



UNIVERSITAT DE  
BARCELONA

Trabajo fin del grado

GRADO DE INFORMÁTICA

Facultad de Matemáticas  
Universitat de Barcelona

---

Estudio y análisis de técnicas de  
consenso para recomendadores  
grupales

---

Autor: Yue Lin

Tutora: Dra. Maria Salamó

Realizado a: Departamento de Matemáticas  
e Informática

Barcelona, 30 de junio de 2016

# Índice

<b>1. Introducción</b>	<b>6</b>
1.1. Descripción del problema y marco del proyecto . . . . .	6
1.2. Motivación del proyecto . . . . .	7
1.3. Objetivos del proyecto . . . . .	8
1.4. Descripción de los capítulos . . . . .	8
<b>2. Estado de arte/Recomendadores</b>	<b>9</b>
2.1. Introducción . . . . .	9
2.2. Sistema de recomendación basado en críticas . . . . .	10
2.3. Recomendadores grupales [1] . . . . .	11
2.4. Sonsacamiento y negociación de las preferencias [4] . . . . .	13
2.5. Técnicas de sonsacamiento de preferencias [2] . . . . .	16
2.6. Agregación de preferencias en sistemas de recomendación grupal [3] . . . . .	17
2.7. Técnicas de consenso en recomendadores grupales [6] . . . . .	19
<b>3. Análisis, diseño e implementación</b>	<b>24</b>
3.1. Introducción . . . . .	24
3.2. Diagrama de paquetes . . . . .	25
3.3. Diagrama de la clase . . . . .	27
3.4. Diseño e implementación . . . . .	31
<b>4. Análisis de resultados</b>	<b>37</b>
4.1. Metodología . . . . .	37
4.2. Análisis y conclusiones de los Resultados . . . . .	39
<b>5. Análisis de coste</b>	<b>56</b>
<b>6. Conclusiones y líneas futuras</b>	<b>57</b>

# Abstarct

## Resumen

Cada día son más personas que buscan en internet opiniones de otros usuarios antes de comprar un producto. Los sistemas recomendadores surgieron ante esta necesidad. Un sistema recomendador muestra una lista de productos según las preferencias indicadas por cada usuario. Hay muchos tipos de recomendadores: basados en contenido, basados en filtrado colaborativo, etc. La mayoría de recomendadores se centran en realizar las recomendaciones a un único usuario en cada interacción usuario-sistema.

En este trabajo final de grado se estudiarán los recomendadores grupales y se centrará el trabajo en el estudio y análisis de estrategias de consenso. Se van a emplear tres tipos de estrategias diferentes para alcanzar el consenso de los usuarios. Para testear estas estrategias se crearán grupos de diferentes tamaños y se dividirán estos grupos en subgrupos según las similitudes entre los miembros de cada grupo. Finalmente, se analizarán los niveles de satisfacción que se alcanzan con las diferentes estrategias en cada uno de estos grupos para concluir qué método es mejor.

## **Resum**

Cada dia són més persones que busquen en internet opinions d'altres usuaris abans de comprar un producte. El sistema recomanador van sorgir degut a aquesta necessitat. Un sistema recomanador mostra una llista de productes segons les preferències indicades per cada usuari. Hi ha molts tipus de recomanadors: basats en contingut, basats en filtrat col·laboratiu, etc. La majoria de recomanadors se centren a realitzar les recomanacions a un únic usuari en cada interacció usuari-sistema.

En aquest treball final de grau s'estudien els recomenadores grupales i se centrarà el treball en l'estudi i anàlisi d'estratègies de consens. Es van a emprar tres tipus d'estratègies diferents per aconseguir el consens dels usuaris. Per testejar aquestes estratègies es crearan grups de diferents grandàries i es dividiran aquests grups en subgrups segons les similituds entre els membres de cada grup. Finalment, s'analitzaran els nivells de satisfacció que s'aconsegueixen amb les diferents estratègies en cadascun d'aquests grups per concloure quin mètode és millor.

## **Summary**

Every day a lot of people search on internet reviews of others about a product before they buy it. This is due to the fact that the recommender system was created. A recommender system displays a list of products according to the user preferences. There are many types of recommenders: content based, collaborative filtering, ... Most of them focus on making recommendations to a single user.

In this degree final project, we focus to study group recommenders and consensus strategies. This project employs three types of strategies to achieve users consensus. We're going to apply these strategies to groups of different sizes and similarities. Finally, we'll analyze users satisfactions with different strategies in each of these groups to conclude what methods are better.

# Agradecimiento

Cuatro años de esfuerzos para llegar hasta aquí, todavía me acuerdo cómo era cuando acababa los estudios de bachillerato, y enfrentarme por la primera vez a las pruebas de selectividad. Los nervios que se acumulaban en aquellos días, y la alegría que me saltó cuando me enteré que me aceptó la petición de estudiar en esta universidad.

Era el comienzo de un viaje nuevo, de venir a Barcelona para emprender mi nueva etapa. Aquí conocí gente nueva con ideas claras, y a mis amigos. Era un largo camino con mucha dificultad al principio. Y los he superado junto a mis familias y amigos.

Primero quiero agradecer a mis familias por apoyarme de forma incondicional durante estos 4 años. A mis padres, por innumerables días que me esperan a casa. Por animarme en estados deprimidos. A mis hermanas, por cuidarme en todos los pequeños detalles, sobre todo en el periodo de los exámenes. A mi tía y a mi primo, por estar siempre interesado en mis situaciones académicas y quienes puedo contar todos los problemas que he tenido y que no pude contar a los padres

En segundo lugar, quiero agradecer a los amigos y profesores, quienes han dedicado parte de sus tiempos en mí. En trabajar conmigo, en enseñarme cosas nuevas. A Edgar Llorente, porque es mi primer amigo aquí. Quien ha sido muy paciente conmigo, sobre todo me ha ayudado muchísimo en el primer año cuando más dificultades tengo. A Christian, Guillem y Dani quienes han trabajado muchísimo conmigo, siempre hemos estado junto aprendiendo. Y todavía me acuerdo de todas aquellas noches que pasábamos juntos para acabar nuestro trabajo.

Quiero agradecer a todos los profesores quienes me han enseñado, y sobre todo tengo un especial agradecimiento a mi tutora, María Salamó, quien siempre ha sido mi guía en la realización del proyecto, por todo el tiempo que se ha dedicado en corregir mi trabajo y darme consejos de mejora.

¡Muchas Gracias a todos!

# 1. Introducción

Con la llegada de los ordenadores personales y internet a los hogares en los años 90 llegó el florecimiento de las páginas webs. Cada vez se almacenan más informaciones en este sitio virtual, las informaciones que se acumulan son inmensas que nos resulta difícil navegar por ellas para encontrar las informaciones deseadas. Ante esta necesidad surgieron diferentes herramientas de filtración de informaciones dedicado a este fin. La más conocida de ella es el buscador, y hoy en día es muy conocido. Google es el gigante que mejor domina en esta sección. Pero existe otras técnicas de filtración de informaciones que las utilizamos todos los días, y la mayoría pasamos desapercibidos de ellos, una de ellos es el sistema de recomendación. Las grandes aerolíneas, tiendas online utilizan los sistemas de recomendación para recomendar a sus usuarios. Tan sólo hay que marcar las características que se desean en el producto final, el sistema busca los productos con estas características y las muestran a los usuarios.

Un sistema de recomendación emplea técnicas de filtración de informaciones para presentar informaciones del interés del usuario según sus preferencias. Y actualmente se emplean en casi todos los dominios (cine, película, restaurante, ...). Su finalidad es la recomendación de ítems(productos) que satisfacen las necesidades de los usuarios.

Hoy en día existen muchos tipos de recomendadores según sus características (forma de recomendación, número de productos que se recomienda, ...). Pero la gran mayoría de los trabajos se centran sus objetivos en recomendar a un sólo usuario. Mientras que hay muchos menos trabajos dedicados en recomendaciones a grupo de usuarios. Las dificultades que se encuentra en los recomendadores grupales son más difíciles de resolver. Porque implican varios usuarios a la vez. Pero sea cual sea el tipo de recomendador el objetivo de todos ellos se puede resumir en la siguiente pregunta: ¿Cómo resolver de forma satisfactoria la toma de decisiones para un grupo de usuarios?

En la mayoría de veces no existe una solución satisfactoria para todos, sino que optamos por una solución que satisface a la mayoría. Y es aquí donde se centra el trabajo, estudiamos las diferentes estrategias de consenso

## 1.1. Descripción del problema y marco del proyecto

Los recomendadores clásicos han mejorado mucho en cuanto a su calidad en la recomendación desde las últimas décadas. Expandiendo su dominio en muchos ámbitos. Un buen ejemplo de ellos es la tienda online de Amazon<sup>1</sup> donde nos recomienda nuevos productos a partir de los que hemos vistos. Aunque sigue sin resolverse los típicos problemas de los recomendadores. Uno de ellos es el fenómeno de 'cold

---

<sup>1</sup><https://www.amazon.es>

start', es decir, que de un usuario se desconocen sus preferencias, y por lo tanto no se sabe qué recomendar. Son informaciones imprescindibles para que el sistema de recomendación pueda interpretar el usuario y hacerle una buena recomendación.

Los recomendadores clásicos destinan su recomendación a un único destinatario en cada interacción usuario sistema. Sin embargo, existe otro tipo de sistema de recomendación en cuanto al número de usuario objetivo se refiere, y este es el sistema de recomendación grupal.

Pero el problema no termina aquí, el proceso de recomendación en un recomendador grupal es aún más complejo. Existe varias estrategias de recomendación; desde la construcción de un modelo grupal de las preferencias[5] a crear una lista colectiva de los ítems más preferidos por cada uno de los usuarios del grupo. El proceso no termina aquí, una vez hecha las recomendaciones para el grupo, los miembros han de decidir con cuál de ellos quedar. En muchos dominios, se ha de escoger un único producto como resultado final.

Y los recomendadores existentes emplean diferentes estrategias para solventar este problema. En este trabajo explicaremos muchos de ellos.

El sistema de recomendación grupal tal como indica su nombre destina sus recomendaciones a varios usuarios en una petición de recomendación. Y debido a esta característica suya, el proceso de recomendación es más complejo que un recomendador tradicional. Primero porque hemos de seleccionar posibles candidatos de productos que pueda satisfacer al grupo. Y una vez recomendados estos productos, son los propios usuarios quienes opta por una de las opciones. He aquí otro de los problemas típicos de este tipo de recomendadores, puesto que en la vida real es difícil satisfacer a un conjunto de usuarios con un único producto, puesto que los miembros del grupo pueden tener preferencias muy parecidas, pero nunca idénticas si el espacio de características de los productos es suficientemente amplio.

La clave de una buena recomendación en este tipo de recomendador grupal consiste en como seleccionar buenos ítems, y recomendarlos al grupo para que ellos decidan con cuál de ellos quedarse. Existen muchas estrategias para seleccionar buenos ítems. En concreto, este proyecto se enmarca en el estudio de las estrategias de consenso en recomendadores grupales.

## **1.2. Motivación del proyecto**

La motivación con el que me ha llevado a realizar dicho proyecto, es la curiosidad de resolución de conflictos en grupos de usuarios cuando hay variedad de opiniones. En estas situaciones, resulta imprescindible manejar adecuadamente las opiniones de cada uno, estudiar sus influencias en el grupo, y su permisividad, etc. Estos factores resultan importantes para el alcance de un acuerdo.



De las opiniones surgen las ideas y de las buenas ideas surgen los avances. Pero un incorrecto manejo de dichas opiniones causará la inestabilidad y la división de los grupos.

### **1.3. Objetivos del proyecto**

El objetivo principal de nuestro trabajo es el estudio de las técnicas de consenso en recomendadores grupales. Además, dentro del marco del proyecto también tenemos una serie de objetivos específicos:

1. Estudio del estado de arte: en el cual estudiaremos la situación actual de los sistemas de recomendación grupal.
2. Implementación de las técnicas de consenso más destacables dentro de la literatura. Dichas estrategias las explicaremos con más detalle en la siguiente sección.
3. Evaluación de las técnicas con diferentes configuraciones de los grupos. Concretamente aplicaremos 4 configuraciones: Nash, Joint, promedio de todos los grupos, promedio del propio grupo.

### **1.4. Descripción de los capítulos**

En este primer capítulo hemos introducido el problema y la situación en la que estamos situados. Ahora en el segundo capítulo hablaré sobre los diferentes tipos de recomendadores grupales que existen y citaremos los sistemas de recomendación más populares, y en el tercer y cuarto capítulo entraremos en detalle sobre el nuestro recomendador implementado, y la estructura del nuestro recomendador. En el quinto capítulo evaluaremos los resultados obtenidos. Para el sexto capítulo analizaremos la proporcionalidad de trabajado realizado en cada uno de estos capítulos. Y concluimos sobre las líneas futuras del desarrollo del nuestro recomendador.

## 2. Estado de arte/Recomendadores

Los sistemas de recomendación son un campo relativamente reciente, pero la mayoría de los trabajos realizados en este campo se centran en sistemas de recomendación que recomiendan a un sólo usuario. Hoy en día con los nuevos avances tecnológicos, las tendencias cambian. No solamente se interesa en recomendar bien a un sólo usuario. Sino que en muchos casos nos encontramos con personas que buscan ofertas o productos para un grupo en vez de un sólo individuo.

El objetivo de este capítulo es describir la situación actual que se encuentran los sistemas de recomendación, y en este proyecto en concreto hablaremos sobre sistemas de recomendación grupal. Además, hablaremos sobre los diferentes tipos de sistemas recomendadores grupales que existen, y citaremos algunos trabajos realizados. También describiremos el sistema de recomendación empleado en el nuestro proyecto.

### 2.1. Introducción

Los sistemas de recomendación son técnicas de filtración de informaciones con el fin de presentar a los usuarios un conjunto de ítems que le interesan. Este tipo de sistemas surgieron hace muy poco. El despliegue de estos sistemas en la sociedad tuvo lugar en la década de los 90s, momento en el cual comienza la llegada masiva de los computadores a los hogares que anteriormente sólo unos pocos afortunados tienen acceso a ellos.

Los primeros sistemas de recomendación son primitivos, las preferencias son cardinales, es decir, se representa con un número. A medida que va mejorando las prestaciones de los computadores, se ha empezado a implementar sistemas de recomendación más complejos. Y la representación de las características ha pasado de ser solamente cardinales a alfanuméricas.

Si se clasifican los sistemas de recomendación según las características, nos daremos cuenta de que la gran mayoría de ellos realizan recomendaciones a un solo usuario. Aunque en los últimos años ha habido muchos trabajos en sistemas de recomendación grupal, los cuales destinan sus recomendaciones a varios usuarios a la vez; todavía hay muchos problemas sin resolverse. Uno de los típicos en todos los sistemas de recomendación es el problema de 'cold start'. Otro de los aspectos más importante en un sistema de recomendación es la modelación de los modelos de preferencias de los usuarios. Para poder recomendar de forma correcta a un usuario, es necesario que el sistema conozca sus preferencias.

En este capítulo, veremos las formas de representación de preferencias de usuarios. Una de las técnicas más recientes es mediante críticas. El sistema empleado en este trabajo está basado en críticas. Existe dos tipos de críticas: las unitarias; y las

compuestas.

En la siguiente sección introduciremos los sistemas de recomendación basados en críticas.

## 2.2. Sistema de recomendación basado en críticas

En este subcapítulo centraremos en dos de los aspectos de los sistemas de recomendación basado en críticas[5]: representación de las críticas y recuperación de las recomendaciones. En el primero destacamos el enfoque alterno para la adquisición de preferencias. En el segundo, destacamos los principales problemas que presenta a la hora de recuperar una recomendación descartada y presentarlo al usuario.

Las críticas son representaciones textuales de las preferencias de los usuarios. Pueden ser de dos tipos: unitarias o compuestas. Las críticas unitarias solamente permiten a los usuarios expresar una preferencia en cada ciclo de interacción. Y esto puede resultar en un proceso de recomendación prolongado. Una alternativa a este problema es la crítica compuesta que permite a un usuario expresar varias preferencias en una misma iteración.

Se puede dar caso de 'dead-end' en un proceso de recomendación basado en críticas estáticos, porque no proporcionan a usuario información sobre las ventajas-desventajas de las críticas.

Como solución se propuso un enfoque más dinámico, donde en cada ciclo de recomendación se formulan las críticas compuestas con el fin de:

1. Enfocar mejor en el recomendador.
2. Eliminar redundancia de la interfaz
3. Eliminar la aprehensión del usuario.

Generar críticas compuestas de forma automática permite acelerar el proceso de recomendación, pero también presenta el inconveniente de la contradicción entre ellas, lo cual penaliza el resultado de la recomendación.

Lo mejor es centrarse en clasificar las críticas compuestas de manera que presenten al usuario solamente aquellas más relevantes. Y este método mejora considerablemente el resultado de la recomendación en comparación con el caso de presentar muchas críticas compuestas a los usuarios. La limitación de este enfoque es que no tienen en cuenta las preferencias de los usuarios.

El éxito de un proceso de recomendación está fuertemente influenciado por las críticas que define el usuario. Es posible que un usuario sobre especifique sus críticas sobre ciertas características que deben tener los ítems a recomendar. Y como

resultado el sistema recomienda una serie de ítems que nos satisfacen las otras preferencias del usuario.

### 2.3. Recomendadores grupales [1]

Un sistema de recomendación es una técnica de filtración de informaciones que tiene como finalidad presentar productos o ítems del interés del usuario. Se surgieron ante la creciente cantidad de informaciones que se almacenan en internet, y las desorganizaciones que existen, las cuales dificultan la búsqueda de las informaciones en periodo de tiempo corto. La cantidad de informaciones que existe hace imposible navegar por la red y encontrar todas las informaciones del interés manualmente.

Como hemos indicado al principio, los sistemas de recomendación se pueden clasificar en dos tipos en cuanto al número de usuarios se refiere. Los sistemas de recomendaciones individuales recomiendan productor o ítems a un único usuario en cada interacción usuario-sistema, mientras que los sistemas de recomendación grupal recomiendan a un grupo de usuario en cada interacción con el sistema. Abajo tenemos el diagrama de clasificación de los sistemas de recomendación según este criterio:



Figura 1: Sistemas de recomendación según el número de usuarios que se destinan la recomendación final

Para poder recomendar a los usuarios, es imprescindible proporcionar al sistema las informaciones de los usuarios (Preferencias, situación geográfica, ...). Dichas informaciones se pueden obtener de forma implícita (el sistema presenta al usuario una serie de productos y este ha de indicar si le gusta o no, a partir de estas indicaciones el sistema adivina las preferencias del usuario) o explícita (el usuario indica sus preferencias directamente al sistema).

Aunque los sistemas de recomendación grupal no han tuvieron los mismos avances como los individuales, se puede clasificar los sistemas de recomendación grupal en muchos tipos: según el número de destinatario final, según la modelación de modelos de características para hacer la recomendación, etc. He aquí abajo el diagrama de clasificación de los sistemas de recomendación grupal (Véase a la figura 2):

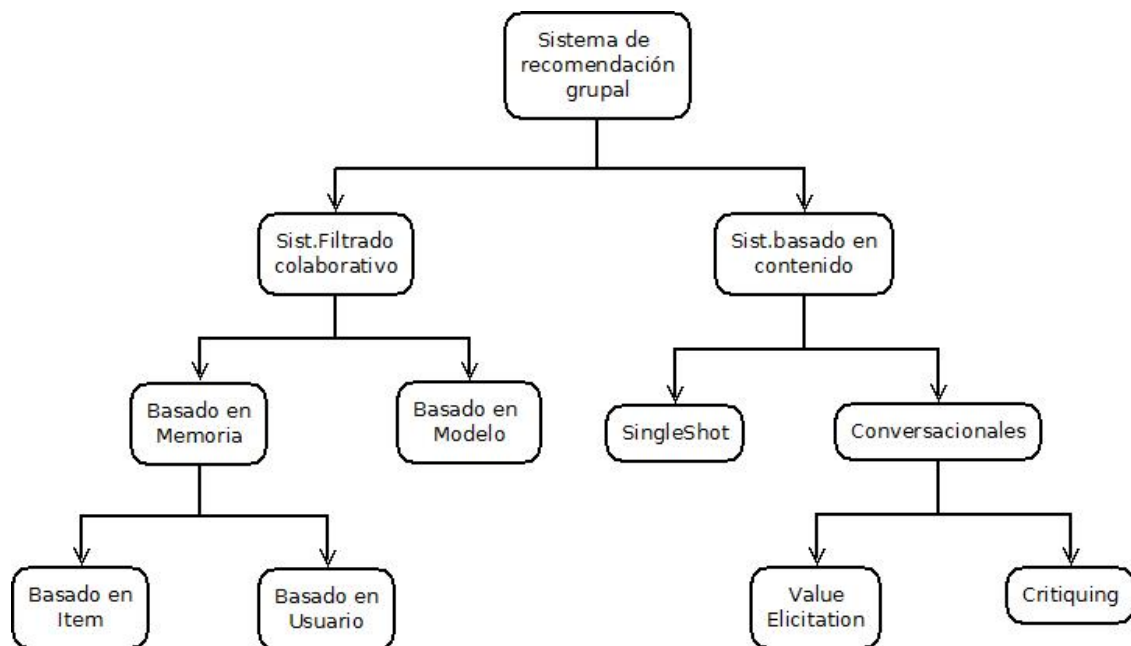


Figura 2: Diagrama de clasificación de los sistemas de recomendación grupal

Aquí describiremos cada uno de ellos:

Para empezar, hablaremos sobre sistemas de recomendación basados en contenidos. Los sistemas basados en contenidos recomiendan a los usuarios basando en productos que estos han valorado.

Cómo se podía observar los sistemas de recomendación basados en contenidos se puede dividir en: singleshot y conversacionales. Las diferencias entre ambos están en el número de recomendaciones que se puede hacer. Un sistema de recomendación singleshot solamente recomienda una vez al usuario, cada interacción de recomendación es independiente y no se tiene en cuenta las recomendaciones hechas en la última interacción. Mientras que, en un sistema de recomendación conversacional, el usuario puede añadir preferencias al modelo de preferencias, y el sistema realiza nuevas recomendaciones teniendo en cuenta las anteriores recomendaciones hechas.

Existe dos tipos de sistemas de recomendación conversacional: el primero de ellos son los sistemas basado en ‘value elicitation’, los cuales representa las preferencias con valores numéricas, esta forma de representación facilita los cálculos computacionales para recomendar, pero debido a su simplicidad, a veces requieres muchas preferencias de este tipo para definir al usuario, y al aumentar el número de preferencias, es más posible que encontramos con preferencias contradictorios. De lo contrario, los sistemas basados en críticas tienen mayor dificultad de interpretar las preferencias de los usuarios, pero permite una mejor definición.

Los sistemas de recomendación basado en filtrado colaborativo surgieron después de los basados en contenidos. Estos sistemas utilizan los perfiles de usuarios para encontrar usuarios con preferencias similares al que queremos recomendar. Recomiendan productos bien valorados por los usuarios similares.

Los sistemas de recomendación basados en filtrado colaborativo se pueden dividir en 2 tipos: los basados en memoria y los basados en modelo. El primero de ellos, buscan usuarios con valoraciones similares a los productos que ambos han valorado. Y una vez encontrado estos usuarios se genera los N ítems mejores valorados por estos usuarios similares. Cuando más productos han valorado ambos de forma similar mayor es la similitud entre ambos. Entre los inconvenientes de este sistema se encuentra la disponibilidad de usuarios similares en el sistema. El segundo de ellos, los sistemas basados en modelo tratan de predecir las valoraciones que otorgarían los usuarios a los productos, y recomendar a ellos aquellos productos con mejores valoraciones.

## 2.4. Sonsacamiento y negociación de las preferencias [4]

Las preferencias del usuario son imprescindibles para poder recomendarle un producto. Cuanto más completo está mejor. Pero cuando se recomienda a un grupo de usuario se puede encontrar con el problema de contradicción entre las preferencias de los usuarios del grupo. Dejar al sistema a resolver esta contradicción suele penalizar las satisfacciones de los usuarios con los productos recomendados. La solución ideal sería exponer este problema a los usuarios para que entre ellos se negocian y resolver el problema.

En este subcapítulo nos centraremos en este problema, aquí explicaremos problema mediante el trabajo realizado en el artículo [4]<sup>2</sup>. Los pasos del proceso de recomendación del sistema de recomendación de este artículo son los siguientes:

1. El usuario selecciona el subconjunto de características de su interés desde el espacio de características de los ítems. Estas características pueden ser positivas o negativas. Se definen como características positivas aquellas que han de tener los ítems recomendados y las características negativas, aquellas que no han de tener los ítems recomendados. Esta doble definición de características puede conllevar controversia a la hora de realizar la recomendación. Es muy probable que una característica deseada de un usuario sea no deseada para otro.
2. Una vez seleccionada las características, estas son visibles a los demás usuarios del grupo. Las características pueden ser sustituidas y serán ordenadas según su relevancia para el usuario.
3. Las preferencias individuales son añadidas constantemente al modelo colectivo, y las recomendaciones que mejor satisface a las preferencias del modelo colectivo son recomendadas.

---

<sup>2</sup>'Preference Elicitation and Negotiation in a Group Recommender System'

4. Además entre usuarios se pueden negociar y discutir sobre sus preferencias.[7] He aquí el punto clave de este proceso iterativo, porque en esta etapa los usuarios mediante conversaciones online pueden discutir y resolver los conflictos de sus preferencias.
5. Los usuarios pueden seleccionar los ítems que les hayan gustado entre los recomendados, y recomendarlos a otros participantes quienes pueden aceptarlos o proponer otras alternativas.

Como se indicó anteriormente, entre usuarios se pueden producir contradicciones en sus preferencias. Y para controlar este problema, en este sistema las operaciones (agregación, eliminación, modificación) con las preferencias conlleva un coste, y cada usuario dispone de un coste limitado para definir sus preferencias, de tal forma que ellos expresen sus preferencias más relevantes.

El sistema prototipo implementado en el artículo [4]<sup>3</sup> emplea técnicas basadas en contenido. Y utiliza una base de datos de 151.000 entradas de hotel con informaciones descriptivas.

La interfície del prototipo consta de las siguientes secciones:

1. Exploración de características: Filtros que permiten a los usuarios realizar búsquedas de características especificadas. Y también es donde guardan las características seleccionadas por los usuarios.
2. Preferencias individuales: Los usuarios pueden clasificar sus preferencias según la importancia que les da. Cada vez que un usuario selecciona una característica del área de exploración, dicha característica es añadida a la lista de preferencias del usuario. Los usuarios pueden especificar dos tipos de características:
  - a) Características positivas: son características que al usuario le interesa tenerlas en el ítem a recomendar.
  - b) Características negativas: son características indeseadas por el usuario. El ítem que se recomienda no debe tener estas características.

En este sistema se asigna a cada característica un coste numérico para facilitar la agregación de característica al modelo de grupos, porque entre las preferencias de los usuarios pueden haber conflictos como por ejemplo que una característica muy deseada por un usuario es vetada por otro.

3. Preferencias grupales: Esta sección muestra las preferencias grupales. Las preferencias individuales se agregan utilizando una variante del método Borda

---

<sup>3</sup>'Preference Elicitation and Negotiation in a Group Recommender System'

Count. Para calcular la puntuación de un nuevo elemento que se añade al modelo individual se utiliza la siguiente ecuación 2.1:

$$PAtt_i = \sum_{j=0 \dots u} \left( \frac{1}{n} (n - p_{ij}) \right) + \frac{\lambda c_i}{n} \quad (2.1)$$

donde  $p_{ij}$  es la puntuación que recibe una preferencia 'j' por parte de un usuario 'i', 'n' es el número total de preferencias en el grupo;  $c_i$  es el coste de la preferencia y  $\lambda$  es un factor de peso que nos permite corregir el coste de las preferencias dentro del modelo grupal. La finalidad es estimar la importancia de las preferencias, y después escoger aquellas más importantes.

La lista de preferencias grupales será calculada ordenando todas las preferencias individuales por el total de puntuación recibida. Las preferencias que son negativas para algunos miembros serán eliminadas del modelo grupal.

4. Chat del grupo: En esta sección, los usuarios debaten sobre las cuestiones que surgen a la hora de tomar la decisión. Permite la comunicación entre los miembros del grupo para el proceso de negociación.
5. ítems recomendados: Los ítems que mejor armonizan con las preferencias grupales son mostrados en esta sección. Para recomendar ítems el sistema utiliza un filtro basado en contenido. Los ítems son descritos por sus características, y el sistema no considera como candidatos de recomendación a todos aquellos ítems que contiene alguna característica vetada por algún usuario.

El sistema calcula la puntuación total correspondiente a cada uno de los ítems candidatos (mediante la variante de Borda Count). Aquellos ítems que tienen la puntuación total por encima de un valor umbral serán recomendados al grupo.

6. Recomendaciones seleccionadas por los usuarios: Aquí se guardan los ítems seleccionados por los miembros del grupo.

Para llegar al consenso final, entre miembros del grupo puede surgir discrepancia sobre una preferencia determinada manifestada por otro miembro y el sistema implementado permite a los miembros chatear de forma online y debatir para llegar a un acuerdo final. También permite a los usuarios realizar peticiones de cambio de preferencias al resto de grupos o proponer recomendaciones. Si la petición fue aprobada por todos, dicha preferencia será añadida al modelo grupal. Para decidir que ítem escoger como la recomendación final se calcula la satisfacción de los miembros con cada uno de los ítems recomendados. La mejor solución es escoger el ítem que tiene el mejor balance de puntuaciones como recomendación final.



Es posible que durante el proceso de decisión del ítem final el grupo apruebe una nueva característica y el sistema propone una nueva lista de recomendaciones. En este caso, se ha de volver a negociar sobre que ítem tomar.

## 2.5. Técnicas de sonsacamiento de preferencias [2]

Para realizar recomendaciones al grupo de usuarios es imprescindible construir el modelo de preferencias grupales a partir de las preferencias de los miembros. Pero si se centra solamente en las preferencias indicadas por usuarios puede ser escasa la información, se necesita emplear un enfoque híbrido en cuanto a obtención fuente de información donde combina diferentes técnicas para obtener información (demográfico, filtrado colaborativo, filtrado de semejante general, etc). Y la interacción entre usuario y sistema es pasivo.

Otro de los enfoques interesantes es poder configurar los mecanismos de agregación como módulos del sistema. He aquí una propuesta en el artículo [2]<sup>4</sup> El sistema escrito en este artículo está basado en el uso de la ontología del dominio para describir las preferencias de los usuarios y los ítems a recomendar. Este sistema está basado en la representación semántica de los ítems y las preferencias de los usuarios, y por lo tanto presenta el siguiente inconveniente: para poder describir los ítems y las preferencias de los usuarios, se han de definir primero la representación semántica del dominio en el que se abarca el sistema.

Las entidades del dominio están organizadas en estructuras jerárquicas, que están conectadas mediante la relación *is-a*. Cuando más profundo es el nivel jerárquico más específica es la clasificación. Las clases de la ontología representan las características (F) comunes en el dominio correspondiente. El nodo hoja representa el ítem a recomendar.

Cada objeto está unido a las características del dominio mediante una arista, que tiene un valor asociado que indica el **grado del interés** de dicha característica del objeto. Por lo que un ítem es descrito por un conjunto de tuplas que representa todas las aristas de entrada del ítem. Las tuplas tienen la siguiente forma:

$$(f, d^{if})$$

donde  $f \in F$  es una característica de la ontología y  $d^{if} \in [0, 100]$  es el grado de interés del objeto a través de dicha característica.

Cada ítem también tiene un contador  $RC^i$  que indica el número de veces de valoraciones positivas recibidas.

En la propuesta aplicaron la definición de ontología tanto en el dominio de las películas como el del turismo.

---

<sup>4</sup>Preference elicitation techniques for group recommender systems'

El sistema guarda las informaciones personales y las preferencias generales del cada usuario. Las informaciones que se guardan son las siguientes:

1. Información demográfica y personal del usuario como edad, género, país.
2. El modelo de preferencias generales del usuario: características en las que le interesa a cada usuario.
3. Información de feedback de usuario en cada interacción con el sistema.

## 2.6. Agregación de preferencias en sistemas de recomendación grupal [3]

En los comités de programa de conferencias o en el proceso de captación de talentos donde un grupo de personas tienen que evaluar un conjunto de candidatos y escoger los adecuados, es muy importante agregar correctamente las preferencias individuales para construir el modelo grupal adecuado.

Una forma de agregar preferencias de usuarios es construir una función de puntuación a partir de las puntuaciones de los usuarios. Esta metodología es viable en algunos casos, pero no funcionaría en muchos otros casos, y las causas son las siguientes:

1. Los usuarios puntúan de forma diferente a un mismo candidato o ítem.
2. Los usuarios pueden ser universalmente negativos o positivos.
3. Puede ser que muchos usuarios solamente evalúan a los ítems mejor puntuados o los peor puntuados.

Una solución alternativa sería considerar las preferencias relativas (el ítem A es mejor que ítem B para un usuario, pero no sabemos cuánto de más). De esta forma construimos modelos de preferencias relativas donde podemos asegurar que un ítem en la posición anterior en la lista de recomendaciones es mejor que a los que le siguen. De esta forma propusieron un algoritmo que evita los problemas de las puntuaciones numéricas.

El algoritmo propuesto aquí es un algoritmo de ordenación basado en grafo. La idea es construir un grafo dirigido ponderado:  $G=(V,E)$  donde cada vértice del grafo representa un candidato puntuado por los miembros y el peso de la arista para pasar de un nodo  $v_i$  a otro  $v_j$  es igual al número de usuarios quienes prefieren  $v_i$  ante  $v_j$ . De esta forma encontrar la ordenación óptima de Kemeny tiene un coste computacional lineal. El objetivo del algoritmo es encontrar la ordenación que minimiza la suma de los costes de las aristas.

Este algoritmo es un algoritmo de búsqueda de vecindad variable. Se empieza con una ordenación aleatoria, y se avanza hacia un local máximo. En cada iteración, se intenta encontrar otro local máximo diferente al anterior. Si el máximo que se ha encontrado es mejor al mejor máximo encontrado hasta ahora, entonces solamente se debe realizar pequeños cambios en la solución para localizar este máximo. Pero si no se puede encontrar un máximo con el mínimo movimiento de diversificación, entonces se realiza movimientos más largos. Una vez que se encuentra un máximo, vuelve a buscar un máximo con el movimiento mínimo. El proceso se repite hasta que no puede mejorar la solución.

Para encontrar el máximo local se utiliza el algoritmo de BESTFIT mejorado con CascadeMoveUp y CascadeMoveDown, el cual examina repetidamente cada vértice, y lo inserta en la posición que mejor mejora la función objetiva. El proceso se repite hasta que no se puede mejorar el resultado de ordenación.

Para conjuntos parciales o pequeños se aplica un proceso de diversificación de  $k$  desplazamientos aleatorios. Donde un vértice elegido aleatoriamente es insertado en una posición aleatoria. Cada vez que el algoritmo de búsqueda falla para encontrar el máximo local incrementa el valor de  $k$  en 1 hasta llegar a  $k$  máximo. Para cada vértice se identifica su vértice directamente conectado mayor y menor.

Para mejorar la calidad y la utilidad de la información se utiliza un algoritmo de categorización que incorpora la puntuación original y la ordenación del consenso. Primero se busca vértices no restringidos para reducir la variabilidad. Para ello se crea  $O_h$ , la lista ordenada por la función  $\text{succ}(p)$ <sup>1</sup> y la lista  $O_l$ , la lista ordenada por la función  $\text{pred}(p)$ <sup>2</sup>.

El siguiente paso consiste en asignar una puntuación extra a cada ítem según la posición que ocupa en la lista de consenso y combinarlo con la puntuación original de cada ítem. Calculamos un coste promedio de desplazamiento a partir de la puntuación promedio de los ítems. Considera los vértices del  $O_h$  como buenos a aquellos que tienen el coste de desplazamiento mayor que un valor umbral  $C_{\text{good}}$ . De forma similar consideramos los vértices de  $O_l$  como malos si el coste de movimiento está sobre un valor umbral  $C_{\text{bad}}$ .

El último paso consiste en identificar los vértices ".en conflicto" (Ver ecuación 2.2). Se calcula el porcentaje de preferencias de los que tienen mejor puntuación en un vértice "no bueno" que un vértice "bueno":

$$\text{conflict}(v) = \frac{\sum_{u \notin G} c_{uv}}{\sum_{u \notin G} c_{uv} + c_{vu}} \quad (2.2)$$

Si este valor excede el valor umbral  $C_{\text{conflict}}$ ,  $v$  es considerado como "bueno pero

---

<sup>1</sup> $\text{succ}(p)$ : función para encontrar el vértice directamente conectado mayor.

<sup>2</sup> $\text{pred}(p)$ : función para encontrar el vértice directamente conectado menor.

conflictivo”. Una técnica similar es utilizada para identificar los vértices ”malos y conflictivos”.

En general utilizar preferencias relativas para proporcionar opiniones de los miembros es más preciso que lo que puede proporcionar la clasificación de consenso.

## 2.7. Técnicas de consenso en recomendadores grupales [6]

En esta sección hablaremos sobre cómo combinar las preferencias individuales para llegar a un consenso:

1. En un recomendador grupal, las experiencias del usuario se adquieren de forma implícita o explícitas.
2. Hay dos formas de generar recomendaciones:
  - a) Generamos el conjunto de ítems a recomendar a partir de los ítems que recomendaremos a cada usuario de forma independiente.
  - b) Construimos un modelo de preferencia grupal a partir de los modelos de preferencias individuales para realizar la recomendación.
3. Posteriormente, se presentan y explican a los usuarios las soluciones propuestas por cada uno de ellos.
4. Finalmente, hay un proceso de consenso (automático o manual) en el que hay que escoger 1 solución de entre todas las propuestas.

En la literatura se ha estudiado muy poco sobre técnicas de consensos. Normalmente, estos se clasifican en 3 tipos:

1. Dispersión estadística.
2. Satisfacción individual
3. Filtrado colaborativo

Antes de centrarnos en cada una de estas estrategias. Introducimos las definiciones de los conceptos básicos:

1. Sea  $P = \{p_1, \dots, p_n\}$  el conjunto de productos o ítems
2. Sea  $U = \{u_1, \dots, u_k\}$  el conjunto de usuarios
3. Sea  $IM = \{IM^{u_1}, \dots, IM^{u_k}\}$  el conjunto de modelos de preferencias de los usuarios.

4. Sea  $IM^{u_j} = \{I_1, \dots, I_{r_{u_j}}\}$  el conjunto de preferencias de un usuario, y  $R$  el número total de preferencias del grupo.

Entonces la medida de satisfacción de un usuario con un ítem se calcula de la siguiente manera:

$$\delta(p_i, IM^{u_j}) = \sum_{s=1}^{r_{u_j}} \omega_s * Satisfies(p_i, I_s) \quad (2.3)$$

donde  $\omega_s$  es una ponderación. La satisfacción del usuario se calcula según la cantidad de preferencias que puede satisfacer el ítem. Y esta es la medida básica e imprescindible para la realización de otras estrategias más complejas.

Lo que se pretende en el consenso es elegir el producto que maximiza la satisfacción de los usuarios. Sin embargo, existen estrategias que emplean otro enfoque. Esta idea de maximización de las satisfacciones se puede describir matemáticamente de la siguiente forma:

$$consensus(P, IM) = \arg \max_{p_i \in P} (strategy(p_i, IM)) \quad (2.4)$$

donde *strategy* se refiere a cada posible función de consenso.

Las funciones de dispersión estadísticas para llegar al consenso son:

La función de Mean calcula el promedio de la satisfacción de todos los miembros de un grupo. Se describe como la suma de las satisfacciones de los usuarios dividido entre el número de usuario.

$$m(p_i, IM) = \frac{1}{k} * \sum_{j=1}^k \delta(p_i, IM^{u_j}) \quad (2.5)$$

Tal y como podemos observar en la ecuación 2.5, la función mean es la suma de las satisfacciones de los usuarios, y luego divide el valor acumulado por el número de usuarios.

Otra función de la dispersión estadística es Purity (Véase a la figura 2.6), la función calcula el porcentaje de preferencias satisfechas por el producto o ítem. Como se puede observar dentro de la fórmula utilizamos la desviación estándar del nivel de satisfacción del grupo. Una desviación estándar pequeña significa que las preferencias son muy similares, mientras que, si la desviación estándar es grande, entonces las preferencias de los usuarios son muy diferentes.

$$purity(p_i, IM) = \frac{\sum_{j=1}^k \delta(p_i, IM^{u_j}) - d(p_i, IM)}{R} \quad (2.6)$$

$purity(p_i, IM) = 1$  significa que el producto satisface a las preferencias de todos los miembros del grupo. Mientras que  $purity(p_i, IM) = 0$  significa que el producto

no satisface nada a los usuarios. De esta manera la mejor recomendación es aquel producto que alcanza mayor valor de 'purity'.

A continuación, mostraremos las funciones basadas en la satisfacción individual de cada usuario:

La primera de ellas es 'Completeness' que se basa en la estrategia de dar mayor prioridad a los usuarios con puntuaciones altas, mientras penalizan los usuarios distantes. Véase la ecuación 2.7:

$$completeness(p_i, IM) = \frac{\sum_{j=1}^k w_j \sqrt{\delta(p_i, IM^{u_j})}}{\sum_{j=1}^k \sqrt{r_{u_j}}} \quad (2.7)$$

donde  $w_j$  es un factor normalizado que cumple las siguientes condiciones:

1.  $\sum_{j=1}^k w_j = 1$  la suma de todos factores  $w_j$  ha de igual a 1
2.  $w_j \geq 0$  los factores toman valores positivos.

Este factor determina el peso que tiene cada usuario dentro de su grupo, aquel miembro que tiene mucho peso en la decisión tendrá un valor grande en  $w_j$ . Mientras que aquel miembro que tiene poco peso en la decisión de consenso tendrá un valor de  $w_j$  pequeño. De esta forma, aunque un usuario está muy satisfecho con una recomendación si su influencia en el grupo es pequeña, la probabilidad de recomendar un producto que se ajusta a sus preferencias será baja. El ítem con mayor probabilidad de recomendación es aquel que mejor satisface a los usuarios más influyentes.

Otra alternativa es usar la función de Suficiencia lógica para calcular la proporción de preferencias satisfechas respecto a los no satisfechas para cada usuario (Esta función se aplica a cada uno de los usuarios). Véase a la ecuación 2.8

$$ls(p_i, IM^{u_j}) = \frac{\delta(p_i, IM^{u_j})}{r_{u_j} - \delta(p_i, IM^{u_j})} \quad (2.8)$$

si  $ls(p_i, IM^{u_j}) = 1$  significa que las cantidades de preferencias satisfechas y no satisfechas son las mismas. Si  $ls(p_i, IM^{u_j}) < 1$  significa que el usuario está insatisfecho con el producto, mientras que para  $ls(p_i, IM^{u_j}) \geq 1$  significa que el usuario está satisfecho con el producto.

Para calcular la suficiencia lógica del grupo se aplica la siguiente fórmula:

$$ls_g(p_i, IM^{u_j}) = \frac{\sum_{j=1}^k ls(p_i, IM^{u_j})}{R} \quad (2.9)$$

donde R es el número total de preferencias en el grupo. Tal y como se observa en la función 2.9 la suficiencia lógica del grupo es la suma de la suficiencia lógica de los miembros divididos por el total de preferencias del grupo.

Además, la función suficiencia lógica puede dar falsos positivos, y para tratarla se realiza una pequeña modificación en la función de la suficiencia lógica. Y la nueva

función queda definida de la siguiente forma (véase a la ecuación 2.10):

$$lscontent(p_i, IM^{u_j}) = \frac{\delta(p_i, IM^{u_j}) + 1}{r_{u_j} - \delta(p_i, IM^{u_j}) + 1} \quad (2.10)$$

Los casos de satisfacción nula son peligrosos, porque puede dar casos de falsos positivos, es decir, se recomienda a los usuarios productos de '0' satisfacción al grupo.

Suficiencia Grupal: basado en *lscontent* que mide la satisfacción y la insatisfacción de cada miembro frente a nivel grupal.

$$gs(p_i, IM^{u_j}) = \frac{\frac{\delta(p_i, IM^{u_j}) + 1}{\sum_{s=1}^k \delta(p_i, IM^{u_s}) + 2}}{\frac{r_{u_j} - \delta(p_i, IM^{u_j}) + 1}{\sum_{s=1}^k r_{u_s} - \delta(p_i, IM^{u_s}) + 2}} \quad (2.11)$$

Y la suficiencia grupal para el grupo se define en la figura 2.12

$$gs_g(p_i, IM) = \left( \frac{\sum_{j=1}^k gs(p_i, IM^{u_j})}{k} \right) \quad (2.12)$$

Tal y como se podía observa la suficiencia grupal del grupo es la suma de la suficiencia grupal de cada miembro dividido por la cantidad de miembros que forman el grupo.

Tal como podíamos observar a mayor *gs* menor satisfacción con el producto recomendado. Por lo tanto, la definición más apropiada para el proceso de consenso es minimizar 'gs'. Y el producto final escogido tendrá 'gsg' pequeño. Esta definición se veía en la siguiente figura:

$$consensus(P, IM) = \arg \min_{p_i \in P} (gs(p_i, IM)) \quad (2.13)$$

Por último, vamos a ver las estrategias propuestas habitualmente en el filtrado colaborativo para llegar a un consenso: En primer lugar, 'Least misery'; la cual define la satisfacción de un grupo como el del miembro menos satisfecho. Véase a la figura 2.14

$$least\_misery(p_i, IM) = \arg \min_{u_j \in U} (\delta(p_i, IM^{u_j})) \quad (2.14)$$

Esta estrategia puede penalizar la satisfacción global de grupo, porque se centra en mejorar la satisfacción del usuario quien peor está satisfecho con la recomendación. Se puede dar caso de que el mejor producto satisface de igual a todos los miembros, y el nivel de satisfacción es baja.

En segundo lugar, 'Most pleasure'; la cual mide la satisfacción del grupo según sea el miembro más satisfecho. Véase a la figura 2.15

$$most\_pleasure(p_i, IM) = \arg \max_{u_j \in U} (\delta(p_i, IM^{u_j})) \quad (2.15)$$

La función de 'Most pleasure' se centra en hacer una buena recomendación a uno de los miembros y penaliza al resto. Motivo por el cual, dicha estrategia no es aplicable en los grupos donde los usuarios tienen preferencias muy diferentes.

En tercer lugar, 'Multiplicativo'; que multiplica las satisfacciones de los usuarios. Véase a la figura 2.16

$$multiplicative(p_i, IM) = \prod_{j=1}^k (\delta(p_i, IM^{u_j})) \quad (2.16)$$

Tal y como se veía en la figura 2.16, si un producto no satisface a un usuario, entonces repercutirá directamente en el resultado de la función. Así el producto que satisface igual de bien a todos los usuarios es el que puede alcanzar la puntuación más alta.

Finalmente, 'Borda count', en esta estrategia los usuarios puntúan los ítems según sus preferencias. Posteriormente se procesan estas listas de ítems de la siguiente forma (Véase a la figura 2.17):

1. Ordena los ítems de acuerdo al nivel de satisfacción de cada usuario.
2. Puntúa el último productor con '1', el penúltimo con '2' y así sucesivamente.

$$borda\_count(p_i, IM) = \sum_{j=1}^k (rank_{u_j}^{p_i}) \quad (2.17)$$

El método determina el producto ganador a recomendar. Tal y como se veía en los experimentos realizados concluimos que los métodos 'Multiplicative', 'Completeness' y 'Borda Count' son los más estables y fiables a la hora de recomendar. Funciona muy bien entre diferentes tipos de usuarios y entre grupos de diferentes tamaños.



### 3. Análisis, diseño e implementación

Cómo dijimos en los anteriores capítulos, el sistema de recomendación empleado en este proyecto es un sistema de recomendación grupal conversacional y basado en críticas. Aquí en este capítulo se analizará el diseño del sistema empleado, y comentaremos sobre sus aspectos de implementación.

#### 3.1. Introducción

No disponemos de un sistema de interacción con usuarios en tiempo real debido a su alto coste, sino que empleamos un sistema de pseudosimulación de recomendación. En el cual disponemos de perfiles de usuarios reales, los ofertas para recomendar y también un fichero que contiene los datos de ensayo de nuestro proceso de consenso.

En concreto, en nuestro caso se han implementado 3 grupos de estrategias de consenso. A continuación, se detallan resumidamente:

1. Dispersión estadística:
  - a) Mean: con esta estrategia intentamos buscar el producto con la mejor satisfacción promedio para los miembros de un grupo.
  - b) Purity: Calcula el porcentaje de características positivas respecto a la cantidad total de características del modelo de preferencias del grupo.
2. Satisfacción individual
  - a) Suficiencia lógica: define la relación entre el nivel de satisfacción e insatisfacción del usuario con el ítem recomendado. Su aplicación para calcular el nivel de satisfacción del grupo con el ítem sería la suma de la suficiencia lógica de cada uno de los usuarios y posteriormente dividimos dicho valor por el número total de preferencias que hay en el grupo.
  - b) Suficiencia grupal: Medición del nivel de la satisfacción y de la insatisfacción de un usuario en relación al resto del grupo. Y se aplica esta estrategia a todos los miembros del grupo para obtener la satisfacción e insatisfacción de cada uno de ellos. Por lo último, obtener el promedio de esta relación.
3. Filtro colaborativo
  - a) Máxima satisfacción: la estrategia se centra en alcanzar la máxima satisfacción en uno de los miembros del grupo. En este caso la satisfacción del grupo es al nivel del miembro más satisfecho. Este es uno de los enfoques más usados cuando no se puede obtener un nivel de satisfacción considerable a nivel del grupo. Penaliza al resto de los miembros para

alcanzar la mejor satisfacción de uno de ellos.

- b) *Mínimo miseria*: esta estrategia consiste en establecer el nivel de satisfacción del grupo al nivel de usuario menos satisfecho. Usualmente los enfoques se establecen en la satisfacción de la mayoría en el caso de que no se puede obtener un nivel de satisfacción buena para todo el grupo. Este último enfoque penaliza a los usuarios minoritarios. En la estrategia de mínimo miseria nuestra atención se centra en satisfacer a este grupo minoritario.
- c) *Multiplicativo*: esta estrategia emplea una visión mediadora entre los dos extremos de la recomendación. Aquí se tiene en cuenta la satisfacción de todos los miembros. Puesto que la satisfacción final es el producto de la satisfacción de todos los miembros. Por lo que, los miembros con satisfacción bajas penalizan directamente al nivel de satisfacción global. La mejor satisfacción se consigue cuando todos los miembros están igual de satisfechos.
- d) *Borda count*: esta estrategia está dividido en 2 pasos. En el primer paso ordenamos los ítems de acuerdo a la satisfacción del grupo (De menos satisfecho a más satisfecho). En el segundo paso vamos asignaremos las puntuaciones a los ítems. De forma que el primer producto recibe un punto, el siguiente dos. Si dos productos satisfacen de igual a los usuarios. La puntuación será repartida entre ellos. Por ejemplo, el segundo producto y el tercer producto satisface igualmente al usuario.

$$\frac{2 + 3}{2} = 2,5$$

## 3.2. Diagrama de paquetes

En esta sección del capítulo, se analiza la estructura del proyecto: cómo están distribuidos los paquetes, qué proyecto contiene qué paquetes, ...

Nuestro trabajo está dividido en 2 proyectos: en uno de ellos tenemos implementado todas las lógicas de recomendación y en el otro; la lógica del consenso. La separación entre la lógica del consenso y la lógica de la recomendación permite la realimentación de cada uno de ellos. De esta forma se consigue un sistema de recomendación configurable. Solamente con redefinir la lógica del consenso se obtiene un sistema de recomendación nuevo. El trabajo requerido se reduce sustancialmente respecto a la reimplementación de la lógica en un único proyecto. En este caso se requiere mayor esfuerzo de para analizar las dependencias de las clases y separarlas. En la figura 3 se puede observar la estructura del paquete del presente trabajo:

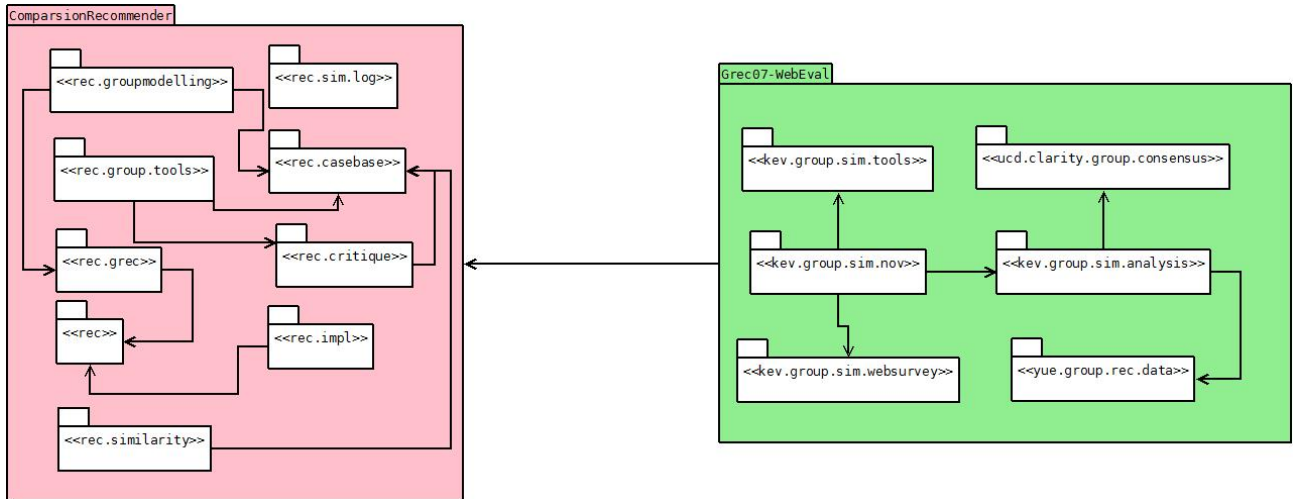


Figura 3: Diagrama de paquetes del proyecto

Tal y como se veía hay dos proyectos en el trabajo: 'Grec07-WebEval' y 'comparisionRecommender'. Las implementaciones y modificaciones se realizan en el proyecto de 'Grec07-WebEval'. Este es el proyecto donde se ha implementado la lógica del usuario. Según el flujo de funcionamiento del programa se observa que el proceso comienza en el paquete 'kev.group.sim.nov'. Este paquete contiene el conjunto de clases donde se han implementado los diferentes 'tests' de consenso. Los paquetes 'kev.group.sim.tools' y 'kev.group.sim.websurvey' contiene el conjunto de clases necesarias para la creación de los perfiles de los usuarios, y también de los grupos. El paquete de 'kev.group.sim.analysis' tal y como se indica el nombre del paquete, este contiene el conjunto de clases que implementa la lógica del proceso de análisis. En el marco del nuestro trabajo, no hay sentido hablar de análisis sin mencionar el consenso. Porque nuestro análisis recae directamente sobre las estrategias de consenso. Partiendo de esta relación, se concluye que ambos paquetes 'kev.group.sim.analysis' y 'ucd.clarity.group.consensus' implementan el proceso de consenso.

Por otra parte, está el proyecto de 'comparisionRecommender', en este proyecto se han implementado el conjunto de paquetes que constituyen la parte de la recomendación. Cabe recordar que el proceso de recomendación se divide en dos etapas: la primera de ellas se recomienda un conjunto de productos a los usuarios, y en la segunda etapa, se debe llegar a un consenso, es decir, qué producto escoger. En el proyecto de 'comparisionRecommender', el trabajo se centra en la primera etapa.

Puesto que la ejecución del programa se empieza siempre desde el paquete 'kev.group.sim.nov', sobre los paquetes del proyecto de 'comparisionRecommender' se hablará solamente sobre la función que realiza cada paquete. En primer lugar, tenemos al paquete 'rec' el cual crea una sesión para cada recomendación que se quiere hacer. El paquete 'rec.grec' contiene las clases que implementa un recomendador sin consenso. El paquete 'rec.groupmodelling' contiene las clases de modelación de los grupos. En cuarto lugar, el paquete 'rec.casebase' permite definir el conjun-

to de productos que se puede recomendar. Y para la representación de las críticas se utiliza las clases contenidas en el paquete 'rec.critiquing'. Existe varios tipos de críticas, y se ha implementado las clases necesarias para tratar todas las que se hacen falta (críticas ordinales, cardinales, unitarias, ...).

El proceso de recomendación se inicializa con la petición del usuario, y el paquete 'rec.group.tools' contiene las clases que implementa este paso.

### 3.3. Diagrama de la clase

Un proceso de recomendación se comienza con la petición del usuario, después el sistema se procederá a analizar las preferencias entradas por el usuario, y busca los productos que satisface dichas preferencias. En la mayoría de los casos, el usuario ha de escoger un único producto entre los recomendados. En el presente trabajo se simula todo el proceso de recomendación con el fin de obtener los resultados de forma rápida y analizarlos. El proceso de recomendación se ha dividido principalmente en 3 pasos: el primero de ellos es la obtención de las preferencias de los usuarios, a continuación, a partir de estas preferencias se recomiendan los productos, y en el último paso se aplica diferentes estrategias de consenso para decidir qué producto escoger.

En las siguientes figuras se muestran la implementación en clases del presente sistema de recomendación (Véase a la figura 4):

La información de los perfiles de los usuarios es imprescindible para poder hacerle una recomendación, la clase con la que se guarda las informaciones de los perfiles es la clase 'UserProfile'. Primero se instancia la clase FileParser para parsear los ficheros de perfiles de los usuarios. Y estos perfiles con los que se utilizan posteriormente para crear los grupos de usuarios.

Es conveniente no realizar solamente el análisis de la satisfacción de los grupos, sino también otros aspectos como: la similitud entre los productos perfectos de cada usuario de un grupo, la similitud entre el producto escogido y el producto perfecto de cada usuario (Véase a la figura 5).

La modelación de las preferencias grupales es básico y imprescindible para que el sistema pueda recomendar algo. Tal como se veía en la figura 6 el cuál modela un modelo con el conjunto de críticas de los usuarios del grupo. Las críticas se dividen son unitarias para facilitar su procesamiento

La clase 'consensusComputer' aplica el conjunto de estrategias de consenso para simular de forma automática el proceso de consenso.

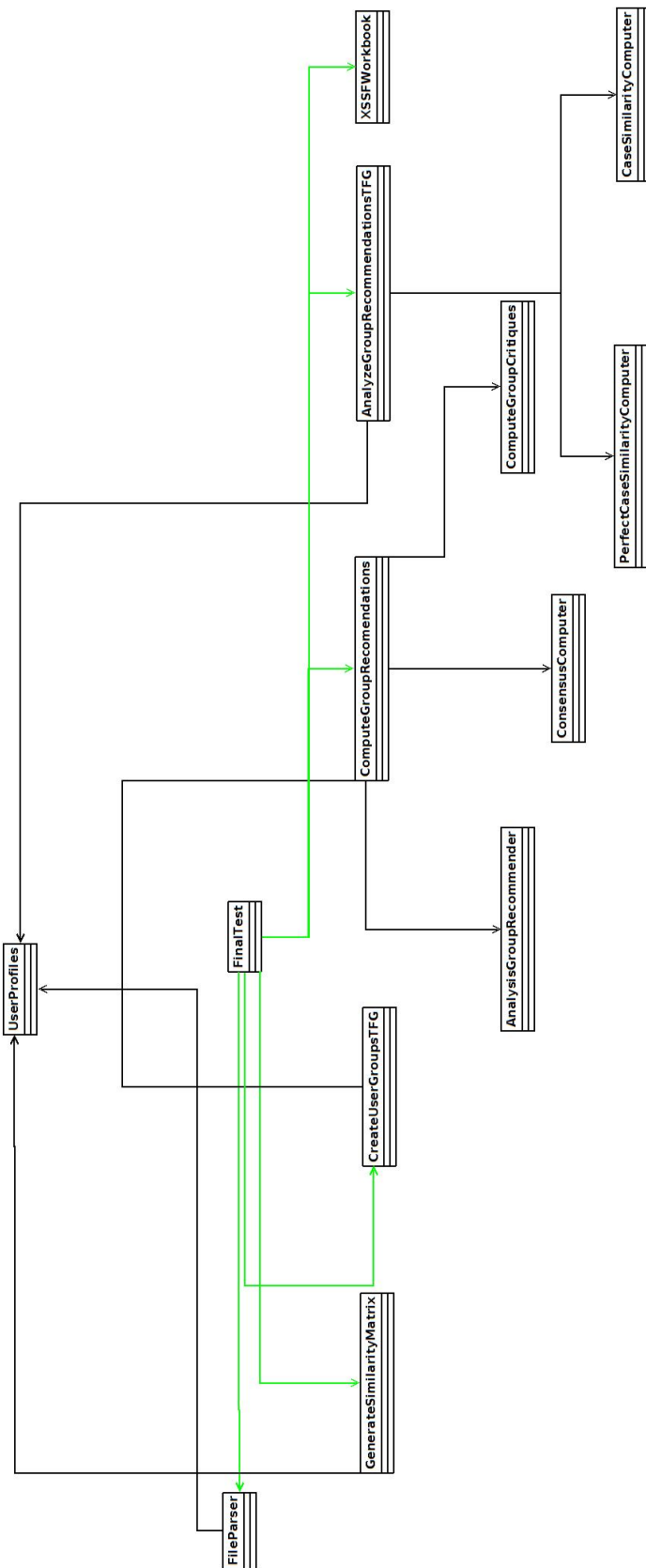


Figura 4: Diagrama de clase

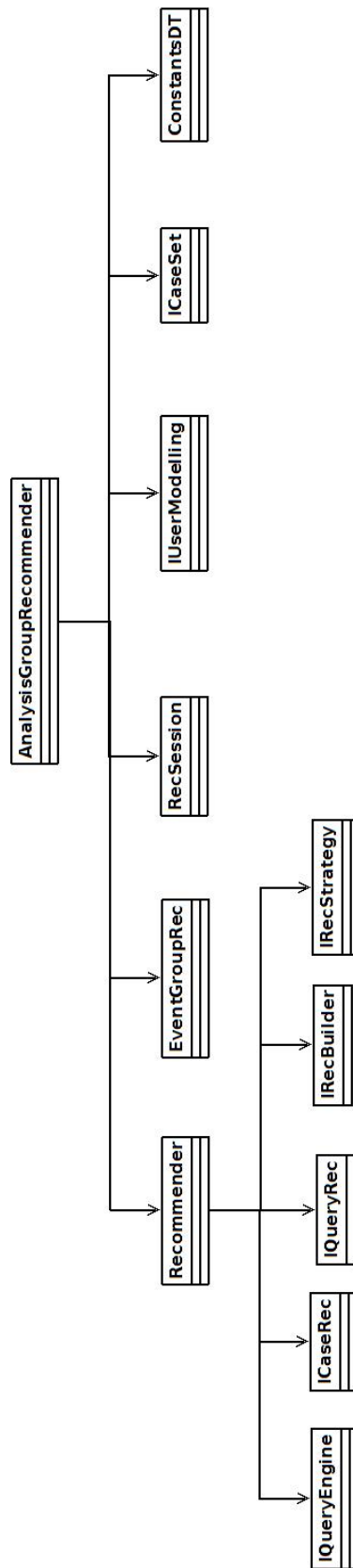


Figura 5: Diagrama de clase correspondiente a `AnalysisGroupRecommender`

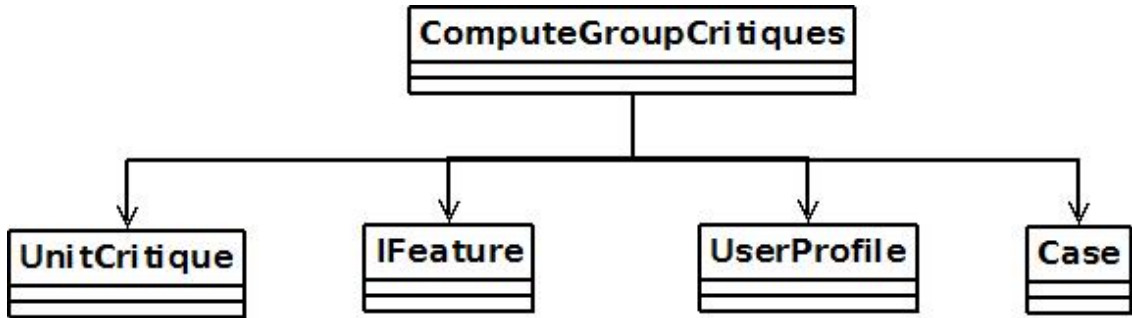


Figura 6: Diagrama de clase correspondiente a ComputeGroupCritiques

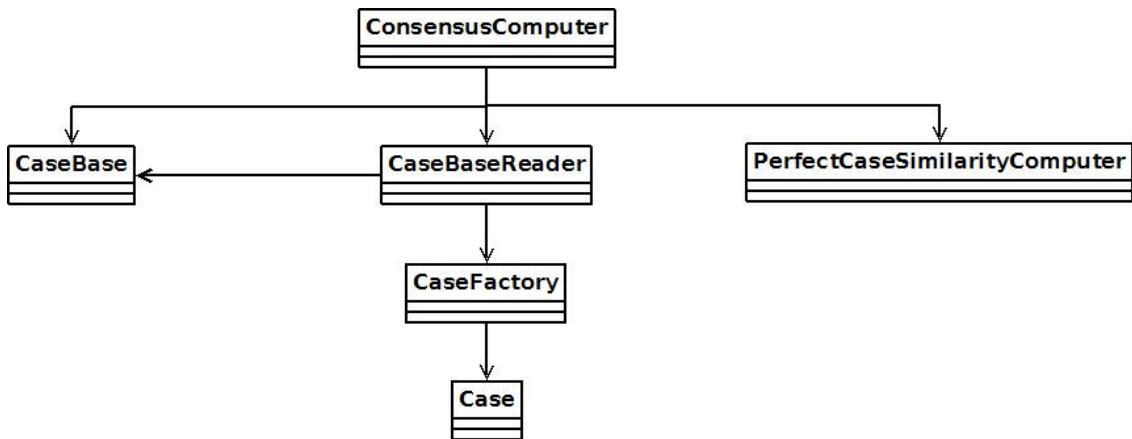


Figura 7: Diagrama de clase correspondiente a ConsensusComputer

### 3.4. Diseño e implementación

La implementación está realizada en Java, no sólo porque es una continuidad de un trabajo, sino también es una plataforma donde tenemos un sinfín de recursos para aprovechar su rendimiento al máximo. En nuestro caso, utilizamos librería que nos facilita nuestra tarea de almacenamiento de datos en un fichero de Excel. El cual nos abre la puerta a muchas comodidades como utilidades de representación de las informaciones. Por ejemplo: gráficas de comparación del rendimiento de los métodos implementados.

Cómo indicamos no disponemos de usuarios reales interaccionando con el sistema en tiempo real. Sino que utilizaremos los perfiles de usuarios reales creados anteriormente, mediante proceso de especificación de preferencias. Se disponen 3 ficheros principales que contiene el conjunto de informaciones básicas: -HotelResorts.txt: el fichero contiene el conjunto de informaciones sobre las ofertas disponibles para recomendar. -QueryFile0.txt: que contiene el conjunto de resultados de la simulación del consenso.

El sistema completo está dividido en dos proyectos diferentes: -comparisonRecommender: contiene el conjunto de clases de soporte. Por ejemplo: el conjunto de clases -GRec07-Web4Val: contiene el conjunto de clases que implementa la lógica funcional del recomendador.

La clase principal de nuestro sistema es 'FinalTest', el funcionamiento de esta clase es muy simple y consiste en los siguientes pasos:

1. En la primera etapa se inicializa el programa, y posteriormente se procede a carga todas las informaciones: perfiles de usuarios, los productos para recomendar. Véase a la figura 8
2. Como se puede observa la clase realiza la llamada a la clase 'FileParser' para extraer las informaciones contenidas en los ficheros de perfiles de usuarios. Y prepara la matriz de la similitud de los usuarios para la generación de grupos de usuarios.
3. En este paso se realizan las recomendaciones a los usuarios, la opción elegido por el sistema es la creación de un modelo de preferencias grupales, a partir del cual se calculan las satisfacciones de los grupos con cada uno de los ítems. Se compara la similitud entre el ítem perfecto y el ítem seleccionado. Y la ítem con la mejor similitud es la recomendada. Véase a las figuras 9, 10 y 11

En esta etapa se calculan más informaciones sobre el ítem recomendado. (Compatibilidad con el ítem perfecto, la similitud entre los casos perfectos de los diferentes usuarios del grupo, etc).



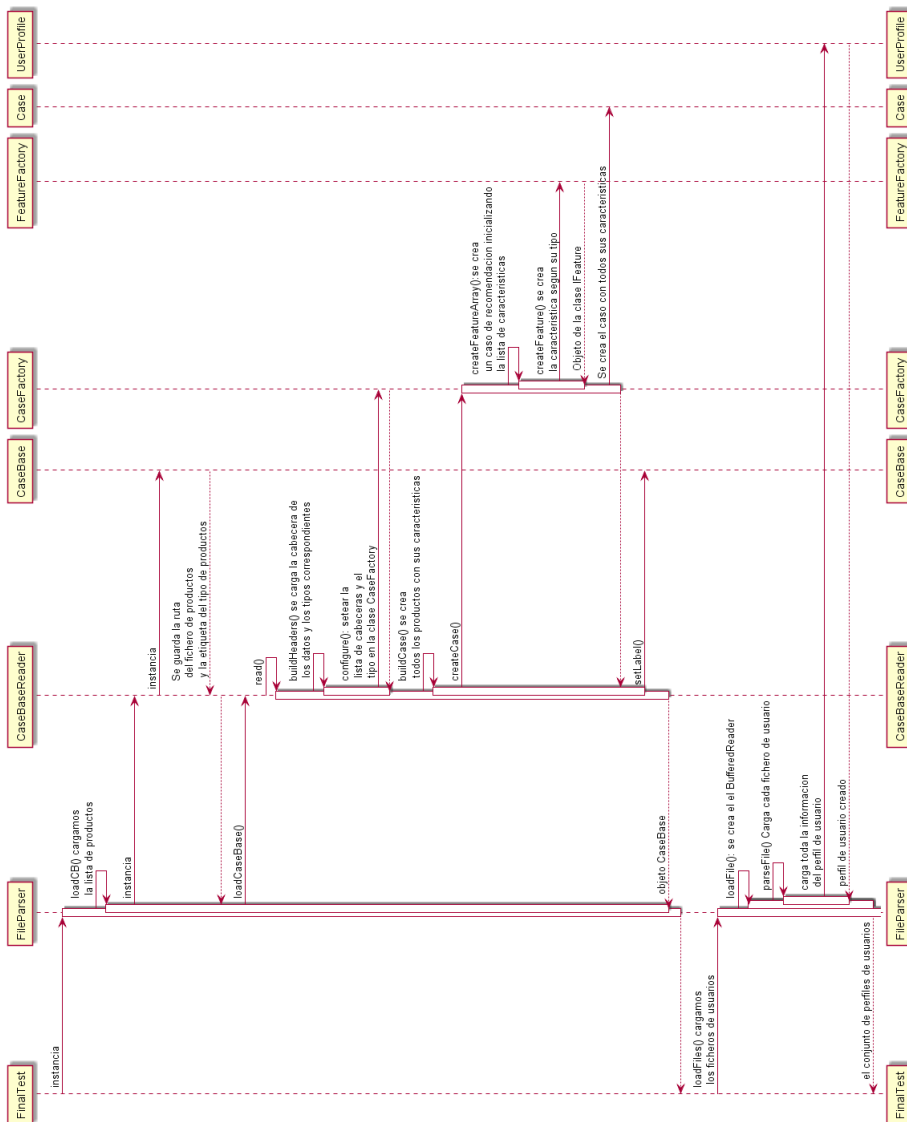


Figura 8: Diagrama de secuencia de la lectura de los datos

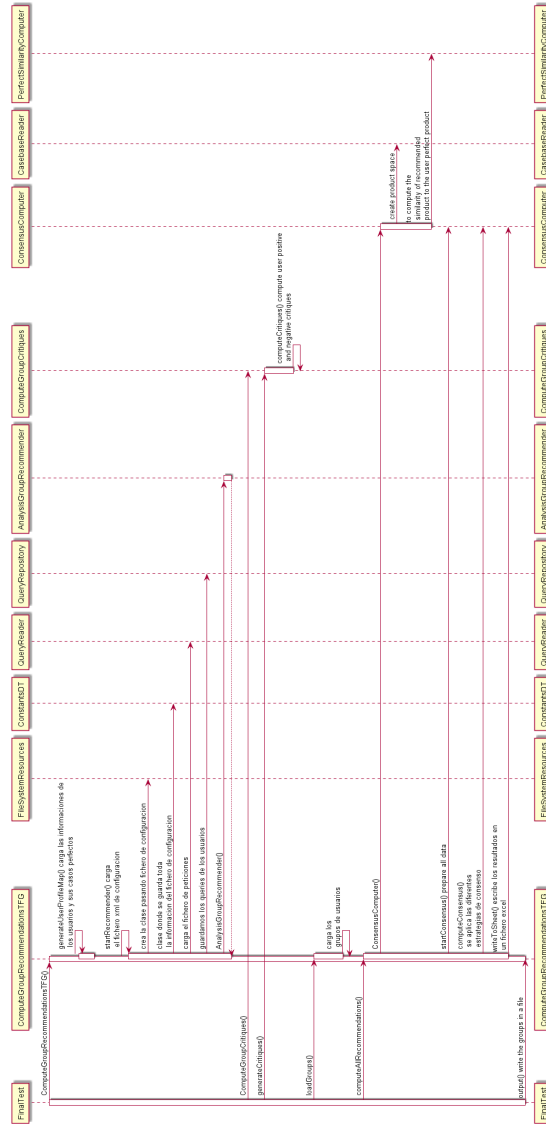


Figura 9: Diagrama de secuencia correspondiente al proceso completo de recomendación

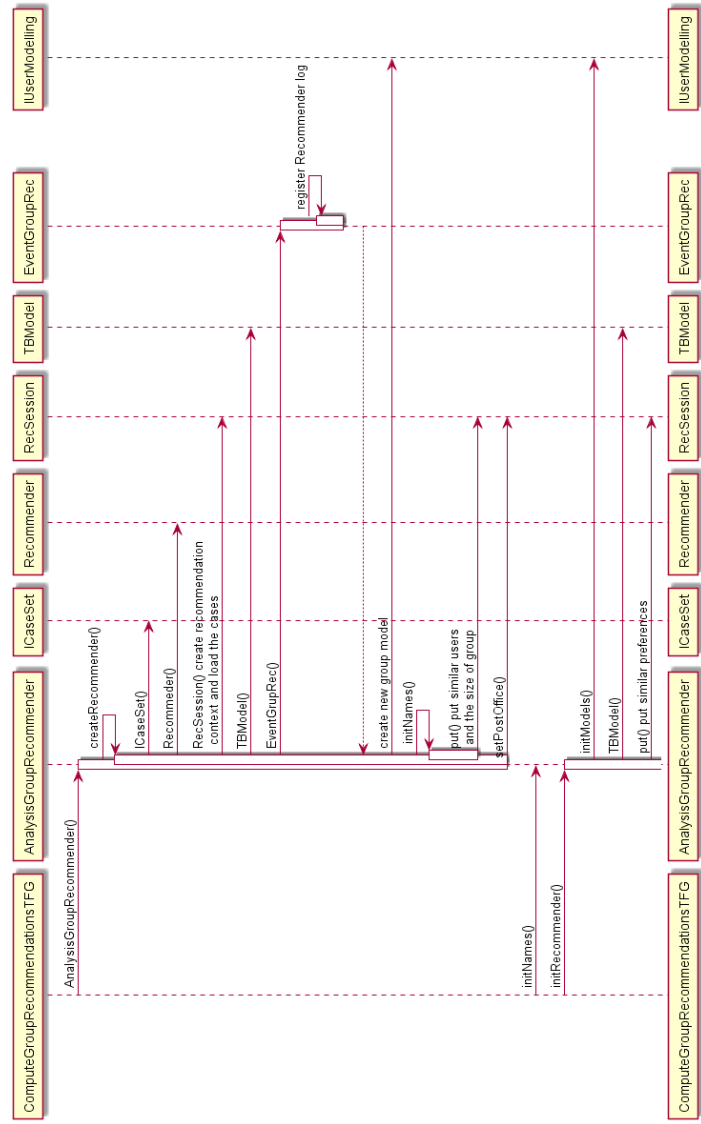


Figura 10: Diagrama de secuencia correspondiente al proceso de recomendación de productos

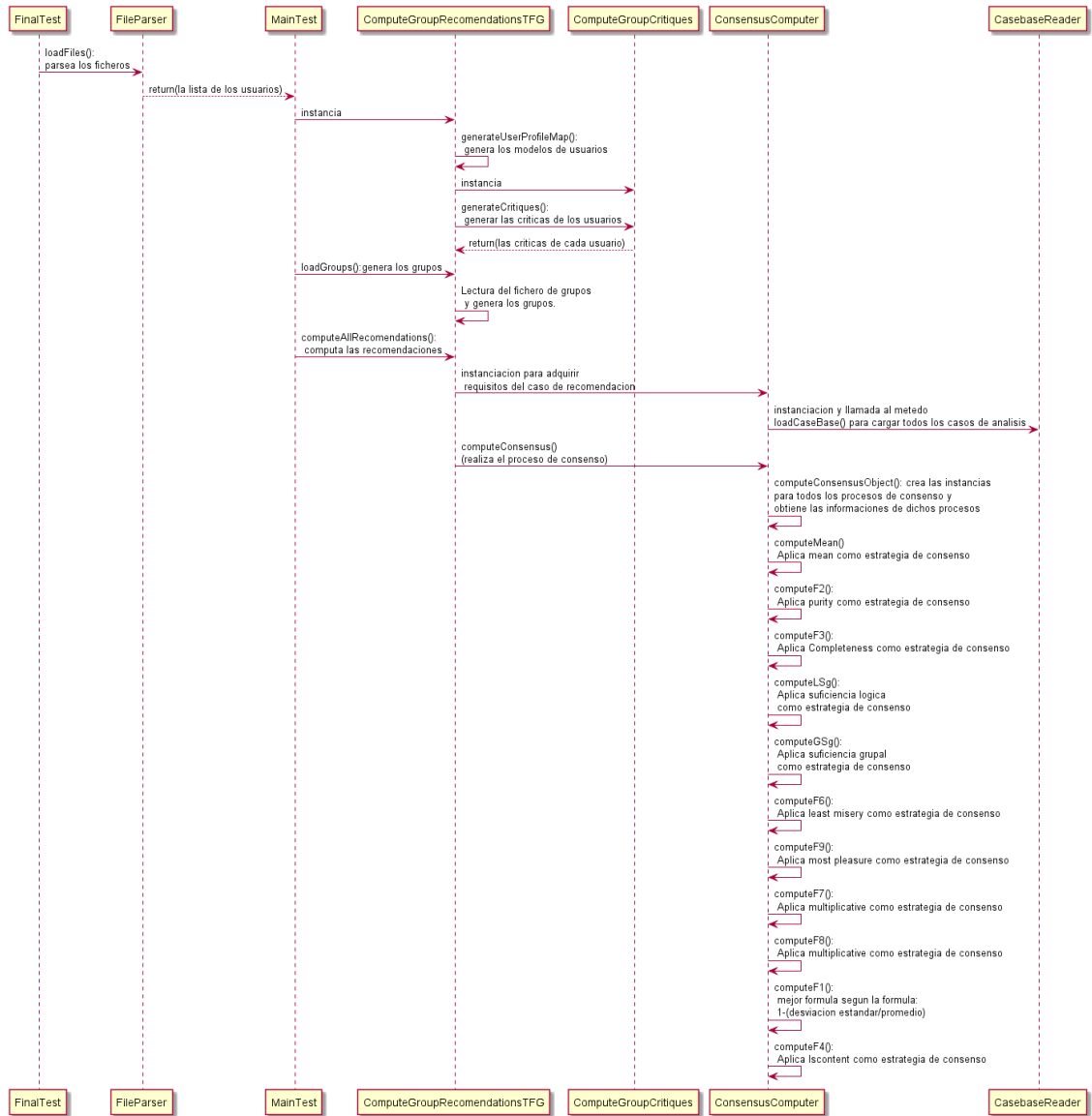


Figura 11: Diagrama de secuencia correspondiente al proceso de consenso

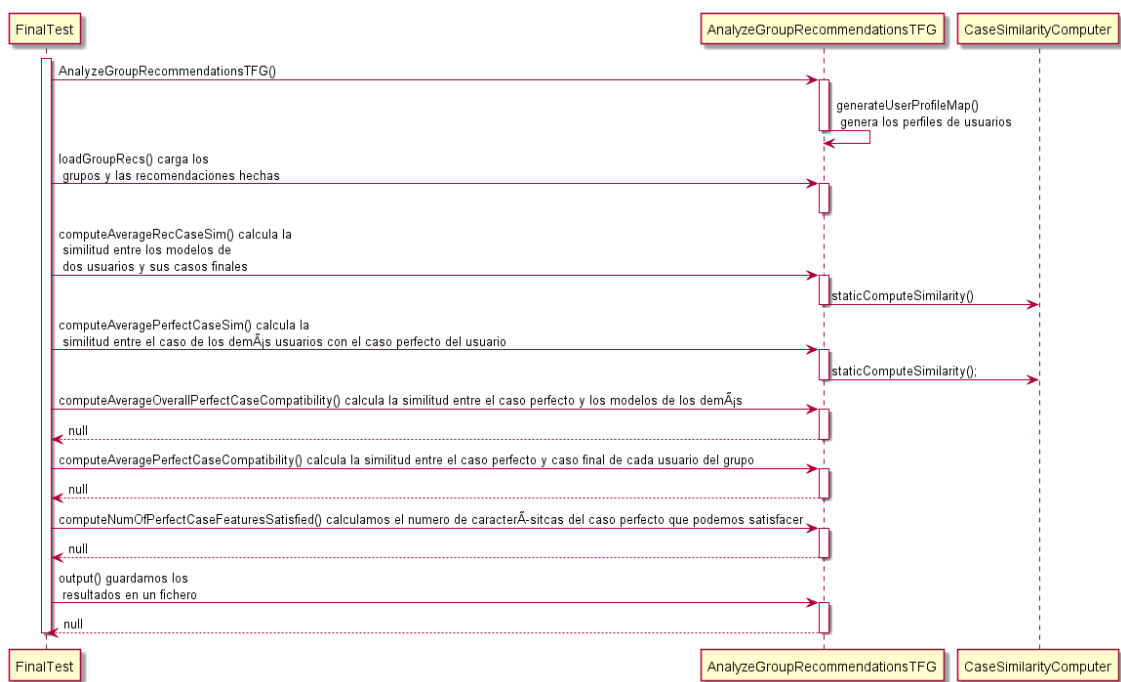


Figura 12: Diagrama de secuencia correspondiente al análisis de similitudes

## 4. Análisis de resultados

En este capítulo se analiza los resultados obtenidos en la recomendación. Para facilitar la visualización de los resultados, representaremos los resultados en gráficas. Idealmente realizar el proceso completo con grupos de usuarios reales proporcionan más fiabilidad al resultado de nuestras estrategias, pero debido a su alto coste temporal como la dificultad de reclutamiento de voluntarios; nuestro sistema ha generado los grupos de usuarios de forma computacional pero utilizando perfiles de usuarios reales.

### 4.1. Metodología

En nuestro trabajo, aunque el proceso de consenso no se ha realizado físicamente con los usuarios reales, pero disponíamos de los perfiles de usuarios reales. En total disponemos 34 usuarios, 151 ofertas de packs de esquí para recomendar. Entre los miembros del grupo puede haber preferencias contradictorias, y estas preferencias afectan directamente al resultado de la recomendación. Pudiendo bajar el nivel de satisfacción del grupo notoriamente. Los grupos con similitudes bajas son más difíciles de satisfacer, la mejor solución es dejar a que los usuarios eligen el ítem de consenso.

Ante estas situaciones, la estrategia mencionada anteriormente de dejar la toma de decisiones a los propios usuarios nos puede beneficiar. La negociación de los usuarios para resolver las preferencias conflictivas puede perdurar bastante tiempo teniendo en cuenta las posibles ausencias en una reunión. Y muchos sistemas opta por crear servicios de mensajerías como grupos de conversaciones, para que los ausentes puedan visualizar las propuestas y optar por alguna de ellas o proponer nuevas. En casos de que los usuarios se conocen bien entre ellos y llevan buenas relaciones, ellos intentarán modificar sus preferencias para conformar a los otros. En el sistema de recomendación del presente proyecto el proceso de consenso es automático, se simula el proceso de consenso. De esta forma, solucionamos de forma más rápida el conflicto, pero se puede bajar el nivel de satisfacción con el producto final.

Realizar el proceso con los usuarios reales es muy costoso, se necesita tiempo y recurso. Por lo que optamos en obtener perfiles completos de usuarios. Y posteriormente generar los grupos de usuarios de forma programada.

Las características de nuestro recomendador se clasifican en nominal y ordinal. Se necesitan 2 tipos de cálculo diferente para la asignación de satisfacción. Para características nominales, se definen 2 casos posibles: satisface (una puntuación de 1) o no (puntuación 0). Para las características ordinales se calcula la similitud entre ambos. Y el criterio para calcularlo es el siguiente (Los valores están normalizados):

$$Similitud = 1 - \frac{abs(x - y)}{Max(x, y)}$$

Donde x, y son las dos características que hemos de comparar.

Para testear el rendimiento de las estrategias, se procede a crear grupos de diferentes tamaños. Nuestro objetivo es crear grupos de usuarios de 3, 4, 6 y 8 personas. Puesto que estos grupos son generados computacionalmente, se necesita alguna estrategia para calcular tanto la similitud entre usuarios como para valorar a los ítems recomendados.

Una vez poder calculado las similitudes de los grupos, clasificamos los grupos en 3 tipos:

1. Grupo Similar: este grupo tiene alta similitud entre sus usuarios, lo cual permite llegar un mayor nivel de satisfacción de los usuarios con la recomendación y es más fácil de hacerles recomendaciones. Este grupo tiene similitud entre 0.53 a 0.82
2. Grupo Mixto: la similitud de este tipo de grupo es más baja que la anterior, porque los usuarios no tienen los perfiles semejantes. Con similitud entre 0.4 a 0.5
3. Grupo Diverso: la similitud de este tipo de grupo es baja, lo cual dificulta mucho el proceso de recomendación, además el nivel de satisfacción de integrantes con la recomendación es bajo.

En los ficheros de perfiles de usuarios, tenemos información del ítem escogido por usuario. Dicho caso se llama 'ChosenCase'. Aquí también tenemos la definición de las características de los ítem perfecto. El ítem perfecto es el producto que satisface totalmente a un usuario. Y este es el que utilizamos para calcular la satisfacción del ítem recomendado. Una recomendación será 'perfecto' si sus características son idénticas a esta.

Una de las ventajas de nuestro recomendador es su configurabilidad, la cual nos permite elegir cómo modelar los grupos de usuarios. En el presente sistema de recomendación se modelan los grupos de 4 formas diferentes. Las cuatro formas de modelación de los grupos son: Nash, Joint, AverageGroup y AverageIndivgroup. A continuación explicaremos cada una de ellas: El primero de ellos que vamos a explicar es Nash, 'Nash' crea modelos de preferencias individuales de cada usuario, y para cada producto que se recomienda se calcula las preferencias que se pueden satisfacer con el producto, a este resultado se le llama la compatibilidad del producto. Y la compatibilidad del ítem con la estrategias de modelación 'Nash' es la multiplicación de las compatibilidades del ítem con cada uno de los miembros. La siguiente estrategia que vamos a explicar es Joint, Joint crear un modelo de preferencias del grupo y para calcular la compatibilidad del producto se utiliza las preferencias del modelo grupal. En el modelo grupal se eliminan las redundancias de las preferencias. En tercer lugar tenemos a AverageGroup, el cual crea los dos modelos de preferencias, y calculan la compatibilidad del producto con cada uno de estos modelos. La compatibilidad final es la combinación de ambas compatibilidades. Para ellos se define un parámetro  $\alpha$  como peso que tendrá la compatibilidad del modelo individual y el peso de la compatibilidad del modelo grupal es  $1-\alpha$ . La fórmula se puede observar en la siguiente figura 4.1:

$$final\_compatibility = \alpha * individual\_compatibility + (1 - \alpha) * global\_compatibility \quad (4.1)$$

Finalmente, vamos a explicar AverageIndivGroup, el cual crea solamente el modelo de preferencias individuales. Y la compatibilidad final es el promedio de la compatibilidad de cada usuario.

Para tener una línea base de comparación, optamos por realizar una recomendación aleatoria a los grupos. Después se calcula el beneficio obtenido con el resultado de la recomendación del mejor ítem respecto al ítem aleatorio. Con cada una de las estrategias se calcula la puntuación de cada uno de los productos, se ordenan de forma descendentes según las puntuaciones recibidas y el producto con la puntuación más alta es el escogido. En promedio, se disponen de 7 críticas nominales y 3 o 4 críticas cardinales por cada usuario.

## 4.2. Análisis y conclusiones de los Resultados

En esta sección analizaremos las gráficas de los resultados, estudiaremos como influye el tamaño del grupo a la satisfacción del grupo. También se estudia si existe alguna relación entre las similitudes de los miembros del grupo con la satisfacción final.

Las gráficas de abajo son generadas a partir de 4 modelaciones distintas de los grupos. El beneficio que nos aporta con la mejor recomendación puede variar en función del ítem aleatorio que se recomienda. Para estimar con mayor precisión el beneficio que se puede obtener con el presente sistema se calcula el beneficio promedio que se obtiene con las 5 mejores recomendaciones. El beneficio es un porcentaje que se calcula de la siguiente forma:

$$beneficio = \frac{score_{top} - score_{random}}{score_{random}} * 100 \quad (4.2)$$

Tal como se veía en la figura 4.2,  $score_{top}$  es la satisfacción del grupo con el producto mejor valorado.  $score_{random}$  es la satisfacción del grupo con el producto recomendado de forma aleatorio.

Los resultados de abajo están calculados con grupos de 3, 4, 6 y 8 usuarios. Y las gráficas que se visualizan corresponde a los siguientes resultados:

1. Satisfacción promedio de los grupos con el producto mejor valorado.
2. Beneficio que se obtiene con el anterior producto.
3. Satisfacción promedio de los grupos con las 5 mejores recomendaciones.



4. Beneficio que se obtiene con las 5 mejores recomendaciones.

Abajo se muestran las gráficas de resultados de satisfacción de los grupos con la mejor recomendación( Véase a las figuras 13, 14, 15 y 16):

En general, la satisfacción de los grupos del mismo tamaño tiende a bajarse cuan-

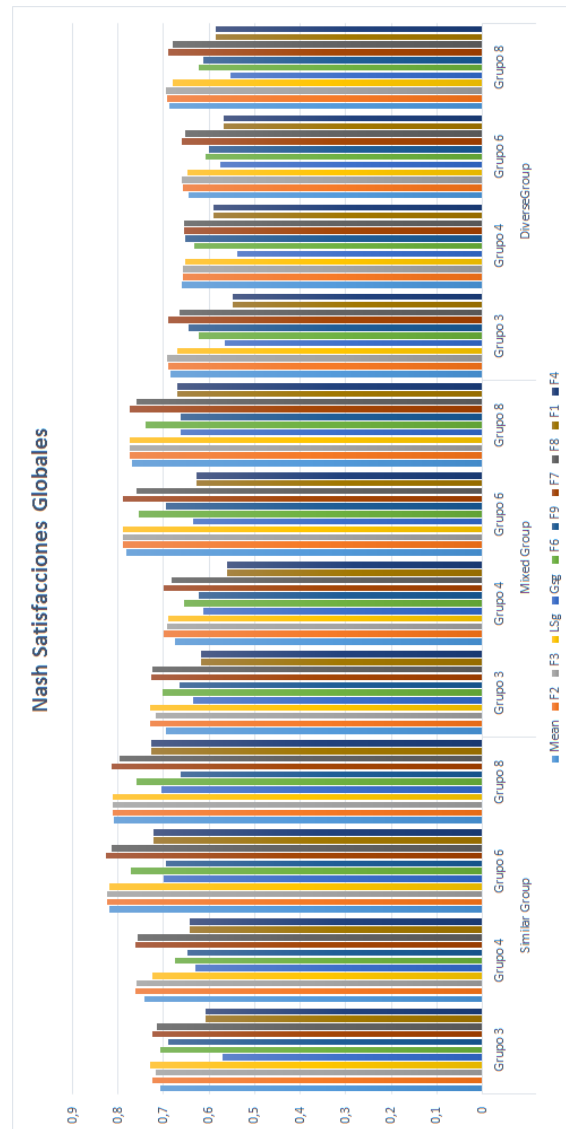


Figura 13: Satisfacción con la modelación Nash

do la similitud entre los usuarios baja. Esta tendencia es muy clara en los grupos de 4, 6 y 8 usuarios. Mientras que con los grupos de 3 usuarios el fenómeno es distinto.

La tendencia de la satisfacción de los grupos es la misma que el caso anterior.

El mismo comportamiento con la modelación AverageAllGroup.

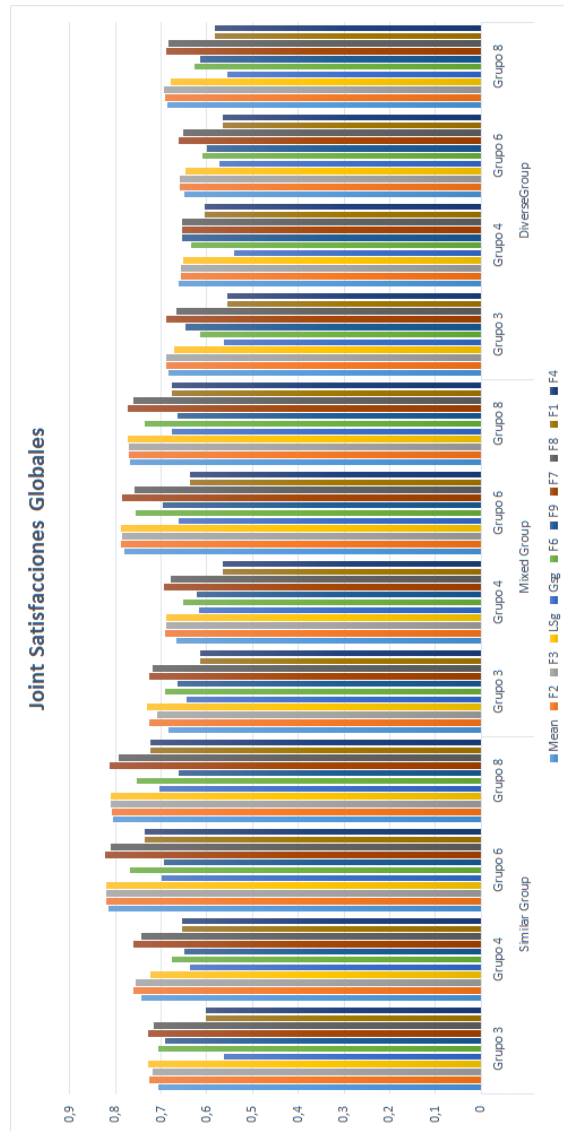


Figura 14: Satisfacción con la modelación Joint

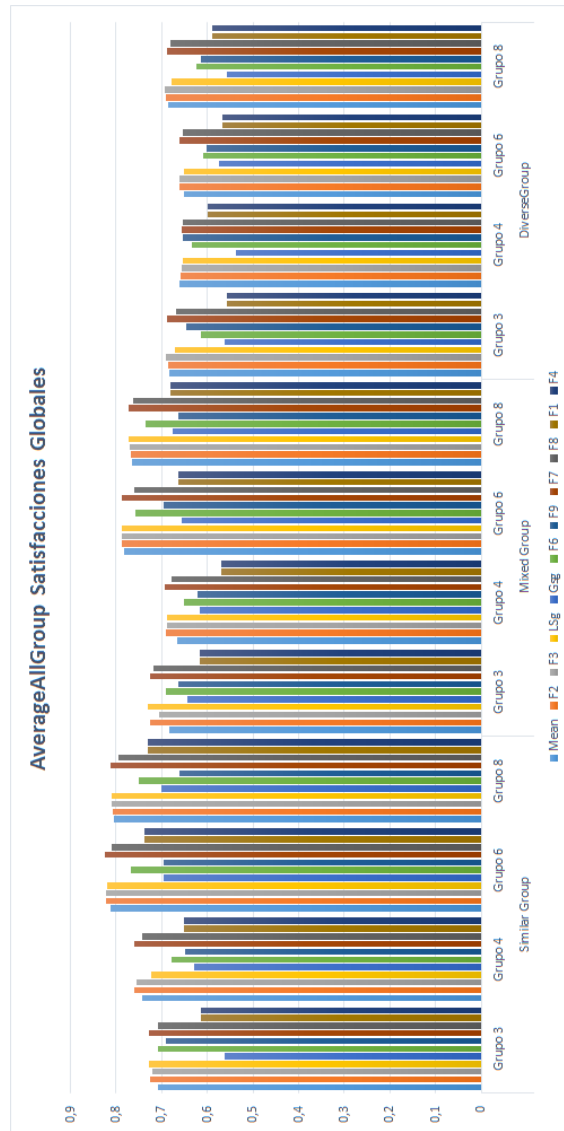


Figura 15: Satisfacción con la modelación AverageAllGroup

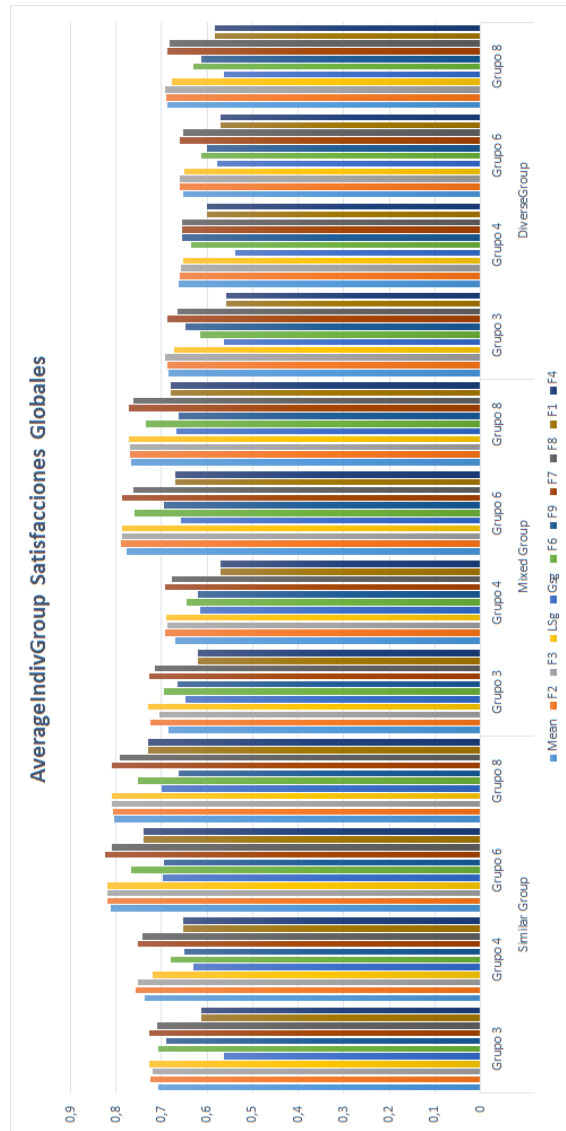


Figura 16: Satisfacción con la modelación AverageIndivGroup

Se puede concluir que para los grupos del mismo tamaño a medida que la similitud entre los usuarios bajan, la satisfacción del grupo con la recomendación también baja. Pero no podemos concluir nada sobre cómo afecta el tamaño del grupo al nivel de satisfacción grupal.

Ahora toca analizar las gráficas del beneficio obtenido con cada una de las estrategias de modelación para la mejor recomendación( Véase a las figuras 17, 18, 19 y 20 ):

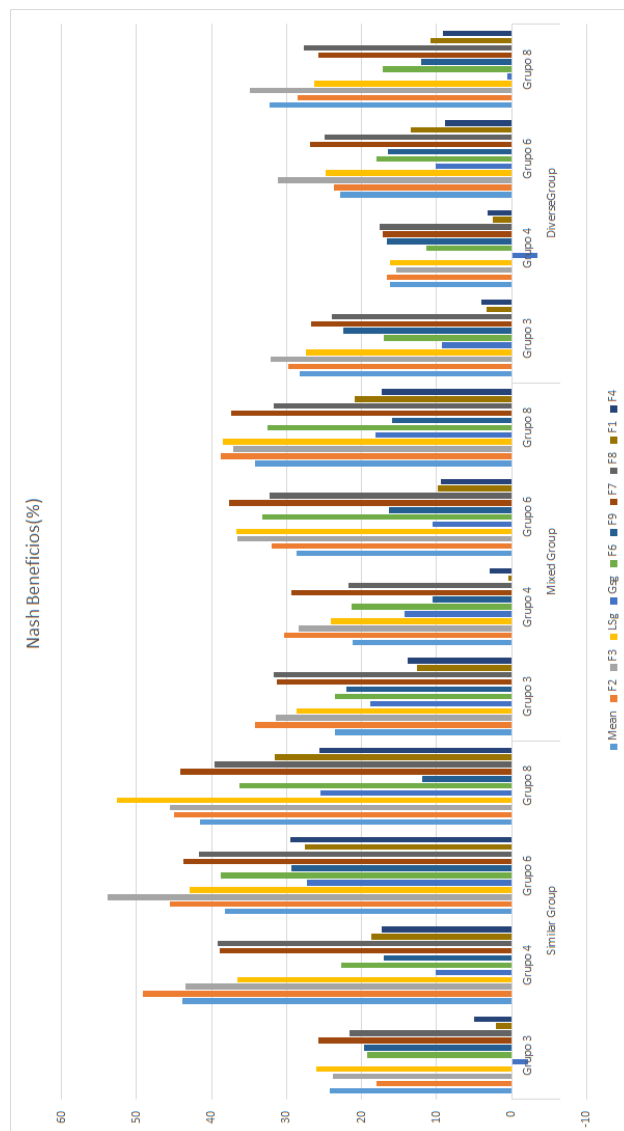


Figura 17: Beneficio obtenido con la modelación Nash

Con la mejor recomendación se obtiene mayor beneficio cuando las preferencias de los usuarios son similares, porque es más fácil de satisfacer a todos si comparte muchas preferencias en común que si se comparten poco. El beneficio que se obtiene puede ser muy relativo, porque dependen de qué producto le recomendamos a

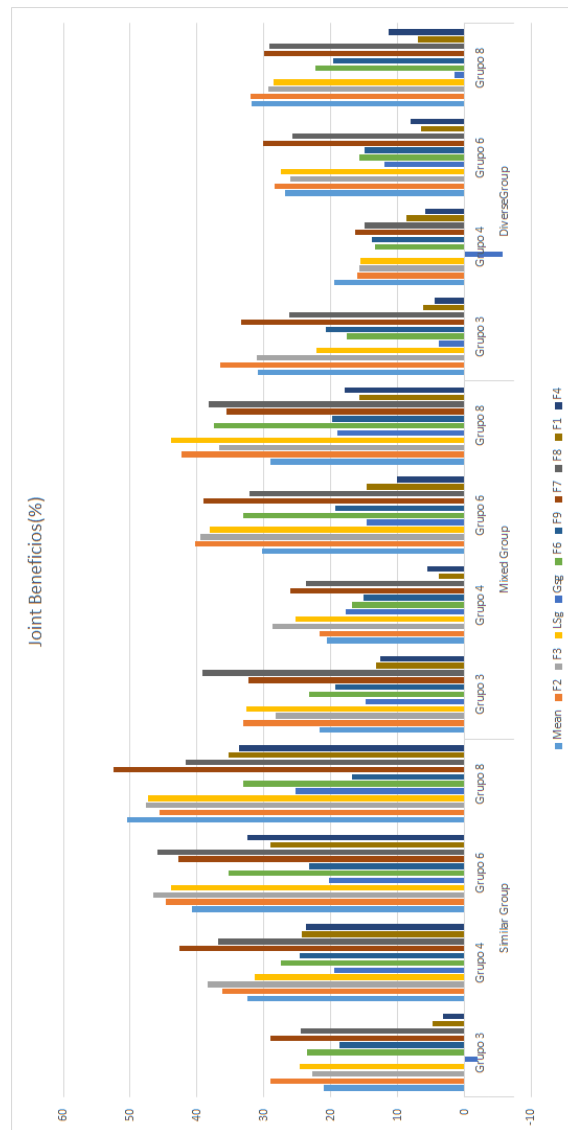


Figura 18: Beneficio obtenido con la modelación Joint

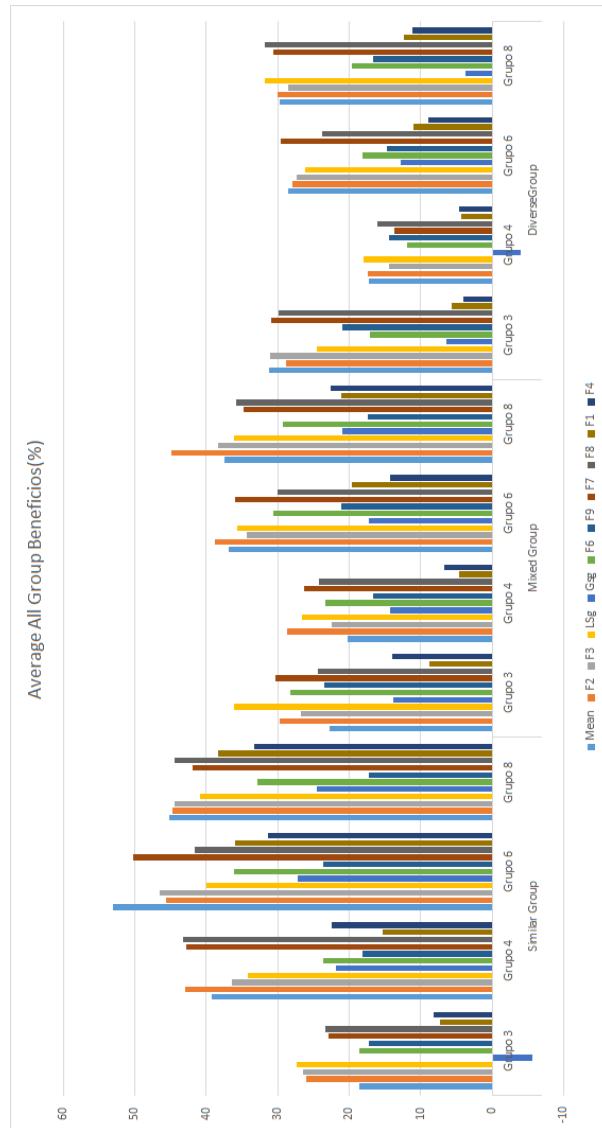


Figura 19: Beneficio obtenido con la modelación AverageAllGroup

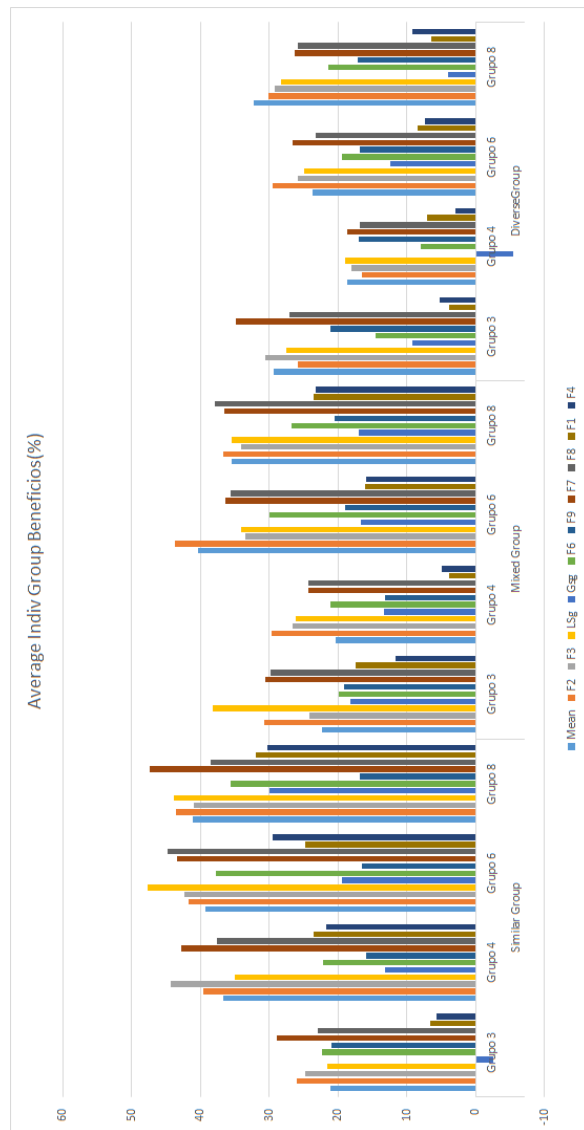


Figura 20: Beneficio obtenido con la modelación AverageIndivGroup



los usuarios. La recomendación está hecha de forma aleatoria, por lo tanto, en el resultado de beneficio podemos obtener beneficios diferentes.

Ahora toca a analizar los resultados obtenidos con las 5 mejores recomendaciones. En primer lugar analizaremos la satisfacción promedio de los grupos con los 5 mejores recomendaciones (Véase a las figuras 21, 22, 23 y 24):

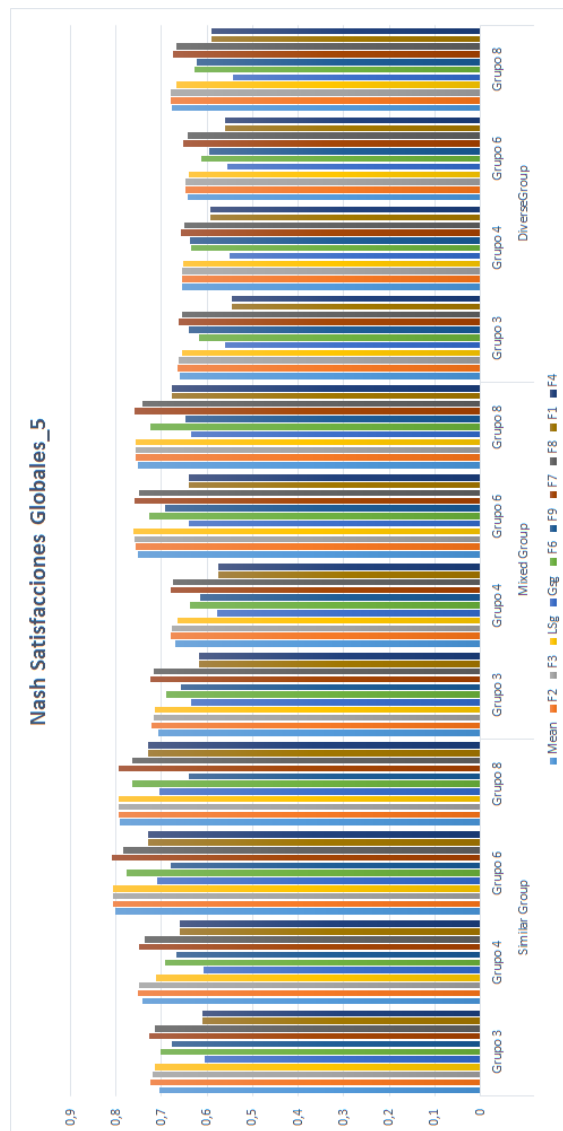


Figura 21: Satisfacción con la modelación Nash

Las satisfacciones de los grupos con los 5 mejores productos marcan líneas similares. La satisfacción entre grupos con los mismos intervalos de similitudes es muy parecida. Y la tendencia que marca los resultados es la misma. Si la similitud entre los usuarios baja, la satisfacción con la mejor recomendación también baja.

Ahora toca estudiar el beneficio que se obtiene con las 5 mejores recomendaciones (Véase a las figuras 25, 26, 27 y 28):

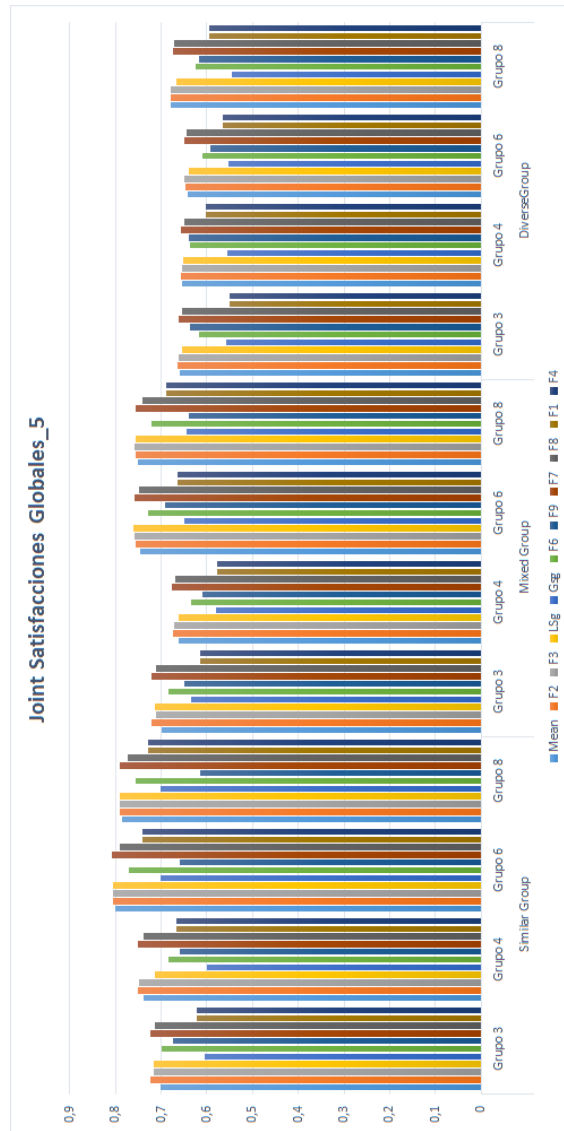


Figura 22: Satisfacción con la modelación Joint

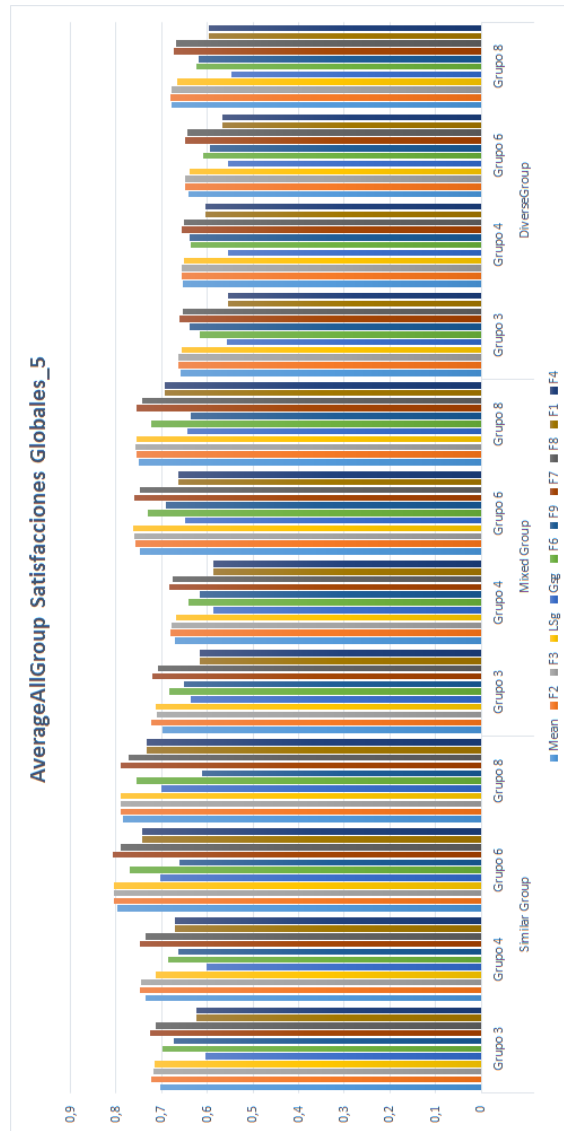


Figura 23: Satisfacción con la modelación AverageAllGroup

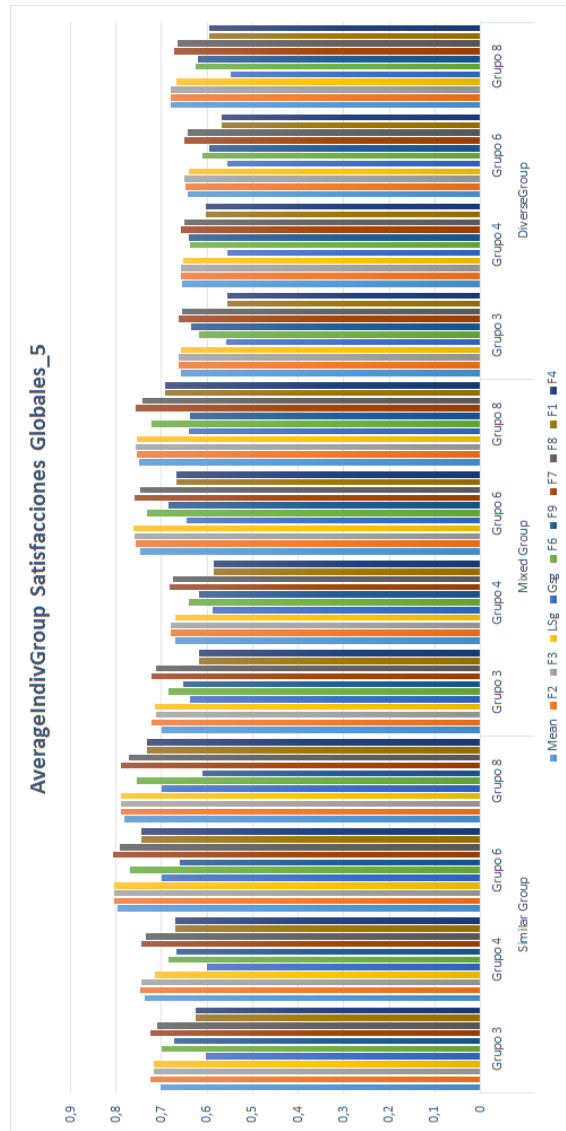


Figura 24: Satisfacción con la modelación AverageIndivGroup

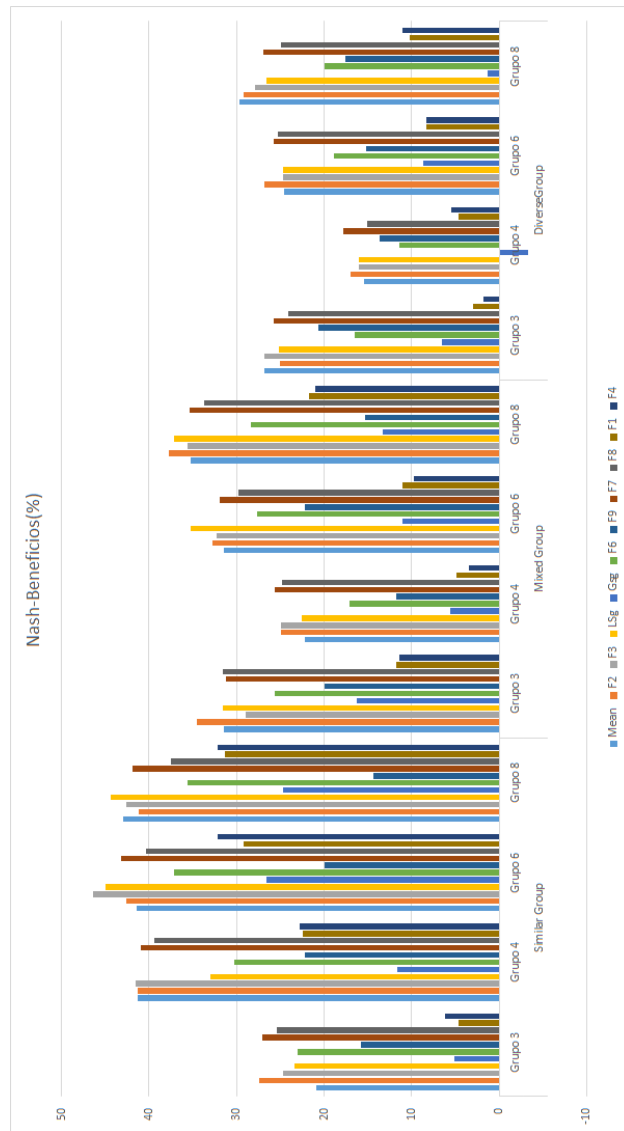


Figura 25: Beneficio obtenido con la modelacion Nash

Vistos los resultados del beneficio de las 5 mejores recomendaciones y del mejor recomendación, se puede concluir que las estrategias que mayor beneficio proporcionan son: mean, purity, completeness, lsg, multiplicative y borda-count.

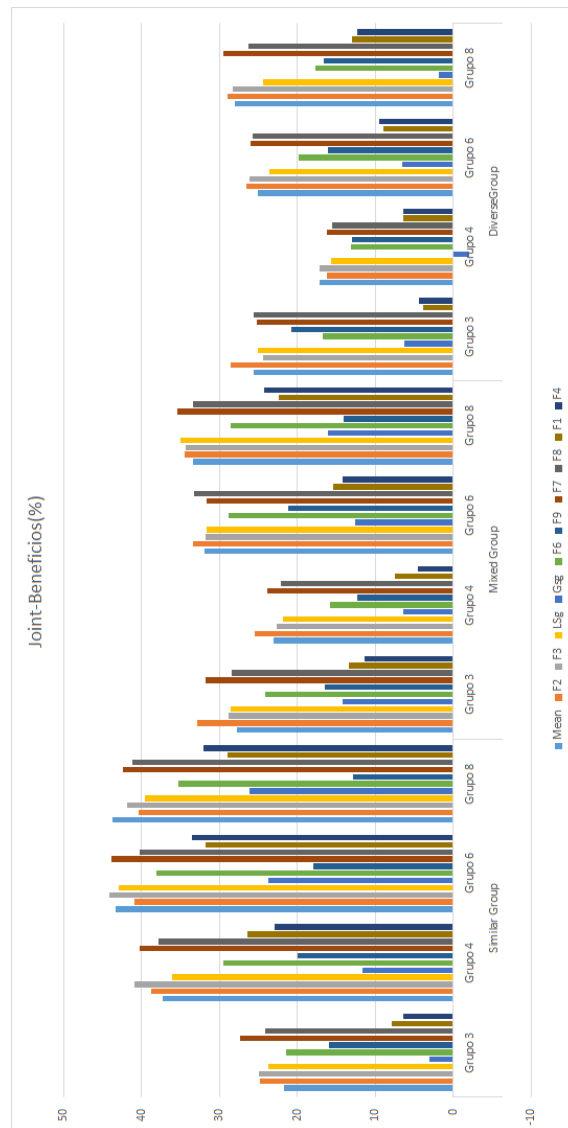


Figura 26: Beneficio obtenido con la modelacion Joint

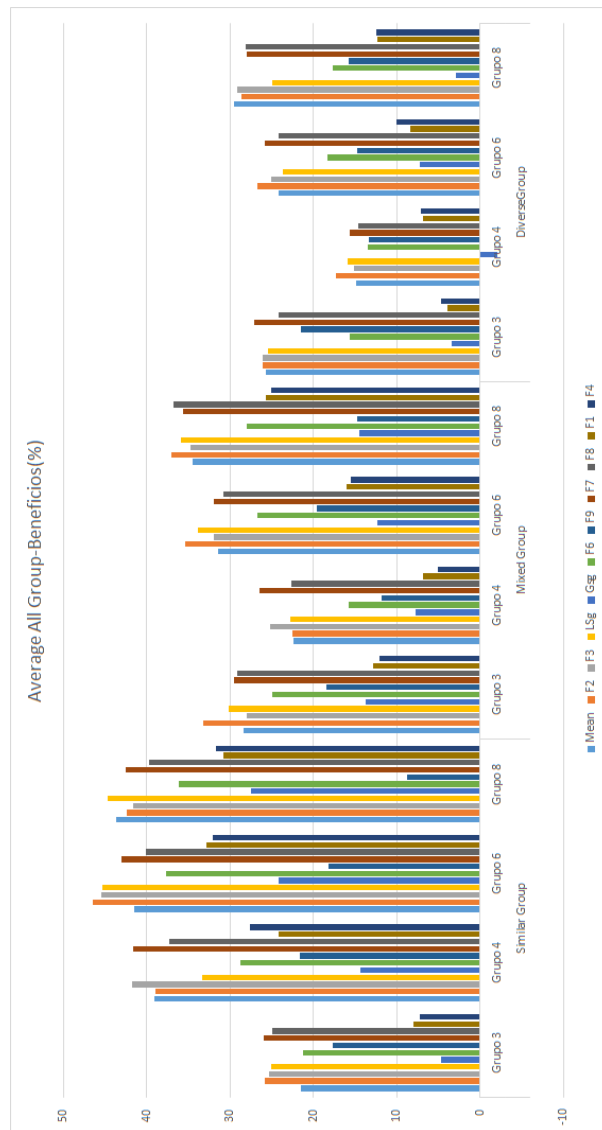


Figura 27: Beneficio obtenido con la modelacion AverageAllGroup

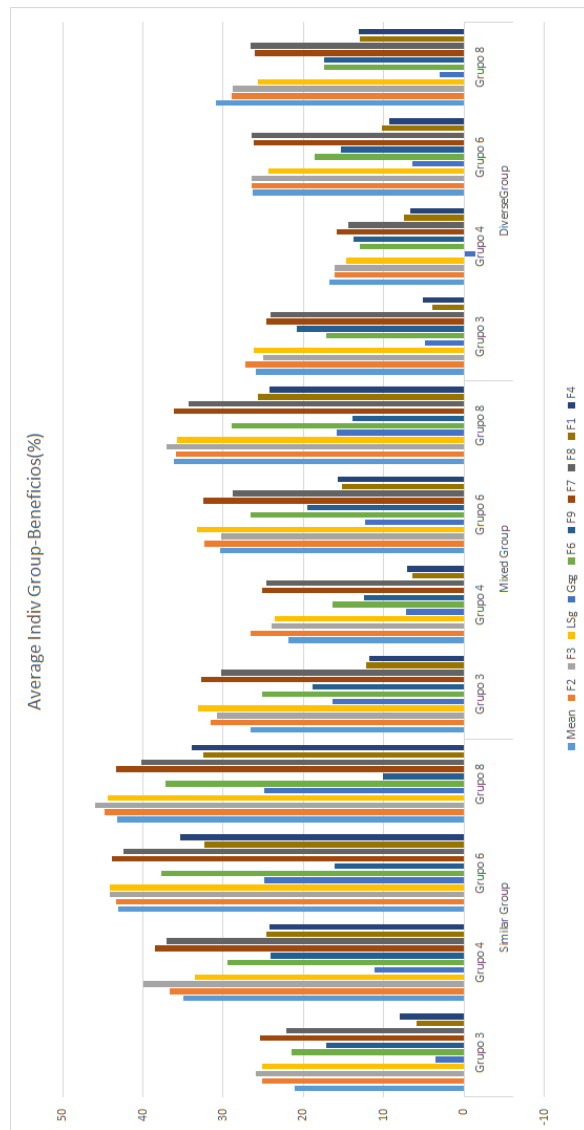


Figura 28: Beneficio obtenido con la modelacion AverageIndivGroup



## 5. Análisis de coste

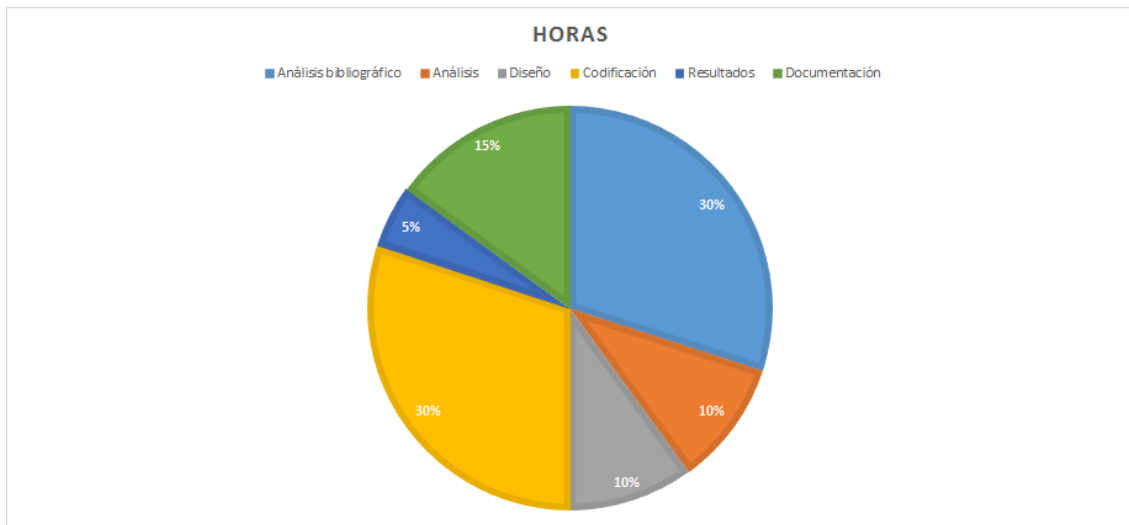


Figura 29: Tabla de dedicación

El punto de partida del proyecto es la introducción al temario, para llegar a dicha finalidad he realizado la lectura de un conjunto de artículos. En los cuáles he comprendido el contexto de los sistemas recomendadores grupales en la actualidad. Además, me introduzco a los problemas que surgen en los sistemas recomendadores y las diferentes estrategias de consenso que se aplican.

En el proceso de análisis, se ha estudiado la estructura de la implementación básica. Las representaciones realizadas para tratar las clases. En el proceso de codificación, se han implementado las estrategias mencionadas y para mejorar la estimación del beneficio se ha realizado la adaptación del recomendador a grupos de diferentes tamaños. Los resultados de las recomendaciones son guardados en un fichero Excel para poder construir la representación gráfica de los datos.

A partir de todas las informaciones y recursos mencionados anteriormente, se inicializa la escritura de esta memoria.

## 6. Conclusiones y líneas futuras

Los trabajos realizados en los sistemas de recomendación grupal implementan estrategias diferentes para la resolución de conflictos en el proceso de recomendación. Pero pocos se han dedicado a estudiar a fondo el beneficio que puede aportar las estrategias. La mayoría de las estrategias están enfocadas específicamente en un dominio concreto, y es beneficioso analizar las diferentes estrategias que se aplican en un dominio para encontrar las mejores estrategias.

En el presente trabajo se trabajó con 12 estrategias diferentes para el proceso de consenso, y para garantizar la fiabilidad de los resultados se han empleado dichas estrategias a grupos de diferentes tamaños y similitudes, además se ha analizado también el resultado con varias recomendaciones a la vez.

Como se pueden observar los resultados de las cuatro configuraciones de modelación de los grupos son muy similares, y en todos ellos se ha cumplido nuestra expectativa de obtener beneficios considerables en las satisfacciones de los usuarios con las estrategias empleadas. En el estudio realizado, los usuarios con preferencias similares son los que mejores se benefician de las estrategias empleadas. Y también se ha cumplido nuestra expectativa de bajada en los niveles de satisfacciones a medida que baje la similitud de los usuarios del grupo.

Además, se puede observar que las estrategias empleadas mejoran los resultados de recomendación comparando con los resultados de recomendaciones aleatorios. Aquí un aspecto muy positivo es testear nuestro sistema con grupos de usuarios reales. Nuestro punto de partida son perfiles de usuarios reales, pero los grupos son generados de forma computacional. Y es muy interesante testear el comportamiento del nuestro sistema con grupo reales. Para ello se requiere la implementación de una interfaz de interacción con los usuarios. Además, si se habilita la comunicación para el proceso de consenso, se obtendrá una línea de comparación real. Con el cual podemos conocer el beneficio real que pueden aportar las estrategias empleadas.

El siguiente paso podría ser la aplicación de la nuestra metodología en otros dominios diferentes al del esquí. Aquí surgirá probablemente el problema de la representación de las preferencias, que nos puede llevar a definir nuevas clases para representarlas. Como todos los recomendadores, si empezamos en un dominio nuevo, la disposición de información es mínimo, y aquí no estaría mal implementar un sistema de captación de informaciones. Captar las informaciones de los usuarios es imprescindible para poder hacerle una recomendación. Cuando más conocemos a un usuario es menos probable que un ítem recomendado sea rechazado. De esta forma, aumentamos nuestra probabilidad de satisfacción. Además, disminuimos el esfuerzo del usuario para obtener una recomendación que le resulta satisfactoria.

## Referencias

- [1] Barry Smyth Anthony Jameson. Recommendation to groups. pages 596–626, 2007.
- [2] Laura Sebastia Eva Onaindia Imma Garcia, Sergio Pajares. Preference elicitation techniques for group recommender systems. pages 1–21, 2011.
- [3] Shriram Krishnamurthi Jacob P.Baskin. Preference aggregation in group recommender systems for committee decision-making. page 4, 2009.
- [4] Jürgen Ziegler Jesús Omar Álvarez Márquez. Preference elicitation and negotiation in a group recommender system. pages 20–37, 2015.
- [5] James Reilly Lorraine McGinty. On the evolution of critiquing recommenders. pages 1–33, 2011.
- [6] Kevin Smyth Barry Maria Salamó, McCarthy. Generating recommendations for consensus negotiation in group personalization services. pages 1–14, 2012.
- [7] Jinyun Yan Stratis Ioannidis, S.Muthukrishnan. A consensus-focused group recommender system. pages 1–5, 2014.