

Tema 5

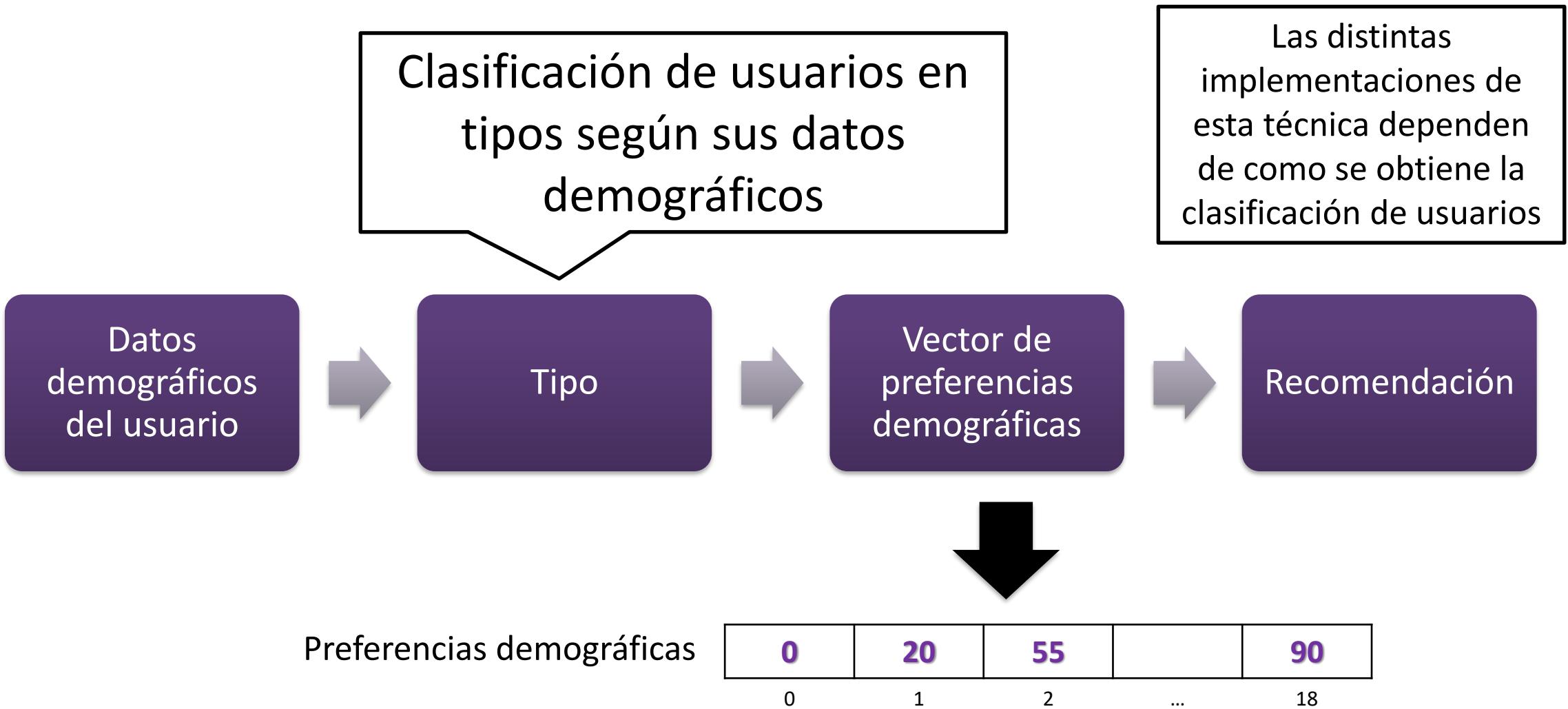
Técnicas de Recomendación Básicas:

SR Colaborativo

SCAR

Sistema complejos Adaptativos y
Recomendación

SR Demográfico



SR Basado en Contenido

Preferencias obtenidas preguntando al usuario u obtenidas de su interacción con el sistema

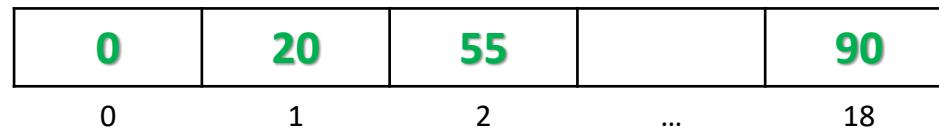
Las distintas implementaciones de esta técnica dependen de como se obtienen las preferencias del usuario

Gustos del usuario

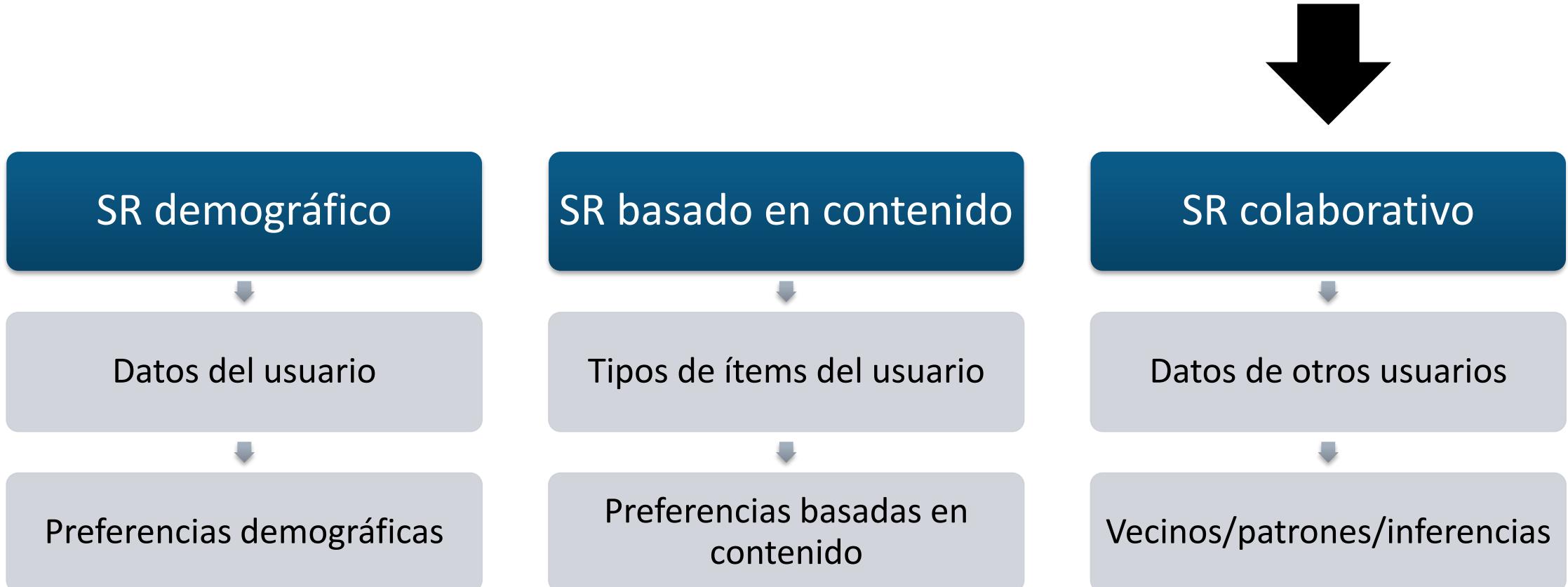
- Vector de preferencias basadas en contenido

Recomendación

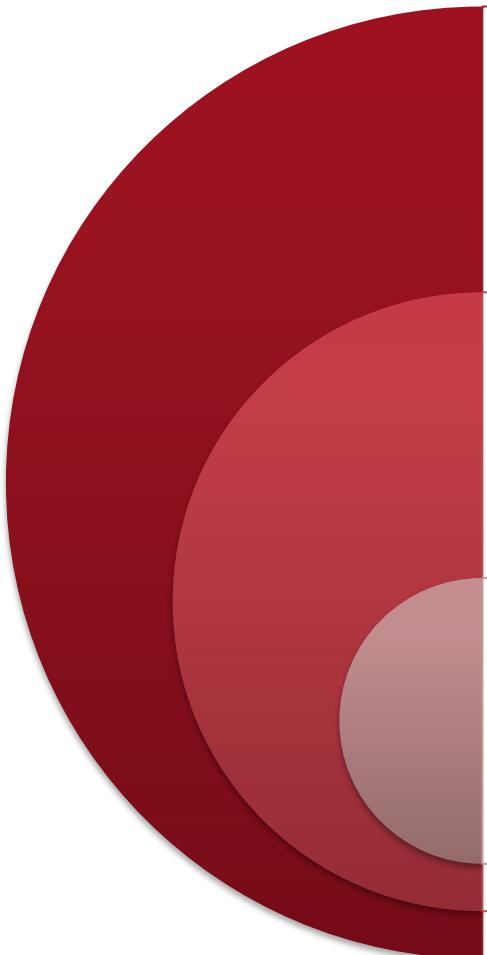
Preferencias basadas en contenido



BRTs



SR Colaborativo



Es la técnica más utilizada

Existen multitud de implementaciones diferentes de la técnica de recomendación colaborativa, pero todas ellas tienen algo en común

Diferentes técnicas como

- Utilizan información de los usuarios del sistema
- Vecinos (KNN): tradicional
- Factorización de matrices
- Técnicas de machine learning

SR Colaborativo



Filtro colaborativo





Ejemplo



Los clientes que vieron este producto también vieron

Página 1 de 10



Cocina día a día: 1095 recetas. 365 menús para las cuatro estaciones (Planeta Cocina)
Karlos Arguiñano
★★★★★ 1.060
Tapa dura 23,70 €



1000 recetas de oro (Planeta Cocina)
Karlos Arguiñano
★★★★★ 62
Tapa blanda 17,00 €



1000 recetas de oro (Planeta Cocina)
Karlos Arguiñano
★★★★★ 704
Tapa dura 23,70 €



En familia con Karlos Arguiñano: Mis mejores recetas para cocinar en casa (Planeta Cocina)
Karlos Arguiñano
★★★★★ 565
Tapa dura 21,75 €



Sabores de siempre: Las recetas que no pasan de moda (Planeta Cocina)
Karlos Arguiñano
★★★★★ 134
Tapa dura 21,75 €



A mi manera: Las recetas fundamentales de la cocina regional española (No Ficción)
Karlos Arguiñano
★★★★★ 70
Tapa dura 21,75 €



SR Colaborativo

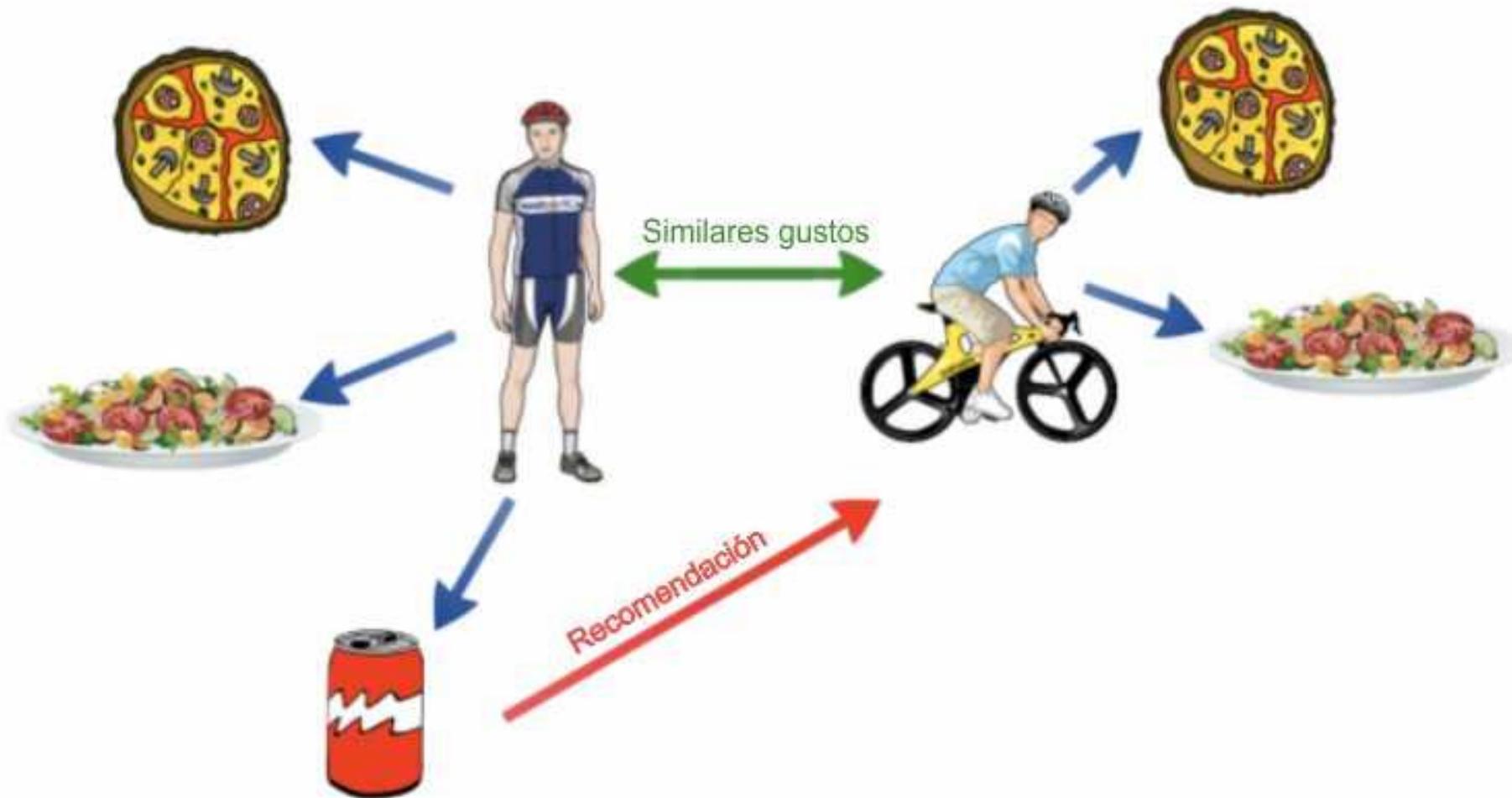
Recomienda ítems en función de los ítems puntuados/comprados/vistos por otros usuarios del sistema, o por las preferencias o características comunes del usuario con el resto de usuarios

El SR colaborativo tradicional define similitudes entre usuarios, y usa los ítems puntuados favorablemente por usuarios similares, para obtener la recomendación

Usuarios similares (vecinos)

- Tienen gustos/características parecidos al usuario actual
- Han puntuado favorablemente ítems que el usuario actual también ha puntuado favorablemente

Ejemplo



SR Colaborativo

Datos base

Ítems

- BD con los ítems a recomendar

Usuarios

- Perfil del usuario
- Perfil de otros usuarios

Información demográfica

- Información personal: edad, género, familia, país,...

Modelo de preferencias del usuario

- Gustos o preferencias
- Tipo de ítems en los que el usuario está interesado

Histórico de interacción del usuario con el sistema

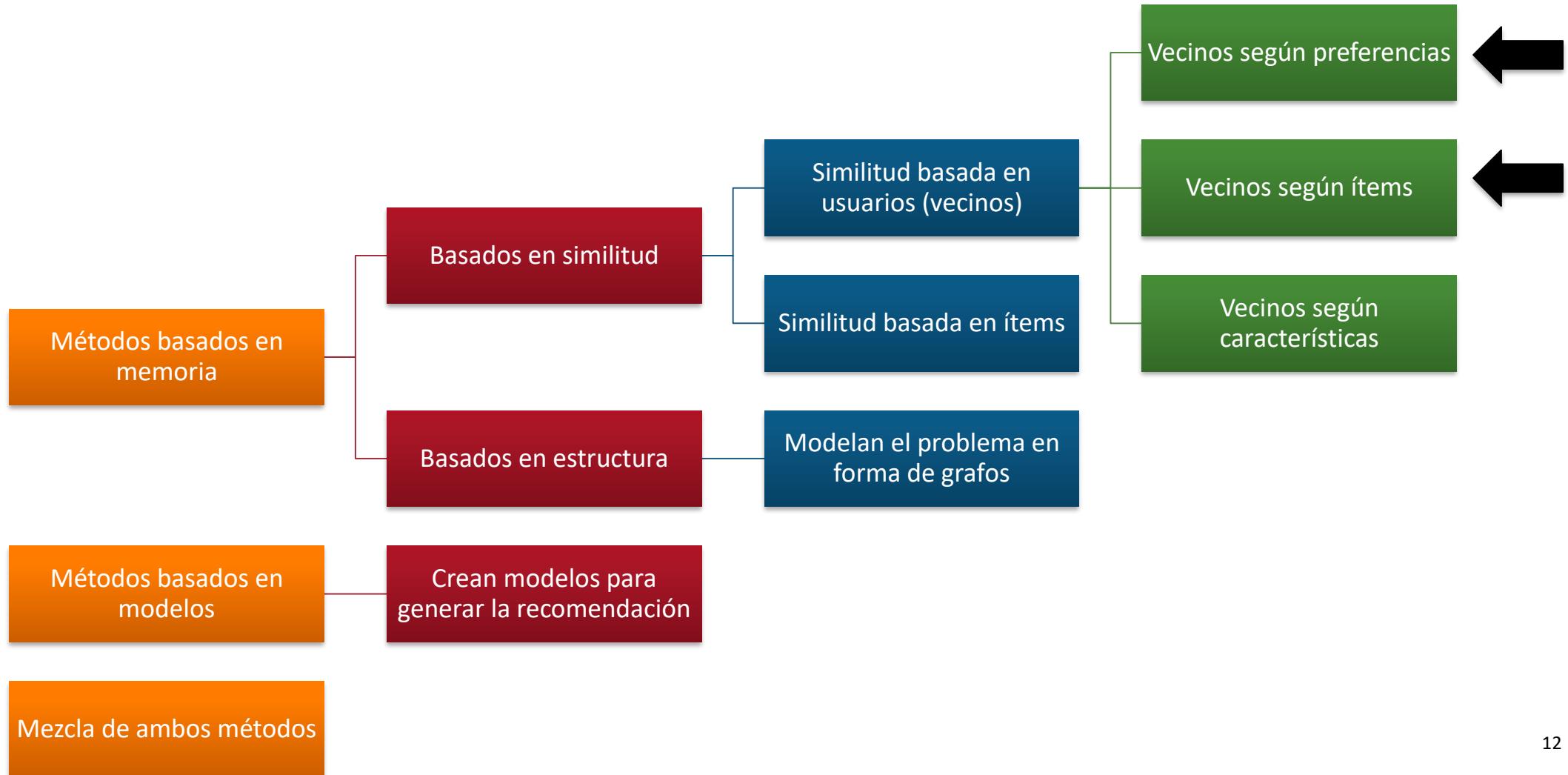
- Conjunto de ítems recomendados, visitados, comprados,... anteriormente por el usuario
- Grado de satisfacción con la recomendación, visita, compra,...

Información interna del SR

- Información que el SR calcula y que facilita el proceso de recomendación
- Clasificación del usuario, usuarios similares,...

Clasificación

Clasificación



Clasificación

Métodos basados en memoria

Usados tradicionalmente

Costosos computacionalmente

Métodos basados en modelos o estadísticas

Se construyen modelos de puntuaciones de usuarios a ítems para obtener la recomendación

Clasificación

The diagram consists of two overlapping circles. The left circle contains the text "Lo que las diferentes estrategias tienen en común". The right circle contains the text "Se basan en la información de otros usuarios del sistema y del usuario que solicita la recomendación". The overlapping area represents the common ground between the two concepts.

Lo que las
diferentes
estrategias
tienen en
común

Se basan en la
información de
otros usuarios
del sistema y del
usuario que
solicita la
recomendación

Clasificación

Collaborative filtering
recommender
systems taxonomy

- H Papadakis, A Papagrigoriou, C Panagiotakis,... Knowledge and Information Systems, volume 64, pages 35–74 (2022)

SR colaborativo basado en memoria

Clasificación SR basados en memoria

Basados en similitud

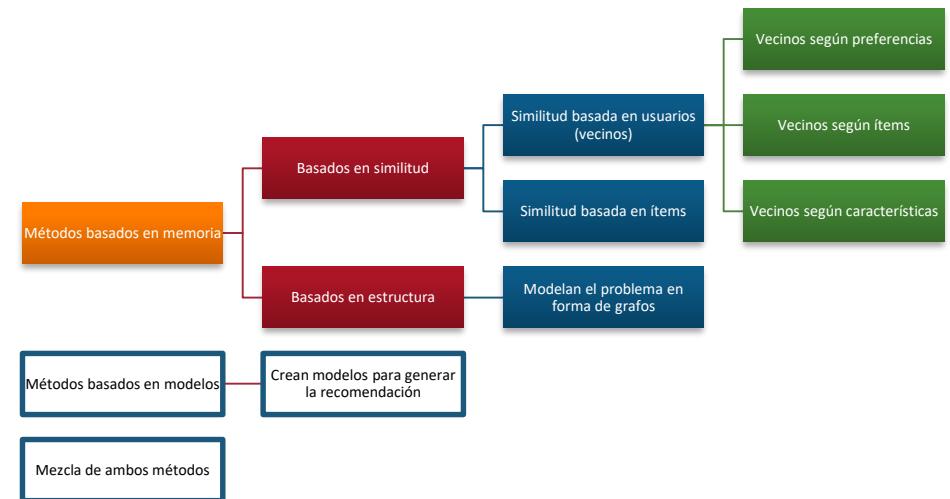
Similitud basada en usuarios (vecinos)

Similitud basada en ítems

Basados en estructura

Modelan el problema en forma de grafos

No los vamos a ver

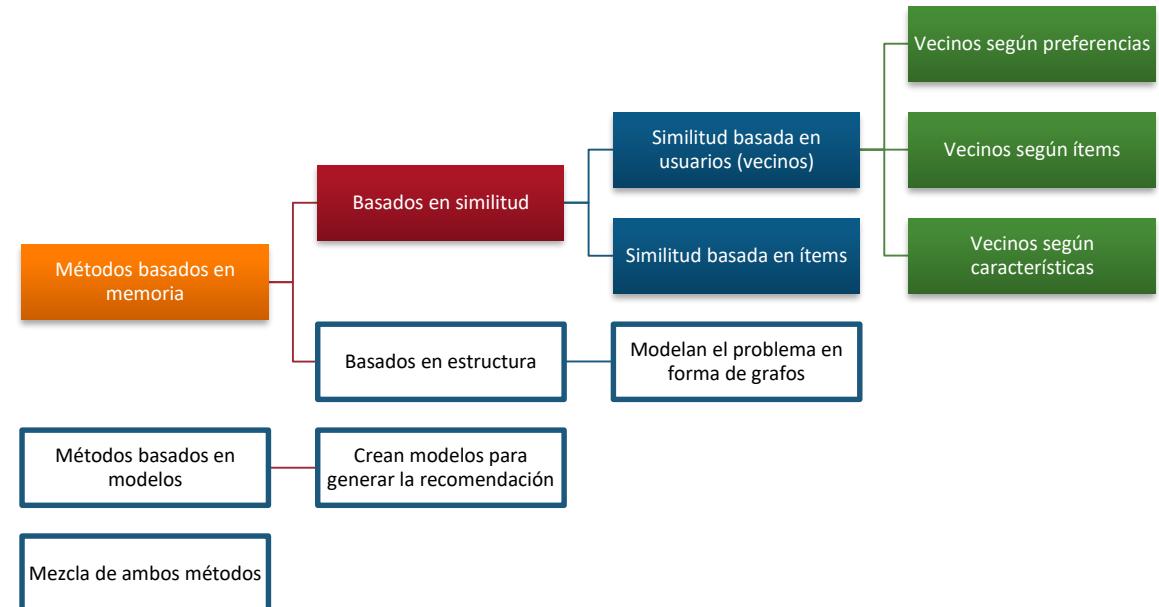


Clasificación SR basados en memoria

Basados en similitud

Similitud basada en usuarios (vecinos)

Similitud basada en ítems



Clasificación SR basados en similitud

Similitud basada en usuarios (vecinos)



Busca vecinos del usuario actual

- Para buscar vecinos puede basarse en las preferencias de los usuarios, en los ítems puntuados de los usuarios o en las características de los usuarios

Similitud basada en ítems



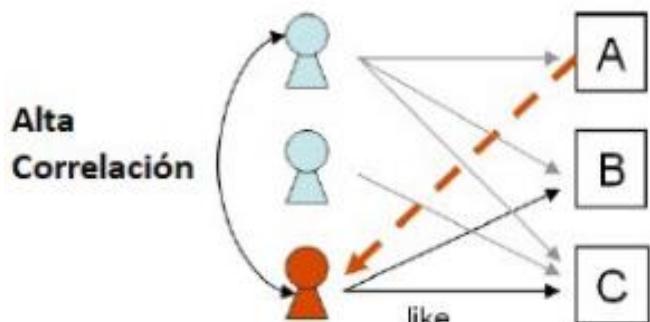
Obtiene similitudes entre los ítems puntuados por los usuarios

- No hay vecinos



Recomienda ítems de los vecinos

SR colaborativo basado en usuarios



User-based filtering
(Grouplens, 1994)

Tomo una muestra de usuarios que comparten **gustos similares**, luego predigo cuánto le gustará a un usuario un ítem dependiendo de cuánto le gustó a otros usuarios.

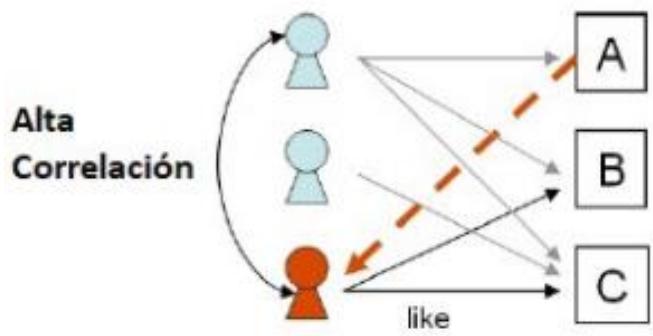
"Puede que te guste porque a "tus amigos" le gustó"

Obtiene la similitud entre los usuarios

Los usuarios de mayor similitud, serán los vecinos

Recomienda ítems de los vecinos

SR colaborativo basado en usuarios



User-based filtering
(Grouplens, 1994)

Tomo una muestra de usuarios que comparten **gustos similares**, luego predigo cuánto le gustará a un usuario un ítem dependiendo de cuánto le gustó a otros usuarios.

"Puede que te guste porque a "tus amigos" le gustó"

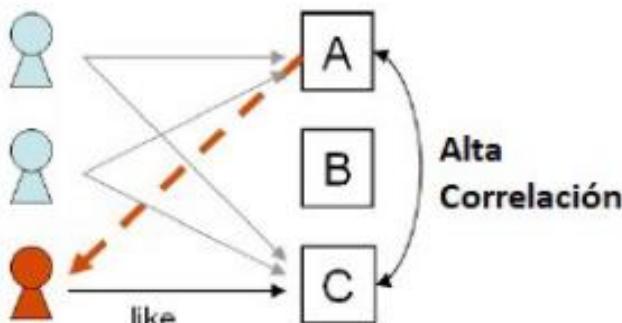
Busca usuarios que puntúen ítems como el usuario rojo

- El primer usuario ha puntuado B y C
- El segundo usuario sólo C
- El primer usuario será vecino

Se recomienda A porque lo ha puntuado su vecino



SR colaborativo basado en ítems



Item-based filtering
(Amazon, 2001)

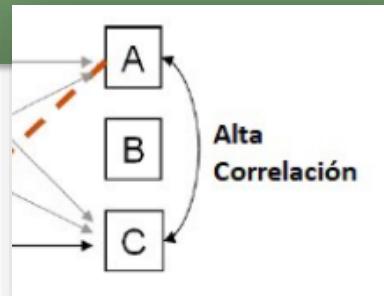
Tomo previamente los ítems que comparten **personas similares** con el "target item", cuánto le gustará al usuario el target item depende de qué tanto le gustó a los otros usuarios los otros ítems.

"Te puede gustar este ítem porque te gustaron los otros ítems"

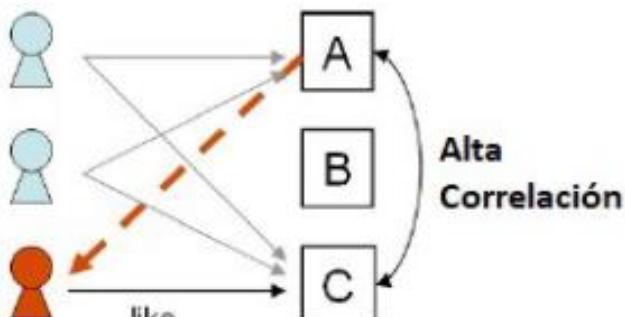
Obtiene una correlación entre los ítems puntuados, sin tener en cuenta el resto de puntuaciones del usuario

Recomienda ítems basados en ese patrón

Los usuarios no se tienen en cuenta



SR colaborativo basado en ítems



Item-based filtering
(Amazon, 2001)

Tomo previamente los ítems que comparten **personas similares** con el "target item", cuánto le gustará al usuario el target item depende de qué tanto le gustó a los otros usuarios los otros ítems.

"Te puede gustar este ítem porque te gustaron los otros ítems"

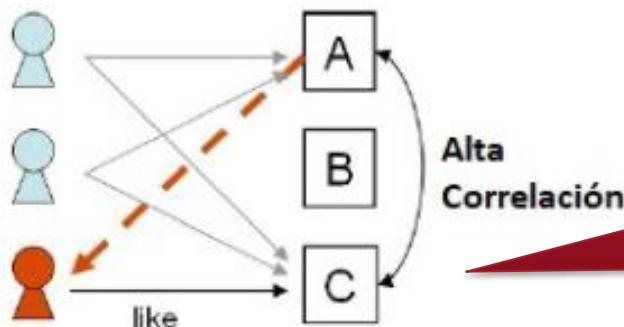


Busca ítems puntuados como C

Cuando se puntúa C también se puntúa A

- Busca ítems puntuados de forma similar

SR colaborativo basado en usuarios vs ítems

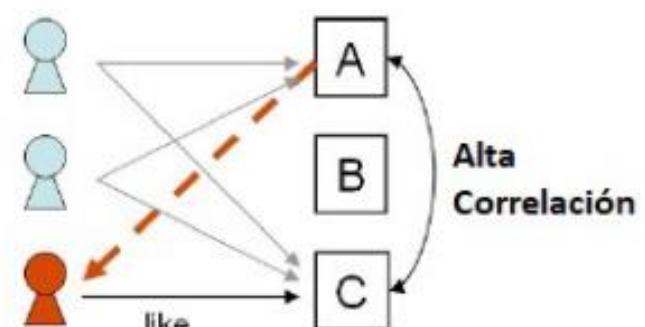


Item-based filtering
(Amazon, 2001)

Tomo previamente los ítems que comparten **personas similares** con el "target item", cuánto le gustará al usuario el target item depende de qué tanto le gustó a los otros usuarios los otros ítems.

"Te puede gustar este ítem porque te gustaron los otros ítems"

Similitud entre la forma de puntuar de los usuarios



Item-based filtering
(Amazon, 2001)

Similitud entre la forma de puntuar cada ítem

Tomo previamente los ítems que comparten **personas similares** con el "target item", cuánto le gustará al usuario el target item depende de qué tanto le gustó a los otros usuarios los otros ítems.

"Te puede gustar este ítem porque te gustaron los otros ítems"

SR colaborativo basado en usuarios vs ítems

Basado en usuarios

Busca vecinos (con gustos o patrones de puntuación similares)

Usa los vecinos en sucesivas recomendaciones

Recomienda ítems puntuados favorablemente por los vecinos

Los vecinos tienen un nivel de afinidad que puede servir para ponderar la recomendación

Basado en ítems

Busca ítems con patrones de puntuación similar a los puntuados por el usuario

Recomienda ítems con un patrón similar

No se guarda la similitud para sucesivas recomendaciones

SR colaborativo basado en usuarios vs ítems

Si tuviésemos una matriz donde los usuarios son las filas y los ítems las columnas:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25		
a	1	4	5		4	3								2		4	2										
b	4				3									5	1	3											
c	5	4	4											3	5												
d				3										3		4	2								3		
e	3			5										5			1									5	4
f	4			1	3	5	4	1						5	4	4	4								3		
g	2	4		4	2		5	1	4	5				4	2	4	5								4		
h	2		1	4	3	5	4	2						5	4	5									5		
i	1			3		5								5		4	4	5							4	3	
j	4				5				1	5				4	4	4											
k	5			4	2	5	1	5		4	2			4	2	4									2		
l		3		3		4	1		4		4	2		4												3	
m	5	3		5	3	5	4	5	3	5	3			4	4	5	4	4									
n	1	4	5		4	5	1	5		4	3			3	4	4	3										
o	4	4			5	4	5							4	2	5	5	5	3								
p	4			5										5	4	2	4	4	5	4	2						
q	3		3						1	5	4	4	4		4										4	3	
r	4	1	4	2					2	5	4			4			5	4	4								
s	2	4	4	5					1		4	2	4	4													
t	1	4	3		4				4	5	5	4	4	4											3		
u	2	1	4	3					1	5	4	2	4	5	4												
v	4	5			4	3	5			2				2											5		
w	2		2	3		5				4	5	4	2		3	4											
x	4		5		3	3			4	5				1													
y	1		3		2	3			3	3	5			4													

Basado en usuarios

Calcula la correlación entre las filas

Usa la correlación para obtener vecinos

- Usa los vecinos para obtener los ítems a recomendar

Basado en ítems

Calcula la correlación entre las columnas

Usa la correlación para obtener los ítems a recomendar

Ejemplo

Basado en usuarios

U1

- Puntúa I1 e I2
- Vecinos similares:
U3
- Recomienda **I3**

U3

- Puntúa I1, I2, I3
- Vecinos similares:
U1, U2 y U5
- Recomienda **I4**

	I1	I2	I3	I4
U1	x	x		
U2	x		x	x
U3	x	x	x	
U4		x		x
U5		x	x	x

	I1	I2	I3	I4
U1	x	x		
U2	x		x	x
U3	x	x	x	
U4		x		x
U5		x	x	x

Ejemplo

Basado en ítems

U1

- Ítems puntuados de forma similar a I1
 - I2 (ya puntuado)
 - I3
- Ítems puntuados de forma similar a I2
 - I1 (ya puntuado)
 - I3
 - I4
- Recomienda I3

U3

- Ítems puntuados de forma similar a I1
 - I2 (ya puntuado)
 - I3 (ya puntuado)
- Ítems puntuados de forma similar a I2:
 - I3 (ya puntuado)
 - I4
- Ítems puntuados de forma similar a I3:
 - I1 (ya puntuado)
 - I2 (ya puntuado)
 - I4
- Recomienda I4

	I1	I2	I3	I4
U1	X	X		
U2	X		X	X
U3	X	X	X	
U4		X		X
U5		X	X	X

	I1	I2	I3	I4
U1	X	X		
U2	X		X	X
U3	X	X	X	
U4		X		X
U5		X	X	X

Ejemplo

Qué recomendamos a U4

Basado en usuarios

- Vecino U5
- Recomienda I3

Basado en ítems

- I2
- I1 (2 puntuaciones)
- I3 (2 puntuaciones)
- I4 (2 puntuaciones)
- I4
- I1 (1 puntuación)
- I2 (ya visto)
- I3 (1 puntuación)
- Recomienda I3 e I1

	I1	I2	I3	I4
U1	x	x		
U2	x		x	x
U3	x	x	x	
U4		x		x
U5		x	x	x

	I1	I2	I3	I4
U1	x	x		
U2	x		x	x
U3	x	x	x	
U4		x		x
U5		x	x	x

SR Colaborativo basado en vecinos (KNN, K-Nearest Neighbour)

Clasificación SR basados en memoria

Basados en similitud

Basados
en
estructura

Similitud basada en usuarios (vecinos)

Similitud
basada en
ítems

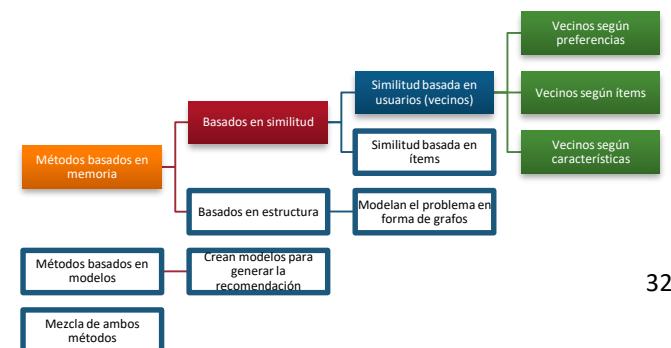
Modelan el
problema en
forma de
grafos

Vecinos según
preferencias

Vecinos según
ítems

Vecinos según
características

Mezcla de
cualquiera de
los anteriores



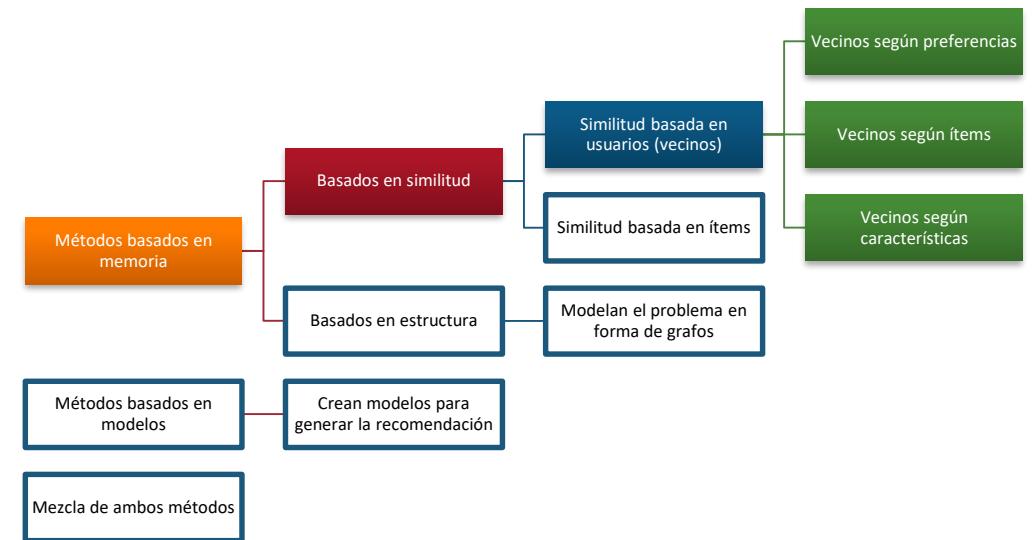
Clasificación SR basados en vecinos

Por simplicidad a los SR

- Basados en memoria
 - Basados en similitud
 - Basados en usuarios

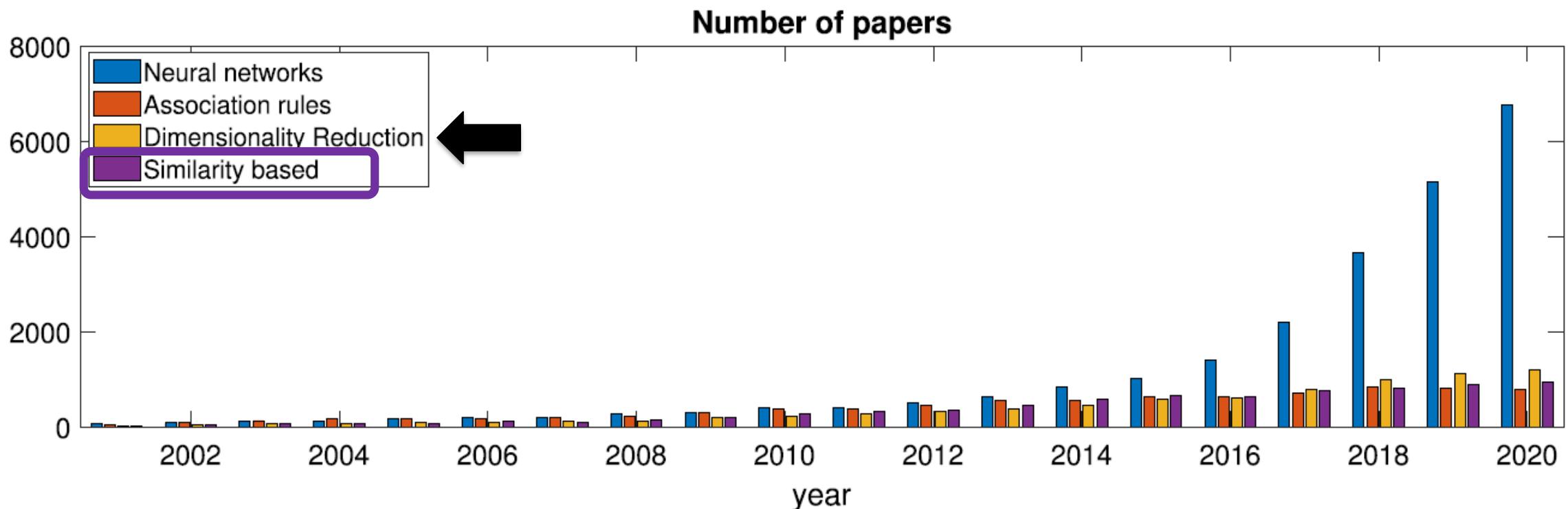
Les llamamos SR basados en vecinos

- No es la forma más adecuada de llamar a estos SR



Artículos sobre SR Colaborativos

Aunque son los SR colaborativos más simples, se sigue investigando en ellos



SR Colaborativo: Vecinos

Obtiene los vecinos del usuario que solicita la recomendación

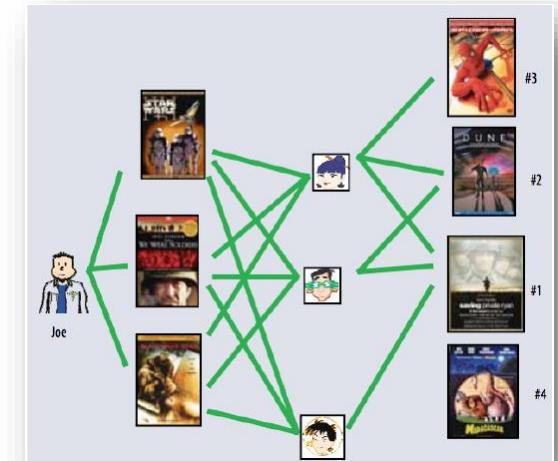
Recomiendan ítems en función del los ítems de los vecinos

Los vecinos son usuarios similares, los que tienen gustos parecidos, o que han puntuado favorablemente ítems similares a los del usuario actual o tienen características similares

Encontrar usuarios vecinos puede ser computacionalmente costoso

- Puede ser un proceso batch
- La lista de vecinos se puede almacenar en el perfil

El número de vecinos no debe superar los **40-50** en sistemas grandes, pues más vecinos sería información redundante, que ralentizaría el proceso y no mejora la recomendación



SR Colaborativo: Vecinos



El proceso de recomendación es similar si nos basamos en preferencias, en ítems o en características

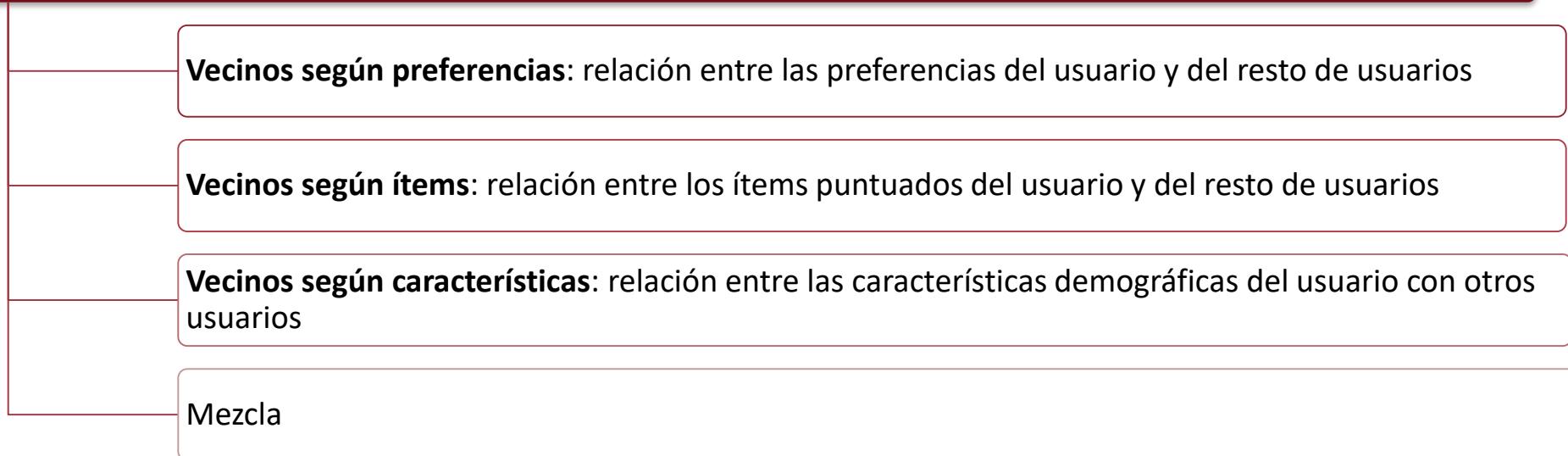
Para obtener vecinos usaremos una matriz para comparar más fácilmente

- La diferencia es la información que se usa para obtener los usuarios vecinos

- Filas: usuarios
- Columnas: preferencias, ítems y/o características

SR colaborativo basado en vecinos

Información usada para hallar similitudes entre los usuarios



Métodos basados
en memoria

Basados en
similitud

Similitud basada en
usuarios (vecinos)

Vecinos según
preferencias

Vecinos según
ítems

Vecinos según
características

Perfil de usuario

Información demográfica

- Información personal: edad, género, familia, país,...

Modelo de preferencias del usuario

- Gustos o preferencias
- Tipo de ítems en los que el usuario está interesado

Histórico de interacción del usuario con el sistema

- Conjunto de ítems recomendados, visitados, comprados,... anteriormente por el usuario
- Grado de satisfacción con la recomendación, visita, compra,...

Información interna del SR

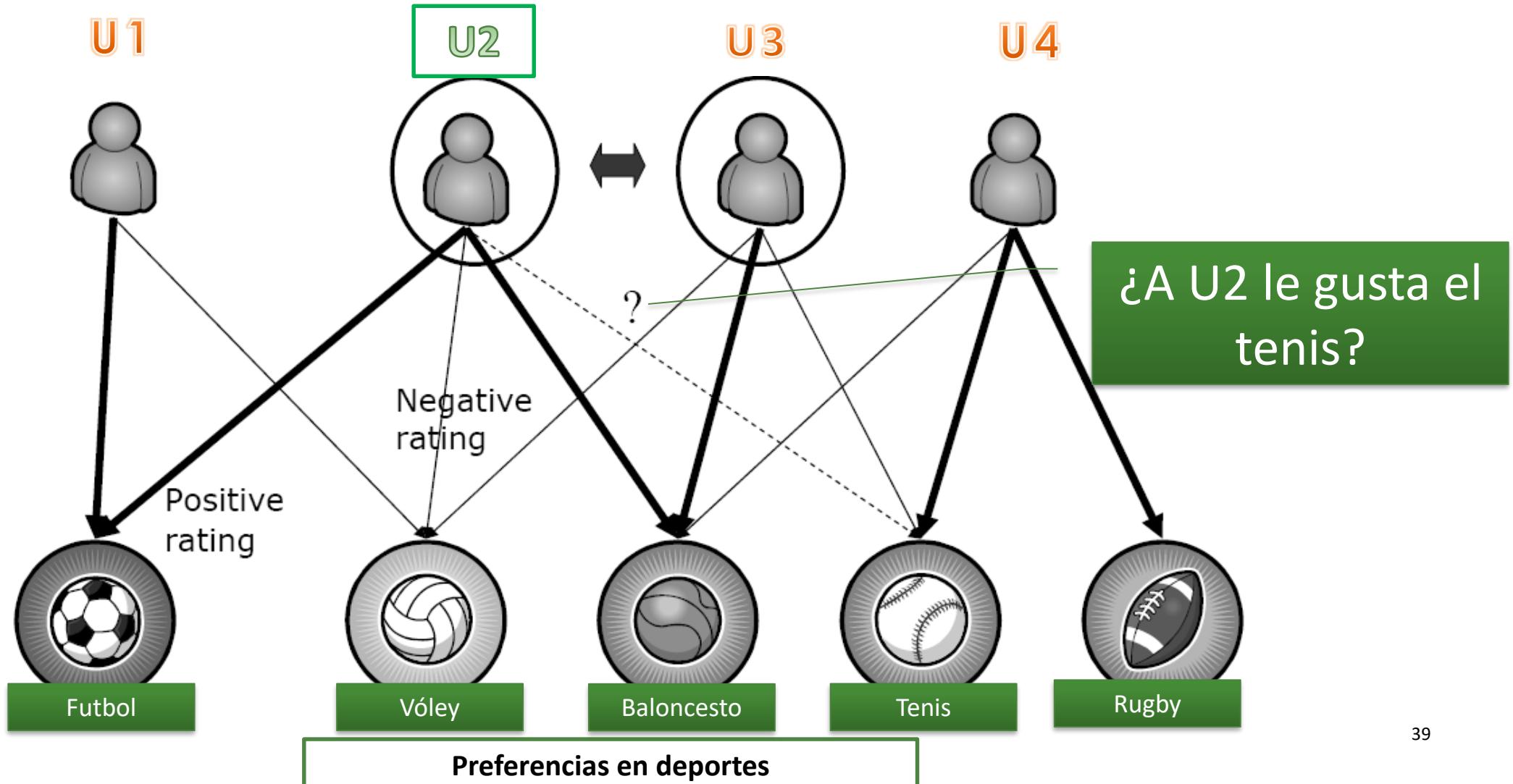
- Información que el SR calcula y que facilita el proceso de recomendación
- Clasificación del usuario, usuarios similares,...

Vecinos según características

Vecinos según preferencias

Vecinos según ítems

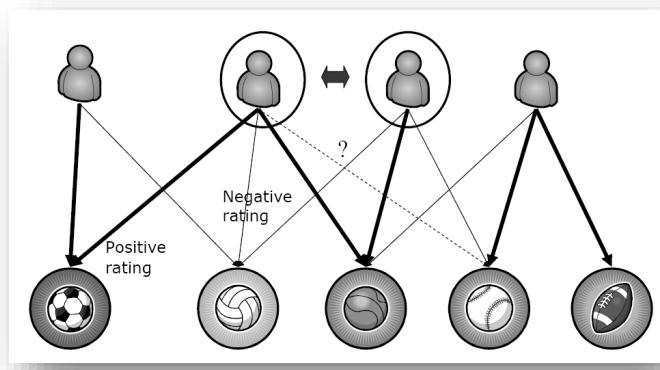
Ejemplo



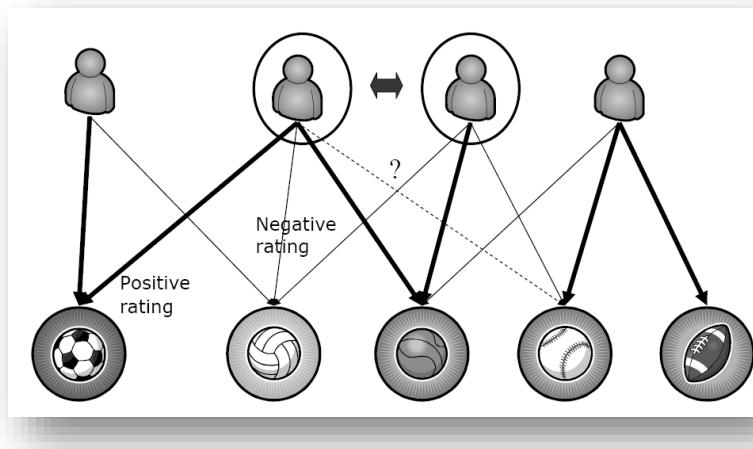
Ejemplo

Crear una matriz con los ratios de todos los usuarios (valores binarios, no tenemos ratios)

	Futbol	Vóley	Baloncesto	Tenis	Rugby
u1	+	-			
u2	+	-	+		
u3		-	+	-	
u4			-	+	+



Ejemplo



	Futbol	Vóley	Baloncesto	Tenis	Rugby
u1	+	-			
u2	+	-	+	?	
u3		-	+	-	
u4			-	+	+

Queremos predecir si al usuario **u2** le gustará el **tenis**

Usuarios similares a u2: u1 y u3

- u1 no tiene puntuado tenis
- Se elije **u3**

u3 ha puntuado negativamente tenis → u2 puntuará negativamente tenis

Ejemplo

	Futbol	Vóley	Baloncesto	Tenis	Rugby
u1	+	-		+	
u2	+	-	+	?	
u3		-	+	-	
u4			-	+	+

Si u1 tuviese puntuado positivamente tenis y u3 negativamente...

No podríamos dar una recomendación, pues ambos usuarios son igual de similares a u2

Ejemplo

	Futbol	Vóley	Baloncesto	Tenis	Rugby
u1	+	-	+	+	
u2	+	-	+	?	
u3		-	+	-	
u4			-	+	+

u1 comparte 3 preferencias con u2

u3 comparte 2 preferencias con u2

u1 es más afín a u2 que u1

- u2 puntuará **positivamente** tenis

Ejemplo

Matriz de calificación de usuarios

Crear una matriz de ratios con los usuarios del sistema

En el ejemplo, se usa similitud entre usuarios basada en ítems

	9	6	8	4	
	2	10	6		8
	5	9		10	7
Active user		?	10	7	8
					?

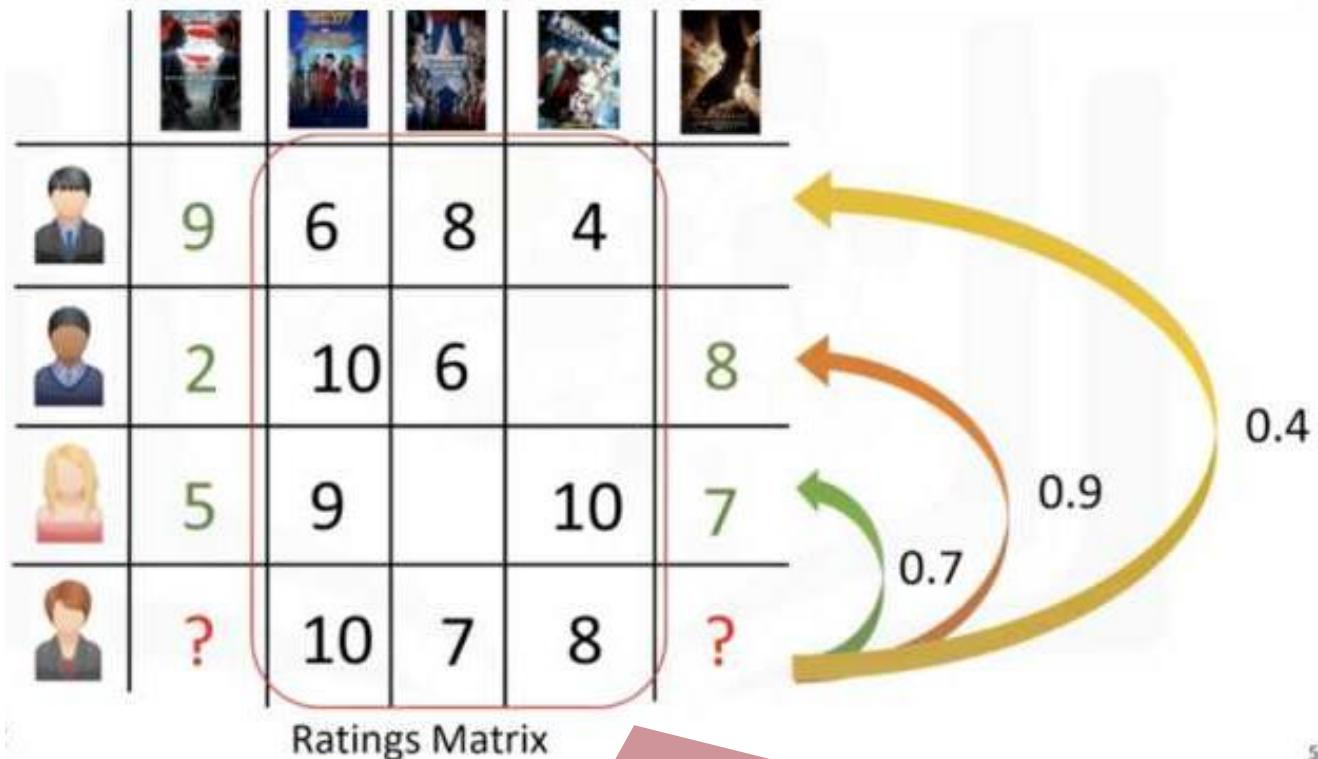
Ratings Matrix

Ejemplo

Calcular el ratio de similitud o afinidad entre el usuario actual y el resto de usuarios

Una vez calculados los ratios de similitud, se pueden elegir los vecinos de similitud mayor que un cierto umbral o los N vecinos de mayor similitud

Calculando similaridad

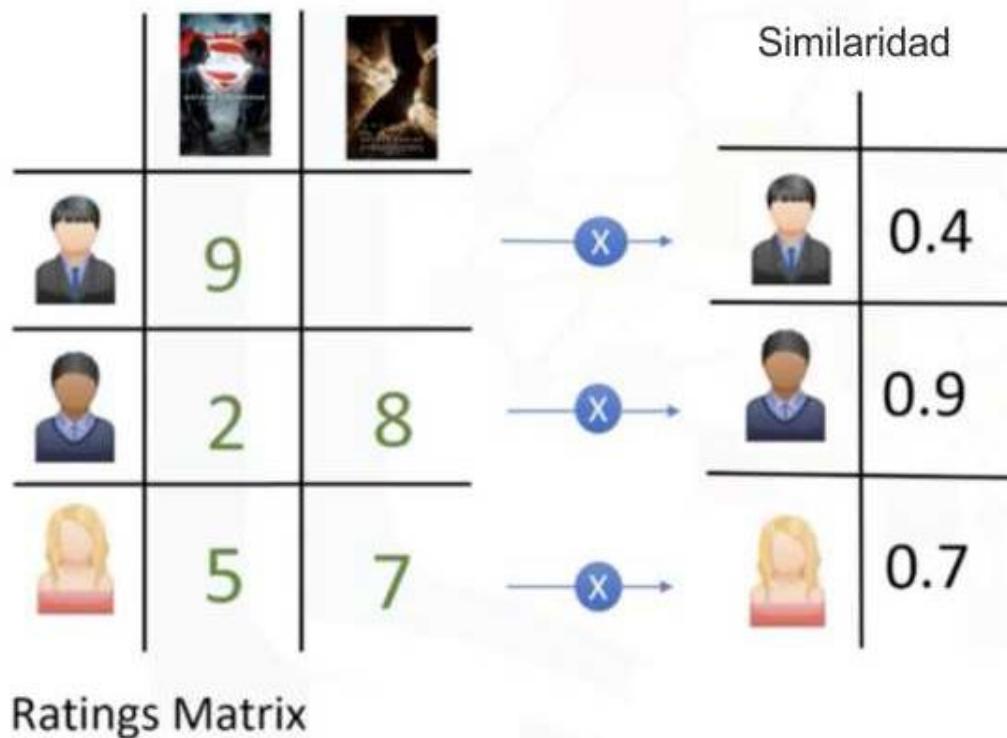


Información usada para calcular la similitud con los usuarios

Ejemplo

Ponderar el ratio dado a cada ítem de cada usuario, el función del ratio de afinidad con el vecino

En este ejemplo se multiplica el ratio dado por el vecino por la afinidad con el vecino



Matriz de calificación ponderada

=

3.6	
1.8	7.2
3.5	4.9

Ejemplo

	Similarity Index		
	0.4	0.9	0.7
Similaridad			
	3.6		
	1.8	7.2	
	3.5	4.9	

Matriz de calificación ponderada

	Weight Sum		
	12.1	(0.9+0.7)	
	8.9	(0.4+0.9+0.7)	

Σ

Para cada ítem, suma los ratios de afinidad de los usuarios que han puntuado el ítem

Para cada ítem, suma los ratios que han dado los usuarios al ítem

Ejemplo

Calcular la predicción del ratio del ítem para el usuario al que se le quiere dar la recomendación

En este ejemplo se multiplica el ratio dado por el vecino por la afinidad con el vecino

Divide el la suma de ratios del ítem por la suma de las similitudes de los usuarios

$$8.9/2=4.45$$

$$12.1/1.6=7.56$$

			Weight Sum
	12.1	(0.9+0.7)	
	8.9	(0.4+0.9+0.7)	
		4.4	7.5

Matriz de recomendación

Ejemplo

Ya hay un ratio calculado para cada ítem

¿Qué se le recomienda?

- ¿Todos los ítems? ¿Sólo un ítem?
- ¿Los ítems que superan un umbral (5, por ejemplo)?

La interfaz de usuario tendrá que decidir, qué mostrar y cómo

- ¿Se muestra una explicación?
- ¿Se muestran los ratios obtenidos? ¿Estrellas?

	4.4	7.5

Proceso de recomendación

1

Calcular los vecinos del usuario

- Los vecinos son el conjunto de usuarios con **mayor similitud** al usuario actual
- Para obtenerlos
 - Crear la matriz de ratios
 - Calcular la similitud entre el usuario y el resto de usuarios
 - Seleccionar los usuarios más afines
- Puede ser un proceso previo
 - Se calculan los vecinos y se almacenan en el perfil

2

Obtener los ítems a recomendar

- Obtener el conjunto de ítems recomendados:
 - Similitud basada en ítems: se obtienen los ítems directamente de los puntuados por los vecinos
 - Similitud basada en preferencias: se obtienen las preferencias de los vecinos y de éstas, los ítems
 - Similitud basada en características: se obtienen las características de los vecinos, y de éstas, los ítems
- Eliminar los ítems ya puntuados por el usuario
- Generar una predicción (ratio) para cada uno de los ítems de los vecinos
- Selecciona los N mejores ítems

1. Calcular los vecinos del usuario

Para calcular la similitud entre los usuarios se pueden usar

Mezcla de información

Preferencias comunes

- Se buscan las similitudes entre las preferencias del usuario y las del resto de usuarios

Ítems comunes

- Se buscan las similitudes entre los ítems puntuados por el usuario actual y otros usuarios

Características comunes

- Se buscan las similitudes entre características del usuario actual y otros usuarios
- No es muy habitual disponer de datos demográficos suficientes, por lo que no aparece en los ejemplos

1. Calcular los vecinos del usuario

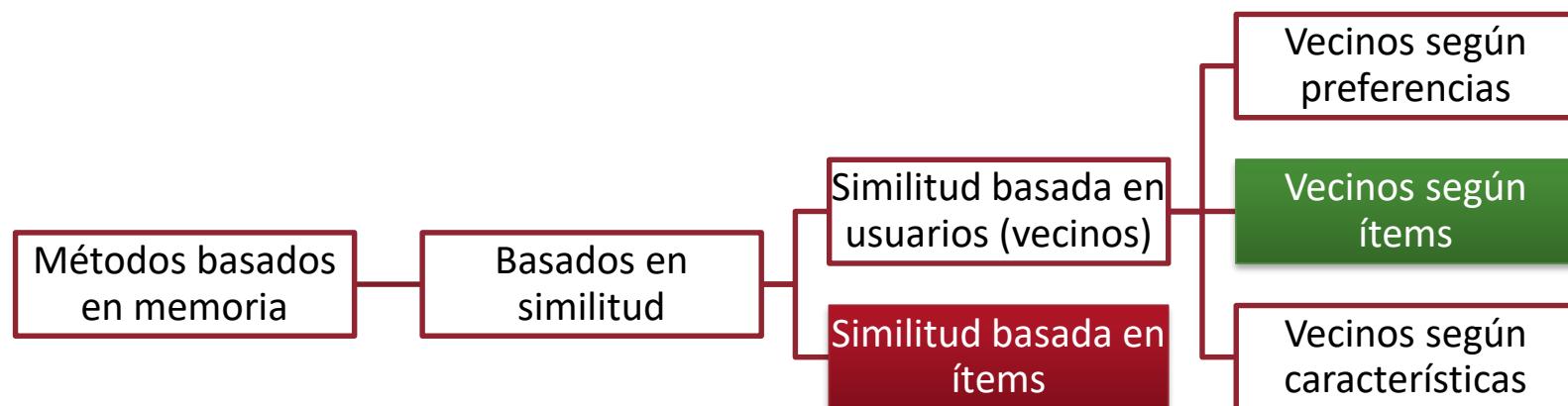
Hay que distinguir entre:

Similitud basada en ítems

Búsqueda de vecinos usando ítems

No tiene en cuenta usuarios, sólo forma en que se puntuán los ítems

Para obtener la similitud entre los usuarios usa los ítems que éstos han puntuado



1. Calcular los vecinos del usuario: Matriz de ratios

Ítems,
preferencias
y/o
características

Se crea una **matriz de ratios**

La matriz contiene

- En un eje, los usuarios
- En otro eje, cada uno de los ítems, preferencias o características

Usuarios

Problema

- La matriz tiene un tamaño muy grande, sobre todo si trabajamos con ítems
- El procesado de la matriz es lento

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
a	1		4	5			4	3						2		4	2								
b		4							3							5	1	3							
c	5		4		4							3		5					4	5					
d							3				5				3		4	2							3
e	3					5		4	5					5				1		5	4				
f		4				1		3	5		4	1		5	4	4	4							3	
g	2	4			4	2			5		1	4	5		4	2	4	5		5					4
h		2		1		4		3	5		4	2		5	4	5									5
i	1				3			5					5		4	4	5		4		3				
j		4			4				5		1		5		4	4									4
k	5				4		2		5		1	5		4	2	4									2
l			3		3				4	1		4		4	2	4									3
m	5	3			5	3		5	4		5	5	3		4	4	5	4	4						4
n	1		4	5				4	5		1	5		4	3	4	4	4	3						
o		4		4				5	4		5			4	2	5	5	5	3						
p			4			5							5	4	2	4	4	5	4	2					
q			3			3					1	5		4	4	4			4		3				
r	4		1	4	2					2		5		4			5	4	4	5	4	4			
s	2		4	4			5			1		4		2	4	4	4	5							
t	1	4		3				4		5	5	5		4		4	4	4							3
u	2	1	4	3					1		5	4		2	4		5	4							
v			4	5				4	3	5			2			2			2						5
w		2		2	3				5		4	5		4	2	3	4	2	3	4					
x	4		5		3		3			4	5			3	3	5		1							
y		1			3		2	3						3	3	5	4								

1. Calcular los vecinos del usuario

La matriz se usa para calcular la similitud del usuario **u** con los demás usuarios

- Usar una matriz permite acceder fácilmente a cada una de las posiciones para calcular correlaciones/similitudes

Para cada usuario **u** se obtiene un ratio de afinidad (**ra**) con cada usuario del sistema

- Se seleccionan los usuarios de mayor afinidad o los que superan un cierto valor de afinidad

Ejemplo

Usar los ítems para calcular la similitud

- Se tienen usuarios de MovieLens que tienen películas almacenadas en su histórico (en su perfil de usuario)
- Se crea una matriz de similitudes con las películas puntuadas por cada usuario (las puntuaciones van del 0 al 5)
- Se usa la matriz para buscar similitudes

	0	1	2	...	1681
Usuario 1	0	5	3		1
Usuario 2	2	1	0		0
...					
Usuario n	0	4	5		4

items: Bloc de notas			
Archivo	Edición	Formato	Ver
Toy Story (1995)			Ayuda
GoldenEye (1995)			
Four Rooms (1995)			
Get Shorty (1995)			
Copycat (1995)			
Shanghai Triad (Yao a yao yao dao waipo qiao) (1995)			
Twelve Monkeys (1995)			
Babe (1995)			
Dead Man Walking (1995)			
Richard III (1995)			
Seven (Se7en) (1995)			
Usual Suspects, The (1995)			
Mighty Aphrodite (1995)			
Postino, Il (1994)			
Mr. Holland's Opus (1995)			
French Twist (Gazon maudit) (1995)			
From Dusk Till Dawn (1996)			
White Balloon, The (1995)			
Antonia's Line (1995)			
Angels and Insects (1995)			
Muppet Treasure Island (1996)			
Braveheart (1995)			
Taxi Driver (1976)			
Rumble in the Bronx (1995)			
Birdcage, The (1996)			

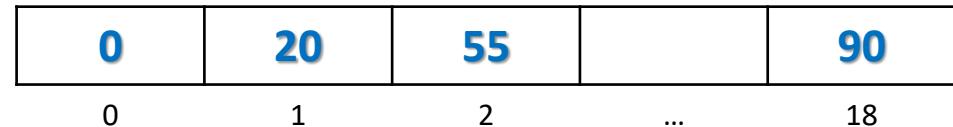
u1_base: Bloc de...			
Archivo	Edición	Formato	Ver
1	1	5	
1	2	3	
1	3	4	
1	4	3	
1	5	3	
1	7	4	
1	8	1	

Ejemplo

Usar las preferencias para calcular la similitud

- En la BD de MovieLens no existen preferencias del usuario
- Tendremos que crear las preferencias del usuario infiriéndolas de los ítems puntuados (en nuestro trabajo serían las preferencias basadas en contenido)
 - Qué ítems ha puntuado favorablemente
 - En qué géneros está clasificado ese ítem
- Las preferencias de un usuario es un vector, donde, por cada categoría de la BD se tiene el ratio de interés del usuario en la categoría

Preferencias
del usuario



genre: Bloc ...	
	Archivo
0	unknown
1	Action
2	Adventure
3	Animation
4	Children's
5	Comedy
6	Crime
7	Documentary
8	Drama
9	Fantasy
10	Film-Noir
11	Horror
12	Musical
13	Mystery
14	Romance
15	Sci-Fi
16	Thriller
17	War
18	Western

Ejemplo

Usar las preferencias para calcular la similitud

- Se tienen usuarios de MovieLens que tienen las siguientes preferencias almacenadas en su perfil de usuario
- Se crea una matriz de similitudes con las preferencias de cada usuario
- Se usa la matriz para buscar similitudes

	0	1	2	...	18
Usuario 1	0	10	40		70
Usuario 2	0	0	60		0
...					
Usuario n	0	20	30		90



1. Calcular los vecinos del usuario

Vecinos según preferencias

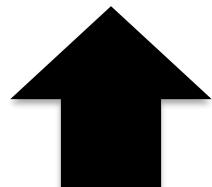
Computacionalmente menos costosa

Vecinos según ítems

La similitud entre vecinos es más exacta

Vecinos según características

Depende de los datos disponibles



1. Calcular los vecinos del usuario

La similitud se suele calcular con el coeficiente de correlación de **Pearson**

- Se usa Pearson por ser capaz de calcular similitudes y diferencias

Coeficiente de correlación

- Indica la situación relativa de los mismos sucesos respecto a la dos variables
- Expresión numérica del **grado de relación existente** entre dos variables y en qué medida se relacionan

Para cada usuario, se obtiene el **coeficiente de correlación de Pearson (r)** entre el usuario **u** y cada uno de los usuarios del sistema

$$\rho_{XY} = \frac{Cov(X, Y)}{\sqrt{Var(X) \cdot Var(Y)}}$$

1. Calcular los vecinos del usuario. Pearson

$r == 1$

- Correlación positiva perfecta
- Dependencia total directa entre las dos variables: cuando una de ellas aumenta, la otra también lo hace en proporción constante

$0 < r < 1$

- Correlación positiva
- A mayor valor, mayor correlación

$r == 0$

- No existe relación lineal

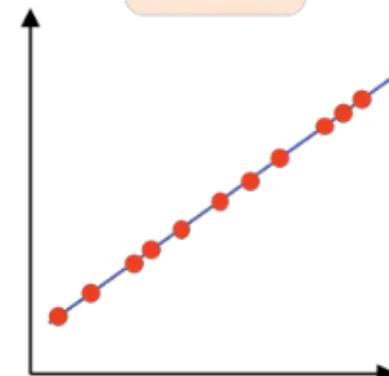
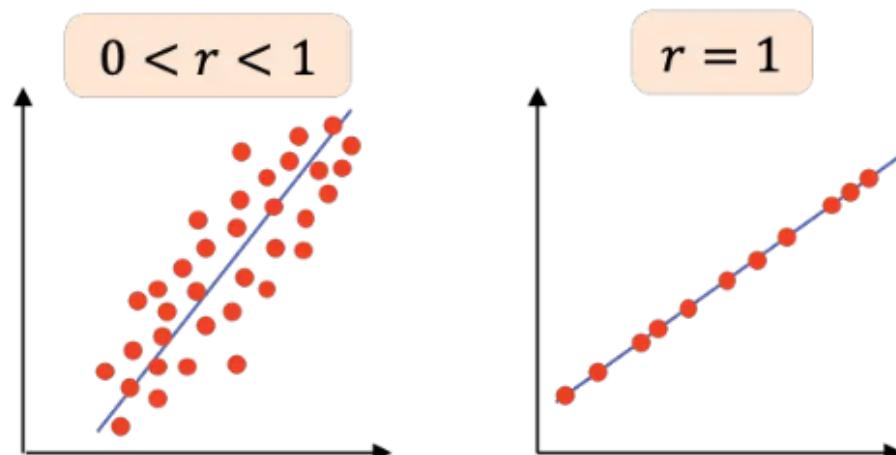
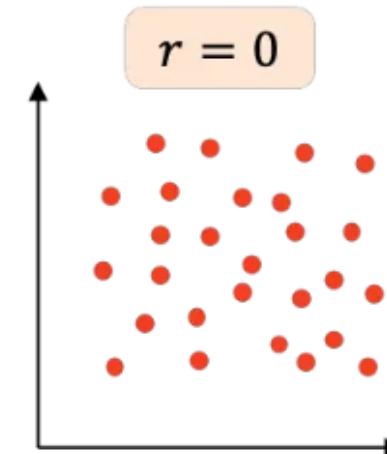
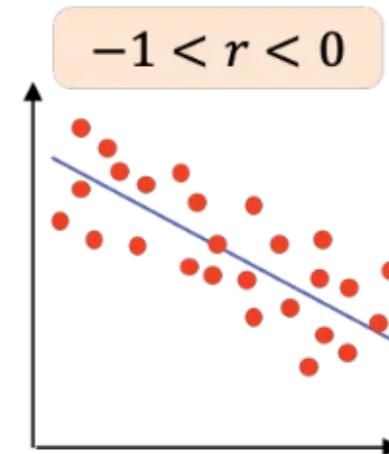
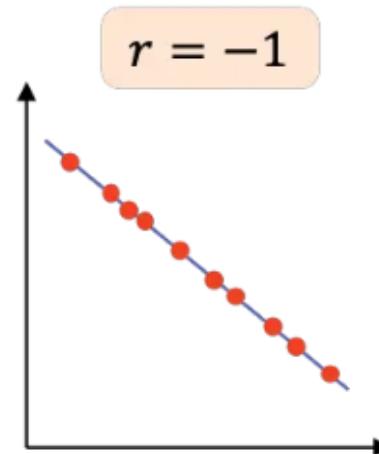
$-1 < r < 0$

- Correlación negativa

$r == -1$

- Correlación negativa perfecta
- Dependencia total inversa entre las dos variables: cuando una de ellas aumenta, la otra disminuye en proporción constante

1. Calcular los vecinos del usuario. Pearson



1. Calcular los vecinos del usuario. Pearson

Nota

- Pearson sólo puede usarse cuando los datos tienen una **distribución normal** o gaussiana (ocurrirá la mayor parte de las veces)
 - https://es.wikipedia.org/wiki/Distribuci%C3%B3n_normal
- Si no tienen esta distribución debería utilizarse **Spearman**
- A la hora de calcular una correlación para un recomendador real, se recomienda analizar antes los datos con algún sw específico (R, SPSS,...)
- <https://datatab.es/tutorial/correlation>

1. Calcular los vecinos del usuario

Proceso

- Antes de comenzar el cálculo de vecinos se debe decidir si se van a usar ítems, preferencias o características para calcular los vecinos (o mezcla)
- Depende de si disponemos o no de preferencias, o sólo de ítems
- Si disponemos de ambas cosas, depende del volumen de datos y del coste computacional

1. Calcular los vecinos del usuario: dispersión de ratios

Una decisión inicial a la hora de obtener el coeficiente de correlación de Pearson es como representar los valores que se van a comparar

Una posición a 0 en las filas de las matrices puede indicar:

- Que el ítem no ha sido puntuado por el usuario (similitud entre ítems)
- Que la preferencia no le interesa al usuario (similitud entre preferencias) o que no la ha puntuado

Hay que tomar un criterio sobre si se usan los 0 o no a la hora de calcular Pearson

- Depende del contexto
- Depende de la cantidad de ítems o preferencias que sean 0

1. Calcular los vecinos del usuario: dispersión de ratios

Matriz original

	I1	I2	I3	I4
U1	66		30	
U2		20	10	
...				

Rellenar a 0 y considerar todas las columnas

	I1	I2	I3	I4
U1	66	0	30	0
U2	0	20	10	0
...				

Usar sólo las columnas que ambos tienen llenas

	I1	I2	I3	I4
U1			30	
U2			10	
...				

1. Calcular los vecinos del usuario: dispersión de ratios

Dispersión de ratios

Cuando hay pocas preferencias o ítems comunes, Pearson puede no funcionar bien debido al tamaño de la muestra

Soluciones

Establecer un mínimo numero de valores comunes para para que un usuario pueda ser candidato a vecino

Utilizar técnicas que permitan inferir los valores que están a 0: Laplace, interpolación, machine learning,...

Usar similitud de usuarios o similitud de ítems en función de que se disponga de mayor cantidad de datos en una u otra

1. Calcular los vecinos del usuario: dispersión de ratios

Dispersión de ratios

Para calcular la similitud sería aconsejable que todos los valores que se comparan tengan un valor

Si usamos preferencias, puede que el usuario tenga un 0 en la preferencia porque no la ha llenado o porque no tenga interés en ella (usuarios con gran parte de las preferencias a 0 serán muy similares)

Si usamos ítems, el usuario puede no haber puntuado el elemento o puede que no le interese

Lo ideal sería usar sólo las preferencias o ítems que ambos han puntuado, pero esto, en muchos casos no da resultados válidos

Una posibilidad es realizar un proceso previo al cálculo de vecinos, se eliminan los usuarios que no tienen un mínimo numero de preferencias o ítems coincidentes

1. Calcular los vecinos del usuario: dispersión de ratios

Vecinos según preferencias

Cuando se calcula pearson usando preferencias (las columnas son las preferencias, es decir las posibles categorías de los ítems)

Es recomendable calcular Pearson usando todo el vector de preferencias, tengan o no la mayor parte de las posiciones a 0

Se tratan los 0 como un valor cualquiera

Es importante tener la coincidencia entre lo que le gusta y lo que no le gusta

1. Calcular los vecinos del usuario: dispersión de ratios

Vecinos según ítems

Cuando se calcula pearson usando ítems (las columnas son los ítems de la BD)

Es mejor usar sólo los ítems que ambos vectores tienen distinto de 0, es decir, los ítems que ambos usuarios han puntuado

De este modo, no se consideran los que ha puntuado un usuario y el otro no lo ha hecho, porque, que no haya puntuado un ítem, no significa que no le guste

Problema: dispersión de ratios

- Se puede solucionar filtrando los ítems que se utilizan usando algún criterio

1. Calcular los vecinos del usuario

Proceso (I)

- Obtener los ítems puntuados (o las preferencias) de los usuarios del sistema
 - **u**, usuario para el que se solicita la recomendación
 - **X**, vector de ítems o preferencias del usuario **u**
 - **u' ∈ U**, otro usuario (se repite el proceso para cada usuario)
 - **Y**, vector de ítems o preferencias del usuario **u'**
 - Calcular la varianza de los datos de ambos usuarios (s_x y s_y)
 - Aplicar la fórmula para **X** (preferencias o ítems del usuario **u**) y para **Y** (preferencias o ítems del usuario **u'**)
 - Calcular la covarianza de **X** e **Y**

$$Var(X) = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}$$

$$Var(Y) = \frac{\sum_{i=1}^n (y_i - \bar{y})^2}{n}$$

$$Cov(X, Y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n}$$

1. Calcular los vecinos del usuario

Proceso (II)

- Calcular el coeficiente de correlación de Pearson r entre los usuarios u y u' (cuyos vectores de ítems o preferencias son X e Y)
- Repetir el proceso con todos los usuarios $u' \in U$
- El ratio de afinidad con el vecino ra puede ser
 - $ra = r$
 - $ra = f(r)$

$$\rho_{XY} = \frac{Cov(X, Y)}{\sqrt{Var(X) \cdot Var(Y)}}$$

1. Calcular los vecinos del usuario

	0	1	2	...	18
X → Usuario 1	0	10	40		70
y → Usuario 2	0	0	60		0
...					
y → Usuario n	0	20	30		90

$$Var(X) = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}$$

$$Var(Y) = \frac{\sum_{i=1}^n (y_i - \bar{y})^2}{n}$$

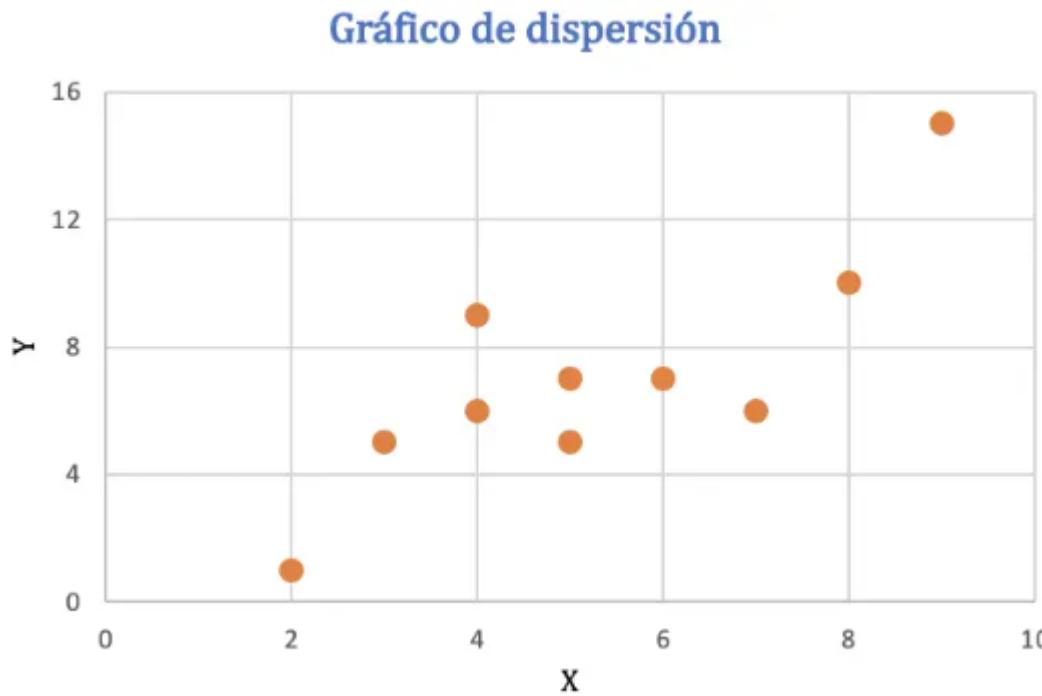
$$Cov(X, Y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n}$$

$$\rho_{XY} = \frac{Cov(X, Y)}{\sqrt{Var(X) \cdot Var(Y)}}$$

Ejemplo

Calcular el coeficiente de correlación de Pearson

x_i	y_i
3	5
5	7
2	1
4	6
9	15
8	10
6	7
5	5
7	6
4	9
53	
71	



$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n} = \frac{53}{10} = 5,3$$

$$\bar{y} = \frac{\sum_{i=1}^n y_i}{n} = \frac{71}{10} = 7,1$$

Ejemplo

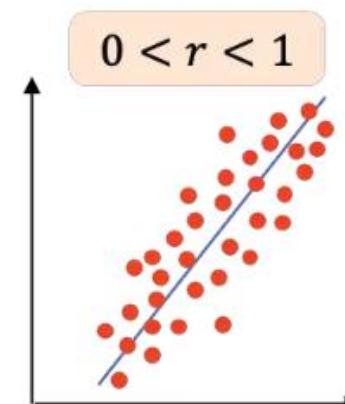
x_i	y_i	$x_i - \bar{x}$	$y_i - \bar{y}$	$(x_i - \bar{x})(y_i - \bar{y})$	$(x_i - \bar{x})^2$	$(y_i - \bar{y})^2$
3	5	-2,3	-2,1	4,83	5,29	4,41
5	7	-0,3	-0,1	0,03	0,09	0,01
2	1	-3,3	-6,1	20,13	10,89	37,21
4	6	-1,3	-1,1	1,43	1,69	1,21
9	15	3,7	7,9	29,23	13,69	62,41
8	10	2,7	2,9	7,83	7,29	8,41
6	7	0,7	-0,1	-0,07	0,49	0,01
5	5	-0,3	-2,1	0,63	0,09	4,41
7	6	1,7	-1,1	-1,87	2,89	1,21
4	9	-1,3	1,9	-2,47	1,69	3,61
53	71			59,7	44,1	122,9

$$Cov(X, Y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n} = \frac{59,7}{10} = 5,97$$

$$Var(X) = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n} = \frac{44,1}{10} = 4,41$$

$$Var(Y) = \frac{\sum_{i=1}^n (y_i - \bar{y})^2}{n} = \frac{122,9}{10} = 12,29$$

$$\rho_{XY} = \frac{Cov(X, Y)}{\sqrt{Var(X) \cdot Var(Y)}} = \frac{5,97}{\sqrt{4,41 \cdot 12,29}} = 0,81$$



1. Calcular los vecinos del usuario

Proceso (III)

Definir un valor mínimo de **ra** (δ), a partir del que se considerará un usuario como vecino

Obtener los usuarios que superen el umbral δ

Si no hay usuarios suficientes, se puede optar por seleccionar los **n** usuarios de mayor afinidad

Incluir a estos usuarios **u'** en la lista de vecinos

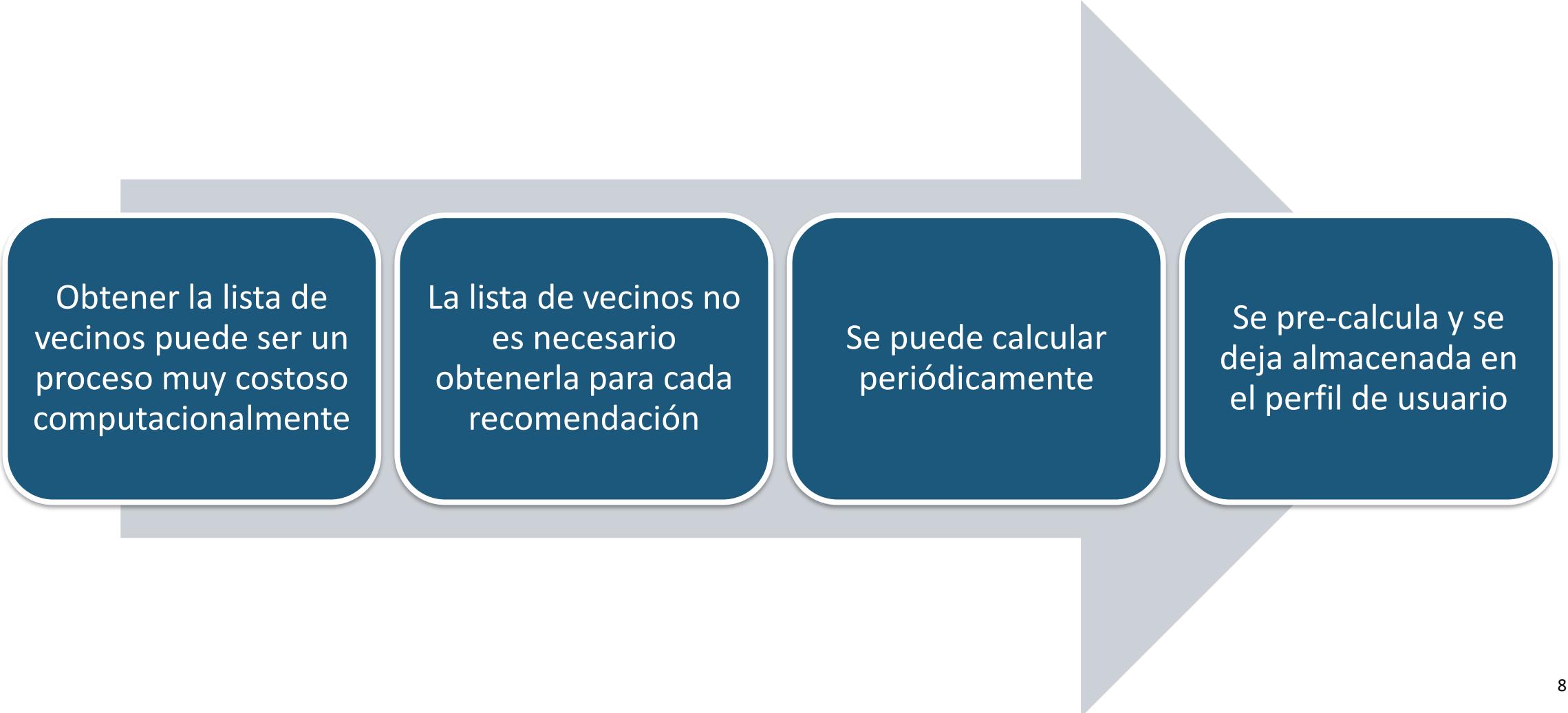
1. Calcular los vecinos del usuario

Lista de vecinos

$$L_u = \{(u', ra_{u'}) / u' \in U, ra_{u'} \in [0..100]\}$$

- u es el usuario para el que se solicita la recomendación
- $ra_{u'}$ es el ratio de afinidad entre u y u' . Puede ser un valor entre 0..100 o en el rango que se desee

1. Calcular los vecinos del usuario



Obtener la lista de vecinos puede ser un proceso muy costoso computacionalmente

La lista de vecinos no es necesario obtenerla para cada recomendación

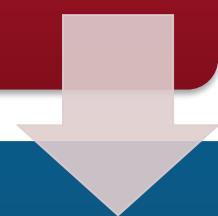
Se puede calcular periódicamente

Se pre-calcula y se deja almacenada en el perfil de usuario

2. Obtener la lista de ítems a recomendar

Recopilar todos los ítems puntuados favorablemente por los vecinos (eliminar repetidos)

- Eliminar los ítems puntuados por u



Calcular el ratio de interés ri de cada ítem para u

- Considerar de que usuario proviene el ítem (ratio de afinidad con el vecino)
- Considerar el ratio que el vecino dio al ítem
- Considerar si el ítem está puntuado por más de un vecino



Obtener una lista ordenada de ítems a recomendar

- Ordenada por ratio de interés del ítem para el usuario u : ri

2. Obtener la lista de ítems a recomendar

Calcular el ratio de interés ri de cada ítem para u

Es la clave del proceso de recomendación

Depende de varios factores y es importante ajustarla para cada sistema en concreto

No hay una fórmula a aplicar, supone entender como se puntúan los datos en cada sistema en concreto y como afectan elementos como: afinidades de usuarios, repeticiones de ítems,...

Ejemplo

Correlación de Pearson entre U1 y el resto de usuarios

	1	2	3	4	5	6	7	8	9	10	11
U1	3	5		2	1	5	10	4	8		
U2	4		6	9	1	6			8	5	
U3	7	10	7		3	7	1		8	6	
U4		2	10	4					8		
U5	10	1				9	8		8		
U6									1		

Ejemplo

Correlación de Pearson entre U1 y el resto de usuarios llenando a 0

	1	2	3	4	5	6	7	8	9	10	11	
U1	3	5	0	2	1	5	10	4	8	0	0	
U2	4	0	6	9	1	6	0	0	8	5	0	-0,08
U3	7	10	7	0	3	7	1	0	8	6	0	0,11
U4	0	2	10	4	0	0	0	0	8	0	0	-0,01
U5	10	1	0	0	0	9	8	0	8	0	0	0,68
U6	0	0	0	0	0	0	0	0	1	0	0	0,45

U6 sería un vecino, pero no parece muy afín. U5 también es vecino. La dispersión de ratios provoca afinidad

Ejemplo

Correlación de Pearson entre U1 y el resto de usuarios, considerando sólo las posiciones coincidentes (con datos)

	1	2	3	4	5	6	7	8	9	10	11	
U1	3	5		2	1	5	10	4	8			
U2	4	-	-	9	1	6	-	-	8	-	-	0,52
U3	7	10	-	-	3	7	1	-	8	6	-	-0,16
U4	-	2	-	4	-	-	-	-	8	-	-	0,65
U5	10	1	-	-	-	9	8	-	8	-	-	0,02
U6	-	-	-	-	-	-	-	-	1	-	-	0,00

Parece mas coherente que los vecinos serán U2 y U4

Ejemplo

Si se calcula el número de ratios coincidentes

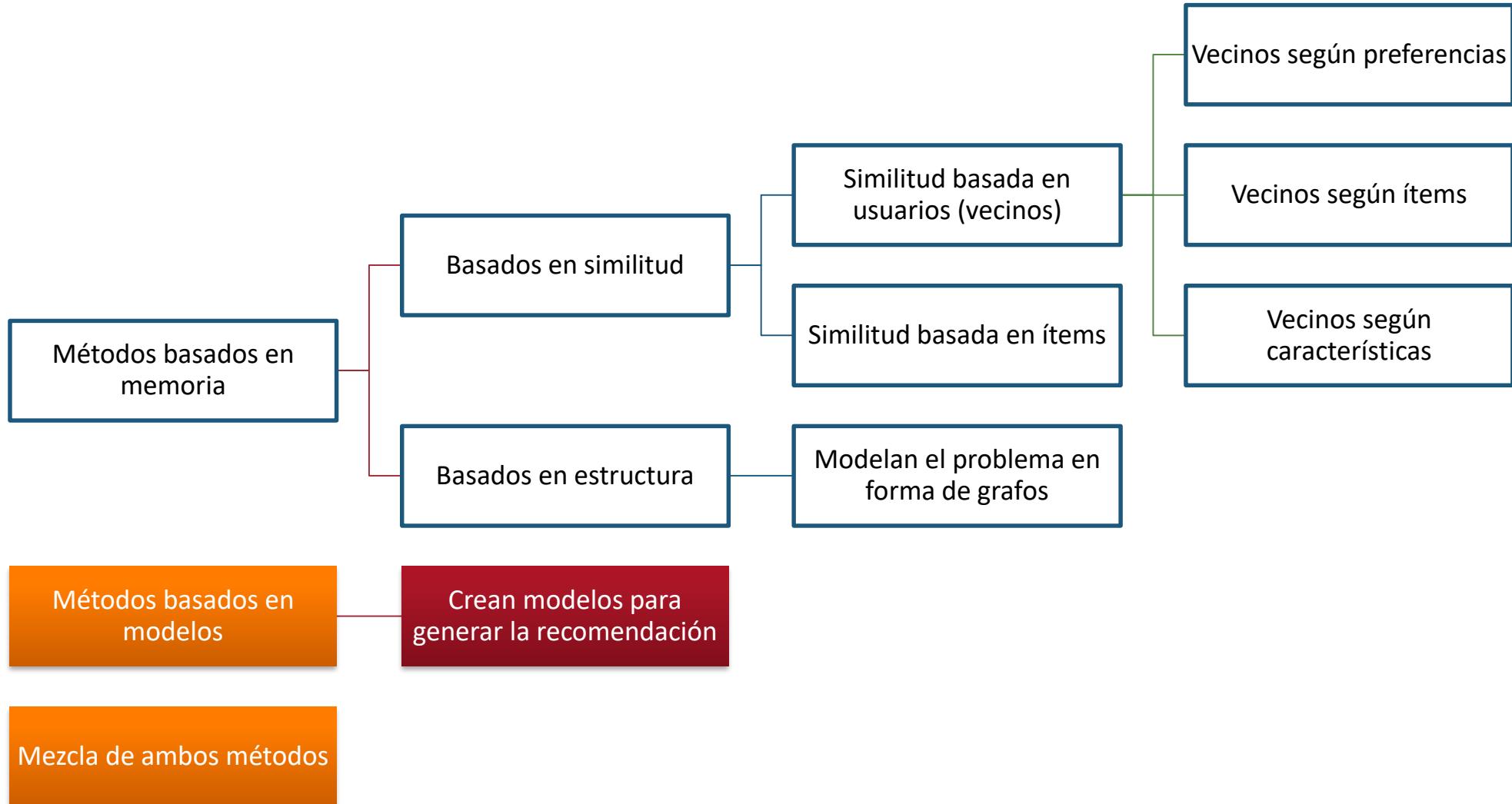
- Por ejemplo, asumir que el ratio es coincidente, si se diferencia en 2 unidades máximo, por encima o por debajo

Los vecinos serían U2 y U3

Usuarios	Puntuación de los ítems (ratios)											Coincidentes
	I1	I2	I3	I4	I5	I6	I7	I8	I9	I10	I11	
U1	3	5		2	1	5	10	4	8			
U2	4		6	9	1	6			8	5		4
U3	7	10	7		3	7	1		8	6		3
U4		2	10	4					8			2
U5	10	1				9	8		8			2
U6									1			0

SR colaborativo basado en modelos

Clasificación

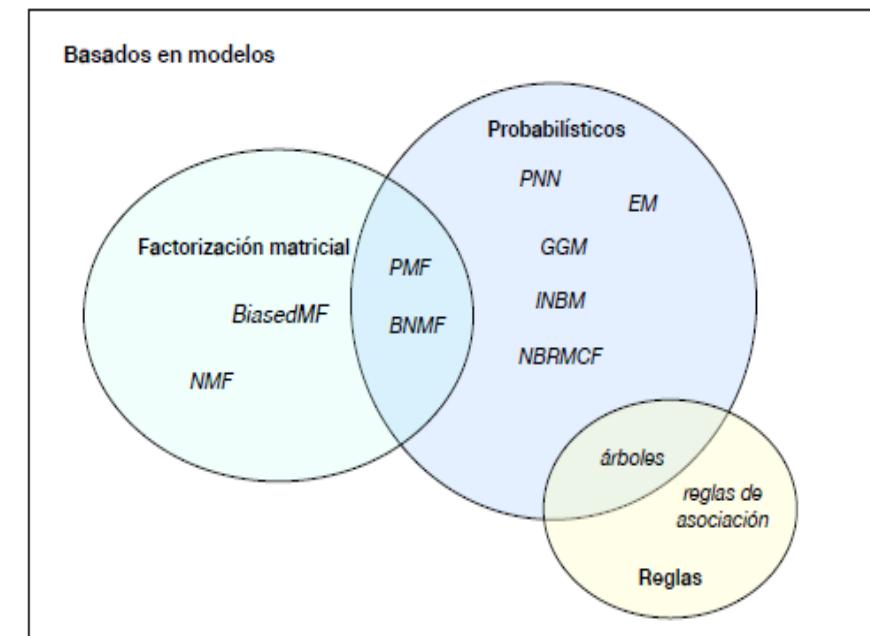


SR Colaborativos basados en modelos

Actualmente la implementación del filtrado colaborativo se realiza utilizando técnicas que crean un modelo de cómo se puntúan los datos

Existen técnicas de todo tipo

- Matrices de factorización o factores latentes
- Modelos probabilísticos
- Modelos basados en reglas
- Modelos basados en métodos difusos y algoritmos genéticos
- Machine learning



SR Colaborativos basados en modelos

Factores latentes

- Descomponen los datos en dos (o más) matrices: una hace referencia a los ítems y otra a los usuarios
- Usan factores para caracterizar a ambos
- La técnica más usada: factorización de matrices

Modelos probabilísticos

- Basados en el Teorema de Bayes
- Infieren la recomendación usando un modelo y cierto conocimiento previo
- Una posible implementación de un modelo probabilístico es usar Redes Neuronales

Modelos basados en reglas

- Modelos basados en árboles
- Modelos basados en reglas de asociación entre los datos

Modelos basados en métodos difusos y algoritmos genéticos

Factores latentes

Caracteriza cada ítem y cada usuario con factores

Los factores se infieren de

- Patrones de puntuación de un usuario a los ítems
- Patrones de puntuación de los usuarios a un ítem

Se utiliza la correspondencia entre los ítems, los usuarios y los factores para recomendar

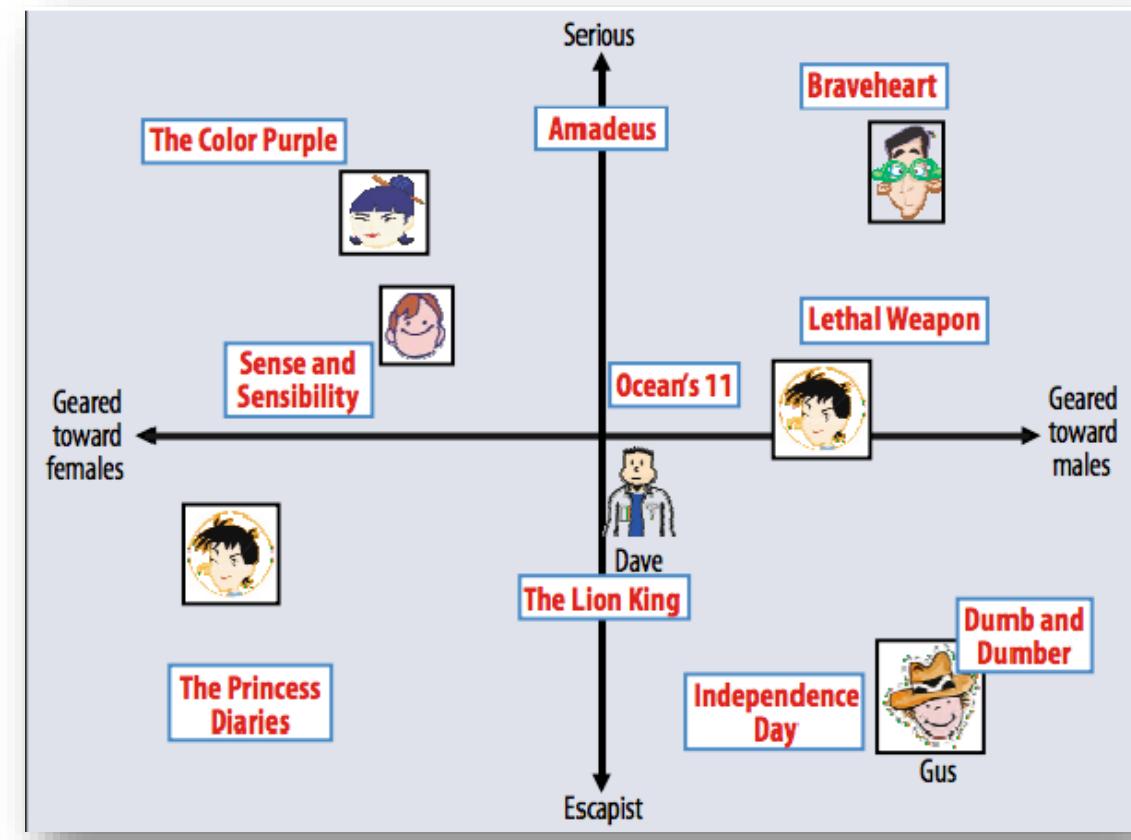
La ventaja de esta técnica es que permite añadir información adicional tanto a los usuarios como a los ítems a recomendar

Factores latentes

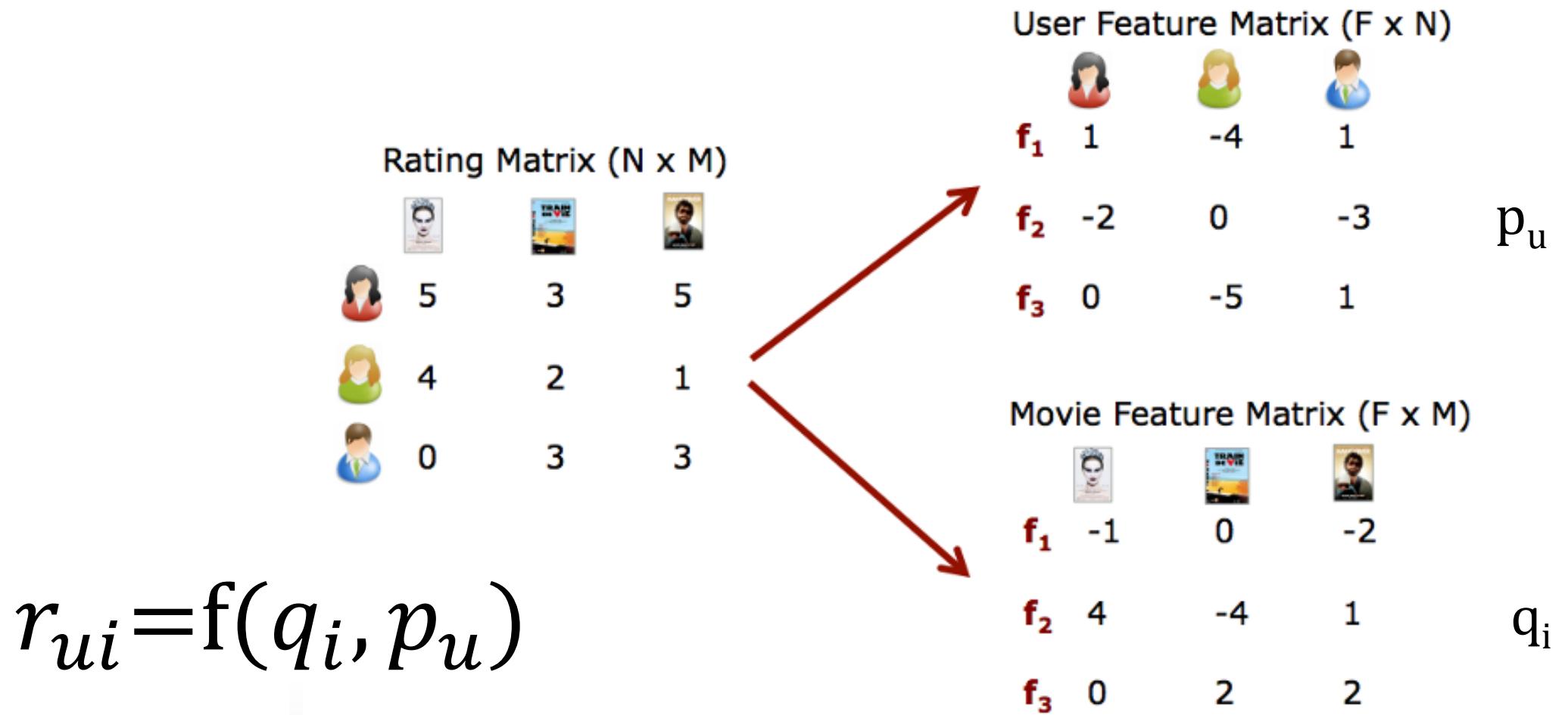
Una de las técnicas basadas en factores que se ha aplicado a la recomendación con mayor éxito es la **factorización de matrices** (Matrix Factorization)

Caracteriza los ítems y los usuarios mediante vectores de factores inferidos de sus patrones de puntuación a los ítems

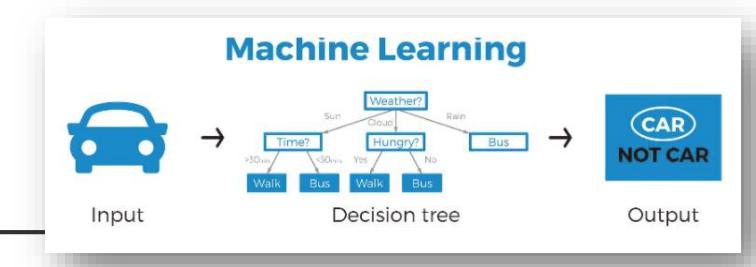
Se utiliza la correspondencia entre los ítems y los factores para recomendar ítems



Factores latentes



Machine Learning



El Machine Learning es una disciplina que hace predicciones en base a preguntas a los datos

- La pregunta que debe responder en este caso es: este usuario al que tengo que hacer una recomendación y que tiene estas características, este comportamiento y que ha calificado previamente estos ítems, ¿qué calificación le daría a este ítem que tiene estas características?

La respuesta del filtro (que hemos entrenado con los datos de los usuarios del sistema) nos daría un ratio de interés del usuario en el ítem, basado en las calificaciones que les han dado otros usuarios que se parecen a él

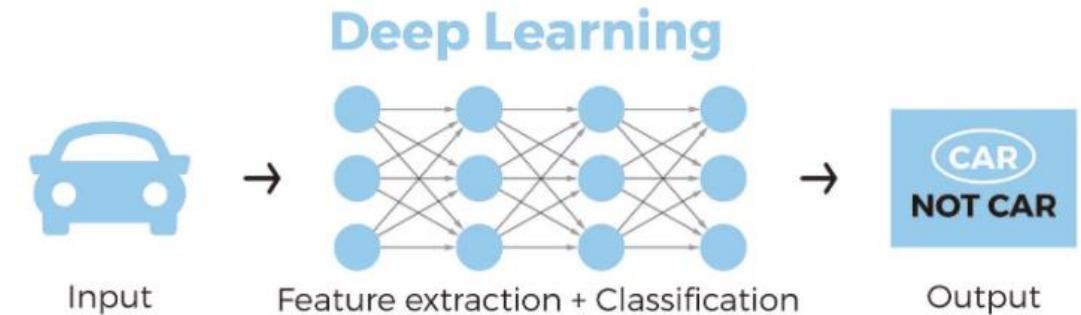
La pregunta se debe repetir con todos los ítems y se obtendrá la predicción de las calificaciones de todos ellos

De todas las respuestas, los ítems que obtengan mejor puntuación serán los recomendados al usuario

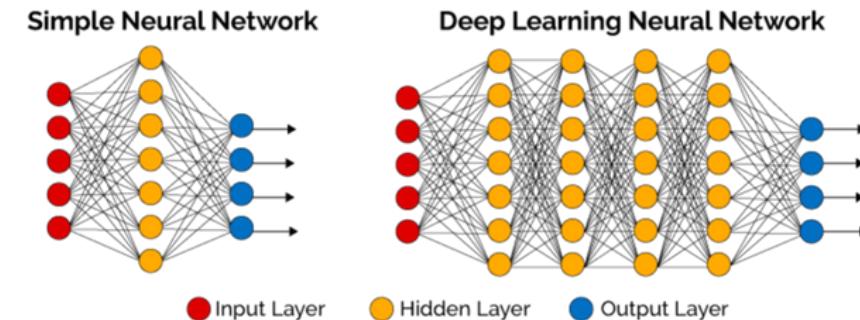
Utiliza algoritmos basados en árboles de decisión y algoritmos de regresión

Deep Learning

El deep learning como concepto es muy similar al machine learning, pero usa algoritmos distintos



Deep learning usa redes neuronales que funcionan de forma muy parecida a las conexiones neuronales biológicas de nuestro cerebro



Ventajas e inconvenientes de los SR colaborativos

Ventajas e inconvenientes

Ventajas

- Independiente de la representación de los ítems a recomendar
- Trabaja bien con objetos complejos (donde la recomendación depende de un número elevado de factores)
- Novedad en la recomendación
 - Los ítems recomendados no siempre son del tipo de los ya vistos por el usuario
- Útil cuando hay una gran cantidad de ítems puntuados, tanto por el propio usuario como por otros usuarios

Ventajas e inconvenientes

Inconvenientes

- Trabaja mal cuando hay pocos datos
 - Pocos datos del usuario actual o pocos datos de otros usuarios
- El número de usuarios en el sistema debe ser alto (problema de cold start)
- Los usuarios del sistema deberían haber puntuado ítems similares a los del usuario actual, si no es así la matriz de ratios será muy dispersa (sparsity o dispersión de ratios)
- No puede dar una recomendación si el usuario es:
 - Nuevo en el sistema
 - Tiene gustos extraños (recomienda ítems “populares”)
- No puede recomendar un ítem que no ha sido puntuado nunca (aunque se puede variar ligeramente esta técnica para que si lo haga)
 - Ítem nuevo
 - Ítem extraño

Ventajas e inconvenientes

Existe una variante del SR colaborativo de vecinos que resuelve el problema del nuevo ítem

Recomienda ítems similares a los ítems de los vecinos (no recomienda directamente los propios ítems de los vecinos)

De esta forma, un nuevo ítem, similar a un ítem del vecino, podría ser recomendado

Problema añadido:
qué es un ítem similar? Similitud cuestionable...

Ventajas e inconvenientes

Nuevo usuario

Nuevo ítem

Sparsity, dispersión
de ratios, “oveja
negra”

Falta de novedad

Si

Si

Si

No

Aunque presenta todos esos problemas es
una técnica muy precisa y la más utilizada.

Funciona mal en casos concretos

Estado actual

Investigación en SR colaborativos

Nuevas implementaciones del SR colaborativo usando distintas técnicas de IA

Sensibilidad al cambio de preferencias

- Ajuste de las recomendaciones a medida que cambian las preferencias y comportamientos del usuario

SR colaborativo sensible al contexto

- Consideran el contexto en el que se realiza una recomendación
- Esto puede incluir factores temporales, geográficos o situacionales

Evaluación continua

- El sistema se evalúa constantemente durante su funcionamiento
- Permite ajustar el comportamiento del sistema de forma automática

Investigación en SR colaborativos

Incremento de la
precisión y/o utilidad de
la recomendación

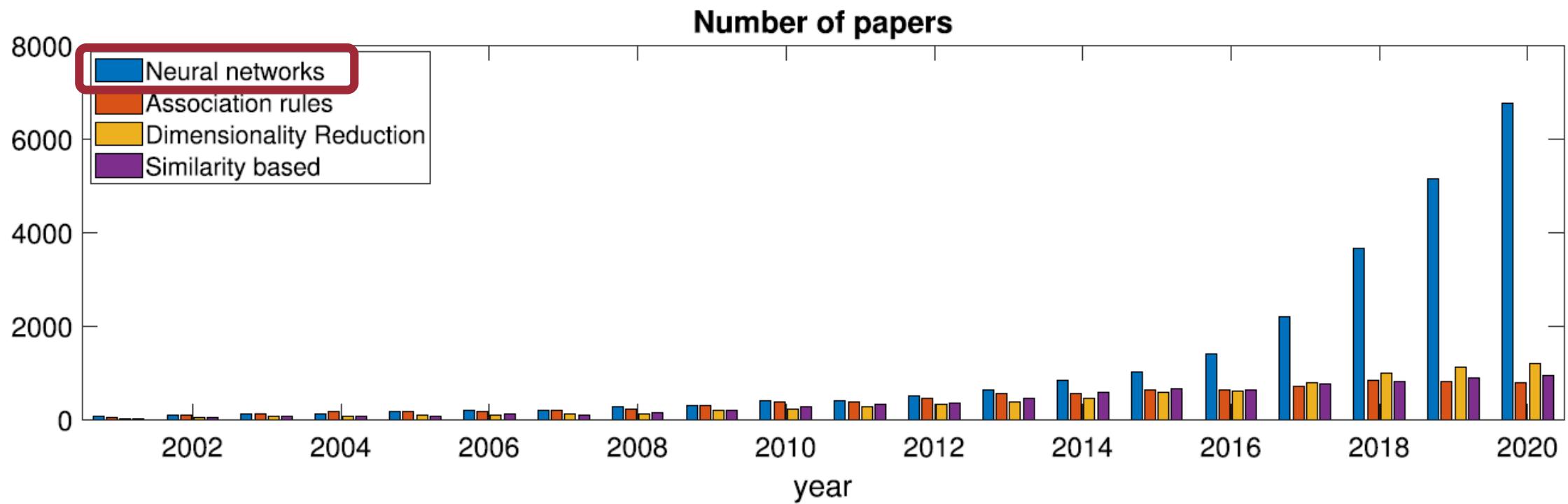
Problemas del SR
colaborativo

- Dispersión de ratios
- Cold start

Uso de factores
complejos (obtenidos
del comportamiento del
usuario/otros usuarios)

Futuro

Investigación en SR colaborativos



**Gracias por vuestra
atención...**