



TRABAJO FIN DE MÁSTER  
MÁSTER UNIVERSITARIO OFICIAL EN CIENCIA DE DATOS  
E INGENIERÍA DE COMPUTADORES

# Sistemas de recomendación de películas para grupos

---

**Autor**

Juan José Sierra González

**Director**

Juan Manuel Fernández Luna



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍAS INFORMÁTICA Y DE  
TELECOMUNICACIÓN

---

Granada, enero de 2020







# Sistemas de recomendación de películas para grupos

---

**Autor**

Juan José Sierra González

**Director**

Juan Manuel Fernández Luna



# **Sistemas de recomendación de películas para grupos**

Juan José Sierra González

**Palabras clave:** incluir palabras clave...

## **Resumen**

Incluir resumen...





# **Group recommender systems for movies**

Juan José Sierra González

**Keywords:** include keywords...

## **Abstract**

Include abstract...



---

Yo, **Juan José Sierra González**, alumno del Máster Universitario Oficial en Ciencia de Datos e Ingeniería de Computadores de la **Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación de la Universidad de Granada**, con DNI 76589592Y, autorizo la ubicación de la siguiente copia de mi Trabajo Fin de Máster en la biblioteca del centro para que pueda ser consultada por las personas que lo deseen.

Fdo: Juan José Sierra González

Granada a 9 de enero de 2020.



---

D. **Juan Manuel Fernández Luna**, Profesor del Área de Uncertainty Treatment in Artificial Intelligence del Departamento de Ciencias de la Computación e Inteligencia Artificial de la Universidad de Granada.

**Informa:**

Que el presente trabajo, titulado ***Sistemas de recomendación de películas a grupos***, ha sido realizado bajo su supervisión por **Juan José Sierra González**, y autoriza la defensa de dicho trabajo ante el tribunal que corresponda.

Y para que conste, expide y firma el presente informe en Granada a 9 de enero de 2020.

**El director:**

**Juan Manuel Fernández Luna**



# Agradecimientos

Incluir agradecimientos...





# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Motivación . . . . .	1
1.2. Objetivos . . . . .	2
<b>2. Planificación</b>	<b>5</b>
2.1. Requisitos de investigación . . . . .	5
2.2. Planificación del trabajo . . . . .	7
2.2.1. Estimación de coste de materiales e infraestructura . .	7
2.2.2. Distribución de tiempo entre tareas . . . . .	8
<b>3. Revisión de la literatura: estado del arte</b>	<b>11</b>
3.1. Análisis de la literatura . . . . .	11
3.1.1. Historia de los sistemas de recomendación . . . . .	11
3.1.2. Recomendación a grupos en la actualidad . . . . .	13
3.2. Propuesta del estado del arte escogida . . . . .	16
<b>4. Nuevas propuestas aportadas al estudio</b>	<b>19</b>
4.1. Método de empatía . . . . .	20
4.2. Método del más cinéfilo . . . . .	23
4.3. Método del más optimista . . . . .	26
4.4. Método de mayor similitud . . . . .	28
<b>Bibliografía</b>	<b>32</b>



# Índice de figuras



# Índice de tablas



# Capítulo 1

## Introducción

En la actualidad, la informática juega un papel fundamental en el día a día de la mayoría de personas. No solo en el ámbito profesional, donde ya lleva más años asentada, sino que cada vez es más común encontrar soluciones informáticas en problemas de la vida cotidiana. Donde antes podían requerirse varios minutos de búsqueda entre páginas de enciclopedias, ahora se resuelven consultas de todo tipo en cuestión de segundos gracias a los grandes volúmenes de información que se almacenan en Internet, y que comparten (y amplían) a diario millones de usuarios.

Pero la información no siempre se presenta de forma clara y concisa, y a menudo se acumulan enormes cantidades de datos aparentemente inservibles. Sin embargo, mediante el manejo de tan grandes cantidades de dichos datos se puede obtener información útil que de otra forma sería prácticamente imposible de detectar. La **ciencia de datos**, encargada de obtener toda esa información compleja, abundante y difusa y transformarla en información útil, está siendo utilizada cada vez con más asiduidad para todo tipo de finalidades. Este estudio se centrará en su aplicación para **recomendar** elementos a los usuarios, o lo que más comúnmente se conoce como **sistemas de recomendación**.

### 1.1. Motivación

Los sistemas de recomendación ya forman parte de la gran mayoría de plataformas web que ofrecen bienes o servicios. Mientras un usuario ojea un elemento de un catálogo en una página web es habitual que encuentre las secciones de “Productos similares” y “Productos recomendados”. Mientras que la primera se basa puramente en características comunes entre los elementos, la segunda es **una recomendación personalizada** para cada usuario.

Debido a que las empresas generan beneficio directamente de realizar una buena recomendación a sus clientes, optimizar el funcionamiento de estos sistemas de recomendación resulta una tarea muy importante. Si bien los sistemas clásicos funcionan bien en términos generales, a cada empresa le interesa que su sistema se comporte de forma correcta en su ámbito, y es en estos detalles donde más interesantes son las mejoras. Y es que para una multinacional, una mejora en sus recomendaciones de un mero 1 % (que puede parecer ínfima) resulta en una inyección de ingresos muy sustancial.

Este estudio analizará uno de los ámbitos más comunes, las **recomendaciones de películas**. Como principal ejemplo, la plataforma de ocio audiovisual **Netflix** es una de las más utilizadas del momento, cuenta con un sistema de recomendación de películas y series muy elaborado, y sin embargo, solo es capaz de recomendar **para un único usuario**. No solo en el ámbito de las películas, sino en casi cualquiera, generalmente las recomendaciones se hacen de forma individual, y hay que buscar un sistema especializado si se quiere recomendar a un grupo de personas. Existen multitud de implementaciones de **sistemas de recomendación a grupos** que tratan de optimizar la elección para satisfacer lo máximo posible a todos los miembros del grupo, o al menos lograr un equilibrio. Este trabajo se esforzará en estudiar las soluciones más actuales a este problema y cómo han mejorado lo que ofrecen las soluciones más clásicas, enmarcadas en el ámbito de las películas.

## 1.2. Objetivos

Como se ha mencionado, ese Trabajo de Fin de Máster pone el foco en los sistemas de recomendación de películas a grupos. Existen múltiples soluciones a este problema, que han ido mejorando con el paso de los años y el progreso logrado mediante investigación. En este estudio se pondrá especial atención en aquellos más recientes, que en esencia serán los que aúnen de forma más eficiente dichos progresos. Una vez estudiados algunos de estos modelos, se tratará de crear una serie de versiones propias que mejoren lo existente.

El plan de ruta del trabajo será partir de un **modelo básico**, o *baseline*, que será el que marque el valor mínimo que debe tener el modelo propio implementado. Se entiende que un modelo peor que un baseline jamás debe ser considerado como bueno, ya que se trata de algoritmos altamente probados y contra numerosas situaciones distintas, y que funcionan bien en términos generales.

Con el modelo básico seleccionado, se procederá a realizar una investigación en la literatura, descubriendo cómo se ha producido la evolución en el



campo de los sistemas de recomendación a grupos, cuáles son las propuestas más innovadoras en los últimos años y, principalmente, cuáles han dado un mejor resultado. Entre estas últimas se seleccionará una entre las más recientes y las que más interesantes y adaptables resulten de cara a futuras modificaciones.

Habiendo elegido un modelo baseline y un modelo actual, el siguiente paso será implementarlos. Se tratará de ajustar lo máximo posible a la información recabada en los *papers* de los algoritmos, para ser fiel a la implementación original de los autores. Por desgracia, ninguna de las versiones más modernas y funcionales es de código abierto y es imposible encontrar una implementación que no haya sido creada a interpretación del *paper*.

Una vez se hayan implementado los dos modelos, se crearán una serie de nuevas soluciones que pongan en práctica lo aprendido durante el estudio en la literatura y las ideas surgidas al mismo tiempo. Se tratará de adaptar el modelo actual en la medida de lo posible, pero utilizando las nuevas técnicas que se crea que puedan suponer un incremento en la eficacia del método.

Por último, con todas las implementaciones hechas, se realizará un estudio experimental donde se pondrán a prueba todos los sistemas de recomendación bajo diferentes circunstancias y parámetros. Se recogerán los datos y posteriormente se evaluará cuál es la mejor solución y si alguna de las soluciones ideadas para este trabajo han logrado mejorar lo existente.



## Capítulo 2

# Planificación

En este capítulo se aborda el plan de trabajo a seguir para la realización de este estudio. En primer lugar se realizará la captación de requisitos necesarios para lograr el objetivo propuesto, y después se detallará la planificación del trabajo, con una estimación en costes tanto de trabajo como de materiales así como una distribución del tiempo entre tareas.

### 2.1. Requisitos de investigación

Al tratarse de un trabajo de investigación, no se puede realizar adecuadamente el análisis de requisitos habitual para un proyecto general de ingeniería. En su lugar, sin embargo, se plantean unos objetivos diferentes que se deben cumplir para llevar a cabo un estudio de estas características. En el caso de este estudio en particular, los requisitos que se pueden plantear son los siguientes:

1. **Realizar una investigación inicial en torno a los sistemas de recomendación:** revisar la literatura buscando información acerca de estos sistemas, comprobar qué evolución han seguido a lo largo de los años y cómo se han llegado a desarrollar los sistemas de recomendación a grupos.
2. **Profundizar en el estado del arte del sistemas de recomendación a grupos:** investigar más a fondo en estos métodos que son en los que realmente se va a centrar el estudio. Obtener los sistemas más destacados del estado del arte.
3. **Seleccionar el algoritmo de referencia del estado del arte:** Entre todos los sistemas de recomendación a grupos que se hayan investigado en el paso anterior, decidir cuál es el que se ajusta mejor a

los requerimientos del estudio y obtener toda la información necesaria para implementarlo.

4. **Diseñar el experimento:** crear los grupos para la experimentación e implementar un algoritmo *baseline* que sirva como punto de referencia. Implementar las estrategias de agregación de soluciones a la hora de experimentar y comprobar su correcto funcionamiento con dicho algoritmo.
5. **Implementar el algoritmo del estado del arte:** implementar el método siguiendo los parámetros y estructura del experimento diseñado. Es necesario contar con el código de los algoritmos para el experimento, por lo que hay que implementar el del estado del arte si no se puede conseguir su código fuente.
6. **Diseñar las nuevas propuestas del estudio:** para el objetivo del estudio hay que plantear nuevas propuestas que puedan competir con el estado del arte. Para ello primero hay que valorar los campos que quedan por experimentar y proponer una serie de algoritmos que se ajusten al perfil buscado.
7. **Implementar las propuestas propias:** una vez decididas las propuestas que van a formar parte del experimento, se deben implementar del mismo modo que el algoritmo del estado del arte. Comprobar de nuevo que toda la funcionalidad del experimento es válida con la nueva implementación.
8. **Obtención de resultados:** obtener a través de la implementación del experimento los resultados del mismo. Se pueden visualizar los datos con tablas y gráficos que sean adecuados para el estudio.
9. **Estudio analítico de los resultados obtenidos:** comparar los resultados de cada método con el método de referencia *baseline* y con el método del estado del arte, así como entre ellos, a fin de descubrir qué algoritmo se comporta mejor para cada caso, tanto entre grupos formados de distinta forma como con estrategias de agregación diferentes.
10. **Extraer conclusiones del estudio realizado:** dar una visión analítica de la eficacia de las propuestas ideadas para el estudio, apoyándose en los resultados obtenidos con respecto a los algoritmos de referencia, y justificar cuáles de ellas son más prometedoras.
11. **Trabajos futuros a realizar:** valorar en qué aspectos se puede seguir progresando en los métodos propuestos en el estudio, así como proponer nuevas tareas o retos dentro de este ámbito de investigación que no se hayan abarcado en el proceso del mismo.

Con los requisitos planteados, la planificación del trabajo debe ocupar los pasos que se han definido y estimar un tiempo con el que se podría solventar cada requisito. Además, debe describir todo el material e infraestructura necesarios y, junto al tiempo estimado, dar una idea del presupuesto que requiere realizar este estudio.

## 2.2. Planificación del trabajo

Este apartado está subdividido en estimación de costes y estimación de tiempo. En un primer lugar se valora el coste de la infraestructura utilizada, así como de todo el material que se haya utilizado y que esté disponible de forma libre y gratuita. A partir de ahí se puede realizar un presupuesto base del que partir. La estimación de tiempo debe ser realista y que permita cumplir los requisitos en un plazo competente, a fin de no necesitar aplazar los tiempos a última hora debido a una mala planificación. De ser así habría que cargar con el consecuente aumento en costes.

### 2.2.1. Estimación de coste de materiales e infraestructura

En primer lugar se tendrá en cuenta el ordenador con la que se ha desarrollado este estudio. Se trata de un ordenador portátil de la marca MSI, del año 2019, y que cuenta con un procesador i7-8750H capaz de llegar a 4.1GHz, así como con 16GB de memoria RAM DDR4-2666, una gráfica GeForce GTX 1060 6GB GDDR5 y dos SSD de 512GB cada uno, uno de los cuales tiene un SO Windows y el otro un SO Linux. Con este equipo se han realizado todas las tareas del experimento: documentación, revisión de la literatura y redacción de la memoria, labores de desarrollo e implementación de los algoritmos y ejecución del experimento para obtención de resultados.

Para obtener la información y documentación requerida para el trabajo ha sido necesaria una red de internet de banda ancha, disponible tanto en el lugar de desarrollo del trabajo como en las instalaciones de la Universidad de Granada cuando ha sido necesario. Gracias a los convenios que la Universidad establece con algunas bases de datos documentales como Scopus [1] ha sido posible acceder a multitud de *papers* y publicaciones sobre el tema a investigar. Además ha sido de mucha utilidad la página ResearchGate [2] para acceder a otros papers no encontrados en Scopus.

El desarrollo del trabajo ha sido realizado completamente sobre el sistema operativo Linux, con una distribución Pop\_OS! [3] basada en Ubuntu 18.10, por lo que es totalmente libre y carente de costes. Las principales herramientas de software utilizadas también han sido de código abierto por lo que no han conllevado coste adicional, utilizando como IDE principal pa-

ra el desarrollo Visual Studio Code [4] y escribiendo la memoria en LaTeX utilizando el software TeXstudio [5], disponible para Ubuntu.

De todo este material expuesto en los párrafos anteriores, el trabajo sólo ha requerido comprar el ordenador portátil personal y la red de banda ancha disponible en el lugar de desarrollo del mismo, dado que se trata de recursos personales. El resto de la infraestructura y materiales o bien han sido productos de software libre o han sido aportados por la Universidad de Granada.

### 2.2.2. Distribución de tiempo entre tareas

Organizar la carga de trabajo en las distintas fases del estudio es una labor que debe plantearse previo a la realización del mismo. Con el planteamiento correctamente definido se pueden decidir algunas fechas de entrega y acotar los tiempos que se le asignan a cada tarea. Gracias a ello, el progreso del proyecto puede seguir el curso planificado, es más sencillo modularizar las tareas y así finalizar en el tiempo estimado.

En primer lugar, la labor de revisión de la literatura ocupará los primeros pasos del trabajo. El estudio debe partir de un punto en el que se conozcan correctamente los detalles de los sistemas de recomendación, con todas sus características. Por supuesto, debe haberse hecho una búsqueda de documentos que tengan dicha información, así como de aquellos *papers* que contengan planteamientos de algoritmos de interés. Se seleccionan las propuestas más interesantes a priori, una vez se ha investigado lo suficiente sobre el tema y se han recabado suficientes ejemplos. La realización de esta tarea puede llevar un mes.

La documentación debe ser un proceso que se lleve de principio a fin, de manera que se vayan tomando notas y documentando adecuadamente conforme se va obteniendo la información requerida. No se puede poner una estimación a priori debido a su carácter cambiante y adaptativo en función del progreso del propio trabajo.

En cuanto al diseño del experimento, es una tarea compleja que lleva una carga de trabajo considerable. Montar todo el entorno, elegir qué estructuras de datos se van a utilizar, cómo automatizar los procesos y cómo almacenar los resultados obtenidos requiere tomar una serie de decisiones cruciales a la hora de definir el proyecto. Cometer un error en este punto supone más tiempo de corrección conforme más avance se haya realizado en el experimento, por lo que es una labor puntillosa a la que merece la pena dedicar tiempo. Al menos se plantean dos semanas o un mes de plazo para terminar toda esta estructura.

Para la implementación del sistema de recomendación del estado del arte

de acuerdo al entorno experimental que se haya montado puede requerir un poco de tiempo, pero al fin y al cabo el algoritmo está definido y no debería suponer más de dos semanas seguir los pasos indicados en la publicación. En algunos casos puede haber determinadas partes que no estén bien especificadas y requiera o bien más labor de investigación o bien tomar alguna decisión propia sobre la implementación (propiamente documentada en su correspondiente apartado).

Con respecto a los nuevos métodos de recomendación para el estudio, cabe destacar que no requiere únicamente labor de implementación sino que también hay que detenerse a analizar el estado del arte y pensar en qué vertientes pueden aportarse al ámbito tratado. Aunque la mayor parte de esta tarea se pueda hacer en el apartado de revisión de la literatura, es en este punto, ya habiendo implementado el algoritmo actual, cuando es más sencillo darse cuenta de aquello que se ha echado en falta durante la investigación e implementación. Como además no se ha limitado desde el primer momento el número de nuevas propuestas que puedan surgir, se ha planteado dejar un mes para todo el proceso de análisis e implementación de dichas propuestas.

Para la ejecución del experimento y la obtención de datos se pueden utilizar un par de días o incluso una semana, a fin de recopilar en tablas los valores obtenidos y conseguir visualizar los datos de manera óptima para que reflejen los objetivos deseados. De esta forma se podrá comprobar de un vistazo sencillo el impacto de cada sistema de recomendación frente al problema planteado en cada caso.

Finalmente, extraer conclusiones de los resultados puede ser más sencillo si se conocen los algoritmos y se han trabajado correctamente, de forma que sea fácil interpretarlos y descubrir patrones en los mismos. Al ser una tarea tan diversa y poco acotada, una extracción de conclusiones puede llevar desde unos días hasta una semana.

En resumen, este trabajo de investigación puede enmarcarse en el tiempo en unos 4 meses, aunque naturalmente esa estimación puede variar en función del tiempo que se le pueda dedicar diariamente al proyecto, especialmente si se compagina con un trabajo a jornada completa. Es de importancia recordar que el tiempo invertido en el experimento también tiene un coste, igual que el de los materiales e infraestructura; en esencia se trata de otro recurso. Partiendo de la base de que este estudio es un trabajo de investigación en un campo puntero tecnológicamente, podría estimarse el coste total del trabajo como una media entre el sueldo a jornada completa de un ingeniero informático junior y el de un investigador de un departamento de universidad durante el transcurso del mismo.

Estimando para una profesión de estas características un valor aproximado de 12 euros/hora, a 4 meses trabajando (20 días al mes) y dedicándole

una media de 4h diarias, se estipularía el presupuesto del proyecto en 2880 euros.



## Capítulo 3

# Revisión de la literatura: estado del arte

En este capítulo se expondrá una revisión de los estudios y propuestas más innovadoras relacionadas con los **sistemas de recomendación a grupos**, y en particular, de aquellos orientados al ámbito de las películas. Se comenzará haciendo un breve inciso sobre cuándo y cómo se originaron los sistemas de recomendación, así como la particularidad de aquellos específicos de recomendación a grupos. Después se tratará su evolución hasta nuestros días, realizando un comentario analítico acerca de lo que han supuesto los sistemas de recomendación y cómo se trasladan actualmente dichos sistemas a situaciones cotidianas, con algunos claros ejemplos que todo el mundo utiliza. Entre ellos se destacarán los que abarcan la **recomendación de películas**, como tema principal del estudio. Por último, se mencionarán algunos de los sistemas de recomendación de películas a grupos más actuales, eligiendo además el que será tomado como referencia para la experimentación como representante del estado del arte de estos recomendadores.

### 3.1. Análisis de la literatura

#### 3.1.1. Historia de los sistemas de recomendación

Queda ya muy atrás el primer sistema de recomendación de la historia, que vio la luz en 1992, cuando Goldberg et al. (1992) presentaron su artículo *“Using collaborative filtering to Weave an Information tapestry”* [6]. En este trabajo se acuña por primera vez el término **“filtrado colaborativo”** (en inglés, *collaborative filtering*), refiriéndose así a la técnica que permite filtrar elementos a un usuario basándose en lo que otros usuarios han opinado de él

anteriormente. Esta técnica sienta las bases de los sistemas de recomendación, y aún a día de hoy se encuentra presente en la mayoría de ellos. La otra alternativa al filtrado colaborativo es el **filtrado basado en contenido** (en inglés, *content-based filtering*). Esta técnica no tiene en cuenta las valoraciones que otros usuarios han dado a los elementos del sistema, sino que se limita a comparar los elementos por características y recomendar aquellos similares. En este experimento, sin embargo, se tratarán únicamente métodos que utilizan el filtrado colaborativo.

Una vez se hubo contemplado el vasto rango de opciones que albergaban los sistemas de recomendación, a través de numerosos trabajos y estudios, se contempló la posibilidad de recomendar ya no solo a individuos, sino a grupos de personas. Se trataba de una situación a la que una sociedad podía enfrentarse en multitud de facetas, y que sin duda merecía la pena explorar. La pionera, y una de las más famosas y reconocidas propuestas sobre este tema, fue **PolyLens** [7]. En esta publicación de 2001, O'Connor et al. plantearon **un sistema de recomendación de películas para grupos**, derivado de la base de datos de valoraciones de películas **MovieLens**, en la que varios trabajos anteriores se habían fundado. Este sistema abrió la veda de un nuevo subgrupo dentro de la ya amplia categoría de sistemas de recomendación.

Como se ha comentado anteriormente, las recomendaciones a películas han levantado mucho interés desde el primer momento. El contenido de ocio audiovisual, junto a los catálogos de tiendas *online*, son los casos de mayor demanda en cuanto a sistemas de recomendación en la sociedad actual. Plataformas como **Netflix**, **Spotify** o **HBO** son ejemplos de los primeros, que utilizan estos sistemas para tratar de mantener al usuario satisfecho ofreciéndole nuevo contenido que sea similar al que hayan visto. Por su parte, **Amazon** o **AliExpress** son ejemplos de los segundos, grandes plataformas que dan cabida a todo tipo de productos y que son capaces de recomendar a sus clientes otros productos similares a los que están buscando. Gracias al sistema de recomendación pertinente, incluso consiguen ofrecer nuevos artículos que detectan que el usuario puede querer comprar, basándose en lo que anteriormente ha buscado o comprado.

Tal fue el impacto de los sistemas de recomendación que la propia plataforma Netflix lanzó en 2006 lo que se conocería como el **Netflix Prize** [8], una competición de sistemas que, basándose únicamente en el filtrado colaborativo (no conociendo datos adicionales sobre películas ni sobre usuarios, solamente las valoraciones que estos habían dado a cada película) fueran capaces de predecir acertadamente dichas valoraciones. El objetivo principal era superar al por aquel entonces actual sistema de recomendación de Netflix, conocido como **Cinematch**, ofreciendo un premio de 1 millón de dólares a aquel sistema capaz de superarlo en un 10 %. Adicionalmente,

como la competición se estimaba que duraría hasta 2011, se concedería un premio anual de 50.000 dólares al método ganador al final de cada año.

Tras varios tira y afloja entre algunos equipos, e incluso agrupaciones entre equipos para forjar mejores ideas juntos, finalmente en 2009 el equipo *'BellKor's Pragmatic Chaos'* se alzó ganador y recibió el gran premio [9], superando al algoritmo anterior por 10.05 %. De esta forma tan original Netflix cambió su sistema de recomendación promoviendo una iniciativa de investigación en este campo, y por supuesto previendo que este avance en su algoritmo les serviría para incrementar la satisfacción de sus clientes, y por ende, para aumentar su valor de mercado.

### 3.1.2. Recomendación a grupos en la actualidad

Tras haber sido puestos en el panorama de investigación con casos como el Netflix Prize y el evidente aumento de su uso conforme las nuevas tecnologías se han vuelto más accesibles para la sociedad, los sistemas de recomendación han contemplado un gran abanico de variedades: algunos más enfocados a la materia que recomendar (por ejemplo, sistemas de recomendación de música o de películas), otros enfocados al tipo de usuario (sistemas de recomendación individual o a grupos) e incluso algunos cuyo propósito no es solamente obtener el mejor resultado (sistemas de recomendación explicativos que sirvan de apoyo a una decisión).

Como el objetivo de este trabajo es orientarse en recomendación a películas para grupos de usuarios, la búsqueda en la literatura de algoritmos que apoyasen el estudio se focalizó en estos dos grandes campos, analizando algunas propuestas de la última década. La recomendación a grupos ha avanzado a muchos niveles desde la presentación de PolyLens, y ahora existen varios frentes abiertos en investigación sobre este tema.

Uno de las primeras fuentes que se recopilaron sobre los sistemas de recomendación a grupos, y que han servido a este estudio para obtener información general sobre los mismos, es el capítulo de Masthoff en el *Recommender Systems Handbook* de 2011 [10], donde hace un repaso a los campos de investigación abiertos de los que se hablaba en el párrafo anterior, y realiza un análisis sobre los pasos a seguir en adelante. Masthoff sienta en primer lugar los motivos por los que es necesario contar con sistemas de recomendación a grupos frente a la recomendación individual que ya se había trabajado más de una década. Aporta una detallada recopilación de las estrategias de agregación que utilizaban los sistemas de recomendación a grupos más interesantes de la época, como el ya mencionado PolyLens [7], el sistema MusicFX [11] que determina la emisora de radio que suena de fondo en un gimnasio en función de los gustos de la gente que haya ejercitándose en ese momento, o el sistema Intrigue [12] que ayuda a grupos de turistas a deci-

dir qué visitar, entre otros. A continuación explica cómo se pueden aplicar los algoritmos de recomendación a grupos para usuarios individuales, dando a entender que estos algoritmos son reutilizables dependiendo del contexto en el que se necesiten. Termina planteando una serie de retos a los que los sistemas de recomendación a grupos se enfrentaban en aquel momento: aportar explicaciones a las recomendaciones propuestas, profundizar en el “problema del inicio frío” para grupos de usuarios o recomendar ítems en un determinado orden.

En el contexto de las recomendaciones ordenadas podrían englobarse los **rankings**, que es hacia donde han evolucionado los recomendadores individuales en las principales páginas de Internet. Es habitual no limitarse a recomendar un único ítem a un usuario, lo que propicia que se investigue en recomendar series de elementos dándoles un determinado orden (por supuesto, se quiere colocar en primer lugar aquel que se prediga que va a tener más aceptación).

Una publicación parecida a la de Masthoff en términos de recopilación, pero más reciente, fue la que realizaron Dara et al. en 2019 [13]. En este trabajo, titulado *A survey on group recommender systems*, ofrecen una nueva perspectiva más actual del estado del arte de los sistemas de recomendación a grupos. Una de las distinciones que hace en cuanto a estos recomendadores es acorde al ámbito en el que se desarrollan: películas, música, viajes o programas/cadenas de televisión. De acuerdo al tema de este estudio, tienen más interés aquellos relacionados con las películas. Además del ya conocido PolyLens, que ya era el principal recomendador de películas a grupos cuando Masthoff et al. desarrollaron su estudio, añade otros diez recomendadores más actuales, de la década de los 2010s. Entre ellos, aparecen dos publicaciones de 2017, las más recientes de la lista, y que podrían entrar en el debate del estado del arte de estos sistemas.

Una de ellas es **OrderedRec**, de Agarwal et al. [14], donde plantean solución a uno de los retos que planteaba Masthoff en su publicación de 2011: las recomendaciones ordenadas. Partiendo de un determinado orden de preferencia de cada usuario, introducen la satisfacción del mismo como atributo, que es calculado en función de la diferencia entre sus preferencias y el orden calculado por el recomendador.

La otra es **Natural Noise Management (NNM)** de Castro et al. [15]. En esta publicación se analiza el caso en el que el usuario aporte sus preferencias explícitamente pero puedan encontrarse inconsistencias debido a lo que se conoce como **ruido natural**. El estudio consiste en tratar de modelar el ruido natural de forma que sea capaz de detectarse sin afectar al rendimiento general del algoritmo de por sí.

Una última aportación que se menciona de 2017 es el caso del recomendador **HappyMovie** [16]. La primera publicación de este sistema es en 2011,

cuando Quijano-Sánchez et al. idean una aplicación capaz de recomendar películas a grupos de usuarios en la red social Facebook. En primera instancia, mediante un test de personalidad la aplicación pide a los usuarios que la van a utilizar que valoren una serie de películas, para así obtener información del usuario y solventar en cierta manera el conocido problema del “inicio frío”. A continuación, a través de los datos de Facebook obtiene la “confianza” que tienen unos usuarios con respecto a otros, para detectar cómo influye cada uno en su recomendación particular. Además, utiliza memoria en el tiempo para mejorar las recomendaciones, no recomendar las mismas películas, y sobre todo, introduce el concepto de “fairness”, para tratar de hacer recomendaciones justas y que todo el mundo esté contento.

Este sistema alcanza una nueva cota en 2017, cuando para el artículo “*Make it personal: A social explanation system applied to group recommendations*” [17] Quijano-Sánchez et al. añaden una nueva dimensión de **explicación en las recomendaciones**. Mediante esta mejora, que a primera vista puede resultar menos influyente de lo que realmente es, consiguen darle a los usuarios una sensación de confianza. Un usuario puede necesitar una explicación cuando se le va a recomendar una película que no le va a gustar pero sin embargo es necesaria para el grupo en un determinado momento, bien sea porque dicho usuario ha visto una serie de películas seguidas que le han gustado más que al grupo o por algún otro motivo distinto. También se especifica la posibilidad de que en grupos de más de cuatro miembros solo se dará una explicación detallada sobre la influencia de los tres más cercanos al usuario que pide la explicación, y sobre el resto se hace un resumen. Posibilitar esto mejora la confianza del usuario en el sistema, y por ende, se ve más propenso a depositar sus elecciones futuras en manos del recomendador.

Una vez situados los sistemas de recomendación a grupos en el panorama de investigación, habiendo comprendido su evolución y su progreso en el tiempo y dónde se encuentran ahora mismo, el objetivo del trabajo era encontrar una propuesta actual (a ser posible, de 2019) que hiciera de referencia a batir para el estudio. El principal requisito que se le pedía a dicha propuesta era que representase bien el estado del arte planteando una idea innovadora y que resultase intuitiva desde el punto de vista algorítmico. Debido a la naturaleza de las nuevas aportaciones que se plantean en este estudio, que buscan automatizar mediante su algoritmo algunas tomas de decisiones que podrían llevarse a cabo tras consenso pactado entre los propios miembros del grupo, era especialmente interesante encontrar una publicación del estado del arte que aportase esta misma condición.

Buscando propuestas de los dos últimos años para el estudio se encontraron una serie de ellas que, pese a pertenecer al ámbito de las películas, no resultaron adecuadas para el estudio por no contar con el factor anteriormente mencionado de “algoritmo comprensible”. Presentado en 2019,

el método “*Evolutionary learning approach to multi-agent negotiation for group recommender systems*” de Choudhary y Bharadwaj [18] plantea una especie de algoritmo genético que trabaja con multiagentes dentro del grupo. Mediante funciones de mutación y cruce se van generando nuevas soluciones que pueden o no mejorar a lo que existía anteriormente, como si de una metaheurística se tratase. En este método en particular, además se incluyen unas técnicas de negociación entre soluciones para llegar a un punto intermedio, lo que al fin y al cabo sería un pseudo-operador de cruce. Dada la necesidad de incluir algoritmos genéticos en el desarrollo de la solución, esta se distancia del perfil que se andaba buscando para el método de referencia del estado del arte. Lo mismo ocurre con la publicación “*Recommender system with grey wolf optimizer and FCM*” de Katarya y Verma, que a pesar de ver la luz por primera vez a través de Internet en 2016, fue publicado en la revista *Neural Computing and Applications* en septiembre de 2018 [19]. Este método sigue una estructura similar al de Choudhary y Bharadwaj, pero en lugar de utilizar algoritmos genéticos utiliza en su lugar algoritmos bioinspirados. En particular, mediante el *Grey Wolf Algorithm*, uno de los bioinspirados más conocidos, busca “acorralar” a través de una serie de iteraciones la mejor solución posible dada la población (miembros del grupo) de la que dispone.

### 3.2. Propuesta del estado del arte escogida

Después de haber buscado de forma extensiva a lo largo de la historia de los sistemas de recomendación a grupos, y tras haber indagado en las publicaciones de los últimos años en particular, una idea llamaba la atención sobre el resto; habiendo realizado el autor de este trabajo un estudio anterior acerca de las **metaheurísticas socioinspiradas** (basando su algoritmo en comportamientos propios de la sociedad humana), resultaba interesante plantear por qué no existían apenas soluciones que trataran de llevar al algoritmo el mismo comportamiento que los propios humanos llevan a cabo a la hora de sus tomas de decisiones. Ya que en general los humanos adaptan sus métodos a lo que comprueban que funciona mejor o peor, se plantea en este estudio un abanico de posibilidades partiendo de este antecedente. Y es por dicha razón que la propuesta del estado del arte escogido debía de tener alguna relación con este tipo de metodología.

Una vez estudiados los algoritmos más recientes, existe uno de ellos que se adapta a este esquema mejor que ningún otro. Publicado en 2018, pero presentado en se trata del método “*Preference-Oriented Group Recommender System*” de Choudhary y Bharadwaj [20]. En palabras de los propios autores, “el principal objetivo de un sistema de recomendación a grupos es identificar las preferencias de cada usuario y encontrar un punto de conce-

sión en el que todos los usuarios se encuentren igualmente de acuerdo”. Este objetivo normalmente supone un reto muy complejo de conseguir, dadas las potenciales preferencias enfrentadas que puedan ocurrir entre los miembros del grupo y que den lugar a puntos intermedios nada satisfactorios para el resto de los usuarios.

Tradicionalmente se ha optado por generar un “perfil de usuario” como resultado de la agregación de las distintas preferencias de los usuarios, o bien se han ido agregando recomendaciones hechas a miembros individuales del grupo para comprobar cuál es la más exitosa. En este trabajo, Choudhary y Bharadwaj proponen una solución, el **recomendador POGRS**, que se acerca más a la primera opción, y que busca obtener un perfil de usuario que simbolizará al grupo y al que se le realizará la recomendación “individual”. Lo que distingue este caso de otros es que, para este método, el perfil del grupo no es generado mediante agregación de preferencias, sino que es **un usuario propiamente del grupo**. Este pequeño cambio de concepto acerca más el algoritmo a un prototipo socioinspirado, donde se puede prescindir de tener un sujeto abstracto y simplemente trabajar con los propios usuarios.

Para escoger quién es el usuario que más representa al grupo, el método calcular una nueva variable que denomina “estatus del usuario” y que mide el grado de representación de cada uno con respecto al grupo. Para calcularlo, en primer lugar se debe conocer los tres elementos preferidos de cada miembro del grupo, a saber: primero, el más preferido (MP); segundo, el preferido (P); y tercero, el menos preferido (LP). El estatus del usuario consiste en calcular cuántos otros usuarios apoyan su primera elección, su segunda y su tercera. Como es lógico, teniendo más influencia en el resultado final que dos coincidan en el primer elemento a que coincidan en el tercero.

El cálculo final del estatus de un usuario consiste en la sumatoria de los apoyos al primer, segundo y tercer elemento de su lista de preferencias por parte del resto de miembros del grupo. Una vez obtenido el estatus se puede saber qué usuario es el más representativo del grupo y por ende, hacer como recomendación grupal la **lista de preferencias de dicho usuario**. De cara a interpretarlo como carácter social, podría decirse que este algoritmo se asemeja a una votación popular, en la que los miembros del grupo aportan su lista de elementos preferida y en base a una cierta mayoría a lo largo de dicha lista se decide cuál está más cerca del hipotético perfil agregado de todos los usuarios.





## Capítulo 4

# Nuevas propuestas aportadas al estudio

En este capítulo se realizará un análisis sobre los distintos métodos ideados para competir contra el sistema de recomendación del estado del arte que se ha escogido como referencia. La idea es proponer una serie de alternativas que puedan resultar interesantes y que se espera que mejoren la eficacia de dicho método.

Como se ha mencionado en el capítulo anterior, el principal criterio que se ha seguido para escoger el método del estado del arte es, aparte de la intrínseca necesidad de tratarse de un algoritmo actual, que reproduzca en su algoritmo un comportamiento social que se asemeje a la toma de decisiones que realizan los seres humanos a la hora de enfrentarse a un problema así. En el caso del sistema de recomendación de referencia, su algoritmo se basa en decidir qué usuario es más representativo para el grupo en función de la lista de preferencias que tiene cada miembro. Para los métodos propios de este estudio se han planteado las siguientes propuestas:

- **Método de empatía**
- **Método del más cinéfilo**
- **Método del más optimista**
- **Método de mayor similitud**

A lo largo de las próximas páginas se realizará un análisis más detallado acerca de cada uno de estos métodos, así como de dónde ha surgido la idea de cada uno y cómo puede asemejarse a un comportamiento social. Además se explicará su funcionamiento mediante pseudocódigo acompañado de una descripción del mismo en lenguaje natural.

## 4.1. Método de empatía

La idea para esta primera propuesta surgió desde el inicio del estudio, ya que fue un factor que no se identificó en prácticamente ningún sistema de recomendación a grupos del estado del arte. En el recomendador *HappyMovie* [16] aparecía el concepto de *fairness*, algo así como la “justicia” a la hora de realizar recomendaciones al grupo. Sin embargo, en dicha publicación no se hace referencia al método que utilizan para medir, estimar y recalculan el valor de *fairness* a lo largo del tiempo. Dada la naturaleza del concepto, este valor tiene necesariamente que ser dinámico y actualizarse en función de lo que haya ocurrido en iteraciones pasadas. En este punto se decidió que una de las nuevas propuestas del estudio fuera **un sistema de recomendación a grupos capaz de ser empático** con todos los miembros del mismo.

La premisa para este algoritmo es sencilla, pero lógica: dar más importancia a las preferencias de aquellos usuarios a los que menos les hayan gustado las últimas películas recomendadas. A un usuario al que no le hayan gustado nada las últimas tres películas recomendadas se le tendrá muy en cuenta a la hora de decidir la siguiente, de la misma manera que a una persona a la que le han maravillado esas mismas películas no le importará que la siguiente le guste un poco menos.

Si se habla de un grupo de amigos que van a ver una película, este algoritmo tiene especial sentido, ya que se entiende que todos los miembros buscan el interés general del grupo además de su propio beneficio. Estar dispuesto a hacer concesiones, o “sacrificar” una elección óptima para que otra parte del grupo obtenga una opción mejor hace que el grupo se equilibre y puede dar lugar a una recomendación interesante desde el punto de vista algorítmico – a través de la experimentación se dictaminará si también lo es desde el punto de vista de la eficacia.

A continuación se muestra un fragmento de pseudocódigo que representa el método de recomendación basado en empatía:

**Algorithm 1** Método de empatía

---

```

1:  $valoraciones \leftarrow predecirValoraciones()$ 
2: for  $usuario$  in  $grupo$  do
3:   Añadir  $\frac{1}{size(grupo)}$  a  $pesos$ 
4:  $ranking \leftarrow []$ 
5: while  $size(películas) > 1$  do
6:    $valoracionesPonderadas \leftarrow valoraciones \times pesos$ 
7:    $películaRecomendada \leftarrow max(valoracionesPonderadas)$ 
8:    $pesos \leftarrow actualizarPesos()$ 
9:   Eliminar  $películaRecomendada$  de  $valoraciones$  y  $películas$ 
10:  Añadir  $películaRecomendada$  a  $ranking$ 
11: Añadir última  $película$  a  $ranking$ 
12: return  $ranking$ 

```

---

**Algorithm 2** Actualización de pesos para el método de empatía

---

```

1:  $rango \leftarrow 0.2$ 
2:  $paso \leftarrow \frac{rango}{size(grupo)-1}$ 
3:  $ordenSatisfacción \leftarrow \textbf{Ordenar } valoraciones[películaRecomendada]$ 
4: for  $posición$  in  $ordenSatisfacción$  do
5:    $pesos[posición] \leftarrow -\frac{rango}{2} + paso \times posición$ 
6: return  $pesos$ 

```

---

En el pseudocódigo se pueden observar dos algoritmos distinguidos: el primero representa el grueso del método, con todo el proceso del sistema de recomendación y que devuelve el ranking final; el segundo es el algoritmo que actualiza los pesos para cada miembro del grupo en función de cuánto les ha gustado la película. En las siguientes líneas se entenderán qué son estos pesos y cómo afectan al criterio de selección diseñado.

En primer lugar, se analizará el algoritmo general del sistema de recomendación. Este método tiene una clara diferencia que destaca sobre el algoritmo del estado del arte, y es la presencia de **pesos** que ponderan la relevancia de cada miembro del grupo, entendiendo por relevancia lo mucho que puede influir un miembro en la toma de una decisión determinada en el transcurso de la construcción del ranking final. Es decir, estos pesos tienen que ser **dinámicos** para adaptarse a las circunstancias que se vayan aprendiendo de las iteraciones anteriores.

La inicialización de valores que se da al comienzo del algoritmo conlleva la creación de un vector de pesos que, por el momento, debe valorar a todos los usuarios por igual. Se entiende que se parte desde un punto base en el que ningún usuario está más satisfecho que otro, por lo tanto todos tienen la misma importancia a la hora de tomar una decisión sobre qué película

deben escoger. Para lograr esta funcionalidad, a cada usuario de un grupo  $G$  debe serle asignado un valor equivalente a  $\frac{1}{|G|}$ , siendo  $|G|$  la cardinalidad de dicho grupo, y así conseguir que todos aporten el mismo porcentaje a la decisión final.

Para poder comenzar a recomendar cualquier película a un grupo, es necesario que se cuente con una estimación de lo que cada miembro va a valorar dicha película. Esto se consigue mediante **predicción** – el método utilizado para ello se explicará en detalle en el apartado correspondiente a experimentación, dado que se puede sustituir fácilmente en función del estudio que se quiera realizar, y qué algoritmo se elija para predecir estas valoraciones no implica cambio alguno en cuanto a cómo se desarrolla el método.

Con la matriz de valoraciones obtenida (películas x miembros del grupo) y los pesos inicializados, se comienza un bucle del que obtener en cada iteración la película seleccionada para ser incluida en el ranking. Para ello, lo primero que hay que hacer es computar los pesos para cada usuario, dándoles a la valoración de cada uno un coeficiente de modificación que se corresponda a la importancia que han de tener en la decisión pertinente. Una vez obtenidas las valoraciones ponderadas, sumando dichas valoraciones para cada película se calcula cuál obtiene un resultado mayor, y por tanto **es la película recomendada**. El último paso de este bucle es añadir la película al ranking final (por supuesto, ordenado), eliminarla de la lista de películas y de la matriz de valoraciones, y **recalcular los pesos** asignados a cada usuario.

Al tratarse de un método un poco más complejo, se ha decidido separar este algoritmo del pseudocódigo del sistema de recomendación, y en cierto modo también realizar un paréntesis en la explicación del mismo para analizar cómo se actualizan los pesos de los usuarios en este método en particular. Como se ha indicado anteriormente, la funcionalidad principal que se persigue con este método es que los usuarios queden **satisfechos de forma empática**, es decir, se busca propiciar que los usuarios que no han visto películas que les gustan tengan más voz a la hora de escoger la siguiente. En este algoritmo se implementa una funcionalidad similar a la del algoritmo AdaBoost [21], disminuyendo el peso de los miembros del grupo satisfechos y aumentando el de aquellos a los que la película les gusta menos. En este caso, los valores no pueden aumentar por encima de 1 y no pueden disminuir por debajo de 0 – esto tiene especial sentido viéndolo desde el punto de vista del lenguaje natural como que nunca se debe “fastidiar” a un miembro del grupo por equilibrar la balanza, es decir, no se quiere dar una influencia negativa a tal usuario. Como mucho, si se da el caso de que dicho usuario está encantado con todas las películas, puede acabar con peso 0 significando esto que no tomará parte en la decisión de la próxima película.

El algoritmo de actualización de pesos empático funciona de la siguiente manera. En primer lugar se ha de fijar un valor para el rango de modificación de los pesos, que englobe por encima y por debajo del 0. En este caso se ha escogido el valor de rango 0.2 tras una serie de pruebas manuales previas al experimento. Este rango provoca que la fluctuación de pesos pueda ir desde un  $-0.1$  hasta un  $+0.1$ . Como la idea es que los miembros se distribuyan en ese espacio, se ha de calcular un *paso* que especifique el escalón que ha de darse entre cada usuario.

Por ejemplo, para poder dar cabida a los 5 usuarios de un grupo  $G$  debe existir un paso equivalente a  $\frac{r}{|G|-1}$ , siendo  $r$  el rango previamente definido y  $|G|$  la cardinalidad del grupo. De esta forma, el valor del paso sería  $p = 0.05$  y los usuarios se distribuirán con los valores  $\{-0.1, -0.05, 0, 0.05, 0.1\}$  en función de su satisfacción con la película seleccionada.

A continuación se calcula, en función de la **valoración original** predicha para cada usuario, no de la valoración ponderada obtenida de multiplicar la predicción por los pesos, el orden de satisfacción de los usuarios con dicha película. Este orden será el que indique qué fluctuación de pesos se le otorga a los usuarios, y así el que más satisfecho haya quedado tendrá un 10 % menos de influencia en la siguiente película, así como el que menos haya quedado tendrá un 10 % más.

Por último, como se ha especificado anteriormente, es necesario limitar los valores obtenidos tras la fluctuación al rango  $[0, 1]$  para que la recomendación tenga sentido desde el punto de vista social en todo momento, que en última instancia es lo que se busca con este experimento; no solo obtener un resultado eficaz, sino comprobar cómo es capaz de comportarse un algoritmo que actúe de manera similar a como lo haría un ser humano.

Una vez entendido cómo funciona la actualización de pesos para el método de empatía, solo queda repetir el bucle hasta que todas las películas hayan sido clasificadas y ordenadas, y devolver el ranking correspondiente. De acuerdo al propósito del método, dicho ranking habrá tenido en cuenta mantener, a lo largo de todas las decisiones, un equilibrio que intente causar la mayor satisfacción global, pero sobre todo que logre que ningún usuario se sienta maltratado por el sistema.

## 4.2. Método del más cinéfilo

El siguiente método sugiere una implementación de la idea de que es más fiable la opinión de alguien que haya visto una gran cantidad de películas. En una colección de actitudes empleadas por los seres humanos a la hora de tomar decisiones no puede faltar una de estas características, pues está claro que de forma habitual, y a veces de forma inconsciente, se da más valor a la

opinión de alguien meramente por ser el más experimentado. Podría decirse que, en general, una persona que ha visto cientos de películas será capaz de aportar criterio a sus decisiones y de escoger buenas películas casi por inercia. Si bien basarse en esta premisa puede ser racional, también existe la opinión de que una persona no tiene por qué representar más al grupo simplemente por haber visto más películas. Los resultados del experimento esclarecerán esta respuesta.

En este método vuelven a aparecer los pesos, que indican qué relevancia tiene un miembro del grupo a la hora de tomar decisiones. La idea principal de este método no es más que contabilizar la cantidad de películas que un usuario ha visualizado (refiriéndose a aquellas películas del conjunto de entrenamiento, claro) y, en función del total de películas que se hayan visto en el grupo, darle un mayor o menor peso a la hora de decidir qué película seleccionar.

A continuación se expone en un fragmento de pseudocódigo el funcionamiento del método en cuestión:

---

**Algorithm 3** Método del más cinéfilo
 

---

```

1: valoraciones  $\leftarrow$  predecirValoraciones()
2: pesos  $\leftarrow$  obtenerPesosCinéfilos()
3: ranking  $\leftarrow$  []
4: valoraciones  $\leftarrow$  valoraciones  $\times$  pesos
5: while size(películas) > 1 do
6:   películaRecomendada  $\leftarrow$  max(valoraciones)
7:   Eliminar películaRecomendada de valoraciones y películas
8:   Añadir películaRecomendada a ranking
9: Añadir última película a ranking
10: return ranking

```

---



---

**Algorithm 4** Obtención de pesos para el método del más cinéfilo
 

---

```

1: for usuario in grupo do
2:   películasVistas[usuario]  $\leftarrow$  obtenerPelículas(usuario)
3: totalPelículasVistas  $\leftarrow$  size(películasVistas)
4: for usuario in películasVistas do
5:   pesos[usuario]  $\leftarrow$   $\frac{\textit{películasVistas}[\textit{usuario}]}{\textit{totalPelículasVistas}}$ 
6: return pesos

```

---

De nuevo se ha separado el pseudocódigo en dos partes claramente diferenciables: el cálculo de los pesos de acuerdo al método del más cinéfilo y el pseudocódigo del método en sí. En este caso sí tiene sentido explicar cómo se calculan los pesos en primer lugar, dado que esta vez **no son pesos dinámicos**. Es decir, el peso se calculará una única vez al inicio de la ejecu-

ción del algoritmo y no se volverá a recalcular durante el proceso. El cálculo de los pesos tan solo se basa en la cantidad de películas que un usuario ha visto y, como es lógico, esta cantidad no variará durante el transcurso del algoritmo.

Lo primero que hay que tener claro a la hora de decidir el grado de cinefilia de un usuario es que únicamente se puede trabajar **con las películas del conjunto de entrenamiento**. El número de películas que ha visto cada usuario se obtiene a través de dichos datos, y nunca del conjunto de validación. Cuando se ha realizado el conteo para todos los miembros del grupo, se calcula el total de películas vistas por el grupo. El peso asignado a cada miembro corresponderá al porcentaje de películas que ha aportado él a ese total, es decir, cuántas de las películas que ha visto en total el grupo corresponden a su visionado.

Realizar el cálculo de esta manera, sin tratar con datos absolutos sino trabajando meramente con lo que hay en el grupo, permite que lo que se valore más que el número de películas total sea en realidad la diferencia existente con respecto al total de otros miembros. Así se da el valor adecuado tanto a grupos que en total han visto pocas películas como a grupos repletos de cinéfilos que hayan visto cientos de ellas. También se contempla de forma intrínseca el caso de que haya un único cinéfilo rodeado de un grupo de casi inexpertos – en esa hipotética situación, el peso a la hora de tomar decisiones del miembro cinéfilo será sustancialmente mayor que el resto, dando a entender que su extenso rodaje puede dirigir al grupo frente a la inexperiencia de sus compañeros.

De nuevo, en este algoritmo hay que predecir las valoraciones de las películas que se van a evaluar. Como se ha indicado en el apartado anterior, este método de predicción se detallará en el capítulo de experimentación, ya que sea cual sea el escogido no afecta al desarrollo del algoritmo que se está analizando. Un detalle que cambia con respecto al algoritmo anterior es que las valoraciones pueden sobrescribirse directamente por sus equivalentes ponderadas, multiplicándolas por los pesos. Para el proceso de selección de películas de este método no volverán a hacer falta nada más que las valoraciones ponderadas, que son las que se tendrán en cuenta a la hora de decidir qué película escoger cada vez.

Con las valoraciones adecuadamente predichas y posteriormente ponderadas, se produce un bucle que terminará cuando no queden películas. En este bucle se calcula en cada iteración cuál es la película que más puntuación recibe, obteniendo dicho dato de la suma de dichas valoraciones ponderadas. Cabe recordar que estas valoraciones ya contienen la información de quiénes son los usuarios más cinéfilos y qué películas les gustan más, con lo cual se puede tratar la matriz con relativa sencillez, sin hacer mayor cálculo. Simplemente cuando se obtenga la película seleccionada, hay que eliminarla de

la matriz para que no vuelva a ser escogida, así como de la lista de películas disponibles para el grupo. Esta película se incluye en el ranking de películas a recomendar.

Cuando todas las películas hayan sido incluidas por orden en el ranking, este se puede devolver como resultado final. Este ranking deberá tener en sus primeras posiciones, como característica descriptiva, una serie de películas que se consideren “objetivamente buenas” por las críticas, ya que la opinión de los usuarios expertos influirá mucho en la toma de decisiones. Por supuesto, esto vendrá determinado por el nivel de experiencia de los miembros del grupo, y esta característica no tiene por qué darse si no se trata con un grupo con suficiente nivel, por lo que es algo a tener en cuenta a la hora de usar este método en un problema real.

### 4.3. Método del más optimista

En este método se plantea una hipótesis consistente en que un usuario que valora más positivamente las películas será más propicio a disfrutarlas y a recomendar películas que sean de gusto e interés general. Si bien en algunos casos un usuario de estas características pudiera ser tachado de influenciable y poco crítico, el planteamiento que se le ha dado a este método parte de la premisa de que siempre será mejor contar con un exceso de positivismo que con una lacra del mismo. Asemejándolo al panorama social que se ha comentado en los apartados anteriores, un miembro del grupo de carácter entusiasta que valore positivamente la mayoría de las películas que ve puede transmitir al grupo una sensación de satisfacción general, mientras que una persona excesivamente crítica y al que no le convenza nada de lo que están viendo puede rápidamente minar la moral del grupo con sus valoraciones negativas.

De nuevo se le ha añadido un componente de pesos que definen cómo de optimista es un miembro del grupo con respecto al resto. De la misma forma que ocurría con los pesos del método del más cinéfilo, se calcularán en función de las valoraciones que dejen los usuarios del grupo, y no se tratará con valores absolutos a fin de adaptar el método a todos los casos posibles.

A continuación se puede apreciar un fragmento de pseudocódigo que representa el algoritmo en cuestión:



**Algorithm 5** Método del más optimista

---

```

1:  $valoraciones \leftarrow predecirValoraciones()$ 
2:  $pesos \leftarrow obtenerPesosOptimistas()$ 
3:  $ranking \leftarrow []$ 
4:  $valoraciones \leftarrow valoraciones \times pesos$ 
5: while  $size(películas) > 1$  do
6:    $películaRecomendada \leftarrow max(valoraciones)$ 
7:   Eliminar  $películaRecomendada$  de  $valoraciones$  y  $películas$ 
8:   Añadir  $películaRecomendada$  a  $ranking$ 
9: Añadir última  $película$  a  $ranking$ 
10: return  $ranking$ 

```

---

**Algorithm 6** Obtención de pesos para el método del más optimista

---

```

1: for  $usuario$  in  $grupo$  do
2:    $mediaValoraciones[usuario] \leftarrow mean(valoraciones[usuario])$ 
3:  $totalMediaValoraciones \leftarrow sum(mediaValoraciones)$ 
4: for  $usuario$  in  $mediaValoraciones$  do
5:    $pesos[usuario] \leftarrow \frac{mediaValoraciones[usuario]}{totalMediaValoraciones}$ 
6: return  $pesos$ 

```

---

Como en casos anteriores aparece el pseudocódigo general del algoritmo separado del cálculo de los pesos para distinguir y hacer más legibles ambos fragmentos. De nuevo aquí se pueden calcular los pesos una única vez al inicio del proceso ya que no van a variar en mitad de la ejecución del algoritmo, sino basándose en las valoraciones dadas a priori.

Estas valoraciones tienen que ser extraídas también del conjunto de entrenamiento, las valoraciones del conjunto de test no tienen cabida en este momento dado que son las que se quieren predecir y se deben omitir para que no sesguen el algoritmo. En este caso lo que se busca obtener es la valoración media que los usuarios dan a las películas que ven, y a raíz de eso comprobar cómo de optimistas son respecto al cine.

Al tratar únicamente con los datos de los miembros del grupo se pueden calcular los pesos de cada usuario en relación al resto de miembros. No existe una escala predefinida que determine que si un usuario valora por encima de una cantidad  $X$  va a ser considerado optimista y si está por debajo pesimista, sino que lo que es considerado optimista vendrá dado por las características de los usuarios que componen el grupo.

Para el cálculo de los pesos de este método se obtiene la valoración media entre todas las que ha dado cada usuario, y se suman, para obtener un total de valoraciones medias. De ese total, a cada usuario se le asigna como peso el porcentaje del total que corresponde a su valoración media, equilibrándose

así de forma autónoma aquellos casos en los que haya mucha diferencia entre las valoraciones medias de los miembros del grupo, así como aquellos en los que los usuarios sean prácticamente similares, tanto si tienen valoraciones medias bajas o altas.

Con las valoraciones obtenidas mediante el algoritmo de predicción que se explicará en el capítulo de experimentación, estas son multiplicadas por los pesos obtenidos en el paso anterior, dando lugar a la matriz de valoraciones ponderadas con la que se trabajará en el resto del proceso algorítmico. Una vez ponderadas, estas valoraciones ya tienen implícita la información acerca del optimismo o pesimismo de cada uno de los usuarios, por lo que se puede trabajar directamente con ellas.

Llegado a este punto, el proceso de selección de película es similar al anterior, simplemente queda calcular la suma de las valoraciones ponderadas y así comprobar cuál es la que tiene mayor puntuación para el grupo constituido. La película elegida se incluye en el ranking de películas a recomendar al grupo, y es eliminada a su vez de la matriz y de la lista de películas disponibles, para que no pueda ser recomendada otra vez y no vuelva a aparecer en el ranking.

Cuando se hayan seleccionado todas las películas, se habrá formado un ranking que se devuelve como respuesta del sistema de recomendación. El ranking que se obtiene de este método de recomendación propicia que se seleccionen películas que le hayan gustado a aquellas personas que mejor valoran en general, por lo que se puede esperar que la tónica general de las películas que copen las primeras posiciones de la lista sean películas de esta índole, al menos en cuanto a los usuarios más optimistas del grupo. Está claro que esto no implica que vayan a gustarle forzosamente al resto de miembros, pero sí que es una estrategia de recomendación interesante desde el punto de vista social – y que a través de la experimentación se busca averiguar si también puede operar a un buen nivel de eficacia.

## 4.4. Método de mayor similitud

La última propuesta planteada difiere un poco de las anteriores, ya que en lugar de partir de un acuerdo entre los miembros del grupo, dándoles a cada uno de ellos una capacidad determinada de influencia en la toma de decisiones, aquí se busca identificar a **un único miembro del grupo** como el representante del mismo. En este aspecto se asemeja más al algoritmo del estado del arte, pero se enfoca de un modo distinto, como se explicará en los próximos párrafos.

Utilizar pesos resulta una forma clara de otorgar balance al grupo dándole una influencia distinta a cada miembro basándose en los datos recopilados

del conjunto de entrenamiento. Para este caso no son adaptables, puesto que se quiere buscar un único perfil de usuario, y para ello hay que encontrar otra forma de decidir qué determina que un usuario represente más al grupo. En el enfoque social que se le está dando a los métodos que componen este estudio, esta propuesta resulta quizá un poco más compleja de ubicar que las anteriores, así como pasaba con el algoritmo del estado del arte escogido.

Se puede decir que en el supuesto de un grupo de amigos que van a ver una película juntos, sabiendo de sobra qué películas les gustan más y cuáles menos a cada uno de los componentes del mismo, esta propuesta trata de definir cómo el propio grupo conoce que, dadas las experiencias pasadas, las películas que más le gustan a un determinado miembro son las que más satisfecho dejan al resto. Esto puede deberse a mera casuística, o puede tener un sentido particular, como que a dicha persona le gusten las películas de un género específico o de un director en concreto.

A continuación se muestra un fragmento de pseudocódigo que ilustra el funcionamiento del método de mayor similitud:

---

**Algorithm 7** Método de mayor similitud
 

---

```

1: valoraciones  $\leftarrow$  predecirValoraciones()
2: matrizPearson  $\leftarrow$  obtenerCorrelaciónPearson()
3: usuarioRepresentante  $\leftarrow$  max(mean(matrizPearson[usuario]))
4: for película in películas do
5:   Añadir valoraciones[usuario][película] a ranking
6: Ordenar ranking decreciente
7: return ranking
  
```

---

A diferencia de los otros métodos, en este no aparece el pseudocódigo del cálculo de pesos aparte, ya que no tiene cabida en este método. Y además, al tratarse de un caso menos extenso en cuanto a complejidad algorítmica se ha considerado mejor incluirlo todo en un único bloque de pseudocódigo.

Lo primero como siempre es predecir las valoraciones individuales del conjunto de validación que se quieren evaluar durante el experimento. En el capítulo dedicado a la experimentación se hará referencia con todo detalle a cómo se han predicho estas valoraciones durante todo el estudio. Por ahora, simplemente cabe recordar que estas valoraciones no se pueden obtener del conjunto de validación sino que deben ser predichas a fin de que el algoritmo sea realmente funcional y aplicable a un problema de la vida real, y no tenga una opinión sesgada en base a dichas valoraciones.

Lo que más destaca de este método es la utilización de la **matriz de correlación de Pearson**. Si bien más adelante en la experimentación se podrá observar cómo esta matriz de Pearson ha sido necesaria en pasos previos al experimento, de cara al pseudocódigo del algoritmo se debe tratar

como si se construyese en este punto. En el capítulo de experimentación se analizará detalladamente cómo se construye esta matriz, pero por ahora lo importante es destacar que representa la correlación existente entre unos y otros usuarios. Y por tanto, el funcionamiento del algoritmo consiste en obtener aquel usuario cuyo coeficiente de Pearson (es decir, su correlación) medio con respecto al resto de miembros del grupo sea mayor. De esa persona se dirá que es el que más se parece a los intereses generales del grupo, y por ello será catalogado como el **usuario representante**.

Este usuario es al que se le hará la recomendación, igual que ocurría en el método del estado del arte. Dicha recomendación, a pesar de que se hace a un único usuario y a las valoraciones predichas para él, queda como resultado del algoritmo y es el ranking global que el sistema de recomendación hace al grupo completo.

Dado que se está utilizando la fórmula de la matriz de correlación de Pearson, en este caso se está implementando una forma de reconocer en base a lo ocurrido en el pasado qué usuario es más representativo del grupo, lo cual puede lograrse en el símil social mediante la experiencia en películas pasadas. No se trata de que un usuario determinado sea capaz de predecir con toda firmeza cuánto va a valorar la película, sino que otros miembros del grupo se sientan identificados con sus elecciones pasadas. Si se da un caso así, no sería de extrañar que una decisión del grupo fuera dejar escoger película a aquella persona que parece representar a la mayoría del grupo. En el capítulo de resultados se comprobará cómo de adecuada es esta propuesta en términos de eficacia.

# Bibliografía

- [1] Elsevier Scopus Database. <https://www.scopus.com>.
- [2] ResearchGate Scientific Knowledge Database. <https://www.researchgate.net>.
- [3] Pop\_OS! by System76. <https://system76.com/pop>.
- [4] Visual Studio Code - Open Source, repositorio de GitHub. <https://github.com/Microsoft/vscode>.
- [5] TeXstudio, página oficial. <https://www.texstudio.org>.
- [6] D. Goldberg, D. Nichols, B.M. Oki, and D. Terry. Using collaborative filtering to Weave an Information tapestry. *Communications of the ACM*, 35:61–70, 1992.
- [7] M. O'Connor, D. Cosley, J.A. Konstan, and J. Riedl. PolyLens: A Recommender System for Groups of Users, 2001.
- [8] Netflix Prize. <https://www.netflixprize.com/>.
- [9] Netflix Prize Forum: Grand Prize awarded to team BellKor's Pragmatic Chaos. [https://www.netflixprize.com/community/topic\\_1537.html](https://www.netflixprize.com/community/topic_1537.html).
- [10] J. Masthoff. Group recommender systems: Combining individual models. *Recommender Systems Handbook*, pages 677–702, 2011.
- [11] J.F. McCarthy and T.D. Anagnost. MusicFX: An Arbiter of Group Preferences for Computer Supported Collaborative Workouts. In *Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work*, CSCW '98, pages 363–372, New York, NY, USA, 1998. ACM.
- [12] L. Ardissono, A. Goy, G. Petrone, M. Segnan, and P. Torasso. Intrigue: personalized recommendation of tourist attractions for desktop and handset devices. *Applied AI, Special Issue on Artificial Intelligence for Cultural Heritage and Digital Libraries*, 17(8-9):687–714, 2003.

- [13] S. Dara, R. Chowdary, and C Kumar. A survey on group recommender systems. *Journal of Intelligent Information Systems*, January 2019.
- [14] A. Agarwal, M. Chakraborty, and R. Chowdary. Does order matter? Effect of order in group recommendation. *Expert Systems with Applications*, 82:115–127, October 2017.
- [15] J. Castro, R. Yera, and L. Martinez. An empirical study of natural noise management in group recommendation systems. *Decision Support Systems*, 94:1–11, February 2017.
- [16] L. Quijano-Sánchez, J. Recio-García, B. Díaz-Agudo, and G. Jiménez-Díaz. Happy Movie: A Group Recommender Application in Facebook. *Proceedings of the 24th International Florida Artificial Intelligence Research Society, FLAIRS - 24*, January 2011.
- [17] L. Quijano-Sánchez, C. Sauer, J. Recio-García, and B. Díaz-Agudo. Make it Personal: A Social Explanation System Applied to Group Recommendations. *Expert Systems with Applications*, 76, January 2017.
- [18] N. Choudhary and K.K. Bharadwaj. Evolutionary learning approach to multi-agent negotiation for group recommender systems. *Multimedia Tools and Applications*, 78:16221–16243, 2019.
- [19] R. Katarya and O.P. Verma. Recommender system with grey wolf optimizer and FCM. *Neural Computing and Applications*, 30:1679–1687, September 2018.
- [20] N. Choudhary and K.K. Bharadwaj. Preference-Oriented Group Recommender System. *Applications of Artificial Intelligence Techniques in Engineering. Advances in Intelligent Systems and Computing*, 698:581–589, 2018.
- [21] Y. Freund and R.E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55:119–139, August 1997.

