Recommender system

Wang Jianfang
王建芳(in Chinese)
September 1, 2020

Plan

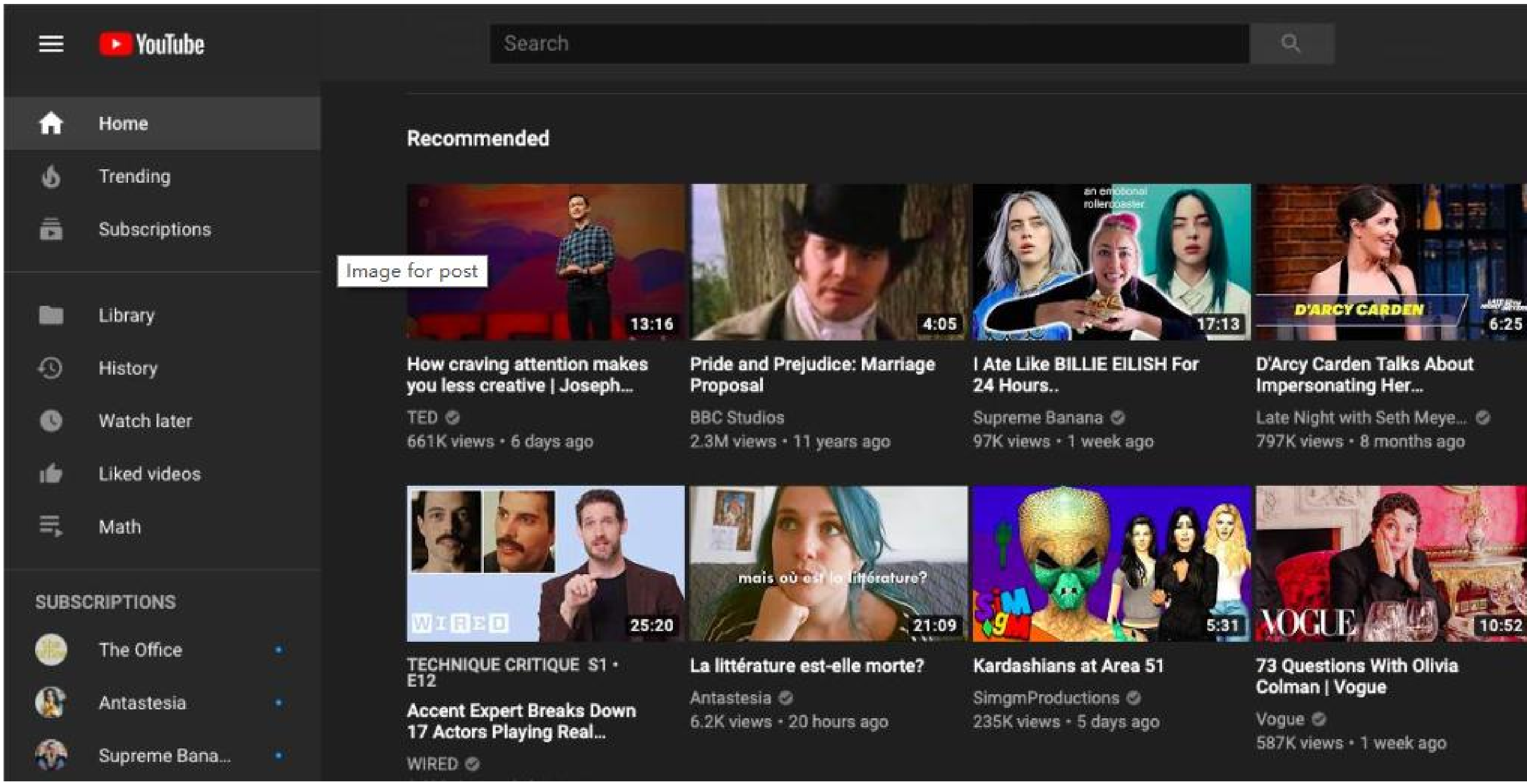
- 24(Lesson)+24(Lab.)

Time/Day	Monday	Tuesday	Wednesday	Thursday
8.00-9.50				Recommender system (Chen Feng, Wang Jianfang) (1-6,202)
10.10-12.00		Recommender system (Chen Feng, Wang Jianfang) (1-6,110)		

Agenda

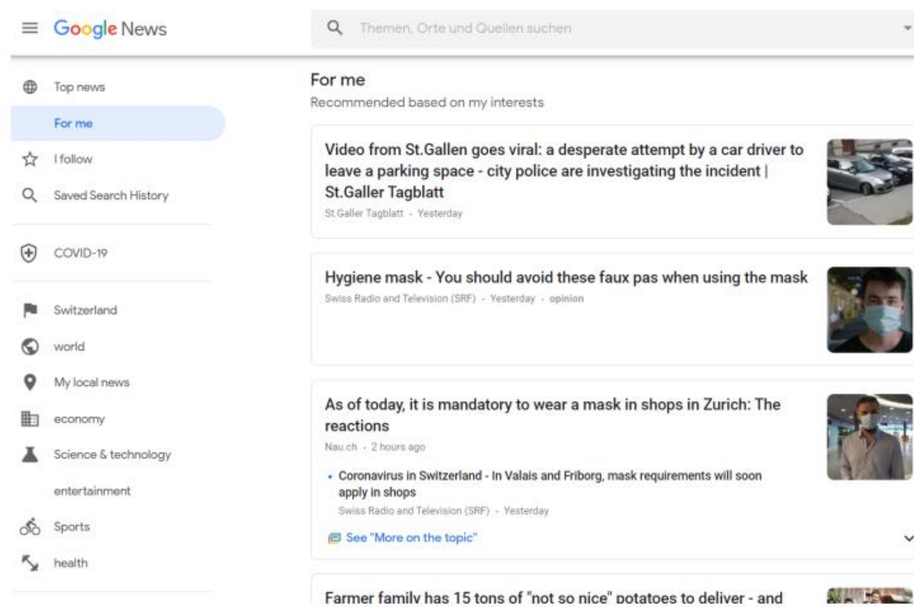
- Concepts
- Collaborative Filtering
- Algorithms (user-based)
- Evaluation Metrics
- Practise

Entertainment



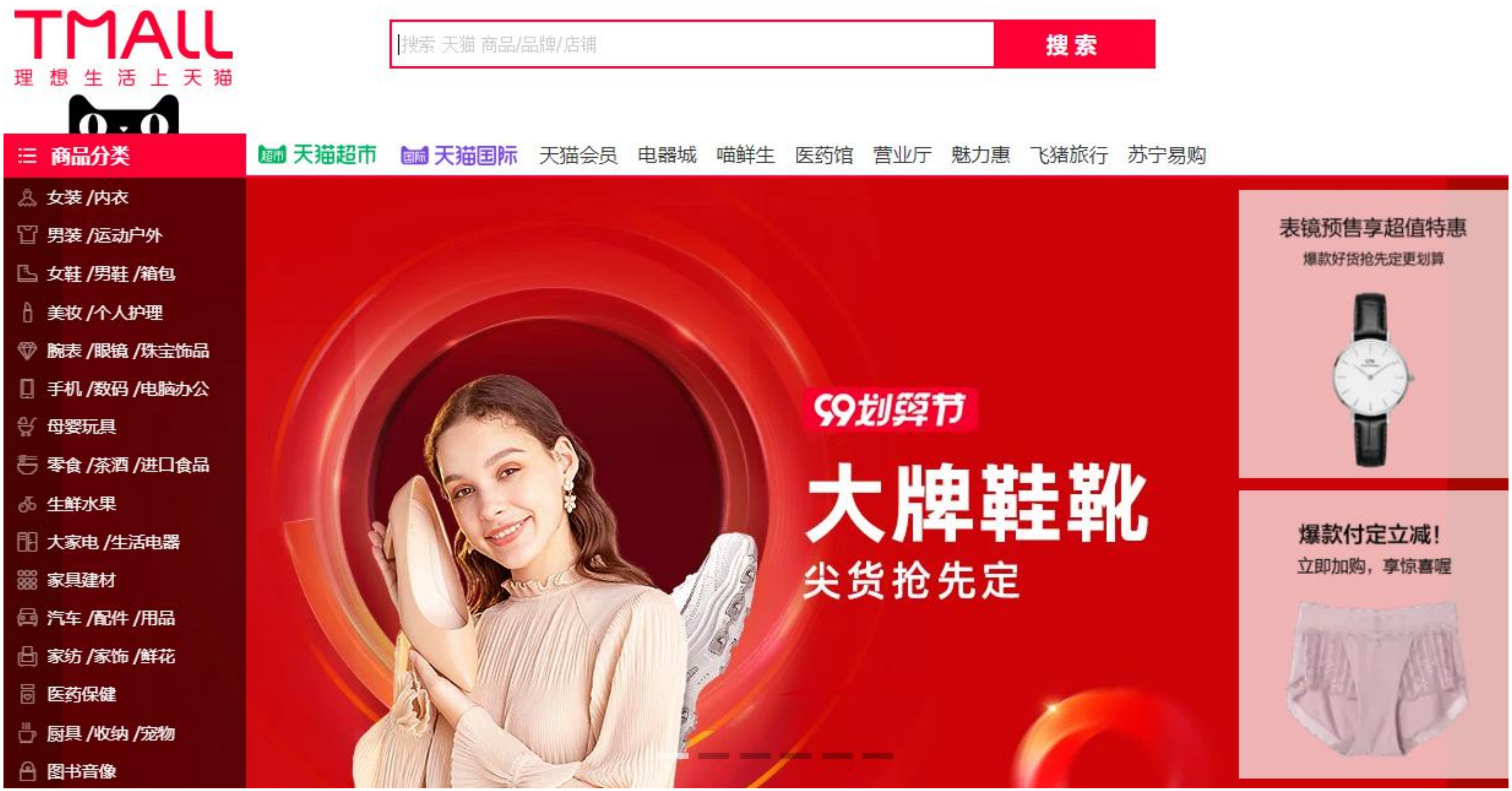
Entertainment

News Recommendation

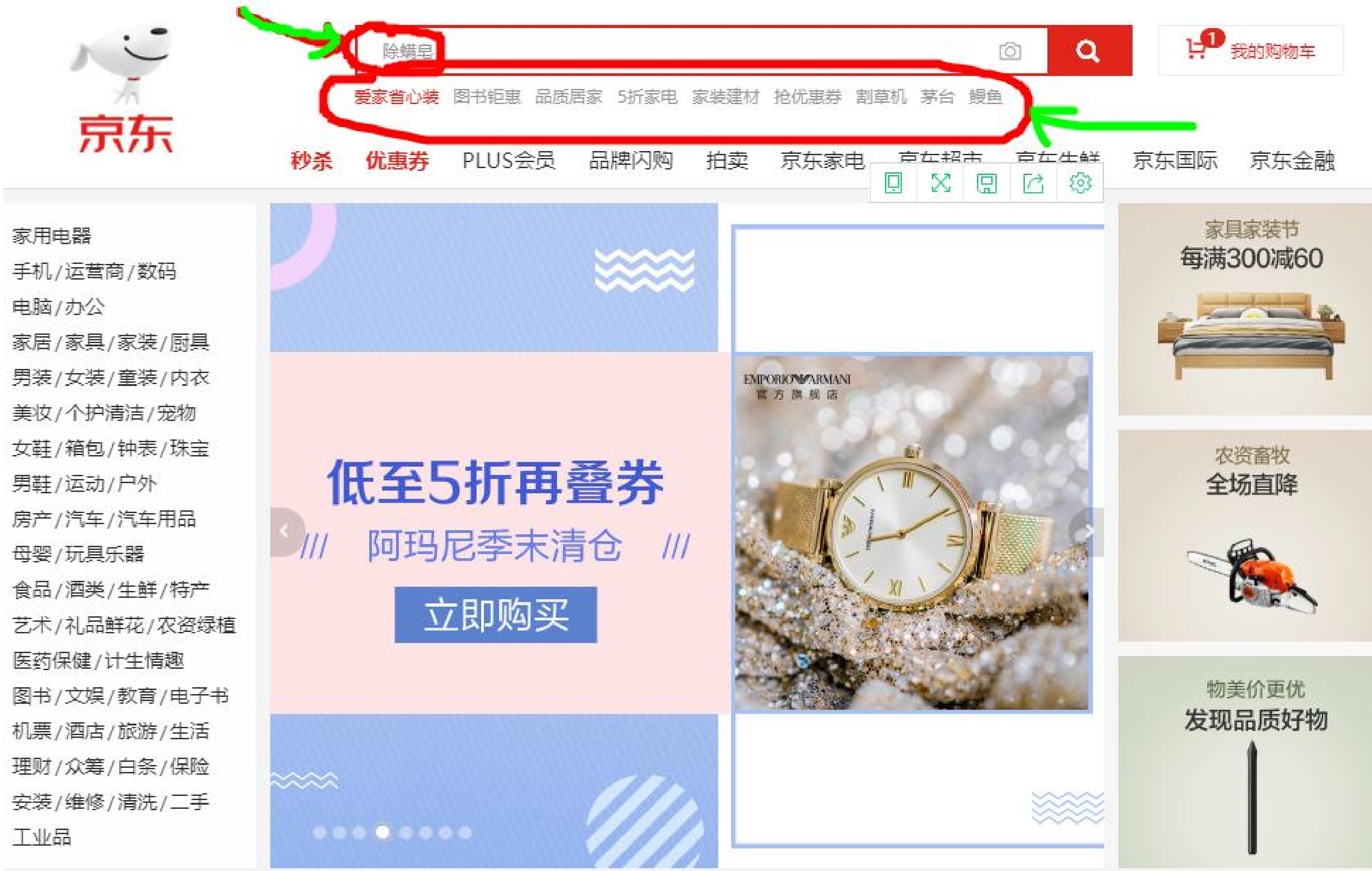


Online Shopping

Online Shopping



Online Shopping



Why are recommender systems so pervasive?

Benefits for customers

- Ease information overload
- Sales assistance (guidance, advisory, persuasion,…)
- …

Benefits for industry

- Improve user experience
- Increase sales
- …

History

Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions
Adomavicius, Gediminas, and Alexander Tuzhilin. 2005

Recommendation Approach	Recommendation Technique: Heuristic-based	Recommendation Technique: Model-based
Content-based	Commonly used techniques: • TF-IDF (information retrieval) • Clustering Representative research examples: • Lang 1995 • Balabanovic & Shoham 1997 • Pazzani & Billsus 1997	Commonly used techniques: • Bayesian classifiers • Clustering • Decision trees • Artificial neural networks Representative research examples: • Pazzani & Billsus 1997 • Mooney et al. 1998 • Mooney & Roy 1999 • Billsus & Pazzani 1999, 2000 • Zhang et al. 2002
Collaborative	Commonly used techniques: • Nearest neighbor (cosine, correlation) • Clustering • Graph theory Representative research examples: • Resnick et al. 1994 • Hill et al. 1995 • Shardanand & Maes 1995 • Breese et al. 1998 • Nakamura & Abe 1998 • Aggarwal et al. 1999 • Delgado & Ishii 1999 • Pennock & Horwitz 1999 • Sarwar et al. 2001	Commonly used techniques: • Bayesian networks • Clustering • Artificial neural networks • Linear regression • Probablistic models Representative research examples: • Billsus & Pazzani 1998 • Breese et al. 1998 • Ungar & Foster 1998 • Chien & George 1999 • Getoor & Sahami 1999 • Pennock & Horwitz 1999 • Goldberg et al. 2001 • Kumar et al. 2001 • Pavlov & Pennock 2002 • Shani et al. 2002 • Yu et al. 2002, 2004 • Hofmann 2003, 2004 • Marlin 2003 • Si & Jin 2003
Hybrid	Combining content-based and collaborative components using: • Linear combination of predicted ratings • Various voting schemes • Incorporating one component as a part of the heuristic for the other	Combining content-based and collaborative components by: • Incorporating one component as a part of the model for the other • Building one unifying model Representative research examples:

History

A Survey of Collaborative Filtering Techniques
Xiaoyuan Su and Taghi M. Khoshgoftaar
2009

Table 2: Overview of collaborative filtering techniques.

CF categories	Representative techniques	Main advantages	Main shortcomings
Memory-based CF	* Neighbor-based CF (item-based/user-based CF algorithms with Pearson/vector cosine correlation) * Item-based/user-based top-*N* recommendations	* easy implementation * new data can be added easily and incrementally * need not consider the content of the items being recommended * scale well with co-rated items	* are dependent on human ratings * performance decrease when data are sparse * cannot recommend for new users and items * have limited scalability for large datasets
Model-based CF	* Bayesian belief nets CF * clustering CF * MDP-based CF * latent semantic CF * sparse factor analysis * CF using dimensionality reduction techniques, for example, *SVD, PCA*	* better address the sparsity, scalability and other problems * improve prediction performance * give an intuitive rationale for recommendations	* expensive model-building * have trade-off between prediction performance and scalability * lose useful information for dimensionality reduction techniques
Hybrid recommenders	* content-based CF recommender, for example, *Fab* * content-boosted CF * hybrid CF combining memory-based and model-based CF algorithms, for example, Personality Diagnosis	* overcome limitations of CF and content-based or other recommenders * improve prediction performance * overcome CF problems such as sparsity and gray sheep	* have increased complexity and expense for implementation * need external information that usually not available

Netflix Prize

The **Netflix Prize** was an open competition for the best collaborative filtering algorithm to predict user ratings for films, based on previous ratings without any other information about the users or films.

On September 21, 2009 we awarded the $1M Grand Prize to team “BellKor’s Pragmatic Chaos”.

https://www.netflixprize.com/

Collaborative Filtering (CF)

- The most prominent approach to generate recommen dations
  - used by large, commercial e-commerce sites
  - well-understood, various algorithms and variations exist
  - applicable in many domains (book, movies, DVDs, ..)
- Approach
  - use the "wisdom of the crowd" to recommend items

- Basic assumption and idea
  - Users give ratings to catalog items (implicitly or explicitly)
  - Customers who had similar tastes in the past, will have similar t astes in the future

Pure CF Approaches

- Input
  - Only a matrix of given user–item ratings
- Output types
  - A (numerical) prediction indicating to what degr ee the current user will like or dislike a certain it em
  - A top-N list of recommended items

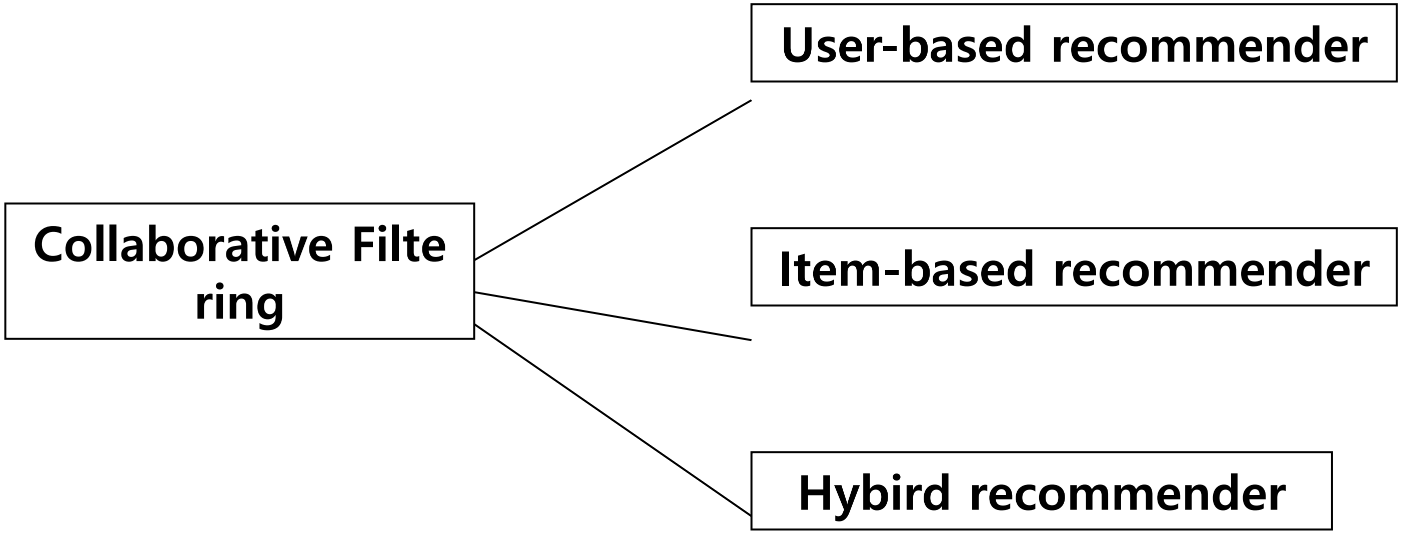
Concepts

- Collaborative Filtering
  - The process of information filtering by collecting human judgments (ratings)
  - “word of mouth”
- User
  - Any individual who provides ratings to a system
- Items
  - Anything for which a human can provide a rating

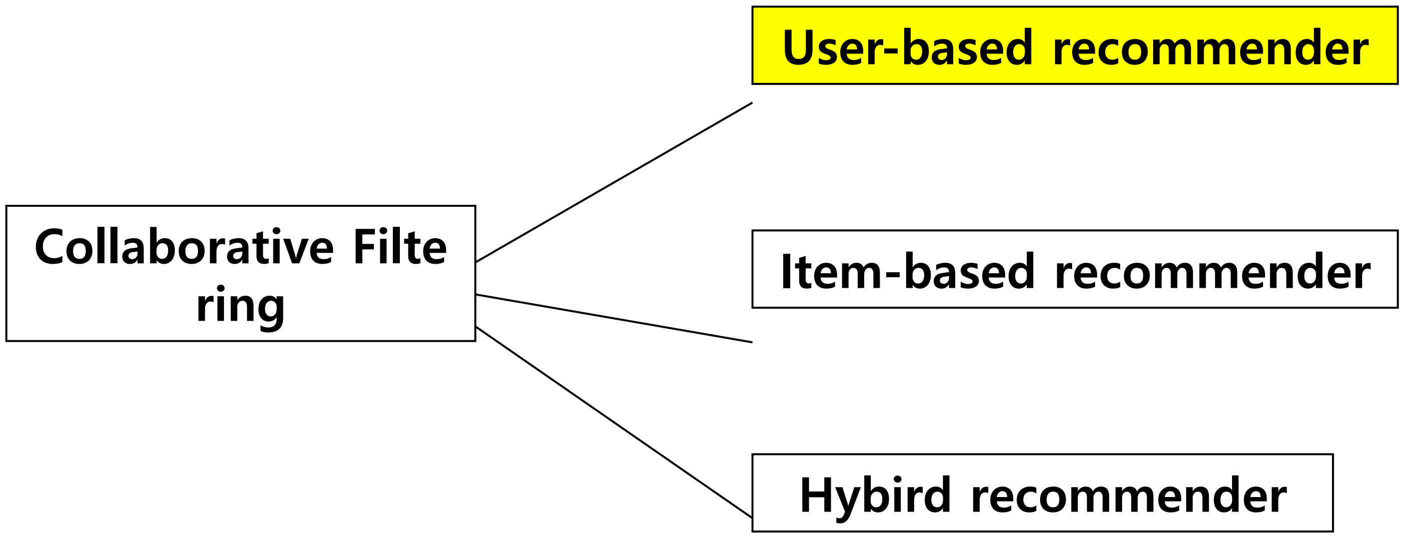
Uses for CF : Domains

- Many items
- Many ratings
- Many more users than items recommended
- Users rate multiple items
- For each user of the community, there are other u sers with common needs or tastes
- Item evaluation requires personal taste
- Items persists
- Taste persists
- Items are homogenous

Algorithms


Collaborative Filte
ring
User-based recommender
Item-based recommender
Hybird recommender

Algorithms


User-based recommender
Collaborative Filte ring
Item-based recommender
Hybird recommender

Algorithms : **User-Based**

- **User-Based Nearest Neighbor**
  - Neighbor = similar users
  - Generate a prediction for an item *i* by analyzing ratings for *i* from users in *u*'s neighborhood

$$pred(u,i) = \bar{r}_u + \frac{\sum_{n \subset neighbors(u)} sim(u,n) \cdot (r_{ni} - \bar{r}_n)}{\sum_{n \subset neighbors(u)} sim(u,n)}$$

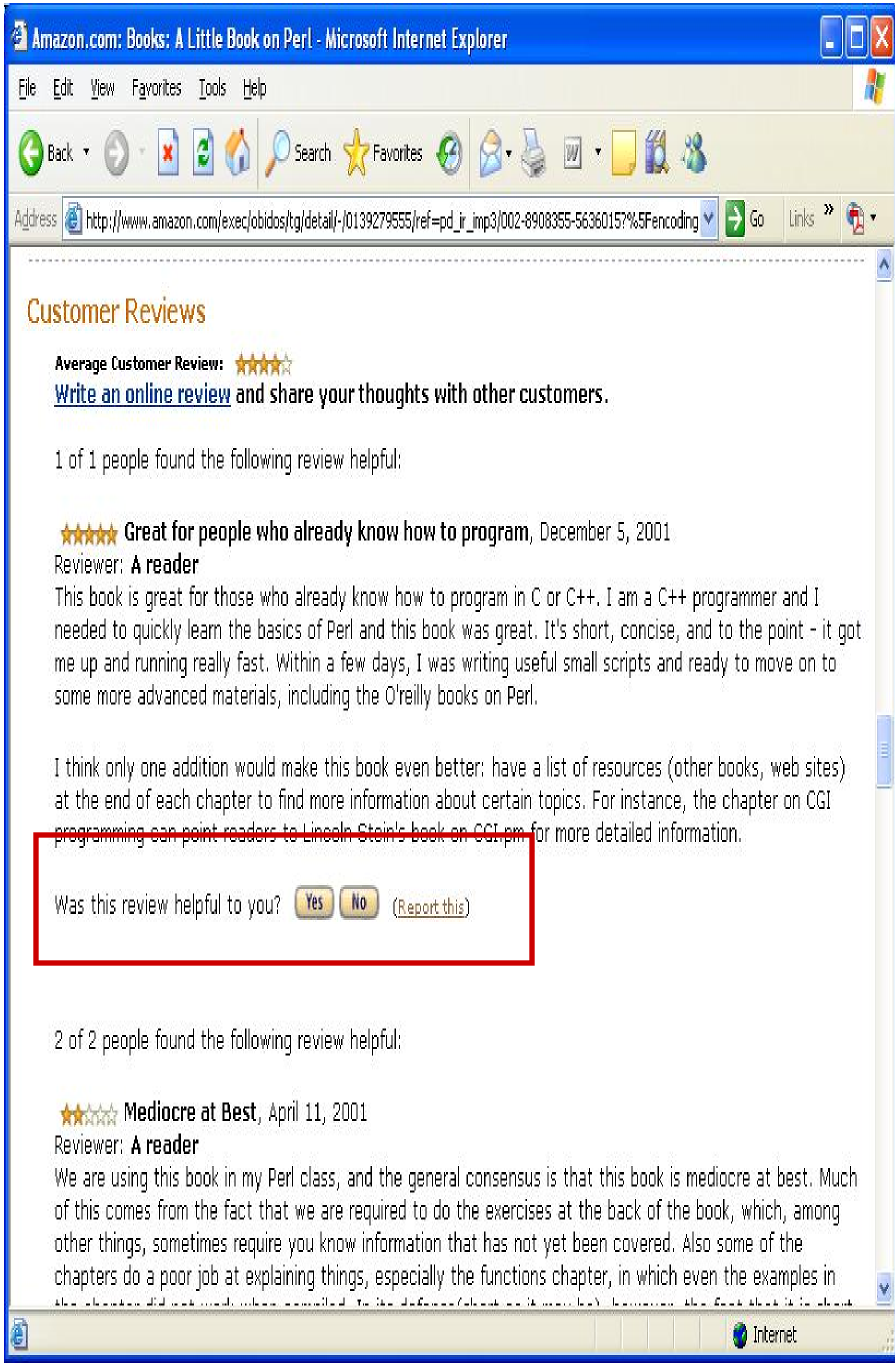
Algorithms : **Item-Based**

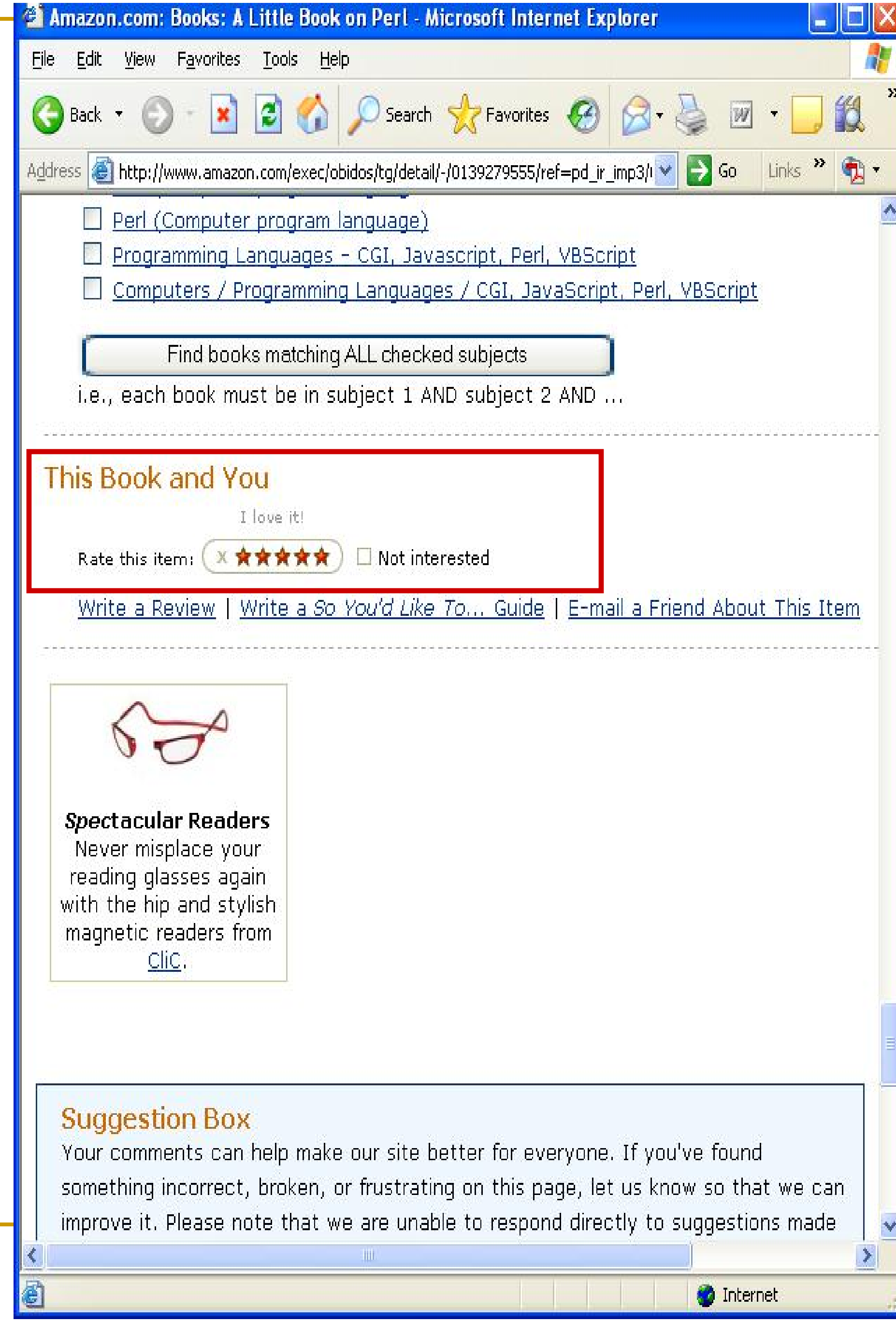
- **Item-Based Nearest Neighbor**
  - Generate predictions based on similarities between items.
  - Prediction for a user *u* and item *i* is composed of a weight ed sum of the user *u*'s ratings for items most similar to *i*.

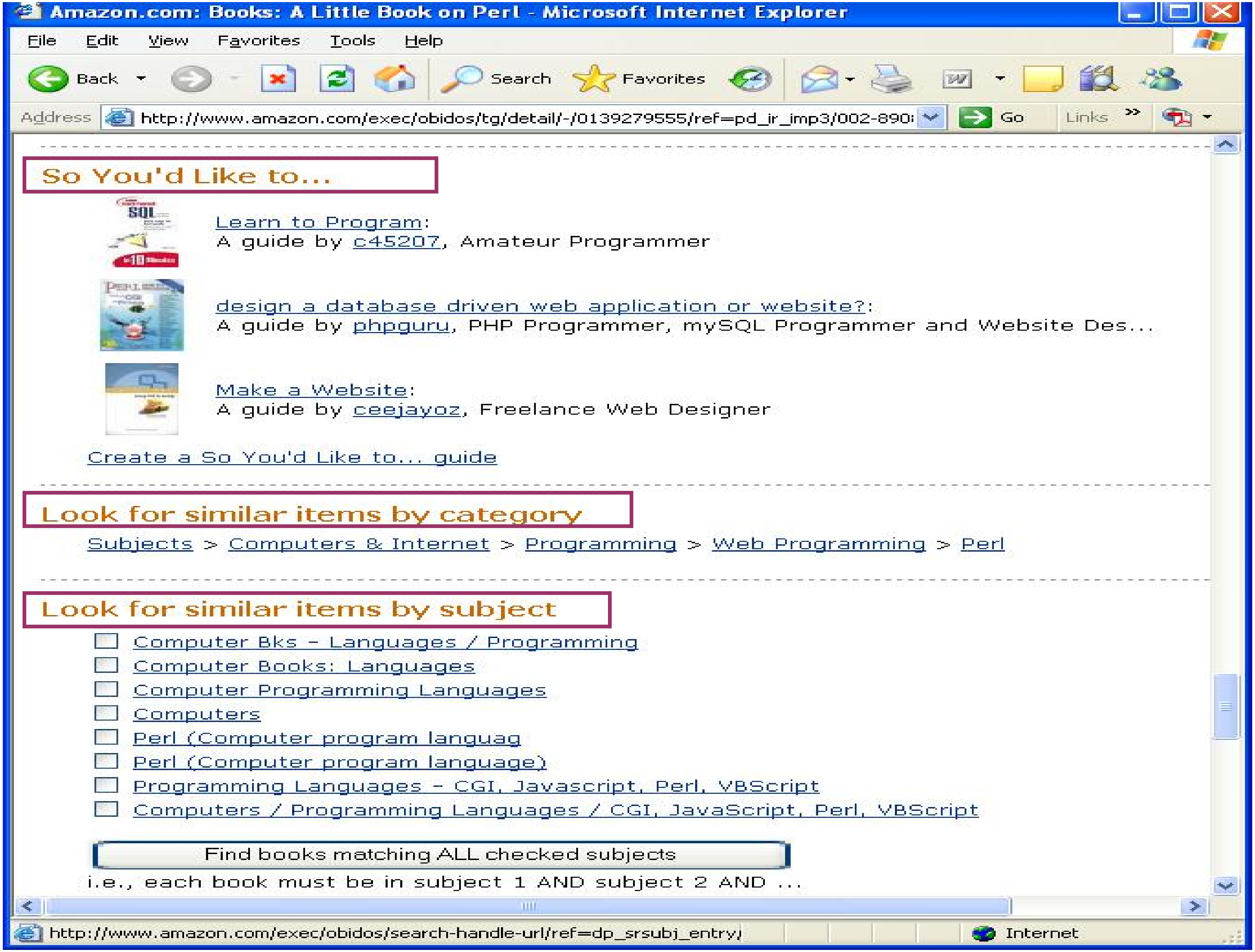
$$pred(u,i) = \frac{\sum_{j \in ratedItems(u)} sim(i,j) \cdot r_{ui}}{\sum_{j \in ratedItems(u)} sim(i,j)}$$

Practical Issues : Ratings

- Explicit vs. Implicit ratings
  - Explicit ratings
    - Users rate themselves for an item
    - Most accurate descriptions of a user's preference
    - Challenging in collecting data
  - Implicit ratings
    - Observations of user behavior
    - Can be collected with little or no cost to user
    - Ratings inference may be imprecise.


Amazon.com: Books: A Little Book on Perl - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://www.amazon.com/exec/obidos/tg/detail/-/0139279555/ref=pd_ir_imp3/002-8908355-5636015?%5Fencoding
Go Links
Customer Reviews
Average Customer Review:
Write an online review and share your thoughts with other customers.
1 of 1 people found the following review helpful:
Great for people who already know how to program, December 5, 2001
Reviewer: A reader
This book is great for those who already know how to program in C or C++. I am a C++ programmer and I needed to quickly learn the basics of Perl and this book was great. It's short, concise, and to the point - it got me up and running really fast. Within a few days, I was writing useful small scripts and ready to move on to some more advanced materials, including the O'reilly books on Perl.
I think only one addition would make this book even better: have a list of resources (other books, web sites) at the end of each chapter to find more information about certain topics. For instance, the chapter on CGI programming can point readers to Lincoln Stein's book on CGI.pm for more detailed information.
Was this review helpful to you? Yes No (Report this)
2 of 2 people found the following review helpful:
Mediocre at Best, April 11, 2001
Reviewer: A reader
We are using this book in my Perl class, and the general consensus is that this book is mediocre at best. Much of this comes from the fact that we are required to do the exercises at the back of the book, which, among other things, sometimes require you know information that has not yet been covered. Also some of the chapters do a poor job at explaining things, especially the functions chapter, in which even the examples in
Internet


Amazon.com: Books: A Little Book on Perl - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://www.amazon.com/exec/obidos/tg/detail/-/0139279555/ref=pd_ir_imp3/
Go Links
Perl (Computer program language)
Programming Languages - CGI, Javascript, Perl, VBScript
Computers / Programming Languages / CGI, JavaScript, Perl, VBScript
Find books matching ALL checked subjects
i.e., each book must be in subject 1 AND subject 2 AND ...
This Book and You
I love it!
Rate this item:
Not interested
Write a Review | Write a So You'd Like To... Guide | E-mail a Friend About This Item
Spectacular Readers
Never misplace your reading glasses again with the hip and stylish magnetic readers from ClIC.
Suggestion Box
Your comments can help make our site better for everyone. If you've found something incorrect, broken, or frustrating on this page, let us know so that we can improve it. Please note that we are unable to respond directly to suggestions made
Internet


Amazon.com: Books: A Little Book on Perl - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://www.amazon.com/exec/obidos/tg/detail/-/0139279555/ref=pd_ir_imp3/002-890
Go Links
So You'd Like to...
Learn to Program:
A guide by c45207, Amateur Programmer
design a database driven web application or website?:
A guide by phpguru, PHP Programmer, mySQL Programmer and Website Des...
Make a Website:
A guide by ceejayoz, Freelance Web Designer
Create a So You'd Like to... guide
Look for similar items by category
Subjects > Computers & Internet > Programming > Web Programming > Perl
Look for similar items by subject
Computer Bks - Languages / Programming
Computer Books: Languages
Computer Programming Languages
Computers
Perl (Computer program languag
Perl (Computer program language)
Programming Languages - CGI, Javascript, Perl, VBScript
Computers / Programming Languages / CGI, JavaScript, Perl, VBScript
Find books matching ALL checked subjects
i.e., each book must be in subject 1 AND subject 2 AND ...
http://www.amazon.com/exec/obidos/search-handle-url/ref=dp_srsubj_entry/
Internet

Practical Issues : Ratings

- Rating Scales
  - Scalar ratings
    - Numerical scales
    - 1-5, 1-7, etc.
  - Binary ratings
    - Agree/Disagree, Good/Bad, etc.
  - Unary ratings
    - Good, Purchase, etc.
    - Absence of rating indicates no information

User-based nearest-neighbor collaborative filtering (1)

- **The basic technique**
  - Given an "active user" (Alice) and an item i not yet seen by Alice
    - find a set of users (peers/nearest neighbors) who liked the same items as Alice in the past **and** who have rated item i
    - use, e.g. the average of their ratings to predict, if Alice will like item i
    - do this for all items Alice has not seen and recommend the best-rated
- **Basic assumption and idea**
  - If users had similar tastes in the past they will have similar tastes in the future
  - User preferences remain stable and consistent over time

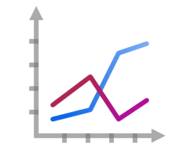
User-based nearest-neighbor collaborative filtering (2)

- Example
  - A database of ratings of the current user, Alice, and some other users is given:

	Item1	Item2	Item3	Item4	Item5
Alice	5	3	4	4	?
User1	3	1	2	3	3
User2	4	3	4	3	5
User3	3	3	1	5	4
User4	1	5	5	2	1

  - Determine whether Alice will like or dislike *Item5*, which Alice has not yet rated or seen

User-based nearest-neighbor collaborative filtering (3)



- Some first questions
  - How do we measure similarity?
  - How many neighbors should we consider?
  - How do we generate a prediction from the neighbors' ratings?

	Item1	Item2	Item3	Item4	Item5
Alice	5	3	4	4	?
User1	3	1	2	3	3
User2	4	3	4	3	5
User3	3	3	1	5	4
User4	1	5	5	2	1

Measuring user similarity (1)

- **A popular similarity measure in user-based CF: Pearson correlation**

  a, b : users

  $r_{a,p}$: rating of user a for item p

  P : set of items, rated both by a and b

  – Possible similarity values between -1 and 1

$$\boldsymbol{sim(a,b)} = \frac{\sum_{p \in P}(r_{a,p} - \bar{r}_a)(r_{b,p} - \bar{r}_b)}{\sqrt{\sum_{p \in P}(r_{a,p} - \bar{r}_a)^2}\sqrt{\sum_{p \in P}(r_{b,p} - \bar{r}_b)^2}}$$

Measuring user similarity (2)

- **A popular similarity measure in user-based CF: Pearson correlation**

  a, b : users

  $r_{a,p}$: rating of user a for item p

  P : set of items, rated both by a and b

  – Possible similarity values between -1 and 1

	Item1	Item2	Item3	Item4	Item5
Alice	5	3	4	4	?
User1	3	1	2	3	3
User2	4	3	4	3	5
User3	3	3	1	5	4
User4	1	5	5	2	1

sim = 0,85

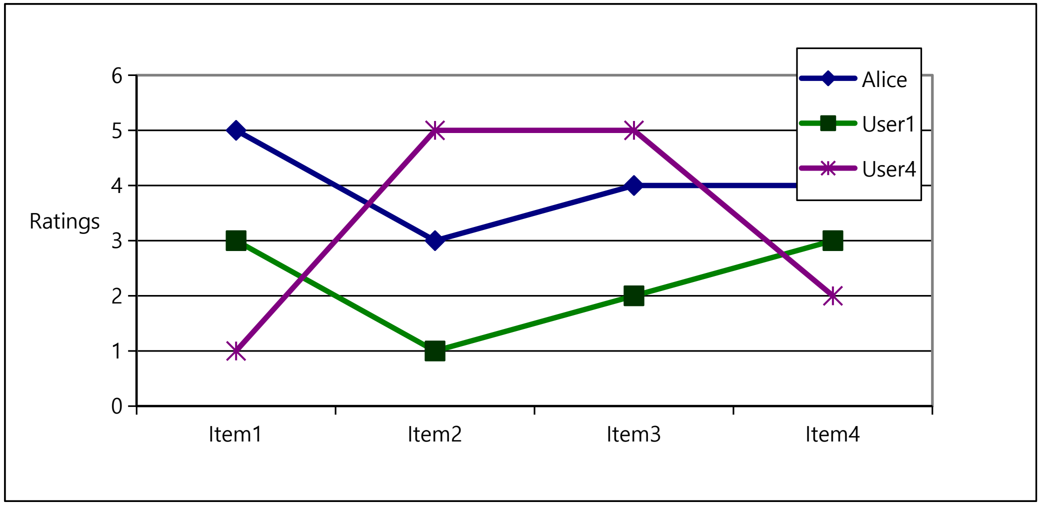
sim = 0,00

sim = 0,70

sim = -0,79

Pearson correlation

- Takes differences in rating behavior into account



- Works well in usual domains, compared with alternative measures
  - such as cosine similarity

Making predictions

- A common prediction function:

$$pred(a,p) = \overline{r_a} + \frac{\sum_{b \in N} sim(a,b) * (r_{b,p} - \overline{r_b})}{\sum_{b \in N} sim(a,b)}$$

- Calculate, whether the neighbors' ratings for the unseen item i are higher or lower than their average
- Combine the rating differences – use the similarity with a as a weight
- Add/subtract the neighbors' bias from the active user's average and use this as a prediction

similarity function

- Pearson

$$sim(a,b) = \frac{\sum_{p \in P}(r_{a,p} - \bar{r}_a)(r_{b,p} - \bar{r}_b)}{\sqrt{\sum_{p \in P}(r_{a,p} - \bar{r}_a)^2}\sqrt{\sum_{p \in P}(r_{b,p} - \bar{r}_b)^2}}$$

- Cosine Similarity

$$w_{uv} = \frac{|N(u) \cap N(v)|}{|N(u) \cup N(v)|}$$

- Jaccard

$$w_{uv} = \frac{|N(u) \cap N(v)|}{\sqrt{|N(u)||N(v)|}}$$

similarity function

- Pearson

$$sim(a,b) = \frac{\sum_{p \in P}(r_{a,p} - \bar{r}_a)(r_{b,p} - \bar{r}_b)}{\sqrt{\sum_{p \in P}(r_{a,p} - \bar{r}_a)^2}\sqrt{\sum_{p \in P}(r_{b,p} - \bar{r}_b)^2}}$$

- Pearson(simpler)

$$r = \frac{\sum XY - \frac{\sum X \sum Y}{N}}{\sqrt{(\sum X^2 - \frac{(\sum X)^2}{N})(\sum Y^2 - \frac{(\sum Y)^2}{N})}}$$

- Pearson(simpler)

$$r = \frac{\sum XY - \frac{\sum X \sum Y}{N}}{\sqrt{\left(\sum X^2 - \frac{(\sum X)^2}{N}\right)\left(\sum Y^2 - \frac{(\sum Y)^2}{N}\right)}}$$

- X rating is [1, 2, 3] , Y rating is [2, 5, 6]. wha t is the person similarity.

Pratise

- How to get the similarity?

$$sim(a,b) = \frac{\sum_{p \in P}(r_{a,p} - \bar{r}_a)(r_{b,p} - \bar{r}_b)}{\sqrt{\sum_{p \in P}(r_{a,p} - \bar{r}_a)^2}\sqrt{\sum_{p \in P}(r_{b,p} - \bar{r}_b)^2}}$$

	Item1	Item2	Item3	Item4	Item5	
Alice	5	3	4	4	?	
User1	3	1	2	3	3	sim = 0,85
User2	4	3	4	3	5	sim = 0,00
User3	3	3	1	5	4	sim = 0,70
User4	1	5	5	2	1	sim = -0,79

similarity function

- Pearson

$$sim(a,b) = \frac{\sum_{p \in P}(r_{a,p} - \bar{r}_a)(r_{b,p} - \bar{r}_b)}{\sqrt{\sum_{p \in P}(r_{a,p} - \bar{r}_a)^2}\sqrt{\sum_{p \in P}(r_{b,p} - \bar{r}_b)^2}}$$

- Pearson correlation coefficient formula is used to calculate the correlation coefficient of XY to verif y the correctness of the code

$$\rho_{X,Y} = \frac{COV(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X-\mu_X)(X-\mu_Y)]}{\sigma_X \sigma_Y} = \frac{E(XY) - E(X)E(Y)}{\sqrt{E[X^2] - E[X]^2}\sqrt{E[Y^2] - E[Y]^2}}$$

Homework

- ex1. gaining the Pearson similarity by the detail process .

(Alice with user1, user2,user3,user4) .

	Item1	Item2	Item3	Item4	Item5
Alice	5	3	4	4	?
User1	3	1	2	3	3
User2	4	3	4	3	5
User3	3	3	1	5	4
User4	1	5	5	2	1

sim = 0,85

sim = 0,00

sim = 0,70

sim = -0,79

ex2. reading the file(1.1-collaborative filtering.pdf) , and memeory th e user-based collaborative filtering algorithm mainly consists of two steps, and understand the Cosine Similarity.

ex3. setup Anaconda3 software.

Questions and Comments?

Thank you!!