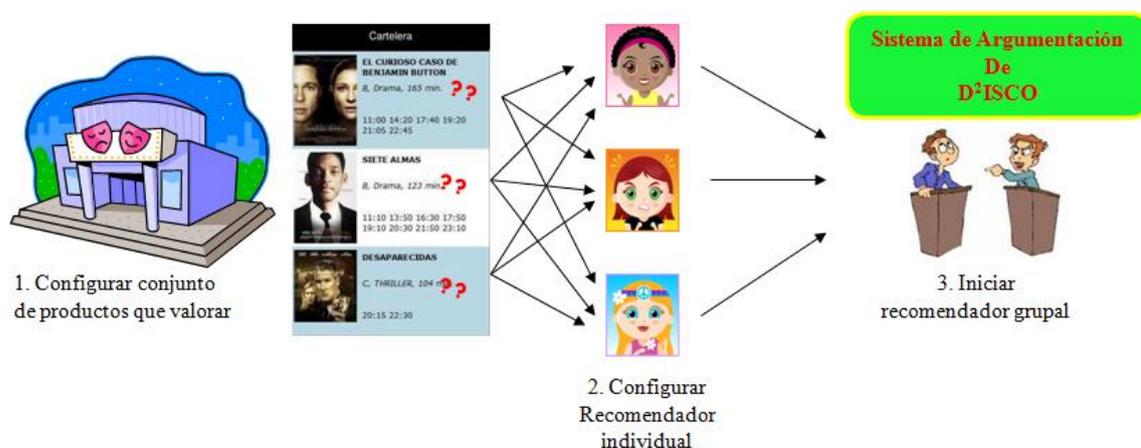


Figura 3.11: Segundo paso: configuración de las queries y el recomendador individual

en las conexiones entre agentes, problema que fue resuelto, por lo que ahora se permite cualquier tipo de configuración en las conexiones entre agentes. Una vez ya está configurado el sistema inicial, un agente del grupo, elegido aleatoriamente irá lanzando las queries una a una, a los agentes a los que está conectado y estos le devolverán el resultado de argumentar con los otros agentes a los que cada uno de ellos está conectado siguiendo el proceso de argumentación explicado en la Sección 3.6 y en González-Sanz et al. (2009).

Cada agente que reciba una query deberá valorarla individualmente, para ello, realizará una consulta al recomendador individual, y obtendrá dos valores, la medida de similitud entre la query lanzada y el caso del recomendador individual y la valoración del usuario del caso recomendado. Para obtener la valoración individual que finalmente le da cada usuario a una query determinada, estos dos valores obtenidos a través del recomendador individual se introducirán en un sistema Fuzzy (Fazlollahi et al., 2000), que permite realizar deliberaciones sobre bases de casos no nítidos y elimina las restricciones que supone tener que utilizar Lógicas Descriptivas sobre los casos. Una vez obtenemos la respuesta del sistema Fuzzy consideramos que esta es la valoración final de ese usuario para ese producto en particular. Ahora deberán argumentar y contra-argumentar sus respectivas valoraciones el agente que lanzó la query y el agente que la recibió para proporcionar al final una única solución consensuada. Para calcular el valor resultado de cada argumentación entre dos usuarios u_i y u_j sobre un producto m se utiliza la siguiente fórmula:

$$\text{argumentacion}(u_i, u_j, m) = pd(u_i, u_j) \cdot (ir_{i,m} \cdot t_{ij}) \quad (3.13)$$

donde $pd(u_i, u_j) = (CMW_j - CMW_i) \cdot \alpha$, el valor α se ha seleccionado experimentalmente (tiene un valor de 5.0) y se utiliza para modificar el impacto

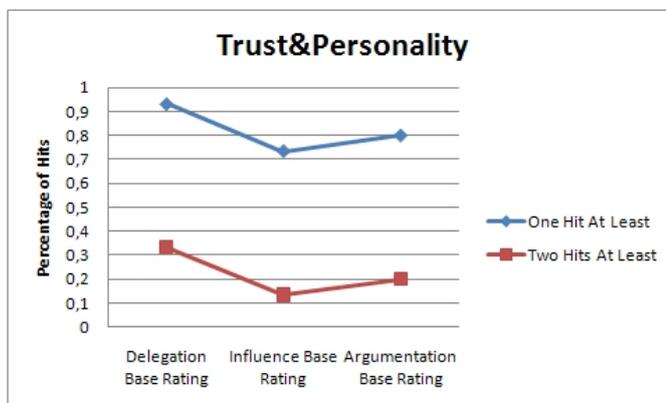


Figura 3.12: Comparación de los resultados obtenidos utilizando los tres métodos propuestos.

de las diferencias de personalidad en el rating modificado, t_{ij} es la confianza existente entre el usuario i y el j y $ir_{i,m}$ es el rating individual del usuario i sobre el producto m , valor que se ha obtenido tras pasar el resultado del recomendador individual para ese usuario y ese producto junto con la similitud de los productos valorados para obtener ese rating por el sistema fuzzy, como hemos mencionado anteriormente.

Este proceso, que es una variación del sistema de argumentación AMAL (Plaza y Ontañón, 2006), se repetirá entre todos los agentes conectados, hasta que le llegue la respuesta final al agente que lanzó la query en primer lugar y obtengamos así la respuesta del sistema. Los resultados preliminares obtenidos hasta el momento, realizados con la configuración del experimento que realizamos con usuarios reales, que se explica en la Sección 4.3, muestran que con este método, utilizando argumentación y sistemas multiagentes, se consigue una mejora respecto del método de ratings basados en influencia, explicado en la Sección 3.4.3. Sin embargo no se llegan a superar los resultados del método de ratings basados en delegación explicado en la Sección 3.4.2. Estos resultados preliminares los podemos ver en la Figura 3.12, que muestra el porcentaje de acierto del recomendador grupal usando los factores de personalidad y confianza, la línea de “One Hit At Least” muestra el porcentaje de veces que el recomendador grupal ofreció entre sus tres primeras opciones un producto que efectivamente el grupo habría seleccionado, y la línea “Two Hits At Least” muestra el porcentaje de veces que el recomendador grupal ofreció entre sus tres primeras opciones dos productos que efectivamente el grupo habría seleccionado, esta última opción es por tanto más difícil de conseguir luego el porcentaje es más bajo.

En la próxima sección explicaremos los pasos necesarios a seguir para implementar un sistema recomendador según nuestro método.

3.7. Reutilización de la metodología presentada a través de plantillas de jCOLIBRI

El proceso general que se debe seguir a la hora de realizar un recomendador grupal, es la plantilla que se muestra en la figura 3.13. Los pasos a seguir son:

- Setup: Realizar la configuración inicial, tanto de los usuarios individualmente como del grupo en sí.
- Obtain Individual Recommendations: Obtener recomendaciones individuales sobre los productos deseados para cada uno de los usuarios que forman parte del grupo a recomendar.
- CombinationIR: Seleccionar un algoritmo para combinar las recomendaciones individuales.
- PostProcessing: Procesar los resultados propuestos por el algoritmo elegido para la combinación.
- Propose Items: Proponer productos.

Más concretamente si se desea realizar un sistema recomendador según nuestro método, se deberán seguir los pasos que muestra la plantilla específica de GRUPITO que se muestra en la Figura 3.14. Seleccionar la configuración inicial:

- Load History: Cargar el historial de previas recomendaciones a grupos previamente formados, en este paso se cargan los productos que ya se han seleccionado grupalmente y los contenidos que están individualmente cada uno de los usuarios de esos grupos con esas selecciones.
- Add Personality: Decidir si se quiere utilizar el factor de personalidad, en cuyo caso es necesario realizar a los usuarios un test de personalidad, el que nosotros aquí proponemos es el test TKI (Thomas y Kilmann, 1974) sin embargo se puede utilizar cualquier otro test de personalidad que indique la actitud de las personas en situaciones conflictivas. Tras realizar el test este debe dar un valor normalizado entre 0 y 1 siendo 0 la representación de una persona muy afable y 1 la representación de una persona muy egoísta.
- Add Trust: Decidir si se quiere utilizar el factor de confianza entre usuarios, en cuyo caso es necesario que los usuarios pertenezcan a una red social a la que el desarrollador tenga acceso. De esa red social se deberán extraer los 10 factores indicados en la fórmula 3.4. Una vez tenemos los valores de esos factores se aplicará la fórmula 3.5 para obtener la confianza entre los dos usuarios deseados.

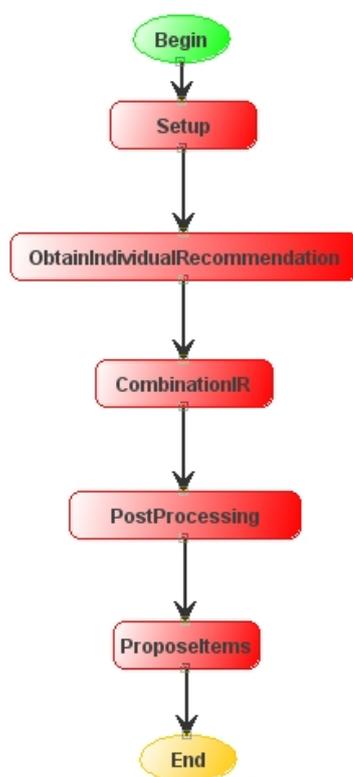


Figura 3.13: Plantilla general para la construcción de sistemas recomendadores grupales

- Obtain Group: Introducir la información necesaria para la recomendación grupal, es decir, un número de identificación de cada uno de los usuarios e indicar a qué grupos pertenecen, con qué otras personas van a realizar la actividad deseada, ya sea comprar algún producto, ir de viaje, al cine, a un restaurante...etc. Además es necesario pedir a los usuarios que valoren productos del mismo tipo a los que se desea recomendar. Estas valoraciones se guardarán como la base de datos personal de cada usuario. Y finalmente, seleccionar los productos a valorar grupalmente por los usuarios, es decir las queries a preguntar.

Una vez tenemos toda la información individual y grupal que forman la configuración inicial del recomendador, comenzamos el proceso de implementación del mismo:

- Obtain Individual Recommendations: Seleccionar un recomendador individual ya sea colaborativo, en cuyo caso se necesita una base de datos con otros usuarios que hayan valorado los productos a recomendar o basado en contenido, en cuyo caso se necesita una base de datos con información sobre las características de los productos. Este paso se

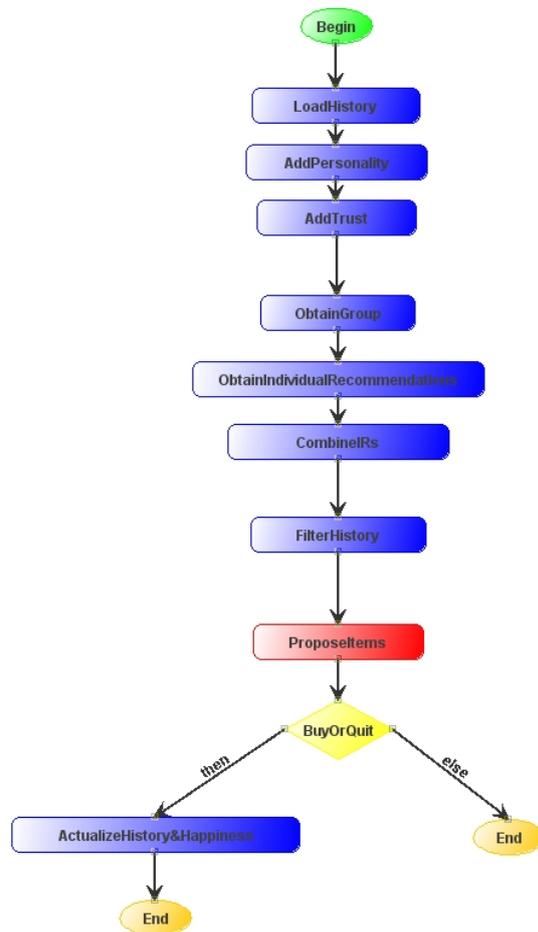


Figura 3.14: Plantilla específica para la construcción de nuestro método de sistemas recomendadores grupales

realizará con un método denominado *IRInvoker*

- **CombineIRs:** Seleccionar un algoritmo de combinación de resultados. Los disponibles métodos disponibles serán: estrategia de agregación promedio (AverSas), estrategia de minimización de la penalización (MinPen), estrategia de minimización de la miseria (LeastMis), Ratings basados en la personalidad con estrategia de agregación promedio (PBR-AverSas), Ratings basados en la personalidad con estrategia de minimización de la penalización (PBR-MinPen), Ratings basados en la personalidad con estrategia de minimización de la miseria (PBR-LeastMis), Ratings basados en la delegación (DBR), Ratings basados en la influencia (IBR), Ratings basados en la delegación con memoria (DBRM), Ratings basados en la influencia con memoria (IBRM).

- **Filter History:** Realizar el proceso de filtrado de memoria, para no repetir productos previamente seleccionados. Este proceso sólo se realizará en caso de que se haya seleccionado un método de combinación de resultados que tenga en cuenta la memoria.
- **Propose Items:** Proponer una serie de productos. Y en caso de que sean aceptados por los usuarios actualizar la ventana de memoria y el grado de satisfacción grupal.

Resumen

En este capítulo hemos desarrollado nuestras teorías para mejorar los sistemas de recomendación para grupos. Para conseguir este objetivo hemos introducido diversos factores novedosos, como son la personalidad, los factores sociales, recomendaciones con memoria y modelos distribuidos y argumentación. Además hemos resumido nuestra aportación en forma de plantilla de forma que sea posible reusar el método aquí propuesto. A continuación llevaremos estas teorías a casos prácticos, para comprobar su validez y alcance.

Capítulo 4

Caso de estudio: Recomendador para películas

4.1. Introducción

Para comprobar el funcionamiento de nuestros métodos, y observar si efectivamente suponen una mejora respecto de los recomendadores para grupos previamente existentes, hemos evaluado nuestros algoritmos usando el dominio de las películas. Hemos elegido este dominio porque es una área con diversas fuentes de datos (datasets) disponibles, luego obtener información acerca de películas se convertía en una tarea relativamente fácil pues sólo había que integrar una base de datos de las existentes a nuestro sistema. También influyó el conocimiento general que cualquier usuario tiene del mundo del cine, que le permite conocer el dominio sin necesidad de ser un experto, con lo que el uso del sistema y la evaluación de resultados resulta más sencilla. A continuación detallamos cómo ha sido la configuración de nuestros experimentos, en qué han consistido, qué esperábamos encontrar y qué hemos encontrado.

4.2. Experimento con datos sintéticos

Inicialmente realizamos un experimento en el que utilizamos datos sintéticos. El motivo de utilizar inicialmente estos datos generados y no datos obtenidos de usuarios reales (como hicimos más tarde) fue que pretendíamos explorar casos extremos que se pudieran dar en situaciones de conflicto a la hora de recomendar a grupos de personas. Para ello necesitábamos tener control de la distribución de los datos, cosa que no ocurría con datos reales. Nuestro experimento se dividió en las siguientes partes: (1) generamos aleatoriamente grupos de usuarios con diferentes tipos de personalidades y de topología social; (2) desarrollamos un recomendador individual siguiendo

el enfoque colaborativo, como se explica en en la sección 3.3; (3) creamos 4 sistemas de recomendadores para grupos distintos que utilizaban el recomendador individual: el primer sistema recomendador era una base que sólo realizaba agregación de preferencias, otro que utilizaba únicamente el dato sobre la personalidad de cada individuo, como se explica en la sección 3.4.1, un recomendador grupal usando sólo la topología social, tal y como se refleja en la ecuación 4.1 y poniendo el valor de la personalidad $pd(u_i, u_j)$ a 0, y el último que tiene en cuenta ambos factores, la personalidad y la topología social. Finalmente, (4) comparamos los resultados obtenidos con los diferentes recomendadores y diversas configuraciones de los grupos sintéticos, pues queríamos ver cómo estas configuraciones influían en la recomendación final.

$$sgr(u_i, m) = \frac{1}{|G|} \sum_{j \in G \wedge j \neq i} (ir_{i,m} + pd(u_i, u_j) + confianza(u_i, u_j)) \quad (4.1)$$

donde $pd(u_i, u_j) = (CMW_j - CMW_i) * \alpha$, $confianza(u_i, u_j)$ indica si los usuarios u_i y u_j son amigos, amigos de amigos o no se conocen y el valor α se ha seleccionado experimentalmente (se ajustó a 5.0) y se utiliza para modificar el impacto de las diferencias de personalidad en el rating modificado. Para generar los grupos de usuarios utilizamos una colección de 100 personas. A cada persona le asignamos un valor aleatorio CMW, para reflejar su personalidad, en los trabajos con valores reales, este es el valor que se calcula usando los resultados del test TKI de personalidad (Thomas y Kilmann, 1974). Como el experimento es sintético no tenemos valores CMW reales, por lo que los generamos. El valor CMW tiene un rango de [1,0], siendo 1 el reflejo de una persona muy egoísta y 0 el de una persona muy cooperativa. En función del rango donde se encontraba este valor teníamos básicamente 5 tipos de personalidades: muy egoísta, egoísta, tolerante, cooperativa y muy cooperativa. Por ejemplo para considerar a una persona muy egoísta su valor CMW debía estar contenido en un rango de [1.9,1.5]. Para poder estudiar los efectos de las distintas personalidades generemos 20 usuarios de cada tipo de personalidad.

Agrupamos a los usuarios en sets de 3, 5, 10, 15, 20 y 40 personas. Para cada tamaño de grupo seleccionamos los componentes del grupo de modo que las *distribuciones de la personalidades* presentaran todas las combinaciones posibles: muy egoísta, egoísta, tolerante, cooperativa y muy cooperativa, muy egoísta & muy cooperativa, muy egoísta & tolerante, y así hasta realizar las 13 combinaciones distintas. Luego al final, contábamos con 76 grupos (13 distribuciones distintas para cada tamaño menos para el de 40 personas que teníamos solo 11 debido a la semejanza de personalidades en grupos tan grandes).

Para simular la red social que tenía cada grupo generamos conexiones de amistad entre ellos aleatoriamente. Este proceso lo repetimos varias veces

hasta que conseguimos una combinación que reflejaba los resultados que esperábamos. La *función de confianza* que aquí utilizamos únicamente analizaba las conexiones entre los miembros del grupo para calcular la confianza dependiendo de la distancia dentro de la red social simulada. Aunque se podía utilizar cualquier algoritmo más complejo para calcular la confianza dentro de la red, o utilizar otros factores sociales como veremos más adelante, elegimos este enfoque sencillo:

$$\text{confianza}(a, b) = \begin{cases} 1, & \text{si distancia}(a,b)=1; \\ 0,5, & \text{si distancia}(a,b)=2; \\ 0, & \text{e.c.c.} \end{cases}$$

El siguiente elemento que necesitábamos para nuestro experimento era contar con una función de evaluación para saber que habría pasado si esto hubiese sido una situación real. Esta función era necesaria para poder evaluar la precisión de la recomendación. Para conseguirlo teníamos que deducir qué películas habrían escogido los usuarios individualmente dada una determinada cartelera de cine, y después cual de esas películas habría elegido el grupo para ver finalmente.

Nuestra función de evaluación se basa en los contenidos de las películas. Como hemos explicado anteriormente, los sistemas recomendadores que estamos estudiando son basados en ratings (colaborativos). Esto implica que el contenido y las características que describen cada uno de los elementos a recomendar no se tienen en cuenta en el proceso de recomendación. Si utilizásemos las características de cada producto estaríamos utilizando una técnica de recomendación distinta, basada en contenido (Bridge et al., 2006) en vez de las recomendaciones basadas en filtrado colaborativo. Este enfoque compara características de productos nuevos, con las características de otros productos que el usuario ya ha seleccionado, y las clasifica en función de su similitud. Por tanto, podemos usar esta clase de técnica basada en contenido para determinar cuáles serían las películas que al usuario realmente le gustan, y más tarde, implementar nuestra función de evaluación.

Para conseguir esto, seleccionamos una lista de 50 películas de la base de datos de MovieLens (Bobadilla et al., 2009) y las calificamos (con un rango entre 0.0 y 5.0) para cada uno de los usuarios en función de un perfil aleatorio que se les asignó a cada uno. Estos perfiles se construyeron de acuerdo a las preferencias típicas que tienen las personas en películas en función de su edad, sexo, preferencias, etc. Personas intelectuales, chicas quinceañeras, infantiles, amantes de las artes marciales, mujeres clásicas, hombres clásicos, fueron algunos de los perfiles que asignamos aleatoriamente usando los datos de MovieLens. Por ejemplo, los ratings que daría una persona con un perfil infantil serían muy altos a las películas de animación, niños o musicales y muy bajos a las de drama, terror o documentales.

A continuación, para cada usuario seleccionamos las películas que hubiesen

valorado con un 4.5 o más y con ellas definimos el conjunto de películas favoritas del usuario. Con esta información, utilizamos test de similaridad basado en contenido para organizar una cartelera de cine en orden de preferencia. Una vez que teníamos la colección de películas organizada, seleccionamos las tres primeras y las marcamos como las favoritas individualmente *if*. Con esto ya teníamos cuál hubiese sido la elección individual de cada usuario si esto hubiese sido una situación real, ahora necesitábamos saber la reacción grupal. Puesto que ya sabíamos cuales serían las películas por la que los usuarios debatirían individualmente, simulamos la situación que se daría en la vida real cuando todos discutiesen sus preferencias, teniendo en cuenta las personalidades de cada uno y la amistad entre ellos. Finalmente obtuvimos las tres películas que hubiese elegido el grupo, las favoritas del grupo *rgf*. Cuando calculamos la eficiencia del recomendador comparamos cuantas de las tres primeras recomendadas pertenecían a este grupo *rgf*.

Una vez que teníamos toda esta información construimos los sistemas recomendadores que explicamos anteriormente utilizando el framework jCOLIBRI (Recio-García et al., 2008).

Para el recomendador individual, que utiliza un enfoque colaborativo utilizamos una colección de usuarios con los que comparar a los nuestros de la base de datos de MovieLens. Seleccionamos a los usuarios de MovieLens que habían visto algunas de las 50 películas que nuestros usuarios habían valorado. Para acortar la lista de películas recomendadas de entre todas las que nos devolvía el recomendador individual sólo tuvimos en cuenta las 15 que pertenecían a la cartelera de cine que previamente habíamos seleccionado para calcular las preferencias de los usuarios, estas películas también habían sido elegidas de entre la base de datos de MovieLens. Una vez que ya teníamos la recomendación para cada usuario sobre la cartelera de cine, el siguiente paso era implementar los recomendadores para grupos.

Para la construcción de los sistemas recomendadores empleamos los elementos que previamente se han descrito: los grupos de usuarios (con las diferentes personalidades y topologías sociales) y el conjunto de evaluación *rgf* para cada grupo.

Primero construimos el recomendador grupal estándar utilizando las funciones de agregación típicas: maximización de la satisfacción media (average satisfaction), minimizar la miseria (least misery) y minimizar la penalización (minimizing penalization), explicadas en la sección 2.1.2. Los resultados obtenidos por este recomendador fueron la base de nuestro experimento.

Una vez que tuvimos construido el recomendador estándar implementamos los otros 3 para las 3 funciones de agregación: el que utilizaba únicamente el dato sobre la personalidad de cada individuo, como se explica en la sección 3.2, el que sólo usaba la topología social, que consistía en implementar la ecuación 4.1 poniendo los valores relativos a la personalidad a 0, y el que combinaba los dos anteriores, que implementa la ecuación 4.1.

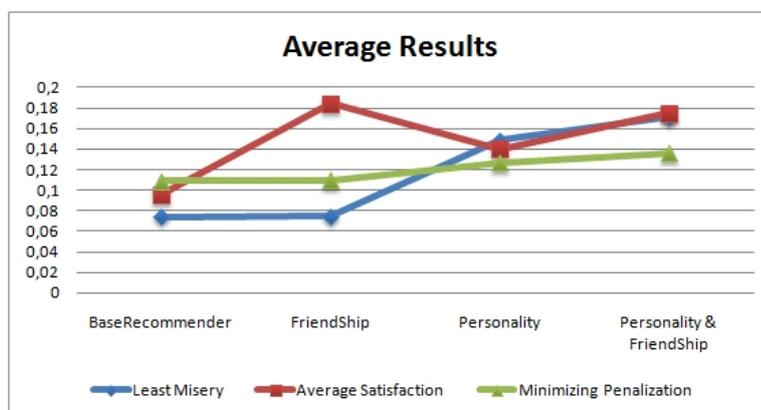


Figura 4.1: Resultados Globales

4.2.1. Resultados del experimento

Como el recomendador individual se construyó con el único propósito de servir de base para el recomendador grupal, no analizaremos su actuación. Sin embargo, cabe destacar que siempre que se mejore este sistema, el sistema en general mejorará ya que todo el sistema recomendador de grupos está basado en las preferencias individuales de cada usuario que nos proporciona el recomendador individual. Otro factor importante a tener en cuenta a la hora de evaluar el funcionamiento del recomendador es la configuración de la colección de películas que forman la cartelera de cine y su similitud con la colección de 50 películas que los usuarios han valorado. Todos estos datos fueron elegidos aleatoriamente por lo que el rendimiento del recomendador puede variar en función de estos valores.

Las gráficas que mostramos en este apartado que analizan las diferentes estrategias de agregación, tienen cuatro columnas diferentes que representan los resultados de los recomendadores cuando se consideran sus relaciones de amistad y personalidad – columna *Personality & Friendship* –, sólo sus relaciones de amistad –columna *Friendship* –, sólo su personalidad –columna *Personality* –, y sólo el recomendador grupal base *BaseRecommender*, en ellas estamos midiendo cuántas de las recomendadas como favoritas del grupo, gf , están dentro de las favoritas reales del grupo, rgf .

La gráfica 4.1 resume los resultados globales para cada una de las tres estrategias de agregación cuando se consideran las relaciones de amistad y personalidad y los compara con los resultados que previamente se podían tener en los sistemas recomendadores sin aplicar nuestro método. La estrategia Least Misery (minimización de la miseria) tenía un 14,9% de aciertos sin tener en cuenta la amistad entre los miembros del grupo –la función de confianza–, y un 17,2% si sí que la teníamos en cuenta. Para el resto de funciones de agregación encontramos resultados similares, véase, para la estrategia average satisfaction (maximización de la satisfacción media) que



Figura 4.2: Resultados con la estrategia Least Misery en función del tamaño del grupo.

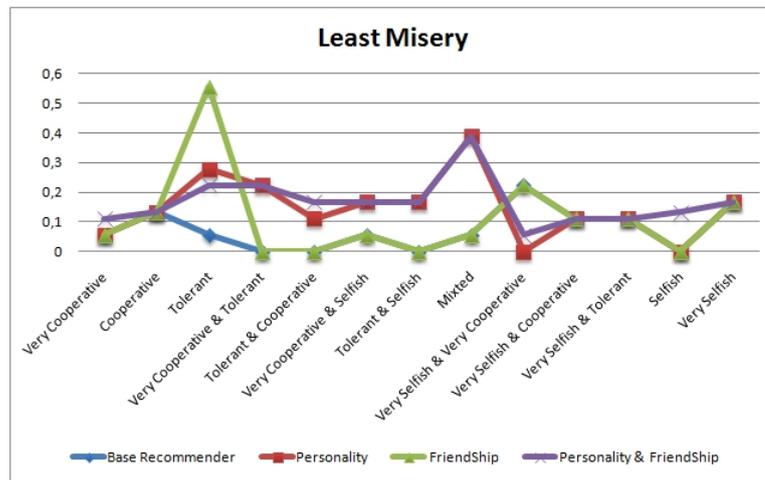


Figura 4.3: Resultados con la estrategia Least Misery en función de la distribución de la personalidad.

tenía un 14,03 % de acierto anteriormente y cuando combinábamos la personalidad y la confianza se obtenía un 17,6 %. Y para la estrategia minimizing penalization (minimización de la penalización) obtuvimos un 12,7 % sin la función de confianza contra un 13,6 % cuando la incluíamos. De este modo podemos observar que la mejor función de agregación es la maximización de la satisfacción media. Estos resultados nos permiten concluir que los sistemas recomendadores para grupos mejoran cuando se utilizan topologías basadas en relaciones sociales.

En cuanto al tamaño del grupo podemos concluir claramente que nuestro algoritmo de recomendación obtiene mejores resultados para grupos pequeños que para los más grandes. Cuando comparamos la relación entre el tamaño del grupo y el porcentaje de acierto, lo que muestran las gráficas

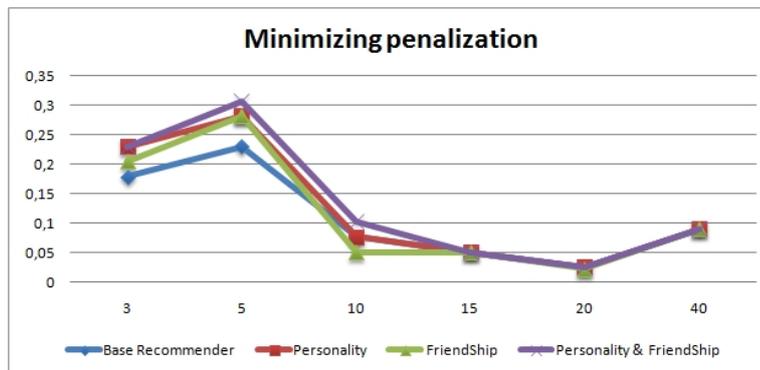


Figura 4.4: Resultados con la estrategia Minimizing Penalization en función del tamaño del grupo.

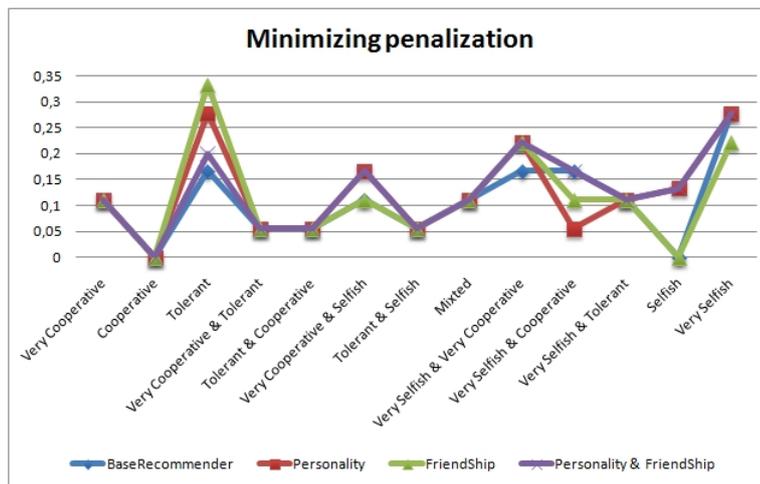


Figura 4.5: Resultados con la estrategia Minimizing Penalization en función de la distribución de la personalidad.

4.2, 4.4 y 4.6, observamos que es un factor determinante, y que cuanto más gente tenía el grupo, peor era la recomendación. Esta situación es muy comprensible ya que un grupo con más gente poseerá más opiniones distintas y por tanto será mucho más difícil llegar a una solución consensuada.

Por otro lado, cuando comparamos cómo afecta a la eficiencia del recomendador la distribución de personalidades en el grupo, en las gráficas 4.3, 4.5 and 4.7, podemos ver que el porcentaje de acierto es mayor en torno al centro de la gráfica, este rango coincide con los grupos que tienen una personalidad más variada, luego las personalidades heterogéneas suponen una mejora en la actuación del recomendador. También podemos observar que en la parte derecha de las gráficas, que es la zona que representa a los grupos más autoritarios o con personas más egoístas, es decir tienen al menos un rol de

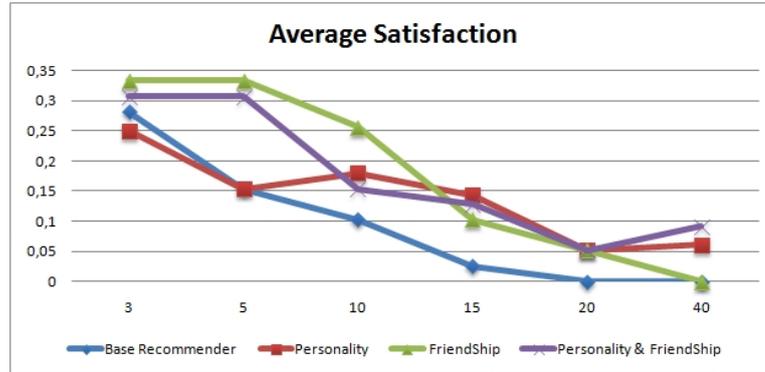


Figura 4.6: Resultados con la estrategia Average Satisfaction en función del tamaño del grupo.

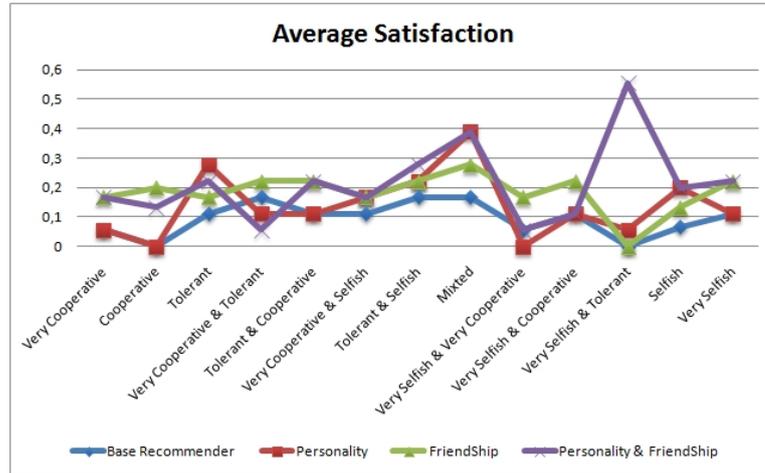


Figura 4.7: Resultados con la estrategia Average Satisfaction en función de la distribución de la personalidad.

líder en el grupo, tiene un mejor porcentaje que los grupos a la izquierda que son los que representan a los grupos formados por personas muy cooperativas. Esto tiene sentido ya que en esos grupos nadie impondrá su opinión y estarán delegando sin llegar a una decisión final. En las gráficas 4.5 y 4.3, podemos ver que hay un pico en el grupo que representa a las personas tolerantes "Tolerant". Estas personas no tienen personalidades extremas por lo que los resultados para este grupo son mejores. Sus personalidades representan los valores medios del factor de personalidad CMW, ni muy egoístas ni muy cooperativas. Es natural que en este grupo los resultados sean mejores porque representan combinaciones de personalidad más realistas, el 65 % de las personas que preguntamos para otro de nuestros experimentos (detallado en la siguiente sección), tenía una personalidad perteneciente al rango que

llamamos "tolerante".

Finalmente podemos concluir que con este experimento observamos una mejora en las recomendaciones para grupos de hasta un 4 %.

4.3. Experimento con datos reales

Una vez que hicimos el experimento con los datos sintéticos, decidimos que debíamos probar dos cosas, la primera era que los resultados de los experimentos anteriores fueran fiables, es decir que la simulación fuera realista, de ese modo las personalidades de diversas personas y obtener los mismos resultados que con personas reales. La segunda era realizar un investigación más a fondo sobre cómo podíamos mejorar aún más las recomendaciones teniendo en cuenta la personalidad y la confianza según las redes sociales. Para hacerlo en vez estudiar simplemente la distancia social entre las personas de un grupo, con la función 4.2 haríamos un análisis más detallado sobre la confianza entre personas, como explicábamos en la sección 3.3. Para conseguir estos dos objetivos decidimos realizar un experimento con usuarios reales.

Para obtener los datos que nuestro experimento necesitaba, creamos dos eventos en dos redes sociales distintas Facebook¹ y Tuenti². En estos eventos les pedimos a algunos de nuestros amigos en la red social que contestarán a 3 cuestionarios³.

El primer cuestionario nos servía para obtener el perfil de personalidad, para ello, les hicimos 30 preguntas del test de personalidad TKI (Thomas y Kilmann, 1974), podemos ver algunas de estas preguntas en la figura 3.2. El segundo cuestionario nos proporcionaba las preferencias individuales de cada usuario en el dominio de las películas. Los usuarios tuvieron que evaluar 50 películas heterogéneas seleccionadas de entre la base de datos de MovieLens (Bobadilla et al., 2009) (las valoraban con un rango de 0.0 a 5.0). De esta manera fuimos capaces de realizar el algoritmo de filtrado colaborativo que calcula las predicciones individuales $ir_{i,m}$, explicado en el experimento anterior, sección 4.2. En la figura 4.8 podemos ver una muestra de algunas de las películas que calificaron nuestros usuarios.

Finalmente, el tercer test le pedía a los usuarios que eligiesen qué 3 películas les gustaban o les habría gustado más ver de entre 15 películas recientes (del año 2009) que representaban una cartelera de cine. Esta cartelera se eligió de forma heterogénea de entre la base de datos de MovieLens. Las tres que eligieron representaban sus tres favoritas. En la figura 4.9 se puede ver una

¹<http://www.facebook.com>

²<http://www.tuenti.com>. Tuenti es la red social más popular en España para los veinteañeros.

³Los cuestionarios están disponibles en <http://www.lara.warhalla.com/> (en Español.)

Cuestionario 2

Califica las siguientes películas

	Batman: El caballero oscuro <input type="text" value="4"/>
	El señor de los anillos: El Retorno del Rey <input type="text" value="3.5"/>
	El Ultimatum de Bourne <input type="text" value="No la he visto"/>

Figura 4.8: Muestra del test de 50 películas que rellenaron nuestros usuarios en un evento de Facebook.

muestra de algunas películas que aparecían en la cartelera que les ofrecimos a nuestros usuarios. El objetivo de este test era tener la información necesaria para luego poder medir la eficacia del recomendador individual. las respuestas de cada uno de los cuestionarios, que recibíamos en el correo, se analizaron para definir el perfil de usuario de cada uno de los participantes. Al final 58 personas participaron en nuestro experimento.

Lo siguiente que necesitábamos era una función de evaluación para medir la precisión del recomendador de grupos y compararlo con qué habría pasado si se hubiese tratado de una situación real. Para poder hacer esto, creamos diferentes grupos con nuestros participantes y les pedimos que simulasen que iban al cine juntos. Les proporcionamos una lista de 15 películas que representaban una cartelera de cine (las mismas 15 que estaban en el tercer cuestionario) y les pedimos que eligiesen que 3 películas irían a ver juntos. Conseguimos reclutar a 15 grupos de 9, 5 y 3 miembros. Las tres películas que cada grupo eligió se guardaron como el conjunto de *favoritas reales del grupo*, rgf . De este modo, para evaluar la eficiencia de nuestro recomendador podemos comparar la colección de las tres primeras películas que nos ofrece el recomendador –el conjunto gf – con las preferencias reales rgf . Utilizamos dos funciones de evaluación para medir el número de películas en gf que también estaban en rgf . La función de evaluación *one-hit* devuelve cuantas



Figura 4.9: Muestra de la cartelera de cine que valoraron nuestros usuarios.

veces el recomendador propone al menos una película que esté en rgf . Lo que significa que, obtenemos gf para cada grupo y comprobamos si hay al menos una de estas películas pertenece a las preferencias reales del grupo. Cuando tenemos este dato devolvemos el porcentaje de éxito. Por ejemplo un 90 % de eficiencia usando el *one-hit* representa que el recomendador ha sugerido al menos una película correcta de entre el 90 % de los grupos evaluados. La segunda función, *two-hits*, representa cuantas veces el conjunto estimado de favoritas gf contiene al menos dos películas que estén en rgf . Obviamente, esta segunda medida es mucho más difícil de conseguir.

Una vez que tenemos los resultados de los test, necesitamos los factores de personalidad y confianza. Para obtener el valor que representa la personalidad calculamos el CMW_i (*Conflict Mode Weight*) de los resultados obtenidos en el primer test de los usuarios.

A continuación calculamos el factor de confianza entre dos usuarios i y j , que representaremos como t_{ij} , para ello, usamos los factores detallados en la sección 3.3. Más específicamente utilizamos los siguientes factores concretos de confianza:

- f_1 : Distancia en la red social: 1.0 si son amigos directos, 0.5 si son amigos de amigos y 0.1 e.c.c.

- f_2 : Número de amigos en común:

$$f_2(u_i, u_j) = \begin{cases} 1,0, & \text{si } >25 \text{ amigos en común;} \\ 0,7, & \text{si } >15 \text{ amigos en común;} \\ 0,5, & \text{si } >10 \text{ amigos en común;} \\ 0,3, & \text{si } >5 \text{ amigos en común;} \\ 0,1, & \text{si } <5 \text{ amigos en común;} \end{cases}$$

- f_3 : Intensidad de la relación: cómo de a menudo se escriben mutuamente en los muros del otro.

$$f_3(u_i, u_j) = \begin{cases} 1,0, & \text{Todos los días;} \\ 0,7, & \text{Casi todos los días;} \\ 0,5, & \text{Cada semana;} \\ 0,3, & \text{Cada mes;} \\ 0,1, & \text{Nunca;} \end{cases}$$

- f_4 : Intimidad de la relación: Clasificamos las relaciones buscando palabras clave que representan los siguientes niveles de intimidad: muy afectivo, familiar/cariños, amigable, profesional, causal, ninguno.

$$f_4(u_i, u_j) = \begin{cases} 1,0, & \text{Muy afectivo;} \\ 0,7, & \text{Desenfadado;} \\ 0,5, & \text{Amigable;} \\ 0,3, & \text{Profesional;} \\ 0,1, & \text{Distante;} \end{cases}$$

- f_5 : Duración: desde hace cuanto se conocen.

$$f_5(u_i, u_j) = \begin{cases} 1,0, & \text{si } >10 \text{ años;} \\ 0,7, & \text{si } >5 \text{ años;} \\ 0,5, & \text{si } >3 \text{ años;} \\ 0,3, & \text{si } >1 \text{ año;} \\ 0,1, & \text{si } <1 \text{ año;} \end{cases}$$

- f_6 : Servicios recíprocos: número de videos/canciones/webs posteados. En Facebook también tuvimos en cuenta los juegos/ aplicaciones que comparten como (like Pet Society, Mafia wars, Music challenge, among others).

- f_7 : Variable estructural: intereses comunes descritos en el perfil de usuario, como películas, libros o intereses generales. De Tuenti también tuvimos en cuenta los lugares a los que van de fiesta, y de Facebook a cuántos grupos en común se habían unido o eran fans.

- f_8 : Distancia social: cuantas de la siguientes propiedades comparten: políticas, educacionales, religiosas y demográficas.

$$f_8(u_i, u_j) = \begin{cases} 1,0, & \text{si 4 propiedades en común;} \\ 0,7, & \text{si 3 propiedades en común;} \\ 0,5, & \text{si 2 propiedades en común;} \\ 0,3, & \text{si 1 propiedad en común;} \\ 0,1, & \text{Sin propiedades en común;} \end{cases}$$

- f_9 : Status: valor que depende de la clase de status: pareja (1.0), familia (0.7), mejores amigos (0.5), amigos (0.3), apenas conocidos (0.1).
- f_{10} : Fotos: Porcentaje de fotos en las que aparecen juntos.

El valor final de confianza t_{ij} es una media ponderada de los valores previamente descritos, tal y como describimos en la sección 3.3, ecuación 3.5. Estos pesos se han obtenido experimentalmente para cada estrategia de recomendación grupal por medio de un algoritmo genético (GA). Nuestro GA maneja una población de vectores de pesos (w_i). Estos vectores se combinan y se mutan para maximizar una función de aptitud. Hemos utilizado dos funciones diferentes de aptitud de acuerdo con las estrategias descritas en las secciones 3.4.2 y 3.4.3 : *Ratings basados en la delegación* y *Ratings basados en la influencia*. Por tanto, los individuos del GA (los pesos) se usan para calcular el factor de confianza t_{ij} que se requiere en estos enfoques. Esta función de aptitud se evalúa utilizando la medida one-hit. Gracias a este enfoque de evaluación hemos sido capaces de explorar una gran cantidad de combinaciones de pesos y hemos obtenido la mejor combinación para cada algoritmo de recomendación para grupos.

La construcción del recomendador se realizó en los siguientes pasos. Primero, construimos el recomendador individual para películas usando el framework jCOLIBRI. Se trata del mismo tipo de recomendador que construimos para el experimento anterior 4.2. Utiliza las valoraciones de la base de datos de MovieLens y los ratings que nuestros voluntarios dieron en el segundo test. Seleccionamos de entre los usuarios de MovieLens, aquellos que habían visto al menos 15 de las 50 películas que nuestros usuarios habían valorado, y que también hubiesen visto más de tres de la cartelera de nuestro tercer cuestionario. Con estos datos, el recomendador individual nos devuelve qué películas, a parte de las 50 que nuestros usuarios ya valoraron, son las que han tenido mayores puntuaciones por los usuarios de la base de datos que son más similares a cada uno de nuestros usuarios. Las películas con valoraciones altas conforman nuestra lista del recomendador individual. Para acortar el tamaño de esa lista, solo seleccionamos una colección de 15 películas, que son las mismas que valoraron nuestros usuarios en el tercer test. La evaluación de este recomendador también se realiza utilizando las medidas de *one-hit* y *two-hits*.

El siguiente paso es construir el sistema de recomendación para grupos. Primero, construimos el recomendador grupal estándar utilizando la función de agregación de average satisfaction (maximización de la satisfacción media). Elegimos esta función entre otras cosas porque era la que mejor resultados proporcionaba en el experimento anterior. Nos referimos a él como *Base* (ecuación 3.6). Luego implementamos los enfoques descritos en las secciones 3.4.1, 3.4.2 y 3.4.3 para construir otros tres recomendadores. El recomendador basado en personalidad (personality aware recommender), al que nos referimos como *Personality* utiliza únicamente los datos sobre la personalidad y sigue el enfoque *Ratings basados en la personalidad* (Ecuación 3.8). El segundo recomendador (al que nos referimos como *Trust*) sólo tiene en cuenta los valores sobre la confianza t_{ij} (los valores CMW están a 0), y finalmente, el recomendador *Personality & Trust* tiene en cuenta ambos factores, la confianza social t_{ij} y la personalidad CMW_i de cada componente del grupo. Los últimos dos recomendadores (*Trust* y *Personality & Trust*) se pueden ejecutar utilizando los dos algoritmos previamente comentados: *delegation based rating* y *influence based rating* (Ecuaciones 3.9 y 3.10).

En el siguiente apartado comentamos los resultados que obtuvimos.

4.3.1. Resultados del experimento

Los resultados del recomendador individual no los comentaremos en detalle ya que se construyó únicamente como base para el recomendador grupal, por lo que no estamos interesados en mejorar sus resultados. En este experimento en concreto con esta configuración de películas, obtenía un 58,6 % de acierto con la métrica de evaluación *one-hit* y un 13,79 % con la de *two-hits*.

La gráfica 4.10 nos muestra el rendimiento de nuestros recomendadores para grupos (considerando personalidad y confianza, sólo confianza, sólo personalidad y el recomendador para grupos base). La gráfica resume el promedio de resultados teniendo en cuenta todos los grupos. De ella podemos concluir que las dos estrategias empleadas para modificar los ratings individuales tienen una actuación similar aunque la de ratings basados en la delegación es ligeramente mejor. Estas gráficas confirman nuestras teorías y muestran que mejoramos las recomendaciones teniendo en cuenta la confianza social y la personalidad de cada individuo. Usando la métrica de evaluación *one-hit* obtenemos una mejor del 17.8 % respecto del recomendador *Base*. Es más, con la medida de evaluación más dura, *two-hits*, devuelve una mejora cercana al 35 %.

Las gráficas 4.11 y 4.12 detallan los resultados de cada algoritmo y tamaño de grupo. Estos gráficos ilustran cómo usando la personalidad se mejora la eficacia, pero si usamos sólo la confianza no se obtienen mejores resultados que con el recomendador base. Sin embargo cuando los combinamos los dos obtenemos la mejor actuación. También hemos estudiado la influencia del tamaño del grupo. Si observamos la relación entre tamaño-

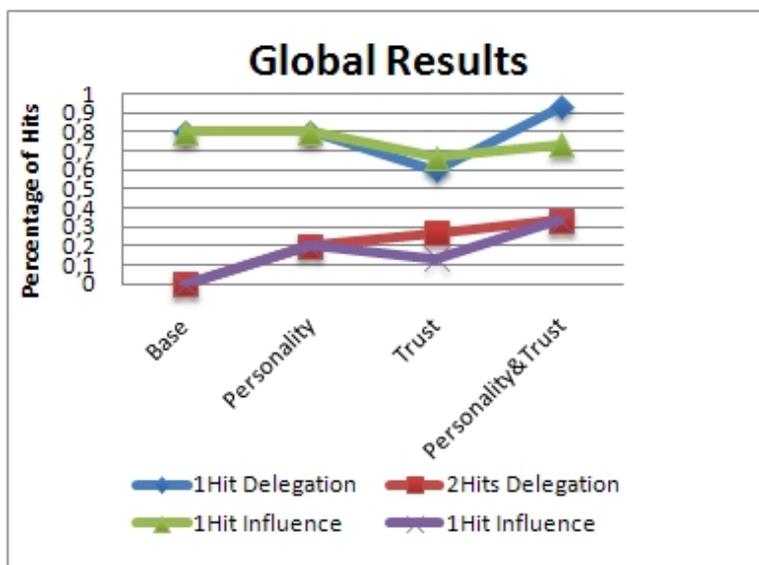


Figura 4.10: Tasa de aciertos para los 4 sistemas recomendadores con nuestros dos enfoques

porcentaje de aciertos en las gráficas 4.11 y 4.12 no podemos observar un comportamiento genérico. Generalmente, cuanto más gente tiene el grupo, peor es la recomendación. En general está bastante claro que nuestro algoritmo realiza mejores recomendaciones para grupos de menor tamaño. Aquí corroboramos lo que concluimos en el anterior experimento (apartado 4.2.1), este factor es lógico, ya que cuanto más gente habrá más opiniones diferentes y por tanto es más difícil llegar a un consenso.

En cuanto a la importancia de cada uno de los factores que conforman el valor de confianza, podemos ver en la gráfica 4.13 cómo se han tenido en cuenta para maximizar la actuación del recomendador. Estos pesos (w_i), son los que obtuvimos usando el Algoritmo Genético (GA). Podemos ver que el factor más relevante es el número de amigos en común, seguido de las fotos, la duración, los intereses comunes y el status. Estos pesos son los que hemos aplicado para los ratings basados en delegación. Por otro lado, cuando realizamos el estudio sobre los factores que amplificaban los resultados del recomendador con los ratings basados en influencia, descubrimos que el mejor era el f_{10} (fotos en común) con una importancia del 0.97%. Repetimos los resultados modificando los parámetros del GA (tamaño de la población, repeticiones, etc) y confirmamos que este resultado y su valor siempre salía muy alto. Sin embargo podemos explicarlo, pues es cierto que este factor (el porcentaje de fotos que aparecen juntos) resume muchos de otros factores: si dos personas aparecen juntas en muchas fotos es porque tienen una larga relación (duración), están muy unidos (status), van juntos a los mismos sitios (intereses), etc.

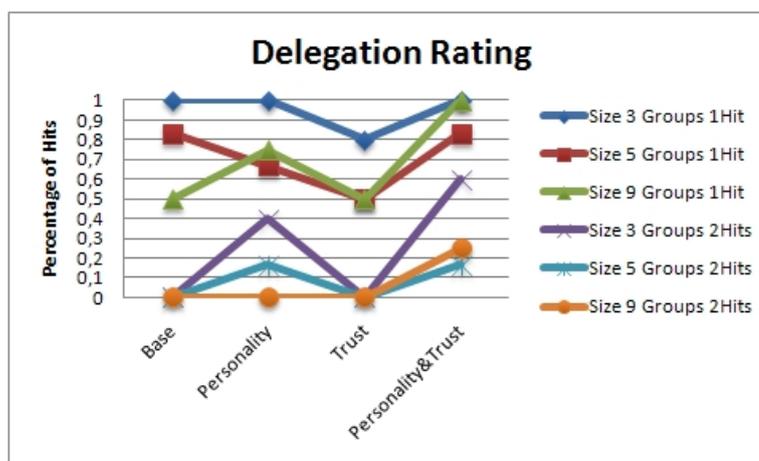


Figura 4.11: Tasa de aciertos para cada tamaño de grupo con los Ratings basados en la delegación

Finalmente, también hemos estudiado si la eficiencia de la recomendación estaba ligada a la distribución de personalidades o de los factores de confianza. Para hacerlo, calculamos la desviación estándar de los valores de personalidad y de los factores de confianza para cada grupo. Después de realizar varios estudios estadísticos, hemos concluido que no existe ninguna correlación entre las distribución de las personalidades y la eficiencia de nuestros algoritmos ni entre la distribución de la confianza y la eficiencia de nuestros algoritmos.

4.4. Experimento con datos reales y memoria

Para concluir nuestros experimentos, decidimos incluir al recomendador para grupos una memoria de los anteriores resultados. De esto modo pretendíamos cubrir dos necesidades para las futuras recomendaciones: no repetir recomendaciones al mismo grupo y poder proporcionar un cierto grado de justicia. Para la primera necesidad nos limitamos simplemente a tener guardada cada película que se recomendaba a cada grupo y si el recomendador proponía una que el grupo ya hubiese visto ésta se rechazaba automáticamente. Para poder asegurar un grado de justicia (sección 3.5), necesitábamos poder saber cómo de contento estaba cada usuario con cada grupo en cada recomendación y en función de cual fuera su nivel de felicidad su opinión tendría más o menos importancia, de esta forma pretendíamos que la felicidad global fuese uniforme y que tras varias recomendaciones todos los usuarios tuvieran un nivel de felicidad aceptable.

Para poder estudiar la evolución de la felicidad consideramos la siguiente función, que muestra cuán de contento está el usuario u_i con el grupo g_j en

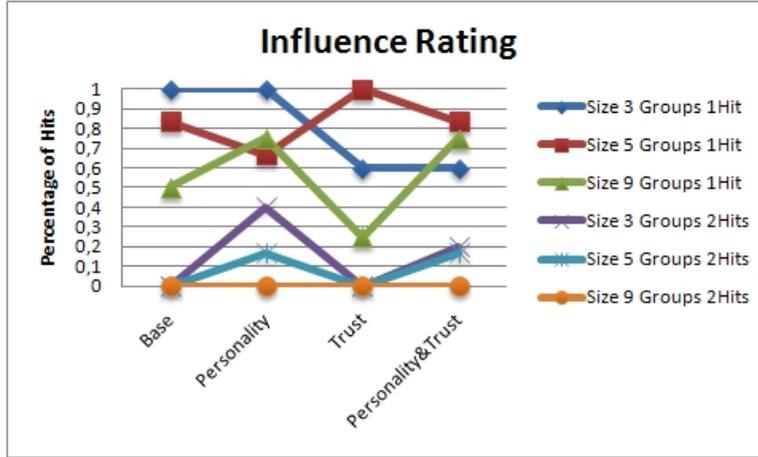


Figura 4.12: Tasa de aciertos para cada tamaño de grupo con los Ratings basados en la influencia

función de la cercanía de las tres primeras películas recomendadas al grupo gf a las películas recomendadas individualmente al usuario. Inicialmente consideramos que cada usuario tiene un 1.0 de felicidad:

$$happy(u_i, g_j) = \begin{cases} 1,0, & \text{si } gf \text{ está entre las 3 primeras;} \\ 0,9, & \text{si } gf \text{ está en la cuarta posición;} \\ 0,8, & \text{si } gf \text{ está entre las 6 primeras;} \\ 0,6, & \text{si } gf \text{ está entre las 8 primeras;} \\ 0,4, & \text{si } gf \text{ está entre las 10 primeras;} \\ 0,2, & \text{si } gf \text{ está entre las 12 primeras;} \\ 0,0, & \text{si } gf \text{ está entre las 15 primeras;} \end{cases}$$

La función de evaluación que utilizaremos para comprobar si la felicidad global para cada usuario en cada grupo mejora es la siguiente

$$gh = \alpha \cdot media(happy(u_i, g_j)) - \beta \cdot desviacionTipica(happy(u_i, g_j)) \quad (4.2)$$

Donde α y β son valores obtenidos experimentalmente, en este caso colocamos ambos valores a 1.

Como ahora no estábamos interesados en comprobar el funcionamiento de los recomendadores aplicando sólo personalidad, sólo confianza, etc. Utilizamos para este experimento el recomendador del apartado anterior, 4.3, que tiene en cuenta la personalidad y los factores sociales, pues es el que mejores resultados obtiene. También utilizamos la misma configuración y los mismos datos de los usuarios y grupos que en el apartado anterior, pues ahora no íbamos a estudiar ni la distribución de personalidad ni el tamaño, así como la misma función de evaluación *one-hit*. Sin embargo, para incluir el factor de la memoria, implementamos las ecuaciones descritas en la sec-

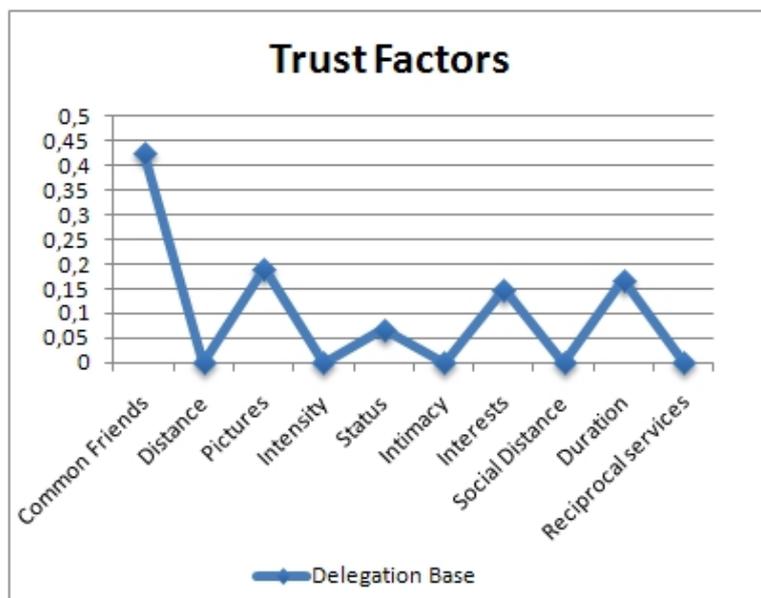


Figura 4.13: Importancia de los factores de confianza

ción 3.5 (ecuaciones 3.11 y 3.12), para los enfoques de *Ratings basados en la delegación* y *Ratings basados en la influencia*.

4.4.1. Resultados del experimento

En esta sección explicamos los resultados que obtuvimos al tener en cuenta la memoria y ejecutar los algoritmos previamente descritos sucesivas veces.

En la gráfica 4.14 vemos que para el enfoque *Ratings basados en la influencia* aunque inicialmente sube la eficiencia del método y casi siempre es mayor que el inicial, a lo largo del tiempo las soluciones que propone el recomendador son peores para el grupo como un todo, lo mismo que ocurre con todas las recomendaciones para el enfoque *Ratings basados en la delegación*. Esta situación al esperábamos y es lógica, ya que si estamos dando importancia a que todo el mundo en el grupo esté contento, no nos estamos centrando en las decisiones buenas para el grupo, las que tomarían después de una discusión sino en las que tomarían dando a elegir a la persona que ha salido más perjudicada en las recomendaciones previas.

En cuanto a la felicidad global, vemos en la gráfica 4.15 que con los dos enfoques ésta tiende a subir en cuanto se realiza más de una recomendación para el grupo. Luego demostramos que tras varias recomendaciones conseguimos que la felicidad global sea alta. Observamos que transcurridas 4 recomendaciones la felicidad tiende a bajar, esto es debido a que en los experimentos no hemos variado la cartelera y cuantas más recomendaciones hacemos menos películas interesantes quedan, tanto individualmente como

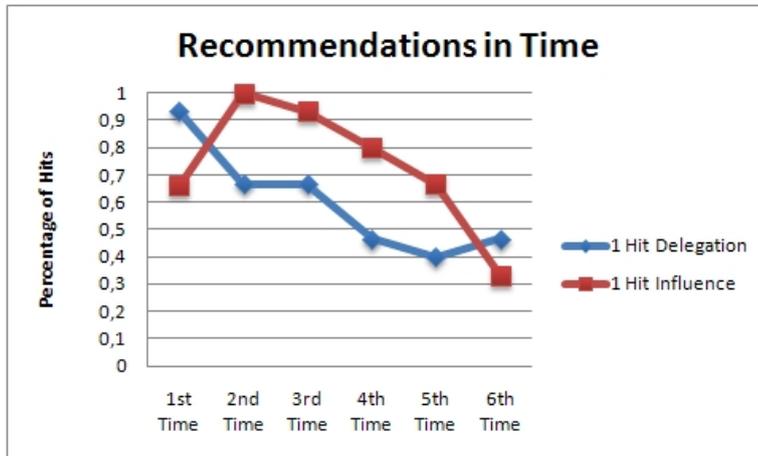


Figura 4.14: Recomendaciones a lo largo del tiempo

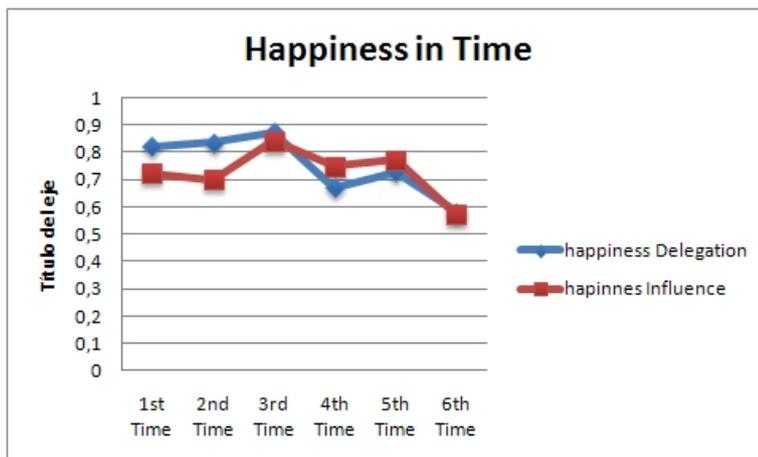


Figura 4.15: Felicidad a lo largo del tiempo

grupalmente. Esto se solucionaría cambiando la cartelera en sucesivas recomendaciones (experimento que planeamos hacer en un futuro).

También hicimos un estudio, considerando perjudicar a los usuarios que habían salido especialmente satisfechos con una recomendación, para que su opinión fuese en la siguiente recomendación menos prioritaria. Para lograr esto simplemente modificamos la fórmula $happy(u_i, g_j)$ y si gf está entre las 3 primeras la función valdrá 1.2. Los resultados de las recomendaciones salieron iguales que los anteriores y los de la felicidad a lo largo del tiempo aunque los valores eran mínimamente distintos la gráfica tiene la misma forma, por lo que no la incluimos. Luego concluimos que perjudicar a los usuarios especialmente satisfechos, no supone ningún cambio luego no es relevante.

Resumen

En este capítulo hemos explicado el desarrollo de tres experimentos realizados en el dominio de las películas. Uno basado en datos simulados, el segundo basado en datos reales y el último también basado en datos reales pero centrado en el estudio de la felicidad de los componentes del grupo a lo largo de diversas recomendaciones. En ellos hemos estudiado diversos factores como la personalidad o los factores sociales, las distribuciones que pueden tener éstas en un grupo, los diferentes tamaños que puede tener un grupo y cómo todo esto afecta en la eficiencia del recomendador. A continuación explicaremos las conclusiones obtenidas a lo largo de este trabajo y las futuras líneas de trabajo que tomaremos.

Capítulo 5

Conclusiones y Trabajo Futuro

5.1. Conclusiones

En este proyecto hemos introducido un método para realizar recomendaciones para grupos basadas en técnicas existentes de filtrado colaborativo teniendo en cuenta la composición de la personalidad del grupo y la estructura del grupo en sí. Una vez mostramos que los perfiles de personalidad pueden mejorar las recomendaciones para un grupo de personas, hemos extendido este enfoque reflejando las relaciones de los usuarios del grupo en un modo realista.

Estas relaciones se miden con factores sociales extraídos de los grupos de personas conectados a través de redes sociales. Algunos ejemplos de estos factores sociales son la distancia en la red social, el número de amigos en común, la intensidad, la intimidad, la duración de la amistad, la distancia social, el porcentaje de fotos en las que aparecen juntos, y otros.

A lo largo de este proyecto hemos estudiado y valorado el funcionamiento de las siguientes medidas de similitud: estrategia de agregación promedio (AverSas), estrategia de minimización de la penalización (MinPen), estrategia de minimización de la miseria (LeastMis), Ratings basados en la personalidad con estrategia de agregación promedio (PBR-AverSas), Ratings basados en la personalidad con estrategia de minimización de la penalización (PBR-MinPen), Ratings basados en la personalidad con estrategia de minimización de la miseria (PBR-LeastMis), Ratings basados en la delegación (DBR), Ratings basados en la influencia (IBR), Ratings basados en la delegación con memoria (DBRM), Ratings basados en la influencia con memoria (IBRM).

Para los experimentos hemos obtenido datos de usuarios reales en el dominio de las películas creando dos eventos en dos redes sociales, Facebook y Tuenti. Han participado en nuestro experimento 58 usuarios de los que obtuvimos 15 grupos. También hemos utilizado datos sintéticos para formar 76 grupos de entre 100 usuarios creados. Ha quedado demostrado que la simulación de estos datos sintéticos es correcta y por tanto sus resultados

relevantes, ya que concuerdan con los resultados de los datos reales.

La estructura del grupo –que ha sido medida a través de los factores sociales entre los miembros del grupo– se utiliza para obtener valores de confianza entre sus miembros. Cada valor de confianza se calcula como una media ponderada de los factores sociales. Estos pesos se han obtenido experimentalmente para cada estrategia de grupo por medio de un algoritmo genético. Esto nos proporciona un resultado colateral: la identificación del factor social más relevante en cuanto a recomendaciones para grupos. El factor más relevante es el número de amigos en común, seguido de las fotos, la duración, los intereses comunes y el status.

Hemos realizado todos los experimentos en el dominio de las películas utilizando datos de MovieLens y hemos estudiado la eficiencia de los distintos enfoques de recomendación para diferentes tamaños de grupos. Concluimos que aunque usando sólo la personalidad se mejora la precisión, pero usando únicamente los valores de confianza entre los componentes del grupo no se obtiene una mejora significativa. Sin embargo cuando combinamos los dos, los valores de personalidad y de confianza obtenemos la mejor actuación del recomendador. En el experimento explicado en la sección 4.3 alcanzamos –utilizando la métrica de evaluación one-hit– una mejoría de un 17.8 % respecto del recomendador base. Es más, con la medida de evaluación más dura, two-hits, proporciona resultados con un 35 % de mejora.

En cuanto a la influencia del tamaño del grupo, la conclusión es que todos los enfoques funcionan mejor con grupos más pequeños, situación bastante lógica ya que cuanto más gente hay más opiniones distintas existen, por tanto es más difícil alcanzar una solución consensuada. Y en cuanto a la distribución de las personalidades concluimos que los grupos con personalidades heterogéneas y al menos un rol de líder suponen una mejora en la actuación del recomendador, esta situación sentido ya que en los grupos sin ninguna persona autoritaria nadie impondrá su opinión y estarán delegando sin llegar a una decisión final y en los grupos en los que todos tienen el mismo tipo de personalidad o todos cederán o nadie cederá por lo que será más difícil ponerse de acuerdo.

Finalmente observamos que guardando un registro de las recomendaciones previas y considerando qué personas han salido perjudicadas en cada una de ellas y por tanto intentado favorecerlas en las siguientes recomendaciones conseguimos un nivel de satisfacción global mejor, por lo que los usuarios estarán inclinados a utilizar más veces el recomendador. Es más en el caso del enfoque *Ratings basados en la influencia* los resultados de las recomendaciones son mejores a lo largo del tiempo.

En resumen las aportaciones principales de este trabajo son:

- El desarrollo de una aplicación de recomendaciones para grupos que combina los factores sociales y la personalidad de cada miembro del grupo para mejorar los resultados de las recomendaciones, utilizando

JCOLIBRI.

- Análisis de las distintas técnicas de recomendación grupal.
- Análisis de los factores sociales que pueden utilizarse en los procesos de recomendación global: personalidad y confianza.
- Propuesta de varios métodos de recomendación grupal que incluyen los factores previamente estudiados.
- La inclusión de un sistema con memoria para evitar repeticiones en las recomendaciones y un control sobre la felicidad de los usuarios con las recomendaciones para asegurar un satisfacción homogénea.
- El desarrollo de una aplicación de recomendaciones para grupos que combina los factores sociales y la personalidad de cada miembro del grupo para mejorar los resultados de las recomendaciones pero mediante modelos distribuidos y argumentación, utilizando la plataforma *D²ISCO* y jCOLIBRI.
- La experimentación y verificación de resultados utilizando usuarios reales y redes sociales.
- Abstracción de la metodología presentada en una plantilla para jCOLIBRI.

5.2. Trabajo futuro

Los resultados que hemos presentado a lo largo de la memoria, sostienen que nuestro método propuesto es efectivo y mejoran los resultados que previamente se obtenían con otros métodos. Además, tenemos pensado realizar los siguientes estudios:

- Aunque hemos realizado los experimentos únicamente en el dominio de las películas, planeamos extender nuestros experimentos a otros dominios, como por ejemplo el dominio de los juegos o de los viajes, y con otros usuarios.
- En este trabajo hemos utilizado las funciones de agregación maximización de la satisfacción media (average satisfaction), minimizar la miseria (least misery) y minimizar la penalización (minimizing penalization). Sin embargo existen otras funciones de agregación que se pueden incluir en nuestro recomendador. Tenemos intención de evaluar el impacto del uso de las diferentes funciones de agregación en nuestro enfoque e implementar por ejemplo algunas de las citadas en Masthoff (2004).

- Combinando nuestros resultados con las ideas propuestas en Amer-Yahia et al. (2009) podríamos proponer recomendadores adaptativos, donde el algoritmo de recomendación se adapte a la distribución de la personalidad en el grupo y a su tamaño
- También queda pendiente la profundización en los Modelos distribuidos y argumentación explicados en la sección 3.6 utilizando jCOLIBRI2 y D^2ISCO así como la realización de experimentos más detallados.
- Otra línea a seguir es la realización de aplicaciones para Facebook para la recomendación de películas para grupos de usuarios. Entre las distintas ideas que tenemos para estas aplicaciones estarían la realización de una que permitiera calcular el factor de personalidad a través de distintos test de personalidad, entre ellos el TKI (Thomas y Kilmann, 1974). Otra posible aplicación sería una que permitiese a los usuarios conocer la confianza que tienen con sus amigos dentro de la red social, en esta aplicación se extraería la información relativa a los perfiles de los usuarios y se calcularía la confianza siguiendo el método de *Recomendaciones basadas en factores sociales* explicado en la sección 3.3. Finalmente se podría construir una última aplicación que utilizase los resultados de las dos anteriores y en la que se realizaran recomendaciones a grupos de usuarios utilizando *GRUPITO* explicado a lo largo de toda esta memoria.

Para finalizar, mencionamos las publicaciones a las ha dado lugar este proyecto:

- Quijano-Sánchez, L., Recio-García, J. A., Díaz-Agudo, B. *Social Based Recommendations to Groups*. 14th UK Workshop on Case-Based Reasoning. CMS Press, University of Greenwich, páginas 46-57, 2009. ISBN 978-1-904521-64-8.
- Recio-García, J. A., Díaz-Agudo, B., González-Sanz S. and Quijano-Sánchez, L. *Distributed Deliberative Recommender Systems*. Journal of Transactions on Computational Collective Intelligence, 2010.
- Quijano-Sánchez, L., Recio-García, J. A., Díaz-Agudo, B. and Jiménez-Díaz G. *How individuals influence social groups*. Enviado al ACM Recommender Systems (RecSys) 2010.
- Quijano-Sánchez, L., Recio-García, J. A., Díaz-Agudo, B. *Personality and Social Trust in Group Recommendations*. Enviado al European Conference on Artificial Intelligence (ECTAI) 2010.

Bibliografía

- AAMODT, A. y PLAZA, E. Case-based reasoning: Foundational issues, methodological variants, and system approaches. *Artificial Intelligence Communications*, vol. 7(1), páginas 39–59, 1994.
- AMER-YAHIA, S., ROY, S. B., CHAWLAT, A., DAS, G. y YU, C. Group recommendation: semantics and efficiency. *Proc. VLDB Endow.*, vol. 2(1), páginas 754–765, 2009. ISSN 2150-8097.
- ANDERSON, C. Calculating latent demand in the long tail. En *KDD*, página 1. 2007.
- ARDISSONO, L., GOY, A., PETRONE, G., SEGNAN, M. y TORASSO, P. Intrigue: Personalized recommendation of tourist attractions for desktop and handset devices. *Applied Artificial Intelligence*, vol. 17(8), páginas 687–714, 2003.
- BOBADILLA, J., SERRADILLA, F. y HERNANDO, A. Collaborative filtering adapted to recommender systems of e-learning. *Knowl.-Based Syst.*, vol. 22(4), páginas 261–265, 2009.
- BRIDGE, D., GÖKER, M. H., MCGINTY, L. y SMYTH, B. Case-based recommender systems. *Knowledge Engineering Review*, vol. 20(3), páginas 315–320, 2006.
- BRIDGE, D. G., GÖKER, M. H., MCGINTY, L. y SMYTH, B. Case-based recommender systems. *Knowledge Eng. Review*, vol. 20(3), páginas 315–320, 2005.
- CHAO, D. L., BALTHROP, J. y FORREST, S. Adaptive radio: achieving consensus using negative preferences. En *GROUP*, páginas 120–123. 2005.
- CHEN, Y.-L., CHENG, L.-C. y CHUANG, C.-N. A group recommendation system with consideration of interactions among group members. *Expert Syst. Appl.*, vol. 34(3), páginas 2082–2090, 2008. ISSN 0957-4174.
- CROSSEN, A., BUDZIK, J. y HAMMOND, K. J. Flytrap: intelligent group music recommendation. En *IUI*, páginas 184–185. 2002.

- DÍAZ-AGUDO, B., GONZÁLEZ-CALERO, P. A., RECIO-GARCÍA, J. A. y SÁNCHEZ-RUIZ-GRANADOS, A. A. Building cbr systems with jcolibri. *Sci. Comput. Program.*, vol. 69(1-3), páginas 68–75, 2007.
- FAZLOLLAHI, B., VAHIDOV, R. M. y ALIEV, R. A. Multi-agent distributed intelligent system based on fuzzy decision making. *Int. J. Intell. Syst.*, vol. 15(9), páginas 849–858, 2000.
- FOWLER, J. H. y CHRISTAKIS, N. A. Dynamic spread of happiness in a large social network: longitudinal analysis over 20 years in the framingham heart study. *Brithis Medical Journal*, 2008.
- GILBERT, E. y KARAHALIOS, K. Predicting tie strength with social media. En *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*, páginas 211–220. ACM, New York, NY, USA, 2009. ISBN 978-1-60558-246-7.
- GOLBECK, J. Combining provenance with trust in social networks for semantic web content filtering. En *Provenance and Annotation of Data, International Provenance and Annotation Workshop, IPAW 2006, Chicago, IL, USA, May 3-5, 2006, Revised Selected Papers* (editado por L. Moreau y I. T. Foster), vol. 4145 de *Lecture Notes in Computer Science*, páginas 101–108. Springer, 2006a. ISBN 3-540-46302-X.
- GOLBECK, J. Generating predictive movie recommendations from trust in social networks. En *iTrust: 4th International Conference on Trust Management*, páginas 93–104. 2006b.
- GONZÁLEZ-SANZ, S., RECIO-GARCÍA, J. A. y DÍAZ-AGUDO, B. D2ISCO: Distributed Deliberative CBR Systems with jCOLIBRI. En *1st Int. Conference on Computational Collective Intelligence.*, vol. 5796 de *LNCS*, páginas 321–332. Springer, 2009. ISBN 978-3-642-04440-3.
- JAMESON, A. More than the sum of its members: challenges for group recommender systems. En *AVI '04: Proceedings of the working conference on Advanced visual interfaces*, páginas 48–54. ACM, New York, NY, USA, 2004. ISBN 1-58113-867-9.
- JAMESON, A., BALDES, S. y KLEINBAUER, T. Two methods for enhancing mutual awareness in a group recommender system. En *AVI*, páginas 447–449. 2004.
- JAMESON, A. y SMYTH, B. Recommendation to groups. En *The Adaptive Web, Methods and Strategies of Web Personalization* (editado por P. Brusilovsky, A. Kobsa y W. Nejdl), vol. 4321 de *Lecture Notes in Computer Science*, páginas 596–627. Springer, 2007. ISBN 978-3-540-72078-2.

- KELLEHER, J. y BRIDGE, D. G. An accurate and scalable collaborative recommender. *Artif. Intell. Rev.*, vol. 21(3-4), páginas 193–213, 2004.
- KONSTAN, J. A., MILLER, B. N., MALTZ, D., HERLOCKER, J. L., GORDON, L. R. y RIEDL, J. Grouplens: Applying collaborative filtering to usenet news. *Commun. ACM*, vol. 40(3), páginas 77–87, 1997.
- LEAKE, E., D. B. Case-based reasoning: Experiences, lessons, and future directions. En *Menlo Park, CA: AAAI Press/MIT Press, Menlo Park, CA*. 1996.
- LIEBERMAN, H., DYKE, N. W. V. y VIVACQUA, A. S. Let's browse: a collaborative browsing agent. *Knowl.-Based Syst.*, vol. 12(8), páginas 427–431, 1999.
- MASTHOFF, J. Group modeling: Selecting a sequence of television items to suit a group of viewers. *User Modeling and User-Adapted Interaction*, vol. 14(1), páginas 37–85, 2004. ISSN 0924-1868.
- MASTHOFF, J. y GATT, A. In pursuit of satisfaction and the prevention of embarrassment: affective state in group recommender systems. *User Modeling and User-Adapted Interaction*, vol. 16(3-4), páginas 281–319, 2006. ISSN 0924-1868.
- MCCARTHY, J. F. Pocket restaurant finder: A situated recommender systems for groups. En *Proceeding of Workshop on Mobile Ad-Hoc Communication at the 2002 ACM Conference on Human Factors in Computer Systems*. 2002.
- MCCARTHY, J. F. y ANAGNOST, T. D. MusicFX: An arbiter of group preferences for computer supported collaborative workouts. En *CSCW '98: Proceedings of the 1998 ACM conference on Computer supported cooperative work*, páginas 363–372. ACM, 1998.
- MCCARTHY, K., SALAMÓ, M., COYLE, L., MCGINTY, L., SMYTH, B. y NIXON, P. Cats: A synchronous approach to collaborative group recommendation. En *FLAIRS Conference*, páginas 86–91. 2006.
- MCDONALD, D. W. Recommending collaboration with social networks: a comparative evaluation. En *CHI '03: Proceedings of the SIGCHI conference on Human factors in computing systems*, páginas 593–600. ACM, New York, NY, USA, 2003. ISBN 1-58113-630-7.
- MCGINTY, L. y SMYTH, B. On the role of diversity in conversational recommender systems. En *ICCB*, páginas 276–290. 2003.
- MILLER, G. A. y FELLBAUM, C. Wordnet then and now. *Language Resources and Evaluation*, vol. 41(2), páginas 209–214, 2007.

- O'CONNOR, M., COSLEY, D., KONSTAN, J. A. y RIEDL, J. PolyLens: a recommender system for groups of users. En *ECSCW'01: Proceedings of the seventh conference on European Conference on Computer Supported Cooperative Work*, páginas 199–218. Kluwer Academic Publishers, Norwell, MA, USA, 2001. ISBN 0-7923-7162-3.
- O'DONOVAN, J. y SMYTH, B. Trust in recommender systems. En *IUI '05: Proceedings of the 10th international conference on Intelligent user interfaces*, páginas 167–174. ACM, New York, NY, USA, 2005. ISBN 1-58113-894-6.
- PAZZANI, M. J. y BILLISUS, D. Content-based recommendation systems. En *The Adaptive Web*, páginas 325–341. 2007.
- PLAZA, E. y ONTAÑÓN, S. Learning collaboration strategies for committees of learning agents. *Autonomous Agents and Multi-Agent Systems*, vol. 13(3), páginas 429–461, 2006.
- RECIO-GARCÍA, J. A., DÍAZ-AGUDO, B. y GONZÁLEZ-CALERO, P. A. Prototyping Recommender Systems in jCOLIBRI. En *Proceedings of the 2008 ACM conference on Recommender Systems*, páginas 243–250. ACM, NY, USA, 2008. ISBN 978-1-60558-093-7.
- RECIO-GARCÍA, J. A., JIMENEZ-DIAZ, G., SÁNCHEZ-RUIZ, A. A. y DÍAZ-AGUDO, B. Personality aware recommendations to groups. En *Procs. of the 2009 ACM Conference on Recommender Systems*, páginas 325–328. ACM, 2009.
- ROGERS, E. *Diffusion of Innovations*. N.Y., Free Press, 1983.
- SCHAFER, J., FRANKOWSKI, D., HERLOCKER, J. y SEN, S. *Collaborative Filtering Recommender Systems*, vol. 4321, capítulo 9, páginas 291–324. Heidelberg S.B, 2007.
- SHIMAZU, H. Expertclerk: Navigating shoppers buying process with the combination of asking and proposing. En *IJCAI*, páginas 1443–1450. 2001.
- SHIMAZU, H. ExpertClerk: A conversational case-based reasoning tool for developing salesclerk agents in e-commerce webshops. *Artificial Intelligence Review*, vol. 18(3–4), páginas 223–244, 2002.
- SINHA, R. R. y SWEARINGEN, K. Comparing recommendations made by online systems and friends. En *DELOS Workshop: Personalisation and Recommender Systems in Digital Libraries*. 2001.
- THOMAS, K. y KILMANN, R. *Thomas-Kilmann Conflict Mode Instrument*. Tuxedo, N.Y., 1974.

-
- VICTOR, P., CORNELIS, C., COCK, M. D. y TEREDESAI, A. Key figure impact in trust-enhanced recommender systems. *AI Commun.*, vol. 21(2-3), páginas 127–143, 2008.
- YAHJA, A. y CARLEY, K. M. Wizer: An automated intelligent tool for model improvement of multi-agent social-network systems. En *FLAIRS Conference*, páginas 44–50. 2005.
- YU, Z., ZHOU, X., HAO, Y. y GU, J. Tv program recommendation for multiple viewers based on user profile merging. *User Modeling and User-Adapted Interaction*, vol. 16(1), páginas 63–82, 2006. ISSN 0924-1868.

El/la abajo firmante, matriculado/a en el Máster en Investigación en Informática de la Facultad de Informática, autoriza a la Universidad Complutense de Madrid (UCM) a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a su autor el presente Trabajo Fin de Máster: 'Impacto de los factores y organizaciones sociales en los procesos de recomendación para grupos', realizado durante el curso académico 2009-2010 bajo la dirección de Belén Díaz Agudo [y con la colaboración externa de dirección de Juan A. Recio García] en el Departamento de Ingeniería del Software e Inteligencia Artificial, y a la Biblioteca de la UCM a depositarlo en el Archivo Institucional E-Prints Complutense con el objeto de incrementar la difusión, uso e impacto del trabajo en Internet y garantizar su preservación y acceso a largo plazo.

