

UNIVERSIDAD COMPLUTENSE DE MADRID
FACULTAD DE INFORMÁTICA



TESIS DOCTORAL

**Sistemas de recomendación y explicaciones basados en grafos
de interacción**

MEMORIA PARA OPTAR AL GRADO DE DOCTOR

PRESENTADA POR

Marta Caro Martínez

Directores

Guillermo Jiménez Díaz
Juan Antonio Recio García

Madrid

© Marta Caro Martínez, 2022

UNIVERSIDAD COMPLUTENSE DE MADRID
FACULTAD DE INFORMÁTICA



TESIS DOCTORAL

Sistemas de recomendación y explicaciones basados en grafos de interacción

MEMORIA PARA OPTAR AL GRADO DE DOCTORA

PRESENTADA POR

Marta Caro Martínez

DIRECTORES

Guillermo Jiménez Díaz
Juan Antonio Recio García

Sistemas de recomendación y
explicaciones basados en grafos de
interacción



TESIS DOCTORAL

Marta Caro Martínez

Directores

Guillermo Jiménez Díaz

Juan Antonio Recio García

Facultad de Informática
Universidad Complutense de Madrid

Madrid, 2021

Documento maquetado con T_EX!S v.1.0.

Sistemas de recomendación y explicaciones basados en grafos de interacción

*Memoria que presenta para optar al título de Doctor en Ingeniería
Informática*

Marta Caro Martínez

Dirigido por:

**Guillermo Jiménez Díaz
Juan Antonio Recio García**

**Facultad de Informática
Universidad Complutense de Madrid**

Madrid, 2021

Para mis abuelos

Agradecimientos

Cuando pienso en aquellas personas sin las cuales no habría podido llevar a cabo esta tesis, me acuerdo de mucha gente, espero acordarme de todos vosotros en estas líneas. Los primeros que me vienen a la cabeza son mis directores Guille y Juanan. Aún recuerdo cuando me matriculé en el doctorado por primera vez esperando que, de alguna manera, se alinearan los planetas para conseguir una beca y continuar mi TFM. La beca no la conseguí en ese momento, pero aún así decidisteis darme una oportunidad y aquí estoy tres años y medio después. Gracias por haber confiado en mí desde entonces, por guiarme, enseñarme y haber puesto tanto empeño en que esta tesis sea la mejor posible. Es un placer trabajar con vosotros y haber sido vuestra doctoranda.

Al grupo GAIA le quiero agradecer su confianza en mí, es también gracias a vosotros por quien pude empezar a financiar mi doctorado y por quien he podido ir a presentar mi trabajo a distintos congresos. Quería hacer especial mención a Belén y Antonio, sin cuya ayuda y consejos los ICCBR no hubieran sido la experiencia tan enriquecedora que fue. Muchas gracias también a Maxi y Diana, por formar parte de estos congresos y de algunas de las mejores y más divertidas experiencias que he vivido durante estos años.

Muchas gracias a los miembros del departamento de Ingeniería de Software e Inteligencia Artificial: Gonzalo, Raquel, Mercedes, Virginia, Lourdes... siempre que he necesitado vuestra ayuda o consejo, habéis estado dispuestos a echar una mano. Otros miembros y profes de la FDI también me han dado ánimos y consejos en este tiempo. No puedo evitar acordarme de Rafa Caballero, quien me dio muy buenos consejos tras acabar la carrera y ha seguido dándome ánimos por los pasillos y en las paradas de autobús. También quería dar las gracias a todos aquellos que alguna vez han formado parte de algún experimento o han contestado alguno de nuestros cuestionarios, sin vosotros, una parte del trabajo descrito en esta memoria no hubiera sido posible.

A mis compañeros y amigos del despacho 409, en especial a Rome, Isma, Marlon, Lolo, Antonio y Mari Cruz, gracias por las charlas de todos los días,

las bromas de los jueves y las patatas de los viernes. En este despacho, se merece mención especial mi amigo Jose. Aparte de por lo anterior, gracias por tu generosidad y ayuda, por tus chistes, y por hacer que formemos el mejor equipo de becarios GAIA. También, quería agradecer a mis amigos del Aula 16 haber sido uno de mis mayores apoyos durante estos años (y antes de empezar la tesis también). Cris, Toni, Pablo, Miguel, Jesús, Luisma y Ali, muchas gracias por los cafés, las terracitas y los grupos de telegram (uno para cada evento), entre otras muchas cosas. Pero sobre todo, muchas gracias por estar siempre para escucharme.

Al resto de mis amigos, muchas gracias por haber seguido ahí a pesar de los años e incluso de la distancia, apoyando de cierta manera esta tesis al preguntarme sobre mis avances en el doctorado. Aquí tengo que hacer una especial mención (ya la última) a mi amigo Christian, por seguir siendo uno de mis mejores y más incondicionales amigos después de once años.

Quiero agradecer a mi familia ser los que verdaderamente han disfrutado y, alguna que otra vez, sufrido conmigo este proceso. A mis tíos y primos, por preocuparos tanto, interesaros siempre y aconsejarme. A mis padres y mi hermana Lidia, por apoyarme incondicionalmente, interesándoos sobre mi día a día en la facultad, sobre qué estaba haciendo, aunque sólo entendiéis que era algo de inteligencia artificial, sobre mis resultados, mis clases, y en general, por todo. Esta tesis en parte es vuestra también.

Por último, quiero acordarme de mis abuelos, gracias a los cuales soy quien soy y por los cuales estoy aquí, sé que estáis orgullosos.

Resumen

En la sociedad actual, con la utilización de nuevas tecnologías en todos los ámbitos de nuestras vidas, el consumo en Internet ha avanzado a pasos agigantados. Los usuarios pueden encontrar infinidad de productos que consumir en cualquier ámbito, especialmente a través del comercio electrónico, las redes sociales y el entretenimiento en *streaming*. Determinar qué productos son los más adecuados según sus necesidades y preferencias puede llegar a ser una tarea complicada debido a la amplitud de la oferta de productos que se pueden encontrar en estas plataformas. Los sistemas de recomendación surgen para resolver este problema, facilitando el proceso de búsqueda y de toma de decisiones. No obstante, cuando los usuarios no confían en el sistema, los sistemas de recomendación no son tan útiles como cabría esperar. Esto se produce debido a que no entienden cómo el sistema ha llegado a la conclusión de que un determinado producto es adecuado para ellos. Los sistemas de explicación en recomendadores ayudan a los usuarios a comprender el funcionamiento del recomendador y las causas de una determinada recomendación.

Los sistemas de recomendación y explicación tradicionales utilizan el conocimiento de las valoraciones que realizan los usuarios o sobre descripciones que especifican distintos atributos de los productos. Sin embargo, no siempre es posible usar estas dos fuentes de conocimiento ya que existen situaciones en las que no se tiene esta información o bien no existe. En esta tesis doctoral se propone resolver este problema a través del estudio y uso de grafos de interacción para generar métodos de recomendación y métodos de explicación para recomendadores. Como consecuencia de esta investigación, en esta tesis doctoral se han obtenido cuatro resultados principales: un modelo conceptual para clasificar y definir sistemas de explicación en recomendadores, un método de visualización interactivo basado en grafos, un método de recomendación basado en grafos de interacción y un método de explicación para recomendadores que también hace uso de grafos de interacción. A diferencia de las técnicas tradicionales, estas dos últimas propuestas, sólo requieren la información sobre las interacciones entre los usuarios y los productos, información a la que siempre se puede acceder. Por tanto, utilizan una mínima cantidad de conocimiento, siendo capaces de generar recomendaciones y ex-

plicaciones en escenarios donde las técnicas tradicionales no pueden. Además de aumentar la precisión de las técnicas clásicas, los métodos propuestos en esta tesis aportan una interpretabilidad que no ofrecen las anteriores técnicas.

Palabras clave: Sistemas de recomendación, Sistemas de explicación, Grafos, Técnicas de predicción de enlaces, Inteligencia Artificial Explicable, XAI, Modelo conceptual, Taxonomía, Ontología, Visualización, Interactividad

Abstract

Nowadays, with new technologies in all areas of our lives, Internet consumption has advanced quickly. Users can find an infinite number of products to consume in any field, especially through e-commerce, social networks and streaming entertainment. Determining which products are the most suitable according to their needs and preferences can become a complicated task due to the breadth of product offerings that can be found on these platforms. Recommendation systems emerge to solve this problem, facilitating the search and decision-making process. However, when users do not trust the system, recommender systems are not as useful as one might expect because they do not understand how the system has concluded that a particular product is suitable for them. Explanation systems in recommenders help users understand how the recommender works and the reasons for a particular recommendation.

Traditional recommender and explanation systems use knowledge from user ratings or item descriptions. However, it is not always possible to use these two knowledge sources as there are situations in which this information is not available or does not exist. In this doctoral thesis, we propose to solve this problem through the study and use of interaction graphs to generate recommendation methods and explanation methods for recommenders. As a consequence of this research, four main results have been obtained in this doctoral thesis: a conceptual model to classify and define explanation systems in recommenders, an interactive visualisation method based on graphs, a recommendation method based on interaction graphs and an explanation method for recommenders that also makes use of interaction graphs. Unlike traditional techniques, these last two proposals only require information about the interactions between users and items, information that is always accessible. Therefore, they use a minimum amount of knowledge, being able to generate recommendations and explanations in scenarios where traditional techniques cannot. In addition to increasing the accuracy of classical techniques, the methods proposed in this thesis provide an interpretability not provided by previous techniques.

Keywords: Recommender systems, Explanation systems, Graphs, Link prediction techniques, eXplainable Artificial Intelligence, XAI, Conceptual model, Taxonomy, Ontology, Visualisation, Interactivity

Índice

Agradecimientos	XI
Resumen	XIII
Abstract	XV
1. Introducción	1
1.1. Hipótesis y objetivos	2
1.2. Plan de trabajo y organización de la tesis	4
1.3. Publicaciones	6
2. Estado del arte	9
2.1. Sistemas de recomendación	10
2.1.1. Tipos de sistemas de recomendación	11
2.1.2. Recomendadores basados en grafos	21
2.1.3. Conocimiento usado por los sistemas de recomendación	23
2.1.4. Retos y nuevas tendencias de los sistemas recomenda-	
dores	25
2.2. Sistemas de explicación	29
2.2.1. Características de los sistemas de explicación para re-	
comendadores	30
2.2.2. Sistemas de explicación para recomendadores	32
2.2.3. Sistemas de explicación para recomendadores basados	
en grafos	34
2.3. Técnicas de predicción de enlaces	39
2.4. Conclusiones	43
3. Modelo conceptual de explicaciones en sistemas de recomen-	
dación	45
3.1. Trabajo relacionado	47
3.2. Representación del modelo	49
3.2.1. Entidades de dominio sobre sistemas de recomendación	49

3.2.2. Relaciones semánticas	50
3.3. Descripción del modelo	50
3.3.1. Motivación	52
3.3.2. Conocimiento	55
3.3.3. Generación	57
3.3.4. Presentación	60
3.4. Validación del modelo	64
3.4.1. Metodología	64
3.4.2. Resultados y discusión	66
3.5. ExRecOnto: implementación ontológica del modelo conceptual	70
3.6. Conclusiones	74
4. Sistemas de recomendación basados en grafos de interacción	77
4.1. Recomendaciones basadas en grafos de interacción	79
4.1.1. Proceso de recomendación basado en grafos de productos	81
4.1.2. Proceso de recomendación basado en grafos de usuarios	84
4.1.3. Medidas de similitud basadas en técnicas de predicción de enlaces	87
4.1.4. Métodos de agregación	90
4.2. Evaluación	93
4.2.1. Análisis del conjunto de datos	94
4.2.2. Método de evaluación	96
4.2.3. Escenario de mínimo conocimiento	97
4.2.4. Métricas de evaluación	98
4.3. Resultados	100
4.3.1. Análisis de la precisión de los grafos de interacción . .	100
4.3.2. Comparación con las técnicas clásicas	104
4.3.3. Explicabilidad de los modelos	108
4.4. Discusión de resultados	109
4.5. Conclusiones	110
5. Sistemas de explicación para recomendadores basados en grafos de interacción	113
5.1. Explicaciones basadas en grafos de interacción	114
5.1.1. Escenario de mínimo conocimiento	115
5.1.2. Definición del sistema de explicación propuesto según el modelo conceptual	117
5.1.3. Proceso de explicación basado en grafos de productos .	118
5.1.4. Proceso de explicación basado en grafos de usuarios .	119
5.2. Evaluación	121
5.2.1. Análisis del conjunto de datos	122

5.2.2. Metodología de evaluación	123
5.2.3. Análisis de los resultados	125
5.3. Conclusiones	131
6. Visualización de explicaciones basada en grafos	133
6.1. El sistema de explicación UPIEM	135
6.2. Visualización interactiva basada en grafos	138
6.2.1. Comparativa con la visualización directa del grafo	142
6.2.2. Definición del sistema de explicación propuesto según el modelo conceptual	143
6.3. Evaluación	144
6.3.1. Conjunto de datos	145
6.3.2. Metodología de evaluación	145
6.3.3. Análisis de los resultados	148
6.4. Conclusiones	154
7. Conclusiones y trabajo futuro	157
7.1. Trabajo futuro	160
A. Implementación de ExRecOnto	163
Bibliografía	167

Índice de figuras

2.1. Clasificación de las estrategias de combinación de métodos recomendadores para obtener sistemas híbridos. Figura inspirada en el trabajo de (Aggarwal, 2016).	18
2.2. Esquema de diseño paralelo de un sistema recomendador híbrido. Figura inspirada en el trabajo de (Aggarwal, 2016). . .	19
2.3. Esquema de diseño secuencial de un sistema recomendador híbrido. Figura inspirada en el trabajo de (Aggarwal, 2016). . .	19
2.4. Ejemplo de la visualización de una explicación proporcionada por el sistema PeerChooser para una recomendación.	37
2.5. Ejemplo de la visualización proporcionada por el sistema TalkExplorer de una explicación para una recomendación.	38
2.6. Ejemplo de predicción de enlaces	40
2.7. Diferencia de funcionamiento entre métodos basados en similitud y basados en aprendizaje	41
2.8. Clasificación de los métodos de predicción de enlaces.	42
3.1. Visión general del modelo conceptual.	51
3.2. Ejemplo de una explicación <i>Ante hoc</i>	58
3.3. Ejemplo de una explicación <i>Post hoc</i>	59
3.4. Ejemplo de una explicación en <i>Lenguaje natural</i>	60
3.5. Ejemplo de una explicación <i>Esquemática</i>	61
3.6. Ejemplo de una explicación <i>Proactiva</i>	64
3.7. Número de sistemas clasificados según la <i>Motivación</i>	69
3.8. Número de sistemas clasificados según el <i>Conocimiento</i>	70
3.9. Número de sistemas clasificados según la <i>Generación</i>	70
3.10. Número de sistemas clasificados según la <i>Presentación</i>	71
3.11. Visualización interactiva de ExRecOnto.	72
3.12. Ejemplo de uso de ExReconto.	74
4.1. Transformación de un grafo bipartito en un grafo no bipartito usando proyección de grafos bipartitos.	80

4.2.	Proceso de recomendación llevado a cabo con el grafo de interacción de productos.	82
4.3.	Proceso de recomendación llevado a cabo con el grafo de interacción de usuarios.	85
4.4.	Estudio del conjunto de datos de MovieLens para aumentar su densidad en la evaluación de los modelos de recomendación.	97
4.5.	Análisis del crecimiento del tamaño k de la lista de recomendación en los modelos de recomendación	107
4.6.	Explicación visual de una recomendación real obtenida con modelos basados en productos.	108
5.1.	Visión general del sistema de explicación basado en grafos de interacción.	116
5.2.	Proceso de explicación llevado a cabo con el grafo de interacción de productos.	119
5.3.	Proceso de explicación llevado a cabo con el grafo de interacción de usuarios.	121
5.4.	Gráfico que representa los resultados obtenidos con el conjunto de datos original en la evaluación de los métodos de explicación.	128
5.5.	Gráfico que representa los resultados obtenidos con el conjunto de datos estratificado en la evaluación de los métodos de explicación.	129
6.1.	Vista general del funcionamiento del sistema de explicación UPIEM.	136
6.2.	Ejemplo de retículo construido con FCA.	137
6.3.	Ejemplo del método de visualización propuesto.	140
6.4.	Interfaz interactiva para cambiar de recomendación.	141
6.5.	Método de eliminación de atributos de un nodo con varios atributos compartidos.	142
6.6.	Ejemplo de cómo se mostraría la explicación mostrada en la Figura 6.3 con una visualización basada en grafos más tradicional.	143
6.7.	Aceptación de cada explicación por parte de los usuarios.	150
6.8.	Respuestas a la segunda sección del cuestionario.	151
6.9.	Análisis de la eficiencia de nuestro sistema de explicación.	152
6.10.	Análisis de la eficiencia y la scrutabilidad de nuestro sistema de explicación.	153
6.11.	Análisis del <i>producto de explicación</i> de UPIEM.	154
A.1.	Visualización parcial de la ontología ExRecOnto con respecto al concepto Motivación	163

A.2. Visualización parcial de la ontología ExRecOnto con respecto al concepto Conocimiento	164
A.3. Visualización parcial de la ontología ExRecOnto con respecto al concepto Generación	164
A.4. Visualización parcial de la ontología ExRecOnto con respecto al concepto Presentación	165
A.5. Visualización de las relaciones existentes en la ontología Ex-RecOnto	166

Índice de Tablas

2.1. Comparación de las características de los tipos de sistemas recomendadores.	12
2.2. Comparativa de tipos de <i>ratings</i>	24
2.3. Comparativa sobre los problemas en los principales tipos de recomendadores.	26
3.1. Colección de sistemas de explicación validados con el modelo conceptual.	66
3.2. Clasificación de los sistemas del conjunto de validación de acuerdo al modelo conceptual propuesto.	67
3.3. Leyenda para la Tabla 3.2 y Figuras 3.7-3.10.	68
4.1. Ejemplo de aplicación de diferentes métodos de agregación . .	92
4.2. Análisis descriptivo del conjunto de datos utilizado para la evaluación de los modelos de recomendación.	95
4.3. Conocimiento mínimo necesario para crear los modelos de recomendación.	98
4.4. Resultados obtenidos en la evaluación de los métodos de recomendación basados en los grafos de productos y de usuarios. 103	
4.5. Comparación entre los resultados obtenidos con los modelos basados en productos.	105
4.6. Comparación entre los resultados obtenidos con los modelos basados en usuarios.	106
5.1. Análisis de los conjuntos de datos utilizados en la evaluación de los modelos de explicación	124
5.2. Valores de RMSE obtenidos en la evaluación con los métodos de explicación aplicados sobre el conjunto de evaluación original 126	
5.3. Valores de RMSE obtenidos en la evaluación con los métodos de explicación aplicados sobre el conjunto de evaluación estratificado	127
5.4. Porcentaje de mejora de los métodos basados en grafos respecto al modelo basado en contenido.	130

6.1. Herramientas y preguntas del cuestionario utilizadas para evaluar las características de UPIEM.	146
6.2. Evaluación con el cuestionario PSSUQ	148
6.3. Preguntas de la segunda parte del cuestionario con sus correspondientes resultados.	149

Capítulo 1

Introducción

Hoy en día con el auge del comercio electrónico y las plataformas de entretenimiento en *streaming*, podemos acceder a una gran oferta de productos que consumir, como pueden ser películas y series, alimentos, restaurantes, libros o experiencias turísticas. La oferta es prácticamente infinita en la Web, lo que por un lado es interesante para los usuarios, quienes tienen más posibilidades de encontrar aquellos productos que están buscando. Sin embargo, esto también puede resultar abrumador, debido a la imposibilidad de localizar aquellos productos que sean más adecuados para ellos. Los sistemas de recomendación surgen para resolver este problema. Su principal objetivo es determinar qué productos se ajustan más a los intereses del usuario objetivo de la recomendación, para ayudarle a tomar una decisión apropiada. Este tipo de sistemas han sido estudiados desde hace varias décadas y están instaurados en todo tipo de plataformas que usamos en el día a día, como Amazon¹, Netflix² o Spotify³.

Uno de los principales problemas que sufren los sistemas de recomendación es la falta de confianza de sus usuarios. Las personas no confían en aquello que no entienden, por lo que si no saben cómo funciona un sistema recomendador o las razones por las que se produce una recomendación, los usuarios no confiarán en el sistema. Esto afecta a la experiencia del usuario, lo que a su vez provoca la reducción del uso del sistema de recomendación y de la propia plataforma. Para solventar este problema han surgido los sistemas de explicación. La Inteligencia Artificial Explicable, XAI por sus siglas en inglés, es uno de los campos más en auge en investigación en ciencias de la computación por la necesidad de que los usuarios conozcan cómo toman las decisiones los sistemas inteligentes. La XAI es especialmente crítica en algunos dominios, como por ejemplo el médico, pero también es importante en el uso de sistemas de recomendación. Implantar sistemas de explicación

¹<https://www.amazon.es>

²<https://www.netflix.com/>

³<https://www.spotify.com/>

en sistemas de recomendación es una necesidad ya que incrementa la utilidad de este tipo de sistemas.

Los sistemas de recomendación y explicación tradicionales están principalmente basados en los *ratings*⁴ con los que los usuarios valoran los productos que consumen, o en las descripciones de los propios productos. Sin embargo, hay situaciones en las que no podemos utilizar este conocimiento, bien porque no existe o porque no podemos acceder a él. Este **escenario de mínimo conocimiento** se da cuando la única información que tenemos sobre los usuarios son las interacciones que han realizado en la plataforma: han comprado un determinado producto, han visto una película o han escuchado una canción, por ejemplo, pero no sabemos las preferencias de esos usuarios por esos productos consumidos. Aunque es posible adaptar las técnicas existentes a este escenario de mínimo conocimiento, no son métodos específicamente pensados para resolver el problema. Son un conjunto de algoritmos que se aplican cuando existen *ratings* de usuarios y que se ajustan para poder usarse en estos escenarios.

En esta tesis doctoral se propone el uso de grafos de interacción, ayudándonos de técnicas de predicción de enlaces, con el objetivo de generar recomendaciones y explicaciones en escenarios de mínimo conocimiento. El principal objetivo es plantear una alternativa a estas técnicas clásicas, generando resultados comparables o incluso mejores, usando menor cantidad de conocimiento. Además, los grafos, junto a las técnicas de predicción de enlaces propuestas en estas tesis, son unas estructuras que pueden aportar una explicabilidad de la que carecen otras técnicas clásicas.

A continuación se formulan la hipótesis y los objetivos de la tesis (Sección 1.1). Tras ello, se expone el plan de trabajo llevado a cabo junto a la organización de la memoria (Sección 1.2). Por último, enumeramos las publicaciones conseguidas en el contexto de esta tesis doctoral (Sección 1.3)

1.1. Hipótesis y objetivos

Las hipótesis planteadas en esta tesis doctoral son que (H1) las técnicas de análisis de grafos permiten generar recomendaciones explicables, además de (H2) explicaciones para otros sistemas de recomendación, en ambos casos requiriendo una cantidad mínima de conocimiento, y adicionalmente que (H3) los grafos son apropiados como técnicas de visualización de explicaciones y, por lo tanto, influyen en la aceptación de la recomendación por parte del usuario. Así podemos definir el objetivo principal de la tesis como el **estudio y desarrollo de métodos de recomendación y explicación basados en grafos de interacción**. Para cumplirlo, hemos dividido este

⁴Se utiliza el término en inglés *ratings* para hablar de las valoraciones que los usuarios realizan sobre los productos, ya que es ampliamente usado y aceptado por la comunidad investigadora.

objetivo principal en cuatro subobjetivos distintos.

El primer objetivo planteado, al que denotamos como Objetivo 0, es **estudiar los sistemas de recomendaciones y distintas técnicas de explicación aplicables**. En la literatura existe una gran cantidad de información sobre sistemas de recomendación. También existen multitud de trabajos y publicaciones sobre sistemas de explicación y XAI aunque, debido al constante crecimiento del campo, en la mayoría de los casos no se encuentra estructurada y claramente definida. Como resultado de una extensa revisión de la literatura, en esta tesis doctoral se propone un modelo conceptual de sistemas de explicación para recomendadores que tiene como objetivo guiar el diseño de estos sistemas. Para ello, en este modelo definimos las posibles características que pueden tener estos sistemas. Además, se propone una ontología que define sistemas de explicación para sistemas de recomendación como formalización del modelo conceptual. Una vez estudiados los sistemas de recomendación y explicación, los siguientes objetivos realizan aportaciones en estas áreas mediante la aplicación de técnicas basadas en grafos de interacción.

Diseñar, desarrollar y evaluar sistemas de recomendación basados en grafos de interacción en escenarios de mínimo conocimiento es el Objetivo 1, propuesto para demostrar la Hipótesis 1. A través de un grafo, podemos representar las interacciones que se han producido entre los usuarios y los productos de un sistema. Los nodos representarían los usuarios y los productos, mientras que los enlaces se crean cuando existe una interacción (si un usuario ha visto una película, ha leído un libro, ha comprado un producto, etc.). Con técnicas de predicción de enlaces, se puede saber si dos nodos son similares y, a través de estas medidas de similitud, generar recomendaciones. La principal y novedosa aportación de los grafos de interacción a los sistemas de recomendación es su aplicabilidad a escenarios de mínimo conocimiento. Ya que el único conocimiento representado en el grafo es el de las interacciones, el mínimo conocimiento que siempre se va a encontrar en un sistema recomendador, proponemos este planteamiento como solución al problema de la escasez de conocimiento para generar recomendaciones en este tipo de escenarios.

Una vez demostrada la viabilidad de los grafos de interacción para desarrollar sistemas de recomendación en escenarios de mínimo conocimiento, el siguiente objetivo de esta tesis consiste en utilizar esta técnicas para también diseñar sistemas de explicación que permitan entender distintos tipos de recomendadores. Por tanto, en el Objetivo 2 de la tesis (correspondiente a resolver la Hipótesis 2), pretendemos **diseñar, desarrollar y evaluar sistemas de explicación basados en grafos de interacción en escenarios de mínimo conocimiento**. De forma análoga al objetivo anterior, usando grafos de interacción podemos generar explicaciones gracias a las similitudes que las técnicas de predicción de enlaces nos pueden proporcionar. Por

tanto, podemos obtener explicaciones **independientemente del método de recomendación aplicado** utilizando menor cantidad de conocimiento que otras técnicas clásicas, ya que los grafos sólo representan la información sobre las interacciones.

Para finalizar, planteamos un tercer Objetivo adicional en esta tesis doctoral (asociado a la Hipótesis 3): **estudiar los grafos como método de visualización para explicar recomendaciones**. En la literatura nos encontramos con muchas y muy diversas alternativas para mostrar explicaciones a los usuarios objetivo de una recomendación. Entre estas, podemos encontrar explicaciones textuales, visuales e interactivas. Los grafos, además de ser una estructura computacional, también sirven como método de visualización en sistemas de recomendación ya que, al acercarse al modelo mental del usuario, es una forma natural y directa de trasladar el conocimiento de la recomendación al usuario. Los grafos incluyen además un componente interactivo que, con mucha frecuencia, resulta útil y satisfactorio para los usuarios. Por ello, la última propuesta de esta tesis doctoral es un método de visualización interactiva basada en grafos de explicaciones para sistemas de recomendación que mejora la explicabilidad de las recomendaciones.

1.2. Plan de trabajo y organización de la tesis

En esta tesis doctoral se ha llevado a cabo un plan de trabajo típico de investigación. Primero se ha definido el problema; después se han formulado las hipótesis que se han descrito en la sección anterior; luego se ha definido el objetivo principal para comprobar esas hipótesis, incluyendo subobjetivos para cumplir el objetivo principal; y por último se han llevado a cabo distintas tareas para cumplir esos objetivos, realizando propuestas de solución y evaluando estas propuestas. Los Capítulos del 3 al 6 corresponden a la ejecución de cada uno de los cuatro objetivos contemplados en esta tesis doctoral. A su vez, cada una de ellas está organizada de acuerdo al plan de trabajo llevado a cabo: motivación, propuesta de solución, evaluación y conclusiones.

En el **Capítulo 2** se hace una revisión de la literatura con respecto a los principales temas que se tratan en la tesis. Se describe la motivación y necesidad de los sistemas de recomendación y su importancia hoy en día, además de los principales tipos de sistemas de recomendación que nos podemos encontrar en la literatura. Con las descripciones de cada tipo de modelo, se aportan ejemplos con el propósito de determinar hasta qué punto se ha estudiado cada tema en la literatura. Esto, junto al estudio de los principales problemas que tienen los sistemas de recomendación, forma parte del conjunto de aportaciones del **Objetivo 0** de la tesis. Posteriormente se hace un estudio sobre XAI y sistemas de explicación. Este estudio no es tan extenso como el realizado sobre recomendadores, ya que en el Capítulo 3 se hace un análisis más en profundidad. Aún así, revisamos las dos fuentes de

conocimiento más importantes que se utilizan en la literatura para generar explicaciones, usando también ejemplos. De nuevo, esta revisión forma parte del **Objetivo 0**. Con el propósito de conocer las herramientas que vamos a utilizar, en el Capítulo 2 también estudiamos los sistemas de recomendación y explicación basados en grafos, los sistemas de recomendación en escenarios de mínimo conocimiento, y las técnicas de predicción de enlaces, siendo estas revisiones aportaciones de los **Objetivos 1 y 2** de la tesis. Lo mismo ocurre con la revisión de los métodos de visualización en explicaciones basadas en grafos, que forman parte de las contribuciones del **Objetivo 3**.

El **Capítulo 3** es tanto un estudio sobre sistemas de explicación para recomendadores como la propuesta de un modelo conceptual para guiar el diseño de este tipo de sistemas. Al describir en profundidad en este modelo los tipos de explicaciones que nos podemos encontrar, estamos dando ejemplos y detallando aspectos teóricos, lo que supone una revisión de la literatura. Además, formalizamos el modelo conceptual con la implementación de la ontología ExRecOnto, que tiene como propósito dirigir el diseño guiado por objetivos de sistemas de explicación en recomendadores. En el **Apéndice A** mostramos esquemas de su implementación. Este capítulo y apéndice son resultados del **Objetivo 0** de esta tesis.

En el **Capítulo 4** planteamos nuestra propuesta de sistema de recomendación que utiliza grafos de interacción junto a técnicas de predicción de enlaces. La propuesta es un conjunto de métodos que se pueden aplicar en escenarios de conocimiento mínimo, donde la única información que se puede adquirir es la información sobre las interacciones. Se evalúa estos métodos comparándolos con otras técnicas clásicas de recomendación que utilizan más cantidad de conocimiento y son menos explicables. Estas tareas forman parte de las contribuciones para conseguir el **Objetivo 1**.

Análogamente, como contribuciones del **Objetivo 2**, se utilizan los grafos de interacción y las técnicas de predicción de enlaces para proponer métodos de generación de explicaciones para recomendadores. Estas propuestas están planteadas y evaluadas en el **Capítulo 5**, donde se compara la efectividad de los métodos descritos con la de un sistema de explicación basado en contenido, que es uno de los más utilizados en la literatura.

En el **Capítulo 6** proponemos un método de visualización interactiva basada en grafos para explicar recomendaciones. La visualización basada en grafos permite a los usuarios entender de manera sencilla por qué una recomendación se ha producido, incrementando a su vez la satisfacción y la confianza en el sistema recomendador. En este capítulo, además de describir el método de visualización basado en grafos y el sistema en el que está integrado, se detalla la evaluación que se llevó a cabo con usuarios reales. Este capítulo está englobado en las contribuciones del **Objetivo 3**.

Por último, en el **Capítulo 7**, se describen las conclusiones obtenidas del trabajo realizado y de los resultados conseguidos para cada objetivo de

la tesis doctoral. También se definen las líneas de trabajo futuro para cada uno de ellos.

1.3. Publicaciones

Los resultados obtenidos en esta tesis doctoral han dado lugar a algunas publicaciones. A continuación, se enumeran estas publicaciones en orden cronológico:

1. CARO-MARTINEZ, M. y JIMENEZ-DIAZ, G. Similar users or similar items? comparing similarity-based approaches for recommender systems in online judges. En *International Conference on Case-Based Reasoning*, páginas 92–107. Springer, 2017. Índice de calidad Congreso CORE2017 B. Categoría *Artificial Intelligence and Image Processing*.
2. CARO-MARTINEZ, M. Recommender systems and explanations based on interaction graphs and link prediction techniques. *International Conference on Case-Based Reasoning*, página 204, 2018. Artículo perteneciente al *Doctoral Consortium*.
3. CARO-MARTINEZ, M., JIMENEZ-DIAZ, G. y RECIO-GARCIA, J. A. A theoretical model of explanations in recommender systems. *International Conference on Case-Based Reasoning*, página 52, 2018. Artículo perteneciente al *Workshop XCBR*.
4. CARO-MARTINEZ, M., RECIO-GARCIA, J. A. y JIMENEZ-DIAZ, G. An algorithm independent case-based explanation approach for recommender systems using interaction graphs. En *International Conference on Case-Based Reasoning*, páginas 17–32. Springer, 2019b. Índice de calidad Congreso CORE2020 B. Categoría *Artificial Intelligence*.
5. JORRO-ARAGONESES, J., CARO-MARTINEZ, M., RECIO-GARCIA, J. A., DIAZ-AGUDO, B. y JIMENEZ-DIAZ, G. Personalized case-based explanation of matrix factorization recommendations. En *International Conference on Case-Based Reasoning*, páginas 140–154. Springer, 2019. Índice de calidad Congreso CORE2020 B. Categoría *Artificial Intelligence*.
6. DIAZ-AGUDO, B., CARO-MARTINEZ, M., RECIO-GARCIA, J. A., JORRO-ARAGONESES, J. y JIMENEZ-DIAZ, G. Explanation of recommenders using formal concept analysis. En *International Conference on Case-Based Reasoning*, páginas 33–48. Springer, 2019. Índice de calidad Congreso CORE2020 B. Categoría *Artificial Intelligence*.
7. CARO-MARTINEZ, M., JIMENEZ-DIAZ, G. y RECIO-GARCIA, J. A. A survey of techniques for the evaluation of explanations in recommender

- systems. *International Conference on Case-Based Reasoning*, 2019a. Artículo perteneciente al *Workshop XCBB*.
8. JORRO-ARAGONESES, J. L., CARO-MARTÍNEZ, M., DÍAZ-AGUDO, B. y RECIO-GARCÍA, J. A. A user-centric evaluation to generate case-based explanations using formal concept analysis. En *International Conference on Case-Based Reasoning*, páginas 195–210. Springer, 2020a. Índice de calidad Congreso CORE2020 B. Categoría *Artificial Intelligence*.
 9. CARO-MARTINEZ, M., JIMENEZ-DIAZ, G. y RECIO-GARCIA, J. A. Conceptual Modeling of Explainable Recommender Systems: An Ontological Formalization to Guide Their Design and Development. *Journal of Artificial Intelligence Research*, vol. 71, páginas 557–589, 2021. Índice de calidad JCR 2020: 2.776. 67/140 (Q2). Categoría *Computer Science - Artificial Intelligence*.
 10. CARO-MARTINEZ, M., JIMENEZ-DIAZ, G. y RECIO-GARCIA, J. A. Local model-agnostic explanations for black-box recommender systems using interaction graphs and link prediction techniques. *International Journal of Interactive Multimedia and Artificial Intelligence*, Pendiente de publicar. Índice de calidad JCR 2020: 3.137. 58/140 (Q2). Categoría *Computer Science - Artificial Intelligence*.

Además, se incluyen a continuación una lista de las publicaciones que presentan algunos de los principales resultados logrados en esta tesis. Éstas se encuentran en estado de revisión en distintas revistas con índice de calidad de alto impacto en el momento de escritura de esta memoria:

1. CARO-MARTINEZ, M., JIMENEZ-DIAZ, G. y RECIO-GARCIA, J. A. A graph-based approach for minimizing the knowledge requirement of explainable recommender systems. Pendiente de aceptar.
2. CARO-MARTINEZ, M., JORRO-ARAGONESES, J. L., DIAZ-AGUDO, B., JIMENEZ-DIAZ, G. y RECIO-GARCIA, J. A. UPIEM: A personalized and interactive explanation method for matrix factorization recommender systems. Pendiente de aceptar.

Por último, se han inscrito los derechos de propiedad intelectual de la ontología obtenida como resultado de esta tesis doctoral en el *Registro Territorial de la Propiedad Intelectual*. A continuación, se describen los detalles de la inscripción:

- **Título:** ExRecOnto.
- **Autores:** Marta Caro Martínez, Guillermo Jiménez Díaz, Juan Antonio Recio García.

- **Objeto de propiedad intelectual:** programa de ordenador.
- **Nº de asiento registral:** 16/2020/1981.
- **Nº de solicitud:** M-7198-19.
- **Fecha de solicitud:** 18/10/2019.
- **Fecha de concesión:** 29/05/2020.

Capítulo 2

Estado del arte

Hoy en día, la cantidad de información a la que podemos acceder es prácticamente infinita debido al desarrollo de las nuevas tecnologías y en especial de Internet. Además, han aparecido plataformas de compra-venta online de productos como eBay¹ y Amazon², o de consumo de entretenimiento como Netflix³ o Spotify⁴, gracias a las cuales, tenemos una amplia oferta de productos que consumir, lo que supone una ventaja para los usuarios. Sin embargo, es tal la oferta de productos que los usuarios pueden tener problemas a la hora de encontrar aquellos productos que de verdad se ajusten a sus necesidades y requisitos (Aggarwal, 2016). Los sistemas de recomendación surgen como solución para resolver la difícil tarea que puede suponer tomar una decisión sobre el consumo o adquisición de un producto en Internet. Ayudan a los usuarios a encontrar productos o experiencias que se adecuen a sus preferencias y necesidades. Los sistemas de recomendación están ampliamente implantados en las plataformas de consumo de las que se ha hablado anteriormente debido a su demostrada utilidad (Kumar et al., 2021).

A pesar de ello, los sistemas de recomendación no siempre son tan útiles como cabría esperar. Muchas veces los usuarios no entienden cómo funciona el sistema recomendador, o no comprenden cuál es la razón de que se les haya recomendado un determinado producto. De ahí que no confíen en el sistema, y se reduzca el uso del recomendador como herramienta de apoyo (Nunes y Jannach, 2017). Para resolver la falta de confianza de los usuarios, han aparecido los sistemas de explicación. Este tipo de sistemas tratan de justificar la recomendación que se le ha otorgado a un determinado usuario, incrementando a su vez el interés por la recomendación (Arrieta et al., 2020).

En este capítulo, como resultado del Objetivo 0 de esta tesis –*estudio*

¹<https://www.ebay.es/>

²<https://www.amazon.es/>

³<https://www.netflix.com/>

⁴<https://www.spotify.com/>

de los sistemas de recomendación y técnicas de explicación aplicables—, se hace una revisión de la literatura sobre los sistemas recomendadores (Sección 2.1) y los sistemas de explicación para recomendadores (Sección 2.2). Se analiza el funcionamiento de estos sistemas, sus características y tipos, aportando trabajos anteriores como ejemplos de lo que se expone. Debido a que en esta tesis doctoral se usan los grafos como herramienta para generar recomendaciones y explicaciones, en este capítulo también se revisa la literatura en cuanto al uso de grafos en sistemas recomendadores (Sección 2.1.2) y de explicación (Sección 2.2.3). Además, se hace un estudio de técnicas de predicción de enlaces en la Sección 2.3, ya que se usan junto a los grafos en esta tesis doctoral. Al final del capítulo, en la Sección 2.4, se extraen unas conclusiones de la revisión realizada.

2.1. Sistemas de recomendación

Se define como sistema de recomendación al sistema inteligente que es capaz de relacionar un usuario con un determinado producto de forma personalizada (Jannach et al., 2010). El objetivo principal de los sistemas recomendadores es encontrar las dependencias o relaciones R que existen entre las interacciones que realizan los usuarios U y las que se producen sobre los productos I . De manera general, podemos especificar una función que defina esta relación como $f : U \times I \rightarrow R$ (Zafarani et al., 2014). Encontrando las relaciones existentes entre usuarios y productos, los sistemas de recomendación son capaces de hacer predicciones sobre: (1) los *ratings* con los que los usuarios van a valorar nuevos productos; o (2) las interacciones que van a producirse en el futuro entre usuarios y productos, encontrando cuáles son los mejores productos a recomendar para un usuario o cuáles son los usuarios que van a consumir un determinado producto. Gracias a estas predicciones, estos sistemas generan recomendaciones para ayudar al usuario a encontrar qué productos, de entre los ofertados, son los más adecuados según sus necesidades, preferencias y/o restricciones. También pueden ayudar a que un usuario que esté interesado en un determinado producto acabe de convenirse sobre si adquirir/consumir el producto o no. Es fundamental que en las recomendaciones se dé una cierta variabilidad de productos, ya que los gustos de los usuarios no son estáticos. Sin embargo, también es importante, dependiendo del dominio del problema, repetir recomendaciones, o productos con los que el usuario ya haya interactuado, como puede ser en compras de alimentos (Schafer et al., 2007).

Para realizar estas tareas, los sistemas recomendadores disponen de los datos que las plataformas poseen sobre los productos que ofrecen, como sus descripciones o propiedades, o la información que recogen sobre los usuarios (Kumar et al., 2021). Esta información sobre los usuarios puede ser explícita o implícita. Los datos **explícitos** son aquellos que los usuarios faci-

litan al sistema de forma voluntaria. Por ejemplo, los *ratings* son una fuente de datos explícita. Sin embargo, los datos **implícitos** son aquellos que el sistema recoge de forma automática a partir de las acciones que el usuario realiza con la plataforma. Para ello, es necesario monitorizar de alguna forma las acciones, por ejemplo guardando las interacciones que realiza el usuario con los productos: haciendo click sobre un producto, comprándolo, midiendo el tiempo que pasa mirándolo, etc. Por lo tanto, este tipo de datos son más fáciles de obtener, ya que no necesitan la intervención proactiva del usuario (Varga, 2019). Existe una clasificación de los tipos de recomendadores basada en el tipo de conocimiento que utilizan para generar las recomendaciones que está muy extendida en la literatura y que se expone a continuación.

2.1.1. Tipos de sistemas de recomendación

Debido a la extensa investigación que se ha llevado a cabo sobre sistemas recomendadores, existe una amplia variedad de los mismos (Aggarwal, 2016). Hay algunos sistemas que muestran información no personalizada a los usuarios, por ejemplo, una plataforma que recomiende películas podría mostrar al usuario la película más vista en la plataforma. Sin embargo, las mejores recomendaciones son las personalizadas, aquellas que se adaptan a cada usuario (Ricci et al., 2015), y es por eso que la investigación en este campo se ha centrado en este tipo de recomendadores (Bobadilla et al., 2013; Karimi et al., 2018). Según la literatura, podemos englobar estos algoritmos de recomendación personalizada en tres grandes grupos: filtrado colaborativo, basados en contenido y basados en conocimiento (Aggarwal, 2016; Bobadilla et al., 2013; Lu et al., 2015). Los sistemas de recomendación basados en **filtrado colaborativo** son aquellos que utilizan los *ratings* que los usuarios realizan sobre los productos del sistema. Los **basados en contenido**, sin embargo, utiliza el conocimiento de las descripciones de los productos y los perfiles de los usuarios. En cambio, si los algoritmos de recomendación utilizan conocimiento externo o requisitos que imponen los propios usuarios, entonces estas técnicas se engloban dentro del grupo de los recomendadores **basados en conocimiento**. Cada uno de estos tipos de algoritmos extraen el conocimiento de distintas fuentes, por lo que van a funcionar de diferente forma. En la Tabla 2.1 se muestra un resumen de las diferencias que podemos encontrar entre los tres grupos principales de sistemas de recomendación. Además, nos podemos encontrar con un cuarto grupo: los sistemas de recomendación **híbridos**, que pueden combinar dos o más algoritmos de entre los anteriores. A continuación se describe cada uno de estos tipos.

Tipo de recomendador	Objetivos del sistema	Input
Filtrado colaborativo	Proporciona recomendaciones calculadas a partir de las valoraciones e interacciones de todos los usuarios (incluido el usuario objetivo) con los productos del sistema.	<i>Ratings</i> del usuario + valoraciones del resto de usuarios
Basado en contenido	Proporciona recomendaciones basándose en las descripciones o atributos de los productos del sistema, favoreciendo los gustos del usuario objetivo.	Interacciones del usuario + atributos de los productos
Basado en conocimiento	Proporciona recomendaciones en base a las restricciones del usuario sobre el conjunto de los productos del sistema.	Restricciones del usuario + atributos de los productos + conocimiento del dominio

Tabla 2.1: Comparación de las principales características de los tres tipos de sistemas recomendadores: filtrado colaborativo, basado en contenido y basado en conocimiento. Tabla inspirada en el trabajo de (Aggarwal, 2016).

2.1.1.1. Filtrado colaborativo

Los sistemas de recomendación basados en filtrado colaborativo, que tienen en cuenta las valoraciones de los usuarios para generar recomendaciones, son posiblemente las técnicas de recomendación más populares (Liu y Lee, 2010). Las valoraciones de los usuarios (U) sobre los productos (I) se representan en una matriz M de filas y columnas, que corresponden a U e I , donde la celda $M[u, i]$ contiene el valor del *rating* que el usuario $u \in U$ ha otorgado para el producto $i \in I$. El objetivo de los sistemas de recomendación basados en filtrado colaborativo es completar esta matriz de *ratings*, prediciendo los valores de las celdas vacías. Para ello, los valores contenidos en M son utilizados con el objetivo de encontrar similitudes entre usuarios o productos, lo cual se fundamenta en que los *ratings* de usuarios con gustos parecidos serán semejantes (Aggarwal, 2016). Sin embargo, el filtrado colaborativo también se puede usar como metodología para generar recomendaciones cuando no existen *ratings*. Abordaremos este tipo de soluciones en la Sección 2.1.3.

Existen dos tipos de métodos de recomendación basados en filtrado colaborativo (Kumar et al., 2021; Aggarwal, 2016; Ricci et al., 2015; Schafer et al., 2007; Ekstrand et al., 2011; Herlocker et al., 2004; Massa y Avesani, 2004; Sarwar et al., 2001; Koren y Bell, 2015):

- Los métodos **basados en memoria** o basados en vecinos. Estos métodos tienen en cuenta las relaciones entre los productos y los usuarios. A su vez, se pueden dividir en dos métodos distintos: los modelos basados en productos y los modelos basados en usuarios (Kumar et al., 2021; Aggarwal, 2016).

- Los **basados en productos** predicen el *rating* de un producto i para un usuario objetivo u teniendo en cuenta los *ratings* que ha otorgado u a productos vecinos de i . Los productos vecinos de i son aquellos productos que los usuarios han valorado de forma similar a i .
- Los **basados en usuarios** predicen el *rating* de un producto i para un usuario objetivo u teniendo en cuenta los *ratings* que los usuarios vecinos de u han otorgado a i . En este caso, los usuarios vecinos de u se definen como los usuarios que han valorado de forma similar los mismos productos que u .
- Los métodos **basados en modelos**. Estos métodos son muy complejos pero, a su vez, consiguen resultados excelentes. Estos métodos engloban técnicas de aprendizaje automático y minería de datos como modelos predictivos (Kumar et al., 2021). Algunos ejemplos de estos métodos son los árboles de decisión o naive Bayes. Sin embargo, la técnica más popular es la factorización de matrices. Esta técnica utiliza la matriz M para encontrar factores latentes, representados por vectores numéricos, que caracterizan las preferencias de los usuarios y las propiedades de los productos del sistema (Koren et al., 2009; Bokde et al., 2015).

Al ser el filtrado colaborativo la técnica más popular a la hora de hacer recomendaciones, en la literatura se puede encontrar gran cantidad de trabajos, especialmente clásicos, que estudian en profundidad esta técnica (Kumar et al., 2021; Aggarwal, 2016; Ricci et al., 2015; Jannach et al., 2010; Su y Khoshgoftaar, 2009; Ekstrand et al., 2011; Jalili et al., 2018; Chen et al., 2018; Hameed et al., 2012). Además podemos encontrar otros muchos trabajos que proponen nuevos métodos de recomendación utilizando de una forma u otro el filtrado colaborativo. Por ejemplo, podemos encontrar el trabajo de Zheng et al. (2010), donde se utiliza esta para realizar recomendaciones sobre nuevos lugares a los que ir o actividades que realizar. En esta publicación se plantea que el GPS es una herramienta muy útil para viajar, pero sin embargo sólo extrae información de un único usuario, lo que no resulta de utilidad para realizar recomendaciones de actividades turísticas. Se pretende resolver este problema con la información acumulada de dispositivos GPS de distintos usuarios para implementar el sistema de recomendación basado en filtrado colaborativo UCLAF (*User-centered Collaborative Location and Activity Filtering*). También, podemos encontrar en el trabajo de Linden et al. (2003) distintas propuestas de sistemas de recomendación para implantar en Amazon, analizando cuál de ellas funciona mejor a la hora de adaptarse al continuo crecimiento de los datos de los que el sistema recomendador hace uso. Entre otras, podemos encontrar filtrado colaborativo basado en vecinos de productos y filtrado colaborativo basado en modelos, usando clustering, como algunas de las propuestas. Según el estudio realizado, el recomenda-

dor basado en filtrado colaborativo usando los vecinos de los productos es el algoritmo que mejor se adapta al problema a resolver en Amazon. En los artículos (Miyahara y Pazzani, 2000; Su y Khoshgoftaar, 2006) se propone un método de filtrado colaborativo basado en modelos: la técnica de aprendizaje automático naive Bayes. En ambos estudios, se llega a la conclusión de que esta técnica obtiene mejor precisión que el filtrado colaborativo basado en memoria. En la literatura también podemos encontrar numerosos trabajos que hacen hincapié en el uso de la factorización de matrices como sistema de recomendación. Un ejemplo es la propuesta de Ma et al. (2008), donde se presenta un método basado en factorización de matrices para implementar recomendaciones en redes sociales. Con este método se consigue obtener una precisión mejor que con otras técnicas, especialmente cuando se produce el problema de la dispersión, que consiste en la reducción de la precisión del recomendador al no tener suficiente información en la matriz de interacciones M . Hablaremos más adelante de este problema en la Sección 2.1.4.

Actualmente, aunque se siguen desarrollando propuestas basadas únicamente en filtrado colaborativo, las tendencias en investigación de este algoritmo también contemplan propuestas híbridas, combinando filtrado colaborativo con otras metodologías. Veremos algunos ejemplos en la Sección 2.1.1.4.

2.1.1.2. Recomendadores basados en contenido

Los sistemas de recomendación basados en contenido son aquellos que tienen en cuenta los atributos de los productos y los perfiles de los usuarios para generar las recomendaciones (Kumar et al., 2021; Aggarwal, 2016; Ricci et al., 2015; Jannach et al., 2010; Lops et al., 2011; Iaquina et al., 2008; Lu et al., 2015). Los atributos de los productos pueden ser descripciones, unas pocas palabras clave o etiquetas que definan las propiedades de los productos. Los perfiles de los usuarios contienen sus preferencias recogidas de sus interacciones anteriores (tanto si la interacción incluye otorgar un *rating* como si no). Por ejemplo, en un sistema de recomendación de películas, si el usuario objetivo ha visto la película “El gran showman”, el sistema puede guardar en su perfil la propiedad “musical” como una preferencia del usuario. De esta forma, el recomendador es capaz de encontrar similitudes entre los atributos de los productos y el perfil del usuario (Aggarwal, 2016). Siguiendo con el ejemplo anterior, el sistema puede analizar los productos de la plataforma para obtener aquellas películas con la propiedad “musical”, que el usuario aún no haya visto, como posibles recomendaciones.

Los sistemas de recomendación basados en contenido, al usar únicamente las descripciones de los productos y los perfiles de los usuarios sin depender de los valores de los *ratings*, se pueden aplicar en escenarios en los que implementar filtrado colaborativo no es posible o beneficioso. Por ejemplo, cuando hay dispersión en los datos o con productos nuevos de la plataforma, sobre los que no se tengan valoraciones. Son especialmente útiles donde hay una

gran riqueza descriptiva de los productos, como puede ser la web semántica. Sin embargo, esto también puede tener sus desventajas ya que extraer la información que pueda ser de utilidad real para el recomendador puede ser costoso si el dominio es grande o complejo. Además, los sistemas basados en contenido también tienen problemas de rendimiento cuando el usuario objetivo es nuevo en el sistema, ya que al no haber realizado interacciones pasadas, no se ha podido crear un perfil de usuario adecuado. Este problema es denominado como *cold start* (Ricci et al., 2015). Otro problema que se puede dar con el uso de este tipo de sistemas es la falta de diversidad en las recomendaciones. Los sistemas basados en contenido capturan los gustos de los usuarios, generando las recomendaciones en base a esto, sin tener en cuenta algún margen de variabilidad en sus necesidades. Abordaremos estos problemas más en profundidad en la Sección 2.1.4.

Un ejemplo de sistema de recomendación basado en contenido es el trabajo de Soares y Viana (2015). Aquí, se propone un método de recomendación que utiliza metadata para construir el perfil del usuario y las descripciones de los productos, incidiendo en cómo el uso de distintos elementos metadata puede incrementar la utilidad de las recomendaciones. Se hace además una comparación con técnicas de filtrado colaborativo teniendo en cuenta el nivel de detalle de las descripciones y la dispersión de los datasets a la hora de evaluar ambos métodos. Los autores concluyen que las técnicas de filtrado colaborativo obtienen mejores resultados en general, pero dependiendo de los elementos metadata empleados, el sistema basado en contenido puede llegar a alcanzar la misma precisión. Esto hace de este tipo de sistemas una buena alternativa cuando el dataset es muy disperso o el coste computacional de usar filtrado colaborativo es muy alto. Otro ejemplo de este tipo de sistemas es la propuesta de Phelan et al. (2011), donde se presenta un recomendador para descubrir nuevas noticias extrayendo información de Twitter⁵. En este caso, se construye el perfil del usuario a partir de las interacciones implícitas del usuario. Esto significa que el usuario no valora el contenido que publica en Twitter, pero la propia publicación del contenido se considera como interacción del usuario. El contenido que publica el usuario es el que se utiliza para construir su perfil. Las descripciones de los productos a recomendar, que en este caso son noticias, se pueden extraer de distintas fuentes: del *feed* o tablón público de Twitter, del *feed* de las personas que sigue, del archivo RSS del usuario (que almacena sus suscripciones) y de los archivos RSS de toda la comunidad de Twitter. En un trabajo reciente (Addagarla y Amalanathan, 2021) encontramos un sistema de recomendación que utiliza características visuales para encontrar similitudes entre los objetos de recomendación, lo cual es un ejemplo novedoso teniendo en cuenta que la mayoría de los sistemas basados en contenido representan el contenido de forma textual, usando características inherentes al producto.

⁵<https://twitter.com/>

El razonamiento basado en casos o *case-based reasoning* (CBR) es una técnica de inteligencia artificial que propone soluciones utilizando una base de casos como fuente de conocimiento. En esta base de casos se almacenan soluciones junto a los problemas que resuelven. A esta tupla problema-solución se le llama experiencia o caso. El sistema consulta la base de casos para extraer una solución de un problema similar al que se intenta resolver, adaptando la solución al escenario actual (Kolodner, 2014; Aamodt y Plaza, 1994). Los sistemas de recomendación basados en casos son un tipo específico de recomendadores basados en contenido, y están fundamentados en el CBR (Lops et al., 2011; Quijano Sánchez, 2015). En los sistemas de recomendación basados en casos, la base de casos contiene las descripciones de los posibles productos a recomendar, mientras que en el perfil de usuario se guardan las descripciones de los productos que el usuario ha valorado con anterioridad. De esta forma, se recuperan casos (productos) similares a productos que le han gustado el usuario objetivo previamente. Existe una gran cantidad de propuestas en la literatura donde se aplica CBR como método de recomendación. Por ejemplo, Jorro-Aragoneses et al. (2017a) proponen un sistema recomendador contextual de planes turísticos. La base de casos representa las descripciones de los productos (actividades turísticas), mientras que el perfil del usuario se construye con sus preferencias pasadas. Además, el sistema es capaz de filtrar la solución de manera que se adapte a las restricciones contextuales del usuario. Sin embargo, el trabajo en (Dong et al., 2013), utiliza reseñas de los usuarios para recomendar productos interesantes. De las reseñas se consiguen características que describen los productos a recomendar y además los sentimientos (positivos, negativos o neutros) que esos productos provocan en los usuarios. Tanto las características como los sentimientos son los atributos descriptivos de los productos que se almacenan en la base de casos.

2.1.1.3. Recomendadores basados en conocimiento

Los sistemas recomendadores basados en conocimiento son aquellos que no tienen en cuenta las valoraciones o interacciones de los usuarios, como sí hacían los dos tipos de algoritmos anteriores. En este caso se tienen en cuenta las restricciones del usuario objetivo, las descripciones de los productos e información del dominio (Aggarwal, 2016). Permitiendo a los usuarios especificar sus restricciones, estos sistemas son capaces de otorgar recomendaciones más personalizadas (Ruiz Iniesta, 2014). Además resuelven algunos de los problemas que poseen los recomendadores anteriores. Los problemas de la dispersión, del *cold start* y del *long tail*, que surge cuando se recomiendan siempre los mismos productos (los más populares), no aparecen en este caso ya que el conocimiento que se utiliza no depende de las interacciones previas de los usuarios (Aggarwal, 2016; Ruiz Iniesta, 2014). Por tanto, los sistemas de recomendación basados en conocimiento son especialmente

adecuados cuando: (1) no hay información suficiente sobre *ratings* o interacciones anteriores de los usuarios; y (2) el usuario quiere decirle al sistema qué tipo de producto quiere adquirir o consumir. A pesar de sus ventajas, su uso también tiene inconvenientes que ya nos encontramos en los sistemas basados en contenido y que están principalmente relacionados con la adquisición de conocimiento. El procesamiento de la información del dominio puede ser complejo en función de su amplitud. También nos encontramos el problema de la sobreespecialización: al determinar las preferencias de forma explícita, el sistema se encuentra trabajando sobre el mismo conocimiento, por lo que las recomendaciones van a tener poca variabilidad. Por último nos encontramos un nuevo problema, que hasta ahora no habíamos visto: el problema del conjunto vacío. Este problema surge cuando no hay recomendaciones que satisfagan las restricciones impuestas por el usuario (Ruiz Iniesta, 2014).

En general los sistemas basados en conocimiento funcionan a través de un **conjunto de reglas** con las que delimitan las posibles soluciones de recomendación en base a las restricciones del usuario. Sin embargo, también se podrían incluir los sistemas **basados en casos** como un tipo de sistema basado en conocimiento si es el usuario quien le dice al sistema cuáles son los productos a los que quiere que se parezcan las recomendaciones (Bridge et al., 2005; Aggarwal, 2016).

Aunque los recomendadores basados en filtrado colaborativo y basado en contenido son los más populares, también podemos encontrar numerosos ejemplos en la literatura de recomendadores basados en conocimiento. Por ejemplo, hallamos un estudio clásico en la propuesta de Zhang et al. (2008) que indica que los recomendadores basados en conocimiento se llevan estudiando bastante años. Zhang et al. (2008) propone un método de visualización para permitirle al usuario indicar al sistema recomendador basado en críticas qué atributos son interesantes para él o ella. Otro ejemplo lo tenemos en la publicación de (Li et al., 2018) donde se propone un chatbot que pregunta a los usuarios sobre sus gustos cinematográficos para recomendarles películas. Por último, en el trabajo de Yang (2010) encontramos un sistema recomendador de artículos de investigación y otros recursos para investigadores y profesores universitarios que hace uso de una ontología como fuente de conocimiento del dominio.

2.1.1.4. Híbridos

Hemos visto anteriormente que cada tipo de recomendador tiene unas debilidades que los otros no tienen y viceversa. En la Sección 2.1.4 se hace una revisión más extensa de estos problemas. Los sistemas de recomendación híbridos surgen como método de solución a estos problemas, combinando distintas técnicas (o las mismas) para intentar mitigar los inconvenientes de cada recomendador (Kumar et al., 2021; Aggarwal, 2016).

Según Aggarwal (2016) existen tres posibles diseños de sistemas de reco-

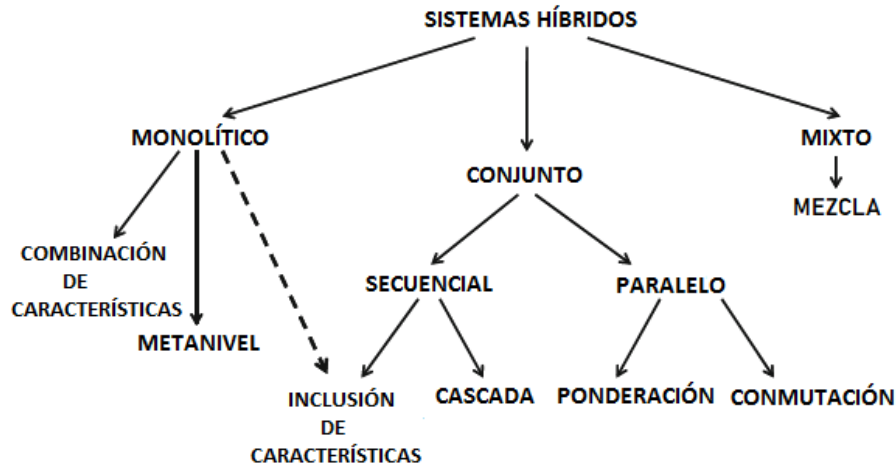


Figura 2.1: Clasificación de las estrategias de combinación de métodos recomendadores para obtener sistemas híbridos. Figura inspirada en el trabajo de (Aggarwal, 2016).

mendación híbridos: *ensemble design* o diseño en conjunto, *monolithic design* o diseño monolítico y *mixed systems* o sistemas mixtos. Se hace esta clasificación en función de los *inputs* usados, los *outputs* obtenidos y los modelos de recomendación que actúan en el sistema. Dentro de cada tipo de sistema híbrido, nos podemos encontrar distintas estrategias para combinar recomendadores (Burke, 2002; Çano y Morisio, 2017) (véase la Figura 2.1).

El **diseño en conjunto** combina varias técnicas de recomendación sobre un único *input* para obtener un único *output*. Dentro de este tipo de sistemas híbridos podemos encontrar dos tipos de estructuras: el **diseño en paralelo** (Figura 2.2) y el **diseño secuencial** (Figura 2.3). El diseño en paralelo combina todos los resultados obtenidos de la aplicación de cada modelo recomendador sobre el *input* para obtener un sólo *output*. Dentro de este diseño, nos podemos encontrar las siguientes técnicas:

Ponderación (*weighted*). Este proceso combina las puntuaciones obtenidas por cada método de recomendación, pudiendo tener cada método un peso distinto en esta combinación. El producto con una puntuación más alta será el recomendado.

Conmutación (*switching*). El sistema de recomendación empleará un algoritmo u otro dependiendo de las necesidades a resolver. Por ejemplo, se podría usar un algoritmo de filtrado colaborativo si hay suficiente información sobre los *ratings* de los productos, o bien un algoritmo basado en conocimiento si queremos evitar el problema del *cold start*.

El **diseño secuencial**, sin embargo, aplica un algoritmo recomendador

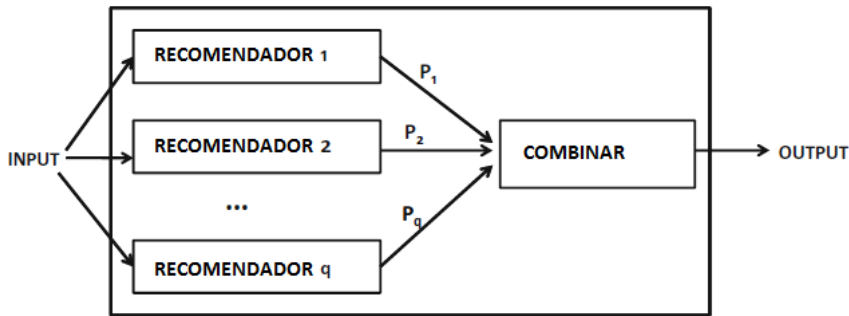


Figura 2.2: Esquema de diseño paralelo de un sistema recomendador híbrido. Figura inspirada en el trabajo de (Aggarwal, 2016).

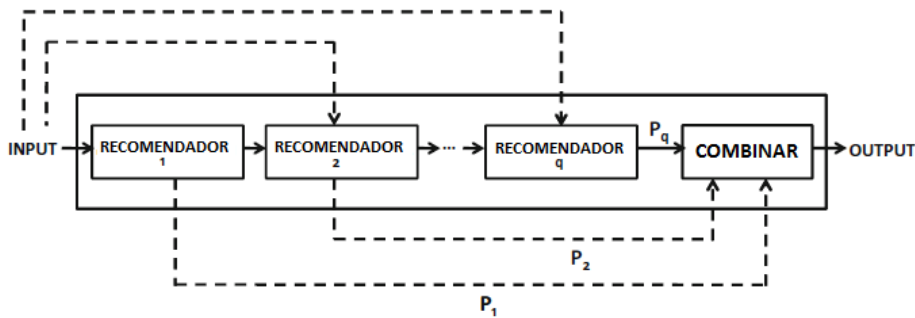


Figura 2.3: Esquema de diseño secuencial de un sistema recomendador híbrido. Figura inspirada en el trabajo de (Aggarwal, 2016).

sobre el *input*, para después aplicar el siguiente recomendador sobre el *output* obtenido del primer modelo. El proceso continúa de forma sucesiva hasta aplicar todos los modelos de recomendación. Al terminar, se combinan todos los *outputs* para elaborar un único resultado. Las técnicas que nos podemos encontrar dentro de este diseño secuencial son:

Inclusión de características (*feature argumentation*). El *output* que genera un método de recomendación, el cual va a ser una característica nueva sobre los productos, va a ser integrado en el *input* que utilizará el siguiente método recomendador.

Cascada (*cascade*). Esta técnica refina las recomendaciones de forma sucesiva: un método recomendador refina la solución otorgada por el anterior. Este proceso se lleva a cabo aplicando un método de recomendación sobre un conjunto de datos, y utilizando el conjunto de datos resultante de este proceso como el *input* para el siguiente recomendador.

Por otro lado, en el **diseño monolítico** se utiliza un algoritmo de recomendación sobre varios *inputs* para obtener un sólo *output*. Las técnicas para llevar a cabo el diseño monolítico son:

Combinación de características (*feature combination*). Se combinan los conjuntos de datos utilizados por todos los métodos de recomendación para obtener un único *input*. A partir de este conjunto de datos combinado se aplica el método de recomendación.

Metanivel (*meta-level*). Un modelo recomendador es usado como *input* del siguiente modelo. El ejemplo más típico es la combinación entre filtrado colaborativo y basado en contenido. En este caso, el filtrado colaborativo se modifica para usar conocimiento basado en contenido, creando grupos de pares de productos que tiene descripciones similares. Después se usa la matriz de *ratings* junto con los grupos de pares para generar las recomendaciones.

Inclusión de características (*feature argumentation*). Aunque en la mayoría de los casos esta técnica se usa como diseño secuencial, también puede aplicarse en un diseño monolítico.

Finalmente, en los **sistemas mixtos** nos encontramos la ejecución de varios algoritmos de recomendación sobre un *input*. La diferencia con el diseño en conjunto es que se obtienen varios *outputs*. Nos podemos encontrar la siguiente técnica:

Mezcla (*mixed*). Todas las recomendaciones obtenidas por diferentes métodos de recomendación son presentada a la vez al usuario.

Actualmente, las líneas de investigación en sistemas de recomendación se centran en el desarrollo de sistemas híbridos para intentar paliar las diferentes debilidades que tienen los algoritmos de recomendación principales (Çano y Morisio, 2017). Como consecuencia, podemos encontrar gran cantidad de ejemplos de trabajos de sistemas híbridos, siendo la combinación entre filtrado colaborativo y basado en contenido una de las mezclas preferidas en la literatura. Por ejemplo, el trabajo de Hannon et al. (2010) propone un sistema híbrido que mezcla filtrado colaborativo y contenido para recomendar usuarios a los que seguir en redes sociales como Twitter. El método puede obtener las recomendaciones para un usuario a partir de las palabras incluidas en sus propios *tweets*, en los *tweets* de sus seguidores o en los *tweets* de las personas a las que sigue (recomendaciones basadas en contenido). Sin embargo, el sistema también genera recomendaciones basadas en filtrado colaborativo generando el perfil del usuario con los usuarios a los que sigue y/o con sus seguidores. Ambas fuentes de conocimiento se mezclan para obtener distintas recomendaciones. En la publicación de Alshammari et al. (2017)

también nos encontramos un sistema híbrido basado en filtrado colaborativo y contenido. El sistema cuenta con dos módulos, uno basado en filtrado colaborativo, que se utiliza cuando hay suficientes *ratings* con los que implementar este algoritmo. El basado en contenido se utiliza en caso contrario, para aliviar el problema del *long tail* del que sufre el filtrado colaborativo. El módulo basado en filtrado colaborativo es un sistema CBR que utiliza los *ratings* para construir la base de casos. En esa base de casos se representa cada usuario por los *ratings* con los que ha valorado los productos. De esta forma, el sistema es capaz de encontrar los usuarios más similares al usuario objetivo. Usando este conocimiento, se predice el *rating* que el usuario va a otorgar a un producto a través de la media ponderada de los *ratings* de los usuarios similares. Véase que en este caso el sistema CBR no funciona con un sistema recomendador basado en contenido, como se usa tradicionalmente, sino como una herramienta para generar filtrado colaborativo. El segundo módulo es el basado en contenido. En este módulo CBR se construye la base de casos caracterizando las películas según sus géneros. Siguiendo esta base, se puede encontrar la película más similar a aquella que queremos recomendar. Se calcula el *rating* predicho de igual forma que en el módulo anterior. Otro ejemplo de sistema de recomendación híbrido lo encontramos en el método propuesto por Riyahi y Sohrabi (2020), que recomienda nuevos temas o preguntas en grupos de discusión. El primer paso que realiza este método es encontrar los usuarios más similares al usuario objetivo teniendo en cuenta sus interacciones implícitas. Usando la información de las interacciones que han realizado el usuario objetivo y sus usuarios similares se puede predecir la preferencia de un usuario por un *post*. El módulo basado en contenido usa los etiquetas de los *posts* para encontrar similitudes. Los *outputs* de ambos módulos se combinan para generar las recomendaciones.

2.1.2. Recomendadores basados en grafos

Aunque los sistemas de recomendación basados en grafos no se incluyen en la clasificación general vista anteriormente, en esta sección estudiamos este tipo específico de recomendadores por su interés en esta tesis doctoral.

Un grafo es una estructura computacional para representar las entidades de un sistema y las relaciones que existen entre esas entidades. Los grafos guardan un gran cantidad de conocimiento de manera estructurada que resulta de gran utilidad en la resolución de problemas sobre múltiples dominios y campos, como la inteligencia artificial. De ahí, el incremento del interés en investigar sobre ellos (Ji et al., 2021; Chicaiza y Valdiviezo-Díaz, 2021; Liu y Duan, 2021). Entre otras aplicaciones reales, nos encontramos el uso de grafos de conocimiento como herramienta para modelar sistemas de recomendación (Ji et al., 2021; Guo et al., 2020; Chicaiza y Valdiviezo-Díaz, 2021; Liu y Duan, 2021). Al integrar información sobre usuarios, productos y otros atributos del sistema, los grafos pueden aportar un razonamiento que

ayuda a los recomendadores a disminuir problemas tradicionales como los de dispersión y el *cold start* (Ji et al., 2021). Además pueden aportar una interpretabilidad a las recomendaciones de la que otras técnicas clásicas de recomendación carecen (Ji et al., 2021; Guo et al., 2020; Wang et al., 2020).

Según la literatura (Guo et al., 2020; Aggarwal, 2016; Huang et al., 2007), podemos encontrar tres tipos de grafos para generar recomendaciones: los grafos de productos, los grafos de usuarios o los grafos de usuarios-productos. En el grafo de productos y en el grafo de usuarios, los nodos representan los productos o los usuarios del sistema, respectivamente. Los enlaces representan la relación que tienen los nodos. Por ejemplo, en un grafo de productos, un enlace entre dos nodos puede representar la marca que ambos productos comparten. En los grafos de usuarios-productos existen dos tipos de nodos: los usuarios y los productos, representando un enlace que un usuario ha interactuado con un producto o le ha gustado ese producto. A partir de esta clasificación, se pueden definir tres tipos de métodos recomendadores (Guo et al., 2020): los basados en incrustación (*embeddings*), que tienen en cuenta el conocimiento representado de las entidades; los basados en conexiones, que tienen en cuenta la información representada sobre las relaciones de las entidades; y los basados en propagación, que mezclan ambos tipos de conocimiento.

Los métodos **basados en *embeddings*** enriquecen el grafo con conocimiento para representar con el mayor nivel de detalle a los usuarios y los productos del sistema. Con esta representación se pueden predecir las preferencias que tienen los usuarios por los productos. Estos métodos tienen en cuenta el contenido de las entidades, sus descripciones y relaciones semánticas. En el método propuesto por Wang et al. (2018b) para generar recomendaciones de noticias, se representa texto de las propias noticias en el grafo de conocimiento (*word embeddings*). El perfil del usuario se construye con el texto de las noticias con las que ha interactuado, de forma que se puedan calcular las similitudes entre el perfil del usuario y las entidades representadas en el grafo.

Los métodos **basados en conexiones** utilizan los patrones de conexión de los nodos (cómo están enlazados en base a las interacciones producidas) en el grafo para encontrar similitudes entre las entidades representadas. El tipo de grafo que se usa en este caso es el de usuarios-productos. Los patrones de conexión se pueden a su vez representar en vectores para encontrar las recomendaciones. En el método de Shi et al. (2015) se obtiene la predicción de *ratings* de un usuario objetivo teniendo en cuenta la suma ponderada de los *ratings* de usuarios similares. Los usuarios similares se calculan a través de las conexiones que existen entre los usuarios representados en el grafo. En el trabajo propuesto por Huang et al. (2007) se estudia la topología de grafos bipartitos aleatorios uti-

lizando métodos probabilísticos para generar recomendaciones.

Los métodos **basados en propagación** surgen para integrar el conocimiento de los dos anteriores y así no perder información, incluyendo el conocimiento de las entidades y el de las conexiones del grafo en base al historial de interacciones. En el método propuesto por Wang et al. (2018a) se representa en el grafo el perfil del usuario objetivo en base a las características de los productos que le han gustado. En función de ese grafo se utiliza la información del historial de los usuarios para generar predicciones de las futuras interacciones que realizará el usuario. Otro ejemplo sería la reciente propuesta de Xie et al. (2021), en la que se representa la información en un grafo de conocimiento bipartido donde los nodos representan a los usuarios y a los productos del sistema. Los nodos usuarios y productos se conectan cuando al usuario le gusta ese producto. Sin embargo, los productos también están enlazados si tienen alguna propiedad en común. El *framework* obtiene recomendaciones utilizando este grafo de conocimiento teniendo en cuenta la inclusión de diversidad en las recomendaciones.

2.1.3. Conocimiento usado por los sistemas de recomendación

Ya hemos comentado anteriormente los dos tipos de interacciones que se pueden utilizar para generar las recomendaciones: las explícitas, en las que los usuarios facilitan información de forma voluntaria al sistema, y las implícitas, que son aquellas que el usuario realiza con el sistema sin darle conocimiento de forma intencionada. Teniendo en cuenta estos dos tipos de interacciones, y la información que el usuario aporta al sistema, podemos definir varios tipos de *ratings*: continuos, basados en intervalos, ordinales, binarios y unarios (Aggarwal, 2016).

Con los *ratings* continuos, los usuarios pueden detallar el nivel de preferencia por un producto a través de una escala con valores continuos. Con este tipo de *ratings*, los usuarios pueden valorar un producto que hayan consumido dando al sistema información sobre el nivel de preferencia del usuario con ese producto. Los *ratings* basados en intervalos y los ordinarios son el tipo de puntuaciones que nos encontramos más usualmente en los sistemas recomendadores. La diferencia entre ellos es que los basados en intervalos permiten al usuario dar un valor discreto al producto, mientras que los ordinarios muestran el intervalo de preferencia a través de categorías. Dependiendo del sistema, el rango de los valores puede variar, siendo el rango de 1 (no me gusta nada) a 5 (me gusta mucho) el intervalo de valores más común (por ejemplo en Amazon o Goodreads⁶). Los *ratings* binarios son aquellos en los que el usuario muestra preferencia, la cual se representaría con un valor 1,

⁶<https://www.goodreads.com>

Usuario	Coco	Jurassic Park	Seven	Regreso al futuro
U1	4		5	
U2		3	4	5
U3	3	5		5
U4		2	2	5

(a) Ejemplo de *ratings* basados en intervalos.

Usuario	Coco	Jurassic Park	Seven	Regreso al futuro
U1	1		1	
U2		0	1	1
U3	0	1		1
U4		0	0	1

(b) Ejemplo de *ratings* binarios.

Usuario	Coco	Jurassic Park	Seven	Regreso al futuro
U1	1		1	
U2			1	1
U3		1		1
U4				1

(c) Ejemplo de *ratings* unarios.Tabla 2.2: Comparativa de tipos de *ratings*: basados en intervalos, binarios y unarios.

o desagrado, representado por 0, por un producto. El último tipo de *ratings* es el unario donde se especifica solamente la interacción con el producto: 1 si se ha producido interacción y \emptyset , si no se ha producido. Los valores de preferencia y desagrado por tanto quedan indefinidos. En este caso, los usuarios pueden otorgar al sistema los *ratings* unarios de forma explícita o implícita (por ejemplo comprando un producto o viendo el producto). El resto de tipos de *ratings* deben ser otorgados de forma explícita por el usuario. En la Tabla 2.2 se muestra un ejemplo de las diferencias entre los distintos tipos de *ratings*.

Los modelos de filtrado colaborativo que se han estudiado en la Sección 2.1.1.1 alcanzan su mayor rendimiento a través del uso de *ratings* continuos y basados en intervalos. Sin embargo, los modelos de filtrado colaborativo se pueden adaptar en escenarios de menor cantidad de conocimiento, como es el caso de los *ratings* binarios o unarios (Aggarwal, 2016). La razón de que exista esta posibilidad es que los mayoría de los algoritmos de filtrado colaborativo basados en modelos son algoritmos de clasificación (Aggarwal, 2016). Por ejemplo, los árboles de decisión, los métodos basados en reglas, naive Bayes, support vector machines y las redes neuronales, entre otras, son técnicas tradicionales de aprendizaje automático que permiten resolver el problema de clasificación y que también se pueden utilizar para implementar filtrado colaborativo (Su y Khoshgoftaar, 2009; Aggarwal, 2016).

En las técnicas de clasificación hace falta una matriz que sirva de fuen-

te de conocimiento. Esta matriz tiene filas y columnas que representan las instancias y las variables del sistema, respectivamente. Las variables pueden ser independientes o dependientes de los valores de otras variables independientes. El objetivo de las técnicas de clasificación es predecir el valor de las variables dependientes para unas determinadas instancias. Usando las técnicas de clasificación como filtrado colaborativo se emplea la matriz de *ratings* como fuente de conocimiento. De esta forma, se puede construir un modelo de predicción de *ratings* utilizando los valores de los *ratings* con los que otros usuarios (las instancias) han puntuado los distintos productos (las variables) que se pretende recomendar a un determinado usuario objetivo. Hace falta generar un modelo para cada usuario-producto, donde el usuario es el objetivo de la recomendación y el producto es la posible recomendación (el producto del que se quiere predecir el *rating*). Por tanto, la variable dependiente de este modelo serán los valores de los *ratings* de ese producto, mientras que las variables independientes corresponderán al resto de productos (Aggarwal, 2016). Como ya se ha comentado, lo interesante de estos métodos es que se pueden aplicar sobre escenarios donde no haya *ratings* con valores numéricos, ya que las variables son asimiladas por las técnicas de clasificación como categorías. Teniendo una matriz de *ratings* binarios donde tengamos únicamente la información de las interacciones de los usuarios se puede predecir si el usuario objetivo va a interactuar o no con el producto a recomendar. Si el modelo predice que el usuario objetivo va a interactuar con ese producto, entonces ese producto se le recomendará al usuario (Aggarwal, 2016).

Los modelos basados en vecinos también se pueden adaptar a *ratings* sin valores numéricos (Aggarwal, 2016), aunque dependiendo de la medida de similitud que se utilice, puede no ser posible adaptar este método a cualquier tipo de escenario (Jain et al., 2020).

2.1.4. Retos y nuevas tendencias de los sistemas recomendadores

Los sistemas de recomendación están ampliamente implantados hoy en día en múltiples plataformas de una gran diversidad de dominios. Surgieron principalmente en plataformas de adquisición de productos, siendo Amazon una de las primeras que empezó a usar sistemas recomendadores de forma comercial (Aggarwal, 2016). En el caso de este tipo de plataformas, el sistema recomendador usa tanto datos explícitos (*ratings*) como implícitos (interacciones de los usuarios) para generar las recomendaciones, debido a la gran cantidad de información que se puede obtener de ambas fuentes. Sin embargo, los sistemas de recomendación han evolucionado considerablemente y están actualmente asentados en casi cualquier dominio imaginable. Un ejemplo destacable es el sistema recomendador de Netflix, una plataforma de *streaming* de series, películas y documentales. Este sistema ha contribuido

Reto	Filtrado colaborativo	Contenido	Conocimiento
Cold start	X	X	
Long tail	X		
Dispersión	X		
Escalabilidad	X	X	X
Conjunto vacío			X
Sobreespecialización		X	X
Sin serendipia		X	X
Vulnerabilidad	X	X	X
Privacidad	X	X	X
Sin contexto	X	X	
Grupos	X	X	X
Vis. e int.	X	X	X
Explicabilidad	X	X	X

Tabla 2.3: Comparativa sobre los problemas y retos a abordar en los principales tipos de sistemas de recomendación, inspirada en el análisis de Soares y Viana (2015).

enormemente en el desarrollo de nuevos algoritmos recomendadores gracias al premio *Netflix Prize contest*, organizado por el propio Netflix, donde los participantes diseñaban sus algoritmos de recomendación (Bell y Koren, 2007; Bennett et al., 2007), siendo precisamente en el contexto de este concurso donde se desarrolló el algoritmo de factorización de matrices (Koren et al., 2009).

A pesar del uso extendido de estos sistemas y la gran cantidad de estudios científicos existentes, aún quedan retos que abordar en este campo de investigación, para algunos de los cuales se pueden encontrar propuestas de solución en la literatura. Aunque en las anteriores secciones ya se han dado unas pinceladas de algunos de los principales problemas que encontramos en los sistemas recomendadores, a continuación se describen con más detalle los retos a abordar (Ricci et al., 2015; Jannach et al., 2010; Aggarwal, 2016). En la Tabla 2.3 se muestran estos problemas y retos de acuerdo a cada tipo de recomendador.

Cold start. Este problema surge cuando hay productos o usuarios nuevos en un sistema. Al no tener información sobre los productos, no pueden ser recomendados. En el caso de los usuarios, ocurre que no pueden recibir recomendaciones personalizadas porque no se conocen sus gustos (Lika et al., 2014; Silva et al., 2019).

Long tail. Es el desequilibrio de la información sobre los productos en un

sistema recomendador: el sistema posee mucha información sobre unos pocos productos (los más populares), mientras que de la mayoría de los productos tiene poca información. Es un problema que se retroalimenta: aquellos productos con más datos son los más consumidos y los que más se recomendarán a otros usuarios. De esta forma, seguirá aumentando su cantidad de datos, sus consumiciones y sus recomendaciones sucesivamente (Park y Tuzhilin, 2008).

Dispersión. La dispersión se produce debido a que se realiza una pequeña proporción de interacciones entre los usuarios y los productos de entre todas las posibles combinaciones. La dispersión es uno de los principales problemas del filtrado colaborativo basado en memoria. Cuando no hay representada suficiente información en la matriz de interacciones \mathbb{M} , su precisión se reduce considerablemente. Sin embargo, los métodos basados en modelos consiguen mejores resultados que los anteriores cuando aparece el problema de la dispersión (Aggarwal, 2016).

Falta de diversidad o sobreespecialización. Significa que las recomendaciones que se le dan a un determinado usuario son muy similares entre sí (Thorat et al., 2015; Kumar et al., 2021; Soares y Viana, 2015). Por ejemplo, en un sistema de recomendación de películas, se le recomendarían al mismo usuario películas del mismo género. Como se ha comentado a lo largo del capítulo, es importante tener en cuenta los gustos del usuario. Sin embargo, los usuarios tienen gustos variados y dinámicos, pueden gustarle características en principio contrapuestas, y pueden cambiar con el tiempo. Siguiendo con el ejemplo de los géneros cinematográficos, a un usuario le puede gustar tanto el género de comedia como el drama. Puede que durante una época de su vida le gusten las películas de animación y luego no sienta tanta predilección por ese tipo de películas. Por lo tanto, los sistemas recomendadores deberían ofrecer elementos que encajen con varios de los gustos del usuario.

Escasez de serendipia. La serendipia es lo que se llamaría el efecto sorpresa en las recomendaciones. Los recomendadores deberían tener en cuenta las consecuencias positivas de incluir algunos elementos novedosos en las recomendaciones ya que, al estar siempre basadas en sus gustos, los usuarios pueden llegar a aburrirse. Por tanto no es sólo interesante que se incluya diversidad en las recomendaciones, sino también serendipia de forma que, de vez en cuando, alguna de las recomendaciones incluya elementos que no coincidan con las preferencias previas del usuario objetivo, e incluso elementos totalmente distintos a sus gustos (Kotkov et al., 2016; De Gemmis et al., 2015).

Escalabilidad. Cuando el conjunto de datos contiene mucha cantidad de

información, el coste computacional de generar las recomendaciones puede llegar a ser muy alto (Thorat et al., 2015; Kumar et al., 2021; Soares y Viana, 2015).

Problema del conjunto vacío. Este problema se da en los sistemas de recomendación donde los usuarios determinan las características de los productos que quieren consumir, como son los recomendadores basados en conocimiento. Puede darse el caso de que no haya, en el conjunto de los productos a recomendar, ninguno que cumpla las especificaciones dadas por el usuario objetivo, por lo que no se podría recomendar ningún producto (Ruiz Inieta, 2014).

Vulnerabilidad del sistema frente a ataques. Los sistemas de recomendación son una de las herramientas fundamentales en las plataformas de comercio electrónico. Es por este motivo que pueden ser objeto de ser manipuladas, por ejemplo siendo valorados algunos productos de forma positiva o negativa de acuerdo a los intereses del atacante (Thorat et al., 2015; Lü et al., 2012).

Privacidad. La recolección de datos de los usuarios es útil para el sistema recomendador, ya que cuanto más sepa sobre sus gustos, mejores recomendaciones será capaz de generar. Sin embargo, esto puede tener coste en la intimidad del usuario, especialmente si esos datos son usados para otros fines que nada tienen que ver con la obtención de las recomendaciones dentro de la plataforma (Caro Martínez, 2017).

No consideración de información contextual. La información de la situación en la que se encuentra el usuario objetivo de la recomendación es indispensable para generar recomendaciones adecuadas. Solucionar este problema o proponer metodologías para ello es una de las tendencias actuales en sistemas de recomendación (Jorro-Aragoneses et al., 2017a; Haruna et al., 2017).

Recomendaciones para grupos. También es una de las tendencias actuales en investigación de recomendadores. Intentar satisfacer las preferencias y necesidades de todos los usuarios, o al menos no desagradar a alguno de ellos, puede ser un reto complejo (Quijano Sánchez, 2015).

Interactividad y visualización. Los algoritmos de recomendación han sido estudiados en las últimas décadas, pero en los últimos años, una de las tendencias de investigación está centrada en el incremento de la satisfacción del usuario con el sistema. En este incremento influye en gran medida cómo se puede interactuar con el sistema y cómo se muestra la recomendación (He et al., 2016).

Falta de explicabilidad. Esta es una de las líneas de investigación más en auge actualmente debido a la necesidad de incrementar la confianza de

los usuarios en la inteligencia artificial. De este problema ha surgido la inteligencia artificial explicable (XAI), en la que también se engloba la inclusión de explicaciones en los sistemas recomendadores. En esta tesis doctoral se ha estudiado en profundidad esta línea de investigación. En la Sección 2.2 se hace un estudio de los sistemas de explicación en recomendadores.

2.2. Sistemas de explicación

En los últimos años, la inteligencia artificial (AI) ha experimentado un significativo auge. La aparición de la inteligencia artificial se produjo hace décadas con el desarrollo de las nuevas tecnologías, pero es ahora, con la necesidad de resolver grandes tareas muy complejas, que el uso de la inteligencia artificial se ha extendido en muchos sectores de la sociedad (West, 2018). El uso de la inteligencia artificial afecta a las vidas de las personas, y es aquí cuando las decisiones que han sido tomadas por un sistema inteligente necesitan ser entendidas por los usuarios, especialmente en contextos críticos, como podría ser la medicina o el sector aeroespacial (Goodman y Flaxman, 2017; Arrieta et al., 2020). Algunos de estos métodos son más fácilmente entendibles por los usuarios, pero la interpretabilidad se va haciendo cada vez menor con cada nuevo método que va apareciendo, más complejo que el anterior. Los sistemas inteligentes son vistos por los usuarios como cajas negras que no entienden, ni saben cómo funcionan: no son transparentes (Arrieta et al., 2020). El problema de la falta de transparencia de la inteligencia artificial es que las personas no confían en aquello que no comprenden, y como consecuencia, los usuarios son reticentes a usar cualquiera de las tecnologías que impliquen alguna inteligencia artificial que no comprendan. Este problema se puede mitigar haciendo más entendibles este tipo de sistemas.

La inteligencia artificial explicable (XAI) es el campo de la inteligencia artificial que intenta abordar este problema (Gunning et al., 2019). Según Arrieta et al. (2020), la XAI tiene como objetivo principal proponer un conjunto de técnicas que consigan hacer más entendibles para los usuarios los modelos de inteligencia artificial, sin perder rendimiento y aumentando la confianza en estos modelos, incrementando a su vez su utilidad.

La necesidad de los sistemas de explicación en sistemas recomendadores ya se ha comentado al inicio del capítulo. Los sistemas de recomendación son de una gran utilidad hoy en día, ayudando a los usuarios a encontrar aquellos productos que más se adaptan a sus necesidades y preferencias. Sin embargo, debido a que no entienden cómo se ha llegado a recomendar ese determinado producto, muchas veces los usuarios no hacen caso de las recomendaciones propuestas, disminuyendo así su utilidad. Los sistemas de explicación contribuyen a aumentar la confianza en el sistema, ayudando también a tomar mejores decisiones (Tintarev, 2007). Para diseñar sistemas de explicación

hay que tener en cuenta distintas características que las explicaciones pueden adoptar, dependiendo de qué se pretende lograr con la explicación, los objetivos de los usuarios, el contenido de la propia explicación, el tipo de recomendador que queramos explicar, o cómo queremos mostrar la explicación. En la literatura, existen diversos trabajos que hacen revisiones de todos estos aspectos, llegando a proponer distintas taxonomías que cubren las características de los sistemas de explicación en sistemas recomendadores y de toma de decisiones. Ejemplos de ello son los trabajos de Nunes y Jannach (2017), de Papadimitriou et al. (2012) o de Friedrich y Zanker (2011). El trabajo de Arrieta et al. (2020) también es un ejemplo de taxonomía de sistemas XAI en general. A continuación, se describen algunos de los conceptos principales que se utilizan para definir modelos de explicación en la literatura, con el objetivo de aclarar la terminología usada en esta tesis doctoral.

2.2.1. Características de los sistemas de explicación para recomendadores

En general, en la literatura podemos encontrar distintos conceptos que definen modelos de explicación para recomendadores. Unos de los que más se repiten son los conceptos de transparencia, interpretabilidad y explicabilidad, los cuales son intercambiables, aunque algunos autores matizan diferencias entre estos conceptos: (Gilpin et al., 2018; Arrieta et al., 2020; Zhang y Chen, 2020).

Transparencia. Un sistema es transparente si el usuario conoce cómo se lleva a cabo el proceso de recomendación (Tintarev, 2007). Por ejemplo, si un sistema basado en reglas le muestra al usuario las reglas que utiliza, entonces el sistema es transparente.

Interpretabilidad. Si un sistema de recomendación es interpretable el usuario entiende las razones detrás de una recomendación (Arrieta et al., 2020). Por ejemplo, siguiendo el ejemplo del sistema basado en reglas, si las reglas utilizadas por el sistema son entendibles por el usuario, entonces el sistema es interpretable. En cambio, si las reglas que utiliza el sistema son muy complicadas para el usuario objetivo, entonces no es interpretable, aunque siga siendo transparente, ya que el usuario puede acceder al razonamiento llevado a cabo por el sistema, pero no entenderlo.

Explicabilidad. Conlleva un paso más allá de la interpretabilidad. Cuando un sistema es explicable, los usuarios entienden el modelo de recomendación, aceptan la recomendación y confían en el sistema. Por tanto la explicabilidad del sistema afecta a la toma de decisiones del usuario. Cualquier sistema explicable es interpretable, pero no ocurre lo mismo a la inversa (Gilpin et al., 2018). En el caso del ejemplo anterior, si

el sistema de reglas es interpretable, y además el usuario confía en el sistema aceptando las reglas que se le muestran como explicación del proceso llevado a cabo, entonces el sistema será explicable.

Recientemente se han definido varios tipos de modelos de explicación (Buhrmester et al., 2019; Ribeiro et al., 2018; Arrieta et al., 2020; Ribeiro et al., 2016; Singh y Anand, 2019; Arya et al., 2019). Los conceptos que definen los tipos de modelos son ampliamente usados actualmente en la literatura.

Desde el punto de vista de la transparencia, se puede definir un sistema recomendador como una caja negra (*black box*), cuando el sistema no es transparente, o como una caja blanca (*white box*), cuando sí es transparente. De acuerdo con esto, un sistema de explicación que apoye a un recomendador puede definirse como:

Ante hoc, cuando el sistema de explicación utiliza el razonamiento utilizado por el recomendador, que es una caja blanca, para generar la explicación.

Post hoc, cuando el sistema de explicación es totalmente independiente del recomendador, ya que es una caja negra y no se puede conocer su funcionamiento.

Según la cantidad de conocimiento que necesite el sistema de explicación, será más o menos interpretable. Se pueden definir dos tipos de modelos:

Local. Si el sistema es local, utiliza una parte del total del conocimiento al que puede acceder, de forma que el usuario puede interpretar más fácilmente una determinada predicción, ya que sólo necesita entender una determinada predicción del modelo para una entrada específica.

Global. Un modelo global utiliza todo el conocimiento disponible para generar una explicación o recomendación. Debido a la cantidad de conocimiento y la dificultad que puede conllevar entender todas las posibles predicciones que se pueden producir, los métodos globales son menos interpretables que los locales.

Dependiendo de la cantidad de recomendadores que pretende explicar el sistema de explicación, se pueden especificar dos tipos de modelos:

Model-specific. El modelo de explicación trata de hacer más comprensible un modelo recomendador específico.

Model-agnostic. El modelo de explicación sirve para explicar cualquier sistema de recomendación.

En el Capítulo 3, hacemos una amplia revisión del estado del arte sobre sistemas de explicación, ampliando los conceptos principales que se han visto

hasta el momento. Como resultado de este estudio, se ha propuesto un nuevo modelo conceptual para clasificar sistemas de explicación en recomendadores, que refina e incluye nuevos conceptos.

2.2.2. Sistemas de explicación para recomendadores

Aunque en el Capítulo 3 definimos una extensa taxonomía para clasificar sistemas de explicación para recomendadores, en general en la literatura encontramos que estos sistemas se pueden englobar en dos tipos principales, teniendo en cuenta la fuente de conocimiento que utilizan. Estas dos principales fuentes de conocimiento son los *ratings* y las descripciones de los productos, al igual que ocurría con los sistemas de recomendación, de manera que vamos a tener dos tipos principales de sistemas de explicación para recomendadores: los basados en filtrado colaborativo y los basados en contenido (Zhang y Chen, 2020).

2.2.2.1. Explicaciones basadas en filtrado colaborativo

El uso del filtrado colaborativo en este caso es análogo al de los sistemas recomendadores. Tanto los métodos basados en vecinos como los basados en modelos se utilizan en la literatura para generar explicaciones. El sistema de explicación obtiene las explicaciones de acuerdo al conjunto de vecinos que el sistema recomendador encuentra para obtener la lista de recomendaciones (Zhang y Chen, 2020). Por ejemplo, en el sistema de recomendación basados en usuarios, el algoritmo encuentra el conjunto de usuarios más similares (sus vecinos) al usuario objetivo. Esta información es la utilizada en sistemas de explicación clásicos como fuente de conocimiento. De este modo, se le puede explicar al usuario que usuarios similares a él o ella han valorado con buenas puntuaciones los productos recomendados (Zhang y Chen, 2020; Herlocker et al., 2000). Análogamente, se pueden extraer explicaciones del algoritmo de filtrado colaborativo basado en productos. A partir del conjunto de productos más similares a aquellos que le han gustado al usuario previamente se obtienen las recomendaciones. En consecuencia, se le explica al usuario que las recomendaciones son productos similares a aquellos que ha valorado positivamente en el pasado (Zhang y Chen, 2020; Sarwar et al., 2001). Según Herlocker et al. (2000), las explicaciones basadas en productos son más útiles, ya que los usuarios están familiarizados con los productos con los que han interactuado previamente. Conocen las características de esos productos, resultando las explicaciones más creíbles desde su punto de vista. Sin embargo, las explicaciones basadas en usuarios pueden reducir la confianza en los sistemas de recomendación, ya que no conocen quiénes son estos usuarios más similares a ellos.

Un estudio clásico en el que se proponen explicaciones basadas en filtrado colaborativo es el trabajo de Herlocker et al. (2000). Los autores proponen

distintas formas de visualizar cómo el algoritmo de filtrado colaborativo genera las recomendaciones. Por ejemplo, proponen una visualización en la que se muestra un gráfico con los *ratings* que los usuarios similares al usuario objetivo han otorgado al producto recomendado. El filtrado colaborativo también es un algoritmo que se puede utilizar para explicar por qué un producto no ha sido recomendado, como es el caso del algoritmo WNCf (*Why-Not in Collaborative Filtering*) (Stratigi et al., 2020). En el trabajo de Wang et al. (2018c), se describe una nueva propuesta denominada *Tree-enhanced Embedding Method* (TEM). Esta propuesta utiliza modelos basados en factores latentes y modelos basados en árboles de decisión para proporcionar recomendaciones. Los autores afirman que el proceso es explicable debido a que las reglas que aplica el algoritmo son transparentes para los usuarios.

2.2.2.2. Explicaciones basadas en contenido

Las explicaciones basadas en contenido son aquellas que utilizan las similitudes entre las características de los productos recomendados y los atributos que se pueden encontrar en el perfil del usuario objetivo (Zhang y Chen, 2020). Las características de los productos se pueden obtener directamente de las descripciones de los propios productos o de otras fuentes, como pueden ser las reseñas de los usuarios o información publicada en redes sociales. Igualmente el perfil del usuario se puede construir a partir de sus interacciones implícitas o explícitas, o del contenido que comparte en la plataforma, con otros usuarios, etc. (Zhang y Chen, 2020).

Un ejemplo de sistema de explicación basado en contenido, lo encontramos en el trabajo clásico de Gedikli et al. (2014). Además de estudiar distintas propuestas de explicación, entre las que se encuentran las ya mencionadas propuestas por Herlocker et al. (2000), propone un método que muestra una nube de etiquetas al usuario. Esa nube de etiquetas es un conjunto de características del producto recomendado. Se pueden mostrar de forma personalizada o no personalizada. La diferencia está en que en la personalizada se muestran las etiquetas en colores, dependiendo de las preferencias del usuario. Si una determinada característica del producto no concuerda con las preferencias del usuario, esa etiqueta se mostrarán en color rojo. En cambio, si esa característica sí es de su interés, se mostrará en azul, y en otro caso, se mostrará gris. Por tanto, nos encontramos ante un ejemplo tradicional en el que el contenido usado es el obtenido a partir de las descripciones de los productos. Sin embargo, en el sistema propuesto por Muhammad et al. (2016), se utilizan las opiniones de los usuarios como el contenido que va a describir los productos y los perfiles de los usuarios. Se trata de un sistema CBR que extrae el contenido de las reseñas de los usuarios para construir la base de casos. En el sistema de recomendación planteado por Wang et al. (2016), se utiliza un algoritmo de filtrado colaborativo en el que, en lugar de predecir los *ratings* de las películas a recomendar, se predicen los géneros

que le van a gustar al usuario. El método incluye un modelo de explicación que usa análisis de sentimientos de las reseñas de los usuarios. Otro ejemplo de sistema de explicación que emplea el análisis de sentimientos de las reseñas de los usuarios es *Explicit Factor Model* (EFM) (Zhang et al., 2014). Utiliza también las características de los productos en combinación con los sentimientos extraídos de las reseñas. El trabajo de Quijano-Sanchez et al. (2017) propone el sistema PSIE (*Personalized Social Individual Explanation*) que explica las decisiones de sistemas de recomendación para grupos también basándose en contenido. Su objetivo principal es mejorar la satisfacción del usuario, la eficiencia percibida y la confianza. En este caso, se utiliza la descripción de la personalidad del usuario y de las relaciones entre los miembros del grupo para generar las explicaciones. Además, propone dos métodos de visualización: uno textual y otro gráfico.

Al igual que ocurría con los sistemas de recomendación, los sistemas de explicación pueden ser sistemas híbridos, combinando distintos algoritmos para intentar paliar las desventajas que cada uno de ellos conlleva. Un ejemplo lo tenemos en la propuesta de Symeonidis et al. (2009), *MoviExplain*, que incluye explicaciones generadas a partir del proceso de recomendación. Utiliza los *ratings* del usuario objetivo, junto a las descripciones de los productos para obtener estas explicaciones. Es capaz de decirle al usuario que se le ha recomendado una película en base a que tiene una determinada característica y esta característica es compartida con otras películas que le han gustado.

2.2.3. Sistemas de explicación para recomendadores basados en grafos

En recientes estudios (Ji et al., 2021; Guo et al., 2020) sobre el uso de grafos para resolver problemas complejos, como puede ser la obtención de recomendaciones, se hace hincapié en la capacidad de los grafos para aportar explicabilidad. Estos trabajos ponen de manifiesto que la integración de grafos de conocimiento en los sistemas recomendadores puede mejorar el razonamiento detrás de la recomendación y, por tanto, su interpretabilidad, siendo los sistemas de recomendación basados en grafos más comprensibles que las técnicas clásicas. De esta forma, los grafos de conocimiento pueden ser interpretables por los usuarios objetivo, lo que puede producir un incremento de la confianza del usuario en el sistema recomendador y de la propia utilidad del sistema. Podemos establecer tres razones por las que los grafos son más interpretables que otras técnicas clásicas (Ji et al., 2021; Guo et al., 2020; Lecue, 2020):

- **Los grafos se acercan al modelo mental del usuario.** La explicabilidad en los grafos está asociada a la propagación de la información en esta estructura computacional, ya que los usuarios pueden interpretar

esta propagación como descubrimiento de la información.

- **La visualización de la explicación es más directa desde los modelos de recomendación basados en grafos que desde otros métodos.** Con los grafos podemos mostrar directamente las conexiones y relaciones inferidas desde la propia causa de la recomendación. En otras técnicas como las de aprendizaje automático no es tan evidente la transformación del proceso de recomendación a una visualización explicable, siendo en ocasiones, como en la redes neuronales una tarea bastante compleja.
- **El proceso de recomendación es explícito para los usuarios.** Los caminos y patrones de preferencias de los usuarios son más observables e identificables: las razones de la recomendación se pueden ver e interpretar siguiendo la secuencia de razonamiento del grafo.

Además de aportar interpretabilidad en sistemas de recomendación, los grafos también pueden ser una herramienta de generación de explicaciones por sí misma, independientemente de si utilizan la información de los *ratings* o de las descripciones de los productos como fuente de conocimiento. Según Guo et al. (2020), hay cuatro maneras distintas de obtener explicaciones en métodos de recomendación basados en grafos:

1. **Encontrando las preferencias del usuario.** En los grafos, se pueden representar los distintos atributos de los productos recomendados y encontrar cuáles son las preferencias de los usuarios con respecto a esos atributos. Determinando qué preferencias tienen más peso para el usuario objetivo, se obtienen las explicaciones.
2. **Definiendo secuencias de conexiones.** Se pueden encontrar los distintos caminos o los distintos subgrafos que relacionan un usuario con un producto recomendado o con el que ha interactuado. Se obtienen las explicaciones traduciendo estas estructuras en reglas entendibles para los usuarios.
3. **Utilizando el peso de los caminos.** Obteniendo cuáles son los caminos con más peso entre el usuario objetivo y los productos recomendados, se pueden obtener explicaciones.
4. **Utilizando el peso de los enlaces.** Con este método se utilizan únicamente los pesos de los enlaces que unen los nodos del grafo con sus vecinos.
5. **Usando aprendizaje por refuerzo.** Aplicando aprendizaje por refuerzo, una técnica de aprendizaje automático (Sutton y Barto, 2018), se puede deducir el proceso de razonamiento del recomendador.

Podemos encontrar algunos ejemplos de sistemas de explicación para recomendadores en la literatura. Por ejemplo, en el trabajo de Barbieri et al. (2014), se propone el sistema de explicación WTFW (*Who to Follow and Why*) que utiliza grafos y técnicas de predicción de enlaces para explicar recomendación de contactos en redes sociales. El sistema no sólo predice un enlace, sino que predice qué tipo de enlace se está formando: uno temático o uno social. Por un lado, un enlace temático se formará entre dos usuarios cuando un usuario está interesado en un determinado tema y el otro usuario es un experto en ese tema. La explicación en este caso es un conjunto de características que describen el tema en cuestión. Por otro lado, un enlace social se creará entre dos usuarios cuando comparten un conjunto grande de contactos. La explicación en este caso es el conjunto de contactos que son los más probables de ser responsables de la recomendación. Xian et al. (2019) describen un nuevo método llamado *Policy-Guided Path Reasoning* (PGPR), que utiliza un grafo de conocimiento, donde se representan las relaciones de los usuarios y los productos para generar recomendaciones. PGPR es un sistema de caja blanca: tiene en cuenta los caminos reales en el grafo para crear recomendaciones explicables. También hay que mencionar el trabajo de Wang et al. (2019), que presenta el nuevo modelo *Knowledge-aware Path Recurrent Network* (KPRN). KPRN también utiliza grafos de conocimiento para hacer recomendaciones que son interpretables. En este caso, en el grafo de conocimiento se representan relaciones directas e indirectas entre un usuario y un producto, lo que significa que existen uno o varios pasos en el camino entre ese usuario y ese producto. El camino se forma cuando un usuario ha interactuado con un producto. En el grafo se encuentran similitudes entre ese producto y otros productos a recomendar, en base a características similares de los productos. Esta información se usa tanto para generar la recomendación, como la explicación.

Sin embargo, cuando se trata de generar explicaciones para sistemas de recomendación, los grafos también se pueden usar como método de visualización (Brath y Jonker, 2015). Existe una infinidad de formas de mostrar datos a través de grafos, teniendo en cuenta multitud de factores. De forma intuitiva, podemos ver que se pueden utilizar los componentes del grafo para visualizar datos, como pueden ser los nodos, los enlaces, los caminos existentes, los componentes conectados, los *clusters* (conjuntos de nodos cercanos), o agrupaciones (conjuntos de nodos relacionados) (Lee et al., 2006). Además, teniendo en cuenta aquello que queramos mostrar, se pueden cambiar las características del componente (tamaño, color, forma), representar los grafos con distintas disposiciones o *layouts* (por ejemplo, radial, en árbol, en cono o hiperbólico), distintas formas de navegar o interactuar con el grafo (por ejemplo, zoom u ojo de pez), diferentes formas de mostrar propiedades (espaciales, temporales), etc. (Herman et al., 2000; Beck et al., 2017; Chen et al., 2015; Hadlak et al., 2015; Kwon et al., 2016). Podemos encontrar varios

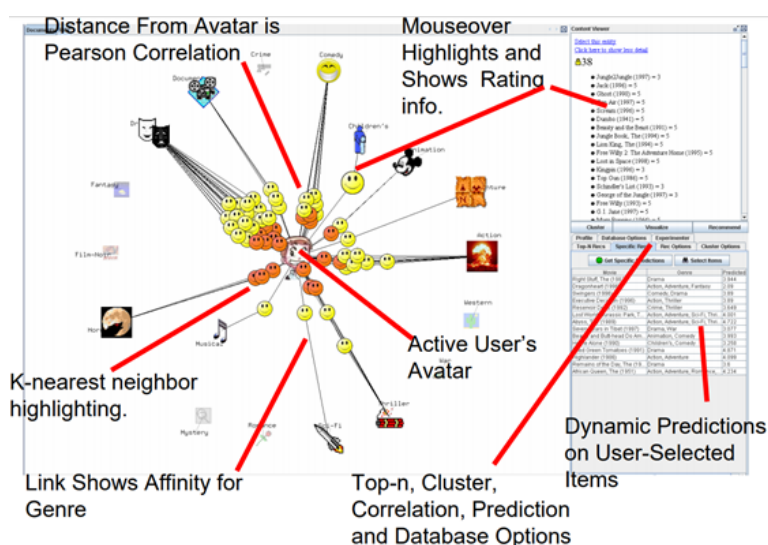


Figura 2.4: Ejemplo de la visualización de una explicación proporcionada por el sistema PeerChooser para una recomendación. Imagen extraída del trabajo de O'Donovan et al. (2008).

ejemplos sobre sistemas de explicaciones para recomendadores que visualizan las explicaciones usando grafos. Por ejemplo, en LinkedIn se muestra una explicación de una recomendación de un contacto mediante un grafo sencillo. Los nodos son el usuario objetivo y el usuario recomendado como contacto. Existen tantos enlaces entre ambos usuarios como características hay en común entre ambos y en cada enlace se muestra cada característica (Brath y Jonker, 2015). Otro ejemplo donde se usa una visualización basada en grafos es el sistema Moodplay (Andjelkovic et al., 2019), que permite a los usuarios buscar recomendaciones de música. La interfaz interactiva de Moodplay permite a los usuarios descubrir nueva música jugando con el grafo de explicación, además de escuchar las recomendaciones en la propia interfaz o acceder a información adicional en Last.fm⁷. Además, pueden ver donde surge la música recomendada en función de las características de la música y los sentimientos que genera, gracias a los colores del grafo. Otro ejemplo lo encontramos en PeerChooser (O'Donovan et al., 2008) (Figura 2.4). En este trabajo se propone una interfaz de visualización que explica las recomendaciones de películas para un sistema recomendador basado en filtrado colaborativo. Utiliza una representación en grafo de las correlaciones entre los usuarios. El usuario objetivo se encuentra en el centro de la visualización mientras que los vecinos del usuario objetivo se encuentran conectados al usuario objetivo mediante enlaces. La longitud del enlace al usuario objetivo está basada en la similitud entre los *ratings* de ambos usuarios. A su vez, esos

⁷<https://www.last.fm/es/>

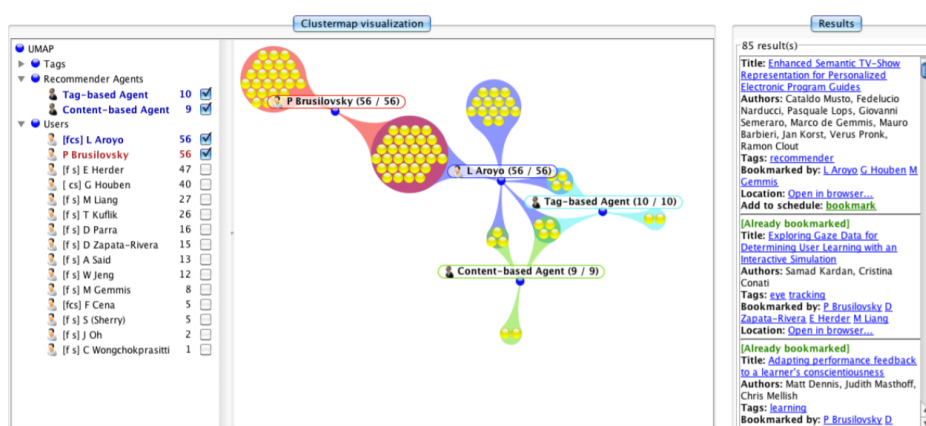


Figura 2.5: Ejemplo de la visualización proporcionada por el sistema TalKExplorer de una explicación para una recomendación. Imagen extraída del trabajo de Verbert et al. (2013).

usuarios vecinos están unidos, mediante otros enlaces, a las preferencias de esos usuarios por los géneros de películas. Al pinchar sobre estos enlaces se pueden ver las recomendaciones de películas y las características de esas películas. En el trabajo de Kangasrääsio et al. (2015) se nos presenta el sistema Intent Radar, un sistema de explicación para motores de búsqueda. Muestra en un *layout* circular las 10 palabras clave más relacionadas a la búsqueda realizada por el usuario. Cuanto más cerca esté la palabra clave del centro, más relevante es esa palabra en la búsqueda. Además, otras palabras clave adicionales se muestran fuera del *layout*. El usuario puede dar *feedback* al modelo moviendo las palabras clave a una nueva localización dentro del *layout*. Cuando el usuario suelta la palabra clave en una nueva posición, el modelo del usuario es entonces actualizado teniendo en cuenta la distancia al centro de esa palabra clave. Así, se muestran nuevos resultados de la búsqueda al usuario, además de una nueva visualización de la explicación. En el trabajo de Parra et al. (2014) nos encontramos un ejemplo donde se usan diagramas de Venn para explicar recomendaciones de artículos científicos. Estos diagramas representan las relaciones entre conjuntos a partir de círculos que se superponen. En este caso, existen tres círculos, donde se engloban tres tipos de artículos: artículos más veces marcados como favoritos, artículos similares a los artículos favoritos del usuario objetivo y los artículos más citados. Estos círculos se superponen, de forma que un artículo que sea de los más citados y similar a uno favorito del usuario pertenecerá a ambos conjuntos. En el centro del diagrama, parte en la cual se solapan los tres círculos, estará el artículo recomendado. Los tamaños de estos círculos pueden modificarse por el usuario objetivo a través de *sliders*. Los usuarios también pueden acceder a información adicional, como resúmenes de los artículos o ver más recomen-

daciones. También se pueden encontrar en la literatura grafos que visualizan la información usando *clusters*, como es el caso de TalkExplorer (Verbert et al., 2013), un sistema que recomienda conferencias y artículos (Figura 2.5). Cada burbuja representa un conjunto de conferencias. El punto que une las burbujas es la propiedad que existe en común entre las conferencias. Esta propiedad puede ser que las conferencias tratan los mismos temas, o temas del mismo campo, o que un mismo usuario ha indicado que ambas conferencias son de su interés. Los usuarios también pueden interactuar con la interfaz haciendo click sobre esas entidades para ver más detalles. Por ejemplo, si pincha sobre un *cluster*, podrá ver la lista de conferencias o charlas dentro de ese conjunto. Como último ejemplo, nos encontramos el método de visualización de TasteWeights (Bostandjiev et al., 2012), un sistema de recomendación de grupos musicales que muestra a los usuarios una explicación de la recomendación teniendo en cuenta las características en común que hay entre los grupos que escucha y los recomendados. Esta información se puede obtener desde Wikipedia⁸, Facebook o Twitter. En la interfaz de explicación nos encontramos un grafo con tres columnas. En la primera columna nos encontramos con los nodos que representan los grupos musicales que le gustan al usuario. En la columna del medio, los nodos representan la descripción que sirve como explicación. En la tercera, están los nodos que representan las recomendaciones. Los enlaces se van a formar entre la primera columna y la segunda, y entre la segunda y la tercera, de forma que se pueda ver cuáles son las características en común en ambos grupos musicales. El usuario objetivo puede interactuar con el grafo, diciéndole al sistema qué fuente de datos es la más interesante a través de *sliders*.

2.3. Técnicas de predicción de enlaces

Las técnicas de predicción de enlaces (*link prediction*) son un conjunto de métodos del campo del análisis de las redes sociales que sirven para predecir nuevos enlaces que se van a formar en un grafo o para determinar enlaces que se han perdido (Liben-Nowell y Kleinberg, 2007; Lü y Zhou, 2011; Wang et al., 2015; Kumar et al., 2020). Como se ha comentado anteriormente, los grafos son estructuras que sirven para representar entidades de un sistema (biológico, social, etc.) y las relaciones que existen entre esas entidades (Wang et al., 2015; Caro Martínez, 2017). Analizar grafos es por tanto una tarea indispensable hoy en día. Sin embargo, puede ser muy compleja debido a que la información del sistema está incompleta en el grafo y a que los grafos son dinámicos: están en constante cambio por la aparición y desaparición de nodos y enlaces (Wang et al., 2015; Caro Martínez, 2017). Es por esto que predecir cómo va a evolucionar un grafo en el tiempo es uno de los principales problemas a abordar en el análisis de grafos (Lü y Zhou,

⁸<https://es.wikipedia.org/>

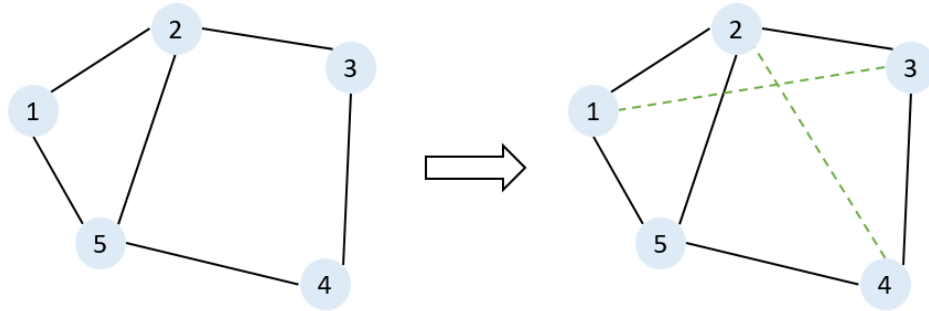


Figura 2.6: Ejemplo de predicción de enlaces en un grafo.

2011). Esta información se puede utilizar para conocer cómo va a evolucionar un sistema, por ejemplo, para predecir el desarrollo de una pandemia o el desarrollo de actividades económicas (Barabási, 2014). Las técnicas de predicción de enlaces surgen como herramienta para ayudar a solventar este problema (Liben-Nowell y Kleinberg, 2007; Lü y Zhou, 2011; Wang et al., 2015; Kumar et al., 2020).

Las técnicas de predicción de enlaces se han utilizado para resolver gran cantidad de problemas (Lü y Zhou, 2011; Wang et al., 2015; Kumar et al., 2020) como descubrimiento de información perdida o desconocida de un sistema (Guimerà y Sales-Pardo, 2009), detección de *spam* en correo electrónico (Huang y Zeng, 2006), aumento de la privacidad y confianza en redes sociales (Al-Oufi et al., 2011), identificación de referencias perdidas en publicaciones (Kc et al., 2009), localización de expertos y colaboraciones en el ámbito académico (Pavlov y Ichise, 2007), encaminamiento en redes (Weiss et al., 2003), etc. Las técnicas de predicción de enlaces también son una herramienta utilizada para generar recomendaciones (Wang et al., 2015; Kumar et al., 2020). Por ejemplo, se utilizan técnicas de predicción de enlaces sobre grafos que representan las preferencias o contenido generado por el usuario (Sadilek et al., 2012; Armentano et al., 2013). Como ya hemos visto, incluso se ha utilizado las técnicas de predicción de enlaces como método para generar explicaciones en recomendaciones (Barbieri et al., 2014).

Un grafo se define como $G = \langle N, L \rangle$, donde N es el conjunto de nodos, que representan las entidades del sistema, y L es el conjunto de enlaces, que unen los nodos si existe una relación entre esos nodos. Este grafo está representado para un determinado instante t . El objetivo de las técnicas de predicción de enlaces es conocer cómo va a ser el conjunto L en el instante t_a , siendo $t_a > t$. Se puede ver un ejemplo de predicción de enlaces en la Figura 2.6, inspirado en un ejemplo de Kumar et al. (2020).

Existen distintas formas de clasificar las técnicas de predicción de enlaces, pero en términos generales, se pueden dividir en dos principales grupos, según Wang et al. (2015): los métodos basados en similitud y los basados en

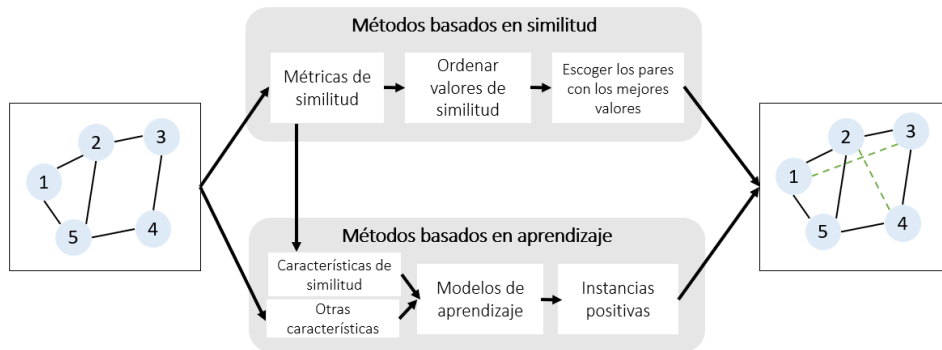


Figura 2.7: Diferencia de funcionamiento entre métodos basados en similitud y basados en aprendizaje. Imagen inspirada en el trabajo de (Aggarwal, 2016).

aprendizaje. Los **métodos basados en similitud** miden la semejanza que existe entre dos nodos no conectados y establecen un valor para esa similitud. Cuanto más alto sea este valor, más probabilidades habrá de que estos nodos se enlacen en un futuro (Lü y Zhou, 2011; Wang et al., 2015; Kumar et al., 2020). Los **métodos basados en aprendizaje** tratan el problema como una clasificación binaria, otorgando un valor positivo al par de nodos que se predice que se van a conectar en el futuro o un valor negativo si no se van a conectar (Wang et al., 2015). En la Figura 2.7 se muestran las diferencias de funcionamiento de ambos grupos de métodos, mientras que en la Figura 2.8 se puede observar un esquema de los distintos tipos de técnicas que existen para cada grupo de métodos, de los que se va a hablar a continuación.

A su vez, se pueden dividir los métodos basados en similitud en tres grandes grupos de medidas: las medidas basadas en nodos, las basadas en topología y las basadas en teorías sociales. Las **medidas basadas en nodos** son aquellas que determinan la similitud de dos nodos en base a las características de esos nodos. Su funcionamiento se basa en la teoría de la homofilia, que argumenta que las personas tienden a relacionarse con las que tienen algo en común, por ejemplo, intereses. Por el contrario, las **medidas basadas en topología** utilizan la estructura del grafo como fuente de conocimiento para establecer la similitud entre dos nodos. A su vez, se clasifican en tres grupos: las medidas basadas en vecinos, las basadas en caminos, y las basadas en caminos aleatorios (*random walks*). Las **medidas basadas en vecinos** tienen en cuenta la información sobre los vecinos del par de nodos del que se quiere establecer su similitud. Esto se basa en el hecho de que las personas se relacionan más con aquellas personas más cercanas. Algunas de las medidas basadas en vecinos son las que se utilizan en esta tesis doctoral debido a que son las técnicas que menos conocimiento requieren (véanse los Capítulos 4 y 5). En segundo lugar, existen las **medidas basadas en ca-**

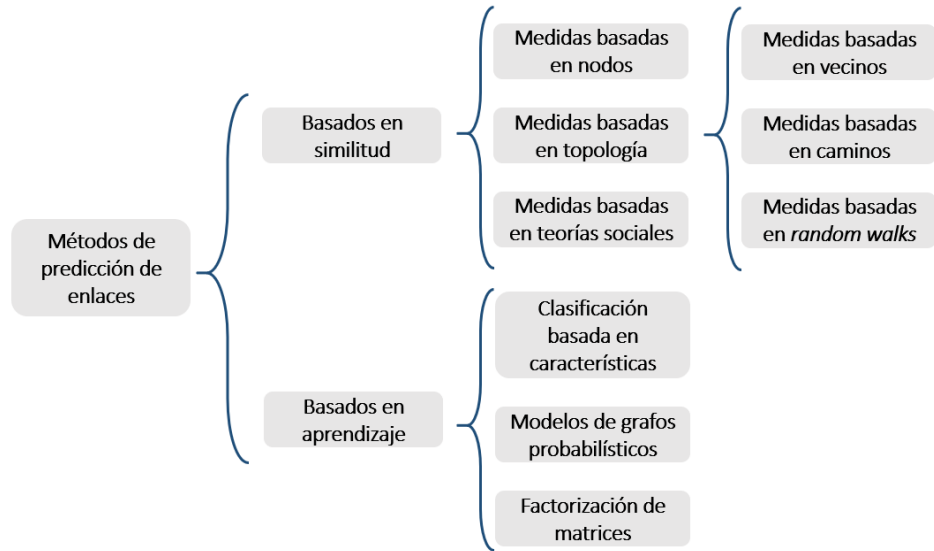


Figura 2.8: Clasificación de los métodos de predicción de enlaces según Wang et al. (2015).

minos, las cuales calculan el valor de similitud entre dos nodos usando la información sobre los caminos que unen ambos nodos. Por último, tenemos las **medidas basadas en caminos aleatorios**. Estas medidas calculan los caminos a partir de las probabilidades de transición de un nodo a sus vecinos. Nótese que las medidas de similitud, tanto éstas basadas en técnicas de predicción de enlaces, como otras encontradas en la literatura son medidas y no métricas. Una medida de similitud otorga una puntuación que representa la similitud entre dos entidades, simbolizando un valor alto mayor similitud que un valor bajo. No cumplen propiedades que sí tienen las métricas, como la desigualdad triangular (Ontañón, 2020).

Volviendo de nuevo a la clasificación de las medidas basadas en similitud, falta por definir en qué consisten las medidas de similitud basadas en **teorías sociales**. Se basan en el uso de la topología del grafo, mejorando su precisión a través de la utilización de información adicional sobre aspectos sociales del sistema. La mezcla de ambas fuentes de conocimiento hace que sean las técnicas de predicción de enlaces basadas en similitud más efectivas (Kumar et al., 2020).

Otros trabajos como Kumar et al. (2020) y Lü y Zhou (2011) dividen los métodos de similitud en base a la cantidad de información a la que acceden: los métodos locales, los globales y los cuasi-locales. Estos conceptos son análogos a los ya mencionados en la Sección 2.2.1. Los métodos basados en similitud **globales** son aquellos que pueden acceder a toda la información del grafo. Un ejemplo sería los métodos basados en caminos. Por su parte, los métodos **locales** utilizan sólo una parte de la estructura del grafo para

generar los valores de similitud. Las medidas basadas en vecinos son locales. Por último, los **cuasi-locales** no necesitan usar la información total del grafo, sólo una parte. Además de esto, utilizan otra fuente de conocimiento. Las medidas basadas en teorías sociales pueden formar parte de este grupo.

En cuanto a los métodos basados en aprendizaje, se pueden establecer tres tipos: clasificación basada en características, modelo de grafo probabilístico y factorización de matrices. El método de **clasificación basada en características** determina que dos nodos del grafo se van a conectar si tienen las mismas características que otro par de nodos ya enlazados. Los **modelos de grafos probabilísticos** usan grafos probabilísticos para generar las predicciones de los enlaces. Un grafo probabilístico es aquel en el que se le asocia un valor de probabilidad a cada par de nodos. El valor se aprende a partir de la estructura del grafo. En el caso de la **factorización de matrices** se extraen factores latentes existentes entre los nodos, de forma que se puede decidir si los nodos se van a conectar o no en el futuro.

En esta tesis doctoral se propone el uso de grafos y técnicas de predicción de enlaces para generar recomendaciones y explicaciones, de una forma que aún no se ha explorado en la literatura, a partir de la aplicación de una proyección bipartita, que se detallará en los próximos capítulos.

2.4. Conclusiones

Como resultado del Objetivo 0 de esta tesis doctoral, en este capítulo se ha hecho una extensa revisión sobre sistemas recomendadores y técnicas de explicación aplicables a este tipo de sistemas. A partir de este estudio, podemos concluir que los sistemas de recomendación se han convertido en una de las herramientas más útiles e importantes de la economía digital. Los usuarios encuentran cada vez más difícil la tarea de tomar decisiones sobre qué productos o experiencias consumir, debido a la oferta tan grande a la que pueden acceder. Los sistemas de recomendación ayudan a los usuarios a encontrar estos productos de acuerdo a sus necesidades y preferencias. Los recomendadores tradicionales se basan en los *ratings* que los usuarios dan a los productos (filtrado colaborativo), o en las descripciones de los propios productos (basados en contenido). Aunque también pueden basarse en información del dominio y restricciones de los usuarios (basados en conocimiento), en general, los sistemas basados en filtrado colaborativo o en contenido son los más usados en la literatura, tanto puros como híbridos, como hemos visto a lo largo del capítulo. Los sistemas híbridos surgen como forma de paliar las desventajas que cada tipo de sistema de recomendador posee. Entre los principales problemas que tienen los distintos tipos de recomendadores se pueden encontrar el *cold start*, el *long tail* o la dispersión.

Otro de los principales problemas que nos encontramos en todos los algoritmos de recomendación es el problema de la falta de confianza por parte

de los usuarios. Si los usuarios no entienden cómo funciona el sistema de recomendación o las razones por las que se ha recomendado un determinado producto, entonces los usuarios no confiarán en el sistema, afectando a la calidad de la recomendación. Si el usuario confía en el sistema, entonces es más probable que utilice el recomendador, aumentando la experiencia de usuario con la propia plataforma, no sólo con el sistema de recomendación. Los sistemas de explicación son los que se encargan de aportar explicabilidad en los sistemas de recomendación, ayudando a los usuarios a entenderlos. Los sistemas de explicación, en general, también se basan en filtrado colaborativo y en contenido.

Hemos visto en este capítulo que los grafos son una de las herramientas que se pueden emplear para implementar sistemas de recomendación y sistemas de explicación. Pueden representar la información de los *ratings* o de las descripciones de los productos. Entre otras, las técnicas de predicción de enlaces, que son un conjunto de métodos que se utilizan para analizar grafos y determinar enlaces que se van a crear o enlaces que han desaparecido en el grafo, se pueden utilizar como método de recomendación, e incluso de explicación de recomendaciones. Los grafos también pueden usarse como herramienta de visualización de explicaciones, aportando en muchos casos interactividad a la interfaz. Esto permite a los usuarios acceder a más información e incluso la posibilidad de conversar con el sistema, para comunicarle si entienden o no la recomendación.

Existen escenarios de *mínimo conocimiento* donde los sistemas de recomendación y explicación tradicionales no se pueden utilizar, ya que no existen *ratings* o descripciones de los productos que puedan usarse como fuente de conocimiento. Aunque el filtrado colaborativo se puede aplicar cuando existen *ratings* binarios (no hay valores de los *ratings*, sólo interacciones positivas o negativas), carece de la interpretabilidad que los grafos pueden aportar, especialmente si se emplean las técnicas de predicción de enlaces locales (aquellas que sólo utilizan una parte de conocimiento representado en el grafo, pero que son más interpretables que las globales). En esta tesis doctoral se propone usar grafos y técnicas de predicción de enlaces para generar recomendaciones y explicaciones en escenarios de mínimo conocimiento y como método alternativo a estas técnicas tradicionales.

Como consecuencia del estudio del estado del arte revisado en este capítulo, en esta tesis doctoral se ha desarrollado un modelo conceptual para definir sistemas de explicación en recomendadores. En el siguiente capítulo, describimos este modelo, cómo lo hemos validado y la ontología implementada a partir del modelo conceptual.

Capítulo 3

Modelo conceptual de explicaciones en sistemas de recomendación

Los sistemas de explicación son una herramienta indispensable hoy en día en el campo de la inteligencia artificial, y en concreto en los sistemas recomendadores. Una explicación puede tener distintas características que van a influir en la aceptación de la recomendación por parte del usuario. Por ejemplo, para tomar una decisión más certera, el usuario puede necesitar confiar en el sistema o disfrutar con el proceso de recomendación, por lo que el sistema podría buscar la confianza o la satisfacción del usuario, respectivamente. También podría buscar que el usuario le preste atención, que aprenda algo nuevo o que pase el máximo tiempo posible en la plataforma (Tintarev y Masthoff, 2015). Además de los distintos propósitos o tipos de usuarios que un sistema de explicación puede tener, también puede, por ejemplo, usar distintas fuentes de conocimiento, mostrar las explicaciones de diversas maneras o permitir al usuario interactuar con él de distintas formas. Por tanto, la heterogeneidad de los sistemas de explicación es muy amplia, y esto puede hacer que los diseñadores de estos sistemas tengan dificultades a la hora de decidir cómo establecer sus características. La importancia a la hora de tomar estas decisiones es alta, ya que el diseño del sistema influye enormemente en la confianza del usuario en el sistema. Por ejemplo, el usuario no procesa igual la información de una explicación si se le muestra textualmente o visualmente. Como consecuencia, es importante que los diseñadores conozcan todas las posibles características, y las interrelaciones y dependencias entre las mismas, que un sistema de explicación puede adoptar para conseguir que las explicaciones se ajusten a lo que se necesite según el contexto.

Como consecuencia de la realización del Objetivo 0 de la tesis –*estudio de los sistemas de recomendación y técnicas de explicación aplicables*–, en

este capítulo se propone un modelo conceptual que ayude a resolver el problema descrito en el párrafo anterior. Un *modelo conceptual* es un conjunto de entidades y características organizadas teniendo en cuenta sus relaciones para definir un concepto con el objetivo de orientar la investigación sobre este concepto (Thalheim, 2014). Por tanto, en nuestro modelo propuesto se definen y describen de forma exhaustiva todos los posibles aspectos que un diseñador debería tener en cuenta para crear un sistema que genere explicaciones exitosas. Como hemos comentado en el Capítulo 2, ya existen taxonomías de clasificación de explicaciones para sistemas inteligentes y sistemas de toma de decisiones, entre ellos, recomendadores. Sin embargo, en esta tesis doctoral se ha realizado un refinamiento respecto a estos modelos anteriores.

De este modo, proporcionamos un nuevo modelo que analiza, integra y mejora los ya existentes incluyendo, además, conceptos novedosos. Incorporamos también las relaciones e interdependencias que existen entre estos conceptos, lo cual no se ha definido en trabajos anteriores. Lo que principalmente diferencia nuestra contribución con respecto a las anteriores taxonomías encontradas en la literatura es la formalización de este modelo, dando lugar a la ontología ExRecOnto. En trabajos anteriores del grupo de investigación se han desarrollado otras ontologías para definir e implementar sistemas CBR (Díaz-Agudo y González-Calero, 2002) y sistemas de recomendación (Jorro-Aragoneses et al., 2020b, 2017b). En estas ontologías se hacían descripciones semánticas de los sistemas para apoyar el razonamiento que se lleva a cabo durante el proceso de implementación del sistema en cuestión. Con ExRecOnto, el objetivo es el mismo. Esta ontología podrá ser utilizada por distintos procesos de ingeniería de software como herramienta de creación de sistemas de explicación, siguiendo una metodología de desarrollo guiada por ontologías. A partir de los requisitos que se pide implementar, la ontología es capaz de definir qué tipo de sistema de explicación se necesita y cuáles son los distintos conceptos que lo van a definir, los principales elementos a implementar y las relaciones existentes entre los elementos del sistema.

Para demostrar su idoneidad, el modelo ha sido validado clasificando una gran variedad de sistemas de explicaciones encontrados en la literatura, en su mayoría para sistemas de recomendación. Sin embargo, también hemos tenido en cuenta sistemas de explicación para otro tipo de sistemas inteligentes, como aprendizaje automático o aplicaciones de planificación. De este modo, demostramos que nuestro modelo puede categorizar una gran variedad de sistemas de explicaciones para distintos sistemas de inteligencia artificial, aunque nuestro modelo esté específicamente enfocado a recomendadores.

Para poder entender cómo se ha definido el modelo conceptual, y por tanto ExRecOnto, es necesario saber cuáles son los tipos de conceptos y relaciones contemplados en esta ontología, antes de conocer los detalles y descripciones de los propios conceptos. Las ideas obtenidas de los trabajos

anteriores se detallan en la Sección 3.1. A continuación, en la Sección 3.2, se hace una definición de cuáles son los tipos de conceptos de nuestro modelo. En la Sección 3.3 continuamos con la descripción del modelo, detallando cada uno de los conceptos representados y el propósito de cada uno de ellos. Más tarde, en la Sección 3.4, se explica cómo se ha validado y las conclusiones obtenidas tras esta evaluación. Por último, la Sección 3.5 especifica la formalización del modelo conceptual descrito anteriormente, dando lugar a ExRecOnto. Tras esto, en la Sección 3.6, se pueden encontrar unas conclusiones del trabajo realizado en el Objetivo 0 de esta tesis doctoral.

3.1. Trabajo relacionado

Los tres principales trabajos en los que nos hemos basado para desarrollar nuestro modelo conceptual son las taxonomías para sistemas de recomendación propuestas por Nunes y Jannach (2017); Friedrich y Zanker (2011) y Papadimitriou et al. (2012).

El trabajo de Nunes y Jannach (2017) es el más completo que hemos estudiado. Hace una extensa revisión de la literatura con respecto a sistemas de explicación para recomendadores y para sistemas de toma de decisiones. A pesar de la extensión y detalle de la taxonomía propuesta, carece de algunos conceptos de los que se habla en otros trabajos. Los principales conceptos que hemos echado en falta son los conceptos **Ante hoc** y **Post hoc**, necesarios hoy en día para definir cualquier tipo de sistema de explicación. También hemos encontrado que no se habla sobre el proceso de interacción que se puede llevar a cabo con el sistema, y que no se definen específicamente los tipos de usuarios que los sistemas de explicación pueden tener. Además, hay conceptos que aparecen incompletos de acuerdo al estudio realizado en esta tesis doctoral. Un ejemplo sería la fuente de conocimiento, ya que en el trabajo de Nunes y Jannach (2017) no se mencionan, por ejemplo, el historial o la información social como tipos de conocimiento para generar explicaciones. También hay conceptos a los que hemos querido darles un nuevo enfoque para aumentar la especificación y mejorar la legibilidad del modelo conceptual. Un ejemplo es el **Formato** de la presentación de la explicación. Nunes y Jannach (2017) definen dos tipos de formatos: lenguaje natural y multimedia, mientras que en nuestro modelo conceptual decidimos hacer una clasificación algo más detallada definiendo los tipos de formatos como **Lenguaje natural**, **Esquemático**, **Visual** y **Otros**, como veremos en este capítulo.

Las taxonomías propuestas por Friedrich y Zanker (2011) y Papadimitriou et al. (2012) son mucho menos extensas que la de Nunes y Jannach (2017), pero a pesar de ello definen algunas de las características de las que ésta carece como los conceptos de caja negra y caja blanca, o algunos tipos adicionales de fuentes de conocimiento. También hemos obtenido ideas de otros trabajos, que han sido tanto revisiones de la literatura como nuevas

propuestas de sistemas de explicación. De ellas hemos obtenido ideas principalmente de tipos de usuarios, de conocimiento, de argumentación o de interactividad. Detallamos a continuación las publicaciones que han inspirado los conceptos definidos en nuestro modelo propuesto.

El trabajo de Sharma y Cosley (2013) presenta varios tipos de explicaciones sociales en un sistema de recomendación de artistas musicales. En concreto, este trabajo promueve el uso de las relaciones sociales como forma de explicar las recomendaciones. En los trabajos de Musto et al. (2016), y Sansonetti et al. (2019), se descubren otros tipos de fuentes de conocimiento para implementar explicaciones, como *Linked Open Data* (LOD). Las propuestas de Lu et al. (2018) y Peña et al. (2020) describen modelos innovadores para mezclar la información textual de las reseñas de los usuarios y el conocimiento de la factorización de matrices para realizar explicaciones. Estos cuatro trabajos hicieron que nos diéramos cuenta de la amplia variedad de fuentes de conocimiento utilizadas en los sistemas de explicación y que contempláramos nuevos conceptos no representados en las taxonomías anteriores. La publicación de Lamche et al. (2014) es una descripción detallada de una explicación interactiva para sistemas de recomendación en una aplicación móvil. En particular, las ideas más útiles que encontramos aquí son el uso de explicaciones interactivas, la forma de presentar una explicación en aplicaciones móviles y el análisis de las diferentes perspectivas que puede tener una explicación. El trabajo de Buchanan y Shortliffe (1984) propone un sistema de explicaciones basado en reglas. Describe un sistema de explicación que justifica el comportamiento del sistema utilizando la información sobre su propio funcionamiento. Este trabajo nos llevó a distinguir de nuevo entre los métodos de explicación *Ante hoc* o *Post hoc*, además de a considerar el proceso de recomendación como fuente de conocimiento. También detalla algunos objetivos de las explicaciones no tan comunes en la literatura, como la *Educación* y el *Debugging*. En el trabajo de Berkovsky et al. (2017), los autores realizaron un estudio que analiza la importancia y el impacto de la confianza en el éxito de los sistemas de recomendación. En concreto, se centra en comparar diferentes tipos de visualizaciones, lo cual confirmó nuestra idea de clasificar los sistemas de explicación según el tipo de visualización que usen para mostrar las explicaciones a los usuarios. El trabajo de Zanker y Ninaus (2010) propone un *framework* basado en conocimiento para la generación de explicaciones. El objetivo de esta publicación es proporcionar transparencia a las recomendaciones basadas en las coincidencias entre las características de los productos y las preferencias de los usuarios. De este estudio, confirmamos la idea de clasificar las explicaciones según la fuente de conocimiento utilizada para generarlas. En el trabajo de Tsai y Brusilovsky (2020), los autores presentan un sistema de recomendación híbrido que permite a los usuarios observar las explicaciones y controlarlas. Con este trabajo confirmamos algunas de las ideas que ya habíamos recogido, como la impor-

tancia de la fuente de conocimiento, los tipos de visualizaciones y algunas de las propiedades críticas de las explicaciones, como la **Escrutabilidad** o la **Transparencia**.

3.2. Representación del modelo

Para entender cómo definimos un sistema de explicación para recomendadores y los conceptos a los que se hace referencia en nuestro modelo conceptual es necesario hacer una descripción de entidades de dominio sobre sistemas recomendadores. Tras esto, vamos a definir los tipos de conceptos y relaciones que se utilizan en el modelo conceptual propuesto.

3.2.1. Entidades de dominio sobre sistemas de recomendación

En el Capítulo 2 se ha hecho una extensa revisión de los sistemas de recomendación, sus características y su funcionamiento. Sin embargo, con el objetivo de entender el modelo conceptual aquí propuesto, se hará un breve resumen de la terminología que consideramos más relevante para comprender las siguientes secciones de este capítulo.

Las principales entidades que intervienen en los sistemas de recomendación son los *usuarios* y los *productos* de la plataforma donde se encuentra implantado el sistema recomendador. Los *usuarios* objetivo de la recomendación son las personas interesadas en conseguir o consumir nuevos productos y que interactúan con el sistema para explorar esos productos en la plataforma. Los *productos* son los objetos o experiencias disponibles en la plataforma y, por tanto, también, en el sistema de recomendación: películas, música, ropa, hoteles, etc.

El proceso de recomendación se basa en diferentes fuentes de datos sobre los usuarios y los productos. Por un lado, las *interacciones* realizadas por los usuarios en el sistema de recomendación son una de las fuentes más explotadas para recomendar nuevos productos a estos usuarios. Una de las formas más utilizadas para identificar las preferencias de los usuarios son los *ratings* o puntuaciones, una forma de valoración explícita de los usuarios sobre un producto. Si un usuario califica un producto con un buen *rating*, entonces se considera que este producto (y sus características) será apropiado para este usuario. Cada producto tiene su propio *contenido*, una representación que describe el producto; por ejemplo, una descripción de texto. Este contenido puede estar representado por un conjunto de *características* que definen el producto, por ejemplo su precio, color o tamaño, entre otras muchas.

Con frecuencia, la recomendación se apoya en las *similitudes* entre usuarios y/o productos. La similitud es la semejanza entre dos elementos; por ejemplo, entre las características de los productos a recomendar y las carac-

terísticas de los productos valorados por los usuarios. Este es el procedimiento más habitual, ya que la recomendación se apoya en las *preferencias* del usuario.

Por último, cada usuario forma parte de la *comunidad* definida por el sistema de recomendación. El comportamiento de cada usuario individual y de la comunidad puede emplearse en el proceso de recomendación. Además, la recomendación puede mejorarse utilizando la información extraída del entorno, las circunstancias del usuario y su interacción con sistemas de terceros. A esto lo llamamos *información contextual*.

3.2.2. Relaciones semánticas

En cualquier modelo semántico, sus conceptos deben estar relacionados mediante relaciones significativas. Concretamente, en este trabajo utilizamos las relaciones básicas que se encuentran en cualquier lenguaje de representación de ontologías (Staab y Studer, 2010):

Composición. Una composición es la combinación de conceptos que definen otro concepto desde diferentes perspectivas. Por ejemplo, veremos que en nuestro modelo definimos el concepto de **Presentación** como una composición de 4 conceptos (**Formato**, **Argumentación**, **Nivel de detalle** e **Interacción**) que describen cómo se presenta una explicación al usuario.

Subconcepto. Un subconcepto es una extensión que refina la definición de este concepto. Representa una relación “*es-un*”. Los subconceptos no son disjuntos, por lo que los sistemas de explicación pueden clasificarse en varios subconceptos. Por ejemplo, nuestro modelo define dos dimensiones más profundas (subconceptos) para determinar los **Objetivos** de un sistema de explicación: los **Objetivos de alto nivel** y los de **bajo nivel**.

Instancia. Una instancia es un ejemplo concreto o un caso de un concepto. Por ejemplo, en nuestro modelo veremos que el concepto de **Formato** tiene cuatro instancias: **Lenguaje natural**, **Esquemático**, **Visual** y **Otros**. Igualmente, varias instancias del mismo concepto pueden caracterizar una explicación.

3.3. Descripción del modelo

Tras el estudio sobre sistemas de explicación y taxonomías orientadas a clasificar este tipo de sistemas, en esta tesis doctoral se propone un nuevo modelo integrados de las diferentes taxonomías existentes (Nunes y Jannach, 2017; Friedrich y Zanker, 2011; Papadimitriou et al., 2012) que resuelve las

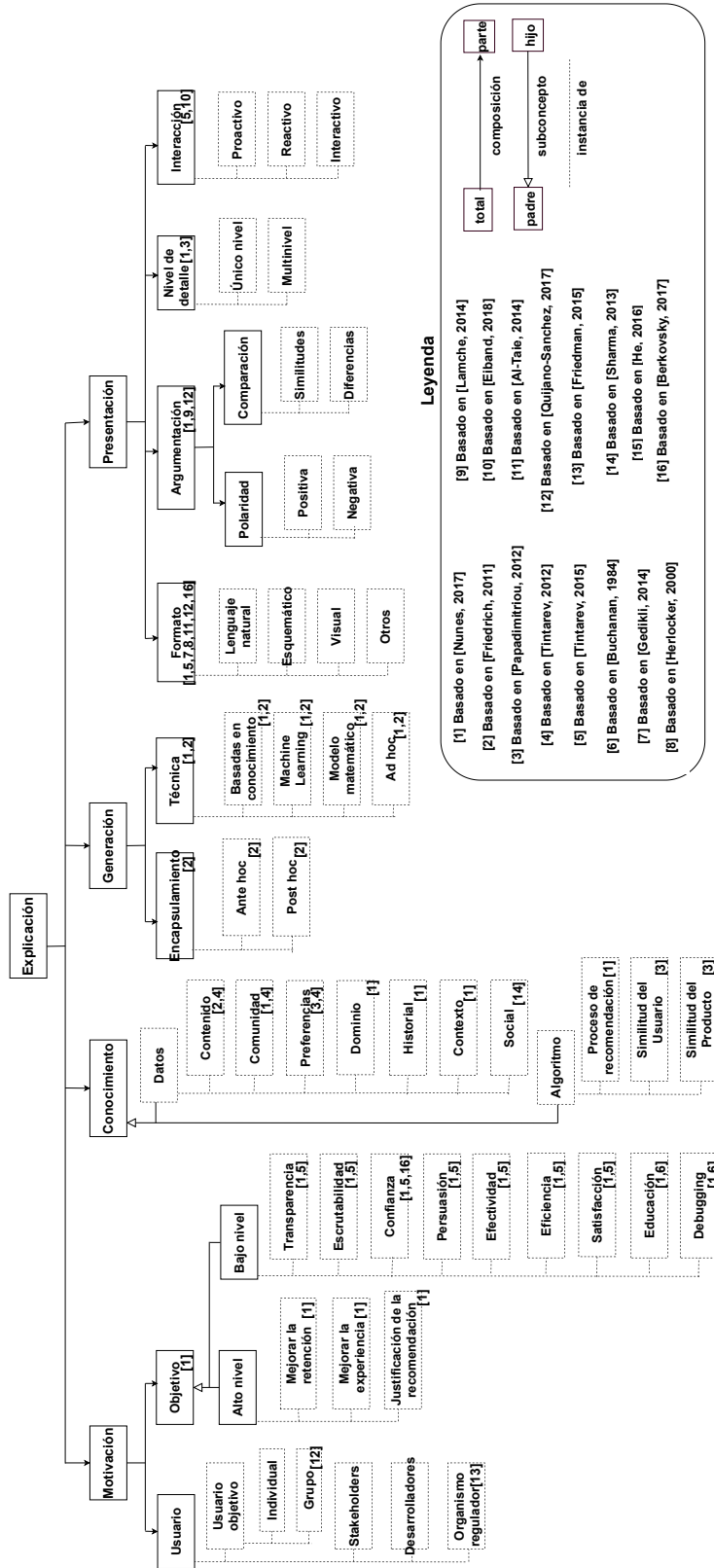


Figura 3.1: Visión general del modelo conceptual.

diferentes carencias identificadas. Se han refinado los conceptos ya vistos en anteriores trabajos, incluyendo además otros nuevos conceptos innovadores.

Nuestro modelo introduce cuatro dimensiones o conceptos principales: **Motivación**, **Conocimiento**, **Generación** y **Presentación**. La Figura 3.1 muestra una visión general del modelo conceptual desarrollado, que se detallará en las siguientes subsecciones.

3.3.1. Motivación

La **Motivación** se refiere al **Objetivo** o propósito de la explicación dependiendo de quién sea el **Usuario** que recibe la explicación. Ambas ideas, objetivo y usuario, componen la motivación de las explicaciones.

3.3.1.1. Usuario

Se supone que los sistemas de recomendación deben satisfacer las necesidades de un usuario final. Sin embargo, las explicaciones no tienen por qué estar dirigidas exclusivamente a ese usuario objetivo de la recomendación. Los objetivos de cada tipo de usuario son distintos, y por tanto, las explicaciones también tienen que ser diferentes. Por tanto, podemos clasificar los sistemas de explicación en función del **Usuario** al que va dirigida la explicación.

Usuario objetivo. Son los usuarios que reciben las recomendaciones realizadas por el sistema. Las explicaciones son relevantes para los usuarios objetivo por la necesidad de entender cómo o por qué se recomienda un producto para valorar si les interesa. Este es el tipo de usuario más común en los sistemas de explicación. Podemos considerar dos tipos de usuarios objetivo: usuarios **individuales** y **grupos** de usuarios. El usuario objetivo es un único usuario cuando el usuario que recibe la recomendación también recibe la explicación. En este caso, la explicación se diseña para justificar la recomendación para ese usuario concreto. Sin embargo, las recomendaciones y, por tanto, también las explicaciones también pueden ir dirigidas a grupos (Jameson y Smyth, 2007; Recio-Garcia et al., 2009), por lo que en esos casos, las explicaciones suelen tratar de describir las limitaciones y condiciones de cada miembro del grupo a la hora de generar la recomendación.

Stakeholders o partes interesadas. Son las personas interesadas en el éxito del sistema de recomendación, como propietarios de productos, inversores o empresarios. No reciben la recomendación, pero necesitan explicaciones para medir el rendimiento y el éxito del proceso de recomendación. Por ejemplo, el sistema de reglas IMVEX ofrece algunas explicaciones al usuario objetivo y otras a los interesados, en función de lo que necesiten saber (Finch, 1998).

Desarrolladores. Los desarrolladores de sistemas de recomendación necesitan saber cómo funciona el recomendador para realizar diferentes tareas, como la depuración, el mantenimiento o la ampliación del sistema. En este caso, las explicaciones se centran en ayudar a los desarrolladores a entender el comportamiento interno del recomendador (Robillard et al., 2010).

Organismo regulador. Este tipo de usuario representa a las organizaciones legislativas y gubernamentales que necesitan saber cómo funcionan los sistemas de recomendación para regular su transparencia y el uso que se hace de ellos (Friedman et al., 2015). Hoy en día, el uso de los sistemas de recomendación se ha extendido incluso incorporándose a los procesos gubernamentales. Por ello, es cada vez más patente la necesidad social de que los organismos reguladores garanticen el correcto comportamiento de los sistemas de recomendación. Un ejemplo claro de esta necesidad es la polémica surgida por la existencia de un posible sesgo en un sistema muy utilizado para predecir riesgos de reincidencia penal (Angwin et al., 2016).

3.3.1.2. Objetivo

El **Objetivo** de una explicación representa el propósito de esa explicación y está estrechamente relacionado con el usuario al que va dirigida, como ya se ha comentado anteriormente. Los objetivos pueden dividirse en objetivos de **Alto nivel** y objetivos de **Bajo nivel**. Los objetivos de alto nivel se centran en el usuario, mientras que los de bajo nivel ayudan a alcanzar los objetivos de alto nivel.

Los objetivos de nivel superior se basan en los objetivos propuestos por Nunes y Jannach (2017):

Mejorar la retención de los usuarios. El objetivo de la explicación es aumentar las probabilidades de que los usuarios vuelvan al sistema de recomendación, lo que suele ser muy relevante para los **Stakeholders**, ya que pretenden mejorar la retención de los **Usuarios** para aumentar sus propios beneficios.

Mejorar la experiencia del usuario. En este caso, el objetivo principal de la explicación es ayudar a los usuarios a disfrutar de la propia experiencia de la recomendación, lo cual está relacionado con la toma de buenas decisiones. Por tanto, la explicación se dirige principalmente al **Usuario objetivo** de la recomendación y a los **Desarrolladores** encargados de implementar el sistema. También está relacionado con la retención de los usuarios, ya que está estrechamente influenciada por la experiencia del usuario: cuanto mejor sea la experiencia, más probable es que el usuario vuelva a usar el sistema recomendador.

Justificación de la recomendación. El objetivo de este tipo de explicaciones es apoyar la recomendación proporcionada al usuario para que entienda por qué se ha recomendado un producto. Este tipo de explicaciones puede ayudar a su vez a mejorar la retención y la experiencia del usuario porque se centra en satisfacer las necesidades de los **Usuarios objetivo** y las tareas de los **Desarrolladores**, enfocándose en el funcionamiento interno del sistema.

Aunque estos son los objetivos propuestos por Nunes y Jannach (2017), otros trabajos como los de Tintarev y Masthoff (2015) y Buchanan y Shortliffe (1984) proponen criterios más específicos para diseñar buenas explicaciones. En esta tesis doctoral se han considerado estos criterios como **Objetivos de bajo nivel**. Podemos diseñar una explicación teniendo en cuenta uno o varios de estos criterios, que además están relacionados entre sí y con los **Objetivos de alto nivel**. Siguiendo los modelos sugeridos en trabajos anteriores (Nunes y Jannach, 2017; Tintarev y Masthoff, 2015; Buchanan y Shortliffe, 1984), hemos incluido los siguientes **Objetivos de bajo nivel** en nuestro modelo:

Efectividad. La explicación tiene que ayudar al usuario a encontrar los productos que necesita o desea (Tintarev y Masthoff, 2015).

Eficiencia. Una buena explicación ayuda al usuario a tomar decisiones rápidamente (Tintarev y Masthoff, 2015).

Confianza. Las explicaciones deben aumentar la fiabilidad del usuario en el sistema. Si los usuarios entienden la explicación y cómo se llega a una determinada recomendación, su confianza en el sistema mejorará (Tintarev y Masthoff, 2015; Buchanan y Shortliffe, 1984; Berkovsky et al., 2017).

Escrutabilidad. Cuando una explicación es escrutable, el usuario puede dar su opinión sobre el funcionamiento del sistema de recomendación al sistema de explicación (Tintarev y Masthoff, 2015).

Persuasión. Una explicación es persuasiva si convence al usuario de que la recomendación es adecuada (Tintarev y Masthoff, 2015).

Satisfacción. Las explicaciones ayudan a aumentar la calidad de las recomendaciones percibida por el usuario, mejorando la experiencia del usuario, y su disfrute general con el sistema (Tintarev y Masthoff, 2015).

Transparencia. Una explicación es transparente cuando especifica cómo y por qué se hace una recomendación (Tintarev y Masthoff, 2015; Buchanan y Shortliffe, 1984). La transparencia y la justificación podrían parecer iguales, pero no lo son. La transparencia aporta información sobre cómo el modelo de recomendación crea las recomendaciones. Sin embargo, la justificación aporta las razones detrás de una recomendación sin tener por qué tener en cuenta el proceso llevado a cabo por el modelo de recomendación.

Educación. Las explicaciones permiten a los usuarios aprender algo nuevo del sistema, lo que además puede conllevar una mejor toma de decisiones. Si esto ocurre, las probabilidades de que un usuario vuelva al sistema de recomendación aumentan (Buchanan y Shortliffe, 1984). Por ejemplo, puede ocurrir que el sistema explique la toma de una decisión en base a un conocimiento científico, lo que permite al usuario aprender a la vez que toma una mejor decisión.

Debugging o depuración. Las explicaciones permiten a los usuarios, que en este caso son los **Desarrolladores**, identificar errores en el sistema de recomendación (Buchanan y Shortliffe, 1984). Aquí incluimos también cualquier otra tarea de ampliación o mantenimiento del software.

3.3.2. Conocimiento

Los sistemas de explicación necesitan una fuente de **Conocimiento** útil para obtener justificaciones efectivas para el **Usuario** (Richter y Weber, 2016; Zanker y Ninaus, 2010). La información que se le muestre al usuario se obtendrá de esta fuente, por lo que es necesario conocer los distintos tipos de conocimiento a los que se puede acudir para generar la mejor explicación posible.

Así, introducimos el **Conocimiento** como un factor novedoso en nuestro modelo. Según la literatura existente, las explicaciones pueden extraer información de dos tipos de fuentes: los **Datos** disponibles para el recomendador o el propio **Algoritmo** utilizado para generar la recomendación. A partir de esta observación, se proponen dos subconceptos diferentes para definir la naturaleza del conocimiento utilizado para generar la explicación. En las siguientes secciones, detallamos estos subconceptos.

3.3.2.1. Datos

Son las explicaciones basadas en la información disponible durante el proceso de recomendación. Hemos identificado diferentes tipos de datos utilizados en las explicaciones:

Contenido. La explicación de un producto recomendado se refiere a la descripción del producto. Aunque muestra información no personalizada, ayuda al usuario a tomar buenas decisiones (Friedrich y Zanker, 2011; Tintarev y Masthoff, 2012). Por ejemplo, “Esta camisa es blanca y su precio es de 8,99€”.

Preferencias. La explicación se refiere a las características del producto teniendo en cuenta las preferencias del usuario (Papadimitriou et al., 2012; Tintarev y Masthoff, 2012). Por ejemplo, “Esta camisa es blanca, un color que te gusta, y su precio es de 8,99€, menos de 10€, tu presupuesto”.

Comunidad. La información utilizada para explicar la recomendación se extrae del comportamiento de la comunidad de usuarios que interactúan con el sistema recomendador (Nunes y Jannach, 2017; Tintarev y Masthoff, 2012). Por ejemplo, “Esta camisa es una de las más vendidas de la tienda”.

Dominio. La información extraída del dominio es la fuente de conocimiento para generar la explicación, como las reseñas realizadas por los usuarios (Nunes y Jannach, 2017). Por ejemplo, “Esta camisa está hecha con algodón orgánico, que se cultiva con métodos y materiales que tienen un bajo impacto en el medio ambiente”. Otro ejemplo es la propuesta de Muhammad et al. (2015), donde encontramos un sistema que explica las recomendaciones extrayendo las características del hotel que se recomienda a partir de las reseñas de los usuarios.

Historial. Este tipo incluye todas las explicaciones que hacen referencia a las experiencias pasadas del usuario y al historial del sistema (Nunes y Jannach, 2017). Por ejemplo, “Esta camisa es blanca, y ayer compraste camisas blancas”.

Contexto. Estos datos se refieren a los requisitos dinámicos de los usuarios objetivo. Por ejemplo, un sistema de recomendación de itinerarios turísticos debe adaptar la recomendación en función de las restricciones meteorológicas cuando el usuario consulta el sistema. En este caso, las preferencias del usuario no cambian (son un conocimiento estático), pero hay que tener en cuenta el contexto para adaptar la recomendación al entorno concreto del usuario (que es dinámico) (Adomavicius y Tuzhilin, 2011). Por ejemplo, “Esta camiseta es adecuada para los días cálidos como hoy”.

Social. La explicación hace referencia a las relaciones personales del usuario con otros usuarios para justificar las recomendaciones (Sharma y Cosley, 2013). Por ejemplo, “Tu hermano se ha comprado esta camiseta” o “Tu mejor amigo también se ha comprado esta camiseta”.

3.3.2.2. Algoritmo

En este caso, las explicaciones usan la información sobre el razonamiento detrás del algoritmo de recomendación. Las explicaciones pueden basarse en todo el **Proceso de recomendación** o en un subcomponente relevante, principalmente información sobre **Similitudes del usuario** objetivo o información sobre **Similitudes del producto** recomendado. Las instancias propuestas para este tipo de conocimiento son las siguientes:

Proceso de Recomendación. Este tipo de explicación muestra cómo el algoritmo realiza una recomendación (Nunes y Jannach, 2017; Eiband et al., 2018). Por ejemplo, “Esta camisa ha sido recomendada porque el sistema ha estimado que es la más barata que se ajusta a tus preferencias”. Otro ejemplo lo encontramos en MYCIN, que es un sistema experto que selecciona el número mínimo de medicamentos para curar infecciones. CHRONICLER explica la recomendación de MYCIN analizando el proceso de decisión del algoritmo usado por MYCIN, mostrando al usuario una crónica de los eventos del sistema (Buchanan, 1984).

Similitud del usuario. La explicación justifica la recomendación a través de las interacciones que usuarios similares al usuario objetivo han realizado con los productos recomendados (Papadimitriou et al., 2012). Por ejemplo, “Esta camisa fue recomendada porque usuarios similares a ti la compraron”.

Similitud del producto. Las explicaciones se basan en la similitud que existe entre los productos recomendados y los productos con los que el usuario interactuó (Papadimitriou et al., 2012). Por ejemplo, “Esta camisa fue recomendada porque compraste camisas similares antes”.

3.3.3. Generación

El concepto de **Generación** se refiere a cómo se obtiene la explicación a partir del conocimiento. Las explicaciones para los sistemas de recomendación pueden generarse utilizando diferentes procesos y técnicas. Lo definimos como una composición a partir del **Encapsulamiento** del modelo y la **Técnica** implementada para generar las explicaciones.

3.3.3.1. Encapsulamiento

Durante el análisis de las técnicas para implementar sistemas de explicación, observamos que los sistemas de recomendación tienen dos enfoques diferentes de **Encapsulamiento**: como sistema de caja blanca o como sistema de caja negra. Un sistema de caja blanca permite a los usuarios conocer el funcionamiento de su algoritmo. En cambio, un sistema de caja negra no

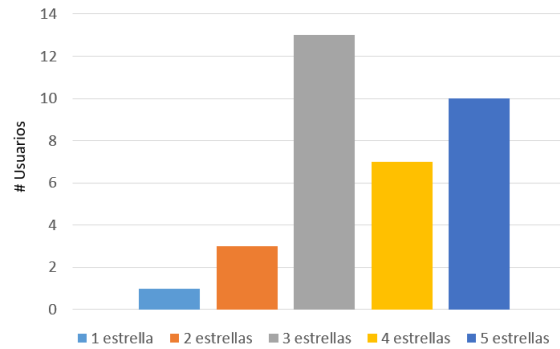


Figura 3.2: Ejemplo de una explicación *Ante hoc*: histograma con las valoraciones de los usuarios para el producto recomendado.

permite conocer su razonamiento interno (Friedrich y Zanker, 2011). De este modo, podemos clasificar el sistema de explicación como *Ante hoc* o *Post hoc*, dependiendo de si el sistema de recomendación actúa como un sistema de caja blanca o de caja negra, respectivamente.

Ante hoc. Las explicaciones son generadas por el propio algoritmo de recomendación, ya que funciona como un sistema de caja blanca (Friedrich y Zanker, 2011). Por ejemplo, las explicaciones propuestas por Herlocker et al. (2000) utilizan histogramas agrupados como un método de explicación *Ante hoc*. Estos histogramas son un tipo de explicación para un recomendador basado en filtrado colaborativo. Muestran un gráfico en el que los usuarios pueden ver las valoraciones de usuarios similares al usuario objetivo para el producto recomendado. Por lo tanto, no se utiliza una técnica distinta al algoritmo recomendador. La Figura 3.2 muestra un ejemplo de explicación *Ante hoc* inspirada en la propuesta de Herlocker et al. (2000).

Post hoc. El sistema de recomendación actúa como un sistema de caja negra, por lo que el sistema de explicación tiene que implementar una técnica externa al recomendador para crear justificaciones (Friedrich y Zanker, 2011). Por ejemplo, las *tagsplanations*, modelo propuesto en (Vig et al., 2009), puede considerarse un sistema de explicación *Post hoc*. Las *tagsplanations* (Figura 3.3) son etiquetas que describen el producto recomendado mostrando además la preferencia del usuario por el mismo. La explicación no se basa en el proceso de recomendación, sino en el contenido y las preferencias.



Figura 3.3: Ejemplo de una explicación *Post hoc*: una nube de etiquetas. Las etiquetas muestran las características del producto recomendado. Las etiquetas en color azul muestran que el usuario objetivo tiene preferencia por esas características, las rojas significan preferencia negativa, y las grises, neutra.

3.3.3.2. Técnica

Independientemente de que el sistema funcione como caja negra o como caja blanca, se pueden utilizar diferentes técnicas para generar explicaciones: razonamiento basado en casos, reglas, minería de datos, etc. Estas técnicas que generan la explicación pueden integrarse en el propio proceso de recomendación cuando la técnica utilizada tanto para generar la explicación y la recomendación es la misma (caja blanca), o como un módulo de explicación *post hoc* (caja negra). Por lo tanto, los sistemas de explicación también deben clasificarse en función del algoritmo utilizado para generar explicaciones, independientemente de su naturaleza *Ante hoc* o *Post hoc*. Los principales tipos de explicación, según su *Técnica*, son (Nunes y Jannach, 2017):

Basadas en conocimiento. Se basan en técnicas que utilizan una fuente de conocimiento para resolver problemas. Ejemplos de sistemas de explicación basados en el conocimiento son CARES o DIRAS, que utilizan el razonamiento basado en casos (CBR) para explicar diagnósticos médicos (Doyle et al., 2003).

Machine learning o aprendizaje automático. Según la definición de Mohri et al. (2012), los algoritmos de aprendizaje automático son técnicas computacionales que utilizan información sobre experiencias pasadas, a la cual el sistema puede acceder para hacer predicciones precisas. En este caso, estas experiencias pasadas se utilizan para explicar las recomendaciones. Por ejemplo, el trabajo de Donkers et al. (2018) aplica técnicas de aprendizaje profundo o *deep learning* a reseñas textuales de los usuarios con el fin de recuperar información y utilizarla para explicar las recomendaciones.

- Se te ha recomendado esta película porque:
- Es una **comedia**, tu género favorito.
 - Te gusta el actor principal es **Hugh Grant**.
 - La película fue **rodada en los 90**. Muchas de tus películas favoritas son de esa década.

Figura 3.4: Ejemplo de una explicación en **Lenguaje natural**.

Modelos matemáticos. Un modelo matemático puede ser utilizado para explicar recomendaciones utilizando teorías y conceptos matemáticos. En este grupo se tienen en cuenta aquellos sistemas que utilizan modelos matemáticos pero que no están incluidos en las técnicas de aprendizaje automático. Por ejemplo, Wang et al. (2016) describe un sistema basado en un modelo probabilístico para explicar recomendaciones mediante el uso de análisis de sentimientos en reseñas.

Ad hoc o específico. El algoritmo de explicación se crea específicamente enfocado para proporcionar las explicaciones requeridas por un determinado sistema de recomendación. Por ejemplo, Musto et al. (2016) propone un sistema de explicaciones que utiliza datos enlazados o *Linked Open Data* (LOD) para generar explicaciones particulares para un sistema recomendador.

3.3.4. Presentación

El concepto de presentación representa la forma con la que el sistema muestra una explicación al usuario. Hemos identificado cuatro conceptos principales que componen el concepto de presentación: el **Formato de visualización**, el tipo de **Argumentación**, el **Nivel de detalle** y el tipo de **Interacción**.

3.3.4.1. Formato de presentación

Las explicaciones pueden mostrarse de diferentes maneras, algunas más elaboradas que otras. Las principales instancias de **Formato** son:

Lenguaje natural. Las justificaciones de una recomendación se muestran como una descripción textual (Figura 3.4). En este caso, es habitual emplear plantillas en las que se muestra un texto predefinido modificándose los valores asociados a esa recomendación concreta. Según los trabajos estudiados, este sería el método más común para presentar una explicación. Podemos observar una explicación en lenguaje natural en el sistema CBR CARES (Doyle et al., 2003), que muestra por qué puede aparecer una complicación en pacientes con diabetes.

Rating	N.º de usuarios similares
★	3
★★	2
★★★	2
★★★★	10
★★★★★	36

Ratings de los usuarios similares para la película recomendada

Figura 3.5: Ejemplo de una explicación **Esquemática**.

Esquemático. Las explicaciones no se muestran en lenguaje natural, sino en un formato de texto simplificado como tablas, listas, etiquetas, *logs* (o registros) o valoraciones. Un ejemplo de explicación esquemática encontrado en la literatura es la propuesta de Herlocker et al. (2000) para MovieLens¹, que justifica por qué se recomienda una película a un usuario a través de valoraciones de usuarios similares (Figura 3.5).

Visual. En este caso, las explicaciones muestran las justificaciones a través de una forma más gráfica, como pueden ser los diagramas o gráficos (histogramas, nubes de etiquetas, etc.). Un ejemplo sería la nube de etiquetas vista anteriormente (Figura 3.3).

Otros. Las explicaciones se generan utilizando formatos de visualización más innovadores; por ejemplo, audio, vídeo, realidad aumentada o mixta, etc. Aunque no hay muchos trabajos que incluyan estas categorías, la taxonomía de Nunes y Jannach (2017) ya incluía un ejemplo de un sistema que utiliza explicaciones de audio y voz para aconsejar a usuarios no expertos sobre los dispositivos eléctricos que deben utilizar (Terano et al., 1989). También hemos encontrado explicaciones desarrolladas en tecnologías innovadoras, como la realidad aumentada, para sistemas de recomendación, en el ámbito del turismo (Gavalas et al., 2014).

3.3.4.2. Argumentación

Las explicaciones pueden mostrar diferentes perspectivas para un producto recomendado: pueden presentar los argumentos a favor (positivos) o en contra (negativos) de este producto para el usuario (Nunes y Jannach, 2017; Quijano-Sanchez et al., 2017; Lamche et al., 2014). Además, estos argumentos, tanto los positivos como los negativos, pueden apoyarse en las

¹<https://movielens.org/>

similitudes o diferencias entre las características del producto y las preferencias del usuario. Por tanto, podemos definir el concepto de **Argumentación** como una composición de **Polaridad** y **Comparación**.

Polaridad En cuanto a la **Polaridad**, podemos definir una explicación como:

- **Positiva.** Este tipo de explicación justifica una recomendación con argumentos positivos para el usuario. Estas explicaciones indican al usuario por qué un producto recomendado es adecuado para él. Por ejemplo: “Esta camisa es de talla media, que es la característica que especificaste”.
- **Negativa.** A veces, las explicaciones muestran características que no se ajustan a las preferencias y gustos del usuario utilizando una argumentación negativa. Por ejemplo, “Esta camiseta es verde, el color que buscas. Sin embargo, está hecha de poliéster y tú quieres camisas de algodón”.

Comparación El concepto de **Comparación** se refiere a la base argumental utilizada para justificar una recomendación.

- **Similitudes.** Estas explicaciones destacan las similitudes entre las características del producto y las preferencias del usuario. Una explicación basada en similitudes tiene polaridad positiva cuando la explicación destaca la correspondencia entre los gustos del usuario y las características del producto. Sin embargo, tiene polaridad negativa si el usuario quiere descartar productos similares porque busca la novedad o la diversidad en sus recomendaciones. Por ejemplo: “Esta camiseta es verde, tu color favorito” es una explicación basada en similitudes. Puede ser positiva si el usuario siempre quiere el mismo tipo de productos, o negativa si quiere productos con características nuevas.
- **Diferencias.** Estas explicaciones destacan las diferencias entre los productos recomendados y los intereses del usuario. Un ejemplo sería: “Esta camisa es amarilla, que no es tu color favorito, pero puede que también te guste”. Aunque este tipo de explicaciones pueden percibirse como explicaciones con polaridad negativa, también pueden ser positivas cuando la explicación debe justificar la diversidad (Ruiz-Iniesta et al., 2009) o la serendipia (Lamche et al., 2014) de la recomendación.

3.3.4.3. Nivel de detalle

El **Nivel de detalle** es el grado de especificación de la explicación. El nivel de detalle puede cambiar para diferentes explicaciones, dependiendo

de lo que el usuario necesita saber o lo que el desarrollador quiere mostrar. Consideramos que se pueden definir explicaciones de un **Único nivel** y de varios niveles (**Multinivel**), siguiendo las taxonomías propuestas por (Nunes y Jannach, 2017; Papadimitriou et al., 2012). Debido a que el nivel de detalle podría resultar un aspecto subjetivo, definimos este concepto teniendo en cuenta tres características: el número de bases de **Conocimiento** utilizadas, el número de tipos de **Formatos** de visualización y la longitud de la explicación:

Único nivel. Basándonos en los conceptos ya descritos de nuestro modelo propuesto, definimos una explicación de un solo nivel como aquella que tiene una sola base de **Conocimiento**, tiene un único **Formato** para mostrar la explicación y/o es una explicación escueta con pocos detalles. Por ejemplo, las explicaciones propuestas por Herlocker et al. (2000) se consideran de un solo nivel en nuestro modelo porque muestran un solo tipo de visualización y muy pocos detalles. Un ejemplo gráfico de este tipo de explicaciones se muestra en la Figura 3.5.

Multinivel. Por otro lado, una explicación **Multinivel** utiliza varias fuentes de **Conocimiento**, tiene diferentes **Formatos** de visualización y/o es una explicación larga con muchos detalles. Por ejemplo, la nube de etiquetas presentada por Gedikli et al. (2014) es una explicación multinivel porque utiliza dos fuentes de conocimiento diferentes basadas en el contenido, cuando sólo muestran el texto de las etiquetas, y las preferencias, cuando también muestran si al usuario le gusta o no esa característica (Figura 3.3).

3.3.4.4. Interacción

Los diferentes enfoques de **Interacción** en una explicación dependen de cómo el usuario obtiene las explicaciones a través del sistema (Tintarev y Masthoff, 2015; Eiband et al., 2018). Los tipos de **Interacción** definen las siguientes instancias:

Proactivo. El sistema se encarga de iniciar la explicación. Se le proporciona la recomendación al usuario, y el usuario no tiene que pedir una justificación porque la explicación se muestra por defecto junto a la recomendación. Encontramos ejemplos de explicaciones proactivas en plataformas comerciales como Amazon, eBay o Netflix. En la Figura 3.6, resaltado en amarillo, podemos ver cómo se muestran en Amazon.

Reactivo. En este caso, el usuario pide una explicación de la recomendación recibida. Por ejemplo, en TopCase (McSherry, 2005), encontramos un ejemplo de sistema conversacional de recomendaciones turísticas en el que los usuarios reciben explicaciones de las recomendaciones si las piden.

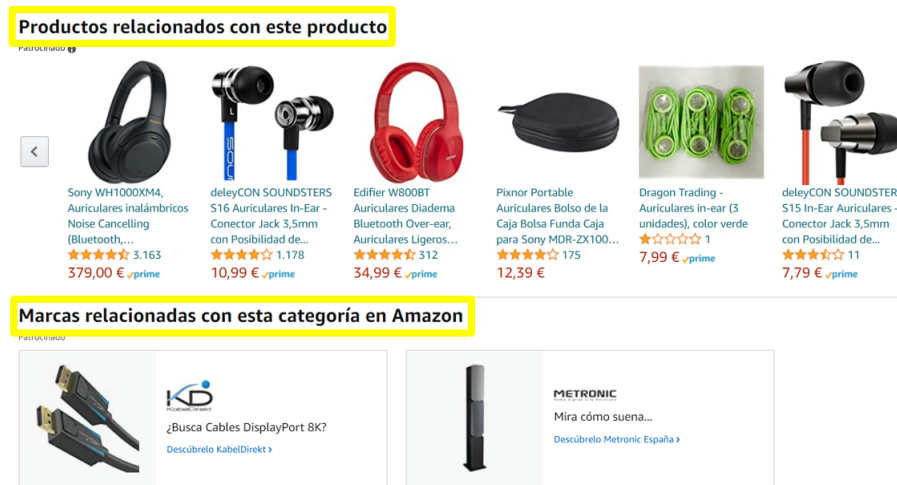


Figura 3.6: Ejemplo de una explicación Proactiva en Amazon: el propio sistema aporta una explicación junto a la recomendación.

Interactivo. La explicación ya está disponible para el usuario (de forma Proactiva o Reactiva), y éste puede interactuar con la explicación para obtener más información. Por ejemplo, Shopr (Lamche et al., 2014), permite a los usuarios interactuar con las explicaciones para mejorar su experiencia de usuario.

3.4. Validación del modelo

La validación tiene como objetivo demostrar la completitud y capacidad descriptiva de nuestro modelo probando si es posible representar en él un conjunto muy amplio y heterogéneo de sistemas de explicación para recomendadores. Para ello, seguimos un proceso iterativo en el que cada modelo intermedio creado fue evaluado, replicando el proceso de validación utilizado por Friedrich y Zanker (2011) y He et al. (2016). En cada iteración se intentaba catalogar todos los sistemas del conjunto de evaluación. Si alguno de ellos no se podía clasificar, se refinaba el modelo. El proceso finalizó cuando nuestro modelo propuesto fue capaz de clasificar todos los sistemas del conjunto de validación.

3.4.1. Metodología

Este conjunto de validación comprende 51 sistemas de explicación descritos en la literatura. El criterio utilizado para crear el conjunto fue incluir la mayor heterogeneidad posible en los sistemas de explicación seleccionados.

Sistema	Título & Referencia
E_1	“Explainable Movie Recommendation Systems by using Story-based Similarity” (Lee y Jung, 2018)
E_2	“Explaining Recommendations by Means of User Reviews” (Donkers et al., 2018)
E_3	“Explaining Complex Scheduling Decisions” (Ludwig et al., 2018)
E_4	“Explaining Contrasting Categories” (Pazzani et al., 2018)
E_5	“Explaining smart heating systems to discourage fiddling with optimized behaviour” (Stumpf et al., 2018)
E_6	“The design and validation of an intuitive confidence measure” (van der Waa et al., 2018)
E_7	“Interactive Explanations in Mobile Shopping Recommender Systems” (Lamche et al., 2014)
E_8, E_9, E_{10}	“A Review of Explanation and Explanation in Case-Based Reasoning” (Doyle et al., 2003)
E_{11}	“How should I explain? A comparison of different explanation types for recommender systems” (Gedikli et al., 2014)
$E_{12} - E_{18}$	“Explaining Collaborative Filtering Recommendations” (Herlocker et al., 2000).
E_{19}	“Tagsplanations: Explaining Recommendations Using Tags” (Vig et al., 2009)
E_{20}	“A case-based reasoning system for aiding detection and classification of nosocomial infections” (Gómez-Vallejo et al., 2016)
E_{21}	“A framework for Explanation of Machine Learning Decisions” (Brinton, 2017)
E_{22}	“Great Explanations: Opinionated Explanations for Recommendations” (Muhammad et al., 2015)
E_{23}	“Knowledge-based systems, viewpoints and the world wide web” (Finch, 1998)
E_{24}	“More Than the Sum of Its Members: Challenges for Group Recommender Systems” (Jameson, 2004)
E_{25}	“Make it personal: A social explanation system applied to group recommendations” (Quijano-Sanchez et al., 2017)
E_{26}	“Crowd-based personalized natural language explanations for recommendations” (Chang et al., 2016)
E_{27}	“Interactive recommender systems: A survey of the state of the art and future research challenges and opportunities” (He et al., 2016)
E_{28}	“Providing Control and Transparency in a Social Recommender System for Academic Conferences” (Tsai y Brusilovsky, 2017)
E_{29}	“Who to Follow and Why: Link Prediction with Explanations” (Barbieri et al., 2014)
E_{30}	“Explanation as a topic of AI research” (Buchanan y Shortliffe, 1984)
E_{31}	“Thinking Positively - Explanatory Feedback for Conversational Recommender Systems” (McCarthy et al., 2004)
E_{32}	“Trust-inspiring explanation interfaces for recommender systems” (Pu y Chen, 2007)
E_{33}	“Generating explanations in context: The system perspective” (Mittal y Paris, 1995)
E_{34}	“Do social Explanations Work? Studying and Modelling the effects of Social Explanations in Recommender Systems” (Sharma y Cosley, 2013)
E_{35}	“Providing Justifications in Recommender Systems” (Symeonidis et al., 2008) and “MovieExplain: A recommender System with Explanations” (Symeonidis et al., 2009)
E_{36}	“The Effectiveness of Personalized Movie Explanations: An Experiment Using Commercial Meta-data” (Tintarev y Masthoff, 2008)
E_{37}	“Explanation in recommender systems” (McSherry, 2005)
E_{38}	“Knowledgeable Explanations for Recommender systems” (Zanker y Ninaus, 2010)
E_{39}	“Explaining Recommendations Using Contexts” (Sato et al., 2018)
E_{40}	“Exploring explanations for matrix factorization recommender systems” (Rastegarpanah et al., 2017)
E_{41}	“User acceptance of knowledge-based system recommendations: Explanations, arguments, and fit” (Giboney et al., 2015)
E_{42}	“Tagcloud-based Explanation with Feedback for recommender systems” (Chen et al., 2013)
E_{43}	“ExpLOD: a Framework for Explaining Recommendations based on the Linked Open Data Cloud” (Musto et al., 2016)
E_{44}	“A Probabilistic Rating Prediction and Explanation Inference Model for Recommender Systems” (Wang et al., 2016)

E_{45}	“Providing Explanations for Recommendations in Reciprocal Environments” (Kleinerman et al., 2018)
E_{46}	“PRINCE: Provider-side Interpretability with Counterfactual Explanations in Recommender Systems” (Ghazimatin et al., 2020)
E_{47}	“A Model of Social Explanations for a Conversational Movie Recommendation System” (Pecune et al., 2019)
E_{48}	“Generating post hoc review-based natural language justifications for recommender systems” (Musto et al., 2020)
E_{49}	“Generate neural template explanations for recommendation” (Li et al., 2020)
E_{50}	“A Model-Agnostic Recommendation Explanation System Based on Knowledge Graph” (Chen y Miyazaki, 2020)
E_{51}	“ASISTO: An integrated intelligent assistant system for power plant operation and training” (Reyes et al., 2011)

Tabla 3.1: Colección de sistemas de explicación validados con el modelo conceptual.

Se seleccionaron trabajos clásicos y propuestas nuevas, métodos tradicionales y técnicas innovadoras. Aunque el conjunto de validación contiene algunos sistemas de explicación para sistemas de toma de decisiones, la mayoría de los sistemas seleccionados eran sistemas de explicaciones para recomendadores. En la Tabla 3.1 se enumeran los sistemas evaluados junto a la publicación donde se presentan. A continuación, se hace un análisis de los resultados obtenidos en la validación.

3.4.2. Resultados y discusión

La clasificación resultante de la validación se presenta en la Tabla 3.2, cuyas abreviaturas se muestran en la Tabla 3.3. Los guiones (“-”) que se encuentran en algunas celdas de la Tabla 3.2 representan que no tenemos suficiente información para clasificar el sistema estudiado en ese concepto. E_5 , E_7 , E_{22} , E_{28} , E_{32} , E_{37} , E_{41} , E_{45} y E_{47} son los únicos sistemas que se pueden clasificar en las 4 dimensiones definidas por nuestro modelo: **Motivación**, **Conocimiento**, **Generación** y **Presentación**.

Desde el punto de vista de la **Motivación** del sistema de explicación (Figura 3.7), en la mayoría de los casos estudiados el **Usuario** que recibe la explicación es un **Usuario objetivo individual**. Sólo hemos encontrado dos propuestas cuyo usuario es un **Stakeholder** (E_{23} y E_{26}), aunque en ambos casos también explican las decisiones a un **Usuario objetivo**. Además, se han encontrado dos ejemplos con un **Grupo** como tipo de usuario. En cuanto al **Objetivo** principal de las explicaciones, observamos que muchos de los estudios no especifican ninguno. Unos pocos trabajos sólo describen sus objetivos de **Bajo nivel**. El objetivo de **Educación** parece ser uno de los menos comunes, ya que sólo hemos encontrado un ejemplo que lo contempla. Según nuestros resultados, los objetivos más habituales son la **Transparencia** y la **Efectividad**.

Con respecto a la fuente de **Conocimiento** (Figura 3.8), la mayoría de

SYS	Usuario	Motivación		Conocimiento		Generación		Presentación				
		Objetivo		DA	AL	TE	EN	FOR	ARG		DET	INT
		LL	TL						CP	PO		
E ₁	-	-	-	-	-	KI	PH	-	-	-	-	-
E ₂	-	T	-	DO	-	ML	PH	-	-	-	-	-
E ₃	TUS	-	-	C	-	KI	ANH	NL/V	SI	P	MTL	PR/RE
E ₄	TUS	-	-	C	-	ML	ANH	NL/V	SI	P	MTL	PR
E ₅	TUS	TR	-	PREF	-	KI	ANH	NL/SC/V	SI	P	MTL	PR/RE
E ₆	TUS	TR	-	HI	-	KI	PH	SC	-	-	SL	-
E ₇	TUS	T/SCR	-	PREF	-	AH	PH	NL/V	SI/DI	P/N	MTL	PR/RE/IN
E ₈	TUS	-	-	HI	-	KI	ANH	NL	SI	P	MTL	PR
E ₉	TUS	-	-	HI	-	KI	ANH	SC	SI	P	-	PR
E ₁₀	-	-	-	HI	-	KI	ANH	-	-	-	-	-
E ₁₁	TUS	-	-	C/PREF	-	ML	PH	V	SI/DI	P/N	MTL	-
E ₁₂	TUS	EF/SA/T	-	-	US	ML	ANH/PH	V	SI	P	SL	-
E ₁₃	TUS	EF/SA/T	-	-	US	ML	ANH	V	SI	P	SL	-
E ₁₄	TUS	EF/SA/T	-	-	US	ML	ANH	SC	SI	P	SL	-
E ₁₅	TUS	EF/SA/T	-	-	-	ML	ANH	SC	SI	P	SL	-
E ₁₆	TUS	EF/SA/T	-	-	US	ML	ANH	SC	SI	P	SL	-
E ₁₇	TUS	EF/SA/T	-	COM	-	ML	ANH	SC	SI	P	SL	-
E ₁₈	TUS	EF/SA/T	-	COM	-	ML	ANH	SC	SI	P	SL	-
E ₁₉	TUS	T/EF/EFY	-	C/PREF	-	AH	PH	V	SI	P	MTL	-
E ₂₀	TUS	-	-	HI	-	KI	ANH	-	-	-	-	-
E ₂₁	-	-	-	C	-	AH	PH	NL	SI	P	SL	-
E ₂₂	TUS	EFY/PE	-	DO/HI/PREF	-	KI	ANH	SC	SI/DI	P/N	MTL	PR/IN
E ₂₃	TUS/STA	-	-	-	RP	KI	ANH	SC	SI/DI	P/N	SL	-
E ₂₄	TUG	-	-	PREF	-	KI	ANH	SC/V	SI	P	MTL	PR
E ₂₅	TUG	PE/EFY/TR	-	S/HI/DO	-	KI	PH	NL/V	SI/DI	P/N	MTL	-
E ₂₆	TUS/STA	EFY/EF/TR/SA	-	C/PREF	-	ML	PH	NL	SI	P	SL	-
E ₂₇	TUS	SA/TR/T	-	PREF/CO	US	-	-	SC/V	-	-	-	IN
E ₂₈	TUS	T	-	C	US/IS	AH	ANH	NL/SC/V	SI	P	MTL	IN
E ₂₉	-	-	-	C/S	-	AH	PH	-	SI	P	-	-
E ₃₀	TUS	-	-	-	RP	KI	ANH	NL	SI/DI	P/N	MTL	IN
E ₃₁	TUS	-	-	PREF	-	KI	PH	SC/V	SI/DI	P	MTL	PR/IN
E ₃₂	TUS	TR/EF/EFY	IU	C/PREF/COM	-	KI	ANH/PH	SC	SI/DI	P	MTL	IN
E ₃₃	TUS	EF	-	CO	RP	KI	ANH	NL	-	-	SL	PR/IN
E ₃₄	TUS	PE	-	S	-	ML	-	V	SI	P	SL	PR
E ₃₅	TUS	-	JR	PREF	-	ML	ANH	NL	SI	P	SL	-
E ₃₆	TUS	EF/TR/PE/T	-	COM/PREF/C	-	-	PH	NL	SI/DI	P/N	SL	-
E ₃₇	TUS	T	JR	PREF/HI	-	KI	ANH	NL	SI	P	SL	PR/IN
E ₃₈	TUS	T	-	PREF	-	KI	PH	NL	SI/DI	P	SL	-
E ₃₉	TUS	PE/EF	-	CO	-	MM	ANH	NL	SI	P	SL	-
E ₄₀	TUS	T	-	-	IS	MM	PH	SC	SI	P	SL	-
E ₄₁	TUS	EF	-	-	RP	KI	ANH	NL	SI	P	SL	PR
E ₄₂	TUS	EF/SA	-	C/PREF	-	ML	ANH	V	-	P/N	SL	PR
E ₄₃	TUS	T/PE/TR/SA/EF	-	PREF	IS	ML	PH	NL	SI	P	-	-
E ₄₄	TUS	EF/PE/SA	-	PREF/DO	IS	MM	ANH	-	SI	P	-	-
E ₄₅	TUG	SA/TR/T/EF	-	PREF	-	KI/MM	PH	NL/SC/V	SI	P	SL	PR
E ₄₆	TUS	SCR/EF	-	HI/C	-	KI	ANH	NL	SI	P	SL	-
E ₄₇	TUS	T/TR/EF/SA	IU	C	IS	KI	PH	NL	SI	P/N	MTL	RE/IN
E ₄₈	TUS	T/TR/EF	JR/IUE	C/CO	-	KI	PH	NL	SI	P	MTL	-
E ₄₉	TUS	-	JR	C	-	ML	ANH	NL	SI/DI	P/N	MTL	-
E ₅₀	TUS	EF	JR	C	IS	KI	PH	NL	SI	P	SL	-
E ₅₁	TUS	ED	-	DO	RP	KI/MM	ANH	SC/V	-	P	MTL	-

Tabla 3.2: Clasificación de los sistemas del conjunto de validación de acuerdo al modelo conceptual propuesto.

Concepto	Término	Acrónimo
	Sistema	SYS
Motivation	Usuario Objetivo Individual	TUS
	Usuario Objetivo Grupo	TUG
	Stakeholders	STA
	Desarrolladores	DEV
	Organismo Regulador	BD
	Objetivos Alto nivel	TL
	Objetivos Bajo nivel	LL
	Transparencia	T
	Escrutabilidad	SCR
	Confianza	TR
	Persuasión	PE
	Efectividad	EF
	Eficiencia	EFY
	Satisfacción	SA
	Educación	ED
	Debugging	DEB
	Mejorar Retención del Usuario	IU
	Mejorar Experiencia del Usuario	IUE
Justificación de la Recomendación	JR	
Conocimiento	Algoritmo	AL
	Datos	DA
	Contenido	C
	Comunidad	COM
	Preferencias	PREF
	Dominio	DO
	Historial	HI
	Contexto	CO
	Social	S
	Proceso de Recomendación	RP

Concepto	Término	Acrónimo
Conocimiento	Similitud del Usuario	US
	Similitud del Producto	IS
Generación	Técnica	TE
	Encapsulamiento	EN
	Ante hoc	ANH
	Post hoc	PH
	Basadas en Conocimiento	KI
	Machine Learning	ML
	Modelos Matemáticos	MM
	Ad hoc	AH
	Formato	FOR
	Lenguaje natural	NL
Presentación	Esquemático	SC
	Visual	V
	Otros	O
	Argumentación	ARG
	Comparación	CP
	Polaridad	PO
	Positiva	P
	Negativa	N
	Similitudes	SI
	Diferencias	DI
	Nivel de detalle	DET
	Detalle Unico nivel	SL
	Detalle Multinivel	MTL
	Interacción	INT
	Proactivo	PR
Reactivo	RE	
Interactivo	IN	

Tabla 3.3: Leyenda para la Tabla 3.2 y Figuras 3.7-3.10.

los trabajos estudiados la especifican. En las publicaciones revisadas en esta validación se utilizan más las fuentes de conocimiento basadas en **Datos** que las basadas en **Algoritmos**. Hay algunos resultados evidentes en la tabla: cuando el sistema utiliza CBR, la fuente de conocimiento es **Historial** y, cuando el sistema utiliza filtrado colaborativo, la fuente de conocimiento más común es **Similitud del usuario**. El tipo **Contenido** parece ser una de las fuentes de conocimiento más utilizadas, y se suele utilizar en combinación con **Preferencias**. Encontramos también algunos sistemas de explicación basados en las restantes opciones, de forma individual o combinada.

Dos columnas representan el concepto **Generación** en la Tabla 3.2 que resumen si el sistema es **Ante hoc** o **Post hoc** (columna **Encapsulamiento**) y el tipo de algoritmo que genera las explicaciones (columna **Técnica**). Se puede ver en la Figura 3.9 que, en nuestro estudio, los sistemas **Ante hoc** son más numerosos que los **Post hoc**. En cuanto a la **Técnica** de explicación, el más utilizado es el **Basado en conocimiento**, seguido de **Machine learning**. También es destacable que hay algunos sistemas cuyos algoritmos son **Ad hoc**, y pocos que utilizan un **Modelo matemático**.

Con respecto al concepto **Presentación**, hemos encontrado resultados variados (Figura 3.10). Varios sistemas presentan dos **Formatos**: **Lenguaje natural** y **Visual**. En este caso, el **Lenguaje natural** se utiliza cuando se explica una recomendación (o decisión) por defecto, y el **Visual**, pa-

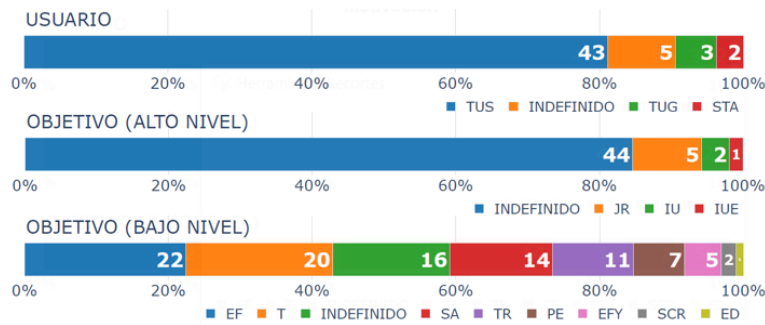


Figura 3.7: Número de sistemas clasificados en la validación de acuerdo al concepto **Motivación** definido en el modelo conceptual propuesto. El significado de las abreviaturas que se muestran en la imagen se encuentra en la Tabla 3.3.

ra explicar con más detalles cuando los usuarios lo solicitan (explicaciones **Reactivas**). En cuanto al concepto de la **Argumentación**, podemos observar que en todos los métodos estudiados, las **Similitudes** se utilizan habitualmente con la argumentación **Positiva**, mientras que las **Diferencias** se emplean mayoritariamente con la argumentación **Negativa**. También se puede concluir que la argumentación **Positiva**, con **Similitudes**, es la más extendida, lo cual es lo más esperable. Por otro lado, el **Nivel de detalle** también varía. Tenemos algunos métodos de explicación muy detallados y otros poco detallados. Por último, no hemos encontrado demasiada información sobre el concepto de **Interacción**. Hemos descubierto que la mayoría de los métodos **Reactivos** son también **Proactivos**. Aún así, el tipo de interacción más habitual parece ser la **Proactiva**. Cabe destacar que sólo hemos encontrado unos pocos sistemas de explicación **Interactivos**.

A partir de la validación, se pueden obtener dos conclusiones principales. Por un lado, nuestro modelo conceptual cumple la tarea de clasificar los sistemas de explicación de un amplio y heterogéneo conjunto de validación. Por otro lado, el modelo contiene algunos conceptos para los que no tenemos ejemplos en la Tabla 3.2. Por ejemplo, no hay ningún método cuyo **Usuario** sea un **Organismo regulador** o un **Desarrollador**. Tampoco hay ningún sistema que contemple el **Objetivo de Debugging**. Aunque estos conceptos son necesarios en el diseño de un sistema de explicación según la literatura científica más teórica, no hemos encontrado ningún sistema que cubra estos conceptos durante la revisión de la literatura. Por lo tanto, pensamos que el modelo propuesto es lo suficientemente completo no sólo para clasificar los sistemas de explicación cuyas características son las más extendidas hasta el momento en la literatura científica, sino para clasificar otros sistemas cuyos conceptos aún no están tan extendidos o están asociados a su despliegue o aplicación práctica.

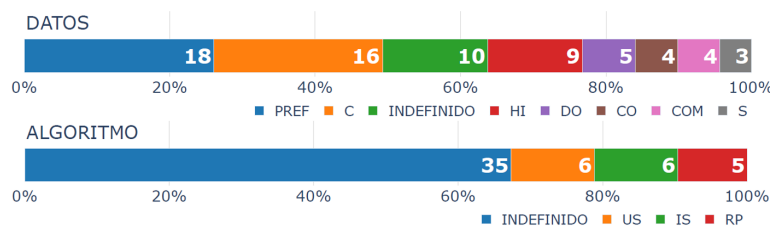


Figura 3.8: Número de sistemas clasificados en la validación de acuerdo al concepto **Conocimiento** definido en el modelo conceptual propuesto. El significado de las abreviaturas que se muestran en la imagen se encuentra en la Tabla 3.3.

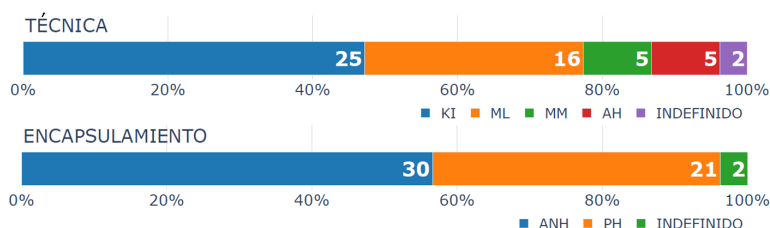


Figura 3.9: Número de sistemas clasificados en la validación de acuerdo al concepto **Generación** definido en el modelo conceptual propuesto. El significado de las abreviaturas que se muestran en la imagen se encuentra en la Tabla 3.3.

3.5. ExRecOnto: implementación ontológica del modelo conceptual

El objetivo de formalizar nuestro modelo conceptual es crear una ontología que pueda ser empleada en el desarrollo de sistemas de explicación para sistemas de recomendación. El desarrollo de software basado en ontologías establece el diseño e implementación de software utilizando una ontología como guía. La propia ontología funciona como una metodología para crear el software (Pan et al., 2012; Knublauch, 2004; Mavetera y Kroeze, 2010; Murtazina y Avdeenko, 2020).

Se ha implementado la ontología asociada al modelo descrito anteriormente mediante el lenguaje OWL, un lenguaje diseñado para representar conocimiento complejo en la Web Semántica (McGuinness et al., 2004). Elegimos este lenguaje por su popularidad y por el alto nivel de expresividad de su sublenguaje OWL-DL, el cual está basado en el razonamiento de la Lógica de Descripción o *Description Logics* (DL) (Parsia y Sirin, 2004).

Las DLs son un conjunto de lenguajes para representar conocimiento que se caracterizan por su expresividad y su semántica claramente definida (Nardi et al., 2003; Baader et al., 2008). Representan el significado de los datos en conceptos conectados por relaciones. Más importantes que las caracte-

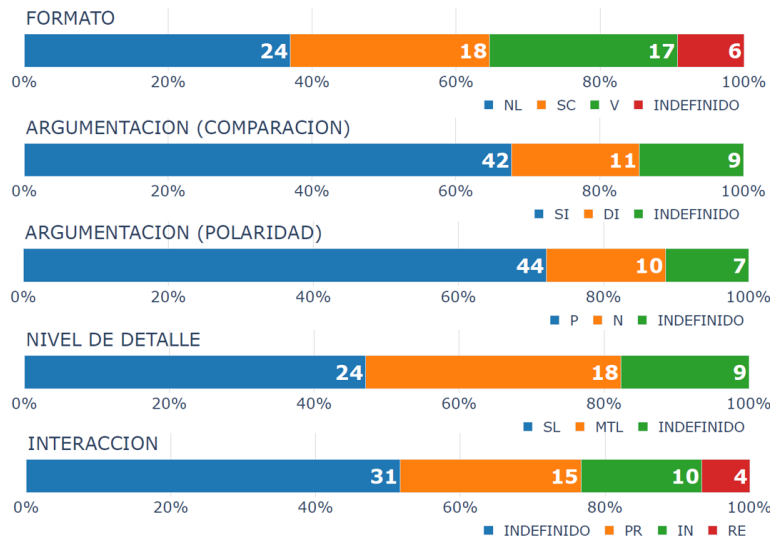


Figura 3.10: Número de sistemas clasificados en la validación de acuerdo al concepto **Presentación** definido en el modelo conceptual propuesto. El significado de las abreviaturas que se muestran en la imagen se encuentra en la Tabla 3.3.

rísticas de representación de las DL son sus mecanismos de razonamiento. Éstos permiten inferir si una descripción es más general que otra y reconocer instancias, con el objetivo de inferir los conceptos que satisface un individuo y las relaciones que satisface una tupla de individuos (Nardi et al., 2003; Baader et al., 2008).

La ontología ExRecOnto es la implementación OWL-DL de nuestro modelo conceptual para diseñar sistemas de explicación en recomendadores, descrito en la Sección 3.3 y esquematizado en la Figura 3.1. En esta ontología, representamos los conceptos, la jerarquía y las relaciones entre estos conceptos definidos en nuestro modelo. Se puede acceder a una visualización interactiva de ExRecOnto (Figura 3.11)² y acceder al repositorio GitHub donde se encuentra el código fuente³.

Las Figuras A.1, A.2, A.3 y A.4 del Apéndice A muestran esquemáticamente los componentes, jerarquía y relaciones de los cuatro conceptos principales de la ontología: **Motivación**, **Conocimiento**, **Generación** y **Presentación**. Estas figuras también incluyen las instancias que pertenecen a cada concepto. Dejando de lado las relaciones estándar de cualquier ontología OWL (*has-individual*, *has-subclass*, etc.) definimos otras relaciones que son necesarias para formalizar nuestro modelo. Mostramos cómo estas relaciones conectan y describen los conceptos de ExRecOnto en la Figura A.5:

²<http://gaia.fdi.ucm.es/ontologies/#exreconto>

³<https://github.com/martcaro/ExRecOnto>

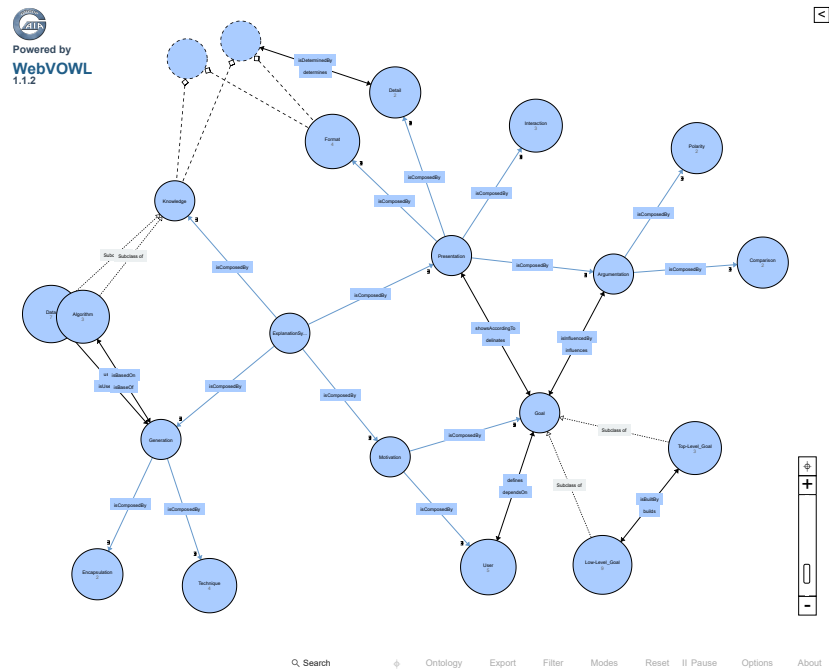


Figura 3.11: Visualización interactiva de ExRecOnto.

- *isComposedBy*. Considera la relación *composición* que hemos definido antes en la Sección 3.2.2.
- *builds* e *isBuiltBy*. Son las relaciones entre los Objetivos de bajo nivel y los de alto nivel: los Objetivos de bajo nivel de una explicación construyen (*builds*) los Objetivos de alto nivel.
- *defines* y *dependsOn*. Estas son las relaciones que se establecen entre Usuario y Objetivo: el usuario define (*defines*) el objetivo de la explicación. Dependiendo de cuál sea el tipo de usuario que va a recibir la explicación, el objetivo de la misma será uno u otro. Por ejemplo, para los *Desarrolladores* será más interesante el objetivo *Debugging*, para los *Stakeholders*, la *Persuasión*, y para los *Usuarios objetivo*, la *Confianza*.
- *influences* e *isInfluencedBy*. El Objetivo de la explicación influye (*influences*) en la *Argumentación*.
- *delinates* y *showsAccordingTo*. La *Presentación* de una explicación tiene una relación con su Objetivo: el objetivo influye (*delinates*) en la presentación. El tipo de *Argumentación* especialmente, como hemos definido antes con *influences* e *isInfluencedBy*, aunque también el resto de conceptos de presentación se pueden ver influidos por el objetivo de

la explicación. Por ejemplo, si el objetivo es aumentar la **Satisfacción del usuario** o **Mejorar su retención**, puede que sea más beneficioso usar un sistema **Visual** e **Interactivo**, que capte la atención del usuario.

- *determines* e *isDeterminedBy*. El **Formato** y el **Conocimiento** determinan (*determines*) el **Nivel de detalle**, como hemos comentado anteriormente, en la Sección 3.3.
- *isBasedOn* e *isBaseOf*. Estas relaciones muestran la conexión entre nuestros conceptos **Generación** y **Algoritmo**: si la base de conocimiento de la explicación es el **Algoritmo**, entonces este conocimiento (*isBaseOf*) en cómo se genera la explicación.
- *isUsedBy* y *uses*. En la implementación de la explicación se utilizan los **Datos** disponibles en la fuente de **Conocimiento** (*isUsedBy*) para su **Generación**.

A continuación, mostramos algunos ejemplos de cómo ExRecOnto puede ayudar a determinar algunas de las características del sistema de explicación a diseñar dependiendo de las restricciones que se impongan. Para ello se utilizan las capacidades de razonamiento de las DL.

Por ejemplo, podemos determinar la **Técnica** del sistema de explicación dependiendo del **Conocimiento** a utilizar. Esto puede resultar útil para los desarrolladores, ya que pueden acotar sus opciones de implementación dependiendo de la fuente de conocimiento a la que puedan acceder o la que quieran usar. En la Figura 3.12a se muestra como ExRecOnto, a través de la relación *uses*, concluye que si queremos utilizar el **Historial** como fuente de **Conocimiento** para desarrollar un sistema de explicación, entonces la **Técnica** a implementar debería ser **Basada en conocimiento**. Con la relación *isBasedOn*, ExRecOnto es capaz de inferir que la **Técnica** a generar, si usamos la **Similitud del usuario** o la **Similitud del producto** como fuente de **Conocimiento**, podría ser cualquier de las definidas en nuestro modelo, ya que esta fuente de conocimiento es más flexible (Figura 3.12b).

Otro ejemplo del uso de ExRecOnto sería especificar qué tipo de **Usuario** sería el de nuestro sistema dependiendo del **Objetivo** que queramos establecer. Conocer los usuarios del sistema en función de los objetivos puede resultar útil para determinar otras características del sistema. Si, por ejemplo, sabemos que el objetivo del sistema es **Debugging** y, por tanto, nuestros usuarios van a ser **Desarrolladores**, puede que no haga falta generar una explicación **Visual**. Sin embargo, si sabemos que nuestros usuarios son **Usuarios objetivo**, entonces no deberíamos descartar la opción de generar una explicación **Visual**. ExRecOnto infiere este conocimiento utilizando la relación *dependsOn*. En el ejemplo de la Figura 3.12c, ExRecOnto determina que si buscamos que nuestro sistema de explicación sea **Efectivo**, los **Usuarios** que



Figura 3.12: Ejemplo de uso de ExReconto.

pueden estar interesados en utilizar el sistema son un **Organismo regulador** o un **Usuario objetivo**, ya sea un **Usuario individual** o un **Grupo** de usuarios. En cambio, si el **Objetivo** del sistema es la **Persuasión**, entonces los **Stakeholders** son el tipo de **Usuario** que está interesado en el desarrollo de este sistema de explicación (Figura 3.12d).

3.6. Conclusiones

En este capítulo se han presentado los resultados de investigación asociados al Objetivo 0. A partir de un exhaustivo análisis del estado del arte de las explicaciones para los sistemas de recomendación, en este capítulo se ha descrito un nuevo modelo conceptual para analizar y categorizar los sistemas de explicación desde diversos puntos de vista, considerando características

y conceptos de los trabajos ya existentes además de otros adicionales. Se incluye también una formalización en forma de ontología de la que carecen otros modelos.

Nuestro modelo propone y detalla cuatro dimensiones o conceptos principales que deben ser tenidos en cuenta a la hora de analizar o generar explicaciones para los sistemas de recomendación: la motivación y los objetivos del usuario, el conocimiento que requiere la generación de una explicación, el propio proceso de implementación del sistema de explicación y la presentación al usuario. En este capítulo se han detallado estos conceptos y definido sus dependencias. Además, se ha recopilado una extensa lista de sistemas de explicación encontrados en la literatura, en su mayoría para sistemas de recomendación, y se han clasificado, según el modelo descrito, para validarlo. Se puede concluir que el modelo propuesto es capaz de clasificar un amplio y heterogéneo conjunto de sistemas, lo que valida su utilidad en el diseño de sistemas de explicación a través de metodologías de desarrollo de software guiado por ontologías.

Se trata de un modelo que describe los sistemas de explicación de forma muy completa. Puede ser una herramienta útil como guía para diseñar y desarrollar sistemas de explicación, ya que en el modelo se contemplan todas las posibles características que los sistemas de explicación para recomendadores pueden poseer en la actualidad. Además, proporcionamos una formalización del modelo conceptual a través de la ontología ExRecOnto, utilizando el lenguaje de la web semántica OWL. De acuerdo con el desarrollo de software basado en ontologías, ExRecOnto es capaz de guiar la implementación e integración de las explicaciones en *frameworks* para desarrollar sistemas de explicación en recomendadores.

El trabajo descrito en este capítulo ha sido publicado en el workshop XCBR para el congreso *International Conference on Case-Based Reasoning* del 2018 (Caro-Martínez et al., 2018) y en la revista *Journal of Artificial Intelligence Research* (Caro-Martínez et al., 2021). Además, los derechos de propiedad intelectual de la ontología ExRecOnto se han inscrito en el *Registro Territorial de la Propiedad Intelectual*.

Una vez que hemos completado nuestro estudio de sistemas de recomendación y de explicación para recomendadores, los siguientes capítulos presentan la aportación de esta tesis doctoral en este campo, mediante la utilización de grafos de interacción tanto para generar recomendaciones (Capítulo 4) como explicaciones en sistemas de recomendación (Capítulo 5).

Capítulo 4

Sistemas de recomendación basados en grafos de interacción

Hemos visto anteriormente que los sistemas de recomendación se basan tradicionalmente en filtrado colaborativo o en contenido (Aggarwal, 2016). Los sistemas de recomendación basados en filtrado colaborativo usan como fuente de conocimiento los *ratings* con los que los usuarios han valorado productos en una plataforma (Ekstrand et al., 2011). Por su parte, los sistemas de recomendación basados en contenido utilizan las descripciones de los productos y la información del perfil del usuario (Lops et al., 2011). Sin embargo, hay ocasiones en las que no podemos aplicar estas técnicas de recomendación bien porque no podemos acceder a estas fuentes de conocimiento, o bien porque no existen.

Por consiguiente, se necesita minimizar el conocimiento requerido para generar recomendaciones. En estas situaciones el mínimo conocimiento que puede utilizar un recomendador es el de la interacción del usuario con los productos del sistema. Para solucionar este problema, proponemos dos modelos diferentes, uno basado en grafos de productos y otro en grafos de usuarios, los cuales analizamos para generar las recomendaciones. Para abordar este análisis utilizamos técnicas de predicción de enlaces o *link prediction techniques*, que son técnicas de Análisis de Redes Sociales que se utilizan para encontrar enlaces que van a aparecer en un grafo o enlaces que se han perdido (Martínez et al., 2016; Sanz-Cruzado et al., 2018). En este trabajo, se usan las técnicas de predicción de enlaces para hallar similitudes entre las entidades representadas en el grafo. Estas similitudes son útiles para encontrar recomendaciones a mostrar al usuario objetivo. Con esta propuesta, pretendemos validar la Hipótesis 1 –*las técnicas de análisis de grafos permiten generar recomendaciones explicables*– y cumplir el Objetivo 1–*diseñar, desarrollar y evaluar sistemas de recomendación basados en grafos de interacción*

en escenarios de mínimo conocimiento.

Este método presenta ventajas en comparación con las técnicas tradicionales, ya que sólo necesita conocimiento sobre las interacciones, sin otros datos adicionales, como las preferencias y los *ratings*, permitiendo implementar sistemas de recomendación en escenarios de mínimo conocimiento donde otras técnicas no se pueden aplicar. Como se ha descrito en el Capítulo 2, algunos sistemas de recomendación populares, como filtrado colaborativo basado en vecinos o en modelos, como las técnicas de aprendizaje automático (árboles de decisión, naive Bayes, redes neuronales, etc.), pueden adaptarse para proporcionar recomendaciones utilizando únicamente el conocimiento de la interacción (Aggarwal, 2016). No suponen una solución específica al problema que se pretende resolver en esta tesis doctoral, ya que, al explotar los valores de los *ratings* como fuente de conocimiento, no están diseñadas para tal objetivo, pero sí pueden obtener resultados aceptables. Sin embargo, cabe destacar que son métodos globales, lo que significa que utilizan todo el conocimiento que tienen disponible para obtener las soluciones, recomendaciones en este caso. En nuestra propuesta, utilizamos medidas de similitud locales basadas en técnicas de predicción de enlaces. Aunque los métodos locales sólo utilizan una parte de todo el conocimiento disponible, lo cual influye en la obtención de resultados ya que no se puede llegar a los resultados a los que sí llegan los globales, tomamos esta decisión en base a las ventajas que pueden aportar los métodos locales. El uso de métodos locales supone proporcionar recomendaciones más interpretables para los usuarios objetivo, lo que puede contribuir a la aceptación de la recomendación.

Interpretar los modelos globales conlleva la comprensión de la totalidad de un modelo, incluyendo todos los posibles caminos de decisión del mismo. Sin embargo, interpretar los modelos locales significa entender los resultados de un modelo entrenado sobre una entrada específica. Entender el proceso llevado a cabo para la toma de una única decisión puede resultar más sencillo que comprender la totalidad del comportamiento del modelo (Slack et al., 2019; Ribeiro et al., 2016). Además de la utilización de métodos locales, la representación del conocimiento a través de grafos puede aportar un enfoque interpretable porque los usuarios perciben los grafos como modelos de propagación de la información que se acercan más a su modelo mental. En la Sección 2.2.3 del Capítulo 2 se detallan las razones por las que los grafos tiene esta propiedad. Por lo tanto, en este trabajo pretendemos demostrar que nuestras propuestas son tan buenas como las técnicas de recomendación clásicas –o incluso mejores– cuando sólo pueden acceder a un conocimiento mínimo, teniendo la ventaja de ser un modelo más explicable.

La Sección 4.1 describe los sistemas de recomendación basados en grafos. Primero, se comenta cómo se utiliza la información de las interacciones y cómo se generan los grafos de interacción para crear los dos métodos de recomendación distintos: el método basado en grafos de productos, detallado

en la Sección 4.1.1, y el método basado en grafos de usuarios, el cual se define en la Sección 4.1.2. Además de la generación de los grafos de interacción, otro de los elementos clave en el desarrollo de nuestra propuesta son las técnicas de predicción de enlaces. En la Sección 4.1.3, comentamos cuáles son las medidas basadas en estas técnicas que usamos en nuestros métodos. También describimos en la Sección 4.1.4 los métodos de agregación necesarios para que nuestros modelos generen recomendaciones adecuadamente.

Para comprobar la efectividad de los grafos de interacción generando recomendaciones, se ha evaluado su precisión comparándola con la de otras técnicas clásicas de recomendación. La Sección 4.2 detalla cómo hemos llevado a cabo esta evaluación. Tras esto, se analizan los resultados obtenidos (Sección 4.3) y se obtienen unas conclusiones globales de este análisis (Sección 4.4). Al final del capítulo, extraemos las conclusiones del trabajo realizado (Sección 4.5).

4.1. Recomendaciones basadas en grafos de interacción

En cualquier plataforma susceptible de tener un sistema de recomendación, los usuarios llevan a cabo interacciones con los productos. Esta es la información que queremos aprovechar para generar las recomendaciones. Para representar este conocimiento en grafos, tenemos que tener en cuenta los elementos que intervienen en una interacción. Esta información sobre las interacciones en un sistema de recomendación se puede representar como una tupla $R = (t, u, i, x)$, donde

- t es la marca de tiempo o *timestamp* en que se realizó la interacción,
- $u \in U$ es el usuario que realizó la interacción,
- $i \in I$ es el producto con el que u ha interactuado en t ,
- x es el valor asociado a la interacción. En muchos casos, x es el *rating* con el que u ha valorado a i en una interacción específica. Por ejemplo, cuando un usuario u califica un libro i en Goodreads, x es la calificación otorgada. Sin embargo, podemos tener otra información en x además de los *ratings*. Por ejemplo, en los jueces online, x es el veredicto proporcionado por la plataforma a u para una solución del problema i que u ha entregado previamente al juez.

Como hemos visto, muchas de las técnicas actuales explotan el valor de x asociado a la interacción para proporcionar una recomendación. En esta tesis doctoral se propone un método para generar recomendaciones de productos cuando los valores x asociados a la interacción no están disponibles o no son

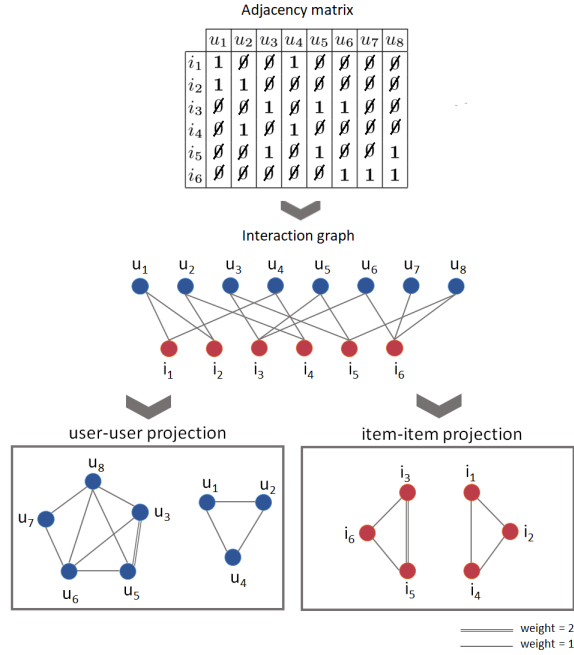


Figura 4.1: Transformación de un grafo bipartito en un grafo no bipartito usando proyección de grafos bipartitos.

útiles. Teniendo esto en cuenta, necesitamos representar la información del sistema recomendador dejando de lado los valores x , sólo considerando si u ha interactuado con i o no. Mediante esta abstracción, podemos construir un grafo que represente las interacciones como un grafo bipartito sin pesos: $G\langle N, L \rangle$. Esto significa que el conjunto de nodos N del grafo pertenecerá al conjunto de usuarios U o al conjunto de productos I . Los enlaces L entre los nodos se forman cuando un nodo que representa al usuario u ha interactuado con un nodo que representa un producto i . Por ejemplo, se formará un enlace entre un nodo de usuario u y un nodo producto i , siendo i una película, si u ha visto i . También podemos representar G como una matriz de adyacencia \mathbb{A} , donde $\mathbb{A}[u, i]$ es igual a 1 si el usuario u ha interactuado con el producto i , \emptyset en caso contrario. Por lo tanto, nuestro modelo basado en grafos es una descripción de las interacciones realizadas en el sistema: representan escenarios de mínimo conocimiento, donde únicamente se utilizan *ratings* unarios que no tienen en cuenta valores positivos o negativos en las interacciones llevadas a cabo.

Para simplificar la tarea de aplicar las técnicas de predicción de enlaces sobre el grafo de interacción, lo primero que realizamos es una proyección de grafos bipartitos o *bipartite network projection* (Zhou et al., 2007). Con este mecanismo, se transforma el grafo original en dos grafos ponderados diferentes: un grafo de nodos productos G_I y un grafo de nodos usuarios

G_U . De esta forma, obtenemos dos grafos distintos, cada uno de ellos con una única clase de nodos, lo que facilita el trabajo con el grafo. Estos dos grafos tienen pesos en los enlaces que guardan información para no perder conocimiento del grafo original. En el grafo de productos, el peso de un enlace representa el número de usuarios que han interactuado con los dos productos enlazados. En cambio, en el grafo de usuarios, el peso del enlace es el número de productos con los que ambos usuarios han interactuado. En la Figura 4.1, podemos ver un ejemplo de *bipartite network projection*.

Debido a que en general se produce una gran cantidad de interacciones en un sistema recomendador, la cantidad de enlaces en nuestros grafos de interacción también puede ser enorme. Para reducir la densidad de los grafos y, por tanto, acortar el tiempo de cálculo sin perder datos sensibles, haciendo a su vez más sencillo el manejo del propio grafo, el segundo paso en el proceso de creación de los grafos es aplicar algoritmos de filtrado de enlaces. En este caso, eliminaremos los enlaces cuyo peso sea inferior a un valor umbral θ .

De la aplicación de la proyección bipartita y del filtro de enlaces, tenemos dos grafos distintos, uno de nodos productos y otro de nodos usuarios, en los que nos basamos para generar dos métodos de recomendación diferentes, usando técnicas de predicción de enlaces. De entre todos los métodos de predicción de enlaces que existen, en este trabajo nos decantamos por usar algunas de las basadas en similitud, aquellas que son locales (véase el Capítulo 2.3). Estos métodos nos permiten cumplir nuestro propósito: nos ayudan a encontrar recomendaciones interesantes para el usuario en base a similitudes entre productos o usuarios, respectivamente, y siendo su funcionamiento interpretable por el usuario. Hemos utilizado varias medidas de similitud con el objetivo de estudiar cuáles son las que ofrecen mejor precisión. En la Sección 4.1.3 detallamos estas medidas. Es importante señalar que los grafos de interacción usados en nuestro modelo no requieren ninguna información adicional sobre los usuarios o las características de los productos. Sólo se va a usar el conocimiento intrínseco al grafo de interacción.

En las siguientes Secciones 4.1.1 y 4.1.2, describimos el proceso de recomendación llevado a cabo con el grafo de interacción basado en productos y con el grafo de interacción basado en usuarios, respectivamente, usando las técnicas de predicción de enlaces.

4.1.1. Proceso de recomendación basado en grafos de productos

El modelo de recomendación basado en el grafo de productos recomienda a un usuario u_t los productos más similares a aquellos con los que u_t ha interactuado previamente. Definimos el grafo $G_I = \langle N_I, L_I \rangle$, donde $N_I = \{n_r, \dots, n_j \mid i_r, i_j \in I, r \neq j\}$ es el conjunto de nodos que representan los productos I del sistema de recomendación y $L_I = \{(n_r, n_j, w_{rj}) \mid i_r, i_j \in I, r \neq j\}$ son los enlaces que existen entre los productos según la proyección

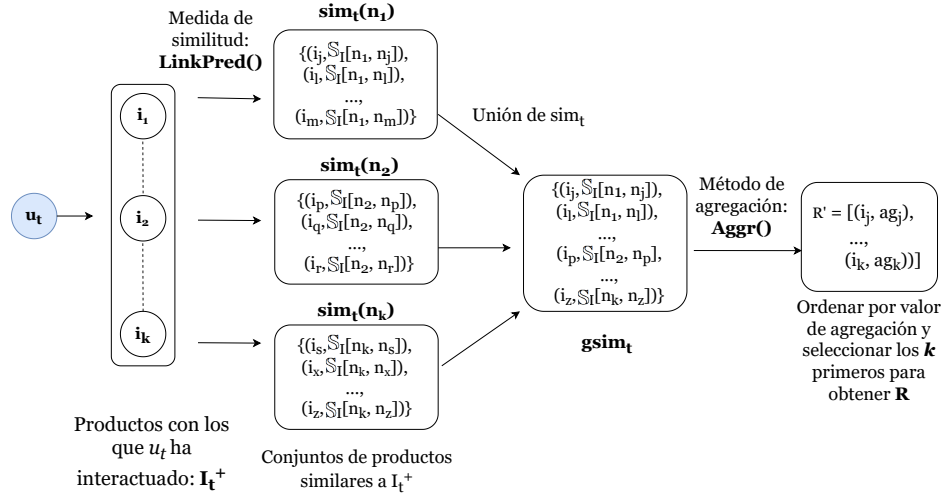


Figura 4.2: Proceso de recomendación llevado a cabo con el grafo de interacción de productos.

bipartita. Dos nodos n_r y n_j están enlazados si existe al menos un usuario que haya interactuado con ambos productos i_r e i_j . El peso del enlace, w_{rj} , representa el número de usuarios que han interactuado con i_r e i_j . Además, como hemos mencionado, eliminamos los enlaces cuyo peso es inferior a un valor umbral de θ para reducir la densidad del grafo ($w_{rj} < \theta$ para cualquiera de los enlaces de L_I).

En la Figura 4.2 se presenta una vista general del proceso de recomendación basado en el grafo de productos. Este proceso ejecuta los siguientes pasos para hacer recomendaciones para un usuario objetivo u_t :

- (P.1.) Construir una matriz de similitud S_I que almacene todos los valores de similitud entre todos los pares de nodos del grafo G_I utilizando un método de predicción de enlaces basado en similitud, $LinkPred()$. Esta matriz representa las similitudes entre los productos del sistema de recomendación. Las medidas de similitud utilizadas en este trabajo se encuentran descritas en la Sección 4.1.3.
- (P.2.) Obtener el conjunto $I_t^+ \subset I$ con los productos con los que el usuario u_t ha interactuado previamente. Por el contrario, $I_t^- = I \setminus I_t^+$ es el conjunto de productos con los que el usuario u_t aún no ha interactuado.
- (P.3.) Considerando que la función $item(n_r)$ devuelve el producto i_r representado por el nodo n_r , para cada nodo n_r tal que $item(n_r) \in I_t^+$, obtener un conjunto $sim_t(n_r)$. Este conjunto $sim_t(n_r)$ contendrá los productos más similares al producto i_r con los que u_t no haya

interactuado aún, ordenados según su similitud en \mathbb{S}_I .

$$\begin{aligned} sim_t(n_r) &= \{(i_j, s_{rj}), (i_k, s_{rk}), \dots \mid s_{rj} \geq s_{rk}\} \quad (4.1) \\ \text{donde, } \forall (i_x, s_{rx}) \in sim_t(n_r) &: \quad i_x = item(n_x) \\ &\quad i_x \in I_t^- \\ &\quad s_{rx} = \mathbb{S}_I[n_r, n_x] \end{aligned}$$

(P.4.) Unir las listas $sim_t(n_r)$ para cada n_r tal que $item(n_r) \in I_t^+$ y obtener una lista global $gsim_t$ que contenga todos los productos similares a I_t^+ , con su valor de similitud asociado, con los que u_t no ha interactuado aún. Esta lista puede contener productos repetidos pero con diferentes valores de similitud:

$$gsim_t = \bigcup_{item(n_r) \in I_t^+} sim_t(n_r) \quad (4.2)$$

(P.5.) Agregar los valores de similitud s_{rx} para aquellos productos que se repiten en la lista $gsim_t$. Hemos experimentado con diferentes métodos de agregación, $Aggr()$, con el objetivo de determinar cuál es el método que ofrece mejor precisión. Los métodos de agregación $Aggr()$ propuestos están descritos en la Sección 4.1.4. Como resultado del proceso de agregación, obtenemos la lista final R' que contiene una única aparición por cada producto similar, con un valor de similitud agregado asociado. Además ordenamos R' en orden decreciente por el valor de agregación.

$$\begin{aligned} R' &= \{(i_j, ag_j), (i_k, ag_k), \dots \mid ag_j \geq ag_k\} \quad (4.3) \\ \text{donde, } \forall (i_x, ag_x) \in R' &: \quad \{(i_x, s_{rx}), (i_x, s_{mx}), \dots, (i_x, s_{zx})\} \in gsim_t \\ &\quad i_x \in I_t^- \\ &\quad i_r, i_m, \dots, i_z \in I_t^+ \\ &\quad ag_x = Aggr(\{s_{rx}, s_{mx}, \dots, s_{zx}\}) \end{aligned}$$

(P.6.) Finalmente, se seleccionan los primeros k productos de R' para formar la lista de recomendación $R = [i_j, \dots, i_z]$. Esta será la lista de productos a recomendar para el usuario objetivo u_t .

El Algoritmo 1 muestra el proceso que se utiliza para calcular la lista de recomendación usando el método basado en grafos de interacción de productos según los pasos anteriores. A continuación, se describe el método de recomendación basado en grafos de usuarios.

Algorithm 1: Algoritmo para calcular la lista de recomendaciones utilizando el método basado en grafos de productos.

Input: $G_I = \{N_I, L_I\}, I_t^+, LinkPred(), Aggr(), k$
Output: R

```

1 foreach  $item(n_r) \in I$  do
2   | foreach  $item(n_j) \neq item(n_r) \in I$  do
3   |   |  $S_I[n_r, n_j] = LinkPred(G_I, n_r, n_j)$ 
4   |   end
5 end
6  $I_t^- = I \setminus I_t^+$ 
7  $R = \{\}, R' = \{\}, gsim = \{\}$ 
8 foreach  $item(n_r) \in I_t^+$  do
9   | foreach  $item(n_j) \in I_t^-$  do
10  |   |  $s_{rj} = S_I[n_r, n_j]$ 
11  |   |  $insert(gsim_t, (item(n_j), s_{rj}))$ 
12  |   end
13 end
14 foreach  $i_x \in gsim_t$  do
15  |  $sims = \{\}$ 
16  | foreach  $(i_x, s) \in gsim_t$  do
17  |   |  $insert(sims, s)$ 
18  |   end
19  |  $ag_x = Aggr(sims)$ 
20  |  $insert(R', (i_x, ag_x))$ 
21 end
22  $sort(R')$ 
23  $R = getFirstK(R', k)$ 
24 return  $R$ 

```

4.1.2. Proceso de recomendación basado en grafos de usuarios

En este caso, el proceso de recomendación utiliza el grafo de usuarios. Este modelo recomienda a un usuario objetivo u_t los productos con los que los usuarios más similares a u_t han interactuado previamente y con los que u_t no ha interactuado todavía. Definimos nuestro grafo como $G_u = \langle N_U, L_U \rangle$, donde $N_U = \{n_t, \dots, n_v \mid u_t, u_v \in U, t \neq v\}$ son los nodos que representan a los usuarios U del sistema. $L_U = \{(n_t, n_v, w_{tv}) \mid u_t, u_v \in U, t \neq v\}$ son los enlaces entre usuarios obtenidos a través de la proyección bipartita. En este caso, dos nodos n_t y n_v están conectados si u_t y u_v han interactuado con al menos un producto en común. El peso de un enlace w_{tv} representa el número de productos comunes con los que los usuarios u_t y u_v han interactuado.

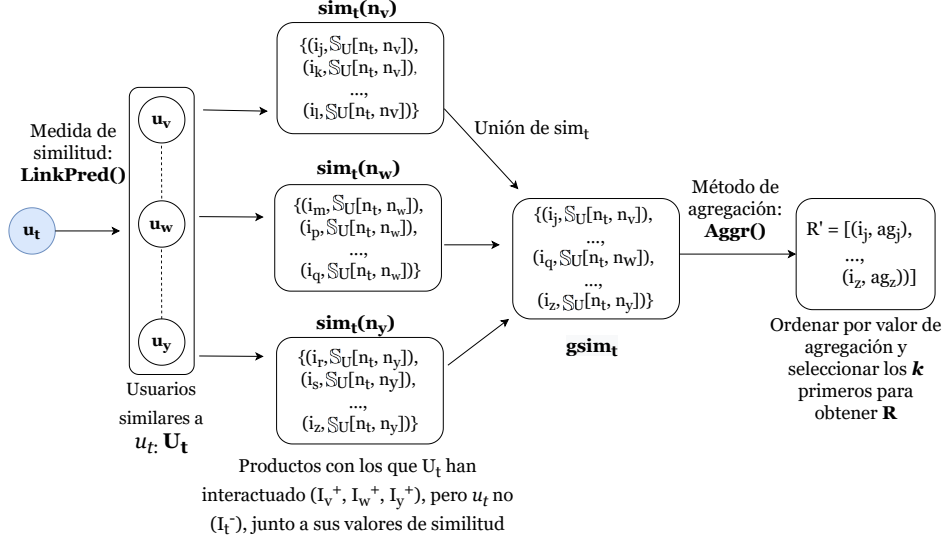


Figura 4.3: Proceso de recomendación llevado a cabo con el grafo de interacción de usuarios.

Como hicimos en el modelo basado en productos, filtramos los enlaces cuyo peso w_{tv} es inferior a un valor umbral de θ .

El proceso de recomendación basado en el grafo de usuarios planteado es análogo al proceso llevado a cabo con el modelo basado en productos. Sin embargo, en este caso, el usuario objetivo de la recomendación, a partir de ahora denotado como u_t , se representa directamente en el grafo a través de su nodo correspondiente n_t . La función $user(n_t)$ devuelve el usuario u_t . La Figura 4.3 presenta una visión general del proceso definido a continuación:

- (P.1.) Construir una matriz de similitud \mathbb{S}_U que almacene todos los valores de similitud entre cada par de nodos del grafo G_U utilizando un método de similitud de predicción de enlaces $LinkPred()$. Estos métodos de similitud son los mismos que se usaron en el modelo basado en productos y están detallados en la Sección 4.1.3. Al ser G_U un grafo de usuarios, la matriz \mathbb{S}_U contiene los valores de similitud entre los usuarios del sistema de recomendación.
- (P.2.) Obtener el conjunto $U_t \subset U$ que contiene los usuarios cuyos nodos son similares a n_t según \mathbb{S}_U . También se calcula el conjunto de los productos con los que u_t no ha interactuado ($I_t^- \in I$) y todos los conjuntos de productos con los que los usuarios similares $u_v \in U_t$ han interactuado ($I_v^+ \in I$).
- (P.3.) Para cada nodo n_v tal que $user(n_v) \in U_t$, obtener el conjunto

$sim_t(n_v)$ de productos con los que el usuario u_v ha interactuado pero el usuario u_t no. En este caso, todos los elementos de $sim_t(n_v)$ tienen el mismo valor de similitud s_{tv} correspondiente a la similitud entre ambos usuarios $\mathbb{S}_U[n_t, n_v]$. Ese conjunto se define como:

$$sim_t(n_v) = \{(i_j, s_{tv}), (i_k, s_{tv}), \dots\} \quad (4.4)$$

donde, $\forall (i_x, s_{tv}) \in sim_t(n_v) \quad : \quad i_x \in I_v^+ \cap I_t^-$
 $s_{tv} = \mathbb{S}_U[n_t, n_v]$

(P.4.) Unir los conjuntos $sim_t(n_v)$ para cada $user(n_v) \in U_t$ y obtener una lista global $gsim_t$ que contenga todos los productos (y su valor de similitud asociado) con los que u_t no ha interactuado. Esta lista puede contener productos repetidos con diferentes valores de similitud:

$$gsim_t = \bigcup_{user(n_v) \in U_t} sim_t(n_v) \quad (4.5)$$

(P.5.) A continuación, agregar los valores de similitud de los elementos que se repiten en la lista $gsim_t$ utilizando una de las funciones de agregación $Aggr()$ propuestas en la Sección 4.1.4. Como resultado, obtenemos la lista final R' donde aparece una única vez cada producto con un valor de agregación asociado. Esta lista R' se ordena decrecientemente por valor de agregación.

$$R' = \{(i_j, ag_j), (i_k, ag_k), \dots \mid ag_j \geq ag_k\} \quad (4.6)$$

donde, $\forall (i_x, ag_x) \in R' \quad : \quad \{(i_x, s_{tv}), (i_x, s_{ty}), \dots, (i_x, s_{tz})\} \in gsim_t$
 $i_x \in I_t^- \cap I_v^+ \cap I_y^+ \cap \dots \cap I_z^+$
 $ag_x = Aggr(\{s_{tv}, s_{ty}, \dots, s_{tz}\})$

(P.6.) Finalmente, escoger los primeros k productos en R' , que se van a almacenar en $R = [i_j, \dots, i_z]$ como la lista de productos a recomendar al usuario objetivo u_t .

El Algoritmo 2 presenta el algoritmo correspondiente al proceso de recomendación basado en grafos de interacción de usuarios que hemos descrito en los pasos anteriores.

En las siguientes secciones, detallamos cuáles son las medidas de similitud basadas en técnicas de predicción de enlaces y los métodos de agregación propuestos tanto en los modelos de grafos de interacción de productos como en los de grafos de interacción de usuarios

Algorithm 2: Algoritmo para el cálculo de la lista de recomendaciones mediante el método basado en grafos de interacción de usuarios.

Input: $G_U = \langle N_U, L_U \rangle, I_t^+, LinkPred(), Aggr(), k$
Output: R

```

1  $I_t^- = I \setminus I_t^+$ 
2  $R = \{\}, R' = \{\}, gsim_t = \{\}$ 
3 foreach  $user(n_v) \neq user(n_t) \in U$  do
4    $s_{tv} = \mathbb{S}_U[n_t, n_v] = LinkPred(G_U, n_t, n_v)$ 
5    $I_v^+ = getItems(user(n_v))$ 
6   foreach  $i_x \in I_t^- \cap I_v^+$  do
7      $insert(gsim_t, (i_x, s_{tv}))$ 
8   end
9 end
10 foreach  $i_x \in gsim_t$  do
11    $sims = \{\}$ 
12   foreach  $(i_x, s) \in gsim_t$  do
13      $insert(sims, s)$ 
14   end
15    $ag_x = Aggr(sims)$ 
16    $insert(R', (i_x, ag_x))$ 
17 end
18  $sort(R')$ 
19  $R = getFirstK(R', k)$ 
20 return  $R$ 

```

4.1.3. Medidas de similitud basadas en técnicas de predicción de enlaces

Las técnicas de predicción de enlaces son un conjunto de métodos del ámbito del Análisis de Redes Sociales que permiten encontrar los nuevos enlaces que van a aparecer o han desaparecido en un grafo (Lü y Zhou, 2011; Wang et al., 2015; Kumar et al., 2020; Martínez et al., 2016; Sanz-Cruzado et al., 2018). Siendo G_t un grafo en el tiempo t , las técnicas de predicción de enlaces se centran en cómo predecir la evolución de los enlaces en G_{t+1} .

Tras la revisión de las técnicas de predicción de enlaces que se ha realizado en esta tesis doctoral, y que se ha detallado en el Capítulo 2, sabemos que existen dos tipos de métodos de predicción de enlaces: los basados en aprendizaje y los basados en similitud. A su vez, existen cuatro grupos de métodos de predicción de enlaces basados en similitud. De estos cuatro grupos, en nuestra propuesta decidimos considerar únicamente medidas basadas en nodos y en vecinos. Tomamos esta decisión teniendo en cuenta la interpre-

tabilidad que las medidas de similitud nos podían proporcionar. En concreto, queríamos utilizar métodos de predicción de enlaces que convirtiesen nuestros métodos propuestos en *modelos locales* para proporcionar recomendaciones lo más explicables posible.

Los modelos locales en la inteligencia artificial explicable utilizan sólo un subconjunto del conjunto de datos para proporcionar explicaciones, mientras que los globales utilizan el conocimiento de todo el conjunto de datos, siendo también más complejos. Este subconjunto de datos de los que los modelos locales hacen uso se obtiene de los elementos vecinos al elemento a explicar (Ribeiro et al., 2016; Singh y Anand, 2019; Arya et al., 2019; Ribeiro et al., 2018). Los modelos locales son más interpretables ya que resulta más sencillo entender una única decisión tomada por el sistema, que la totalidad de su funcionamiento (Slack et al., 2019; Ribeiro et al., 2016). Los modelos locales son adecuados cuando conocemos la región de datos que nos interesa utilizar para obtener la solución de las recomendación (Hand y Vinciotti, 2003), como ocurre en nuestra propuesta, donde sabemos que los nodos cercanos a los nodos objetivo son buenos candidatos a la hora de generar recomendaciones. Aunque los métodos basados en caminos y caminos aleatorios también pueden lograr una buena precisión, no se consideran en este trabajo al ser modelos globales.

Los métodos basados en similitud asignan una puntuación de semejanza entre dos entidades del grafo, por lo que no tiene por qué tomar valores entre $[0, 1]$ como ocurre con otras medidas de similitud tradicionales. Como se ha descrito anteriormente en las Secciones 4.1.1 y 4.1.2, con nuestros modelos pretendemos construir una matriz de similitud \mathbb{S} donde almacenar las similitudes entre cada par de nodos del grafo de interacción, nuestra única fuente de conocimiento. De este modo, $\mathbb{S}[n_a, n_b] = \text{LinkPred}(G, n_a, n_b)$ es el valor de similitud entre el nodo n_a y el nodo n_b calculado por el método de predicción de enlaces $\text{LinkPred}()$ aplicado al grafo G . Cuanto mayor sea la similitud entre dos nodos no conectados, mayor será la posibilidad de que estén conectados en un futuro. En el caso de nuestra propuesta, al usar métodos locales, la similitud entre muchos de los nodos del grafo será inexistente, ya que no se dispone de información suficiente para calcular la puntuación de similitud.

En este trabajo, proponemos un conjunto de medidas de similitud extraídas de la literatura incluyendo también nuestras propias variaciones ponderadas (Liben-Nowell y Kleinberg, 2007; Wang et al., 2015; Lü y Zhou, 2010; Kumar et al., 2020; Martínez et al., 2016; Sanz-Cruzado et al., 2018; Lichtenwalter et al., 2010). Para definir estos métodos, se necesita precisar la notación a utilizar:

- $G = \langle N, L \rangle$ es el grafo de interacción, donde N es el conjunto de nodos y L , el de enlaces.

- $\mathcal{N}(n)$ representa el conjunto de vecinos del nodo $n \in N$.
- $|\mathcal{N}(n)|$ representa el número de vecinos (grado) del nodo n .
- n_a y n_b son un par de nodos cualesquiera del grafo G .
- w_{ab} representa el peso del enlace entre los nodos n_a y n_b . El peso es 0 si no hay enlace entre ambos nodos ($w_{ab} = 0$ si $(n_a, n_b) \notin L$).
- $\mathcal{W}(n_a) = \sum w_{ax} : n_a \neq n_x \in N$ representa el grado ponderado del nodo n_a , es decir, la suma de los pesos de los enlaces directamente conectados con el nodo n_a .

A continuación, detallamos las medidas de similitud basadas en predicción de enlaces (denotadas como $LinkPred()$) que hemos propuesto en esta tesis. La mayoría de ellas tienen dos versiones –una ponderada y otra no ponderada.

Edge Weight (EW). Esta medida mide la similitud entre dos nodos como el peso del enlace entre ellos.

$$LinkPred() = EW(G, n_a, n_b) = w_{ab} \quad (4.7)$$

También existe una versión no ponderada de esta medida definida como $EW(G, n_a, n_b) = 1$ si $w_{ab} \neq 0$, pero la descartamos por su evidente simplicidad.

Common Neighbours (CN). En este caso, la similitud entre dos nodos es el número de vecinos que tienen en común. Cuanto mayor sea la intersección de los conjuntos de vecinos de dos nodos cualesquiera, mayor será la probabilidad de asociación futura entre ellos. **Weighted Common Neighbours (WCN)** es la versión ponderada de esta medida.

$$LinkPred() = CN(G, n_a, n_b) = |\mathcal{N}(n_a) \cap \mathcal{N}(n_b)| \quad (4.8)$$

$$LinkPred() = WCN(G, n_a, n_b) = \sum_{n_z \in \{\mathcal{N}(n_a) \cap \mathcal{N}(n_b)\}} w_{az} + w_{zb} \quad (4.9)$$

Jaccard Neighbours (JN). Esta medida es una mejora de CN ya que mide el número de vecinos comunes de n_a y n_b en comparación con el número de vecinos totales de ambos nodos. No tiene una versión ponderada.

$$LinkPred() = JN(G, n_a, n_b) = \frac{|\mathcal{N}(n_a) \cap \mathcal{N}(n_b)|}{|\mathcal{N}(n_a) \cup \mathcal{N}(n_b)|} \quad (4.10)$$

Adar/Adamic (AA). Esta medida también mide la intersección de los conjuntos de vecinos de dos nodos en el grafo, pero hace hincapié en el conjunto solapado más pequeño. La versión ponderada de esta medida es Weighted Adar/Adamic (WAA).

$$LinkPred() = AA(G, n_a, n_b) = \sum_{n_z \in \{\mathcal{N}(n_a) \cap \mathcal{N}(n_b)\}} \frac{1}{\log|\mathcal{N}(n_z)|} \quad (4.11)$$

$$LinkPred() = WAA(G, n_a, n_b) = \sum_{n_z \in \{\mathcal{N}(n_a) \cap \mathcal{N}(n_b)\}} \frac{w_{az} + w_{zb}}{\log(1 + \mathcal{W}(n_z))} \quad (4.12)$$

Preferential Attachment (PA). Se basa en la consideración de que los nodos crean enlaces con mayor probabilidad con aquellos que tienen un mayor número de enlaces. La probabilidad de crear un enlace entre los nodos n_a y n_b se calcula como el producto del grado de los nodos n_a y n_b . Así, cuanto mayor sea el grado de ambos nodos, mayor será la probabilidad de crear un enlace. Weighted Preferential Attachment (WPA) es su versión ponderada, en la que se tienen en cuenta los pesos de los enlaces al calcular el grado de los nodos n_a y n_b .

$$LinkPred() = PA(G, n_a, n_b) = |\mathcal{N}(n_a)| \cdot |\mathcal{N}(n_b)| \quad (4.13)$$

$$LinkPred() = WPA(G, n_a, n_b) = \mathcal{W}(n_a) \cdot \mathcal{W}(n_b) \quad (4.14)$$

4.1.4. Métodos de agregación

Como hemos visto anteriormente, los modelos de recomendación basados en grafos necesitan un método de agregación para unir la lista de elementos recuperados, al que hemos denotado como $Aggr()$ en las Ecuaciones 4.3 y 4.6. Partimos de un estado en el que tenemos los conjuntos $sim_t(n_x)$ de productos a recomendar con sus valores de similitud asociados (Ecuaciones 4.2 y 4.5). Sabemos que en $gsim_t = \{(n_j, s_j), (n_k, s_k), \dots\}$, unión de los productos $sim_t(n_x)$, podemos tener productos a recomendar repetidos. El objetivo de los métodos de agregación es combinar de alguna manera la similitud de aquellos productos que se repiten en la lista $gsim_t$, tanto en el modelo basado en un grafo de productos, como en el basado en un grafo de usuarios. De esta forma, en R' tendremos una única aparición de cada producto a recomendar con un único valor de similitud agregado asociado (Ecuaciones 4.3 y 4.6). En esta tesis doctoral se proponen cuatro métodos de agregación. Por un lado tenemos el métodos de agregación basado en la similitud máxima, y por otro lado, los sistemas de votación.

Máxima similitud o *Highest Similarity* (HS). Es el método más sencillo que proponemos. El valor de agregación para un producto i_x es el mayor valor de similitud s_x que tiene i_x en todas sus apariciones en la lista $gsim_t$.

$$Aggr() = HS(gsim_t) = \max(s_x) \quad (4.15)$$

Votación simple o *Simple voting* (SV). El valor de similitud agregado ag_x es el número de veces que el producto i_x aparece en la lista de elementos $gsim_t$.

$$Aggr() = SV(gsim_t) = \text{count}(i_x) \quad (4.16)$$

Votación ponderada o *Weighted voting* (WV). El valor de agregación ag_x es la suma ponderada de los valores de similitud del producto i_x en la lista $gsim_t$.

$$Aggr() = WV(gsim_t) = \sum \frac{s_x}{\sum s_j} \quad (4.17)$$

donde s_j es cada valor de similitud asociado a $i_x \in gsim_t$.

Votación posicional o *Positional voting* (PV). En este caso, el valor de similitud agregado ag_x depende de la posición de i_x en la lista ordenada $gsim_t$:

$$Aggr() = PV(\text{sort}(gsim_t)) = \frac{1}{\sum \text{pos}(i_x)} \quad (4.18)$$

donde $\text{pos}()$ es una función que devuelve la posición de $i_x \in gsim_t$, empezando por 1. Es importante tener en cuenta que, en este caso, hay que ordenar $gsim_t$ por valor de similitud en orden decreciente.

Para ilustrar el funcionamiento de los sistemas de votación, mostramos un ejemplo (véase la Tabla 4.1) basado en un grafo de nodos usuarios. Por un lado, tenemos U_t , conjunto que guarda los cuatro usuarios similares a nuestro usuario objetivo u_t . Por otro, tenemos que $gsim_t = \{i_1, i_2, i_3, i_4, i_5\}$ contiene los elementos con los los usuarios en U_t han interactuado pero con los que u_t aún no. Los productos en $gsim_t$ están repetidos y en cada aparición hay un valor de similitud asociado distinto. Como podemos observar, cada sistema de votación genera diferentes listas de recomendación. Por ejemplo, si elegimos $k = 2$, la lista de productos recomendados a u_t es $[i_2, i_1]$ para todos los sistemas de votación, pero si elegimos $k = 3$, obtenemos dos listas diferentes: $[i_2, i_1, i_3]$ para el sistema de votación simple y $[i_2, i_1, i_5]$ para

U_t	$sim_t(n_v)$	$\mathbb{S}_U[n_t, n_v]$
n_3	$\{i_1, i_2, i_5\}$	5
n_1	$\{i_1, i_2, i_4\}$	4
n_2	$\{i_2, i_3, i_5\}$	2
n_4	$\{i_1, i_2, i_3\}$	1

(a) U_t : usuarios similares a u_t . $sim_t(n_v)$: productos con los que el usuario u_v ha interactuado previamente, ordenados decrecientemente por el valor de similitud extraído de \mathbb{S}_U . $\mathbb{S}_U[n_t, n_v]$: similitud entre n_t y n_v calculada con una medida de similitud de predicción de enlaces.

	SV	WV	PV
ag_1	$1 + 1 + 1 = 3$	$5/12 + 4/12 + 1/12 = 0,83$	$1/1 + 1/2 + 1/4 = 1,75$
ag_2	$1 + 1 + 1 + 1 = 4$	$5/12 + 4/12 + 2/12 + 1/12 = 1$	$1/1 + 1/2 + 1/3 + 1/4 = 2,08$
ag_3	$1 + 1 = 2$	$2/12 + 1/12 = 0,25$	$1/3 + 1/4 = 0,58$
ag_4	1	$4/12 = 0,33$	$1/2 = 0,5$
ag_5	$1 + 1 = 2$	$5/12 + 2/12 = 0,58$	$1/1 + 1/3 = 1,33$
R'	$[i_2, i_1, i_3, i_5, i_4]$	$[i_2, i_1, i_5, i_4, i_3]$	$[i_2, i_1, i_5, i_3, i_4]$

(b) Resultado de aplicar diferentes métodos de agregación a los datos descritos en la tabla anterior.

Tabla 4.1: Ejemplo de aplicación de diferentes métodos de agregación a un grafo basado en usuarios. La Tabla (a) describe la información obtenida del grafo y los métodos de similitud, y la Tabla (b), el resultado de aplicar los diferentes métodos de agregación.

los sistemas de votación ponderada y posicional. De este modo, se puede observar que la elección del sistema de votación a utilizar en nuestro sistema de recomendación tiene un impacto relevante en el resultado.

Teniendo en cuenta las medidas de similitud y los métodos de agregación propuestos en la tesis, podríamos considerar al filtrado colaborativo adaptado al escenario de mínimo conocimiento como una de las configuraciones específicas de los métodos basados en grafos de interacción. El comportamiento del filtrado colaborativo se basa en encontrar los elementos más parecidos a aquellos que le han gustado al usuario, o en encontrar los elementos que les han gustado a los usuarios más similares al usuario objetivo, y devolver los que tienen la similitud más alta. Recordemos que el escenario de mínimo conocimiento era aquel en el que solo disponemos de *ratings* unarios para generar recomendaciones, es decir, interacciones sin valor asociado al *rating*. Podríamos redefinir el filtrado colaborativo para funcionar en el escenario de conocimiento mínimo usando como medida de similitud entre dos elementos: (1) el número de usuarios que han interactuado con dos productos (en el modelo basado en productos) o; (2) el número de productos con los que dos usuarios han interactuado (en el modelo basado en usuarios). Esta

medida de similitud es equivalente a la medida EW definida para los modelos basados en grafos. Si posteriormente recomendamos los elementos con mayor similitud, significa que la adaptación de filtrado colaborativo al escenario de mínimo conocimiento es equivalente a los modelos basados en grafos de interacción configurados con la medida de similitud EW y el método de agregación basado en la máxima similitud.

4.2. Evaluación

El objetivo de la evaluación llevada a cabo es comprobar nuestra Hipótesis 1: que los grafos de interacción pueden generar recomendaciones explicables usando una mínima cantidad de conocimiento. Estas recomendaciones deben ser aceptables o válidas para los usuarios objetivo. Para ello, queremos evaluar si la precisión de los grafos es comparable a la de las técnicas clásicas de recomendación en escenarios de mínimo conocimiento, lo que aportaría una propuesta de solución en este tipo de casos. Cuando no se dispone de información adicional sobre características de productos y perfiles de usuario o sobre el valor asociado a la interacción, por ejemplo *ratings*, esperamos que nuestra propuesta proporcione recomendaciones de igual o incluso mejor calidad que las técnicas de recomendación tradicionales. Para poder hacer esta comparación es necesario que durante la evaluación sólo se use el conocimiento sobre si se han producido o no interacciones entre usuarios y productos.

Como hemos visto en el Capítulo 2, según la literatura existen varios métodos de recomendación que se pueden usar cuando no hay descripciones o valores asociados a las interacciones (Aggarwal, 2016; Lü et al., 2012; Miranda y Jorge, 2008; Volkovs y Yu, 2015). De entre ellos, hemos seleccionado algunos de los más representativos para evaluar nuestra propuesta. Una técnica que hemos seleccionado es el filtrado colaborativo basado en vecinos (en el capítulo nos referiremos a esta técnica como filtrado colaborativo), ya que es una de las técnicas más populares a la hora de hacer recomendaciones, también en las situaciones en las que no hay conocimiento, en las cuáles estas técnicas son adaptadas. Para llevar a cabo la evaluación, hemos elegido tanto el modelo basado en usuarios, como el basado en productos. Además, hemos seleccionado técnicas de filtrado colaborativo basado en modelos: técnicas de aprendizaje automático que se pueden utilizar en el caso de tener sólo información sobre la existencia de las interacciones (Ricci et al., 2015; Aioli, 2013): árboles de decisión, naive Bayes, redes neuronales, random forest y SVM. En el capítulo nos referiremos a ellas como técnicas de aprendizaje automático. El filtrado colaborativo basado en factorización de matrices no se ha contemplado en esta comparación por ser un algoritmo totalmente enfocado en los valores de los *ratings*.

Para comparar el filtrado colaborativo basado en vecinos y las técnicas

de aprendizaje automático con nuestros métodos propuestos, reduciremos la cantidad de conocimiento a utilizar, para que las técnicas clásicas puedan ser evaluadas en el escenario de mínimo conocimiento disponible en el que se puedan utilizar. Este escenario será lo más parecido posible al escenario de mínimo conocimiento en el que se pueden utilizar los métodos basados en grafos. Sin embargo, nunca van a ser iguales, ya que las técnicas clásicas requieren mayor cantidad de conocimiento mínimo que los modelos basados en grafos de interacción. En esta evaluación también tratamos de extraer algunas conclusiones con respecto a la explicabilidad de dichos métodos.

La implementación de los modelos basados en grafos, los resultados obtenidos de la evaluación y el código de análisis del conjunto de datos se encuentran en un repositorio GitHub¹ para poder consultarse.

A continuación, después de presentar el conjunto de datos a utilizar (Sección 4.2.1), detallamos el método de evaluación llevado a cabo (Sección 4.2.2), así como los escenarios de mínimo conocimiento tenidos en cuenta en la evaluación (Sección 4.2.3). Al final, describimos las métricas de evaluación usadas para medir la precisión de los modelos de recomendación (Sección 4.2.4).

4.2.1. Análisis del conjunto de datos

El conjunto de datos elegido para la evaluación es el conjunto de datos de MovieLens de 100.000 interacciones², ya que es ampliamente utilizado en la literatura. También lo elegimos con el objetivo de comprobar el comportamiento de los métodos evaluados en conjuntos de datos grandes. Anteriormente habíamos aplicado los métodos en grafos en jueces en línea, siendo el conjunto de datos mucho menor. Este dataset de MovieLens incluye *ratings* en forma de tupla $R = (t, u, i, x)$, donde u es el usuario que interviene en la interacción, i es la película que u ha visto, x es la valoración que u ha proporcionado a i y t es la marca de tiempo en la que u valoró i .

Mientras realizábamos la evaluación, nos dimos cuenta de que el conjunto de datos, a pesar de contener gran cantidad de interacciones, contiene escaso conocimiento para obtener resultados significativos de cualquier método de recomendación. Podemos ver la baja densidad que posee el conjunto de datos original de MovieLens en la Tabla 4.2. Además, resultaba difícil encontrar una cantidad adecuada de usuarios que hubiesen visto suficientes películas tanto en el conjunto de entrenamiento como en el de evaluación. Como consecuencia, decidimos analizar el conjunto de datos para saber cómo conseguir un subconjunto con el que pudiéramos trabajar adecuadamente: aumentando la densidad sin perder conocimiento. En la Figura 4.4, mostramos el número de usuarios (eje Y) que han interactuado con una determinada cantidad de las películas más populares (eje X) del conjunto de datos. Definimos la pelí-

¹<https://github.com/martcaro/Graph-Recommendations>

²<https://grouplens.org/datasets/movielens/100k/>

Métrica	ML_100K	ML_50	ML_25	ML_12
# Ratings	100.000	15.954	34.030	42.482
# Productos	1.682	200	200	200
# Usuarios	943	129	376	611
Densidad	0,063	0,618	0,453	0,348
Productos				
Máximo # ratings por producto	583	126	348	499
Mediana # ratings por producto	27	80	162,5	197,5
Media # ratings por producto	59,453	79,77	170,15	212,41
Mínimo # ratings por producto	1	22	54	84
Usuarios				
Máximo # ratings por usuario	737	173	173	173
Mediana # ratings por usuario	65	121	86	63
Media # ratings por usuario	106,045	123,674	90,505	69,529
Mínimo # ratings por usuario	20	100	50	25
Ratings				
% Ratings ≥ 4	55,375	62,774	64,331	64,672
% Ratings ≤ 4	44,625	37,226	35,669	35,328

Tabla 4.2: Análisis descriptivo del conjunto de datos utilizado para la evaluación de los modelos de recomendación. ML_100K es el conjunto de datos original de MovieLens con la información de 100K interacciones. ML_50, ML_25 y ML_12 son los conjuntos de datos extraídos de ML_100K que sólo contienen las interacciones en las que intervienen las 200 películas más populares y los usuarios que han interactuado con al menos el 50 %, el 25 % y el 12,5 %, respectivamente para cada conjunto, de esos productos.

cula más popular como la película que más ha sido valorada por los usuarios del conjunto de datos. Además, estudiamos los usuarios más activos, que son los que han valorado una mayor cantidad de películas. Para cada línea del gráfico, mostramos el número de usuarios que han interactuado con: el 100 % de las películas más populares (línea azul), al menos el 75 % de las películas más populares (línea roja), al menos el 50 % de las películas más populares (línea verde), al menos el 25 % de las películas más populares (línea morada) y al menos el 12,5 % de las películas más populares (línea naranja). De esta forma, en la línea azul de la Figura 4.4, se puede ver el número de usuarios que han visto, por ejemplo, las 200, 600 ó 1.000 películas más populares. Si observamos la línea roja, podemos ver el número de usuarios que han interactuado con al menos el 75 % de las 200, 600 ó 1.000 películas más populares. Lo mismo ocurre con el resto de líneas y sus respectivos porcentajes.

Teniendo en cuenta los resultados mostrados en la gráfica, decidimos estudiar las densidades del subconjunto de los 200 productos más populares, ya que es relativamente pequeño, pero con conocimiento suficiente. Encon-

tramos que las densidades son de 0,80 (conjunto del 75 %), 0,61 (50 %), 0,45 (25 %) y 0,34 (12,5 %), lo que nos ofrece un rango de datos suficientemente heterogéneo (marcamos estos valores en la línea negra de la Figura 4.4). El dataset del 75 % de las películas posee una densidad alta pero que carece de suficientes usuarios para realizar una evaluación significativa. Sin embargo, decidimos llevar a cabo la evaluación con el conjunto de las 200 películas más populares y con los usuarios que hubiesen interactuado con al menos el 50 %, el 25 % y el 12,5 % de esas películas, ya que con estos porcentajes, los datos permiten aplicar los algoritmos a estudiar en la evaluación.

Tras el preprocesamiento de datos, realizamos un análisis exploratorio para obtener sus descriptores estadísticos relevantes y confirmar su validez. En este caso, seguimos el modelo propuesto en el trabajo de Dooms et al. (2016) para hacer una comparación de los conjuntos de datos que intervienen en la evaluación, tal y como hicimos en nuestros trabajos anteriores (Caro-Martinez y Jimenez-Diaz, 2017; Caro-Martinez et al., 2019b). El resultado de este análisis descriptivo se muestra en la Tabla 4.2. Comparando las densidades, el número de *ratings*, de usuarios y de productos de los subconjuntos preprocesados con los del dataset original, podemos observar que los conjuntos de datos utilizados en la evaluación tienen suficiente conocimiento para obtener conclusiones significativas. Así pues, hemos realizado el mismo experimento con diferentes conjuntos de datos, aplicando la misma configuración y llevando a cabo el mismo proceso.

4.2.2. Método de evaluación

Para saber si las recomendaciones proporcionadas por nuestros métodos eran interesantes para los usuarios objetivo, necesitábamos construir un modelo para cada usuario u_t . En primer lugar, necesitábamos dividir el conjunto para obtener los conjuntos de entrenamiento y de evaluación adecuados con los que poder medir la precisión de los métodos de recomendación. Decidimos obtenerlos de igual forma para cada uno de los tres conjuntos de datos: a través de *cross-validation*, escogiendo de forma aleatoria el 10 % de los productos (un total de 20 películas de las 200 más populares) para crear el conjunto de evaluación y el 90 % restante para construir el de entrenamiento. Por tanto, en el conjunto de evaluación se consideran todas aquellas interacciones llevadas a cabo por el usuario objetivo u_t con esas 20 películas escogidas aleatoriamente. A su vez, las interacciones llevadas a cabo por u_t con estas películas son eliminadas del conjunto de entrenamiento. En el proceso de experimentación, los métodos de recomendación sólo van a poder recomendar productos de este conjunto de evaluación. De esta forma, vamos a medir si los métodos de recomendación predicen correctamente el interés de u_t por estos productos.

En la siguiente serie de experimentos, comparamos la precisión de los métodos basados en grafos con otras técnicas de recomendación, en concreto

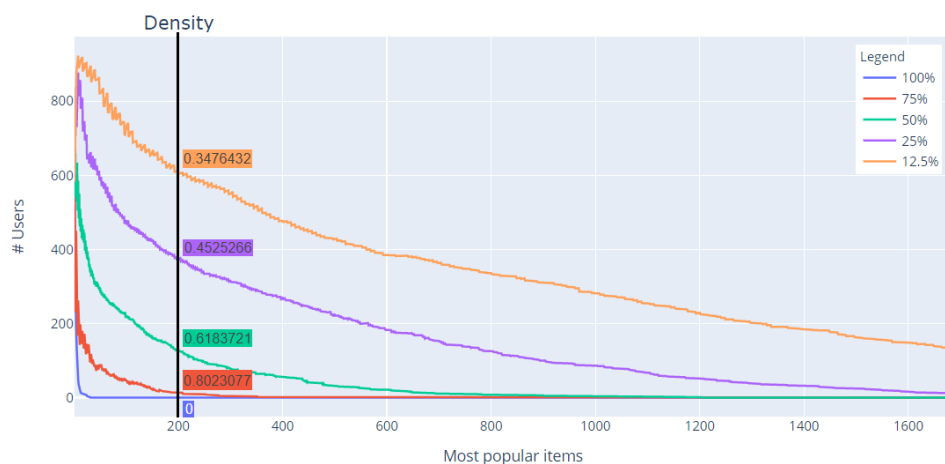


Figura 4.4: Estudio del conjunto de datos de MovieLens para aumentar su densidad. El gráfico muestra el número de usuarios que han interactuado con todas las películas más populares (línea azul), al menos el 75 % de esas películas (línea verde), al menos el 50 % (línea morada) y al menos el 12,5 % (línea naranja). También estudiamos la densidad (en la línea vertical negra) del dataset de las 200 películas más populares, que es el utilizado en la evaluación.

con algoritmos de filtrado colaborativo basado en usuarios y productos, y técnicas de aprendizaje automático. El filtrado colaborativo lo implementamos con el coeficiente de Pearson como medida de similitud, ya que es una de las medidas más utilizadas en la implementación de este tipo de algoritmos (Liu et al., 2014). Sin embargo, explota los valores de los *ratings* para obtener las similitudes entre los elementos, por lo que queríamos comprobar cómo se comporta con valores de *ratings* implícitos en escenarios de mínimo conocimiento.

4.2.3. Escenario de mínimo conocimiento

Como se ha explicado anteriormente, debemos adaptar el conjunto de datos para que estos modelos proporcionen recomendaciones en un escenario de conocimiento mínimo, transformando las valoraciones originales para que representen únicamente las interacciones entre el producto y el usuario. Este proceso se ilustra en la Tabla 4.3. Debemos tener en cuenta que los recomendadores de filtrado colaborativo no pueden trabajar con valores desconocidos, teniendo que utilizar, al menos, la representación *ternaria*. Necesitan representar tanto las interacciones positivas (cuyos *ratings* valen 4 ó 5), como las negativas (*ratings* con valor menor que 4) y las no-interacciones. Sin embargo, los modelos de aprendizaje automático pueden funcionar como mínimo con la representación *binaria* en la que las valoraciones positivas

<i>Rating</i> original	Ternario	Binario	Unario
5-4	1	1	1
3-1	0	0	-
Sin <i>rating</i>	-1	-	-
Modelo	CF	ML	Grafos

Tabla 4.3: Conocimiento mínimo necesario para crear los modelos de recomendación. El filtrado colaborativo (CF) y el aprendizaje automático (ML) requieren representaciones ternarias o binarias, respectivamente, mientras que los enfoques basados en grafos pueden aplicarse al escenario unario.

se codifican como 1, las negativas son 0, y los valores desconocidos no se incluyen en el modelo. Según Aggarwal (2016), existe una representación alternativa que codifica los *ratings* negativos y desconocidos con el mismo valor (normalmente 0). Sin embargo, introduce un sesgo importante. Por último, las representaciones *unarias* sólo representan los *ratings* positivos con 1. Si el usuario no ha interactuado o no le ha gustado un producto, el valor correspondiente no se especifica.

Los métodos basados en grafos son los únicos adecuados para el escenario unario, lo que reduce considerablemente el conocimiento requerido por el recomendador. Esto significa que, de los modelos estudiados en esta evaluación, los basados en grafos son los únicos que se pueden implementar en los escenarios donde sólo existan interacciones positivas, como es el caso de los jueces en línea. Así, la comparación debe realizarse teniendo en cuenta que no es posible evaluar estas técnicas con el mismo nivel de conocimiento. Los métodos basados en grafos se evalúan con el escenario de conocimiento más mínimo (representación unaria), pero la precisión de los modelos de aprendizaje automático y filtrado colaborativo se ve potenciada por la representación explícita del conocimiento negativo y desconocido.

A continuación, presentamos las métricas de evaluación utilizadas para comparar la precisión.

4.2.4. Métricas de evaluación

Cada método de recomendación va a obtener una lista R ordenada con k productos a recomendar a un determinado usuario objetivo u_t . Para medir la precisión de los métodos, aplicamos métricas comúnmente encontradas en la literatura (Herlocker et al., 2004; Pazzani, 1999; Said y Bellogín, 2014; Powers, 2011; McFee y Lanckriet, 2010), que evalúan las listas de recomendación obtenidas. Para calcular la precisión, estas métricas utilizan la información de los *true positives* o verdaderos positivos, a los que denotamos como TP . TP son los elementos relevantes para el usuario objetivo

u_t dentro de la lista de recomendación R obtenida por el recomendador, es decir, aquellos elementos, de entre los que le han gustado a u_t en el conjunto de evaluación, recomendados por el método recomendador. A continuación detallamos cuáles son las métricas de evaluación utilizadas en este trabajo:

- *One hit*. Mide si existe al menos un elemento relevante para el usuario objetivo dentro de la lista de recomendación.

$$1H@k = \begin{cases} 1, & \text{si } |TP| \geq 1 \\ 0, & \text{en otro caso} \end{cases} \quad (4.19)$$

donde $|TP|$ es el número de elementos relevantes calculados por el método de recomendación.

- *Precision@k*. Esta métrica muestra la proporción de productos que son relevantes para el usuario objetivo dentro de la lista de productos recomendados. El parámetro k se refiere al tamaño de la lista de recomendaciones. La precisión se calcula como:

$$P@k = \frac{|TP|}{k} \quad (4.20)$$

donde $|TP|$ es el número de elementos relevantes recuperados y k es el tamaño de la lista de recomendación.

- *R-precision (precision@R)*. Señala la proporción de elementos relevantes para el usuario objetivo entre los elementos más destacados del conjunto de evaluación. Establecimos este conjunto de elementos más destacados en el conjunto de evaluación como el conjunto de elementos accesibles en el grafo. Como hemos mencionado, nuestros métodos basados en grafos son modelos locales. Por lo tanto, sólo podemos llegar a un conjunto específico de nodos en el grafo para generar las recomendaciones. Este conjunto específico de nodos es al que consideramos como conjunto accesible. En el grafo basado en productos, el conjunto accesible es el conjunto de nodos enlazados a los nodos productos con los que el usuario objetivo ha interactuado. Sin embargo, en el grafo basado en usuarios, el conjunto accesible es el conjunto de productos con los que han interactuado usuarios cuyos nodo están enlazados al nodo del usuario objetivo. En este caso, la precisión se calcula como:

$$P@R = \frac{|TP|}{R} \quad (4.21)$$

donde $|TP|$ es el número de elementos relevantes recuperados y R es el tamaño del conjunto de elementos accesibles en el conjunto de evaluación. Es importante señalar que esta métrica no depende de k , sino

de la longitud del conjunto accesible, por lo que el valor de *R-precision* será fijo para cada modelo de recomendación aunque modifiquemos k .

Otra métrica de evaluación muy utilizada en la literatura es *recall@k* (Herlocker et al., 2004; Pazzani, 1999; Said y Bellogín, 2014; Powers, 2011; McFee y Lanckriet, 2010). Esta métrica en concreto no ha sido empleada en este trabajo ya que no resulta de utilidad. *Recall@k* calcula el porcentaje del total de elementos relevantes (*true positives*) que han sido recomendados por el sistema recomendador:

$$R@k = \frac{|TP|}{E} \quad (4.22)$$

donde E es el número de productos con los que el usuario ha interactuado en el conjunto de evaluación.

Nuestros métodos basados en grafos son locales, lo que significa que no podemos calcular todas las posibles soluciones, y, por tanto, tampoco podemos llegar a obtener como recomendación la totalidad de verdaderos positivos. De ahí que hayamos decidido usar como alternativa la métrica *R-precision*, ya que es equivalente a *recall@k* teniendo en cuenta el conjunto de elementos accesibles, en lugar de los elementos del conjunto de evaluación.

4.3. Resultados

Una vez que hemos visto cuál es la metodología que hemos llevado a cabo para realizar la evaluación que demuestra la Hipótesis 1, en esta sección analizamos los resultados obtenidos. En la Sección 4.3.1 detallamos la precisión conseguida con los métodos basados en grafos, mientras que en la Sección 4.3.2 comparamos las precisiones de los métodos basados en grafos con las obtenidas con las técnicas clásicas. Por último, en la Sección 4.3.3 discutimos la explicabilidad de los modelos evaluados. En el repositorio GitHub indicado antes se puede acceder a todos los resultados conseguidos en esta evaluación³.

4.3.1. Análisis de la precisión de los grafos de interacción

El primer paso de nuestra evaluación es aplicar nuestros métodos de recomendación basados en grafos al conjunto de datos de MovieLens y evaluar su precisión. El conjunto de datos se preprocesa de acuerdo con la Sección 4.2. Por lo tanto, vamos a llevar a cabo tres series de experimentos con los tres conjuntos de datos distintos que se han comentado anteriormente: los subconjuntos de datos que contienen las interacciones realizadas por los usuarios

³<https://github.com/martcaro/Graph-Recommendations/tree/main/results>

que han interactuado con el 12,5 %, el 25 % y el 50 % de las 200 películas más populares. En las tres series vamos a llevar a cabo el mismo proceso de experimentación que se indica a continuación.

Para cada usuario objetivo u_t , el conjunto de entrenamiento se utiliza para construir la matriz de adyacencia que hemos definido en la Sección 4.1. En esta matriz de adyacencia, $\mathbb{A}[u_t, i_j] = 1$ si el usuario u_t ha valorado la película i_j de forma positiva, $\mathbb{A}[u_t, i_j] = \emptyset$ en cualquier otro caso. A partir de la matriz \mathbb{A} , construimos los dos grafos descritos en la Sección 4.1.1 y 4.1.2: el grafo basado en productos G_I y el grafo basado en usuarios G_U , respectivamente. La densidad de los grafos se reduce filtrando sus enlaces utilizando un valor de umbral $\theta = 5$ (véase la Sección 4.1).

En este punto, podemos crear los modelos de recomendación basados en grafos detallados en este capítulo. Construimos todos los métodos posibles a partir de la combinación de las medidas de similitud descritas en la Sección 4.1.3 y de los métodos de agregación propuestos en la Sección 4.1.4. Después, los aplicamos sobre el conjunto de entrenamiento construido para u_t acorde a la configuración que hemos indicado en la Sección 4.2.2. Como resultado, para u_t obtenemos listas de películas recomendadas. Decidimos obtener listas de recomendación con distintos tamaños k y así evaluar también esta característica. En esta evaluación $k \in \{3, 5, 10\}$.

Por último, evaluamos cada modelo de recomendación usando el conjunto de evaluación construido para u_t , cuya creación hemos descrito también en la Sección 4.2.1, aplicando las métricas especificadas en la Sección 4.2.4. El resultado final para cada métrica de evaluación es una media de los resultados obtenidos para todos los usuarios.

A modo de ejemplo, ya que la cantidad de datos extraída en la evaluación es muy grande, algunos de los resultados obtenidos se encuentran en la Tabla 4.4: las tres primeras columnas corresponden al modelo basado en productos y las tres últimas, al basado en usuarios. Los resultados corresponden a los experimentos realizados con $k = 10$ y el subconjunto de datos con las interacciones realizadas por usuarios que han interactuado con al menos el 50 % de las 200 películas más populares. Aún así, se pueden consultar todos los resultados obtenidos en la evaluación en el repositorio de GitHub que hemos mencionado anteriormente. Analizando los resultados de la Tabla 4.4, podemos encontrar algunos patrones en el comportamiento de los métodos, que detallamos a continuación.

En general, todos los modelos obtienen excelentes resultados usando la métrica *one hit*. Siempre obtenemos un valor de 1, excepto en el modelo basado en productos si utilizamos el sistema de votación posicional, que alcanza un valor de 0,99. Esto significa que, obteniendo una lista de recomendaciones de tamaño 10, los métodos basados en grafos siempre son capaces de conseguir recomendar al menos una película interesante para el usuario objetivo. Este buen dato es esperable ya que hay un 99 % de probabilidades de acertar

al menos una recomendación de entre esas 10.

En lo que respecta a la *precision@k* y a la *R-precision*, nuestras propuestas obtienen resultados relevantes, observándose algunas diferencias entre el método basado en productos y el basado en usuarios. En el modelo basado en productos, nos encontramos que, con respecto a la *precision@k*, el sistema de votación ponderada y la máxima similitud son los métodos de agregación que mejor funcionan. Ambos alcanzan un valor de 0,66, mientras que la votación simple y la posicional alcanzan 0,56 y 0,55, respectivamente. La razón de estos resultados podría ser que los dos métodos de agregación que logran mejores resultados utilizan el valor de similitud directamente para generar el valor de similitud agregado, lo cual parece ser una herramienta eficaz en este caso. En el caso de la votación simple y posicional, sin embargo, se usa el valor de similitud de forma indirecta para generar el valor de similitud agregado. Los resultados de *precision@k* con el modelo basado en usuarios también rondan el valor de 0,66, aunque llega a alcanzar un valor de 0,68 usando el sistema de votación posicional y EW como medida de similitud. En este caso, no hay un patrón claro de cuál es el método de agregación que mejor funciona, ya que los resultados son muy similares entre sí.

A partir de los valores obtenidos con *R-precision*, se puede apreciar un comportamiento similar. En los métodos basados en productos, de nuevo tenemos que la máxima similitud y el sistema de votación ponderada son los métodos de agregación que consiguen una mejor precisión (0,66). Los sistemas de votación simple y posicional llegan a obtener 0,57 y 0,53, respectivamente. En este caso se puede ver que las técnicas de predicción de enlaces que usan pesos podrían conseguir una precisión algo menor. WAA, WCN y WPA alcanzan un valor de 0,65, mientras que el resto consiguen 0,66.

La diferencia más significativa entre los comportamientos del modelo basado en productos y usuarios está en los resultados de *R-precision*. Los valores del modelo basado en usuarios no son tan buenos como los logrados con el modelo basado en productos. En cuanto a las medidas de similitud y métodos de agregación, podemos establecer algunas conclusiones con respecto al modelo basado en grafos de usuarios. PA, con el método de agregación basado en la máxima similitud y CN, con la votación posicional consiguen los mejores resultados (0,44). Sin embargo, JN también obtiene puntuaciones similares (0,43 con la votación posicional y 0,41 con la máxima similitud). Este comportamiento podría estar relacionado con que PA considera la importancia de los nodos en los extremos de los enlaces. CN y JN tienen en cuenta las relaciones en las vecindades. En cambio, el resto de las medidas de similitud utilizan los pesos de los enlaces. WAA, WCN y WPA, alcanzan el valor de 0,42, todas ellas utilizando la votación posicional. Por tanto, aquí podemos ver el mismo patrón que en los resultados de *R-precision* obtenidos con el grafo de productos: las medidas que usan pesos consiguen una precisión algo menor que las otras medidas de similitud. También podríamos concluir que

<i>LinkPred()</i>	<i>Aggr()</i>	Producto			Usuario		
		1H@k	P@k	P@R	1H@k	P@k	P@R
AA	HS	1,00*+	0,66*	0,66*+	1,00*+	0,64	0,43
	SV	1,00*+	0,56	0,57	1,00*+	0,66	0,36
	WV	1,00*+	0,66*	0,66*+	1,00*+	0,66	0,36
	PV	0,99	0,55	0,53	1,00*+	0,65	0,40
CN	HS	1,00*+	0,66*	0,66*+	1,00*+	0,64	0,43
	SV	1,00*+	0,56	0,57	1,00*+	0,66	0,36
	WV	1,00*+	0,66*	0,66*+	1,00*+	0,66	0,36
	PV	0,99	0,55	0,53	1,00*+	0,65	0,44*
EW	HS	1,00*+	0,66*	0,65	1,00*+	0,66	0,22
	SV	1,00*+	0,56	0,57	1,00*+	0,66	0,36
	WV	1,00*+	0,66*	0,66*+	1,00*+	0,66	0,36
	PV	0,99	0,55	0,53	1,00*+	0,68*+	0,31
JN	HS	1,00*+	0,66*	0,66*+	1,00*+	0,63	0,41
	SV	1,00*+	0,56	0,57	1,00*+	0,66	0,36
	WV	1,00*+	0,66*	0,66*+	1,00*+	0,66	0,36
	PV	0,99	0,55	0,53	1,00*+	0,65	0,43
PA	HS	1,00*+	0,66*	0,66*+	1,00*+	0,64	0,44*
	SV	1,00*+	0,56	0,57	1,00*+	0,66	0,36
	WV	1,00*+	0,66*	0,66*+	1,00*+	0,66	0,36
	PV	0,99	0,55	0,53	1,00*+	0,66	0,39
WAA	HS	1,00*+	0,66*	0,65	1,00*+	0,62	0,40
	SV	1,00*+	0,56	0,57	1,00*+	0,66	0,36
	WV	1,00*+	0,66*	0,65	1,00*+	0,66	0,36
	PV	0,99	0,55	0,53	1,00*+	0,66	0,42
WCN	HS	1,00*+	0,66*	0,65	1,00*+	0,62	0,40
	SV	1,00*+	0,56	0,57	1,00*+	0,66	0,36
	WV	1,00*+	0,66*	0,65	1,00*+	0,66	0,36
	PV	0,99	0,55	0,53	1,00*+	0,66	0,42
WPA	HS	1,00*+	0,66*	0,65	1,00*+	0,62	0,40
	SV	1,00*+	0,56	0,57	1,00*+	0,66	0,36
	WV	1,00*+	0,66*	0,65	1,00*+	0,66	0,36
	PV	0,99	0,55	0,53	1,00*+	0,66	0,42

Tabla 4.4: Resultados obtenidos con los grafos basados en productos y usuarios cuando $k = 10$ y los usuarios han interactuado con al menos el 50% de las 200 películas más populares. Los mejores resultados entre los métodos de agregación aparecen en negrita. Los mejores valores obtenidos por cada métrica de evaluación, para cada modelo de recomendación, están marcados con *. Los mejores resultados obtenidos para cada métrica de evaluación, independientemente del modelo, se indican con +. *LinkPred()* denota la medida de similitud utilizada y *Aggr()* es el método de agregación usado. HS hace referencia al método de agregación basado en la máxima similitud, SV, a la votación simple, WV, a la votación ponderada y PV, a la votación posicional.

la votación posicional es el método de agregación que mejor funciona en la mayoría de los casos con las puntuaciones más altas, aunque no es el mejor método de agregación si utilizamos AA, EW y PA. Por lo tanto, la votación posicional ordena mejor la lista de recomendaciones, lo cual es comprensible debido a la naturaleza de su funcionamiento.

4.3.2. Comparación con las técnicas clásicas

En esta evaluación se comparan las precisiones de los métodos basados en grafos con otras técnicas clásicas de recomendación: filtrado colaborativo basado en productos y basado en usuarios (usando el coeficiente de Pearson), árboles de decisión, naive Bayes, redes neuronales, random forest y SVM. El conjunto de datos utilizado es el especificado para cada modelo en la Sección 4.2.3: con *ratings* ternarios para el filtrado colaborativo, y binarios para las técnicas de aprendizaje automático.

Como se ha comentado en la Sección 4.3.1, para hacer el cálculo de la *R-precision*, necesitamos tener un conjunto accesible de elementos, que va a ser distinto para el modelo basado en el grafo de productos y para el grafo de usuarios. Por este motivo hemos creado dos series de experimentos: para evaluar las técnicas clásicas con cada conjunto de elementos accesibles. Sin embargo, para las técnicas de aprendizaje automático no tenemos un método basado en productos y un método basado en usuarios. Por tanto, los resultados en *one hit* y *precision@k* serán los mismos para ambas series de experimentos ya que la elección del conjunto de elementos accesibles no afecta a estas métricas. En el filtrado colaborativo basado en vecinos sí veremos diferencias en los resultados de estas métricas, ya que tenemos un modelo basado en productos y otro basado en usuarios.

Los conjuntos de entrenamiento y evaluación se han construido para cada usuario objetivo u_t como ya hemos descrito en la Sección 4.2.2. Las técnicas clásicas se aplican sobre el conjunto de entrenamiento para obtener listas de recomendación para ese usuario objetivo u_t . De nuevo, el tamaño de la lista de recomendaciones es $k \in \{3, 5, 10\}$. Utilizamos el conjunto de evaluación para comparar las recomendaciones proporcionadas por cada modelo con el conjunto de productos del conjunto de evaluación, de igual forma a como se han creado en la implementación de los métodos basados en grafos. Sin embargo, recordemos que las técnicas de filtrado colaborativo utilizan como conocimiento mínimo la representación ternaria y los árboles de decisión, naive Bayes, las redes neuronales, random forest y SVM, la representación binaria, mientras que los grafos de interacción usan la representación unaria.

Las Tablas 4.5 y 4.6 muestran los resultados de estas dos series de experimentos, los experimentos llevados a cabo con el modelo basado en productos y con el modelo basado en usuarios, haciendo una comparación con los resultados obtenidos con los modelos basados en grafos. La medida de similitud basada en predicción de enlaces utilizada es la que ofrece resultados con ma-

Mét.	Dat.	k	HS	SV	WV	PV	DT	NB	NN	RF	SVM	CF
1H@k	12,5 %	3	0,79	0,59	0,79	0,58	0,60	0,52	0,60	0,58	0,52	0,89
		5	0,88	0,76	0,87	0,73	0,64	0,54	0,63	0,61	0,54	0,94
		10	0,90	0,89	0,90	0,91	0,64	0,57	0,64	0,62	0,55	0,97
	25 %	3	0,97	0,89	0,97	0,77	0,88	0,76	0,88	0,88	0,88	0,98
		5	0,99	0,94	0,98	0,90	0,94	0,80	0,94	0,94	0,93	0,99
		10	0,99	0,99	1,00	0,99	0,95	0,82	0,95	0,95	0,93	1,00
	50 %	3	0,99	0,97	0,99	0,95	0,96	0,95	0,96	0,95	0,96	0,98
		5	1,00	1,00	1,00	0,98	0,98	0,98	0,98	0,97	0,96	0,99
		10	1,00	1,00	1,00	0,99	0,98	0,98	0,98	0,97	0,96	1,00
P@k	12,5 %	3	0,57	0,31	0,57	0,32	0,36	0,31	0,35	0,34	0,30	0,59
		5	0,48	0,32	0,52	0,32	0,29	0,25	0,28	0,27	0,24	0,54
		10	0,42	0,31	0,43	0,32	0,18	0,20	0,17	0,16	0,14	0,46
	25 %	3	0,76	0,60	0,77	0,42	0,58	0,48	0,58	0,58	0,56	0,76
		5	0,70	0,51	0,69	0,42	0,51	0,42	0,51	0,51	0,47	0,68
		10	0,59	0,44	0,59	0,41	0,33	0,33	0,33	0,33	0,30	0,59
	50 %	3	0,70	0,58	0,71	0,60	0,68	0,63	0,68	0,67	0,66	0,73
		5	0,70	0,56	0,69	0,58	0,61	0,59	0,61	0,61	0,59	0,70
		10	0,66	0,56	0,66	0,55	0,40	0,44	0,40	0,40	0,38	0,66
P@R	12,5 %	R	0,46	0,31	0,49	0,30	0,22	0,23	0,21	0,20	0,17	0,54
	25 %	R	0,62	0,48	0,62	0,42	0,34	0,33	0,34	0,34	0,31	0,62
	50 %	R	0,66	0,57	0,66	0,53	0,38	0,41	0,38	0,38	0,36	0,66

Tabla 4.5: Comparación entre los resultados obtenidos con los modelos basados en grafos usando las métricas de evaluación *one hit*, *precision@k* y *R-precision*. En la tabla se pueden ver los resultados obtenidos con los grafos de interacción de productos que utilizan el método de agregación basado en la máxima similitud (HS), la votación simple (SV), la votación ponderada (WV) y la votación posicional (PV), además de los resultados obtenidos con filtrado colaborativo basado en vecinos (CF), los árboles de decisión (DT), naive Bayes (NB), redes neuronales (NN), random forest (RF) y SVM. Los mejores resultados de cada fila están marcados en negrita. Se han marcado también aquellos resultados que se diferencian en un 0,01 con los mejores resultados al no ser esta diferencia estadísticamente significativa.

yor precisión. A partir de estos resultados, podemos concluir que los métodos basados en grafos son los métodos que consiguen una mayor precisión con menor cantidad de conocimiento (representación unaria). Sus resultados son bastante similares a los obtenidos con filtrado colaborativo, algoritmo que usa una representación ternaria, especialmente si observamos la Tabla 4.5, donde se muestran los resultados de los modelos basados en productos. Está claro que las técnicas de aprendizaje automático funcionan peor que el filtrado colaborativo y los modelos basados en grafos aún utilizando una representación binaria. No podemos observar una gran diferencia entre las técnicas evaluadas si observamos los resultados de *one hit*, ya que es la métrica más sencilla, pero esta diferencia es notable si observamos *precision@k* y *R-precision*. Entre las técnicas de aprendizaje automático tampoco hay ninguna que destaque sobre las demás. Por ejemplo, con el dataset del 25 %

Mét.	Dat.	k	HS	SV	WV	PV	DT	NB	NN	RF	SVM	CF	
1H@k	12,5 %	3	0,87	0,80	0,80	0,89	0,60	0,52	0,60	0,58	0,52	0,90	
		5	0,95	0,88	0,88	0,94	0,63	0,54	0,63	0,61	0,54	0,95	
		10	0,99	0,91	0,91	0,98	0,64	0,57	0,64	0,62	0,55	0,99	
	25 %	3	0,92	0,97	0,97	0,98	0,88	0,76	0,88	0,88	0,88	0,88	0,95
		5	0,99	0,98	0,98	0,99	0,94	0,80	0,94	0,94	0,93	0,99	
		10	1,00	0,99	1,00	1,00	0,95	0,82	0,95	0,95	0,93	1,00	
	50 %	3	0,97	0,99	1,00	0,98	0,96	0,95	0,96	0,95	0,96	0,96	0,95
		5	1,00	1,00	1,00	1,00	0,98	0,98	0,98	0,97	0,96	1,00	
		10	1,00	1,00	1,00	1,00	0,98	0,98	0,98	0,97	0,96	1,00	
P@k	12,5 %	3	0,54	0,58	0,58	0,63	0,35	0,31	0,35	0,34	0,30	0,57	
		5	0,52	0,49	0,52	0,56	0,29	0,25	0,28	0,27	0,24	0,55	
		10	0,46	0,42	0,43	0,46	0,18	0,20	0,17	0,16	0,14	0,47	
	25 %	3	0,61	0,76	0,77	0,78	0,58	0,48	0,58	0,58	0,56	0,65	
		5	0,61	0,69	0,69	0,71	0,51	0,42	0,51	0,51	0,47	0,64	
		10	0,56	0,59	0,59	0,60	0,33	0,33	0,33	0,33	0,30	0,60	
	50 %	3	0,68	0,68	0,74	0,77	0,68	0,63	0,68	0,67	0,66	0,70	
		5	0,67	0,68	0,69	0,72	0,61	0,59	0,61	0,61	0,59	0,68	
		10	0,66	0,66	0,66	0,68	0,40	0,44	0,40	0,40	0,38	0,66	
P@R	12,5 %	R	0,35	0,26	0,28	0,32	0,05	0,05	0,05	0,05	0,04	0,11	
	25 %	R	0,33	0,49	0,49	0,49	0,07	0,07	0,07	0,07	0,06	0,13	
	50 %	R	0,44	0,36	0,36	0,44	0,09	0,11	0,09	0,09	0,09	0,16	

Tabla 4.6: Comparación entre los resultados obtenidos con los modelos basados en usuarios usando las métricas de evaluación *one hit*, *precision@k* y *R-precision*. En la tabla se pueden ver los resultados obtenidos con los grafos de interacción de usuarios que utilizan el método de agregación basado en la máxima similitud (HS), la votación simple (SV), la votación ponderada (WV) y la votación posicional (PV), además de los resultados obtenidos con filtrado colaborativo basado en vecinos (CF), los árboles de decisión (DT), naive Bayes (NB), redes neuronales (NN), random forest (RF) y SVM. Los mejores resultados de cada fila están marcados en negrita. Se han marcado también aquellos resultados que se diferencian en un 0,01 con los mejores resultados al no ser esta diferencia estadísticamente significativa.

y $k = 5$, podemos observar que el mejor resultado obtenido con *precision@k* con los grafos de productos es 0,70, usando el método de agregación basado en la máxima similitud. Con esta misma métrica de evaluación, usando filtrado colaborativo se consigue 0,68, mientras que con técnicas de aprendizaje automático (en concreto, árboles de decisión, redes neuronales y random forest) conseguimos 0,51.

Otro ejemplo sería el comportamiento observado en los modelos basados en usuarios (véase la Tabla 4.6). Si observamos los resultados de *R-precision*, podemos ver que, con el dataset del 25 % de los usuarios, conseguimos un valor de 0,49 usando los modelos basados en grafos con los sistemas de votación. Sin embargo con filtrado colaborativo sólo se llega a 0,13, mientras que con las técnicas de aprendizaje automático, se alcanza únicamente 0,07 (con todas las técnicas excepto con SVM). Es especialmente destacable es-

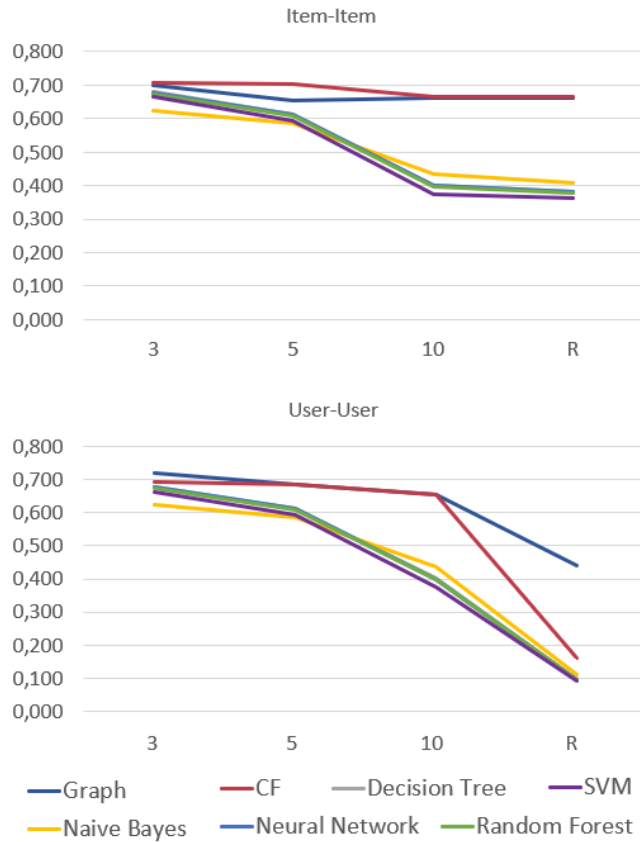


Figura 4.5: Análisis del crecimiento del tamaño k de la lista de recomendación en los modelos de recomendación evaluados. En la figura se muestran los valores obtenidos con la métrica $precision@k$ siendo $k \in \{3, 5, 10, R\}$ en los modelos basados en productos y en usuarios, utilizando el subconjunto del 50 %.

ta diferencia observada con R -precision en los modelos basados en usuarios, comportamiento que no veíamos en los modelos basados en productos, donde la diferencia de valores de R -precision no es tan marcada. Esto significa que, si basamos la recomendación en los usuarios similares, los grafos funcionan mejor a la hora de encontrar elementos relevantes.

Entre los resultados obtenidos con los subconjuntos de datos del 50 % y 25 % de los usuarios no hay grandes diferencias. Sin embargo, podemos observar un patrón diferente en los resultados con el conjunto de datos en el que los usuarios sólo interactuaron con el 12,5 % de las películas. En este caso, los valores obtenidos disminuyen, y podemos concluir que la precisión obtenida con este subconjunto es inferior a la precisión conseguida con conjuntos de datos más densos, independientemente del modelo de recomendación que estemos evaluando. Esto corrobora nuestra suposición inicial sobre el pro-

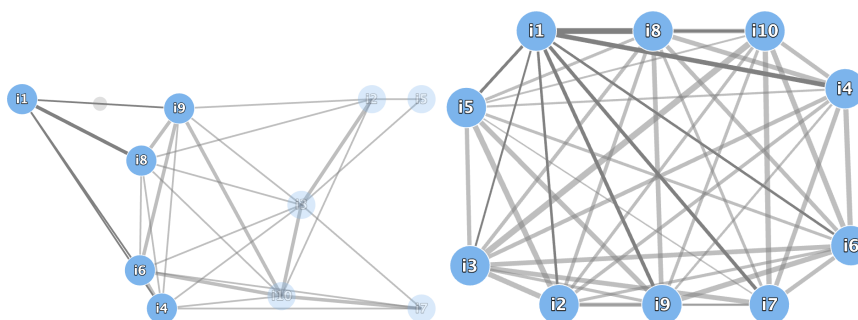


Figura 4.6: Explicación visual de una recomendación real obtenida con modelos basados en productos. Los nodos representan los productos: el producto 1 es el que se recomienda; los productos 4, 6, 8 y 9 son con los que ha interactuado el usuario objetivo. A la izquierda: recomendación basada en grafos (modelo local). A la derecha: filtrado colaborativo (modelo global).

blema de la dispersión del dataset al generar recomendaciones, como ya se comentó en la Sección 4.2.1. Por tanto, la dispersión del conjunto de datos afecta a la precisión del modelo recomendador, independientemente de cuál sea este modelo.

En la Figura 4.5, comparamos la precisión de todos los modelos con respecto al tamaño de la lista de elementos a recomendar (valor de k). Este gráfico se ha realizado utilizando únicamente el conjunto de datos cuyos usuarios han interactuado con al menos el 50% de las películas, ya que, aunque las precisiones varíen, el patrón de comportamiento de los modelos no cambia con el aumento o disminución de la densidad del dataset. Los recomendadores basados en grafos que se han representado en la figura son los que obtuvieron la mejor precisión según la Sección 4.3.1. También cabe destacar que sólo se ha representado la *precision@k*, ya que *one hit* es una métrica más sencilla donde, como ya hemos visto, no se ve tan claramente la diferencia entre las precisiones de los modelos. Podemos ver de nuevo que los grafos ofrecen mejores precisiones que las técnicas clásicas, independientemente del tamaño de la lista de recomendación, especialmente en los valores de *R-precision* de los modelos basados en usuarios.

4.3.3. Explicabilidad de los modelos

En términos de explicabilidad, podemos discutir algunas conclusiones. Sabiendo que los métodos globales son menos interpretables que los locales según la literatura, podemos asumir que los grafos de interacción son más explicables que el resto de los métodos de recomendación evaluados, usando además una menor cantidad de conocimiento. En la Figura 4.6 se puede ver la explicación gráfica de ambos modelos con un ejemplo real. Por un lado, el filtrado colaborativo (a la derecha en la figura) es un método global que reali-

za predicciones utilizando la similitud de cualquier elemento del sistema. Por lo tanto, en la representación gráfica, todos los elementos están conectados con todos. Sin embargo, los métodos basados en grafos (a la izquierda en la figura) son locales. Las medidas de similitud basadas en predicción de enlaces que usan nuestros métodos sólo utilizan la información de los vecindarios, por lo que sólo se representa la similitud entre unos determinados pares de nodos. Esto puede resultar más comprensible, ya que se puede intuir que las conexiones se refieren a la probabilidad de recomendación o similitud.

4.4. Discusión de resultados

Finalmente, podemos extraer algunas conclusiones globales del análisis realizado:

- Dependiendo de la medida de similitud y del método de agregación utilizado en los métodos basados en grafos, podemos obtener diferentes resultados. La elección de los mismos es significativa.
- Los recomendadores basados en grafos de productos funcionan mejor si utilizamos la votación ponderada y Edge Weight como medida de similitud. En el caso del grafo de usuarios, su configuración más eficaz no está tan clara y depende de la densidad de información en el conjunto de datos.
- En general, nuestros métodos basados en grafos y el filtrado colaborativo obtienen mejores resultados que las técnicas de aprendizaje automático en escenarios de mínimo conocimiento. Aunque en la literatura las técnicas de aprendizaje automático se utilizan para hacer recomendaciones cuando no existe conocimiento explícito, no son herramientas específicas para abordar el problema de hacer recomendaciones. El filtrado colaborativo y los métodos basados en grafos sí son métodos diseñados para resolver este problema. Esta podría ser la razón de estos resultados.
- La similitud en la precisión obtenida entre los grafos de interacción y el filtrado colaborativo radica en que podemos considerar el filtrado colaborativo como una configuración específica de los grafos de interacción en escenarios de mínimo conocimiento. Sin embargo, este escenario no es el mismo para ambos métodos: los grafos utilizan una representación unaria, mientras que el filtrado colaborativo emplea la representación ternaria. Por lo tanto, los grafos requieren menor cantidad de conocimiento.
- De acuerdo con la literatura, los sistemas de recomendación basados en grafos son considerados más explicables que los métodos tradicionales,

ya que éstos son métodos globales mientras que los modelos basados en grafos son métodos locales.

- Si comparamos los modelos basados en productos y en usuarios (independientemente de si son grafos o técnicas clásicas) los modelos basados en productos son mejores que los basados en usuarios. Tiene sentido porque con los modelos basados en productos extraemos las recomendaciones directamente a partir de los elementos con los que el usuario objetivo ha interactuado. Con el modelo basado en usuarios, tenemos que obtener primero el conjunto de usuarios similares al usuario objetivo y obtener los productos a recomendar después. No obstante, si queremos utilizar un modelo basado en usuarios, la mejor opción es utilizar los métodos basados en grafos, por encima de las técnicas clásicas, ya que proporcionan una mejor precisión que éstas y además son más interpretables.
- La dispersión en el conjunto de datos usado afecta a la precisión del sistema recomendador, independientemente del método de recomendación que usemos. Cuando más denso sea el conjunto de datos, mejores recomendaciones se van a conseguir. Esto, como ya hemos visto, es uno de los problemas a resolver en sistemas de recomendación. En esta evaluación, se ha tratado la dispersión con el objetivo de poder extraer resultados y así evaluar técnicas tan distintas.

Por tanto, con nuestra propuesta podemos generar recomendaciones en situaciones en las que es más difícil aplicar técnicas clásicas. Requieren menor cantidad de conocimiento y a la vez son capaces de obtener, al menos, los mismos resultados que las técnicas tradicionales. Además tienen la ventaja de que son interpretables, por lo que los usuarios pueden entender mejor el sistema y las recomendaciones recibidas.

4.5. Conclusiones

Los sistemas de recomendación son una de las herramientas más ventajosas hoy en día en economía digital porque permiten acceder a una mejor selección de los productos o las experiencias que queremos consumir. Tradicionalmente los recomendadores se basan en información explícita, siendo los *ratings* y las descripciones de los productos y los usuarios las fuentes de conocimiento más usadas para generar recomendaciones. Como hemos visto, este conocimiento no siempre está disponible. En la literatura, encontramos diversas técnicas clásicas que se pueden utilizar para proporcionar recomendaciones pero son métodos que no están adaptados a este tipo de escenario de conocimiento mínimo.

En esta tesis doctoral se propone un modelo para generar recomendaciones en escenarios de mínimo conocimiento utilizando la información sobre

las interacciones que realizan los usuarios sobre los productos del sistema, información a la que siempre se va a poder acceder. Esta fuente de conocimiento va a ser representada en grafos, los cuáles van a ser utilizados como base para aplicar técnicas de predicción de enlaces. En esta propuesta, las técnicas de predicción de enlaces se usan para encontrar similitudes entre las entidades del grafo y así generar recomendaciones interesantes para un usuario objetivo. Hemos propuesto también varios métodos de agregación, necesarios para poder implementar correctamente los modelos propuestos y eliminar repeticiones en las listas de recomendación.

La principal característica de nuestros métodos es que no necesitan información adicional para poder generar recomendaciones, únicamente el conocimiento sobre las interacciones llevadas a cabo. Además de esto, nuestro sistema parece ser más explicable que las técnicas clásicas estudiadas en la literatura. Para empezar, nuestro métodos utilizan los grafos como estructura base para generar las recomendaciones. El uso de este tipo de estructuras hace que el funcionamiento del sistema sea más interpretable para los usuarios. Igualmente, que nuestros métodos basados en grafos sean modelos locales aporta interpretabilidad al sistema.

Para comprobar la eficacia de nuestra propuesta, hemos llevado a cabo una evaluación en la que hacemos una comparación entre las precisiones de los métodos basados en grafos de interacción y de las técnicas clásicas que se utilizan en escenarios donde no hay *ratings* o descripciones disponibles. Las técnicas clásicas escogidas comprenden el filtrado colaborativo basado en vecinos y algunas técnicas de aprendizaje automático. Una particularidad importante de esta evaluación es que hemos necesitado transformar el conocimiento del conjunto de datos utilizado para simular un escenario de mínimo conocimiento, sustituyendo la información explícita por implícita. De esta forma, evaluamos los métodos en el contexto del problema que queremos resolver. Sin embargo cada uno de los métodos requiere una mínima cantidad de conocimiento distinta. El filtrado colaborativo necesita representación ternaria y las técnicas de aprendizaje automático, binaria. Los grafos de interacción son los únicos métodos que se pueden aplicar con una representación unaria (cuando sólo existen interacciones positivas), por lo que son las técnicas que menor cantidad de conocimiento necesitan. Por tanto, los modelos basados en grafos pueden aplicarse a cualquier tipo de escenarios, mientras que el filtrado colaborativo y las técnicas de aprendizaje automático no pueden.

Teniendo en cuenta los resultados obtenidos, podemos deducir que nuestra propuesta proporciona resultados similares a los de las técnicas clásicas, incluso mejores en algunos casos. Podemos utilizar nuestra propuesta sin perder precisión cuando no tenemos suficiente información para utilizar técnicas clásicas con la ventaja de que son más interpretables. Por tanto, concluimos que hemos cumplido el Objetivo 1, desarrollando métodos de recomendación

basados en grafos y validando la Hipótesis 1 planteada.

Un primer resultado del trabajo descrito en este capítulo ha sido publicado en el congreso *International Conference on Case-Based Reasoning* del 2017 (Caro-Martinez y Jimenez-Diaz, 2017). La metodología completa y la evaluación que se ha descrito en este capítulo se encuentra pendiente de publicar en el momento de escritura de esta memoria (Caro-Martinez et al., Pendiente de aceptar.).

A continuación, el siguiente capítulo describe cómo hemos abordado el Objetivo 2 planteado en esta tesis en el que proponíamos demostrar que los métodos basados en grafos de interacción sirven como sistemas de explicación en escenarios de mínimo conocimiento.

Capítulo 5

Sistemas de explicación para recomendadores basados en grafos de interacción

Como hemos comentado a lo largo de esta tesis doctoral, a pesar del claro beneficio que supone usar sistemas de recomendación, pueden no ser tan eficaces como cabría esperar. Esto sucede porque los usuarios no entienden cómo funcionan o las razones detrás de la recomendación, por lo que desconfían de ellos y no les prestan atención. Esta cuestión es crítica en algunos contextos y campos como en la medicina (Darlington, 2011). Sin embargo, proporcionar explicaciones en las recomendaciones de actividades cotidianas, como el comercio electrónico o el entretenimiento, también es importante porque aumenta la credibilidad del sistema y la fidelidad del usuario (Walek y Fojtik, 2020). Como solución a este problema surgen los sistemas de explicación en los recomendadores (Kunkel et al., 2019). Los sistemas de explicaciones tratan de aclarar por qué se ha proporcionado una recomendación a un usuario objetivo (Sharma y Ray, 2016).

Hemos visto que los sistemas de recomendación tradicionales están basados en filtrado colaborativo o en contenido si hacen uso de los *ratings* o las descripciones de los productos y los usuarios, respectivamente, para recuperar los productos que pueden tener más interés para el usuario objetivo (Ekstrand et al., 2011; Lops et al., 2011). En el capítulo anterior, veíamos que este conocimiento no siempre está disponible, por lo que propusimos métodos de recomendación basados en grafos de interacción y técnicas de predicción de enlaces para resolver el problema. Esta situación también se puede dar cuando se necesita generar explicaciones para recomendaciones, bien porque no podemos acceder a este conocimiento o bien porque no existe. Por ejemplo, nos encontramos con este problema cuando queremos dar explicaciones a un usuario de un sistema de recomendación de caja negra o *black-box* (Sinha y Swearingen, 2002; Friedrich y Zanker, 2011). Los reco-

mendadores de caja negra son aquellos que no son transparentes, por lo que no podemos conocer cómo funcionan y cuál es el conocimiento del que hacen uso para obtener las recomendaciones. En capítulos anteriores, destacábamos que los grafos son estructuras que se usan para resolver tareas complejas y que además aportan interpretabilidad al sistema (Ji et al., 2021; Guo et al., 2020). También hacíamos hincapié en su naturaleza de métodos locales, lo que permite contribuir a aumentar la explicabilidad del sistema (Ribeiro et al., 2016; Singh y Anand, 2019; Arya et al., 2019; Ribeiro et al., 2018). Como consecuencia, en el Objetivo 2 de la tesis proponemos *diseñar, desarrollar y evaluar sistemas de explicación basados en grafos de interacción en escenarios de mínimo conocimiento e independientes del tipo de recomendador*, mostrando así su versatilidad y aplicabilidad. Nuestra Hipótesis 2 es que *las técnicas de análisis de grafos permiten generar explicaciones para sistemas de recomendación* de forma que pueden lograr explicaciones más eficaces que otras técnicas clásicas, además *pudiendo ser aplicadas en contextos en los que hay conocimiento mínimo*.

En la Sección 5.1 especificamos cómo se construyen los grafos de interacción para diseñar nuestros métodos de explicación, teniendo en cuenta que van a tener semejanzas con la propuesta del Capítulo 4. También incluimos la descripción del escenario de mínimo conocimiento para el que se usa nuestro método propuesto (Sección 5.1.1), además de la definición de nuestro método con el modelo conceptual propuesto en el Capítulo 3 (Sección 5.1.2). De nuevo vamos a generar dos modelos distintos, uno basado en un grafo de productos y otro, en un grafo de usuarios. El proceso que se lleva a cabo para generar explicaciones con ambos modelos se describe en las Secciones 5.1.3 y 5.1.4, respectivamente.

Nuestra propuesta de métodos de explicación basados en grafos de interacción y técnicas de predicción de enlaces necesita ser evaluada para demostrar la hipótesis planteada. En la Sección 5.2 demostramos la efectividad de nuestra propuesta llevando a cabo una evaluación. Al igual que hicimos en el Capítulo 4, antes de explicar los resultados obtenidos (Sección 5.2.3), hacemos un análisis del conjunto de datos utilizado (Sección 5.2.1) y comentamos cómo se ha llevado a cabo el experimento (Sección 5.2.2). Al final del capítulo, en la Sección 5.3, extraemos unas conclusiones.

5.1. Explicaciones basadas en grafos de interacción

El método de explicación basado en grafos de interacción y técnicas de predicción de enlaces se puede definir como un modelo *surrogate local model-agnostic*. Un modelo *local model-agnostic* es un modelo post hoc (concepto que hemos detallado en la Sección 3.3.3.1) que intenta explicar el comportamiento de un modelo de caja negra, centrándose en una parte del conocimiento para proporcionar explicaciones. A cambio, los usuarios pueden

entender mejor cómo funciona el sistema al ser más interpretables que los modelos globales (Ribeiro et al., 2016; Singh y Anand, 2019; Ribeiro et al., 2018). Un modelo *surrogate* es un método interpretable que apoya a otro más complejo (Arya et al., 2019).

De forma análoga a como generamos las recomendaciones, al obtener explicaciones para sistemas recomendadores de cajas negras, utilizamos el conocimiento de las interacciones entre los usuarios y los productos, las cuales siempre van a existir, y lo representamos en grafos. Es importante remarcar que nuestro método de explicación basado en grafos es independiente del sistema de recomendación: no requiere información sobre el funcionamiento del recomendador para obtener las explicaciones, ni sobre *ratings* o descripciones y, por lo tanto, es adecuado para complementar sistemas de recomendación de caja negra e idóneo para escenarios de mínimo conocimiento. Aunque la información adicional puede aumentar la efectividad de la explicación, también aumenta la dependencia del algoritmo de recomendación.

Nuestro sistema de explicación propuesto es un *sistema de explicación basado en ejemplos o casos explicativos*. Los casos explicativos son un conjunto de productos con los que el usuario objetivo ha interactuado anteriormente y que son similares al producto recomendado. Se le muestran al usuario objetivo los ejemplos de explicación para que compare estos productos con el producto recomendado y evaluar si la recomendación es adecuada e interesante para él o ella (Cai et al., 2019; van der Waa et al., 2021; Adhikari, 2018). Por ejemplo, una explicación de este tipo sería: “Se te ha recomendado el producto X porque es similar a otros productos que has comprado/visto/-leído, como Y, Z y K”. De este modo, la explicación se personaliza para este usuario.

La visión general de nuestro método de explicación se presenta en la Figura 5.1. En ella podemos ver dos partes diferenciadas: a la izquierda, el sistema recomendador, y a la derecha, el sistema de explicación. El sistema de recomendación es una caja negra de la que lo único que podemos obtener es su input (el conocimiento de las interacciones entre usuarios y productos) y su output (los productos recomendados). En nuestro sistema de explicación utilizamos este input como fuente de conocimiento para generar los grafos, cuya construcción es prácticamente la misma que la que se lleva a cabo para los métodos de recomendación y que se ha detallado en la Sección 4.1. A partir de estos grafos se aplican las técnicas de predicción de enlaces para obtener las similitudes entre los elementos del grafo, y así obtener el conjunto personalizado de casos de explicación que se le pueden mostrar al usuario junto a las recomendaciones.

5.1.1. Escenario de mínimo conocimiento

De igual forma que en el modelo de recomendación, no usamos el valor de la interacción, sino que una interacción será representada en el grafo como un

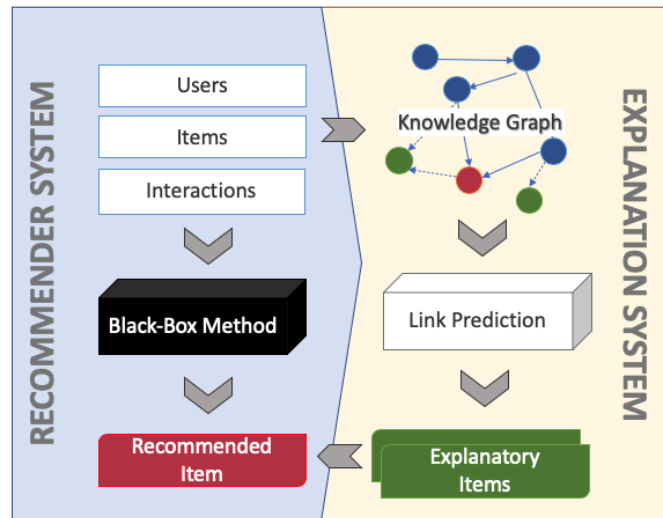


Figura 5.1: Visión general del sistema de explicación basado en grafos de interacción.

rating unario. Sin embargo, en el método de explicación no tendremos representadas únicamente interacciones positivas, como hacíamos en los modelos de recomendación. Tener en cuenta las interacciones negativas puede parecer inútil o incluso perjudicial en el contexto de una explicación pero, por otra parte, podemos perder conocimiento importante si no las consideramos. Esta decisión se basa en los siguientes argumentos:

1. **Ayudar a los usuarios a encontrar una correlación entre las recomendaciones y las interacciones anteriores.** Los productos que no han gustado al usuario tienen atributos que pueden gustarle. El usuario tenía un interés mínimo en este producto cuando interactuó con él. Por lo tanto, podría encontrar útil una recomendación similar. Por ejemplo, un usuario ha visto una película de terror que no le ha gustado. Esto no significa que no le guste este tipo de películas necesariamente. Tal vez esta película en particular no le gustó. Por lo tanto, si el sistema le recomienda una nueva película de terror, la película que había visto anteriormente puede ser una explicación razonable para esta recomendación. El usuario puede ver por qué el sistema le recomendó esta película y las conexiones entre las películas de acuerdo con sus preferencias.
2. **Ayudar a descartar recomendaciones.** Podría ser valioso para el usuario objetivo descartar las recomendaciones que no son interesantes. Aunque el sistema de recomendación haya utilizado las preferencias del usuario objetivo para obtener una recomendación, este elemento podría no ser una buena recomendación. Los usuarios tienen muchos

intereses diferentes, y necesitan diferentes productos dependiendo de su contexto (Quijano-Sanchez et al., 2017). Por ejemplo, un usuario objetivo quiere ver una película en Halloween con sus amigos y quieren ver una película de terror. Aunque al usuario objetivo le gustan las películas de Disney, estas películas no son adecuadas para él o ella en este contexto. Con las interacciones negativas, puede evaluar mejor las recomendaciones.

3. **Las interacciones negativas son útiles para proporcionar confianza y fidelidad.** Los usuarios necesitan confiar en el sistema, y sólo podemos proporcionarles confianza si los usuarios saben cómo funciona (Tintarev, 2007). El usuario necesita saber por qué el recomendador le ofrece una recomendación que no le interesa. Puede entender cómo funciona el sistema, incluso si éste se equivoca. Esta información proporciona confianza y fidelidad, aumentando la satisfacción del usuario (Walek y Fojtik, 2020).

Por tanto, vamos a construir los grafos de interacción de forma análoga a como lo hicimos en la Sección 4.1, pero incluyendo en la representación unaria las interacciones negativas o neutras. Partimos entonces de dos grafos distintos obtenidos de la proyección bipartita del grafo original de interacciones.

5.1.2. Definición del sistema de explicación propuesto según el modelo conceptual

Siguiendo el modelo conceptual definido en el Capítulo 3, podemos determinar algunos de los conceptos del método de explicación basado en grafos propuesto en este capítulo. Según el concepto de **Motivación**, podemos decir que el **Usuario** del sistema es un **Usuario objetivo**, ya que el usuario objetivo de la explicación es el mismo que el de la recomendación. Con respecto a los **Objetivos** que tendría nuestro método, el de **Alto nivel** sería **Justificar la recomendación**, mientras que el de **Bajo nivel** sería mejorar la **Efectividad**. Buscamos que el usuario conozca por qué un producto se le ha recomendado y que sepa encontrar cuáles son los productos que más le interesan. Creemos que la inclusión de interacciones negativas en el modelo podrían mejorar la efectividad del sistema de explicación. En cuanto al concepto de **Conocimiento**, como ya hemos comentado, sería la **Similitud del producto** o la **Similitud del usuario**, dependiendo de si el grafo tiene nodos productos o usuarios, respectivamente. Con respecto a la **Generación**, ya hemos comentado que los métodos basados en grafos tienen una **Encapsulación Post hoc**, mientras que la **Técnica** sería *Basada en conocimiento*. Por último nos quedaría definir las características con respecto al concepto de **Presentación**. El único concepto que podemos establecer es el de **Argumentación**, ya que los de **Formato**, **Nivel de detalle**

e **Interacción** dependen en gran medida de cómo sea la interfaz con la que se presente la explicación al usuario. Con respecto a la **Polaridad**, podemos decir que el sistema presenta argumentaciones **Positivas** y **Negativas**, ya que, al usar las interacciones negativas en el modelo, el usuario podría distinguir mejor si la recomendación no le interesa. En cuanto a la **Comparación**, el sistema emplearía argumentaciones basadas en **Similitudes**, ya que el funcionamiento de nuestra propuesta se fundamenta en la extracción de productos similares a otros con los que ha interactuado el usuario objetivo.

A continuación se van a definir los procesos llevados a cabo por los métodos basados en grafos de productos (Sección 5.1.3) y usuarios (Sección 5.1.4), respectivamente, para generar explicaciones basadas en ejemplos.

5.1.3. Proceso de explicación basado en grafos de productos

El proceso de explicación se define para un usuario objetivo u_t , que recibe la recomendación del producto i_r . El objetivo de nuestro sistema es recuperar la mejor lista de casos de explicación $E(i_r, u_t) = [i_j, \dots, i_k]$ que ayude a u_t a entender por qué el sistema de recomendación de caja negra recomendó i_r .

Definimos nuestro grafo de productos como $G_I = \langle N_I, L_I \rangle$, donde N_I es el conjunto de nodos que representan los productos, y L_I es el conjunto de enlaces que conectan los nodos. Podemos definir los enlaces como $L = \{(n_r, n_j, w_{rj}) | i_r, i_j \in I, r \neq j\}$. Los nodos n_r y n_j representan los elementos conectados por un enlace, y w_{rj} es el peso del enlace. El peso w_{rj} es el número de usuarios diferentes que han interactuado con ambos productos i_r e i_j . Si no hay ningún usuario que haya interactuado con ambos productos, entonces no existirá enlace entre n_r y n_j . Aplicamos también el filtro de los enlaces que comentamos en el capítulo anterior para eliminar todos los enlaces cuyo peso w_{rj} sea inferior a un valor θ .

La Figura 5.2 muestra el proceso de creación de la lista de ejemplos de explicación $E(i_r, u_t)$ para un usuario objetivo u_t que ha recibido i_r como recomendación. Los pasos seguidos en este proceso son los siguientes:

- (P.1.) Construir una matriz de similitud \mathbb{S}_I que contenga los valores de similitud entre todos los nodos del grafo G_I utilizando la medida de similitud de predicción de enlaces $LinkPred()$. Las medidas de similitud utilizadas en esta propuesta son las mismas que las de los métodos de recomendación basados en grafos (véase la Sección 4.1.3).
- (P.2.) Sabiendo que la función $item(n_r)$ devuelve el producto i_r que el nodo n_r representa, construir un conjunto de productos candidatos a ser ejemplos de explicación para u_t : $sim_t(n_r)$ que incluye los elementos más similares a i_r utilizando los valores de similitud en \mathbb{S}_I . El valor $s_{rj} = \mathbb{S}_I[n_r, n_j]$ representa la similitud entre i_r e i_j . El conjunto $sim_t(n_r)$ tiene que estar ordenado decrecientemente por valor de similitud.

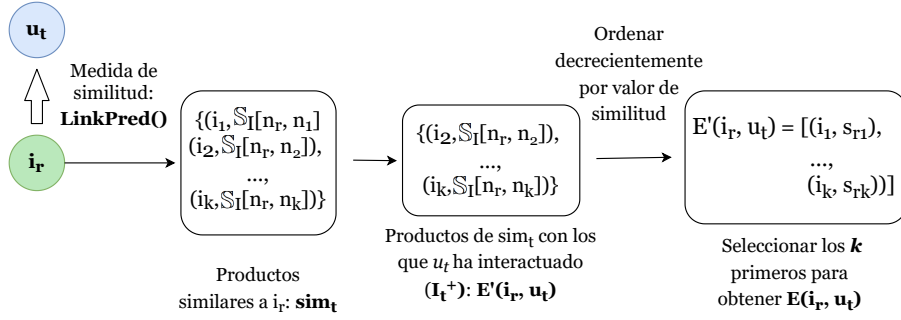


Figura 5.2: Proceso de explicación llevado a cabo con el grafo de interacción de productos.

$$sim_t(n_r) = \{(i_j, s_{rj}), (i_k, s_{rk}), \dots \mid s_{rj} \geq s_{rk}\} \quad (5.1)$$

$$\text{donde, } \forall (i_x, s_{rx}) \in sim_t(n_r) \quad : \quad \begin{aligned} i_x &= item(n_x) \\ s_{rx} &= S_I[n_r, n_x] \end{aligned}$$

- (P.3.) Obtener un subconjunto de $sim_t(n_r)$ eliminando todos los productos de este conjunto con los que u_t no ha interactuado todavía y que conformen el conjunto de potenciales casos de explicación: $E'(i_r, u_t)$. Para ello, hay que calcular previamente el conjunto de productos con los que u_t ha interactuado: $I_t^+ \subset I$.

$$E'(i_r, u_t) = \{(i_j, s_{rj}), (i_k, s_{rk}), \dots \mid s_{rj} \geq s_{rk}\} \quad (5.2)$$

$$\text{donde, } \forall (i_x, s_{rx}) \in E'(i_r, u_t) \quad : \quad \begin{aligned} (i_x, s_{rx}) &\in sim_t(n_r) \\ i_x &\in I_t^+ \end{aligned}$$

- (P.4.) Teniendo el conjunto $E'(i_r, u_t)$ ordenado de forma decreciente por valor de similitud, escoger los primeros k elementos de este conjunto. Estos productos van a ser los elementos que forman la lista de ejemplos de explicación $E(i_r, u_t) = [i_j, \dots, i_z]$ para la recomendación i_r y el usuario objetivo u_t .

5.1.4. Proceso de explicación basado en grafos de usuarios

En este caso, el método de explicación genera los casos de explicación a partir del grafo de usuarios, que definimos como $G_U = \langle N_U, L_U \rangle$ donde el conjunto de nodos N_U representa los usuarios U del sistema. Un enlace del conjunto $L_U = \{(n_t, n_v, w_{tv}) \mid u_t, u_v \in U, t \neq v\}$ conecta dos nodos u_t y u_v

cuando han interactuado con al menos un producto en común. El peso del enlace w_{tv} es el número de productos con los que ambos han interactuado. De nuevo, al tratarse de un grafo con alta densidad, aplicamos filtrado de enlaces para eliminar todos los enlaces cuyo peso w sea inferior al valor umbral θ .

El proceso para crear la lista de casos de explicación $E(i_r, u_t)$ para un producto recomendado i_r al usuario u_t es el siguiente (Figura 5.3):

- (P.1.) Construir una matriz de similitud \mathbb{S}_U que guarde todos los valores de similitud que existen entre cada par de nodos del grafo utilizando la medida basada en predicción de enlaces $LinkPred()$. De nuevo, $LinkPred()$ es una de las medidas de similitud de predicción de enlaces propuestas en la Sección 4.1.3).
- (P.2.) A partir de \mathbb{S}_U , obtener el conjunto $U_t \subset U$ que contiene los usuarios más parecidos a u_t . Definir el conjunto de productos con los que u_t ha interactuado previamente ($I_t^+ \in I$) y el conjunto de productos con los que los usuarios $u_v \in U_t$ han interactuado ($I_v^+ \in I$).
- (P.3.) Sabiendo que $user(n_v)$ devuelve el usuario u_v que el nodo n_v representa, para cada nodo n_v tal que $user(n_v) \in U_t$, calcular el conjunto $sim_t(n_v)$ de productos con los que ambos usuarios u_t y u_v han interactuado. Todos los productos de $sim_t(n_v)$ tienen el mismo valor de similitud asociado, que es el valor de similitud entre los nodos n_t y n_v que representan a u_t y u_v . El conjunto $sim_t(n_v)$ se define como:

$$sim_t(n_v) = \{(i_j, s_{tv}), (i_k, s_{tv}), \dots\} \quad (5.3)$$

donde, $\forall (i_x, s_{tv}) \in sim_t(n_v) : \begin{aligned} & i_x \in I_v^+ \cap I_t^+ \\ & user(n_v) \in U_t \\ & s_{tv} = \mathbb{S}_U[n_t, n_v] \end{aligned}$

- (P.4.) Obtener una lista $gsim_t$ con todos los productos, junto a sus valores de similitud, almacenados en los conjuntos $sim_t(n_v)$ de cada $user(n_v) \in U_t$. Por tanto, $gsim_t$ va a contener el conjunto de productos con los que u_t y U_t han interactuado, pudiendo haber repeticiones de un mismo producto con diferente valor de similitud asociado.

$$gsim_t = \bigcup_{user(n_v) \in U_t} sim_t(n_v) \quad (5.4)$$

- (P.5.) Eliminar los productos repetidos en $gsim_t$. Para ello, para cada elemento repetido en $gsim_t$, se guarda ese elemento con el mayor valor

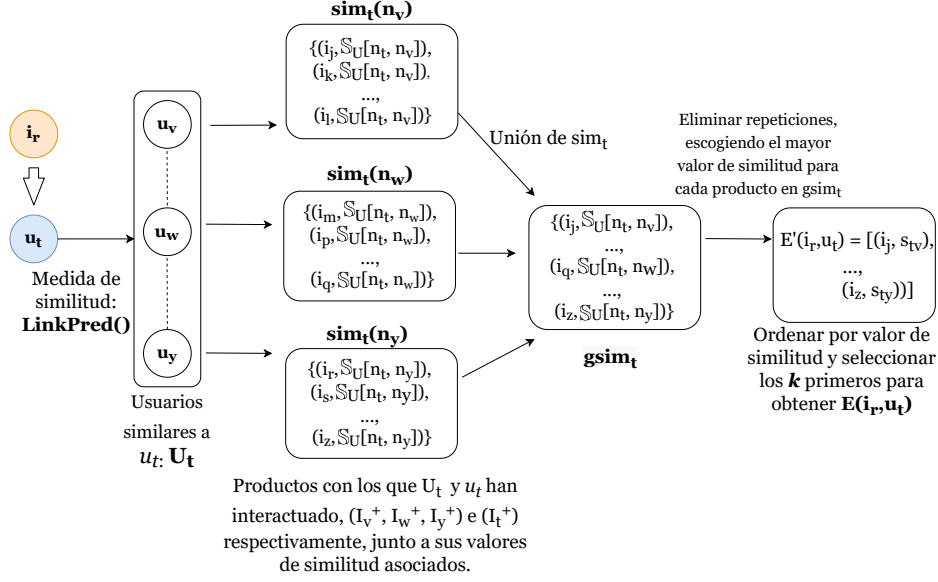


Figura 5.3: Proceso de explicación llevado a cabo con el grafo de interacción de usuarios.

de similitud que tenga en gsim_t . Como resultado, se obtiene una lista de productos candidatos a ser los ejemplos de explicación, en la que hay una única aparición de cada producto candidato.

$$E'(i_r, u_t) = \{(i_j, s_{tv}), (i_k, s_{tw}), \dots \mid s_{tv} \geq s_{tw}\} \quad (5.5)$$

donde, $\forall (i_x, s_{tx}) \in E'(i_r, u_t) : \{(i_x, s_{tw}), \dots, (i_x, s_{tv}) \mid s_{tv} \geq s_{tw}\} \in \text{gsim}_t$
 $i_x \in I_t^+ \cap I_v^- \cap \dots \cap I_w^-$

- (P.6.) Estando el conjunto $E'(i_r, u_t)$ ordenado decrecientemente por valor de similitud, elegir los k primeros productos de $E'(i_r, u_t)$ para formar la lista de casos de explicación $E(i_r, u_t) = [i_j, \dots, i_z]$ que se va a mostrar al usuario objetivo u_t con la recomendación i_r .

5.2. Evaluación

En la Hipótesis 2 de la tesis suponemos que podemos generar explicaciones con grafos de interacción usando una mínima cantidad de conocimiento independientemente del tipo de recomendador. Para comprobar esta teoría, nuestro objetivo en la evaluación llevada a cabo es demostrar que nuestro método es tan efectivo como otros sistemas de explicación basados en explicación, pero usando una menor cantidad de conocimiento. De esta forma,

podemos proponer una solución de sistema de explicación para recomendadores que no necesite acceder a las fuentes de conocimiento tradicionales, como *ratings* y descripciones. Tampoco necesitamos información de cómo funciona el recomendador al que apoya, lo que lo hace independiente del sistema de recomendación. Hemos diseñado una evaluación en la que comparamos los métodos basados en grafos con un sistema de explicación basado en contenido que utiliza las descripciones de los productos para generar los casos de explicación (Sørmo et al., 2005). Al igual que nuestros métodos, este sistema proporcionará casos de explicación para un usuario objetivo y un determinado producto que se le recomienda al usuario. Además, al contrario que nuestra propuesta, es un modelo global, lo que también puede influir en la interpretabilidad del sistema.

Para nuestra evaluación, necesitamos tener un sistema recomendador del que obtener elementos recomendados para un usuario objetivo. Hemos decidido implementar un sistema de recomendación basado en factorización de matrices porque es uno de los algoritmos más eficaces para generar recomendaciones hoy en día (Bennett et al., 2007). Sin embargo, es un algoritmo complejo que no es transparente ni fácil de entender para los usuarios que reciben las recomendaciones. Por ello, los recomendadores basados en factorización de matrices son un buen ejemplo de sistema que tiene la necesidad de un sistema de explicación. En el proceso de evaluación descrito a continuación, utilizamos este recomendador como un sistema de caja negra, ya que los métodos basados en grafos son métodos *surrogate local model-agnostic*. Cabe destacar que el haber usado filtrado colaborativo basado en factorización de matrices en la evaluación no quiere decir que nuestros métodos sólo se puedan usar para este tipo de recomendadores. Se pueden aplicar a cualquier sistema de recomendación, siendo más interesante usar nuestra propuesta en casos de recomendadores de cajas negras, ya que nuestros modelos basados en grafos no requieren información de cómo funciona el algoritmo de recomendación. La implementación de los modelos basados en grafos evaluados se encuentra en un repositorio de GitHub¹.

A continuación, detallamos la evaluación llevada a cabo. En la Sección 5.2.1, describimos el conjunto de datos que hemos utilizado para realizar la evaluación. En la siguiente Sección 5.2.2, explicamos la metodología de evaluación utilizada. Por último, en la Sección 5.2.3 se discuten los resultados de la evaluación.

5.2.1. Análisis del conjunto de datos

En el experimento, combinamos dos conjuntos de datos diferentes para construir el conjunto de datos a utilizar en la evaluación. Por un lado, de nuevo utilizamos el conjunto de datos de MovieLens 100K ya que es muy

¹<https://github.com/martcaro/GraphBasedExplanations>

usado en la literatura para evaluar los sistemas de recomendación. De este dataset, sólo las tuplas $\langle u, i \rangle$, donde u es un usuario del sistema e i en una película con la que ha interactuado u , es la información que requiere nuestro método de explicación basado en grafos. Sin embargo, el sistema basado en contenido que usamos en la evaluación requiere más conocimiento para comparar y encontrar las similitudes entre los productos. Por lo tanto, necesitamos un conjunto de datos adicional con información extra sobre los productos recomendados.

El conjunto de datos elegido es el de IMDb². Este conjunto de datos contiene información sobre géneros, actores, directores, etc. de 5.000 películas. Esta información complementará el conjunto de datos de MovieLens para implementar el sistema de explicación basado en contenido. Sin embargo, no todas las películas del conjunto de datos de MovieLens aparecen en el de IMDb. Por lo tanto, filtramos el conjunto de datos de MovieLens para sólo tener las interacciones de las películas que también aparecen en el conjunto de datos de IMDb. Hemos denominado D al conjunto de datos resultante. Dividimos D en el conjunto de entrenamiento D_t con el 90 % de las interacciones, elegidas aleatoriamente, y el conjunto de evaluación D_e con el 10 % restante.

Antes de realizar la evaluación, llevamos a cabo un análisis sobre el conjunto de datos, de igual forma a la que hicimos en la Sección 4.2.1. La Tabla 5.1 muestra los resultados de este análisis.

A partir de este análisis, encontramos un sesgo significativo en el conjunto de datos D_e : está desequilibrado en cuanto al número de elementos asociados a cada valor de *rating*. En consecuencia, decidimos crear un conjunto de evaluación estratificado D'_e que evita este sesgo porque contiene el mismo número de interacciones (35) para cada valor de *rating*. Hemos seleccionado esta cantidad de elementos porque es la cantidad de interacciones para el menor valor de *rating* (2,5) que existe en el conjunto de datos.

5.2.2. Metodología de evaluación

El proceso de experimentación comienza con la construcción de los métodos de explicación basados en grafos y basados en contenido usando el conjunto de entrenamiento. Implementamos varias versiones de los métodos basados en grafos con respecto a las medidas de similitud descritas en la Sección 4.1.3: AA, CN, EW, JN, PA, WAA, WCN y WPA. También configuramos el umbral $\theta = 5$ (véase la Sección 5.1) ya que consideramos este valor suficiente para reducir la densidad sin eliminar el conocimiento esencial. El sistema de explicación basado en contenido con el que nos comparamos en la evaluación recupera los productos más similares al producto recomendado i_r teniendo en cuenta las características de las películas en el conjunto de

²<https://www.imdb.com/>

Metric	ML	D	D_t	D_e	D'_e
# Ratings	100.000	11.477	10.330	1.147	280
# Productos	1.682	164	164	145	109
# Usuarios	943	587	584	394	134
Densidad	0,06	0,12	0,11	0,02	0,02
Productos					
Máximo # ratings por producto	583	329	305	30	10
Mediana del # ratings por producto	27	43,5	39	5	2
Media del # ratings por producto	59,45	69,98	62,99	7,91	2,57
Mínimo # ratings por producto	1	1	1	1	1
Usuarios					
Máximo # ratings por usuario	737	128	113	15	11
Mediana del # ratings por usuario	65	12	11	2	1
Media del # ratings por usuario	106,05	19,55	17,69	2,91	2,09
Mínimo # ratings por usuario	20	1	1	1	1
Ratings					
% Ratings ≤ 2	17,48	11,36	11,35	11,5	25
% Ratings >2 y ≤ 3	27,14	28,20	28,14	28,77	25
% Ratings >3 y ≤ 4	34,17	35,70	35,80	34,87	25
% Ratings >4 y ≤ 5	21,20	24,73	24,71	24,85	25

Tabla 5.1: Análisis de los conjuntos de datos utilizados en la evaluación. ML es el conjunto de datos original de MovieLens con 100.000 interacciones. D es el subconjunto de datos de MovieLens que contiene aquellas interacciones en las que intervienen películas que aparecen en el dataset de IMDb. D_t es el conjunto de entrenamiento. D_e es el conjunto de datos de evaluación. D'_e es el conjunto de evaluación estratificada, en el que se ha eliminado el sesgo de D_e .

datos de IMDb. En este caso, utilizamos la distancia Euclídea, la similitud de coseno y el coeficiente de Jaccard como medidas de similitud, ya que son algunas de las más utilizadas en la literatura. Jaccard es una medida especialmente interesante como *baseline* de nuestra evaluación ya que funciona de forma análoga a como funcionan las medidas de similitud utilizadas por nuestros grafos de interacción. Jaccard consideraría dos elementos similares si tienen el mismo atributo. Sin embargo, si ninguno de los dos productos tienen un determinado atributo, no los considera similares, como sí hacen las otras medidas de similitud. En este punto, es importante destacar dos características principales del método de explicación basado en contenido: (1) como hemos comentado anteriormente, necesita más conocimiento que el método de explicación basado en grafos y; (2) no proporciona explicaciones personalizadas para el usuario objetivo u_t , sino que simplemente busca casos de explicación similares al producto recomendado i_r , sin tener en cuenta a u_t .

Tanto los métodos basados en grafos como el basado en contenido generan la lista $E(i_r, u_t)$ de casos de explicación ordenados decrecientemente por importancia. El tamaño de la lista k también se ha evaluado adquiriendo distintos valores $k \in \{1, 10\}$. El proceso experimental continúa evaluando la efectividad de los métodos de explicación a partir de la similitud entre

el producto recomendado i_r y los mejores casos de explicación en $E(i_r, u_t)$. Usamos la métrica del error cuadrático medio (*Root Mean Square Error*: RMSE) para comparar el *rating* de i_r predicho por el sistema de recomendación basado en factorización de matrices y los *ratings* reales en el conjunto de evaluación de los productos de explicación en $E(i_r, u_t)$ recuperados por cada método. Los valores RMSE más bajos representan que el método es más preciso obteniendo los casos de explicación. Los valores más altos de RMSE indican mayor error de los métodos y peor precisión. Es importante señalar que, para cada k , hemos eliminado a los usuarios que no interactúan con al menos k películas en el conjunto de evaluación. Por lo tanto, si un usuario objetivo u sólo ha valorado cuatro películas, entonces no es apto para ser evaluado cuando la lista de casos de explicación tiene $k \geq 5$. Para que este proceso sea fiable, hemos realizado la evaluación sobre los dos conjuntos de evaluación descritos anteriormente: D_e y D'_e . La evaluación estratificada se repite 100 veces, obteniendo aleatoriamente D'_e de D_e en cada iteración.

5.2.3. Análisis de los resultados

Las Tablas 5.2 y 5.3 muestran las medias de los valores de RMSE obtenidos para cada modelo basado en grafos y basado en contenido usando el conjunto de datos original (D_e) y el conjunto de datos estratificado (D'_e), respectivamente. Llama la atención la similitud de los resultados obtenidos en ambos conjuntos de datos, aunque esta similitud no se da entre los métodos evaluados.

En la Tabla 5.2, podemos observar las diferencias entre las medidas de similitud utilizadas en la evaluación y su tasa de error cuando se aplican al conjunto de datos original D_e . Para el método basado en el grafo de productos, los valores de la medida de similitud JN destacan sobre el resto. Este comportamiento se puede apreciar para todos los valores de k . JN siempre mejora la precisión de las otras medidas de similitud con una diferencia de aproximadamente 10 % a 20 %. Nuestra explicación para este comportamiento es que JN considera el número de vecinos comunes en comparación con el número de vecinos totales para obtener los elementos más similares al recomendado. Por tanto esta medida tiene en cuenta la similitud sobre la diversidad de los conjuntos.

Por otro lado, no encontramos este patrón en el métodos basado en el grafo de usuarios. Los resultados de todas las medidas de similitud son muy similares y no hay una claramente mejor. Cuando $k = 1$, WAA y WCN son las medidas que encuentran el elementos más similar. Para el resto de valores de k , la mejor medida de similitud varía entre CN y EW, siendo CN la que mayor número de veces es superior. Podemos concluir que CN puede ser la mejor medida de similitud para el método basado en el grafo de usuarios, pero las medidas de similitud que usan la información de los pesos de los enlaces también consiguen resultados destacables.

	1	2	3	4	5	6	7	8	9	10
I - AA	1,095	0,865	0,792	0,769	0,747	0,723	0,702	0,686	0,678	0,664
I - CN	1,035	0,821	0,750	0,713	0,683	0,666	0,649	0,640	0,636	0,635
I - EW	1,087	0,878	0,797	0,754	0,734	0,706	0,679	0,666	0,651	0,639
I - JN	0,961	0,734	0,658	0,624	0,599	0,573	0,562	0,549	0,540	0,534
I - PA	1,126	0,897	0,806	0,789	0,782	0,762	0,741	0,728	0,718	0,706
I - WAA	1,113	0,968	0,908	0,855	0,822	0,794	0,775	0,747	0,723	0,705
I - WCN	1,113	0,968	0,907	0,855	0,821	0,796	0,778	0,749	0,723	0,704
I - WPA	1,115	0,970	0,910	0,852	0,821	0,793	0,779	0,747	0,723	0,703
U - AA	0,865	0,798	0,759	0,741	0,720	0,705	0,696	0,690	0,688	0,685
U - CN	0,877	0,788	0,758	0,734	0,715	0,701	0,692	0,688	0,685	0,681
U - EW	0,871	0,778	0,744	0,736	0,718	0,699	0,695	0,684	0,688	0,684
U - JN	0,874	0,805	0,777	0,745	0,728	0,711	0,704	0,703	0,695	0,686
U - PA	0,865	0,791	0,752	0,742	0,720	0,706	0,695	0,690	0,689	0,685
U - WAA	0,864	0,793	0,753	0,743	0,720	0,706	0,695	0,690	0,690	0,684
U - WCN	0,864	0,793	0,753	0,743	0,720	0,706	0,695	0,690	0,690	0,684
U - WPA	0,865	0,791	0,752	0,743	0,720	0,706	0,695	0,690	0,690	0,684
Cosine	0,973	1,036	1,064	1,078	1,087	1,100	1,101	1,104	1,108	1,111
Euclidean	0,966	1,032	1,063	1,079	1,092	1,092	1,099	1,100	1,102	1,105
Jaccard	0,974	1,037	1,064	1,078	1,087	1,099	1,101	1,104	1,109	1,111

Tabla 5.2: Valores de RMSE obtenidos en la evaluación con los métodos de explicación aplicados sobre el conjunto de evaluación original (D_e). Los valores de las columnas corresponden a $k \in \{1, 10\}$. Los mejores resultados entre las medidas de similitud de cada modelo aparecen en negrita. Los mejores valores en cada columna están subrayados. I hace referencia al método basado en grafos de productos, mientras que U, al de grafos de usuarios. Al lado de I o U se indica la medida de similitud basada en predicción de enlaces utilizada.

En cuanto a los resultados del método basado en contenido, la mejor medida de similitud también está clara: la distancia Euclídea. Para casi todos los valores de k , ésta es la medida que mejor funciona. También se puede observar que la diferencia entre los resultados de las medidas de similitud es mínima, siendo incluso menor que las diferencias que se pueden percibir entre los resultados de los métodos basados en grafos.

Además, podemos obtener nuevas conclusiones comparando los mejores resultados conseguidos con cada método de explicación. Este análisis se muestra en la Figura 5.4, donde presentamos los resultados de cada método con su mejor medida de similitud: JN para el método basado en el grafo de productos, CN para el grafo basado en los usuarios y la distancia Euclídea para el método basado en contenido.

Por un lado, se observa un comportamiento heterogéneo respecto al parámetro k . En el caso del sistema basado en contenido, el valor del RMSE aumenta cuanto más alto sea el valor de k , aunque se va estabilizando con el incremento. Por lo tanto, la tasa de error del sistema empeora. Sin embargo, en el caso de los modelos basados en grafos, el comportamiento es totalmente

	1	2	3	4	5	6	7	8	9	10
I - AA	1,164	0,947	0,861	0,839	0,817	0,796	0,774	0,764	0,754	0,738
I - CN	1,084	0,893	0,811	0,761	0,739	0,721	0,696	0,695	0,695	0,698
I - EW	1,142	0,938	0,863	0,827	0,802	0,766	0,743	0,736	0,714	0,702
I - JN	1,004	0,748	0,673	0,643	0,629	0,597	0,584	0,573	0,566	0,566
I - PA	1,182	1,000	0,902	0,883	0,873	0,852	0,835	0,819	0,809	0,793
I - WAA	1,156	1,061	1,002	0,944	0,922	0,894	0,874	0,838	0,810	0,790
I - WCN	1,149	1,054	0,994	0,937	0,917	0,890	0,873	0,837	0,807	0,787
I - WPA	1,152	1,065	1,011	0,940	0,921	0,889	0,873	0,837	0,812	0,789
U - AA	0,883	0,830	0,794	0,765	0,741	0,731	0,717	0,716	0,715	0,713
U - CN	0,898	0,820	0,787	0,758	0,740	0,727	0,715	0,715	0,713	0,710
U - EW	0,888	0,804	0,758	0,749	0,733	0,710	0,707	0,698	0,706	0,704
U - JN	0,881	0,833	0,811	0,771	0,755	0,738	0,733	0,733	0,725	0,716
U - PA	<u>0,883</u>	0,820	0,778	0,765	0,740	0,727	0,709	0,708	0,711	0,708
U - WAA	0,879	0,821	0,779	0,766	0,740	0,728	0,710	0,709	0,712	0,708
U - WCN	0,881	0,823	0,779	0,767	0,741	0,729	0,711	0,710	0,713	0,709
U - WPA	<u>0,881</u>	0,820	0,778	0,767	0,742	0,730	0,712	0,712	0,715	0,711
Cosine	1,117	1,130	1,125	1,125	1,125	1,120	1,129	1,125	1,121	1,121
Euclidean	1,090	1,105	1,102	1,110	1,105	1,106	1,106	1,111	1,112	1,112
Jaccard	1,052	1,052	1,054	1,067	1,084	1,097	1,100	1,099	1,102	1,100

Tabla 5.3: Valores de RMSE obtenidos en la evaluación con los métodos de explicación aplicados sobre el conjunto de evaluación estratificado (D'_e). Los valores de las columnas corresponden a $k \in \{1, 10\}$. Los mejores resultados entre las medidas de similitud de cada modelo aparecen en negrita. Los mejores valores en cada columna están subrayados. I hace referencia al método basado en grafos de productos, mientras que U, al de grafos de usuarios. Al lado de I o U se indica la medida de similitud basada en predicción de enlaces utilizada.

diferente: al aumentar k , el valor del RMSE disminuye hasta estabilizarse. La estabilización de RMSE cuanto mayor es k se debe a que el algoritmo recupera una mayor cantidad de elementos de explicación, por tanto, es más difícil cometer un error significativo. Vemos que aproximadamente a partir de $k = 4$ o $k = 5$ es cuando se produce esta estabilización, lo que significa que aportar más de 4 ejemplos de explicación no contribuye a aumentar (ni disminuir) la efectividad de la explicación.

De forma global, los mejores valores de RMSE los obtienen siempre los métodos basados en grafos. Cuando $k = 1$, los mejores resultados se obtienen con la propuesta basada en el grafos de usuarios, aunque su efectividad es similar a la obtenida con los otros dos sistemas. En el resto de los casos, los mejores resultados se obtienen con la propuesta basada en el grafos de productos. La diferencia entre los resultados alcanzados con los métodos basados en grafos no es muy significativa, pero la diferencia es muy notable entre los sistemas basados en grafos y el basado en contenido. En la Figura 5.4, se puede observar claramente esta diferencia, especialmente cuando k se estabiliza. También podemos concluir que el método basado en el grafo de productos

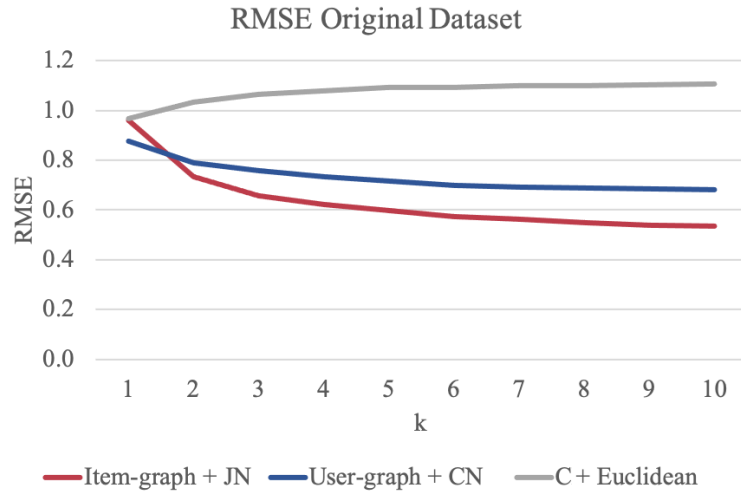


Figura 5.4: Gráfico que representa los resultados obtenidos con el conjunto de datos original (D_e). Para cada modelo estudiado, hemos elegido representar las medidas de similitud que obtienen mejores resultados en la evaluación. En el eje Y representamos el valor del RMSE. Consideramos el número de casos de explicación recuperados en el eje X .

se comporta mejor porque la recuperación de productos similares es sencilla. En este método, obtenemos los ejemplos de explicación directamente desde el grafo de productos. En el grafo basado en usuarios, aplicamos las medidas de similitud para obtener los usuarios más parecidos al usuario objetivo y, a continuación, obtenemos los productos con los que estos usuarios similares han interactuado anteriormente. Por tanto, no es un método tan inmediato como nuestra otra propuesta.

La Tabla 5.3 muestra los resultados de la evaluación con el conjunto de datos estratificado D'_e . La tendencia de los resultados es similar a la que encontramos en la Tabla 5.2.

En los resultados del método basado en el grafo de productos, podemos ver que JN es la medida de similitud que mejores resultados aporta. Por lo tanto, la eliminación del sesgo del conjunto de datos no cambia la comparación entre las medidas de similitud que vimos en la tabla anterior. De nuevo, podemos observar la diferencia destacable entre JN y el resto de las medidas de similitud.

En cuanto al método basado en el grafo de usuarios, se pueden observar conclusiones parecidas a las que vimos en la Tabla 5.2. Con $k = 1$, WCN, WPA y JN obtienen el mejor valor de RMSE. Podemos observar que para el resto de valores de k , CN no consigue obtener resultados tan prometedores como en la Tabla 5.2, siendo EW la mejor medida de similitud para $k \geq 2$. Por lo tanto, podemos decir que las medidas de similitud que funcionan en

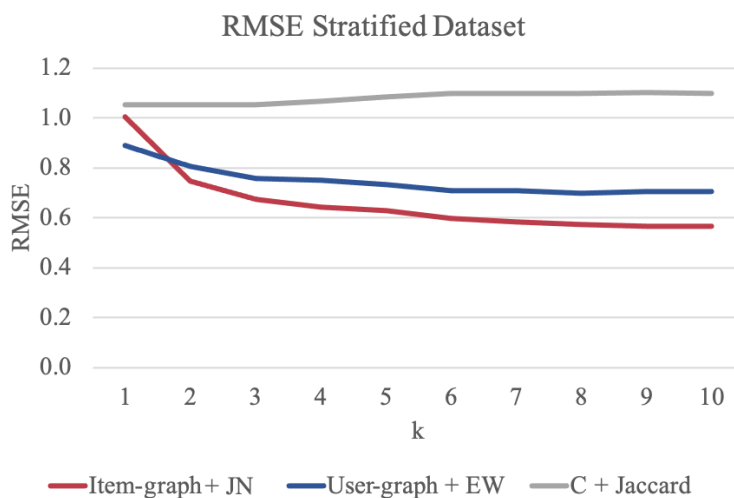


Figura 5.5: Gráfico que representa los resultados obtenidos con el conjunto de datos estratificado (D'_e). Para cada modelo estudiado, hemos elegido representar las medidas de similitud que obtienen mejores resultados en la evaluación. En el eje Y representamos el valor del RMSE. Consideramos el número de casos de explicación recuperados en el eje X .

base al conocimiento del peso de los enlaces proporcionan mejores resultados en el modelo basado en grafos de usuarios.

En el caso del sistema basado en el contenido, las conclusiones cambian. En la Tabla 5.2, la distancia Euclídea alcanza las mejores puntuaciones para la mayoría de los valores de k . Sin embargo, en la Tabla 5.3, es Jaccard la medida que consigue los mejores resultados.

Para comparar la efectividad de los tres modelos cuando se aplican al conjunto de datos estratificado D'_e , hemos creado la Figura 5.5. También hemos decidido representar en esta gráfica la mejor medida de similitud para cada método: la medida de similitud JN para el método basado en el grafos de productos, EW para el método basado en el grafo de usuarios y Jaccard para el método basado en contenido. De nuevo, se puede distinguir la misma tendencia que vimos con el conjunto de datos original, consiguiendo los grafos (especialmente el de productos) mejores valores de RMSE que el basado en contenido. También nos encontramos aquí la estabilización de los resultados con el aumento de k .

Si comparamos los resultados obtenidos en las Tablas 5.2 y 5.3 y las Figuras 5.4 y 5.5, se puede observar que los resultados son muy similares. Las formas de los gráficos son prácticamente idénticas, por lo que podemos concluir que, al contrario de nuestra suposición inicial, el sesgo no tiene un impacto relevante a la hora de generar las explicaciones. Este resultado supone una ventaja adicional a nuestra propuesta, ya que también es aplicable

		1	2	3	4	5	6	7	8	9	10
D_e	I-JN	0,49	28,83	38,09	42,18	45,17	47,50	48,83	50,10	50,99	51,64
	U-CN	9,26	23,65	28,67	31,94	34,50	35,83	36,99	37,46	37,84	38,35
D'_e	I-JN	4,56	28,90	36,15	39,74	41,97	45,58	46,91	47,86	48,64	48,55
	U-EW	15,56	23,54	28,10	29,82	32,34	35,29	35,72	36,45	35,91	36,03

Tabla 5.4: Porcentaje de mejora de los métodos basados en grafos respecto al modelo basado en contenido. El conjunto de evaluación utilizado en la comparación se señala en la primera columna a la izquierda. En la segunda columna, I hace referencia al grafo de productos y U, al grafo de usuarios. Al lado de I y U, se indica la medida de similitud de predicción de enlaces. El mejor porcentaje de mejora en cada fila está marcado en negrita.

en conjuntos de datos desbalanceados.

Como resumen global de los resultados obtenidos, en la Tabla 5.4 hemos decidido mostrar el porcentaje de mejora de las mejores configuraciones de los métodos basados en grafos con respecto a la mejor configuración del método basado en contenido. Las mejores configuraciones para cada método las obtenemos de los resultados mostrados en las Tablas 5.2 y 5.3. Las dos primeras filas hacen referencia a los resultados conseguidos con el conjunto de evaluación original D_e , mientras que las dos últimas hacen referencia a los del conjunto estratificado D'_e . Por ejemplo, en la primera fila de la Tabla 5.4, vemos el porcentaje de mejora del método basado en grafos de productos utilizando JN con respecto al sistema basado en contenido utilizando la distancia Euclídea para el conjunto de evaluación original D_e . Para todas las filas, podemos ver que los modelos basados en grafos mejoran la efectividad de las explicaciones, llegando a ser un 50 % mejores que los basados en contenido.

Como conclusión final de esta evaluación, podemos decir que los resultados que hemos obtenido validan nuestra Hipótesis 2. Conseguimos mejores resultados con los métodos basados en grafos que con el método basado en contenido, independientemente del tipo de grafo o de la medida de similitud utilizada, tanto con el conjunto de datos original como con el estratificado. Por lo tanto, podemos considerar que nuestras propuestas basadas en grafos podrían funcionar mejor que los modelos basados en contenido para proporcionar explicaciones para recomendadores, utilizando además una menor cantidad de conocimiento, siendo personalizadas al usuario y generadas independientemente del proceso de recomendación.

Respecto a la comparación entre el método basado en un grafo de usuarios y el método basado en un grafo de productos, podemos concluir que el modelo basado en productos podría resultar más efectivo para un usuario objetivo porque en el proceso de explicación más sencillo y en el que interviene el producto recomendado, al contrario de lo que ocurre en el modelo basado en

el grafo de usuarios. Además, las explicaciones basadas en usuarios podrían ser menos fiables que las basadas en productos, porque los usuarios objetivo están más familiarizados con los productos con los que han interactuado que con usuarios similares a los que no conocen.

5.3. Conclusiones

En este capítulo se propone un nuevo método *surrogate local model-agnostic* para proporcionar explicaciones a los sistemas de recomendación de caja negra utilizando grafos de conocimiento. Con esta propuesta se valida la Hipótesis 2 de la tesis, cumpliendo el Objetivo 2. Esta propuesta es una solución para poder implementar sistemas de explicación en situaciones en las que técnicas clásicas no pueden aplicarse.

El conocimiento que siempre se puede obtener de un sistema de recomendación es el de las interacciones previas que se han producido entre los usuarios y los productos. En esta tesis doctoral, proponemos únicamente representar este conocimiento en un grafo, de productos o de usuarios, para generar un método de explicación para recomendadores. Esta propuesta no necesita información sobre *ratings* o descripciones ni ningún otro conocimiento adicional del sistema de recomendación. Con el método de explicación basado en grafos se le puede proporcionar al usuario objetivo de la recomendación una explicación personalizada basada en ejemplos.

Una de las principales novedades de este modelo propuesto con respecto al de recomendación, descrito en el Capítulo 4, es que representamos todas las interacciones realizadas por los usuarios (positivas, negativas y neutras). La representación se hace de igual manera para todos los tipos de interacciones, no se tienen en cuenta sus valores. En consecuencia, los ejemplos de explicación que se ofrecen a los usuarios objetivo pueden ser productos que no les han gustado; por lo tanto, podrían decidir mejor si el producto recomendado es de su interés o no.

Para validar nuestro método, realizamos una evaluación experimental. Su objetivo es comparar la efectividad de nuestros métodos con la de una técnica clásica de explicación que necesite mayor cantidad de conocimiento para generar explicaciones: un sistema de explicación basado en contenido. En los resultados de evaluación los modelos basados en grafos logran una efectividad mayor que el basado en contenido, necesitando un nivel de conocimiento menor. Además, cuanto más crece la lista de ejemplos de explicación, mejor es la precisión del sistema basado en grafos, en contraste con el comportamiento del sistema basado en contenido. En general, el grafo basado en productos parece ser el método más eficaz. Podemos concluir que nuestros métodos de explicación basados en grafos de conocimiento son una buena solución para explicar recomendaciones de caja negra cuando no podemos aplicar otras técnicas clásicas e incluso también en situaciones en las que estas técnicas

se pueden usar ya que, al estar personalizadas, pueden ofrecer ejemplos de explicación más efectivos para el usuario objetivo de la recomendación.

El modelo de explicación basado en grafos de interacción de productos ha sido publicado en el congreso *International Conference on Case-Based Reasoning* de 2019 (Caro-Martinez et al., 2019b). El trabajo completo incluyendo la comparativa entre los modelos basados en grafos de productos y usuarios se encuentra aceptado en la revista *International Journal of Interactive Multimedia and Artificial Intelligence* aunque pendiente de publicar en el momento de escritura de esta memoria (Caro-Martinez et al., Pendiente de publicar).

Una vez validada la utilidad de los grafos de interacción como método de explicación de recomendadores, en el siguiente capítulo evaluaremos la idoneidad de los grafos como método de visualización de las explicaciones.

Capítulo 6

Visualización de explicaciones basada en grafos

Para que las explicaciones en los sistemas de recomendación sean útiles tienen que estar diseñadas de acuerdo a las necesidades del usuario objetivo, como ya vimos en el Capítulo 3 y se demuestra con los resultados del Capítulo 5. Por ello, en la generación de sistemas de explicación, la fuente de conocimiento que utiliza el sistema es tan importante como el contenido de las explicaciones y la forma de presentarlas al usuario objetivo.

En esta tesis doctoral se ha propuesto el uso de grafos como estructura computacional para generar recomendaciones y explicaciones personalizadas a los usuarios. Sin embargo, los grafos también se utilizan como forma de visualización de datos y redes sociales (Chen et al., 2015; Beck et al., 2017; Herman et al., 2000). Además, con el incremento del interés y la necesidad de la XAI, se han propuesto nuevas técnicas de visualización en sistemas como los de recomendación, siendo los grafos una de las propuestas recurrentes en la literatura, como hemos visto al revisar el estado del arte en el Capítulo 2 (Parra et al., 2014; Verbert et al., 2013; Kangasrääsiö et al., 2015; O'Donovan et al., 2008). Los distintos componentes de los grafos (nodos, enlaces, caminos, etc.) permiten explicar recomendaciones de formas diversas, incluyendo en muchos casos una interactividad que ayuda a explorar las explicaciones y recomendaciones de una forma más innovadora (Andjelkovic et al., 2019; Kangasrääsiö et al., 2015; O'Donovan et al., 2008). Es por esto que en esta tesis doctoral nos hayamos propuesto estudiar el uso de grafos como método de visualización de explicaciones para recomendadores.

En este capítulo, tratamos de abordar el Objetivo 3 de la tesis, donde se plantea *analizar los grafos como método de visualización para explicaciones en recomendadores* con el propósito de comprobar la Hipótesis 3 *–los grafos son apropiados como técnicas de visualización de explicaciones y, por lo tanto, influyen en la aceptación de la recomendación por parte del usuario*. Como resultado, se propone un método de visualización interactiva basada

en grafos, incluido dentro de un sistema de explicación que hemos denominado UPIEM (*User-centric and Personalized Interactive Explanation Method*). Al centrarnos en la visualización, proponemos un sistema con características diferentes a los presentados anteriormente, que nos permite explorar todo el potencial de los grafos. UPIEM es un modelo de explicaciones global *ante hoc* específico al sistema recomendador, que no explica el proceso de recomendación a los usuarios sino que explica las causas de las recomendaciones. Es un modelo global, ya que utiliza todo el conocimiento disponible, siendo este conocimiento otorgado por el sistema recomendador (*ante hoc*). Las características más novedosas del sistema propuesto son la forma de visualizar las explicaciones y la interactividad ofrecida por el sistema, aspectos que influyen en la comprensión y aceptación de las recomendaciones. UPIEM utiliza un método de visualización que usa grafos para mostrar explicaciones basadas en ejemplos y la relación entre los productos que el usuario previamente ha valorado y los productos obtenidos por el algoritmo de recomendación.

Además, el sistema propone un proceso de explicación interactivo y conversacional donde el usuario define lo que necesita de las explicaciones en base a la interacción que realiza con UPIEM. Como consecuencia, los usuarios pueden seleccionar las causas que son importantes para entender la recomendación. La hipótesis principal de esta propuesta es que incluir al usuario en el diseño de la propia explicación puede ayudar a que sea más útil. De esta forma, UPIEM genera las explicaciones como un proceso cíclico donde las explicaciones ayudan al usuario a aceptar o rechazar una recomendación, mientras que el usuario, con sus interacciones, ayuda a personalizar y mejorar las razones que justifican la recomendación. Por ejemplo, en el dominio del cine el sistema puede explicar una recomendación de una película de Disney porque al usuario le gustan estas películas; sin embargo, el usuario puede decirle al sistema que en este momento no le apetece ver una película de Disney. En consecuencia, el sistema cambia la explicación e incluso puede recomendarle otra película al usuario. Por tanto, las explicaciones afectan al comportamiento de los usuarios hacia el sistema y a las recomendaciones proporcionadas (Cardoso et al., 2019). A su vez, la situación del usuario, llamado modelo mental en algunos trabajos (Ngo et al., 2020; Kulesza et al., 2012; Zenebe et al., 2005), afecta a la forma con la que ese usuario interactúa con las explicaciones.

El funcionamiento de UPIEM está descrito en la Sección 6.1, donde definimos las tres etapas de diseño del sistema. La Sección 6.2 detalla el método de visualización basado en grafos propuesto en esta tesis doctoral. En la Sección 6.3 determinamos cómo hemos efectuado la evaluación, el conjunto de datos elegido (Sección 6.3.1), la metodología de evaluación (Sección 6.3.2) y el análisis de los resultados (Sección 6.3.3). Al final del capítulo, en la Sección 6.4, se comentan las conclusiones del trabajo realizado.

6.1. El sistema de explicación UPIEM

UPIEM es un sistema de explicación para recomendadores basados en factorización de matrices (MF). La factorización de matrices es un tipo de algoritmo que ofrece buen rendimiento a la hora de generar recomendaciones, pero que a su vez es una caja negra para los usuarios (Aggarwal, 2016). De esta forma, los usuarios de este tipo de sistemas de recomendación pueden no encontrar útiles sus recomendaciones al no confiar en su funcionamiento. UPIEM aprovecha el conocimiento extraído de la factorización de matrices (es un sistema específico al modelo o *model-specific*), usando *case based reasoning* (CBR) y *formal concept analysis* (FCA), para generar el contenido de las explicaciones. La visualización basada en grafos es el elemento clave del sistema porque permite mostrar al usuario esta información de forma interactiva y conversacional, incrementando la aceptación de la recomendación.

UPIEM se ha desarrollado conforme al modelo de diseño presentado por Miller (2019) para afrontar la XAI desde la perspectiva de las ciencias sociales. Miller argumenta que la IA explicable puede beneficiarse de los modelos existentes sobre cómo las personas definen, generan, seleccionan, presentan y evalúan las explicaciones. Según Miller, el proceso de diseño de una explicación consta de tres pasos principales. El primer paso es el *proceso cognitivo*, donde se define una explicación para un evento dado, llamado *explanandum*, identificándose algunas causas del evento (o *explanans*). En el siguiente paso, se crea la explicación propiamente dicha (o el *producto*). Por último, se produce el *proceso social* que consiste en la transferencia de conocimiento del sistema que explica el *explanandum* hacia el usuario que recibe la explicación.

En trabajos anteriores (Jorro-Aragoneses et al., 2019; Diaz-Agudo et al., 2019; Jorro-Aragoneses et al., 2020a), presentamos varias contribuciones que abordan los primeros pasos del modelo de explicación aquí propuesto. La Figura 6.1 muestra una visión general del proceso de explicación llevado a cabo por UPIEM, ejemplificado para el dominio del cine. Partimos de una recomendación obtenida con MF. El algoritmo de MF recomienda un elemento (el *explanandum*, al cual denotamos como *rec*) a un usuario objetivo a partir de las valoraciones que ha realizado anteriormente. MF se basa en los factores latentes que de alguna manera caracterizan tanto a los productos como a los usuarios como vectores numéricos. En la figura, podemos ver los pasos que realiza UPIEM para generar una explicación para *rec*, y que explicamos a continuación:

Proceso cognitivo usando CBR. Aprovechamos los factores latentes para encontrar los *explanans* del producto recomendado por el sistema de caja negra. Siguiendo el esquema CBR descrito en nuestro trabajo anterior (Jorro-Aragoneses et al., 2019), construimos una base de casos con los factores latentes, capturando las preferencias del usuario, y

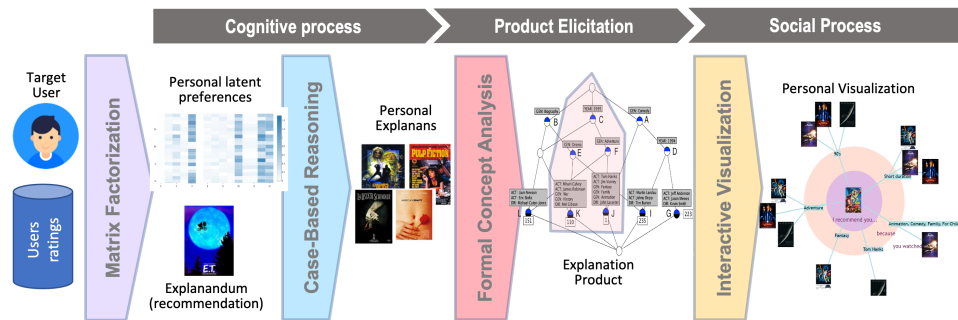
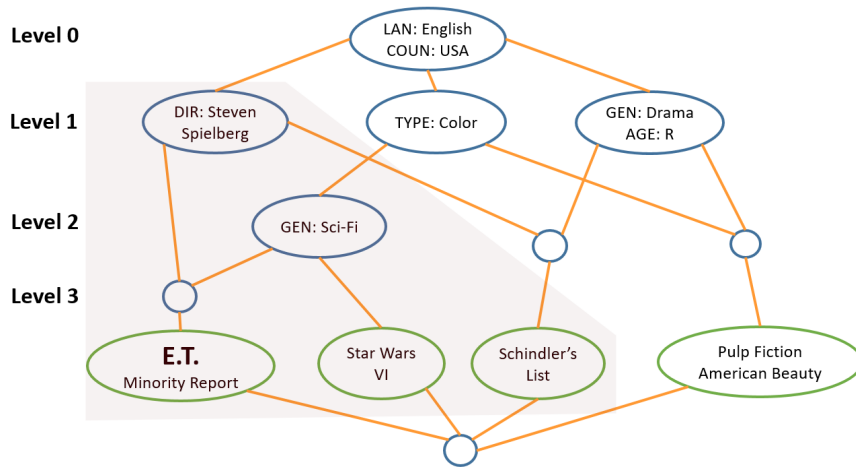


Figura 6.1: Vista general del funcionamiento del sistema de explicación UPIEM.

aplicamos una medida de similitud para obtener un conjunto de ejemplos de explicación, al cual denotamos EI (*explanation items*). Este conjunto EI constituye el conjunto personalizado de *explanans* para el usuario objetivo, ya que explica las causas de la recomendación para ese usuario concreto. Por tanto, podemos decir que UPIEM es un sistema *ante hoc* y específico al modelo, ya que necesita la información del recomendador para generar las explicaciones.

Obtención del producto de la explicación usando FCA. A simple vista, los *explanans* pueden no ser una causa que justifique la recomendación de forma obvia para los usuarios. Por lo tanto, el producto final de la explicación tiene que incluir información adicional sobre las relaciones entre el *explanandum* y los *explanans*. En nuestra propuesta, estas relaciones se construyen a partir de los atributos que tienen en común el producto recomendado y los *explanans*. Esta tarea se resuelve mediante FCA, un método que permite formalizar los conceptos de manera que se puedan analizar las relaciones que existen entre los conceptos representados. FCA construye un retículo conceptual que relaciona objetos y atributos comunes entre los objetos (Birkhoff, 1940). Es importante señalar que en UPIEM este retículo es único para el usuario objetivo de la explicación, ya que se construye utilizando la recomendación (*rec*) y los *explanans* (EI) obtenidos de forma personalizada. A continuación, podemos clasificar *rec* en el retículo para inferir los atributos compartidos por *rec* y EI . La Figura 6.2 muestra un ejemplo de retículo creado sobre el dominio del cine, donde “E.T.” es la película recomendada *rec*, y el resto de películas pertenecen al conjunto de *explanans* EI . En esa misma figura, debajo del retículo, se muestra una tabla con las posibles explicaciones de salida en función de cómo recorramos el retículo. La parte sombreada del retículo es la que contiene el producto de explicación. Esta decisión se tomó teniendo en cuenta las conclusiones que obtuvimos sobre la mejor manera de



*“The system recommends you **E.T.** because...”*

Level 0	<i>“...shares LAN:English, COUN:USA with Star Wars VI, Schindler’s List, Pulp Fiction, American Beauty, and Minority Report.”</i>
Level 1	<i>“...shares DIR:Steven Spielberg with Minority Report and Schindler’s List”</i> <i>“...shares TYPE:Color with Minority Report, Star Wars VI, Pulp Fiction, and American Beauty”</i>
Level 2	<i>“...shares GEN:Sci-Fi with Minority Report, Star Wars VI”</i>
Level 3	<i>“...shares DIR:Steven Spielberg, GEN:Sci-Fi with Minority Report”</i>

Figura 6.2: Ejemplo de retículo construido con $rec = \{E.T.\}$ y $EI = \{Minority Report, Star Wars VI, Schindler's list, Pulp Fiction, American Beauty\}$ para obtener sus atributos compartidos. La tabla de abajo muestra las explicaciones extraídas del retículo. Los elementos en la zona sombreada representan el producto final de la explicación.

recorrer el retículo para obtener el producto de explicación más útil para el usuario objetivo (niveles 2 y 3 de la tabla en la Figura 6.2) (Díaz-Agudo et al., 2019; Jorro-Aragoneses et al., 2020a).

Proceso social: visualización interactiva basada en grafos. En este último paso, se propone un método interactivo para visualizar el *producto* de explicación con el fin de mejorar la comprensión de la recomendación, la eficacia del sistema de explicación, y la satisfacción en el proceso. Esta visualización muestra, mediante grafos, la recomendación y la explicación, obtenida mediante FCA, al usuario objetivo. Los nodos del grafo van a representar *rec* y los ejemplos de explicación *EI*, mientras que los enlaces van a representar que existe una relación entre *rec* y *EI* identificada por los atributos inferidos en el retículo. De esta forma, el usuario puede ver cuáles son las propiedades en común que poseen y, en consecuencia, por qué *rec* es un elemento adecuado

como recomendación. Además, la interfaz permite que el usuario converse con el sistema a través de acciones interactivas para ajustar la explicación a su modelo mental. Para ello, el usuario puede eliminar aquellos atributos que no le parezcan adecuados como explicación. Como respuesta, UPIEM puede indicarle al usuario otra recomendación y explicación que puede serle de mayor utilidad. La interfaz también incluye funcionalidades adicionales como deshacer acciones. Este módulo de visualización corresponde a la comprobación de la Hipótesis 3 y al cumplimiento del Objetivo 3 de esta tesis doctoral, por lo que en la siguiente sección se detalla en profundidad.

6.2. Visualización interactiva basada en grafos

Como parte de su revisión de la literatura sobre XAI, Miller (2019) destacó que las explicaciones son sociales, constituyen un proceso de transferencia de conocimiento donde la información sobre la causa del suceso que se está explicando juega un papel importante. Por tanto, hay que realizar esta transferencia de conocimiento de forma adecuada para que el usuario sea capaz de entender las causas que se le están exponiendo. Por esta razón, pensamos que mostrar todo el retículo de conceptos, obtenido de la aplicación de FCA, al usuario objetivo podría resultar confuso, especialmente si el retículo que se forma es grande y representa mucha información. En esta tesis doctoral se propone un nuevo método de visualización de explicaciones, con el que pretendemos mostrar la información contenida en el retículo de forma intuitiva usando grafos. Este método de visualización se fundamenta en otros hallazgos del campo de las ciencias sociales que, según Miller, la mayoría de los investigadores y profesionales de la inteligencia artificial desconocen actualmente:

1. *Las explicaciones deben presentarse como parte de una conversación o interacción.* En consecuencia, nuestro método de visualización debe estar diseñado para ser interactivo y conversacional, de forma que el usuario pueda indicarle al sistema si una recomendación no le interesa lo suficiente para que, en consecuencia, el sistema pueda reaccionar proporcionándole nuevas recomendaciones que el usuario pueda entender mejor.
2. *Sólo un pequeño conjunto de causas de la explicación va a resultar realmente útil a los usuarios objetivo, ya que los usuarios rara vez, o nunca, esperan una explicación completa de un evento.* Por lo tanto, aprovechando la interactividad de nuestra estrategia de visualización, el sistema propuesto debe dejar que los usuarios, dependiendo de su contexto y sus preferencias actuales, seleccionen ese conjunto específico de razones que les ayuden a interpretar la recomendación. De esta

forma, el usuario puede adaptar la explicación a su propio modelo mental mediante la selección de los atributos que mejor relacionan el elemento recomendado y los *explanans*.

3. *Los valores numéricos y estadísticos calculados para generar el conjunto de causas de un evento no son útiles ni interpretables para los usuarios objetivo de la explicación.* Por lo tanto, nuestra visualización no debe incluir ningún valor de similitud entre *rec* y los *explanans*. Esta es otra razón que justifica el uso de FCA, ya que permiten obtener relaciones interpretables entre *rec* y *EI*.

La utilización de grafos para implementar el proceso social permite cumplir los prerequisites anteriores: interactividad y explicabilidad. Gracias al grafo, los usuarios pueden observar las conexiones existentes entre una recomendación y sus correspondientes *explanans* a través de sus atributos compartidos, conservando todo el conocimiento proporcionado por el retículo de una forma intuitiva. Las explicaciones se representan mediante un grafo que es una simplificación del grafo real que se podría crear. Generamos esta simplificación con el objetivo de ayudar a mejorar la interpretabilidad de la explicación. En la Sección 6.2.2 detallamos las razones detrás de esta simplificación. Un ejemplo ilustrativo de nuestra propuesta de visualización para una recomendación *rec* se presenta en la Figura 6.3. Las explicaciones se muestran a través de un grafo con tres áreas circulares concéntricas. De dentro a fuera, se muestra el elemento recomendado, los atributos compartidos y los ejemplos de explicación según la subestructura seleccionada del retículo. Los ejemplos de explicación estarán agrupados por los atributos que tengan en común con el elemento recomendado. Por ejemplo, podemos ver en la Figura 6.3 que el atributo “Tom Hanks” sólo enlaza con un ejemplo de explicación porque es la única película que comparte ese atributo con la película recomendada. Sin embargo, el atributo “Adventure” enlaza con un grupo de tres películas, ya que esas tres son del género “Adventure”, el mismo que tiene la película recomendada. Por otro lado, si el elemento recomendado y un grupo de productos comparten varios atributos, todos esos atributos estarán representados en el mismo nodo del grafo de visualización. Por ejemplo, se puede observar en la Figura 6.3 que los atributos “Animation”, “Comedy”, “Family”, “For Children” son atributos que pertenecen a un único nodo.

Una característica clave de nuestro método de visualización es la interacción conversacional que permite al usuario adaptar la explicación a su propio modelo mental. De esta forma, el usuario puede entender por qué la recomendación es adecuada (o no), a la vez que el sistema determina cuáles son las recomendaciones que más le pueden interesar en base a su modelo mental. Por ello, en la interfaz que proponemos, el usuario va a disponer de k recomendaciones distintas, con cuyas explicaciones puede interactuar para explorar cuál de las recomendaciones se adapta mejor a sus necesidades

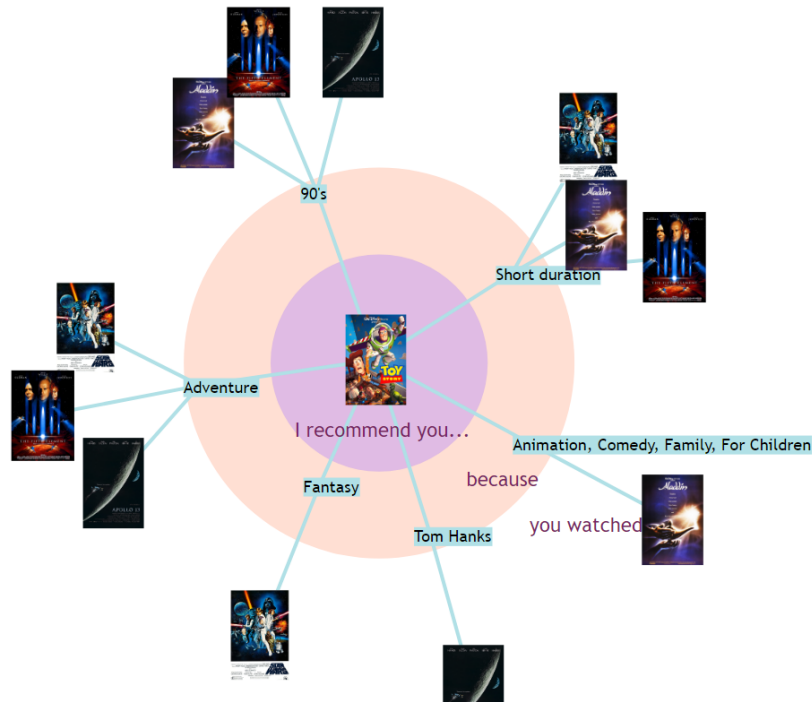


Figura 6.3: Ejemplo del método de visualización propuesto (de dentro hacia fuera: el elemento recomendado, los atributos compartidos y los ejemplos de explicación).

teniendo en cuenta la explicación otorgada para cada recomendación. Para ello, el sistema de explicación recibe k recomendaciones para un usuario objetivo, que se muestran en la esquina superior izquierda (Figura 6.4). Para cada una de estas recomendaciones, se va a visualizar su explicación teniendo en cuenta el proceso descrito hasta el momento con el módulo CBR, el módulo FCA y la visualización basada en grafos. El usuario puede cambiar entre las recomendaciones e interactuar con sus grafos de explicación correspondientes, pero el sistema siempre destaca aquella recomendación que es la más explicable. En nuestro sistema, definimos una recomendación como la más explicable cuando su grafo de explicación correspondiente es el que más conocimiento aporta al usuario objetivo, es decir, es el grafo que tiene el mayor número de enlaces entre el nodo del elemento recomendado y los nodos atributo. Por ejemplo, vemos en la Figura 6.4 que, aunque el usuario está observando la recomendación de la película “En busca del arca perdida” el sistema destaca la recomendación de la película “Toy Story” porque es la más explicable. En la Figura 6.3, vemos que “Toy Story” tiene 6 enlaces que conectan la película recomendada con nodos atributo, mientras que en la Figura 6.4, “En busca del arca perdida” tiene 4 enlaces de este tipo.

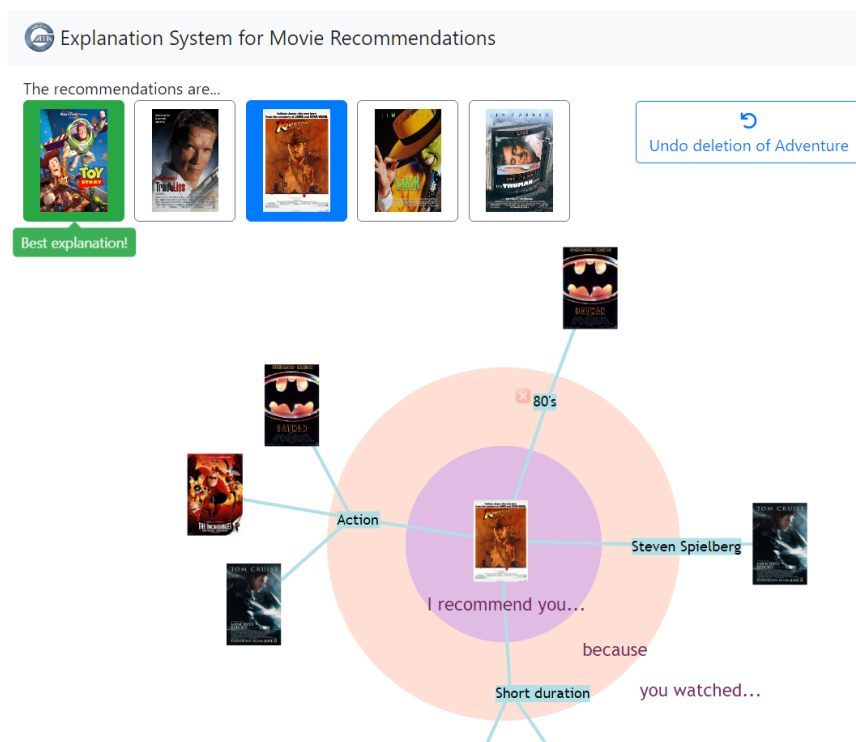


Figura 6.4: Interfaz interactiva para cambiar de recomendación.

Para adaptar la explicación al modelo mental del usuario, éste puede eliminar los atributos que, desde su punto de vista, no son relevantes para la explicación. Este atributo, al no ser relevante para el usuario en el contexto en el que se encuentra, se eliminará también del resto de recomendaciones, por lo que no aparecerá en ninguna de las explicaciones que se le muestran al usuario. El sistema de explicación también incluye una funcionalidad para deshacer esta acción si el usuario no está satisfecho con la explicación visual resultante. Si el nodo contiene varios atributos, el sistema permite al usuario seleccionar qué atributos quiere eliminar (véase la Figura 6.5). Si el usuario elimina demasiados atributos y reduce la explicabilidad de esta recomendación, el sistema sugerirá automáticamente una recomendación diferente como “mejor explicación”.

El sistema de explicación también incluye funcionalidades para que el usuario pueda explorar los detalles de la explicación. Los usuarios pueden mover los nodos del grafo para distribuirlo como mejor les parezca. Además pueden obtener una descripción detallada de cada elemento mediante acciones de *hover* y *zoom* del ratón. Una demostración online del sistema de explicación interactivo basado en grafos está disponible online¹.

¹<https://ucm-gaia.github.io/Graph-based-explanation/es>

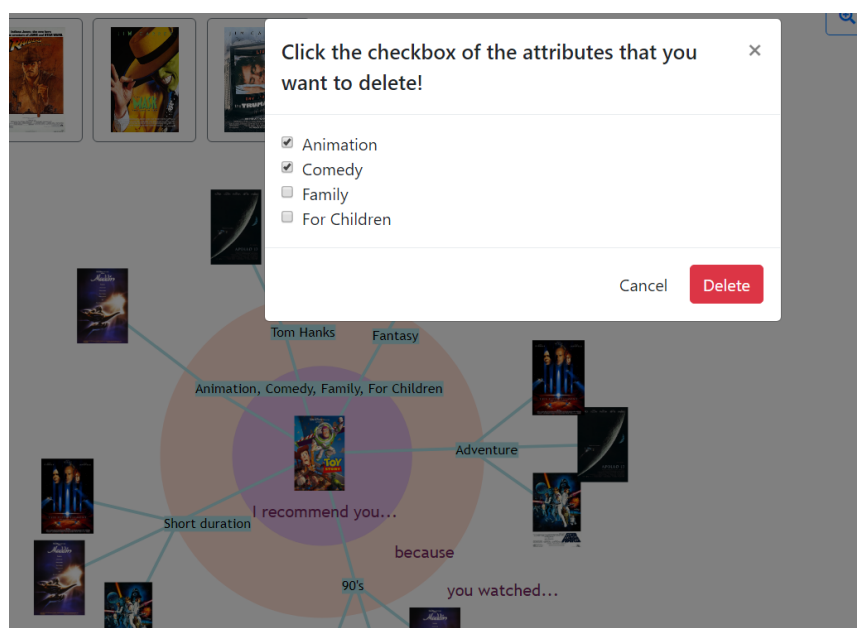


Figura 6.5: Método de eliminación de atributos de un nodo con varios atributos compartidos.

6.2.1. Comparativa con la visualización directa del grafo

Teniendo en cuenta cómo hemos utilizado el conocimiento extraído del módulo FCA para mostrar la explicación, queremos hacer una comparación con una visualización en la que no utilicemos este conocimiento, para discutir cuáles podrían ser las ventajas de nuestra propuesta. En la Figura 6.6 mostramos un ejemplo de cómo sería la representación de la explicación mostrada en la Figura 6.3 pero a través de una visualización basada en grafos que no utiliza el conocimiento extraído del módulo FCA. En la Figura 6.6 los nodos del grafo representan las películas que intervienen en la explicación (tanto la película recomendada como los casos de explicación), mientras que los enlaces representan los atributos en común que tienen esas películas. Los grosores de los enlaces son diferentes dependiendo de la cantidad de atributos que compartan las películas conectadas: cuantos más atributos compartan las películas, mayor será el grosor del enlace. Además, en los enlaces existe una gran cantidad de etiquetas que describen los atributos en común. Esta representación tiene algo de semejanza con respecto a cómo representamos la información en los grafos de interacción propuestos en los Capítulos 4 y 5, especialmente con el grafo de productos. De igual forma, en los grafos de interacción basados en productos y en la visualización de la Figura 6.6 tenemos los nodos representando los productos. Los enlaces se utilizan como la representación del nexo de unión entre los nodos del grafo, aunque en el caso de los grafos de interacción este nexo es el conjunto de usuarios que han

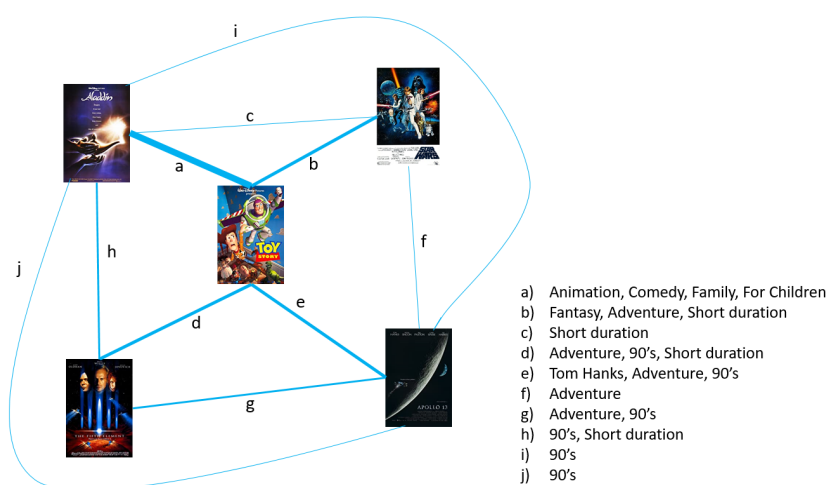


Figura 6.6: Ejemplo de cómo se mostraría la explicación mostrada en la Figura 6.3 con una visualización basada en grafos más tradicional. Los enlaces representan la relación entre las dos películas enlazadas. El grosor del enlace indica el número de atributos que comparten las películas. En la etiqueta del enlace se pueden leer estos atributos.

interactuado con los productos enlazados, mientras que en la visualización de la Figura 6.6 tenemos un enlace cuando los productos tienen atributos en común. Además, en los grafos de interacción, cuanto mayor era el número de usuarios en común que habían interactuado con dos productos conectados, mayor era el peso del enlace. De igual forma, aquí tenemos que cuanto mayor es el número de atributos que tienen en común ambas películas, mayor es el grosor del enlace. Con esta representación, la visualización podría resultar abrumadora para los usuarios debido a la gran información contenida en los enlaces. El módulo FCA ofrece una simplificación de la información distribuida en los enlaces, de forma que se puedan ver claramente cuáles son los atributos del elemento recomendado que intervienen en la explicación y de manera más sencilla.

6.2.2. Definición del sistema de explicación propuesto según el modelo conceptual

Podemos hacer también una clasificación de UPIEM con el modelo conceptual definido en el Capítulo 2. Teniendo en cuenta la **Motivación** del sistema, se puede definir el tipo de **Usuario** como **Usuario objetivo**, ya que el usuario que recibe la recomendación es el mismo que recibe la explicación. En cuanto a los **Objetivos** del sistema, los de **Bajo nivel** son los que indicamos a lo largo de la sección de evaluación: **Satisfacción**, **Confianza**, **Eficiencia** y **Escrutabilidad**. Estos objetivos ayudan a obte-

ner los objetivos de Alto nivel: **Justificar la recomendación** y **Mejorar la experiencia del usuario**. Con respecto al **Conocimiento**, UPIEM utiliza el conocimiento de la **Similitud entre productos** y el **Contenido** de estos productos. A lo largo de las secciones anteriores hemos comentado que en cuanto a **Encapsulamiento**, es un sistema **Ante hoc**, ya que para generar la explicación utiliza información del sistema recomendador. La **Técnica** para generar la explicación es la **Basada en conocimiento**, puesto que usamos CBR y FCA como métodos para obtener el producto final de la explicación. El concepto de **Presentación** es el más interesante a la hora de describir UPIEM. El **Formato** que utiliza es **Visual**, mientras que su **Argumentación** tiene una **Polaridad Positiva** y una **Comparación basada en similitudes**. En el caso de la **Interacción**, UPIEM es **Interactivo** y **Proactivo**.

6.3. Evaluación

Le evaluación de nuestra propuesta de visualización de explicaciones basada en grafos se ha realizado con usuarios reales utilizando un ejemplo de UPIEM centrado en el dominio del cine. En un repositorio de GitHub² se encuentra el código de implementación de este ejemplo. La evaluación se define en función de las siguientes hipótesis:

1. Hipótesis 3.1 (**H3.1**). El sistema de explicación basado en grafos propuesto en este capítulo resulta **satisfactorio** y **confiable** para los usuarios.
2. Hipótesis 3.2 (**H3.2**). La **visualización basada en grafos** y la **interacción conversacional** son interesantes y útiles para los usuarios.
3. Hipótesis 3.3 (**H3.3**). El método de explicación basado en grafos es **eficiente** y **escrutable**.
4. Hipótesis 3.4 (**H3.4**). Los **productos** de la explicación que proporciona nuestro sistema de visualización basada en grafos son útiles para los usuarios.

A continuación, detallamos los conjuntos de datos utilizados para construir el sistema de explicación basado en UPIEM que utilizamos en la evaluación (Sección 6.3.1). Posteriormente, describimos la metodología de evaluación, incluyendo las tareas realizadas por los participantes y los mecanismos empleados durante los experimentos (Sección 6.3.2). Por último, discutimos los resultados obtenidos (Sección 6.3.3).

²<https://github.com/UCM-GAIA/Graph-based-explanation/>

6.3.1. Conjunto de datos

Siguiendo nuestros trabajos anteriores, esta evaluación también se ha realizado en el dominio del cine, debido a su popularidad en el campo de los sistemas recomendadores. Los datos empleados por el recomendador y el sistema de explicación se extraen de los conjuntos de datos de MovieLens 100K e IMDb. Como vimos en el Capítulo 5, en la Tabla 5.1, el conjunto de datos de MovieLens incluye 100.000 valoraciones de 943 usuarios para 1.682 películas. Sin embargo, ya que necesitamos utilizar también el conjunto de datos de IMDb y en ese conjunto no se encuentran todas las películas almacenadas en el de MovieLens, tenemos que hacer un filtro, de forma que sólo las películas en el conjunto de datos de IMDb aparezcan en el de MovieLens. Por tanto, según la Tabla 5.1, nuestro conjunto de datos resultante D contendrá 11.477 interacciones realizadas por 587 usuarios sobre 164 películas. En esta evaluación, empleamos D para aplicar el algoritmo de recomendación basado en MF con el objetivo de (a) obtener el conjunto de $k = 5$ productos recomendados (*rec*, el *explanandum*); y (b) construir la base de casos para seleccionar el conjunto de ejemplos de explicación EI (los *explanans* para cada *rec*).

En el anterior capítulo, el conjunto de datos de IMDb era necesario para poder construir el sistema de explicación basado en contenido con el que comparábamos nuestros métodos de explicación basados en grafos. Sin embargo, en la evaluación de UPIEM necesitamos utilizar el conjunto de datos de IMDb para obtener los atributos de las películas y construir un retículo FCA para generar el *producto* de la explicación. De acuerdo con las conclusiones que obtuvimos de nuestro estudio de usuarios sobre cómo aprovechar el método FCA para generar explicaciones (Jorro-Aragoneses et al., 2020a), aplicamos una serie de filtros y cambios a este conjunto de datos para eliminar atributos comunes, recodificar variables, etc.

6.3.2. Metodología de evaluación

La metodología de evaluación propuesta para los experimentos con usuarios reales combina: (1) un análisis cuantitativo de los *logs* de las interacciones de los participantes con la aplicación y (2) un análisis cualitativo de las respuestas de los participantes a cuestionarios posteriores.

La primera parte de la evaluación se realiza a través de la *utilización de la aplicación* basada en grafos por parte de los usuarios. Tras un breve tutorial para aprender cómo funciona el sistema, los usuarios comienzan a interactuar con cinco propuestas de ejemplo. En cada una de las propuestas, los usuarios pueden observar $k = 5$ recomendaciones distintas para cada una de las cuales, el usuario puede observar la explicación en forma de grafo interactivo. Los usuarios pueden moverse entre esas cinco explicaciones, además de interactuar con la explicación, eliminando atributos, a través del proceso

Hipótesis	Parte I		Parte II	
	Logs	Feedback	PSSUQ	Test
H3.1	X		I1-I16	I17-I20
H3.2			I13-I15	I19,I20,I26
H3.3	X			I21-I23
H3.4		X		I24-I26

Tabla 6.1: Herramientas y preguntas del cuestionario utilizadas para evaluar las características de UPIEM según las hipótesis planteadas. La Parte I corresponde a la interacción con el sistema UPIEM. La Parte II es el cuestionario final.

conversacional descrito en la Sección 6.2 (véase la Figura 6.4, donde se muestra un ejemplo de propuesta de recomendaciones y explicaciones). Cuando el participante decide que está satisfecho (o no) con la explicación que está visualizando, entonces debe indicarlo al sistema mediante los botones que se muestran en la interfaz con ese objetivo. A continuación, el participante rellena un *formulario de feedback*. En este cuestionario, los usuarios seleccionan las características del sistema UPIEM que consideran útiles para esa propuesta, eligiendo entre los atributos, los ejemplos de explicación y/o su organización visual. Después, el usuario pasaría a ver la siguiente propuesta de ejemplo, continuando así con el proceso de forma sucesiva hasta terminar de valorar las cinco propuestas. En cualquier caso, el sistema registra las interacciones realizadas por los usuarios, como el cambio entre las películas recomendadas, la eliminación de atributos o la anulación de estas eliminaciones.

La segunda parte de la evaluación es un *cuestionario* estructurado en dos secciones. En la primera sección, los participantes rellenan un *Post-Study System Usability Questionnaire* (PSSUQ, versión 3) (Sauro y Lewis, 2016). El cuestionario PSSUQ es un test estandarizado para medir la usabilidad de un sistema percibida por el usuario. Este test define 16 preguntas para evaluar la **utilidad** del sistema (SYSUSE, preguntas 1-6), la **calidad de la información** (INFOQUAL, preguntas 7-12) y la **calidad de la interfaz** (INTERQUAL, preguntas 13-15). La segunda sección consta de 10 preguntas diseñadas y personalizadas en el contexto de este trabajo, relativas a la visualización basada en grafos y a las principales características de nuestro sistema de explicación (ver Tabla 6.3). Utilizamos una escala Likert de 7 puntos (de 1-muy de acuerdo a 7-muy en desacuerdo) siguiendo la estructura del cuestionario PSSUQ. El cuestionario termina con un campo de texto libre para incluir cualquier otro comentario o sugerencia sobre el sistema de explicación.

Todas estas herramientas de evaluación nos permiten validar nuestras hipótesis de la siguiente manera (ver Tabla 6.1):

1. El test PSSUQ (preguntas I1-I16) mide la **satisfacción** y **confianza** en el sistema. Este test se complementa mediante el análisis de los *logs* y con la segunda sección del cuestionario que se pide cumplimentar a los usuarios, ya que también contiene preguntas específicas para evaluar estas características (preguntas I17-I20).
2. Además, el test PSSUQ, específicamente la dimensión INTERQUAL, permite evaluar la **visualización** y el proceso de **interacción** conversacional. Esta evaluación se ve reforzada por varias preguntas específicas en la segunda sección del cuestionario (preguntas I19, I20 e I26).
3. Para medir la **eficiencia** y la **escrutabilidad** del sistema, analizamos el número y los tipos de interacciones que han llevado a cabo los usuarios con el sistema de explicación, así como el tiempo total empleado por los usuarios en estas interacciones. Esta información la obtenemos de los *logs* de interacciones. Esta evaluación se complementa cualitativamente con el análisis de varias preguntas de la segunda sección del cuestionario (preguntas I21-I23).
4. El análisis del *formulario de feedback* incluye también la evaluación de los *productos de explicación* en términos de **calidad** y **utilidad**. De nuevo, este análisis se complementa con otras preguntas de la segunda sección del cuestionario (preguntas I24-I26) y con el análisis de los *logs*.

Treinta y cinco usuarios participaron en la evaluación usando nuestra herramienta online³. No se han recogido datos de carácter personal o demográfico en el experimento, a excepción de un nombre o alias con el objetivo de emparejar los resultados obtenidos de la interacción con la herramienta y de las encuestas. A pesar de ello, conocemos su perfil debido a que los participantes son personas conocidas y de nuestro entorno. Doce de los treinta y cinco son mujeres (aproximadamente el 34%), mientras que el resto son hombres. La mayor parte de las personas que han realizado el experimento son personas jóvenes entre los 20 y los 40 años, siendo muchas de ellas personas que han estudiado o trabajan en la Facultad de Informática de la Universidad Complutense de Madrid. A pesar de ello, la mayoría no son expertos en sistemas de recomendación ni en sistemas de explicaciones. Además contamos con personas que no tienen relación con el mundo de la informática y que están vinculados con otros ámbitos como la psicología, trabajo social o la geología, por ejemplo. Los resultados recogidos se detallan en las siguientes secciones.

Dimensión	Preguntas	Media de UPIEM	SD de UPIEM	Media en Sauro y Lewis (2016)
PSSUQ	I1-16	2,12	1,19	2,82
SYSUSE	I1-6	2,20	1,22	2,80
INFOQUAL	I7-12	2,19	1,24	3,02
INTERQUAL	I13-15	1,92	1,10	2,49

Tabla 6.2: Evaluación con el cuestionario PSSUQ. Los valores más bajos indican una mayor facilidad de uso.

6.3.3. Análisis de los resultados

Como mostramos en la Tabla 6.2, nuestro sistema de explicación alcanza una puntuación media global de 2,12 en el test PSSUQ, siendo además el valor de la desviación típica de 1,19. Este valor podemos compararlo con las medias incluidas en el trabajo de Sauro y Lewis (2016), donde se analiza mediante el test PSSUQ un total de 21 sistemas. Podemos concluir que nuestros resultados en el cuestionario PSSUQ son claramente mejores que esas puntuaciones medias teniendo en cuenta que cuanto más baja es la puntuación, mejor es la percepción de los usuarios hacia el sistema. Esto significa que UPIEM parece conseguir una mayor aceptación que otros sistemas encontrados en la literatura, por lo que podría ofrecer una mejor experiencia de usuario. Podemos ver la comparación de las medias en la Tabla 6.2

Nuestras preguntas de evaluación incluidas en la segunda sección del cuestionario se muestran en la Tabla 6.3 junto con las puntuaciones medias resultantes (1,93 de media global) y las desviaciones típicas (1,04 de media global). Al igual que en PSSUQ, cuanto más bajas son las puntuaciones, mejor es la percepción de los usuarios con respecto al sistema. Como se puede observar, en esta sección del cuestionario también obtenemos resultados muy positivos.

A continuación, analizamos los resultados desde el punto de vista de cada hipótesis de evaluación.

6.3.3.1. (H3.1) Satisfacción y confianza

La primera hipótesis de nuestro experimento consiste en demostrar que la visualización basada en grafos fomenta la satisfacción y la confianza de los usuarios con las explicaciones proporcionadas por nuestro sistema. Ambas características se miden mediante el test PSSUQ y las preguntas I17-I20 del cuestionario final. En la Tabla 6.2 mostramos los resultados obtenidos por el test PSSUQ, que globalmente consigue una puntuación media de 2,12 y una desviación típica de 1,19. Estos resultados positivos se ven corroborados por las puntuaciones medias obtenidas en las preguntas I17-I20 con sus desvia-

³<https://ucm-gaia.github.io/Graph-based-explanation/es>

Pregunta	Enunciado	Media	SD
I17	<i>Creo que el sistema me ha ayudado a entender por qué se me ha recomendado una determinada película.</i>	1,97	0,92
I18	<i>He sentido confianza en las explicaciones que me estaba dando el sistema para cada recomendación.</i>	2,20	1,05
I19	<i>Estoy satisfecho/a con la visualización y la interactividad del sistema.</i>	1,80	0,90
I20	<i>Estoy satisfecho/a con las distintas opciones que ofrece el sistema (modificación de explicaciones, cambios de recomendación...)</i>	2,00	0,98
I21	<i>He llegado rápidamente a una recomendación y su explicación con la que me he sentido satisfecho/a.</i>	2,23	1,31
I22	<i>El sistema me ha permitido decirle que esa recomendación no me interesaba.</i>	1,59	0,86
I23	<i>El sistema me ha permitido decirle que esa explicación no me interesaba.</i>	1,70	1,03
I24	<i>Los atributos mostrados en las explicaciones me han resultado útiles para entender la recomendación.</i>	2,05	1,21
I25	<i>Las películas usadas como ejemplos en la explicación me han resultado útiles para entender por qué se me ha recomendado una película.</i>	1,94	1,06
I26	<i>Los agrupamientos de las películas a través de los atributos me han resultado útiles para entender la recomendación.</i>	1,83	1,12
Media global		1,93	1,04

Tabla 6.3: Preguntas de la segunda parte del cuestionario con sus correspondientes valores medios y desviaciones típicas para las 35 respuestas/participantes (1-muy de acuerdo a 7-muy en desacuerdo).

ciones típicas (Tabla 6.3). Teniendo en cuenta que las respuestas van de 1 (muy positivo) a 7 (totalmente negativo), podemos confirmar nuestra primera hipótesis y afirmar que los participantes comparten una opinión positiva respecto a su satisfacción y confianza en las explicaciones proporcionadas por UPIEM.

Esta hipótesis también se confirma al analizar la cantidad de usuarios que aceptan cada explicación, tal y como se muestra en la Figura 6.7. Los resultados muestran que los usuarios aceptaron la mayoría de las propuestas de UPIEM para explicar las recomendaciones después de interactuar con la interfaz de visualización. Esto refuerza la conclusión de que los usuarios están satisfechos con las explicaciones generadas.

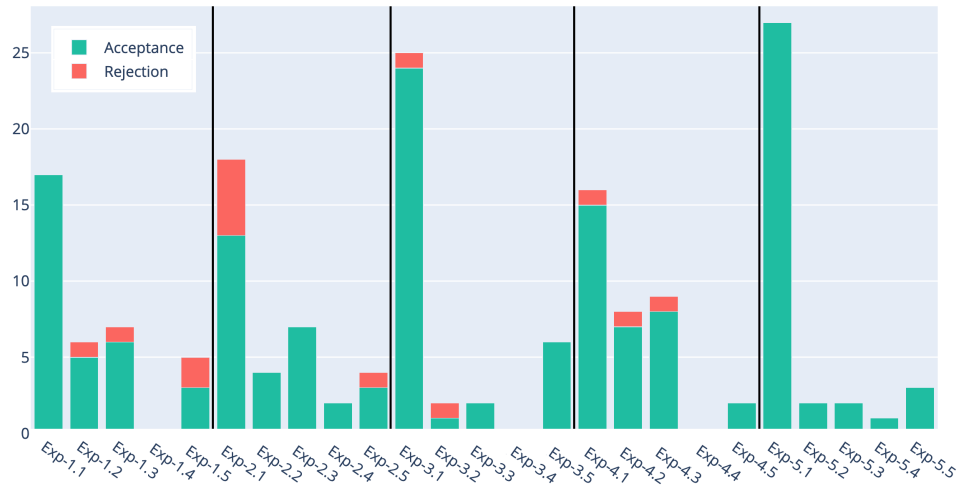


Figura 6.7: Aceptación de cada explicación por parte de los usuarios. Los participantes trabajaron con 5 propuestas, cada una de las cuáles tenía 5 recomendaciones distintas con su correspondiente explicación. Por lo tanto $Exp - x.y$ se refiere a la y -ésima explicación en la x -ésima propuesta.

6.3.3.2. (H3.2) Visualización e interactividad

La principal herramienta de validación de la visualización e interactividad de UPIEM es la dimensión INTERQUAL (calidad de la interfaz) del test PSSUQ (véase la Tabla 6.2). En esta dimensión, nuestro sistema propuesto obtiene una puntuación media de 1,92 y una desviación típica de 1,10. Se trata de la mejor puntuación entre todas las dimensiones validadas por el test PSSUQ, lo que sugiere que la calidad de la interfaz basada en grafos es la faceta más apreciada de nuestro sistema.

Sin embargo, el análisis de los comentarios de los participantes al final del cuestionario global indica que algunos de ellos esperaban características adicionales en el sistema de explicación. Por ejemplo, algunos participantes querían eliminar algunos ejemplos de explicación o querían atributos adicionales, como el escenario de la película, el argumento o los premios obtenidos. Algunos usuarios también comentaron que la interfaz podría mejorarse visualizando las similitudes entre los atributos y las películas. Este fue un comentario de una minoría de los usuarios, y está en desacuerdo con uno de los principios enunciados por Miller (2019) –“*Los valores numéricos y estadísticos calculados para generar el conjunto de causas de un evento no son útiles ni interpretables para los usuarios objetivo de la explicación*”– que, como se explica en la sección 6.2, se utilizó para diseñar el método de visualización. Esta es una cuestión que debe ser analizada en futuros trabajos tratando de identificar el perfil de los usuarios que sí requieren este tipo de información adicional (por ejemplo, usuarios con algún tipo de conocimiento

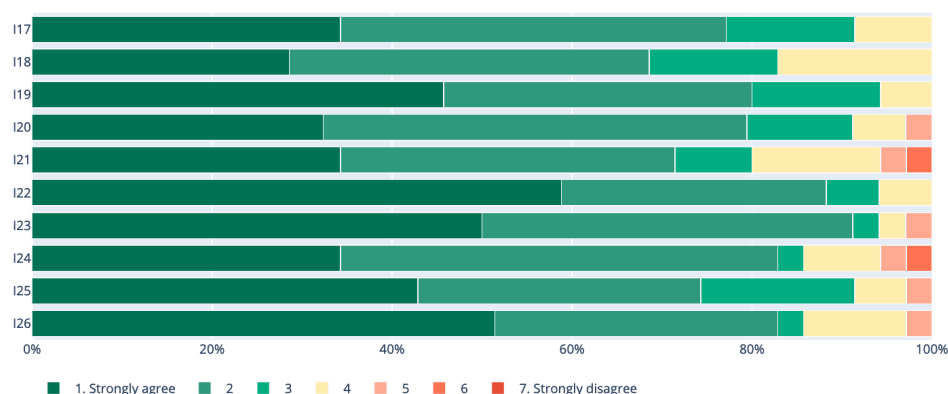


Figura 6.8: Respuestas a la segunda sección del cuestionario. Cada fila corresponde a una pregunta del cuestionario contemplado en la Tabla 6.3. Las respuestas a las preguntas van desde 1-muy de acuerdo hasta 7-muy en desacuerdo (los valores más bajos representan mejores resultados).

sobre recomendadores).

Además, la segunda sección del cuestionario global incluye algunas preguntas (I19, I20, I26) cuyo objetivo es medir la satisfacción con los procesos de visualización e interacción. Como podemos observar en la Figura 6.8, estas preguntas obtienen resultados muy positivos.

6.3.3.3. (H3.3) Eficiencia y scrutabilidad

El siguiente análisis de los resultados obtenidos en la evaluación se lleva a cabo para validar la eficiencia y la scrutabilidad de UPIEM. En primer lugar, se analizan dos variables cuantitativas registradas en los *logs* para estimar la eficiencia del sistema: (1) el tiempo empleado por el participante en interactuar con cada explicación (Figura 6.9a) y (2) el número de interacciones realizadas por cada usuario (Figura 6.9b). Observamos que el valor de ambas variables disminuye en las sucesivas propuestas mostradas al usuario, cuanto más han interactuado los usuarios con el sistema. Esto indica que la eficiencia aumenta progresivamente, ya que el tiempo necesario para entender la explicación disminuye gradualmente. También se puede extraer la misma conclusión de la Figura 6.10a, que muestra que las interacciones de cambio de explicación para un mismo ejemplo se utilizan con mayor frecuencia durante las interacciones iniciales, mientras los participantes aún se están familiarizando con el sistema. Además, en la última propuesta de recomendación, los participantes proporcionaron los resultados y comentarios más satisfactorios.

Los resultados mostrados en la Figura 6.10a también nos permiten evaluar la scrutabilidad de UPIEM. El significado de la acción “eliminar” (“re-

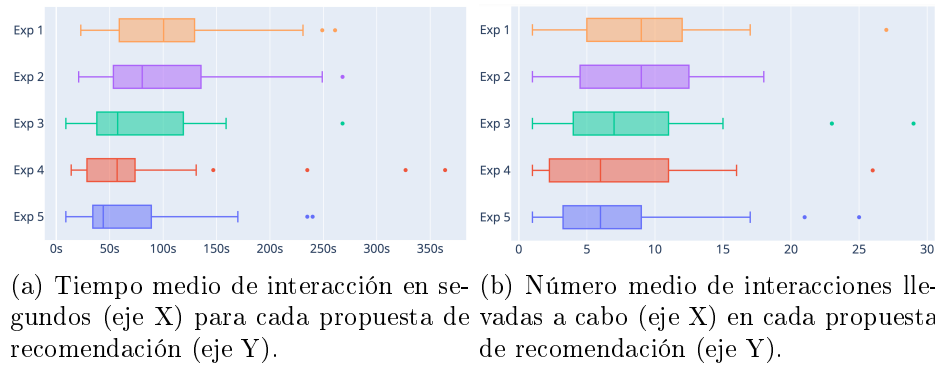


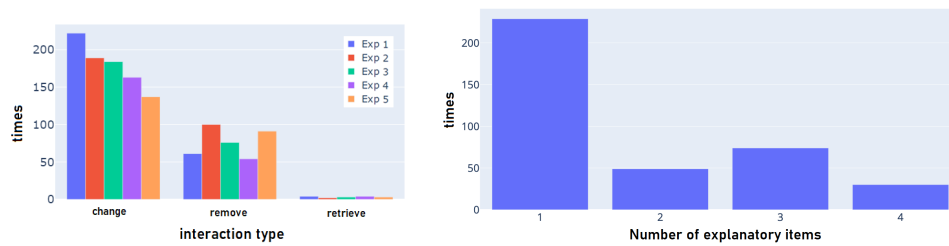
Figura 6.9: Análisis de la eficiencia de nuestro sistema de explicación.

move” en la figura) indica que un atributo no es interesante para el usuario. Podemos ver en esa figura que los participantes en la evaluación utilizan este tipo de interacción, incluso cuando ya se han familiarizado con el sistema. Esto implica que se usa con el objetivo claro de eliminar ese atributo y no para comprobar qué ocurre al eliminarlo. Por lo que podemos concluir que es una funcionalidad útil para los usuarios ya que permite personalizar la explicación de acuerdo a su modelo mental. Además, analizando el número de ejemplos de explicación vinculados a los atributos eliminados durante cada ejemplo de recomendación (Figura 6.10b), podemos concluir que los usuarios no consideran útiles los atributos vinculados con un solo ejemplo de explicación. Eliminar los enlaces que sólo conectan la recomendación a un único ejemplo de explicación podría ser un cambio a valorar en una versión futura de UPIEM. Por otro lado, la Figura 6.10a muestra que la funcionalidad “deshacer” (“*undo*” en la figura) se utiliza, aunque mínimamente. Esto puede indicar que los usuarios no suelen equivocarse al eliminar atributos, pero es una funcionalidad útil cuando comenten errores. Por lo tanto, concluimos que nuestro sistema es escrutable porque los usuarios pueden indicar al sistema por qué una recomendación o una explicación no les interesa y deshacer esta acción si es necesario.

Por último, también evaluamos la eficiencia y escrutabilidad del sistema en la segunda sección del cuestionario (véase la Figura 6.8, preguntas I21-I23). Estas preguntas ahondan directamente en ambas características, obteniendo algunos de los mejores resultados del cuestionario. Estos resultados apoyan nuestras conclusiones del análisis de los *logs* de interacción.

6.3.3.4. (H3.4) Productos de la explicación

La calidad de los productos de las explicaciones se evalúa mediante los formularios de *feedback* y las preguntas I24-I26 de la segunda sección del cuestionario. En primer lugar, analizamos los formularios de *feedback* recogi-



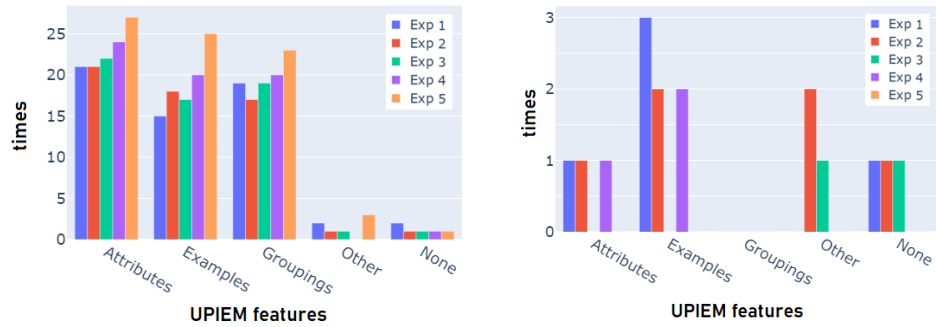
(a) Frecuencia (eje Y) de cada tipo de interacción (eje X) realizada por los participantes.

(b) Número de veces (eje Y) que se eliminaron los atributos según el número de ejemplos de explicación vinculados a ellos (eje X).

Figura 6.10: Análisis de la eficiencia y la escrutabilidad de nuestro sistema de explicación.

dos después de cada explicación. En estos cuestionarios preguntamos directamente por las características del producto de explicación que llevaron al usuario a aceptar o descartar una explicación. Las tres características principales estudiadas que pueden influir en la aceptación de la explicación por parte del usuario son obtenidas mediante FCA: los *atributos*, los *ejemplos* de explicación y las *agrupaciones* de los ejemplos. Como podemos observar en la Figura 6.11a, las tres características parecen ser igualmente importantes para los usuarios a la hora de valorar positivamente una explicación. Esta similitud nos lleva a concluir que la calidad de una explicación radica en la combinación de las tres características. Sin embargo, como podemos observar en la Figura 6.11b, no hay un patrón claro detrás de las características que llevaron a los usuarios a descartar una explicación, aunque los ejemplos elegidos en la explicación parecen ser algo más determinantes que el resto de las características.

Las preguntas I24-I26 del cuestionario también se diseñaron para evaluar la utilidad de estas tres características. Los resultados de la Figura 6.8, donde podemos ver los resultados positivos obtenidos en estas preguntas, validan las conclusiones de nuestro análisis anterior. Sin embargo, encontramos algunas diferencias si comparamos el análisis de esta figura con la Figura 6.11a, que muestra el análisis del cuestionario de *feedback*. Según esta última figura, podemos observar que los *atributos* parecen ser el componente más útil para los usuarios. En cambio, en la pregunta I24, en la cual preguntamos sobre la utilidad de los atributos de la explicación, se obtienen peores resultados que en las preguntas que hacen referencia a la utilidad de los ejemplos de explicación (pregunta I25) o a los agrupamientos de películas (I26). Una posible explicación de este comportamiento se encuentra en el análisis de los *logs*. Observamos que hay algunos tipos de atributos que los usuarios eliminaron en muchos casos, por ejemplo, la duración de la película, la década y los actores de las películas de animación. Por lo tanto, algunos tipos de atribu-



(a) Las razones por las que a los usuarios les gustaron las explicaciones.

(b) Las razones por las que a los usuarios no les gustaron las explicaciones.

Figura 6.11: Análisis del *producto de explicación* de UPIEM.

tos pueden resultar redundantes, a pesar de que expliquen adecuadamente la recomendación.

6.4. Conclusiones

En este capítulo se ha descrito un nuevo método de visualización basado en grafos que hemos propuesto en esta tesis doctoral para explicar recomendaciones de forma interactiva y conversacional. Es un método que pretende mejorar la interpretabilidad de los sistemas de recomendación, favoreciendo su aceptación, a través de la simplificación en la representación de la información. Con esta propuesta queremos investigar cómo de útiles son los grafos como método de visualización de explicaciones, de forma que se pueda comprobar el beneficio de este tipo de estructuras desde un punto de vista no contemplado en capítulos anteriores de esta tesis doctoral. Nuestra propuesta de visualización ha sido integrada en nuestro sistema de explicación UPIEM (*User-centric and Personalized Interactive Explanation System*), un sistema de apoyo a recomendadores basados en factorización de matrices, que usa CBR y FCA para crear el producto de la explicación. Con todo ello pretendemos abordar el Objetivo 3 y la Hipótesis 3 de la tesis, en el que proponíamos *el estudio de los grafos como herramienta de visualización de explicaciones para recomendadores*.

Para determinar la validez de nuestro método de visualización, llevamos a cabo una evaluación con usuarios reales. La hipótesis planteada en esta evaluación es que el mecanismo de explicación mediante visualización basada en grafos de UPIEM es eficiente, escrutable, fiable y satisfactorio para el usuario. Como resumen general de la evaluación, UPIEM obtiene altas puntuaciones tanto en el test PSSUQ como en el cuestionario de evaluación específico. Sus resultados, junto con los análisis de los formularios de *feedback*

y los *logs*, indican que los participantes están satisfechos con el producto de explicación, especialmente con las agrupaciones de ejemplos de explicación similares usando atributos. Además, las características de UPIEM mejor valoradas por los usuarios son el método de visualización e interacción de la interfaz. Los resultados de evaluación confirman que, según la opinión de los usuarios, el sistema de visualización interactivo basado en grafos de UPIEM hace que el sistema de explicación sea satisfactorio y atractivo.

Algunas de las etapas de diseño y construcción de UPIEM han sido publicadas en el congreso *International Conference on Case-Based Reasoning* de 2019 y 2020 (Jorro-Aragoneses et al., 2019; Diaz-Agudo et al., 2019; Jorro-Aragoneses et al., 2020a). El trabajo completo de desarrollo del método de visualización de grafos y de UPIEM, así como la evaluación descrita en este capítulo se encuentra pendiente de publicar en el momento de escritura de esta memoria (Caro-Martinez et al., Pendiente de aceptar).

Los resultados obtenidos confirman las hipótesis definidas y, por tanto, podemos concluir que los grafos como método de visualización pueden aportar una componente útil y satisfactoria a la hora de generar explicaciones para sistemas de recomendación.

Capítulo 7

Conclusiones y trabajo futuro

El desarrollo de la Web en las últimas décadas ha propiciado la realización de multitud de tareas online, como la compra de productos, la búsqueda de empleo, actividades sociales, o ver películas y series. Estas tareas suponen el acceso a una gran cantidad de información y a una amplia oferta de productos que consumir. Esto conlleva una gran ventaja puesto que los usuarios pueden llegar a encontrar productos que de otra forma les hubiera resultado imposible. Sin embargo, también puede entrañar un problema porque, ante tal oferta, los usuarios pueden no saber cuáles son los productos que más se ajustan a sus necesidades. Como solución a este problema, surgen los sistemas de recomendación, que tratan de ayudar a los usuarios a tomar mejores decisiones en cuanto a los productos que consumir. Sin embargo, los usuarios pueden desconfiar de estos sistemas si no los entienden. Esto supone un gran inconveniente, ya que si los usuarios no confían en el sistema de recomendación, es más probable que dejen de usarlo, reduciendo su utilidad y la experiencia del usuario con la plataforma. Como respuesta a este problema, los sistemas de explicación ayudan al usuario a entender cómo funciona el recomendador o cuáles son las razones por las que se ha recomendado un producto a un usuario.

Los sistemas de recomendación y explicación tradicionales utilizan el conocimiento de los *ratings* (los basados en filtrado colaborativo) y de las descripciones de los productos (basados en contenido). Sin embargo, hay situaciones en las que no existen estos recursos, o no se puede acceder a ellos. Este escenario de mínimo conocimiento aparece cuando los usuarios sólo interactúan con los productos, sin dar al sistema información explícita sobre sus preferencias. Si no hay descripciones de los productos y sin valoraciones otorgadas por los usuarios, no se pueden aplicar el filtrado colaborativo ni los algoritmos basados en contenido. En la literatura, observamos que las técnicas de filtrado colaborativo se pueden adaptar a este tipo de escenarios. Sin embargo, son técnicas que tratan de explotar el valor de los *ratings* para generar las recomendaciones y explicaciones y, por tanto, su rendimiento no

es el deseable.

En base a este problema, nuestra propuesta es que los grafos pueden ser una alternativa a los métodos de recomendación y explicación tradicionales ya que la construcción de los mismos puede requerir menor cantidad de conocimiento. Como consecuencia, hemos definido las siguientes hipótesis a explorar en esta tesis doctoral: (H1) las técnicas de análisis de grafos permiten generar recomendaciones explicables requiriendo una cantidad mínima de conocimiento; (H2) estas técnicas también permiten obtener explicaciones en escenarios de mínimo conocimiento para distintos tipos de sistemas de recomendación y; (H3) los grafos sirven, además, como técnica de visualización de explicaciones por lo que influyen en la aceptación de la recomendación por parte del usuario. Como consecuencia, el objetivo principal de esta tesis doctoral ha sido el estudio y desarrollo de sistemas de recomendación y explicación para recomendadores con grafos de interacción. Para alcanzarlo, se han definido cuatro objetivos en esta tesis doctoral: (Objetivo 0) estudiar los sistemas de recomendaciones y distintas técnicas de explicación aplicables; (Objetivo 1) diseñar, desarrollar y evaluar sistemas de recomendación basados en grafos de interacción en escenarios de mínimo conocimiento; (Objetivo 2) diseñar, desarrollar y evaluar sistemas de explicación basados en grafos de interacción en escenarios de mínimo conocimiento e independientes del tipo de sistema de recomendación; y (Objetivo 3) estudiar los grafos como método de visualización de explicaciones para recomendadores.

La primera etapa de la tesis ha consistido en la revisión de la literatura. En ella hemos repasado los principales conceptos sobre sistemas de recomendación, grafos y técnicas de predicción de enlaces. Sin embargo, el principal trabajo que se ha llevado a cabo durante esta revisión ha sido el estudio en profundidad sobre los sistemas de explicación, la Inteligencia Artificial Explicable y, especialmente, las explicaciones en sistemas de recomendación. Como resultado de este amplio estudio se ha desarrollado el modelo conceptual que se ha presentado como primer resultado de esta tesis doctoral y que tiene como objetivo ayudar al diseño de sistemas de explicación para recomendadores. Teniendo la información estructurada sobre este tipo de sistemas, los diseñadores pueden acudir a ella para conocer las características que este tipo de sistemas pueden tener. De esta forma, pueden comparar las características, y determinar cuáles de ellas se adaptan mejor a sus necesidades. Para ello, se ha formalizado el modelo conceptual dando lugar a la ontología ExRecOnto. Esta ontología pretende guiar la implementación de sistemas de explicación para recomendadores a través de un diseño guiado por objetivos. La evaluación del modelo a través de la clasificación de 51 propuestas de sistemas de explicación ha validado la utilidad del modelo conceptual propuesto. A pesar de ello, también puede tener debilidades, ya que, al ser un campo en constante cambio, puede hacer falta hacer revisiones y modificaciones del modelo en el futuro, como ya nos ha ocurrido durante

el trascurso de la tesis. Tras el trabajo realizado, se puede concluir que el Objetivo 0 se ha cumplido.

Tras esto, el trabajo se centró en el Objetivo 1 para probar la Hipótesis 1. Como resultado, en esta tesis doctoral hemos propuesto métodos de recomendación basados en grafos de interacción, los cuales representan la información sobre las interacciones entre los usuarios y los productos (por ejemplo, si un usuario ha visto una película, ha realizado un problema, ha leído un libro, etc.) mediante grafos, usándose estas estructuras junto a técnicas de predicción de enlaces para encontrar similitudes entre las entidades representadas. A través de esto, se pueden encontrar recomendaciones personalizadas para los usuarios. En la tesis, se ha pretendido determinar si los métodos propuestos pueden mejorar de alguna forma las técnicas de recomendación que se utilizan en la literatura para obtener recomendaciones en escenarios de mínimo conocimiento. Para ello, se ha llevado a cabo una evaluación donde se comparan las precisiones de los métodos basados en grafos y algunas de las técnicas de filtrado colaborativo que se pueden usar en este contexto. De los resultados podemos decir que, en general, los sistemas de recomendación basados en grafos de interacción propuestos en la tesis obtienen recomendaciones de igual o incluso mayor precisión que el filtrado colaborativo, pero utilizando menor cantidad de conocimiento. A su vez, como discutimos a lo largo de la tesis y partiendo de los resultados encontrados en la literatura, los sistemas de recomendación basados en grafos son más interpretables que los tradicionales. La razón principal es que son métodos locales, que utilizan información sólo de una parte específica del conocimiento, lo cual ayuda a su interpretación. En definitiva, podemos concluir que los sistemas de recomendación basados en grafos suponen una alternativa más interpretable a las técnicas tradicionales, pudiendo ser aplicada en escenarios de mínimo conocimiento. Por tanto, hemos cumplido el Objetivo 1, comprobando la Hipótesis 1.

De forma análoga al trabajo anterior, para cumplir con el Objetivo 2 y validar la Hipótesis 2 de la tesis, hemos diseñado y evaluado una propuesta para generar explicaciones basadas en ejemplos para recomendaciones a través de grafos de interacción y técnicas de predicción de enlaces. Para ello, hemos realizado una comparación del modelo de explicación propuesto con métodos de explicación tradicionales. Tras los resultados obtenidos, hemos podido concluir que el sistema de explicación basado en grafos obtiene mejores ejemplos de explicación. Con esto, hemos cumplido el Objetivo 2 de la tesis, concluyendo que los grafos de interacción con las técnicas de predicción de enlaces son una alternativa viable y efectiva para ofrecer explicaciones a recomendadores de caja negra en escenarios de mínimo conocimiento, cuando no tenemos información sobre *ratings* o descripciones de los productos.

Por último, hemos analizado la visualización de explicaciones a través de grafos como definimos en la Hipótesis 3 y el Objetivo 3. Como resultado,

se ha propuesto un nuevo método de visualización interactiva para explicar recomendaciones mediante grafos. La visualización basada en grafos nos ha permitido relacionar entidades y características de forma innovadora e intuitiva, incluyendo además la posibilidad de crear una conversación entre el sistema y el usuario con el fin de que la explicación resultante se adecue al modelo mental del usuario. En este caso, decidimos evaluar el sistema con usuarios reales, ya que este tipo de evaluación permite obtener información sobre propiedades de la explicación como la confianza, la escrutabilidad, la satisfacción o la eficiencia. Además, los usuarios pudieron dar opiniones directas sobre el método de visualización que se ha propuesto en esta tesis. Los resultados de la evaluación nos han permitido concluir que los usuarios están satisfechos con las explicaciones, siendo la interfaz interactiva basada en grafos el componente mejor valorado del sistema de explicación.

7.1. Trabajo futuro

Aunque en este trabajo se han cumplido los objetivos definidos en la tesis y se han realizado aportaciones suficientes para comprobar las hipótesis planteadas, podemos definir algunas líneas de trabajo futuro para complementar los resultados obtenidos. A continuación explicamos el trabajo futuro de cada objetivo de la tesis.

Con respecto al **Objetivo 0**, se ha desarrollado, como resultado de la revisión de la literatura realizada, un modelo conceptual que define y guía el proceso de desarrollo de sistemas de explicación en recomendadores. El modelo ha sido validado clasificando 51 sistemas distintos encontrados en la literatura, resultando la validación exitosa, ya que se pudo clasificar cada uno de ellos con los distintos aspectos definidos en el modelo. Sin embargo, el modelo puede tener algunas limitaciones. Actualmente, la XAI es uno de los campos más en auge en investigación en informática y en ciencias de la computación. De ahí que esté en constante evolución y sea de esperar que aparezcan propuestas pioneras en un futuro. Puede que los nuevos sistemas que van a surgir contemplen aspectos innovadores no cubiertos por la versión actual del modelo conceptual. Sin embargo, el proceso de diseño iterativo seguido durante la creación del modelo puede aplicarse de nuevo para mejorarlo con los nuevos conceptos que puedan aparecer. También puede producirse que, aunque se haga referencia a los mismos conceptos definidos en nuestro modelo propuesto, se establezca una nomenclatura distinta para esos conceptos. Si esto ocurriese, se podría crear una nueva versión del modelo conceptual cambiando los nombres de los conceptos por aquellos que de forma generalizada están apareciendo en la literatura. Una de las líneas de futuro más interesante para este objetivo sería unir ExRecOnto con otras ontologías, por ejemplo con las desarrolladas por el grupo de investigación, especialmente con RecOnto, que modela sistemas de recomendación.

En cuanto al **Objetivo 1**, hemos desarrollado distintos métodos basados en grafos de interacción para generar recomendaciones interpretables en escenarios de mínimo conocimiento. Hemos evaluado estos métodos comparándolos con un conjunto de técnicas que se pueden aplicar en este tipo de escenarios. Como trabajo futuro, podemos seguir comparando nuestros métodos con más técnicas de aprendizaje automático, como la regresión, para acabar de corroborar la hipótesis planteada. También podemos evaluar nuestros sistemas de recomendación con usuarios reales, para obtener conclusiones más fiables del trabajo realizado. Este tipo de evaluaciones pueden proporcionar resultados más precisos en cuanto a la satisfacción y la experiencia del usuario. Por lo tanto, sería necesario evaluar nuestros métodos con usuarios reales para medir estas importantes características. Además, nos podría aportar conclusiones sobre la interpretabilidad de nuestros métodos. También podemos abordar como trabajo futuro el problema del *cold start*, o arranque en frío. Los sistemas de recomendación sufren de este problema cuando no tienen información suficiente sobre determinados usuarios o productos porque no ha habido interacciones en las que hayan intervenido. Nuestros métodos basados en grafos también experimentan este problema, por lo que podríamos investigar cómo podríamos solucionarlo. Otra línea de trabajo futuro sería el uso de otras técnicas de análisis de redes sociales como las de predicción de enlaces globales, que en términos de interpretabilidad son peores que las usadas en la tesis, pero podemos investigar cómo de satisfactorias son en términos de precisión. Además de esto, podríamos validar nuestra hipótesis con diferentes conjuntos de datos de otros dominios. Por ejemplo, creemos que los recomendadores en el dominio de la música pueden aprovechar nuestro modelo propuesto ya que sus conjuntos de datos suelen carecer de valoraciones.

En relación al **Objetivo 2**, hemos desarrollado métodos basados en grafos de interacción para obtener explicaciones, y los hemos comparado con un método basado en contenido, uno de las técnicas tradicionales para generar explicaciones. Partiendo de este trabajo, podemos determinar varias líneas de trabajo futuro, algunas de ellas comunes al trabajo futuro ya definido para las técnicas de recomendación basadas en grafos. Una de ellas sería contemplar distintas alternativas para intentar paliar el problema del *cold start* que pueden sufrir nuestros métodos. También aquí sería interesante aplicar los métodos basados en grafos para generar explicaciones en otros dominios que no sean el de las películas. Además, podríamos comparar nuestros métodos con otras técnicas clásicas, como el filtrado colaborativo, de forma similar a como hemos evaluado el Objetivo 1. Otra línea de investigación futura es aplicar los métodos de agregación que ya hemos propuesto en el objetivo anterior (en la Sección 4.1.4) a los modelos de explicación basados en grafos. También podríamos utilizar técnicas de predicción de enlaces globales y comparar sus resultados con los resultados de este trabajo. Otra línea esencial

como trabajo futuro, identificada también en el Objetivo 1, es llevar a cabo una evaluación con usuarios reales, ya que pueden proporcionar una opinión más precisa sobre la efectividad de los métodos basados en grafos. Además, pueden proporcionar un análisis sobre otras características de explicación más difícilmente evaluables, como la confianza del usuario o su satisfacción. Finalmente, podemos establecer como línea de trabajo futuro la creación de un *framework* que generalice los métodos basados en grafos aquí propuestos, tanto los métodos de recomendación como los de explicación.

Por último, como aportación del **Objetivo 3** de la tesis, hemos desarrollado un método de visualización interactiva basada en grafos y lo hemos evaluado con usuarios reales. A partir del trabajo y la evaluación realizada podemos establecer varias líneas de trabajo futuro. El análisis de los comentarios de los participantes al final del cuestionario global que rellenaron en la evaluación indica que algunos participantes esperaban características adicionales en el sistema de explicación. Por ejemplo, algunos participantes querían eliminar los casos de explicación o querían atributos adicionales, como el escenario, el argumento o los premios obtenidos por la película. Si incluimos más fuentes de recursos, se podrían añadir atributos adicionales interesantes para los usuarios. En la misma línea de lo anterior, hemos observado que hay algunos tipos de atributos que los usuarios eliminaron en muchas ocasiones, como la duración de la película, la década y los actores de las películas de animación. Por lo tanto, aunque estos atributos podían ser útiles para explicar la recomendación, resultaban repetitivos al encontrarse en muchas de las explicaciones mostradas a los usuarios. Para resolver esto, en trabajo futuro, tenemos que mejorar las descripciones de cada elemento para encontrar más atributos útiles y diversos para relacionar las recomendaciones con los casos de explicación. También pudimos observar en el análisis de los resultados que los usuarios eliminaban con bastante mayor frecuencia aquellos atributos que conectaban la recomendación con un único elemento de explicación. Como trabajo futuro podríamos eliminar estos atributos, ya que no parecen ser de utilidad para los usuarios.

Por otro lado, algunos usuarios también comentaron que la interfaz podría mejorarse mostrando los valores de similitud entre los atributos y las películas. Esta es una cuestión que debe analizarse en el futuro tratando de identificar el perfil de los usuarios que sí requieren este tipo de información adicional (por ejemplo, los usuarios con cierta experiencia en sistemas de recomendación). Por último, podemos establecer algunas líneas de trabajo futuro siguiendo las ideas que ya se han comentado en el resto de los objetivos, por ejemplo, aplicando la visualización basada en grafos sobre otros sistemas de explicación o haciendo una evaluación de comparación con otros tipos de interfaces de visualización con el objetivo de comprobar cómo mejora nuestra propuesta las características de lo propuesto en la literatura.

Apéndice A

Implementación de ExRecOnto

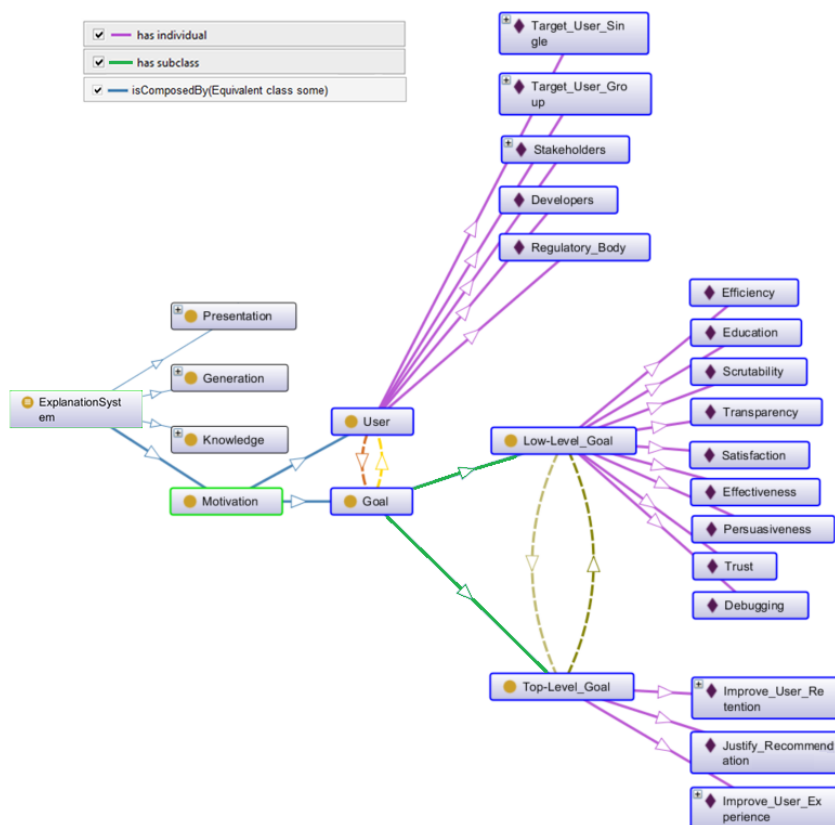


Figura A.1: Visualización parcial de la ontología ExRecOnto con respecto al concepto Motivación.

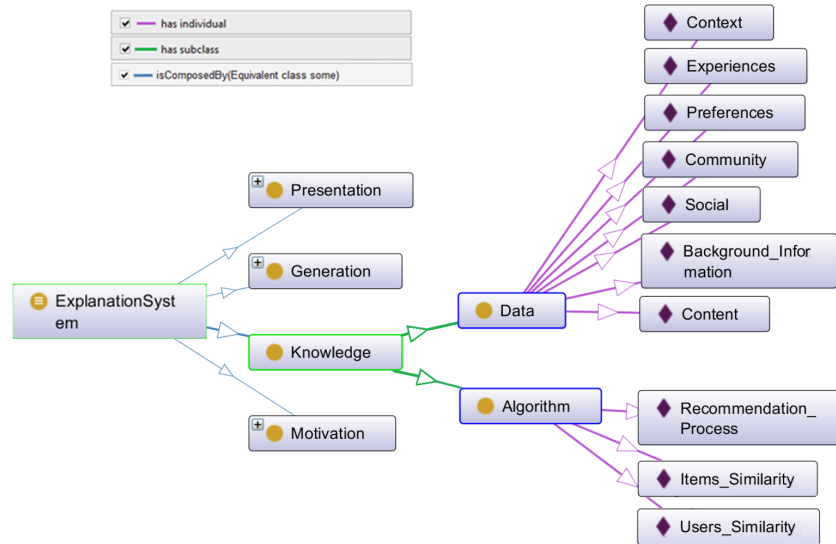


Figura A.2: Visualización parcial de la ontología ExRecOnto con respecto al concepto Conocimiento.

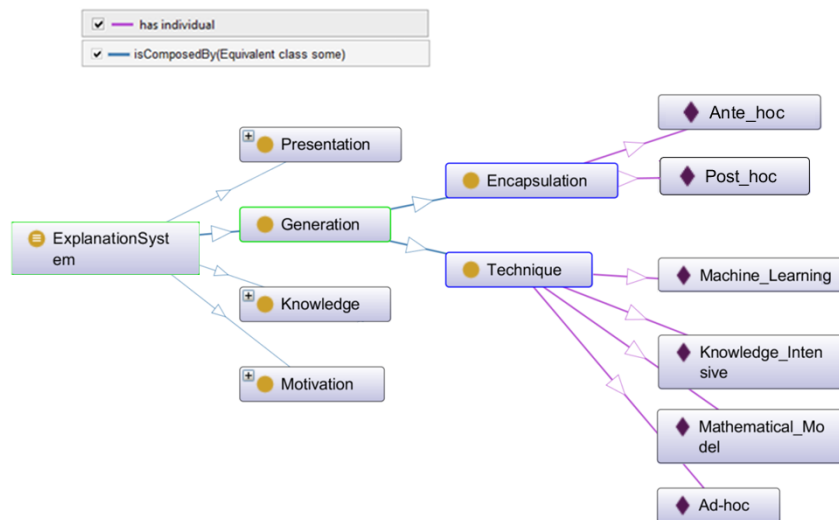


Figura A.3: Visualización parcial de la ontología ExRecOnto con respecto al concepto Generación.

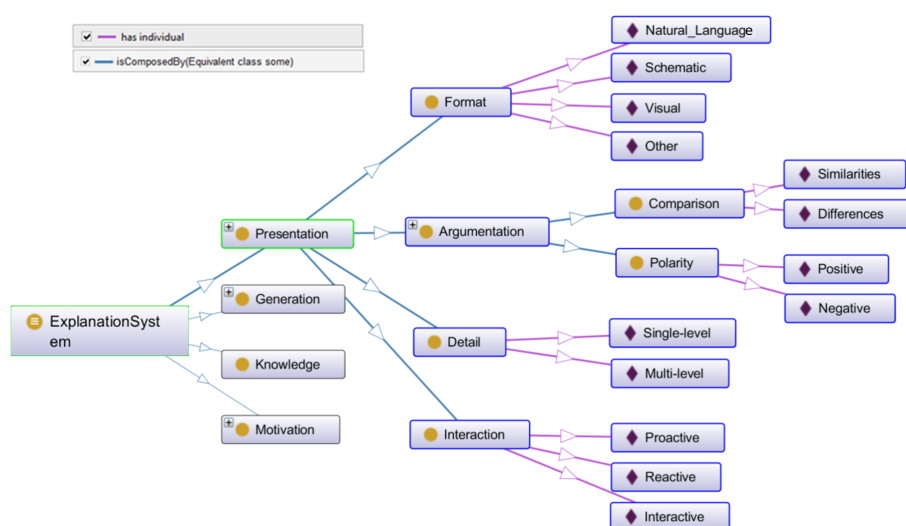


Figura A.4: Visualización parcial de la ontología ExRecOnto con respecto al concepto **Presentación**.

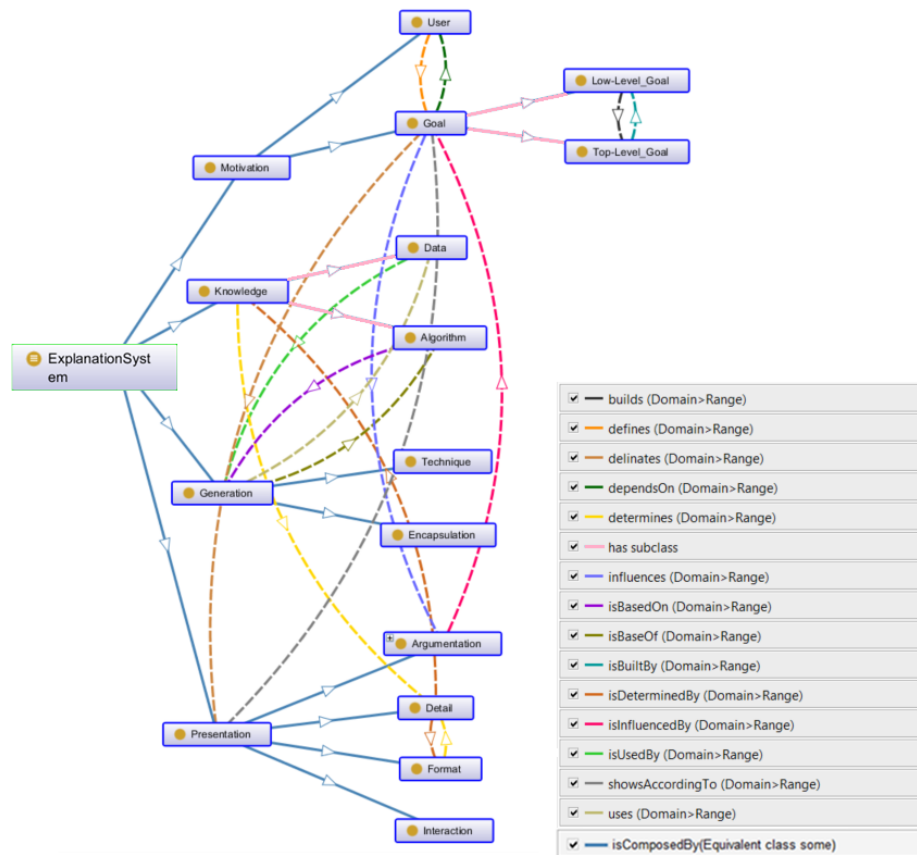


Figura A.5: Visualización de las relaciones existentes en la ontología ExRecOnto

Bibliografía

- AAMODT, A. y PLAZA, E. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI communications*, vol. 7(1), páginas 39–59, 1994.
- ADDAGARLA, S. K. y AMALANATHAN, A. e-SimNet: a visual similar product recommender system for e-Commerce. *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 22(1), páginas 563–570, 2021.
- ADHIKARI, A. Example and feature importance-based explanations for black-box machine learning models. 2018.
- ADOMAVICIUS, G. y TUZHILIN, A. Context-aware recommender systems. En *Recommender systems handbook*, páginas 217–253. Springer, 2011.
- AGGARWAL, C. C. *Recommender systems*. Springer, 2016.
- AIOLLI, F. Efficient top-n recommendation for very large scale binary rated datasets. En *Proceedings of the 7th ACM conference on Recommender systems*, páginas 273–280. 2013.
- AL-OUFI, S., KIM, H.-N. y EL SADDIK, A. Controlling privacy with trust-aware link prediction in online social networks. En *Proceedings of the Third International Conference on Internet Multimedia Computing and Service*, páginas 86–89. 2011.
- ALSHAMMARI, G., JORRO-ARAGONESES, J. L., KAPETANAKIS, S., PETRIDIS, M., RECIO-GARCÍA, J. A. y DÍAZ-AGUDO, B. A hybrid CBR approach for the long tail problem in recommender systems. En *International Conference on Case-Based Reasoning*, páginas 35–45. Springer, 2017.
- ANDJELKOVIC, I., PARRA, D. y O'DONOVAN, J. Moodplay: Interactive music recommendation based on artists' mood similarity. *International Journal of Human-Computer Studies*, vol. 121, páginas 142–159, 2019.
- ANGWIN, J., LARSON, J., MATTU, S. y KIRCHNER, L. Machine Bias: There's software used across the country to predict future criminals. And it's biased against blacks. 2016.

- ARMENTANO, M. G., GODOY, D. y AMANDI, A. A. Followee recommendation based on text analysis of micro-blogging activity. *Information systems*, vol. 38(8), páginas 1116–1127, 2013.
- ARRIETA, A. B., DÍAZ-RODRÍGUEZ, N., DEL SER, J., BENNETOT, A., TABIK, S., BARBADO, A., GARCÍA, S., GIL-LÓPEZ, S., MOLINA, D., BENJAMINS, R. ET AL. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, vol. 58, páginas 82–115, 2020.
- ARYA, V., BELLAMY, R. K., CHEN, P.-Y., DHURANDHAR, A., HIND, M., HOFFMAN, S. C., HOUDE, S., LIAO, Q. V., LUSS, R., MOJSILOVIC, A. ET AL. One Explanation Does Not Fit All: A Toolkit and Taxonomy of AI Explainability Techniques. 2019.
- BAADER, F., HORROCKS, I. y SATTLER, U. Description logics. *Foundations of Artificial Intelligence*, vol. 3, páginas 135–179, 2008.
- BARABÁSI, A.-L. Network science book. *Network Science*, vol. 625, 2014.
- BARBIERI, N., BONCHI, F. y MANCO, G. Who to follow and why: link prediction with explanations. En *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, páginas 1266–1275. ACM, 2014.
- BECK, F., BURCH, M., DIEHL, S. y WEISKOPF, D. A taxonomy and survey of dynamic graph visualization. En *Computer Graphics Forum*, vol. 36, páginas 133–159. Wiley Online Library, 2017.
- BELL, R. M. y KOREN, Y. Lessons from the Netflix prize challenge. *ACM SIGKDD Explorations Newsletter*, vol. 9(2), páginas 75–79, 2007.
- BENNETT, J., LANNING, S. ET AL. The Netflix prize. En *Proceedings of KDD cup and workshop*, vol. 2007, página 35. New York, NY, USA., 2007.
- BERKOVSKY, S., TAIB, R. y CONWAY, D. How to recommend?: User trust factors in movie recommender systems. En *Proceedings of the 22nd International Conference on Intelligent User Interfaces*, páginas 287–300. ACM, 2017.
- BIRKHOFF, G. *Lattice theory*, vol. 25. American Mathematical Soc., 1940.
- BOBADILLA, J., ORTEGA, F., HERNANDO, A. y GUTIÉRREZ, A. Recommender systems survey. *Knowledge-based systems*, vol. 46, páginas 109–132, 2013.
- BOKDE, D., GIRASE, S. y MUKHOPADHYAY, D. Matrix factorization model in collaborative filtering algorithms: A survey. *Procedia Computer Science*, vol. 49, páginas 136–146, 2015.

- BOSTANDJIEV, S., O'DONOVAN, J. y HÖLLERER, T. TasteWeights: a visual interactive hybrid recommender system. En *Proceedings of the sixth ACM conference on Recommender systems*, páginas 35–42. 2012.
- BRATH, R. y JONKER, D. *Graph analysis and visualization: discovering business opportunity in linked data*. John Wiley & Sons, 2015.
- BRIDGE, D., GÖKER, M. H., MCGINTY, L. y SMYTH, B. Case-based recommender systems. *The Knowledge Engineering Review*, vol. 20(3), página 315, 2005.
- BRINTON, C. A Framework for explanation of machine learning decisions. En *IJCAI-17 Workshop on Explainable AI (XAI)*, página 14. 2017.
- BUCHANAN, B. Rule based expert systems. *The MYCIN Experiments of the Stanford Heuristic Programming Project*, 1984.
- BUCHANAN, B. y SHORTLIFFE, E. The MYCIN experiments of the stanford heuristic programming project. *Reading, MA: Addison-Wasley*, 1984.
- BUHRMESTER, V., MÜNCH, D. y ARENS, M. Analysis of explainers of black box deep neural networks for computer vision: A survey. 2019.
- BURKE, R. Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, vol. 12(4), páginas 331–370, 2002.
- CAI, C. J., JONGEJAN, J. y HOLBROOK, J. The effects of example-based explanations in a machine learning interface. En *Proceedings of the 24th international conference on intelligent user interfaces*, páginas 258–262. 2019.
- ÇANO, E. y MORISIO, M. Hybrid recommender systems: A systematic literature review. *Intelligent Data Analysis*, vol. 21(6), páginas 1487–1524, 2017.
- CARDOSO, B., SEDRAKYAN, G., GUTIÉRREZ, F., PARRA, D., BRUSILOVSKY, P. y VERBERT, K. Intersectionexplorer, a multi-perspective approach for exploring recommendations. *International Journal of Human-Computer Studies*, vol. 121, páginas 73–92, 2019.
- CARO MARTÍNEZ, M. Sistemas de recomendación basados en técnicas de predicción de enlaces para jueces en línea. 2017.
- CARO-MARTINEZ, M. Recommender systems and explanations based on interaction graphs and link prediction techniques. *International Conference on Case-Based Reasoning*, página 204, 2018.

- CARO-MARTINEZ, M. y JIMENEZ-DIAZ, G. Similar users or similar items? comparing similarity-based approaches for recommender systems in online judges. En *International Conference on Case-Based Reasoning*, páginas 92–107. Springer, 2017.
- CARO-MARTINEZ, M., JIMENEZ-DIAZ, G. y RECIO-GARCIA, J. A. A theoretical model of explanations in recommender systems. *International Conference on Case-Based Reasoning*, página 52, 2018.
- CARO-MARTINEZ, M., JIMENEZ-DIAZ, G. y RECIO-GARCIA, J. A. A survey of techniques for the evaluation of explanations in recommender systems. *International Conference on Case-Based Reasoning*, 2019a.
- CARO-MARTINEZ, M., JIMENEZ-DIAZ, G. y RECIO-GARCIA, J. A. Conceptual Modeling of Explainable Recommender Systems: An Ontological Formalization to Guide Their Design and Development. *Journal of Artificial Intelligence Research*, vol. 71, páginas 557–589, 2021.
- CARO-MARTINEZ, M., JIMENEZ-DIAZ, G. y RECIO-GARCIA, J. A. A graph-based approach for minimizing the knowledge requirement of explainable recommender systems. Pendiente de aceptar.
- CARO-MARTINEZ, M., JIMENEZ-DIAZ, G. y RECIO-GARCIA, J. A. Local model-agnostic explanations for black-box recommender systems using interaction graphs and link prediction techniques. *International Journal of Interactive Multimedia and Artificial Intelligence*, Pendiente de publicar.
- CARO-MARTINEZ, M., JORRO-ARAGONESES, J. L., DIAZ-AGUDO, B., JIMENEZ-DIAZ, G. y RECIO-GARCIA, J. A. UPIEM: A personalized and interactive explanation method for matrix factorization recommender systems. Pendiente de aceptar.
- CARO-MARTINEZ, M., RECIO-GARCIA, J. A. y JIMENEZ-DIAZ, G. An algorithm independent case-based explanation approach for recommender systems using interaction graphs. En *International Conference on Case-Based Reasoning*, páginas 17–32. Springer, 2019b.
- CHANG, S., HARPER, F. M. y TERVEEN, L. G. Crowd-based personalized natural language explanations for recommendations. En *Proceedings of the 10th ACM Conference on Recommender Systems*, páginas 175–182. ACM, 2016.
- CHEN, R., HUA, Q., CHANG, Y.-S., WANG, B., ZHANG, L. y KONG, X. A survey of collaborative filtering-based recommender systems: From traditional methods to hybrid methods based on social networks. *IEEE Access*, vol. 6, páginas 64301–64320, 2018.

- CHEN, W., GUO, F. y WANG, F.-Y. A survey of traffic data visualization. *IEEE Transactions on Intelligent Transportation Systems*, vol. 16(6), páginas 2970–2984, 2015.
- CHEN, W., HSU, W. y LEE, M. L. Tagcloud-based explanation with feedback for recommender systems. En *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, páginas 945–948. ACM, 2013.
- CHEN, Y. y MIYAZAKI, J. A model-agnostic recommendation explanation system based on knowledge graph. En *International Conference on Database and Expert Systems Applications*, páginas 149–163. Springer, 2020.
- CHICAIZA, J. y VALDIVIEZO-DIAZ, P. A comprehensive survey of knowledge graph-based recommender systems: Technologies, development, and contributions. *Information*, vol. 12(6), página 232, 2021.
- DARLINGTON, K. W. Designing for explanation in health care applications of expert systems. *Sage Open*, vol. 1(1), 2011.
- DE GEMMIS, M., LOPS, P., SEMERARO, G. y MUSTO, C. An investigation on the serendipity problem in recommender systems. *Information Processing & Management*, vol. 51(5), páginas 695–717, 2015.
- DIAZ-AGUDO, B., CARO-MARTINEZ, M., RECIO-GARCIA, J. A., JORRO-ARAGONESES, J. y JIMENEZ-DIAZ, G. Explanation of recommenders using formal concept analysis. En *International Conference on Case-Based Reasoning*, páginas 33–48. Springer, 2019.
- DIAZ-AGUDO, B. y GONZÁLEZ-CALERO, P. A. CBR_{Onto}: a task/method ontology for CBR. *Procs. of the 15th International FLAIRS*, vol. 2, páginas 101–106, 2002.
- DONG, R., SCHAAL, M., O'MAHONY, M. P., MCCARTHY, K. y SMYTH, B. Opinionated product recommendation. En *International Conference on Case-Based Reasoning*, páginas 44–58. Springer, 2013.
- DONKERS, T., LOEPP, B. y ZIEGLER, J. Explaining Recommendations by Means of User Reviews. En *Workshop On Explainable Smart Systems (EXSS)*. 2018.
- DOOMS, S., BELLOGÍN, A., PESSEMIER, T. D. y MARTENS, L. A framework for dataset benchmarking and its application to a new movie rating dataset. *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 7(3), página 41, 2016.
- DOYLE, D., TSYMBAL, A. y CUNNINGHAM, P. A review of explanation and explanation in case-based reasoning. Informe técnico, Trinity College Dublin, Department of Computer Science, 2003.

- EIBAND, M., SCHNEIDER, H. y BUSCHEK, D. Normative vs Pragmatic: Two Perspectives on the Design of Explanations in Intelligent Systems. En *Workshop On Explainable Smart Systems (EXSS)*. 2018.
- EKSTRAND, M. D., RIEDL, J. T. y KONSTAN, J. A. *Collaborative filtering recommender systems*. Now Publishers Inc, 2011.
- FINCH, I. Knowledge-based systems, viewpoints and the world wide web. IET, 1998.
- FRIEDMAN, A., KNIJNENBURG, B. P., VANHECKE, K., MARTENS, L. y BERKOVSKY, S. Privacy aspects of recommender systems. En *Recommender Systems Handbook*, páginas 649–688. Springer, 2015.
- FRIEDRICH, G. y ZANKER, M. A taxonomy for generating explanations in recommender systems. *AI Magazine*, vol. 32(3), páginas 90–98, 2011.
- GAVALAS, D., KONSTANTOPOULOS, C., MASTAKAS, K. y PANTZIOU, G. Mobile recommender systems in tourism. *Journal of network and computer applications*, vol. 39, páginas 319–333, 2014.
- GEDIKLI, F., JANNACH, D. y GE, M. How should I explain? A comparison of different explanation types for recommender systems. *International Journal of Human-Computer Studies*, vol. 72(4), páginas 367–382, 2014.
- GHAZIMATIN, A., BALALAU, O., SAHA ROY, R. y WEIKUM, G. Prince: Provider-side interpretability with counterfactual explanations in recommender systems. En *Proceedings of the 13th International Conference on Web Search and Data Mining*, páginas 196–204. 2020.
- GIBONEY, J. S., BROWN, S. A., LOWRY, P. B. y NUNAMAKER JR, J. F. User acceptance of knowledge-based system recommendations: Explanations, arguments, and fit. *Decision Support Systems*, vol. 72, páginas 1–10, 2015.
- GILPIN, L. H., BAU, D., YUAN, B. Z., BAJWA, A., SPECTER, M. y KAGAL, L. Explaining explanations: An overview of interpretability of machine learning. En *2018 IEEE 5th International Conference on data science and advanced analytics (DSAA)*, páginas 80–89. IEEE, 2018.
- GÓMEZ-VALLEJO, H., URIEL-LATORRE, B., SANDE-MEIJIDE, M., VILLAMARÍN-BELLO, B., PAVÓN, R., FDEZ-RIVEROLA, F. y GLEZ-PEÑA, D. A case-based reasoning system for aiding detection and classification of nosocomial infections. *Decision Support Systems*, vol. 84, páginas 104–116, 2016.
- GOODMAN, B. y FLAXMAN, S. European union regulations on algorithmic decision-making and a “right to explanation”. *AI magazine*, vol. 38(3), páginas 50–57, 2017.

- GUIMERÀ, R. y SALES-PARDO, M. Missing and spurious interactions and the reconstruction of complex networks. *Proceedings of the National Academy of Sciences*, vol. 106(52), páginas 22073–22078, 2009.
- GUNNING, D., STEFIK, M., CHOI, J., MILLER, T., STUMPF, S. y YANG, G.-Z. XAI-Explainable artificial intelligence. *Science Robotics*, vol. 4(37), 2019.
- GUO, Q., ZHUANG, F., QIN, C., ZHU, H., XIE, X., XIONG, H. y HE, Q. A survey on knowledge graph-based recommender systems. *IEEE Transactions on Knowledge and Data Engineering*, 2020.
- HADLAK, S., SCHUMANN, H. y SCHULZ, H.-J. A survey of multi-faceted graph visualization. En *EuroVis (STARs)*, páginas 1–20. 2015.
- HAMEED, M. A., AL JADAAN, O. y RAMACHANDRAM, S. Collaborative filtering based recommendation system: A survey. *International Journal on Computer Science and Engineering*, vol. 4(5), página 859, 2012.
- HAND, D. J. y VINCIOTTI, V. Local versus global models for classification problems: fitting models where it matters. *The American Statistician*, vol. 57(2), páginas 124–131, 2003.
- HANNON, J., BENNETT, M. y SMYTH, B. Recommending Twitter users to follow using content and collaborative filtering approaches. En *Proceedings of the fourth ACM conference on Recommender systems*, páginas 199–206. ACM, 2010.
- HARUNA, K., AKMAR ISMAIL, M., SUHENDROYONO, S., DAMIASIH, D., PIEREWAN, A. C., CHIROMA, H. y HERAWAN, T. Context-aware recommender system: A review of recent developmental process and future research direction. *Applied Sciences*, vol. 7(12), página 1211, 2017.
- HE, C., PARRA, D. y VERBERT, K. Interactive recommender systems: A survey of the state of the art and future research challenges and opportunities. *Expert Systems with Applications*, vol. 56, páginas 9–27, 2016.
- HERLOCKER, J. L., KONSTAN, J. A. y RIEDL, J. Explaining collaborative filtering recommendations. En *Proceedings of the 2000 ACM conference on Computer supported cooperative work*, páginas 241–250. ACM, 2000.
- HERLOCKER, J. L., KONSTAN, J. A., TERVEEN, L. G. y RIEDL, J. T. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, vol. 22(1), páginas 5–53, 2004.
- HERMAN, I., MELANÇON, G. y MARSHALL, M. S. Graph visualization and navigation in information visualization: A survey. *IEEE Transactions on visualization and computer graphics*, vol. 6(1), páginas 24–43, 2000.

- HUANG, Z. y ZENG, D. D. A link prediction approach to anomalous email detection. En *2006 IEEE International Conference on Systems, Man and Cybernetics*, vol. 2, páginas 1131–1136. IEEE, 2006.
- HUANG, Z., ZENG, D. D. y CHEN, H. Analyzing consumer-product graphs: Empirical findings and applications in recommender systems. *Management science*, vol. 53(7), páginas 1146–1164, 2007.
- IAQUINTA, L., DE GEMMIS, M., LOPS, P., SEMERARO, G., FILANNINO, M. y MOLINO, P. Introducing serendipity in a content-based recommender system. En *2008 Eighth International Conference on Hybrid Intelligent Systems*, páginas 168–173. IEEE, 2008.
- JAIN, G., MAHARA, T. y TRIPATHI, K. N. A survey of similarity measures for collaborative filtering-based recommender system. En *Soft computing: theories and applications*, páginas 343–352. Springer, 2020.
- JALILI, M., AHMADIAN, S., IZADI, M., MORADI, P. y SALEHI, M. Evaluating collaborative filtering recommender algorithms: a survey. *IEEE access*, vol. 6, páginas 74003–74024, 2018.
- JAMESON, A. More than the sum of its members: challenges for group recommender systems. En *Proceedings of the working conference on Advanced visual interfaces*, páginas 48–54. ACM, 2004.
- JAMESON, A. y SMYTH, B. Recommendation to groups. En *The adaptive web*, páginas 596–627. Springer, 2007.
- JANNACH, D., ZANKER, M., FELFERNIG, A. y FRIEDRICH, G. *Recommender systems: an introduction*. Cambridge University Press, 2010.
- JI, S., PAN, S., CAMBRIA, E., MARTTINEN, P. y PHILIP, S. Y. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Transactions on Neural Networks and Learning Systems*, 2021.
- JORRO-ARAGONESES, J., CARO-MARTINEZ, M., RECIO-GARCIA, J. A., DIAZ-AGUDO, B. y JIMENEZ-DIAZ, G. Personalized case-based explanation of matrix factorization recommendations. En *International Conference on Case-Based Reasoning*, páginas 140–154. Springer, 2019.
- JORRO-ARAGONESES, J. L., AGUDO, M. B. D. y GARCÍA, J. A. R. Madrid live: A context-aware recomendar system of leisure plans. En *2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI)*, páginas 796–801. IEEE, 2017a.
- JORRO-ARAGONESES, J. L., CARO-MARTÍNEZ, M., DÍAZ-AGUDO, B. y RECIO-GARCÍA, J. A. A user-centric evaluation to generate case-based explanations using formal concept analysis. En *International Conference on Case-Based Reasoning*, páginas 195–210. Springer, 2020a.

- JORRO-ARAGONESES, J. L., CERON-RIOS, G. M., DÍAZ-AGUDO, M. B., RECIO-GARCÍA, J. A. y LÓPEZ-GUTIERREZ, D. M. RecOnto: an ontology to model recommender systems and its components. En *2017 IEEE 29th International Conference on Tools with Artificial Intelligence (IC-TAI)*, páginas 815–821. IEEE, 2017b.
- JORRO-ARAGONESES, J. L., DÍAZ-AGUDO, B., RECIO-GARCÍA, J. A. y JIMENEZ-DÍAZ, G. RecoLibry Suite: a set of intelligent tools for the development of recommender systems. *Automated Software Engineering*, vol. 27(1), páginas 63–89, 2020b.
- KANGASRÄÄSIÖ, A., GLOWACKA, D. y KASKI, S. Improving controllability and predictability of interactive recommendation interfaces for exploratory search. En *Proceedings of the 20th international conference on intelligent user interfaces*, páginas 247–251. 2015.
- KARIMI, M., JANNACH, D. y JUGOVAC, M. News recommender systems—survey and roads ahead. *Information Processing & Management*, vol. 54(6), páginas 1203–1227, 2018.
- KC, M., CHAU, R., HAGENBUCHNER, M., TSOI, A. C. y LEE, V. A machine learning approach to link prediction for interlinked documents. En *International Workshop of the Initiative for the Evaluation of XML Retrieval*, páginas 342–354. Springer, 2009.
- KLEINERMAN, A., ROSENFELD, A. y KRAUS, S. Providing explanations for recommendations in reciprocal environments. En *Proceedings of the 12th ACM Conference on Recommender Systems*, páginas 22–30. ACM, 2018.
- KNUBLAUCH, H. Ontology-driven software development in the context of the semantic web: An example scenario with Protege/OWL. En *1st International workshop on the model-driven semantic web (MDSW2004)*, páginas 381–401. Monterey, California, USA, 2004.
- KOLODNER, J. *Case-based reasoning*. Morgan Kaufmann, 2014.
- KOREN, Y. y BELL, R. Advances in collaborative filtering. *Recommender systems handbook*, páginas 77–118, 2015.
- KOREN, Y., BELL, R. y VOLINSKY, C. Matrix factorization techniques for recommender systems. *Computer*, (8), páginas 30–37, 2009.
- KOTKOV, D., WANG, S. y VEIJALAINEN, J. A survey of serendipity in recommender systems. *Knowledge-Based Systems*, vol. 111, páginas 180–192, 2016.
- KULESZA, T., STUMPF, S., BURNETT, M. y KWAN, I. Tell me more? The effects of mental model soundness on personalizing an intelligent agent. En

- Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, páginas 1–10. ACM, USA, 2012.
- KUMAR, A., SINGH, S. S., SINGH, K. y BISWAS, B. Link prediction techniques, applications, and performance: A survey. *Physica A: Statistical Mechanics and its Applications*, vol. 553, página 124289, 2020.
- KUMAR, P. P., VAIRACHILAI, S., POTLURI, S. y MOHANTY, S. N. *Recommender Systems: Algorithms and Applications*. CRC Press, 2021.
- KUNKEL, J., DONKERS, T., MICHAEL, L., BARBU, C.-M. y ZIEGLER, J. Let me explain: impact of personal and impersonal explanations on trust in recommender systems. En *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, páginas 1–12. 2019.
- KWON, O.-H., MUELDER, C., LEE, K. y MA, K.-L. A study of layout, rendering, and interaction methods for immersive graph visualization. *IEEE transactions on visualization and computer graphics*, vol. 22(7), páginas 1802–1815, 2016.
- LAMCHE, B., ADIGÜZEL, U. y WÖRNDL, W. Interactive explanations in mobile shopping recommender systems. En *Joint Workshop on Interfaces and Human Decision Making in Recommender Systems*, página 14. 2014.
- LECUE, F. On the role of knowledge graphs in explainable AI. *Semantic Web*, vol. 11(1), páginas 41–51, 2020.
- LEE, B., PLAISANT, C., PARR, C. S., FEKETE, J.-D. y HENRY, N. Task taxonomy for graph visualization. En *Proceedings of the 2006 AVI workshop on Beyond time and errors: novel evaluation methods for information visualization*, páginas 1–5. 2006.
- LEE, O.-J. y JUNG, J. J. Explainable Movie Recommendation Systems by using Story-based Similarity. En *Workshop On Explainable Smart Systems (EXSS)*. 2018.
- LI, L., ZHANG, Y. y CHEN, L. Generate neural template explanations for recommendation. En *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, páginas 755–764. 2020.
- LI, R., KAHOU, S., SCHULZ, H., MICHALSKI, V., CHARLIN, L. y PAL, C. Towards deep conversational recommendations. 2018.
- LIBEN-NOWELL, D. y KLEINBERG, J. The link-prediction problem for social networks. *Journal of the American society for information science and technology*, vol. 58(7), páginas 1019–1031, 2007.

- LICHTENWALTER, R. N., LUSSIER, J. T. y CHAWLA, N. V. New perspectives and methods in link prediction. En *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, páginas 243–252. 2010.
- LIKA, B., KOLOMVATSOS, K. y HADJIEFTHYMIADES, S. Facing the cold start problem in recommender systems. *Expert Systems with Applications*, vol. 41(4), páginas 2065–2073, 2014.
- LINDEN, G., SMITH, B. y YORK, J. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, vol. 7(1), páginas 76–80, 2003.
- LIU, F. y LEE, H. J. Use of social network information to enhance collaborative filtering performance. *Expert systems with applications*, vol. 37(7), páginas 4772–4778, 2010.
- LIU, H., HU, Z., MIAN, A., TIAN, H. y ZHU, X. A new user similarity model to improve the accuracy of collaborative filtering. *Knowledge-Based Systems*, vol. 56, páginas 156–166, 2014.
- LIU, J. y DUAN, L. A survey on knowledge graph-based recommender systems. En *2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, vol. 5, páginas 2450–2453. IEEE, 2021.
- LOPS, P., DE GEMMIS, M. y SEMERARO, G. Content-based recommender systems: State of the art and trends. En *Recommender systems handbook*, páginas 73–105. Springer, 2011.
- LU, J., WU, D., MAO, M., WANG, W. y ZHANG, G. Recommender system application developments: a survey. *Decision Support Systems*, vol. 74, páginas 12–32, 2015.
- LÜ, L., MEDO, M., YEUNG, C. H., ZHANG, Y.-C., ZHANG, Z.-K. y ZHOU, T. Recommender systems. *Physics reports*, vol. 519(1), páginas 1–49, 2012.
- LÜ, L. y ZHOU, T. Link prediction in weighted networks: The role of weak ties. *EPL (Europhysics Letters)*, vol. 89(1), página 18001, 2010.
- LÜ, L. y ZHOU, T. Link prediction in complex networks: A survey. *Physica A: statistical mechanics and its applications*, vol. 390(6), páginas 1150–1170, 2011.
- LU, Y., DONG, R. y SMYTH, B. Coevolutionary recommendation model: Mutual learning between ratings and reviews. En *Proceedings of the 2018 World Wide Web Conference*, páginas 773–782. 2018.

- LUDWIG, J., KALTON, A. y STOTTLER, R. Explaining Complex Scheduling Decisions. En *Workshop On Explainable Smart Systems (EXSS)*. 2018.
- MA, H., YANG, H., LYU, M. R. y KING, I. Sorec: social recommendation using probabilistic matrix factorization. En *Proceedings of the 17th ACM conference on Information and knowledge management*, páginas 931–940. ACM, 2008.
- MARTÍNEZ, V., BERZAL, F. y CUBERO, J.-C. A survey of link prediction in complex networks. *ACM computing surveys (CSUR)*, vol. 49(4), páginas 1–33, 2016.
- MASSA, P. y AVESANI, P. Trust-aware collaborative filtering for recommender systems. En *OTM Confederated International Conferences “On the Move to Meaningful Internet Systems”*, páginas 492–508. Springer, 2004.
- MAVETERA, N. y KROEZE, J. H. An ontology-driven software development framework. 2010.
- MCCARTHY, K., REILLY, J., MCGINTY, L. y SMYTH, B. Thinking positively-explanatory feedback for conversational recommender systems. En *Proceedings of the European Conference on Case-Based Reasoning (ECCBR-04) Explanation Workshop*, páginas 115–124. 2004.
- MC FEE, B. y LANCKRIET, G. R. Metric learning to rank. En *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, páginas 775–782. 2010.
- MCGUINNESS, D. L., VAN HARMELEN, F. ET AL. OWL web ontology language overview. *W3C recommendation*, vol. 10, 2004.
- MC SHERRY, D. Explanation in recommender systems. *Artificial Intelligence Review*, vol. 24(2), páginas 179–197, 2005.
- MILLER, T. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, vol. 267, páginas 1–38, 2019.
- MIRANDA, C. y JORGE, A. M. Incremental collaborative filtering for binary ratings. En *2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, vol. 1, páginas 389–392. IEEE, 2008.
- MITTAL, V. O. y PARIS, C. L. Generating explanations in context: The system perspective. *Expert Systems with Applications*, vol. 8(4), páginas 491–503, 1995.
- MIYAHARA, K. y PAZZANI, M. J. Collaborative filtering with the simple bayesian classifier. En *Pacific Rim International conference on artificial intelligence*, páginas 679–689. Springer, 2000.

- MOHRI, M., ROSTAMIZADEH, A. y TALWALKAR, A. *Foundations of machine learning*. MIT press, 2012.
- MUHAMMAD, K., LAWLOR, A., RAFTER, R. y SMYTH, B. Great explanations: Opinionated explanations for recommendations. En *International Conference on Case-Based Reasoning*, páginas 244–258. Springer, 2015.
- MUHAMMAD, K. I., LAWLOR, A. y SMYTH, B. A Live-USER Study of Opinionated Explanations for Recommender Systems. En *Proceedings of the 21st International Conference on Intelligent User Interfaces*, páginas 256–260. 2016.
- MURTAZINA, M. y AVDEENKO, T. The ontology-driven approach to intelligent support of requirements engineering in agile software development. En *2020 International Conference on Information Technology and Nanotechnology (ITNT)*, páginas 1–6. IEEE, 2020.
- MUSTO, C., DE GEMMIS, M., LOPS, P. y SEMERARO, G. Generating post hoc review-based natural language justifications for recommender systems. *User Modeling and User-Adapted Interaction*, páginas 1–45, 2020.
- MUSTO, C., NARDUCCI, F., LOPS, P., DE GEMMIS, M. y SEMERARO, G. ExpLOD: a framework for explaining recommendations based on the linked open data cloud. En *Proceedings of the 10th ACM Conference on Recommender Systems*, páginas 151–154. ACM, 2016.
- NARDI, D., BRACHMAN, R. J. ET AL. An introduction to description logics. *Description logic handbook*, vol. 1, página 40, 2003.
- NGO, T., KUNKEL, J. y ZIEGLER, J. Exploring mental models for transparent and controllable recommender systems: A qualitative study. En *Proceedings of the 28th ACM Conference on User Modeling, Adaptation and Personalization*, páginas 183–191. AMC, Italy, 2020.
- NUNES, I. y JANNACH, D. A systematic review and taxonomy of explanations in decision support and recommender systems. *User Modeling and User-Adapted Interaction*, vol. 27(3-5), páginas 393–444, 2017.
- O'DONOVAN, J., SMYTH, B., GRETARSSON, B., BOSTANDJIEV, S. y HÖLLERER, T. PeerChooser: visual interactive recommendation. En *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, páginas 1085–1088. 2008.
- ONTAÑÓN, S. An overview of distance and similarity functions for structured data. *Artificial Intelligence Review*, vol. 53(7), páginas 5309–5351, 2020.
- PAN, J. Z., STAAB, S., ASSMANN, U., EBERT, J. y ZHAO, Y. *Ontology-driven software development*. Springer Science & Business Media, 2012.

- PAPADIMITRIOU, A., SYMEONIDIS, P. y MANOLOPOULOS, Y. A generalized taxonomy of explanations styles for traditional and social recommender systems. *Data Mining and Knowledge Discovery*, vol. 24(3), páginas 555–583, 2012.
- PARK, Y.-J. y TUZHILIN, A. The long tail of recommender systems and how to leverage it. En *Proceedings of the 2008 ACM conference on Recommender systems*, páginas 11–18. 2008.
- PARRA, D., BRUSILOVSKY, P. y TRATTNER, C. See what you want to see: visual user-driven approach for hybrid recommendation. En *Proceedings of the 19th international conference on Intelligent User Interfaces*, páginas 235–240. 2014.
- PARSIA, B. y SIRIN, E. Pellet: An OWL DL reasoner. En *Third international semantic web conference-poster*, vol. 18, página 13. Citeseer, 2004.
- PAVLOV, M. y ICHISE, R. Finding experts by link prediction in co-authorship networks. *FEWS*, vol. 290, páginas 42–55, 2007.
- PAZZANI, M., FEGHAHATI, A., SHELTON, C. y SEITZ, A. Explaining Contrasting Categories. En *Workshop On Explainable Smart Systems (EXSS)*. 2018.
- PAZZANI, M. J. A framework for collaborative, content-based and demographic filtering. *Artificial intelligence review*, vol. 13(5-6), páginas 393–408, 1999.
- PECUNE, F., MURALI, S., TSAI, V., MATSUYAMA, Y. y CASSELL, J. A model of social explanations for a conversational movie recommendation system. En *Proceedings of the 7th International Conference on Human-Agent Interaction*, páginas 135–143. 2019.
- PEÑA, F. J., O'REILLY-MORGAN, D., TRAGOS, E. Z., HURLEY, N., DURIAKOVA, E., SMYTH, B. y LAWLOR, A. Combining rating and review data by initializing latent factor models with topic models for top-n recommendation. En *Fourteenth ACM Conference on Recommender Systems*, páginas 438–443. 2020.
- PHELAN, O., MCCARTHY, K., BENNETT, M. y SMYTH, B. Terms of a feather: Content-based news recommendation and discovery using Twitter. En *European Conference on Information Retrieval*, páginas 448–459. Springer, 2011.
- POWERS, D. M. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. 2011.

- PU, P. y CHEN, L. Trust-inspiring explanation interfaces for recommender systems. *Knowledge-Based Systems*, vol. 20(6), páginas 542–556, 2007.
- QUIJANO SÁNCHEZ, L. Impacto de los factores u organizaciones sociales en los procesos de recomendación para grupos. *Tesis Doctoral*. 2015.
- QUIJANO-SANCHEZ, L., SAUER, C., RECIO-GARCIA, J. A. y DIAZ-AGUDO, B. Make it personal: a social explanation system applied to group recommendations. *Expert Systems with Applications*, vol. 76, páginas 36–48, 2017.
- RASTEGARPANAH, B., CROVELLA, M. y GUMMADI, K. P. Exploring Explanations for Matrix Factorization Recommender Systems. 2017.
- RECIO-GARCIA, J. A., JIMENEZ-DIAZ, G., SANCHEZ-RUIZ, A. A. y DIAZ-AGUDO, B. Personality aware recommendations to groups. En *Proceedings of the third ACM conference on Recommender systems*, páginas 325–328. ACM, 2009.
- REYES, A., IBARGUENGOYTIA, P. H., ELIZALDE, F., SÁNCHEZ, L. y NAVA, A. Asisto: An integrated intelligent assistant system for power plant operation and training. En *2011 16th International Conference on Intelligent System Applications to Power Systems*, páginas 1–6. IEEE, 2011.
- RIBEIRO, M. T., SINGH, S. y GUESTRIN, C. “Why should I trust you?” Explaining the predictions of any classifier. En *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, páginas 1135–1144. 2016.
- RIBEIRO, M. T., SINGH, S. y GUESTRIN, C. Anchors: High-precision model-agnostic explanations. En *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32. 2018.
- RICCI, F., ROKACH, L. y SHAPIRA, B. Recommender systems: Introduction and challenges. En *Recommender systems handbook*, páginas 1–34. Springer, 2015.
- RICHTER, M. M. y WEBER, R. O. *Case-based reasoning*. Springer, 2016.
- RIYABI, M. y SOHRABI, M. K. Providing effective recommendations in discussion groups using a new hybrid recommender system based on implicit ratings and semantic similarity. *Electronic Commerce Research and Applications*, vol. 40, página 100938, 2020.
- ROBILLARD, M., WALKER, R. y ZIMMERMANN, T. Recommendation systems for software engineering. *IEEE software*, vol. 27(4), páginas 80–86, 2010.

- RUIZ INIESTA, A. Estrategias de recomendación basadas en conocimiento para la localización personalizada de recursos en repositorios educativos. *Tesis doctoral*. 2014.
- RUIZ-INIESTA, A., JIMENEZ-DIAZ, G. y GOMEZ-ALBARRAN, M. Recommendation in repositories of learning objects: A proactive approach that exploits diversity and navigation-by-proposing. En *Advanced Learning Technologies, 2009. ICALT 2009. Ninth IEEE International Conference on*, páginas 543–545. IEEE, 2009.
- SADILEK, A., KAUTZ, H. y BIGHAM, J. P. Finding your friends and following them to where you are. En *Proceedings of the fifth ACM international conference on Web search and data mining*, páginas 723–732. 2012.
- SAID, A. y BELLOGÍN, A. Comparative recommender system evaluation: benchmarking recommendation frameworks. En *Proceedings of the 8th ACM Conference on Recommender systems*, páginas 129–136. ACM, 2014.
- SANSONETTI, G., GASPARETTI, F., MICARELLI, A., CENA, F. y GENA, C. Enhancing cultural recommendations through social and linked open data. *User Modeling and User-Adapted Interaction*, vol. 29(1), páginas 121–159, 2019.
- SANZ-CRUZADO, J., PEPA, S. M. y CASTELLS, P. Structural novelty and diversity in link prediction. En *Companion Proceedings of the The Web Conference 2018*, páginas 1347–1351. 2018.
- SARWAR, B. M., KARYPIS, G., KONSTAN, J. A., RIEDL, J. ET AL. Item-based collaborative filtering recommendation algorithms. *WWW*, vol. 1, páginas 285–295, 2001.
- SATO, M., AHSAN, B., NAGATANI, K., SONODA, T., ZHANG, Q. y OHKUMA, T. Explaining Recommendations Using Contexts. En *23rd International Conference on Intelligent User Interfaces*, páginas 659–664. ACM, 2018.
- SAURO, J. y LEWIS, J. R. *Quantifying the user experience: Practical statistics for user research*. Morgan Kaufmann, 2016.
- SCHAFFER, J. B., FRANKOWSKI, D., HERLOCKER, J. y SEN, S. Collaborative filtering recommender systems. En *The adaptive web*, páginas 291–324. Springer, 2007.
- SHARMA, A. y COSLEY, D. Do social explanations work?: studying and modeling the effects of social explanations in recommender systems. En *Proceedings of the 22nd International Conference on World Wide Web*, páginas 1133–1144. ACM, 2013.

- SHARMA, R. y RAY, S. Explanations in recommender systems: an overview. *International Journal of Business Information Systems*, vol. 23(2), páginas 248–262, 2016.
- SHI, C., ZHANG, Z., LUO, P., YU, P. S., YUE, Y. y WU, B. Semantic path based personalized recommendation on weighted heterogeneous information networks. En *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, páginas 453–462. 2015.
- SILVA, N., CARVALHO, D., PEREIRA, A. C., MOURÃO, F. y ROCHA, L. The pure cold-start problem: A deep study about how to conquer first-time users in recommendations domains. *Information Systems*, vol. 80, páginas 1–12, 2019.
- SINGH, J. y ANAND, A. EXS: Explainable search using local model agnostic interpretability. En *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, páginas 770–773. 2019.
- SINHA, R. y SWEARINGEN, K. The role of transparency in recommender systems. En *CHI'02 extended abstracts on Human factors in computing systems*, páginas 830–831. ACM, 2002.
- SLACK, D., FRIEDLER, S., ROY, C. y SCHEIDEGGER, C. Assessing the local interpretability of machine learning models. En *NeurIPS Workshop on Human-Centric Machine Learning (HCML)*. 2019.
- SOARES, M. y VIANA, P. Tuning metadata for better movie content-based recommendation systems. *Multimedia Tools and Applications*, vol. 74(17), páginas 7015–7036, 2015.
- SØRMO, F., CASSENS, J. y AAMODT, A. Explanation in case-based reasoning—perspectives and goals. *Artificial Intelligence Review*, vol. 24(2), páginas 109–143, 2005.
- STAAB, S. y STUDER, R. *Handbook on Ontologies*. Springer Science & Business Media, 2010.
- STRATIGI, M., TZOMPANAKI, K. y STEFANIDIS, K. Why-not questions & explanations for collaborative filtering. En *International Conference on Web Information Systems Engineering*, páginas 301–315. Springer, 2020.
- STUMPF, S., SKREBE, S., AYMER, G. y HOBSON, J. Explaining smart heating systems to discourage fiddling with optimized behavior. En *Workshop On Explainable Smart Systems (EXSS)*. 2018.
- SU, X. y KHOSHGOFTAAR, T. M. Collaborative filtering for multi-class data using belief nets algorithms. En *2006 18th IEEE international conference*

- on Tools with Artificial Intelligence (ICTAI'06)*, páginas 497–504. IEEE, 2006.
- SU, X. y KHOSHGOFTAAR, T. M. A survey of collaborative filtering techniques. *Advances in artificial intelligence*, vol. 2009, 2009.
- SUTTON, R. S. y BARTO, A. G. *Reinforcement learning: An introduction*. MIT press, 2018.
- SYMEONIDIS, P., NANOPOULOS, A. y MANOLOPOULOS, Y. Providing justifications in recommender systems. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 38(6), páginas 1262–1272, 2008.
- SYMEONIDIS, P., NANOPOULOS, A. y MANOLOPOULOS, Y. MoviExplain: a recommender system with explanations. En *Proceedings of the third ACM conference on Recommender systems*, páginas 317–320. ACM, 2009.
- TERANO, T., SUZUKI, M., ONODA, T., UENISHI, K. y MATSUURA, T. CSES: an approach to integrating graphic, music and voice information into a user-friendly interface. En *Industrial Applications of Machine Intelligence and Vision, 1989., International Workshop on*, páginas 349–354. IEEE, 1989.
- THALHEIM, B. The conceptual model \equiv an adequate and faithful artifact enhanced by concepts. *Frontiers in Artificial Intelligence and Applications*, vol. 260, páginas 241–254, 2014.
- THORAT, P. B., GOUDAR, R. y BARVE, S. Survey on collaborative filtering, content-based filtering and hybrid recommendation system. *International Journal of Computer Applications*, vol. 110(4), páginas 31–36, 2015.
- TINTAREV, N. Explaining recommendations. En *International Conference on User Modeling*, páginas 470–474. Springer, 2007.
- TINTAREV, N. y MASTHOFF, J. The effectiveness of personalized movie explanations: An experiment using commercial meta-data. En *International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, páginas 204–213. Springer, 2008.
- TINTAREV, N. y MASTHOFF, J. Evaluating the effectiveness of explanations for recommender systems. *User Modeling and User-Adapted Interaction*, vol. 22(4-5), páginas 399–439, 2012.
- TINTAREV, N. y MASTHOFF, J. Explaining recommendations: Design and evaluation. En *Recommender Systems Handbook*, páginas 353–382. Springer, 2015.

- TSAI, C.-H. y BRUSILOVSKY, P. Providing Control and Transparency in a Social Recommender System for Academic Conferences. En *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization*, páginas 313–317. ACM, 2017.
- TSAI, C.-H. y BRUSILOVSKY, P. The effects of controllability and explainability in a social recommender system. *User Modeling and User-Adapted Interaction*, páginas 1–37, 2020.
- VARGA, E. Recommender systems. En *Practical Data Science with Python 3*, páginas 317–339. Springer, 2019.
- VERBERT, K., PARRA, D., BRUSILOVSKY, P. y DUVAL, E. Visualizing recommendations to support exploration, transparency and controllability. En *Proceedings of the 2013 international conference on Intelligent user interfaces*, páginas 351–362. 2013.
- VIG, J., SEN, S. y RIEDL, J. Tagsplanations: explaining recommendations using tags. En *Proceedings of the 14th International Conference on Intelligent User Interfaces*, páginas 47–56. ACM, 2009.
- VOLKOV, M. y YU, G. W. Effective latent models for binary feedback in recommender systems. En *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, páginas 313–322. 2015.
- VAN DER WAA, J., VAN DIGGELEN, J. y NEERINCX, M. The design and validation of an intuitive confidence measure. En *Workshop On Explainable Smart Systems (EXSS)*, vol. 2, página 1. 2018.
- VAN DER WAA, J., NIEUWBURG, E., CREMERS, A. y NEERINCX, M. Evaluating XAI: A comparison of rule-based and example-based explanations. *Artificial Intelligence*, vol. 291, página 103404, 2021.
- WALEK, B. y FOJTIK, V. A hybrid recommender system for recommending relevant movies using an expert system. *Expert Systems with Applications*, vol. 158, página 113452, 2020.
- WANG, H., FI, Q., LIU, L. y SONG, W. A probabilistic rating prediction and explanation inference model for recommender systems. *China Communications*, vol. 13(2), páginas 79–94, 2016.
- WANG, H., ZHANG, F., WANG, J., ZHAO, M., LI, W., XIE, X. y GUO, M. Ripplenet: Propagating user preferences on the knowledge graph for recommender systems. En *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, páginas 417–426. 2018a.

- WANG, H., ZHANG, F., XIE, X. y GUO, M. DKN: Deep knowledge-aware network for news recommendation. En *Proceedings of the 2018 World Wide Web conference*, páginas 1835–1844. 2018b.
- WANG, P., XU, B., WU, Y. y ZHOU, X. Link prediction in social networks: the state-of-the-art. *Science China Information Sciences*, vol. 58(1), páginas 1–38, 2015.
- WANG, S., HU, L., WANG, Y., HE, X., SHENG, Q. Z., ORGUN, M., CAO, L., WANG, N., RICCI, F. y YU, P. S. Graph learning approaches to recommender systems: A review. *arXiv preprint arXiv:2004.11718*, 2020.
- WANG, X., HE, X., FENG, F., NIE, L. y CHUA, T.-S. TEM: Tree-enhanced embedding model for explainable recommendation. En *Proceedings of the 2018 World Wide Web Conference*, páginas 1543–1552. 2018c.
- WANG, X., WANG, D., XU, C., HE, X., CAO, Y. y CHUA, T.-S. Explainable reasoning over knowledge graphs for recommendation. En *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, páginas 5329–5336. 2019.
- WEISS, E., KUROWSKI, K., HISCHKE, S. y XU, B. Avoiding route breakage in ad hoc networks using link prediction. En *Proceedings of the Eighth IEEE Symposium on Computers and Communications. ISCC 2003*, páginas 57–62. IEEE, 2003.
- WEST, D. M. *The future of work: Robots, AI, and automation*. Brookings Institution Press, 2018.
- XIAN, Y., FU, Z., MUTHUKRISHNAN, S., DE MELO, G. y ZHANG, Y. Reinforcement knowledge graph reasoning for explainable recommendation. En *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, páginas 285–294. 2019.
- XIE, L., HU, Z., CAI, X., ZHANG, W. y CHEN, J. Explainable recommendation based on knowledge graph and multi-objective optimization. *Complex & Intelligent Systems*, vol. 7(3), páginas 1241–1252, 2021.
- YANG, S.-Y. Developing an ontology-supported information integration and recommendation system for scholars. *Expert Systems with Applications*, vol. 37(10), páginas 7065–7079, 2010.
- ZAFARANI, R., ABBASI, M. A. y LIU, H. *Social media mining: an introduction*. Cambridge University Press, 2014.
- ZANKER, M. y NINAUS, D. Knowledgeable explanations for recommender systems. En *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, vol. 1, páginas 657–660. IEEE, 2010.

- ZENEBE, A., OZOK, A. y NORCIO, A. F. Personalized recommender systems in e-Commerce and m-Commerce: a comparative study. En *Conference on Human-Computer Interaction (HCI International)*, páginas 1–11. Springer, USA, 2005.
- ZHANG, J., JONES, N. y PU, P. A visual interface for critiquing-based recommender systems. En *Proceedings of the 9th ACM conference on Electronic commerce*, páginas 230–239. 2008.
- ZHANG, Y. y CHEN, X. Explainable recommendation: A survey and new perspectives. *Foundations and Trends in Information Retrieval*, vol. 14(1), páginas 1–101, 2020.
- ZHANG, Y., LAI, G., ZHANG, M., ZHANG, Y., LIU, Y. y MA, S. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. En *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, páginas 83–92. 2014.
- ZHENG, V. W., CAO, B., ZHENG, Y., XIE, X. y YANG, Q. Collaborative filtering meets mobile recommendation: A user-centered approach. En *Twenty-Fourth AAAI Conference on Artificial Intelligence*. 2010.
- ZHOU, T., REN, J., MEDO, M. y ZHANG, Y.-C. Bipartite network projection and personal recommendation. *Physical Review E*, vol. 76(4), 2007.

