Information Retrieval

Lab 7 - Collaborative filtering

Collaborative filtering

- Methods that allow the prediction of user preferences based on information about other users
- Mainly used in recommendation systems
- Makes it easy to personalize recommendations
- Basic assumption - users which share similar preferences in the past will share similar preferences in the future

Rakieta do Squasha WILSON Ultra Triad

Inni klienci oglądali również

40,00 zł

SMART z kurierem

PIŁKI DO SQUASHA DUNLOP zestaw 3 sztuki do wyboru

399,00 zł

SMART z kurierem

WILSON PRO Staff L - rakieta do squasha

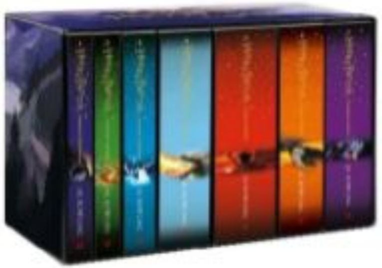
69,99 zł

SMART z kurierem

SPARTAN Rakieta Rakietka Do Gry W Squasha

Harry Potter i Kamień Filozoficzny J.K. Rowling

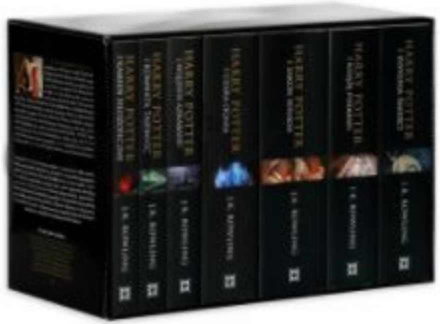
Inni klienci oglądali również



180,27 zł

SMART z kurierem

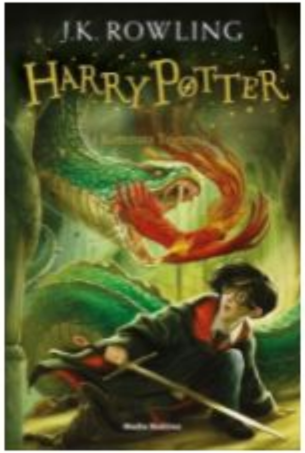
HARRY POTTER 7 TOMÓW W ETUI J. K. Rowling PAKIET



248,83 zł

SMART z kurierem

HARRY POTTER J.K. ROWLING PAKIET 7 TOMÓW ETUI



27,69 zł

SMART z kurierem

Harry Potter i Komnata Tajemnic. Tom 2. Rowling

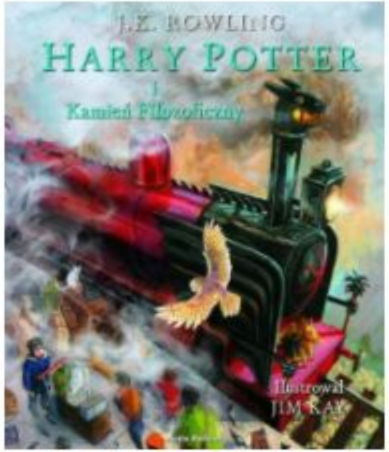
Harry Potter i Kamień Filozoficzny J.K. Rowling

Oferty sponsorowane, które mogą Cię zainteresować

23,43 zł

SMART z kurierem

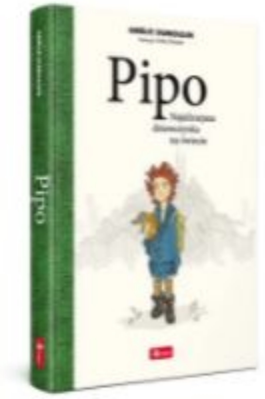
Harry Potter i kamień filozoficzny J.K. Rowling



39,87 zł

SMART z kurierem

Harry Potter i Kamień Filozoficzny J.K. Rowling



32,95 zł

SMART z kurierem

PIPO najsilniejsza dziewczynka świata powieść 2021

Basic idea

To predict how the user will rate an item:

1. Find similar users
2. Check how they rated this item
3. Aggregate their ratings to a prediction value

Basic idea

To predict how the user will rate an item:

1. Find similar users
2. Check how they rated this item
3. Aggregate their ratings to a prediction value

Alternatively:

1. Find similar items
2. Check how they were rated by the user
3. Aggregate their ratings to a prediction value

Example data

r_{ui}	Harry Potter (i_1)	Titanic (i_2)	Matrix (i_3)
Alice (u_1)	4	5	2
Bob (u_2)	3	???	5
Carol (u_3)	2		4
Dave (u_4)		1	3

Similarity measures

$$\text{cosine\_sim}(u, v) = \frac{\sum\limits_{i \in I_{uv}} r_{ui} \cdot r_{vi}}{\sqrt{\sum\limits_{i \in I_{uv}} r_{ui}^2} \cdot \sqrt{\sum\limits_{i \in I_{uv}} r_{vi}^2}}$$

Similarity measures

$$\text{msd\_distance}(u, v) = \frac{1}{|I_{uv}|} \cdot \sum_{i \in I_{uv}} (r_{ui} - r_{vi})^2$$

$$\text{msd\_sim}(u, v) = \frac{1}{\text{msd}(u, v) + 1}$$

Similarity measures

$$\text{pearson\_sim}(u, v) = \frac{\sum_{i \in I_{uv}} (r_{ui} - \mu_u) \cdot (r_{vi} - \mu_v)}{\sqrt{\sum_{i \in I_{uv}} (r_{ui} - \mu_u)^2} \cdot \sqrt{\sum_{i \in I_{uv}} (r_{vi} - \mu_v)^2}}$$

How to deal with the new user?

Problem:

- no information about the preferences
- no ratings for any item

Useful data:

- Demographic information provided during registration (e.g. age, sex, education, profession)
- User location sharing / IP location
- Preferences indicated during registration (e.g. asking about favorite genres of movies, music)

	Age	Sex	Profession	Matrix (i_3)
u_1	45	M	Engineer	5
u_2	32	F	Office worker	4
u_3	51	F	Teacher	2
u_4	23	M	Student	4
u_5	34	M	Office worker	4
u_6	20	F	Student	3
u_7	69	F	Pensioner	1
u_8	53	M	Office Worker	5
u_9	43	M	Teacher	4
u_{10}	39	F	Office Worker	2
u_{11}	50	M	Office worker	???

Problem:

- There are no users with a similar profile
- For some, individual features are similar or the same

Solution:

- Naive Bayes classifier

Bayes' theorem

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

Proof based on conditional probability:

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

$$P(B|A) = \frac{P(A \cap B)}{P(A)}$$

$$P(A|B) \cdot P(B) = P(B|A) \cdot P(A)$$

Naive Bayes classifier

$$P(C_k|\mathbf{x}) = \frac{P(\mathbf{x}|C_k) \cdot P(C_k)}{P(\mathbf{x})} \qquad \mathbf{x} = [x_1, x_2, \ldots, x_n]$$

$$P(\mathbf{x}|C_k) \cdot P(C_k) = P(x_1, x_2, \ldots, x_n, C_k)$$

$$P(x_1, x_2, \ldots, x_n, C_k) = P(x_1|x_2, \ldots, x_n, C_k) \cdot P(x_2, \ldots, x_n, C_k)$$

Assuming that x_1, x_2, ..., x_n are independent:

$$P(x_1, x_2, \ldots, x_n, C_k) = P(x_1|C_k) \cdot P(x_2, \ldots, x_n, C_k)$$

$$P(x_2, x_3, \ldots, x_n, C_k) = P(x_2|C_k) \cdot P(x_3, \ldots, x_n, C_k)$$

Finally:

$$P(x_1, x_2, \ldots, x_n, C_k) = P(C_k) \cdot \prod_{i=1}^{n} P(x_i|C_k)$$

	Age	Sex	Profession	Matrix (i_3)
u_1	45	M	Engineer	5
u_2	32	F	Office worker	4
u_3	51	F	Teacher	2
u_4	23	M	Student	4
u_5	34	M	Office worker	4
u_6	20	F	Student	3
u_7	69	F	Pensioner	1
u_8	53	M	Office Worker	5
u_9	43	M	Teacher	4
u_{10}	39	F	Office Worker	2
u_{11}	50	M	Office worker	???

$$x_1 = [\mathrm{Age} > 40]$$

$$x_2 = [\mathrm{Sex} = \mathrm{M}]$$

$$x_3 = [\mathrm{Profession} = \mathrm{Office\ Worker}]$$

$$P(x_1 | r_{ui_3} = 4) = 0.25$$

$$P(x_2 | r_{ui_3} = 4) = 0.75$$

$$P(x_3 | r_{ui_3} = 4) = 0.5$$

$$P(r_{ui_3} = 4) = 0.4$$

$$P(r_{ui_3} = 4 | \mathbf{x}) = \frac{0.25 \cdot 0.5 \cdot 0.75 \cdot 0.4}{P(\mathbf{x})}$$

$$P(r_{ui_3} = 4 | \mathbf{x}) \sim 0.0375$$

Algorithms

	Harry Potter (i_1)	Titanic (i_2)	Matrix (i_3)
Alice (u_1)	4	5	2
Bob (u_2)	3	???	5
Carol (u_3)	2		4
Dave (u_4)		1	3

Algorithms

First idea - count the average rating for this item from other users:

$$\hat{r}_{u_2 i_2} = \frac{5+1}{2} = 3$$

This approach does not take advantage of any similarities among users

	Harry Potter (i_1)	Titanic (i_2)	Matrix (i_3)
Alice (u_1)	4	5	2
Bob (u_2)	3	???	5
Carol (u_3)	2		4
Dave (u_4)		1	3

Algorithms

Next idea - Slope One

Measure how much on average other items were better/worse than the predicted one. Only consider the ratings of users who rated both of them.

$$b_{i_2 i_1} = \frac{5-4}{1} = 1$$

$$b_{i_2 i_3} = \frac{(5-2)+(1-3)}{2} = 0.5$$

	Harry Potter (i_1)	Titanic (i_2)	Matrix (i_3)
Alice (u_1)	4	5	2
Bob (u_2)	3	???	5
Carol (u_3)	2		4
Dave (u_4)		1	3

Algorithms

Based on the mean difference from the Harry Potter, the prediction should be 4. Based on the mean difference from the Matrix, the prediction should reach 5.5 (off scale).

The final prediction is determined as a weighted average, where the weights are the number of users that were used to calculate the averages.

	Harry Potter (i_1)	Titanic (i_2)	Matrix (i_3)
Alice (u_1)	4	5	2
Bob (u_2)	3	???	5
Carol (u_3)	2		4
Dave (u_4)		1	3

$$\hat{r}_{u_2 i_2} = \frac{2 \cdot 5.5 + 1 \cdot 4}{2 + 1} = 5$$

Algorithms

k-NN

1) Calculate similarity to each user with e.g. cosine similarity

	Harry Potter (i_1)	Titanic (i_2)	Matrix (i_3)
Alice (u_1)	4	5	2
Bob (u_2)	3	???	5
Carol (u_3)	2		4
Dave (u_4)		1	3

$$\text{cosine\_sim}(u_2 u_1) = \frac{3 \cdot 4 + 5 \cdot 2}{\sqrt{3^2 + 5^2} \cdot \sqrt{4^2 + 2^2}} = 0.8437$$

$$\text{cosine\_sim}(u_2 u_3) = \frac{3 \cdot 2 + 5 \cdot 4}{\sqrt{3^2 + 5^2} \cdot \sqrt{2^2 + 4^2}} = 0.9971$$

$$\text{cosine\_sim}(u_2 u_4) = \frac{5 \cdot 3}{\sqrt{3^2} \cdot \sqrt{5^2}} = 1$$

2) Estimate rating based on the ratings of the (k=2) Nearest Neighbors ratings. Use the similarity as a weight for a weighted average:

$$\hat{r}_{u_2 i_2} = \frac{1 \cdot 1 + 0.8437 \cdot 5}{1 + 0.8437} = 2.8304$$

Singular Value Decomposition - SVD

- decomposition of a matrix into three specific matrices

- formula: $A = U\Sigma V^T$

- allows to reconstruct the matrix A - in the case of reduction of the number of eigenvalues and eigenvectors, with limited accuracy

- U and V - unitary matrices, consisting in columns of eigenvectors of AA^T and A^TA matrices (called left-singular vectors and right-singular vectors of A)

- Σ - diagonal matrix with the singular values of M

- SVD is only possible when the matrix is fully filled. This is not the case at CF. How to solve it?

SVD-inspired - Matrix Factorization-based algorithms

1) Create two matrices that will show the values of *latent factors* for users (p) and items (q) - fill them with random numbers
2) Use stochastic gradient descent (SGD) algorithm for known values, to optimize the error function
3) At each step, determine the prediction estimation error for the selected user-item pair according to the formula:

$$\hat{r}_{ui} = p_u \cdot q_i^T \qquad e_{ui} = \frac{(r_{ui} - \hat{r}_{ui})^2}{2} \qquad e'_{ui} = r_{ui} - \hat{r}_{ui}$$

And update the values in matrices to reduce the error:

$$p'_u = p_u + \gamma \cdot e'_{ui} \cdot q_i \qquad q'_i = q_i + \gamma \cdot e'_{ui} \cdot p_u$$

4) After optimizing the matrices, multiply them to get a single matrix with predictions for all user-item pairs in the resulting matrix (including those that were unknown at the beginning)

Matrix Factorization-based algorithms

$$\hat{r}_{u_3 i_2} = 1.5 \cdot 1.3 + 1.1 \cdot 1.4 = 3.49$$

$$e_{u_3 i_2} = \frac{(5 - 3.49)^2}{2} = 1.14005$$

$$e'_{u_3 i_2} = 5 - 3.49 = 1.51$$

$$p'_{u_3 f_1} = 1.5 + 0.1 \cdot 1.51 \cdot 1.3 = 1.6963$$

$$p'_{u_3 f_2} = 1.1 + 0.1 \cdot 1.51 \cdot 1.4 = 1.3114$$

$$q'_{i_2 f_1} = 1.3 + 0.1 \cdot 1.51 \cdot 1.5 = 1.5265$$

$$q'_{i_2 f_2} = 1.4 + 0.1 \cdot 1.51 \cdot 1.1 = 1.5661$$

$$\hat{r}_{u3i2} = 1.6963 \cdot 1.5265 + 1.3114 \cdot 1.5661 = 4.6432$$

	Harry Potter (i_1)	Titanic (i_2)	Matrix (i_3)
Alice (u_1)	4	5	2
Bob (u_2)	3	???	
Carol (u_3)	2	5	4
Dave (u_4)		1	3

p	f1	f2
u1	0.2	2.0
u2	1.7	2.2
u3	1.5	1.1
u4	0.1	1.2

q	f1	f2
i1	1.8	1.3
i2	1.3	1.4
i3	1.5	0.4