

Sistemas Recomendadores

IIC-3633

Introducción

Esta clase

1. Actividades principales del semestre: Tarea , Presentaciones de Papers y Proyecto Final
2. Definición y un poco de historia
3. Algoritmos de recomendación

Evaluación

- 1 tarea
- prácticos
- lectura de papers de materia
- presentación papers (seminario)
- avances de proyecto (3 entregas)
- proyecto final

Reglas de evaluaciones:

https://docs.google.com/presentation/d/1zjBLBUQnjRfWbx_djy4zf9wstK4vED7O/edit?usp=sharing&ouid=103607418722343842699&rtpof=true&sd=true

Tarea (recomendación multimodal)

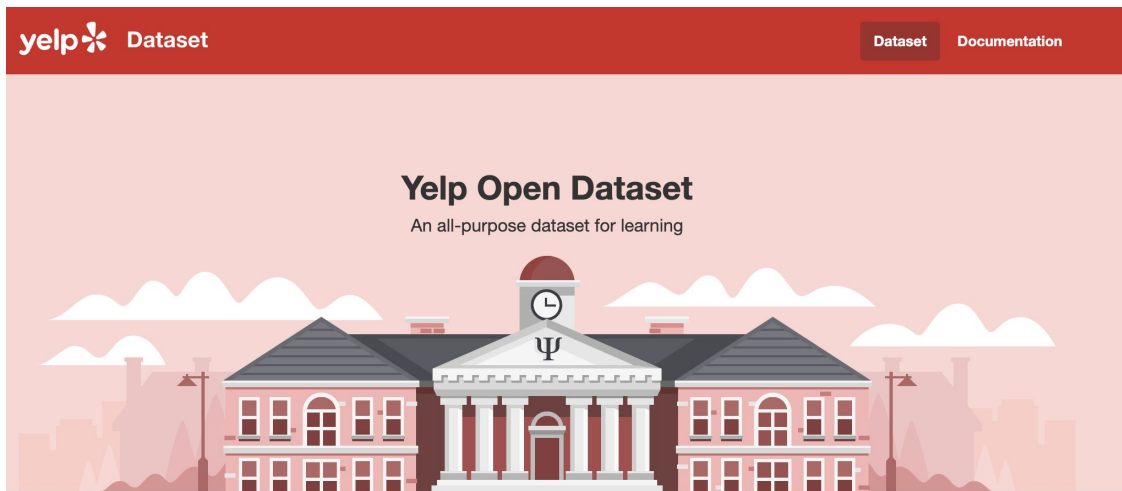
~100,000 interacciones
usuario-item

~9,800 usuarios

~9,500 negocios

100,000 fotos de negocios

metadata de los negocios
(estado, categoria, nombre)



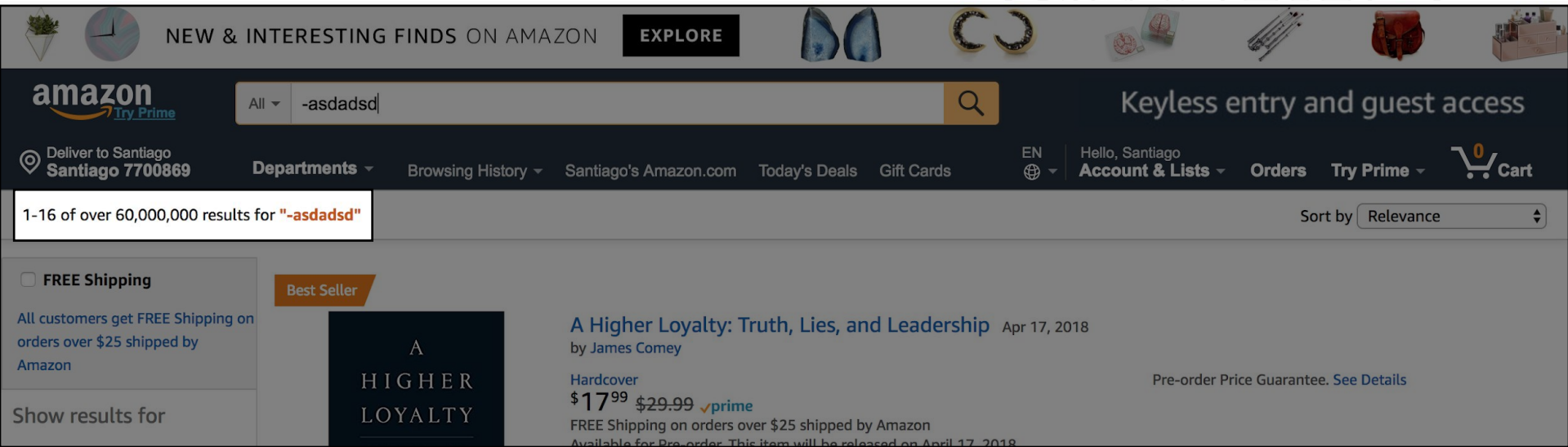
Calendario (provisorio)

	Semana	Sala	Ayudantia	Martes	Jueves	Lecturas	Enunciados	Deadlines alumnos
I	7-ago 11-ago	B24 / B17		Intro	No personalizado + UB CF	CF		
II	14-ago 18-ago	B24 / B17		FERIADO	IB CF			
III	21-ago 25-ago	B24 / B17	surprise	FunkSVD	MF - ALS	MF	Tarea 1 (Ju 25 agosto)	
IV	28-ago 1-sept	B24 / B17		MF - BPR	Evaluacion I	Evaluation		
V	4-sept 8-sept	B24 / B17	implicit feedback	Evaluacion II	Content based I	Content-based		
VI	11-sept 15-sept	B24 / B17		Content based II	Content based III	Context-based		T1: Viernes 15 de Sept
VII	18-sept 22-sept	B24 / B17		FERIADO	Contextual recommendation			
VIII	25-sept 29-sept	B24 / B17		Ensambls	Ensambls + context	Deep learning	Enunciado Proy final (Lu 25 sept)	
IX	2-oct 6-oct	B24 / B17	ensambls fastFM	Deep learning I	Deep learning II			
X	9-oct 13-oct	B24 / B17		User centric recommendation + ideas proyecto	Fairness accountability and ethics in recsys + ideas proyecto	Explainable recommendation		Propuestas proyecto (Vi 13 oct)
XI	16-oct 20-oct	B24 / B17	MAB wiser	Reinforcement learning	Futuro de sistemas recomendadores			
XII	23-oct 27-oct	B24 / B17		Presentaciones paper seminario	Presentaciones paper seminario		feedback propuestas	
XIII	30-oct 3-nov	B24 / B17		Presentaciones paper seminario	Presentaciones paper seminario			Avance proyecto (Vi 3 nov)
XIV	6-nov 10-nov	B24 / B17		Presentaciones paper seminario	Presentaciones paper seminario			
XV	13-nov 17-nov	B24 / B17		Presentaciones paper seminario	Presentaciones paper seminario		feedback avance	
XVI	20-nov 24-nov	B24 / B17		Presentaciones paper seminario	Presentaciones paper seminario			
XVII	27-nov 1-dic	B24 / B17		LIBRE proyecto final / presentaciones	LIBRE proyecto final / presentaciones			
XVIII	4-dic 8-dic	B24 / B17		Examen LUNES 4 DICIEMBRE				Presentacion final posters

Objetivos de sistemas recomendadores

- Ayudar al usuario con la sobrecarga de la información y que se ajuste a sus preferencias.
- Muchas alternativas , tiempo limitado para escoger.

Amazon: + de 60.000.000 de productos para escoger.



The screenshot displays the Amazon homepage with a search bar containing the text "-asdadsd". Below the search bar, a notification box states "1-16 of over 60,000,000 results for '-asdadsd'". The main content area features a book titled "A Higher Loyalty: Truth, Lies, and Leadership" by James Comey, published on April 17, 2018. The book is available in hardcover for \$17.99, down from \$29.99, and is eligible for Prime shipping. The page also includes a "FREE Shipping" banner and a "Show results for" section.

NEW & INTERESTING FINDS ON AMAZON EXPLORE

amazon Try Prime

All ▾ -asdadsd 🔍

Keyless entry and guest access

Deliver to Santiago
Santiago 7700869

Departments ▾ Browsing History ▾ Santiago's Amazon.com Today's Deals Gift Cards

EN ▾ Hello, Santiago
Account & Lists ▾ Orders Try Prime ▾ Cart

1-16 of over 60,000,000 results for "-asdadsd" Sort by Relevance ▾

☐ FREE Shipping

All customers get FREE Shipping on orders over \$25 shipped by Amazon

Show results for

Best Seller

A
HIGHER
LOYALTY

A Higher Loyalty: Truth, Lies, and Leadership Apr 17, 2018
by James Comey

Hardcover
\$17.99 \$29.99 ✓prime

FREE Shipping on orders over \$25 shipped by Amazon
Available for Pre-order. This item will be released on April 17, 2018.

Pre-order Price Guarantee. See Details

Problema de recomendación

- Seleccionar un conjunto de ítems para un usuario C que maximicen su utilidad.
- La función de utilidad genera una predicción de **rating**

$$\forall c \in C, s'_c = \arg \max_{s \in S} u(c, s)$$

$u : C \times S \rightarrow R$, función de utilidad

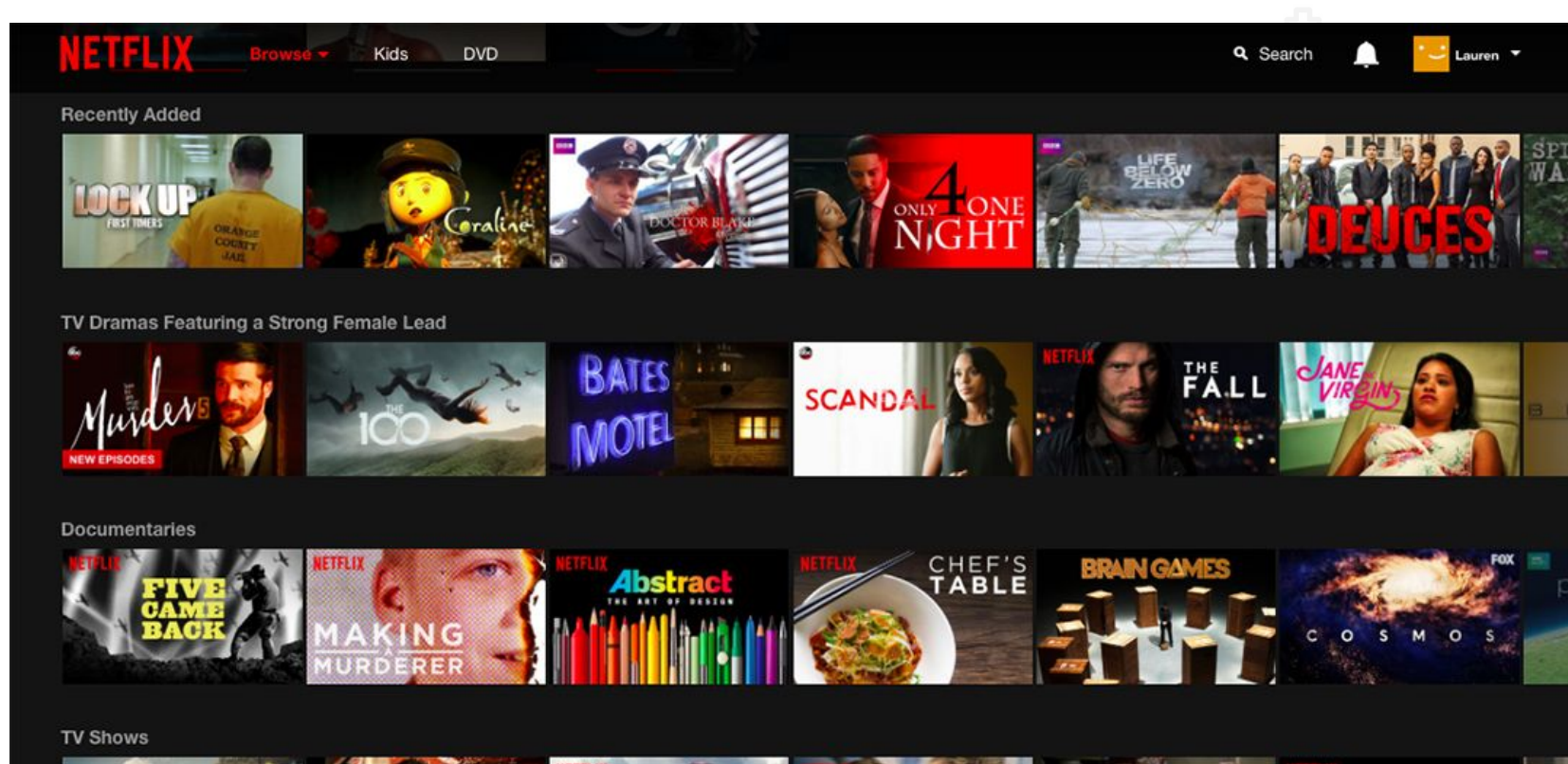
R : conjunto de ítems recomendados

C : conjunto de usuarios

S : conjunto de ítems

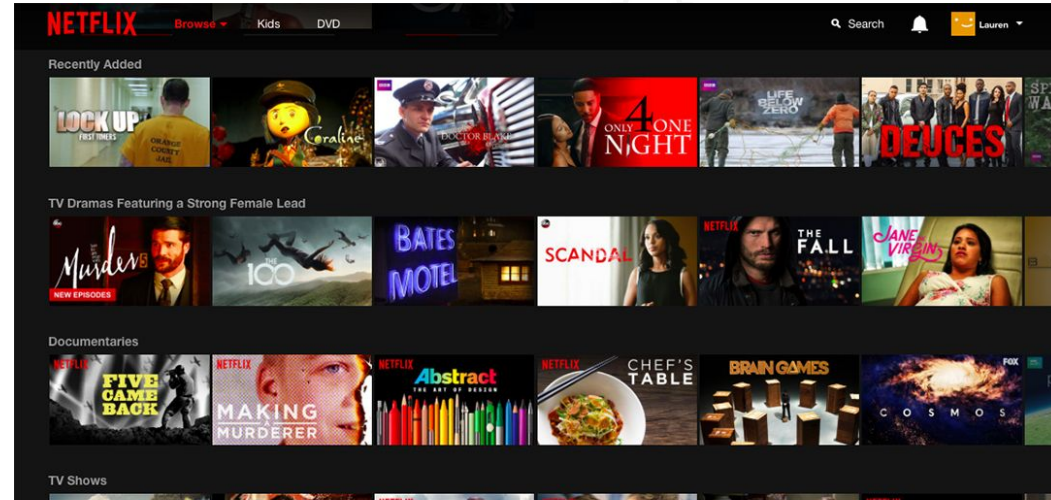


¿Dónde entran los algoritmos de recomendación en este caso?



¿Dónde entran los algoritmos de recomendación en este caso?

1. Ordenar los géneros de las películas dependiendo de mis intereses.
2. Dentro de cada género ordenar de ese género cuál es la que más me puede interesar.
3. Cambiar la carátula de la imagen dependiendo de mis intereses.
4. Más populares , más vistas en Chile.



Recomendaciones no personalizadas

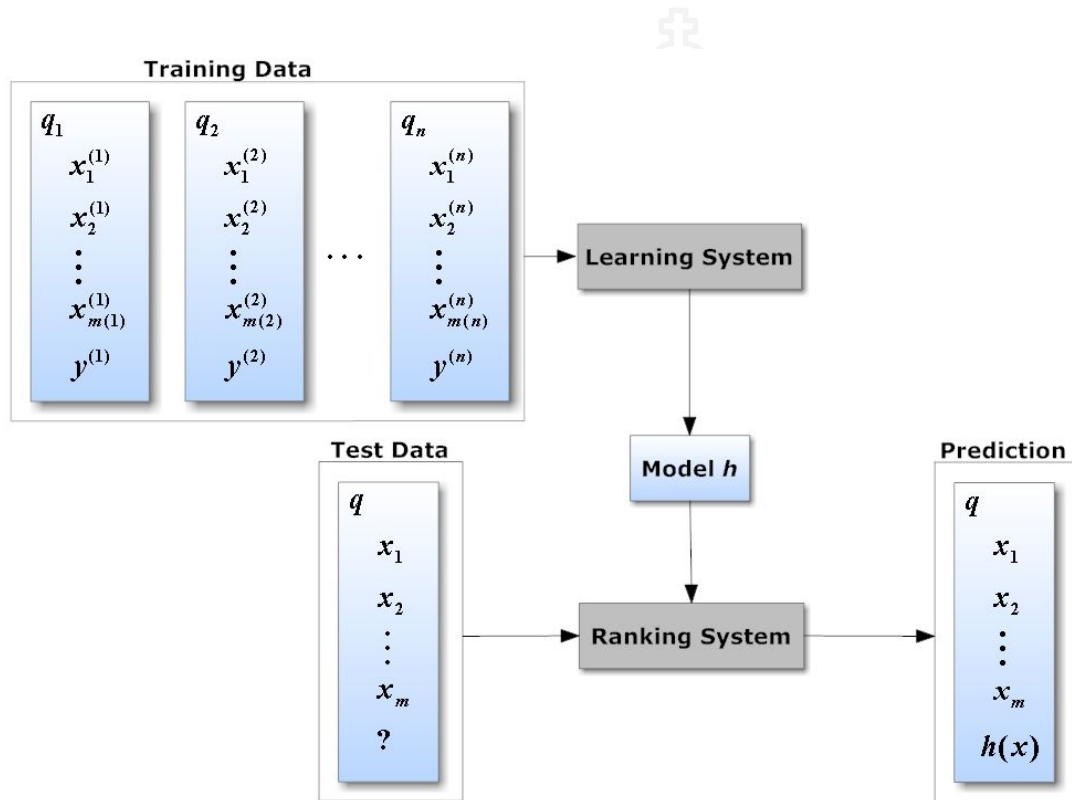
Recomendación que no toma información del usuario.



Learning to rank (no personalizado)

Learning to rank es una tarea dentro de Machine Learning que:

Dada una **query** y un **conjunto de items candidatos**, el objetivo es aprender a **ordenar** los que son **más relevantes** para la query.

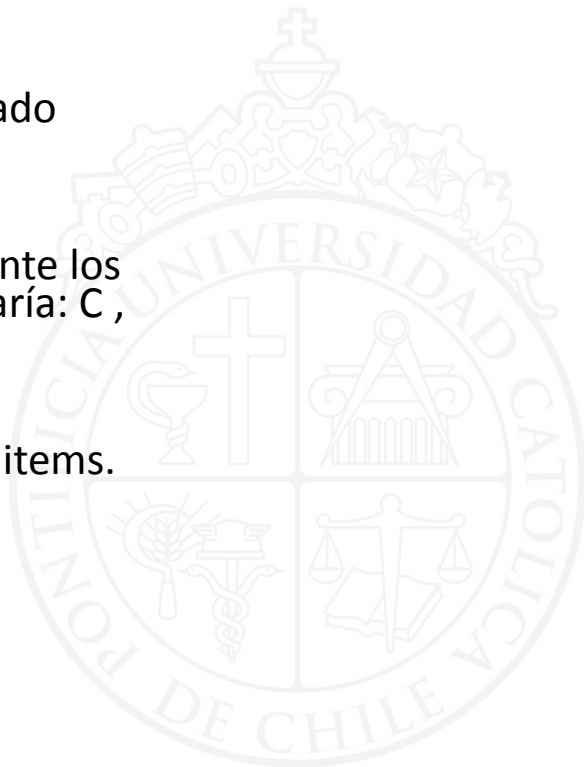


Learning to rank

POINTWISE: Obtiene un score de relevancia de cada resultado dada una QUERY y los ordena de mayor a menor.

PAIRWISE: Compara relevancia en pares de items y finalmente los ordena por transitividad. Ej. $A > B$ y $A < C$, entonces resultaría: C, A, B

LISTWISE: Compara relevancia de listas de un tamaño N de items.
Ej. $A > C > B$ y $C > D > F \rightarrow A, C, B, D, F$



Recomendación personalizada

- Considera información del usuario:
 - Ítems con los que ha interactuado históricamente (rating, compra/reproducción, like, etc)
 - Información de su perfil
 - Información contextual (ubicación , día , hora, etc..)



Predicción de Ratings

- Los algoritmos que veremos el resto de esta clase simplifican el problema de recomendación considerándolo como un problema de predicción de ratings basado en coocurrencia :

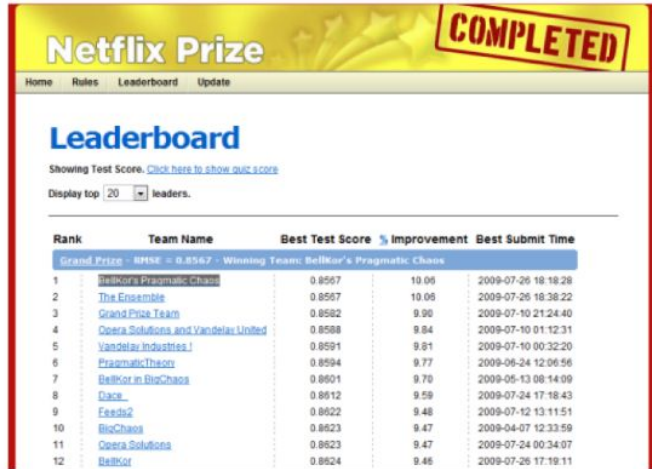
Predict!

	Item 1	Item 2	...	Item m
User 1	1	5		4
User 2	5	1		?
...				
User n	2	5		?

- ... y qué tan buena es mi predicción?

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (\hat{r}_{ui} - r_{ui})^2}{n}}$$

Ejemplo: Netflix Prize (2007 - 2009)



Rank	Team Name	Best Test Score	% Improvement	Best Submit Time
Grand Prize - RMSE = 0.8567 - Winning Team: BellKor's Pragmatic Chaos				
1	BellKor's Pragmatic Chaos	0.8567	10.06	2009-07-26 18:18:28
2	The Ensemble	0.8567	10.06	2009-07-26 18:38:22
3	Grand Prize Team	0.8582	9.90	2009-07-10 21:24:40
4	Covers Solutions and Vandelay United	0.8588	9.84	2009-07-10 01:12:31
5	Vandelay Industries!	0.8591	9.81	2009-07-10 00:32:20
6	Pragmatic Theory	0.8594	9.77	2009-06-24 12:06:56
7	BellKor in BigChaos	0.8601	9.70	2009-05-13 08:14:09
8	Coca...	0.8612	9.59	2009-07-24 17:18:43
9	Feedz	0.8622	9.48	2009-07-12 13:11:51
10	BioChaos	0.8623	9.47	2009-04-07 12:33:59
11	Covers Solutions	0.8623	9.47	2009-07-24 00:34:07
12	BellKor	0.8624	9.46	2009-07-26 17:19:11

¿En qué consistía?

Predecir ratings de Netflix (puerta a puerta) y disminuir en al menos un 10% el RMSE del modelo ya existente (MovieMatch)

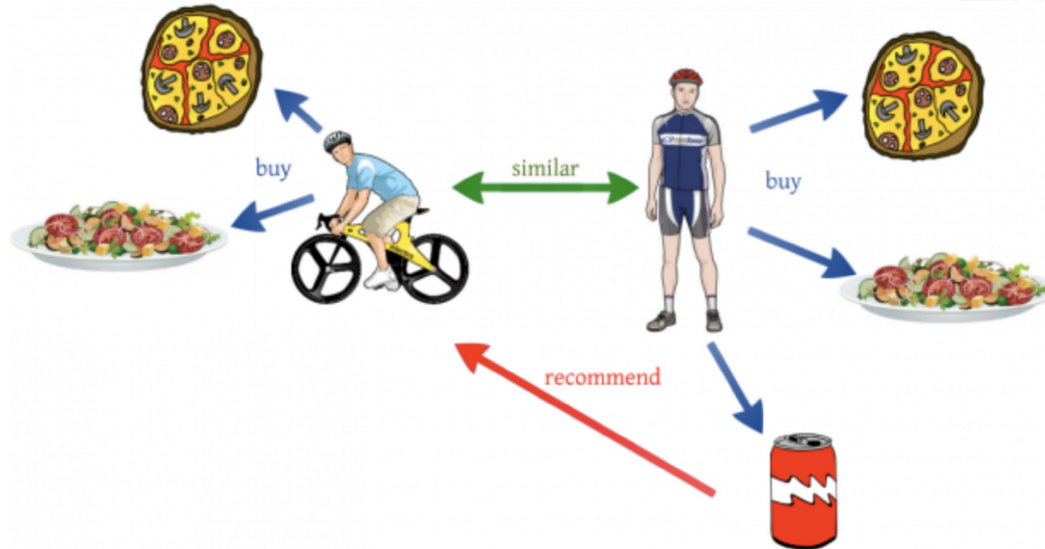
Rank	Team Name	Best Test Score	% Improvement	Best Submit Time
Grand Prize - RMSE = 0.8567 - Winning Team: BellKor's Pragmatic Chaos				
1	BellKor's Pragmatic Chaos	0.8567	10.06	2009-07-26 18:18:28
2	The Ensemble	0.8567	10.06	2009-07-26 18:38:22
3	Grand Prize Team	0.8582	9.90	2009-07-10 21:24:40



1 millón de
dólares
20 minutos
antes!!

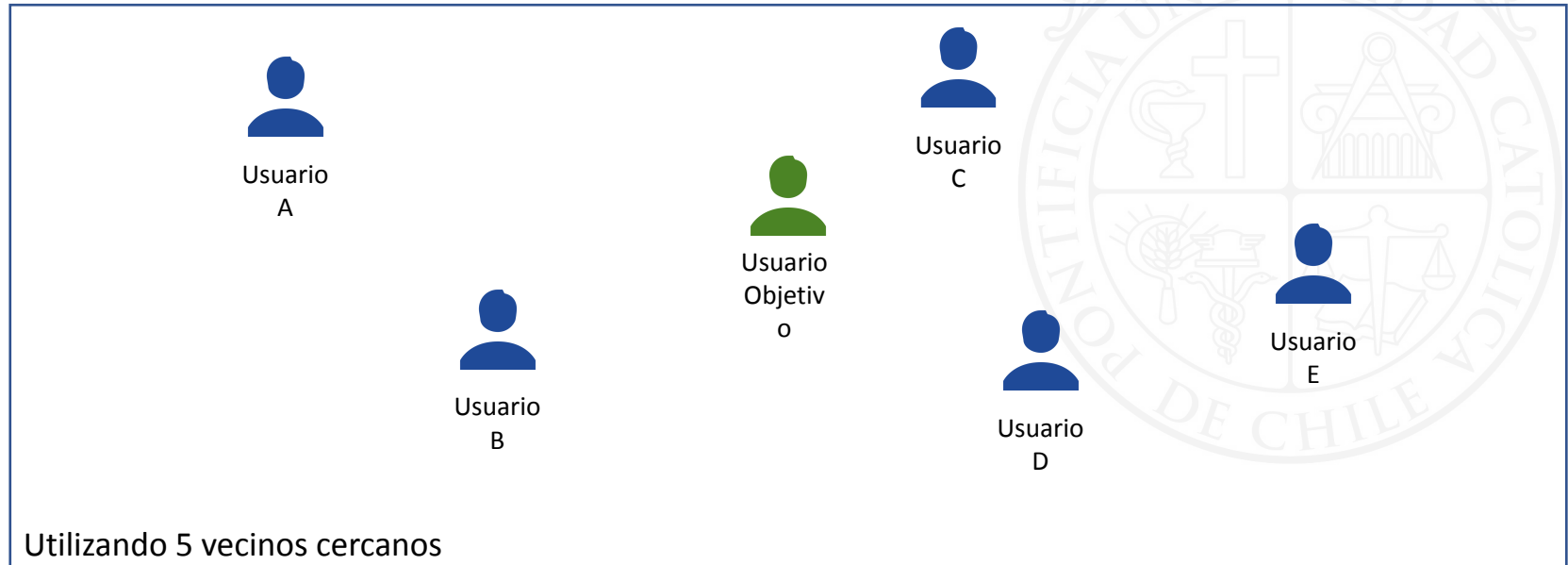
Filtrado Colaborativo basado en Usuarios

- **Objetivo:** buscar a usuarios similares y recomendar usando una suma ponderada con una métrica de similaridad



Filtrado Colaborativo basado en Usuarios

- **Objetivo:** buscar a usuarios similares y recomendar usando una suma ponderada con una métrica de similaridad



Similaridad entre usuarios

- Similaridad entre usuarios: se basa en los patrones de calificación normalizando por el promedio de cada usuario y en el denominador por la norma de la desviaciones.

$$\text{sim}(u, v) = \frac{\sum_{i \in I} [(r(u, i) - \text{avg}(u)) \cdot (r(v, i) - \text{avg}(v))]}{\sqrt{\sum_{i \in I} (r(u, i) - \text{avg}(u))^2} \cdot \sqrt{\sum_{i \in I} (r(v, i) - \text{avg}(v))^2}}$$

- **sim(u, v)**: Similaridad entre usuarios u y v.
- **I**: Conjunto de ítems que ambos usuarios han calificado.
- **r(u, i) , r(v, i)**: Calificaciones del usuario u y v para el ítem i.
- **avg(u) , avg(v)**: Calificación promedio del usuario u y v.

Generación de recomendaciones

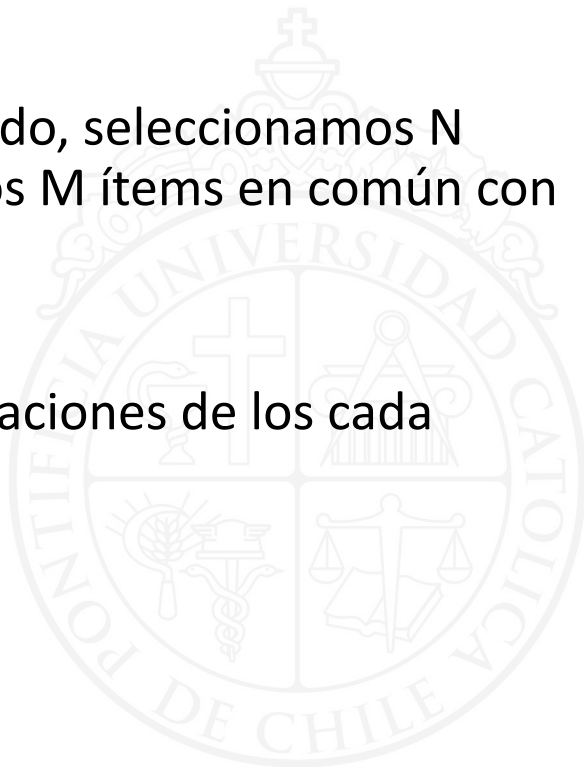
Objetivo: recomendar ítems que aún no han sido calificados por el usuario objetivo, pero que tienen calificaciones altas de usuarios similares.

$$P(u, i) = \text{avg}(u) + \frac{\sum_{v \in V} [\text{sim}(u, v) \cdot (r(v, i) - \text{avg}(v))]}{\sum_{v \in V} |\text{sim}(u, v)|}$$

- $P(u, i)$: rating predicho del ítem i para el usuario u .
- V : Conjunto de los N usuarios más similares a u que han calificado el ítem i .
- i : Los ítems considerados son solo aquellos que han sido calificados por los usuarios en V y no por el usuario u .
- $\text{sim}(u, v)$: Similaridad entre usuarios u y v .
- $r(v, i)$: Calificación del usuario v para el ítem i .
- $\text{avg}(v)$, $\text{avg}(u)$: Calificación promedio del usuario v y del usuario u .

Proceso Completo para Todos los Usuarios

- Para cada par de usuario e ítem no calificado, seleccionamos N usuarios similares que comparten al menos M ítems en común con el usuario objetivo.
- Usamos la similaridad para predecir calificaciones de los cada usuario con los ítems.



Generando la Lista de Recomendaciones

- Ordenamos los ítems que el usuario aún no ha calificado según los ratings predichos de mayor a menor.
- Seleccionamos los primeros K ítems para recomendar al usuario.



¿Qué deficiencias puede tener el filtrado colaborativo basado en usuarios?



Debilidades

- Alto costo computacional de recorrer todas las combinaciones de usuarios e item.
- Cold start problem: nuevos usuarios y nuevos items me obligan a tener que recalcular todo.

Soluciones: limitar el número de usuarios similares considerados, técnicas de factorización de matrices para reducir la dimensionalidad.