

Sistemas Recomendadores

IIC-3633

Evaluación de Sistemas Recomendadores
Parte 1

Esta clase

1. Comentarios Factorización Matricial.
2. Recomendación basada en contenido (intuición).
3. Partición en set de entrenamiento, validación y testing.
4. Evaluación de sistemas recomendadores con métricas de ranking.

Ayudantía y lecturas

Lunes: Ayudantía Feedback Implícito

Lectura: Evaluación de sistemas recomendadores








Comentarios Paper Matrix Factorization

LM

?

Al tener las funciones $b_i(t)$, $b_u(t)$ y $p_u(t)$ se podría analizar el comportamiento temporal de dichas funciones para realizar predicciones de si algun item en cierto periodo de tiempo sea mas probable que le guste a la gente o no. Como ejemplo básico, analizar si una película veraniega sea mas probable que le guste a la gente en periodos de verano que de invierno. De esta forma, se podrían tomar mejores conclusiones de cuándo liberar un ítem para que sea mas gustado.

✓



Aug 29 9:03 pm

Un approach interesante sería generar distintas matrices de preferencia dependiendo de la temporalidad de los datos , o añadir esta temporalidad al bias de usuarios e ítems.

Comentarios Paper Matrix Factorization

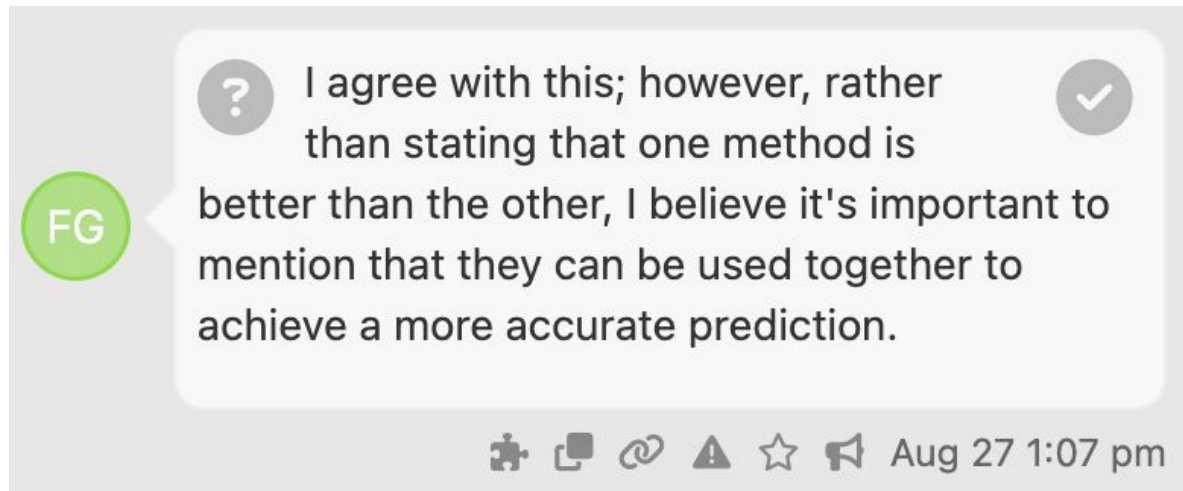
AR

? Una red neuronal es capaz de aprender los features relevantes que llevan a un usuario a puntuar un ítem como bueno o malo. Con esto quiero decir que las redes llegan a aprender qué características de las canciones hacen que nos gusten más o menos. Tener que identificar estas características relevantes podría ser imposible para un humano, ya que normalmente son características adquiridas por nuestro subconsciente. Por lo tanto, sería un punto a tener en cuenta que podría favorecer la mejora de los resultados, utilizar redes neuronales para aplicar un sistema de recomendación content based filtering. ✓

Esto es justamente la función de las redes neuronales para convertir contenido en vectores de características implícitas, no basta con usar el contenido raw.

Imágenes, textos, videos , etc... tienen que pasar por una Red para generar features.

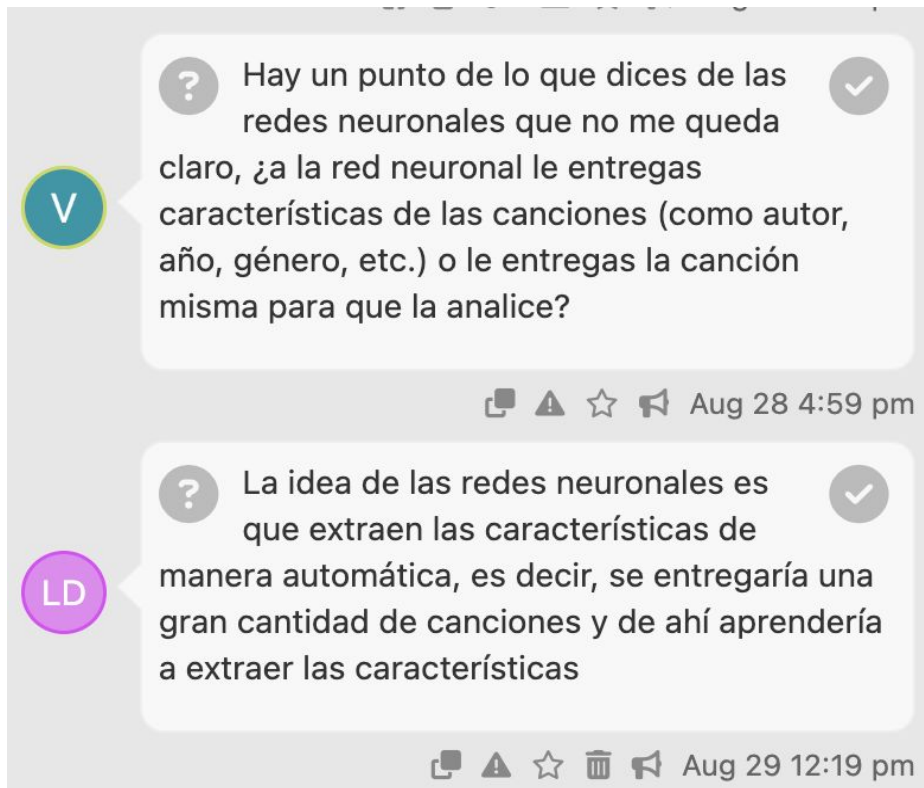
Comentarios Paper Matrix Factorization



Este estudiante plantea que no se le puede sacar más provecho a un recomendador si combinamos distintos algoritmos.

Ej. usar contenido para filtrar mi vecindario y luego aplicar user-based CF.

Comentarios Paper Matrix Factorization



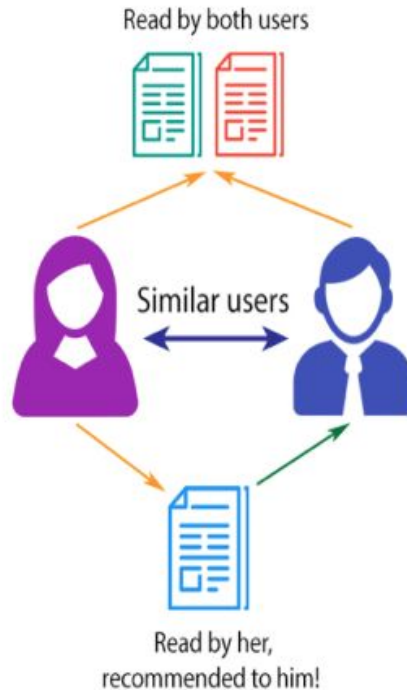
Aquí se comenta que se le puede entregar a una red neuronal un conjunto de canciones o imágenes para aprender un vector de características.

Pero estas características se aprenden para resolver una tarea en la salida de la red. La tarea puede ser predecir el rating para un usuario, incorporando el contenido del ítem.

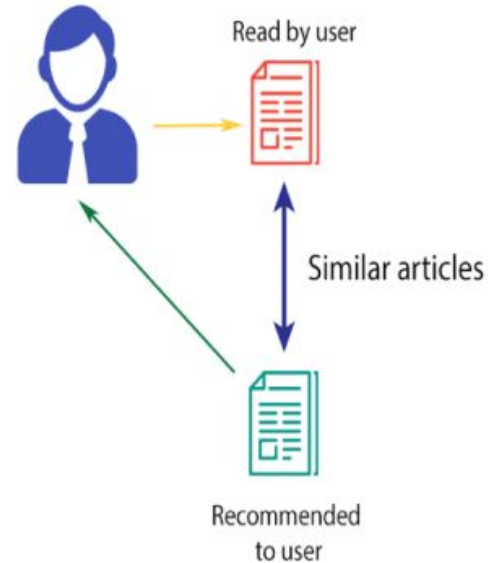
Los factores latentes que hemos visto se aprenden con la tarea de “replicar” la matriz de ratings.

Intuición de recomendación basada en contenido

COLLABORATIVE FILTERING



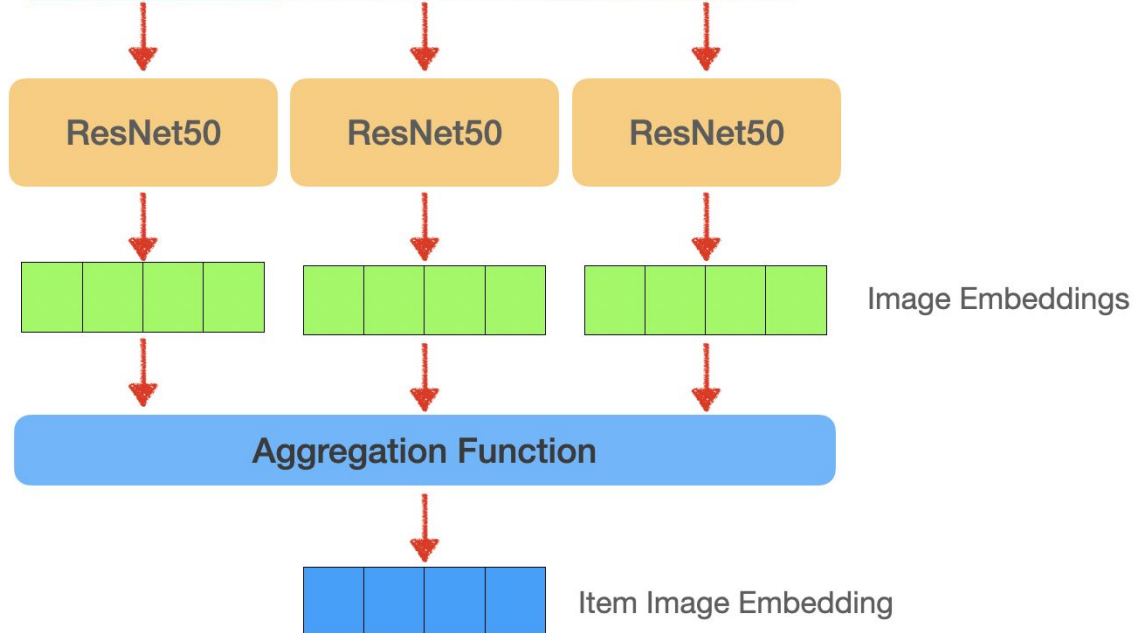
CONTENT-BASED FILTERING



Tips para recomendación basada en contenido Tarea 1

Photos for Street Burger

All (29) Food (18) Outside (6) Inside (2) Menu (1)



TIENDA

STREET BURGER

🏠 ESTADO

📍 UBICACIÓN

🕒 HORARIO

Abierta

Ver mapa

12:00 - 23:00

TIPO DE ATENCIÓN

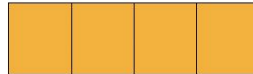
¿Cómo funcionan los servicios? ?

DESCRIPCIÓN

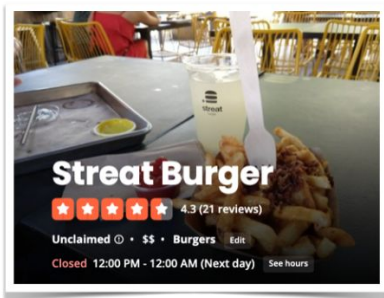
Nuestras burgers están hechas solo con los mejores ingredientes, donde la protagonista es nuestra carne americana Black Angus 100% natural. Nuestras papas fritas crujientes, doradas, onduladas y con el toque justo de sal, nuestras papas "crinkle cut" son tan buenas con nuestras burgers como lo son solas.



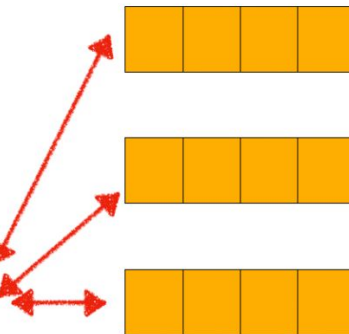
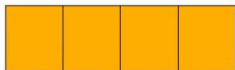
TF-IDF



Item Text Embeddings

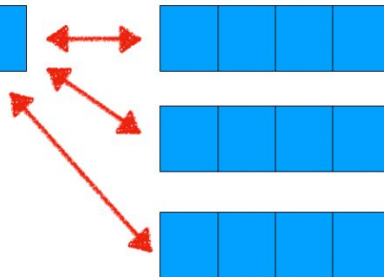


Item text embedding



...

Item image embedding



...

TOP 3

Fuente Alemana
0.7

Burguesia
0.65

Jose Ramon 277
0.63

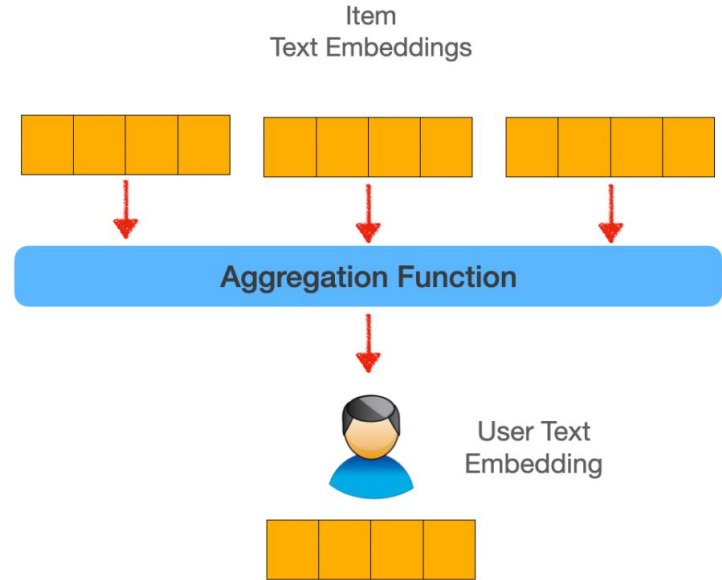
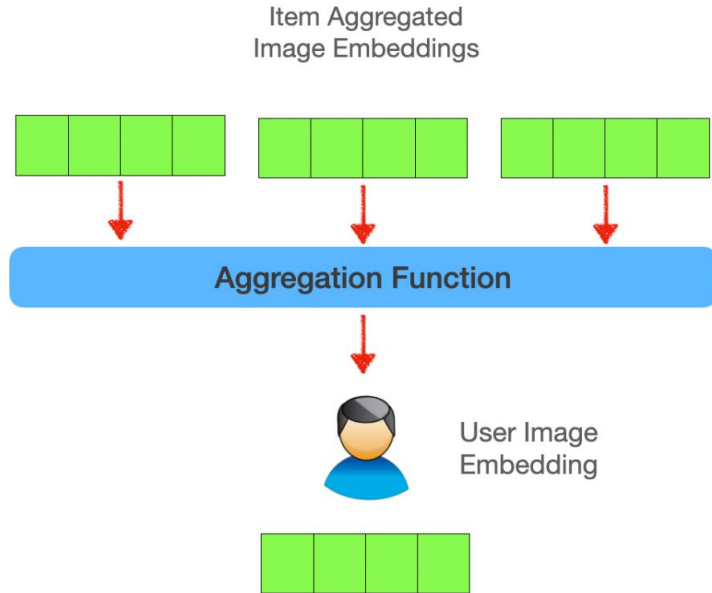
TOP 3

Albedrío
0.82

Mc Donalds
0.75

Fuente Alemana
0.63

¿Qué pasa si el usuario ha consumido muchos items?



Evaluación de sistemas recomendadores con métricas ranking

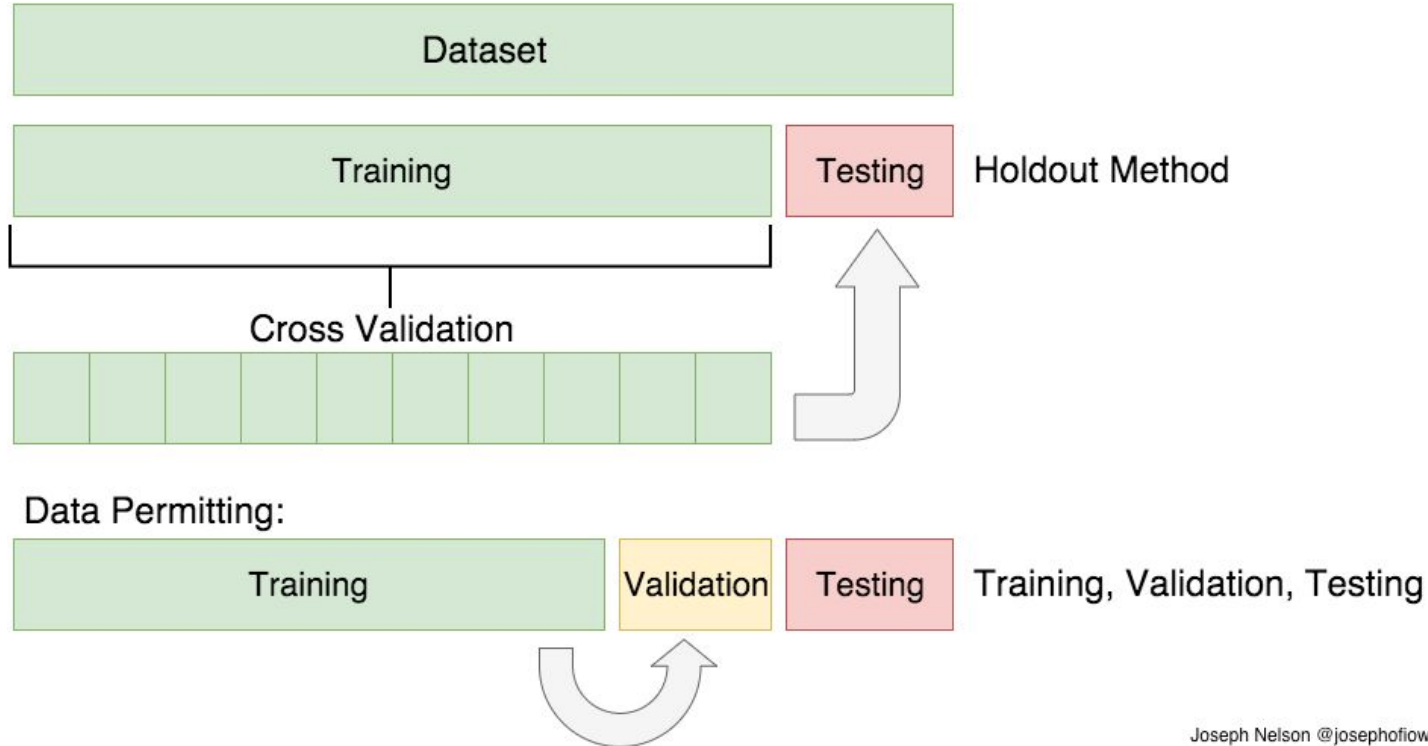
Partición de set de entrenamiento, validación y test

Set de entrenamiento: para entrenar el modelo.

Set de validación: para ver cómo está funcionando el entrenamiento.

Set de test: para testear con datos lo más parecidos a la realidad.

Partición de set de train, validación y test (cross validation)



¿Qué problemas nos puede traer hacer partición aleatoria del set de entrenamiento , validación y test para recomendación?

Consideraciones de partición de sets para recomendación (tradicional)

1. Los mismos usuarios se tienen que repetir en entrenamiento y test.
2. Todos los ítems del catálogo tienen que estar en entrenamiento, validación y test.
3. Tenemos que cortar por fecha para simular que el modelo predice interacciones futuras.

¿Cómo medimos el error para saber si estoy
recomendando bien?

Métricas de error de rating

Mean Absolute Error

$$MAE = \frac{1}{N} \sum_{i=1}^N |\hat{r}_{ui} - r_{ui}|$$

Mean Square Error

$$MSE = \frac{1}{N} \sum_{i=1}^N (\hat{r}_{ui} - r_{ui})^2$$

Root Mean Square Error

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

donde

N : cantidad de pares (usuario, item) datos evaluados

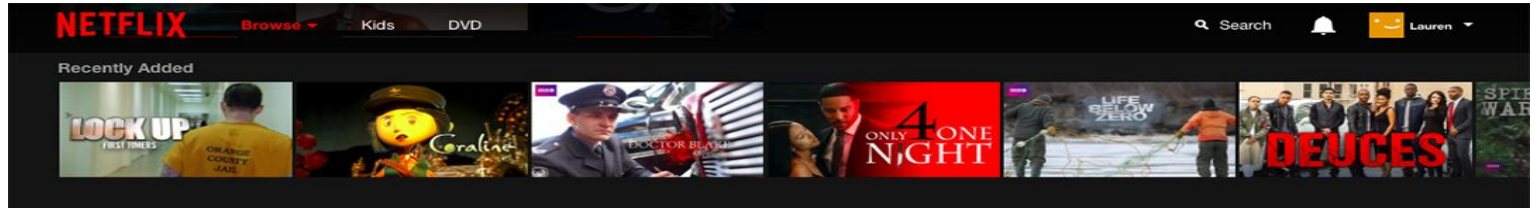
\hat{r}_{ui} : predicción del rating del usuario u al item i

r_{ui} : valor del rating que el usuario u le asignó al item i

Métricas de ranking

Métricas de ranking

- En recomendación me interesa más que predecir un rating, entregar una lista ordenada de acuerdo a las preferencias de cada usuario.



1

2

...

N

¿Cómo evaluamos si la lista está bien ordenada?

Métricas de ranking

Usuario:

Item 12 , item 32 , item 2 , item 24 , item 283, item 5

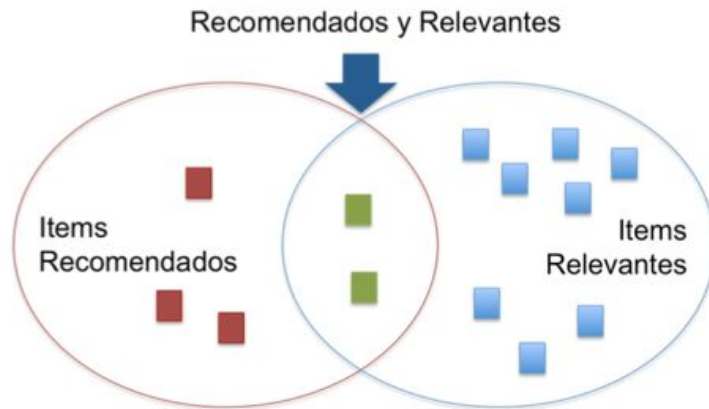
enero enero febrero febrero marzo marzo

Queremos que estos ítems
aparezcan en la lista de las
recomendaciones

Precisión y recall

Items Recomendados:

Son los items que me sugiere el modelo en un determinado orden de más a menos relevante.



Ítems Relevantes:

Yo conozco de mis datos de entrenamiento que estos ítems son relevantes para el usuario.

Son mis etiquetas con las que entreno el modelo

$$\text{Precisión} = \frac{|\text{Recomendados} \cap \text{Relevante}|}{|\text{Recomendados}|}$$

$$\text{Recall} = \frac{|\text{Recomendados} \cap \text{Relevantes}|}{|\text{Relevantes}|}$$

Ejemplo 1: *Precision & Recall*

Total relevantes: ■ x20 ■ Item Recomendado ■ Item Relevante

Recomendador 1 ■ ■ ■ ■ ■ ■ ■ ■ ■ ■

corte = @10

$$\text{Precisión} = \frac{|\text{Recomendados} \cap \text{Relevante}|}{|\text{Recomendados}|} = \frac{5}{10} = 0.5$$

$$\text{Recall} = \frac{|\text{Recomendados} \cap \text{Relevantes}|}{|\text{Relevantes}|} = \frac{5}{20} = 0.25$$

Recomendador 2 ■ ■ ■ ■ ■

corte = @5

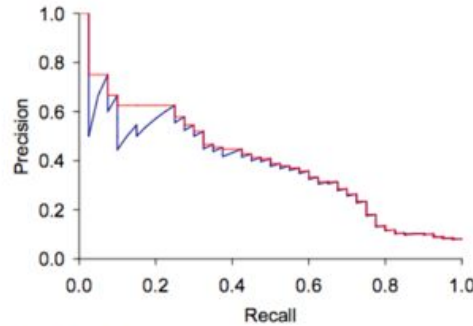
$$\text{Precisión} = \frac{|\text{Recomendados} \cap \text{Relevante}|}{|\text{Recomendados}|} = \frac{3}{5} = 0.6$$

$$\text{Recall} = \frac{|\text{Recomendados} \cap \text{Relevantes}|}{|\text{Relevantes}|} = \frac{3}{20} = 0.15$$

Voy a tener **recall máximo si recomiendo todo el catálogo** , pero voy a tener muy baja precisión porque el denominador va a ser muy grande...

Compromiso entre Precisión y Recall

- Al aumentar el *recall* (la proporción de elementos relevantes) disminuimos la precisión, por ejemplo al aumentar la cantidad de ítems relevantes.



► Figure 8.2 Precision/recall graph.



RECALL MÁXIMO , MÍNIMA PRECISIÓN
RECOMIENDO TODO PERO A UN COSTO
DE MENOS PRECISIÓN

- Para controlar este efecto se utiliza la razón armónica o *F1 score*

$$F_{\beta=1} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

¿Qué problema no resuelve precision, recall, MRR?



Mean Average Precision (MAP)

Average Precision (AP)

- AP se calcula promediando cada vez que encontramos un elemento relevante (*recall point*) sobre una lista única

$$AP = \frac{\sum_{k=1}^n P@i \cdot \text{rel}(k)}{|\text{Relevantes}|}$$

Rel(i) = 0 o Rel(i) = 1
p@k divide por k

Mean Average Precision (MAP)

- Considera el promedio de AP sobre un conjunto de listas recomendadas a todos los usuarios

$$MAP = \frac{1}{N} \sum_{u=1}^N AP(u)$$

N: número de usuarios o listas recomendadas



Discounted Cumulative Gain

- **DCG**: Discounted Cumulative Gain, mide la ganancia al ordenar la lista de forma correcta

$$\text{DCG}_p = \sum_{i=1}^p \frac{2^{\text{rel}_i} - 1}{\log_2(1 + i)}$$

- **nDCG**: Normalized Discounted Cumulative Gain

$$\text{nDCG}_p = \frac{\text{DCG}_p}{i\text{DCG}_p}$$

iDCG : es el DCG ideal donde todos los elementos son relevantes.



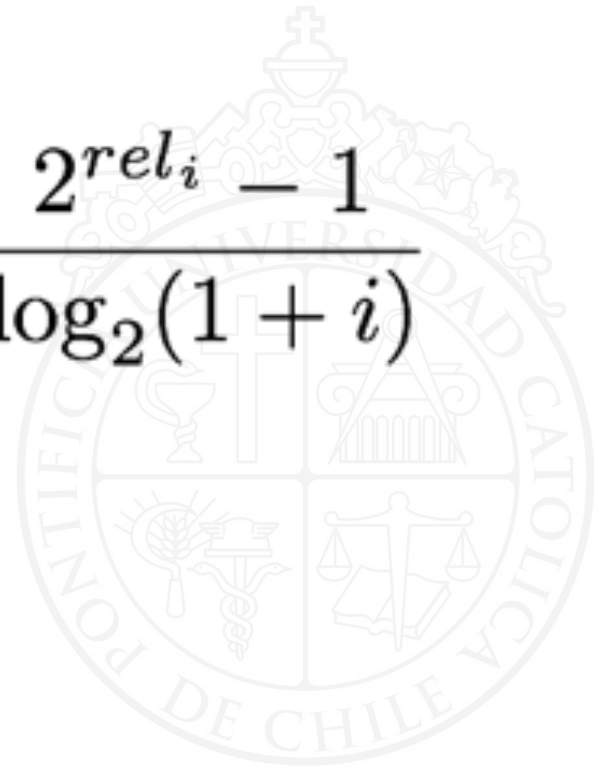
10 items

i23 , i34 , i55

[i23 , i33, i 89, i55, i90]

[1 , 0, 0, 1, 0]

$$\text{DCG}_p = \sum_{i=1}^p \frac{2^{\text{rel}_i} - 1}{\log_2(1 + i)}$$



Cobertura

Cobertura: acercar a la gente a TODO el contenido. Si las recomendaciones están sesgadas a una proporción de X ítems , no es un buen signo.

Item Coverage

Porcentaje de ítems que son recomendados por lo menos una vez

User Coverage

Porcentaje de usuarios a los cuales se les pudo hacer una recomendación

EJ.

AP MODELO 1 = 0.4

AP MODELO 2 = 0.7

PERO LA COBERTURA DEL MODELO 1 PROVIENE DEL 70% DEL CATALOGO ES MEJOR M1

Diversidad

PAIRWISE DIVERSITY: Diversidad promedio que encuentro en el contenido de los items recomendados con respecto a todo el catálogo.

De dos recomendadores igual de buenos voy a preferir el que sea más diverso.

En RRSS si no promuevo la diversidad pueden haber FILTER BUBBLES.



En esta clase

Evaluación de recomendación basada

- métricas de ranking
- cobertura
- diversidad

