Sistemas de Recomendación

Arturo Sánchez Palacio

24, 27 y 28 de Enero de 2020

Sistemas de Filtrado Colaborativo

Estructura sección

- Filtrado colaborativo usuario a usuario.
- Filtrado colaborativo item a item.

Introducción

$$s(j) = \frac{\sum_{i \in \Omega_j} r_{ij}}{|\Omega_j|}$$

Rating medio

$$s(i,j) = \frac{\sum_{i' \in \Omega_j} r_{i'j}}{|\Omega_j|}$$

Ω_j Conjunto de usuarios que han valorado j

r_{ij} Rating del usuario i al item j

$R_{N \times M}$ Matriz de valoraciones usuario-item

Filtrado colaborativo usuario a usuario

Limitaciones de la recomendación por media:

- La media es impersonal. Todas las opiniones tienen el mismo peso.
- No todo el mundo valora con el mismo rasero (optimista vs. pesimista).

Filtrado colaborativo usuario a usuario

Desviaciones.

Empleamos la desviación para contrarrestar el sesgo personal.

$$dev(i,j) = r(i,j) - \bar{r}_i$$

Filtrado colaborativo usuario a usuario

Media de desviaciones:

$$\hat{dev}(i,j) = \frac{1}{|\Omega_j|} \sum_{i' \in \Omega_j} r(i',j) - \bar{r}_{i'}$$

$$s(i,j) = \bar{r}_i + \frac{\sum_{i' \in \Omega_j} r(i',j) - \bar{r}_{i'}}{|\Omega_j|} = \bar{r}_i + \hat{dev}(i,j)$$

Filtrado colaborativo usuario a usuario

Media impersonal:

	Romeo y Julieta	Crepúsculo	A 3 metros sobre el cielo	MacBeth
Usuario A	5 *	3*	1*	4*
Usuario B	3*	5*	5*	**?**
Usuario C	2*	4,5*	5*	2*
Usuario D	1*	4*	5*	1*

Filtrado colaborativo usuario a usuario

Impersonalidad de la media:

Una posible solución es añadir pesos que ponderen las opiniones:

$$s(i,j) = \bar{r}_i + \frac{\sum_{i' \in \Omega_j} w_{ii'} \{ r_{i'j} - \bar{r}_{i'} \}}{\sum_{i' \in \Omega_j} |w_{ii'}|}$$

Filtrado colaborativo usuario a usuario

Problema: ¿Cómo definir los pesos?

$$\varrho_{xy} = \frac{\sum_{i=1}^{N}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i-\bar{x})^2}\sqrt{\sum_{i=1}^{N}(y_i-\bar{y})^2}}$$

Filtrado colaborativo usuario a usuario

Problema: Trabajamos con matrices dispersas (sparse matrix)

$$w_{ii'} = \frac{\sum_{j \in \Psi_{ii'}} (r_{ij} - \bar{r}_i)(r_{i'j} - \bar{r}_{i'})}{\sqrt{\sum_{j \in \Psi_{ii'}} (r_{ij} - \bar{r}_i)^2} \sqrt{\sum_{j \in \Psi_{ii'}} (r_{i'j} - \bar{r}_{i'})^2}}$$

Ψ_i Películas evaluadas por i $\psi_{i'}$ Películas evaluadas por i' $\psi_{ii'} = \psi_i \cap \psi_{i'}$

Filtrado colaborativo usuario a usuario

Vecinos:

- Considerar a todos los vecinos es poco eficiente.
- Fijamos un umbral y consideramos a los n más similares (vecinos).
- Se suele considerar un conjunto entre 25 y 50 vecinos.

Filtrado colaborativo usuario a usuario

Preparación del ejercicio:

- Aún no estamos trabajando con Machine Learning.
- Compite con algoritmos mucho más complejos.
- Exploración de los datos.

Filtrado colaborativo usuario a usuario

Complejidad

- La matriz de usuarios-ratings es NxM (N número de usuarios y M número de películas).
- Para calcular la similaridad entre dos usuarios recorremos las M películas: O(M).
- Para una sola predicción de un usuario necesitamos hallar la de los N usuarios (aunque nos quedemos con las K mejores). O(N)
- O(N) usuarios y O(M) cálculos ⟹ O(NM)
- Esto sería para un usuario. Una empresa calcula para todos luego $O(N^2M)$

Filtrado colaborativo usuario a usuario

Big Data:

100,000 usuarios necesitarían 10 mil millones de pesos.

Cada peso es un 32 bit-float luego 40 GB de pesos.

Se escapa en tamaño y complejidad.

Solución: Muestreamos

Filtrado colaborativo usuario a usuario

Muestreo no aleatorio:

- Elegimos los N mejores usuarios y las M mejores películas.
- Resulta en una matriz más densa.
- Experimentar para lograr un N y M válido.

Filtrado colaborativo usuario a usuario

Complejidad en la realidad:

- No se trabaja en tiempo real.
- La implementación se realiza aparte del usuario.
- Programación de tareas.

Filtrado colaborativo usuario a usuario

Método de evaluación:

Como intentamos predecir la puntuación de un supuesto usuario usamos error medio cuadrático.

Esto es **meramente didáctico**. Como hemos hablado las métricas se obtienen con la puesta en producción.

Filtrado colaborativo usuario a usuario

Preprocesamiento de los datos:

Notebook: collaborative_filtering_exercise_preprocessing.ipynb

Filtrado colaborativo usuario a usuario

Construcción del modelo:

Notebook: collaborative_filtering_user_user.ipynb

Filtrado colaborativo item a item

Idea básica:

	Item A	Item B	Item C	Item D
Usuario 1	✔	✘	✔	✔
Usuario 2		✔	✘	✘
Usuario 3	✔	✔	✘	
Usuario 4	✘		✔	
Usuario 5	✔	✔	?	✘

Por filas vemos usuarios similares, por columnas productos similares.

Filtrado colaborativo item a item

Comparación de métodos:

- En el filtrado usuario a usuario, elegimos items para el usuario porque esos items han gustado a usuarios similares.
- En el filtrado item a item, elegimos items para el usuario porque al usuario le han gustado items similares.
- El filtrado item a item devuelve resultados (ligeramente) mejores que el usuario a usuario.
- El filtrado colaborativo es más rápido pues: $O(NM^2)$
- Además normalmente se tienen más datos al comparar productos que usuarios.

Filtrado colaborativo item a item

Interpretación naïve:

- Los productos tienen sentimientos y prefieren unos usuarios a otros.
- Si dos productos son similares les gustarán usuarios similares.

Filtrado colaborativo item a item

Las fórmulas son análogas a las vistas para el filtrado usuario a usuario:

$$w_{jj'} = \frac{\sum_{i \in \Omega_{jj'}} (r_{ij} - \bar{r}_j)(r_{ij'} - \bar{r}_{j'})}{\sqrt{\sum_{i \in \Omega_{jj'}} (r_{ij} - \bar{r}_j)^2} \sqrt{\sum_{i \in \Omega_{jj'}} (r_{ij'} - \bar{r}_{j'})^2}}$$

Ω_j Usuarios que han evaluado el producto j

$\Omega_{jj'}$ Usuarios que han evaluado los productos j y j'

$\bar{r}_j$ Rating medio del **producto** j

Filtrado colaborativo item a item

Las fórmulas son análogas a las vistas para el filtrado usuario a usuario:

$$s(i,j) = \bar{r}_j + \frac{\sum_{j' \in \Psi_i} w_{jj'}(r_{ij'} - \bar{r}_{j'})}{\sum_{j' \in \Psi_i} |w_{jj'}|}$$

Ψ_i Productos valorados por el usuario i

Filtrado colaborativo item a item

Construcción del modelo:

Notebook: collaborative_filtering_item_item.ipynb

Conclusiones

- En esta sección nuestros sistemas de recomendación ya tienen en cuenta características del usuario y del producto (son personalizados).
- Aún son sistemas de reglas. No hemos implementado ningún modelo de Inteligencia Artificial/Machine Learning.
- Afrontamos el sesgo de personalidad empleando las desviaciones.
- Construimos pesos para usuarios/productos similares mediante el coeficiente de correlación de Pearson.

Bonus

Se podría plantear como un modelo de Machine Learning. Más en concreto como una regresión lineal:

$$\hat{d}(i,j) = \sum_{i' \in \Omega_j} w_{ii'} d(i',j)$$

Los pesos w serían los parámetros a aprender d(i',j) sería x.