GENERAL ASSEMBLY

INTRO TO DATA SCIENCE

RECOMMENDATION SYSTEMS, COLLABORATIVE FILTERING

AGENDA

- Recommendation Systems
- Collaborative Filtering
- Demo

RECOMMENDATION SYSTEMS

RECOMMENDATION SYSTEMS

Customers Who Bought This Item Also Bought

Pitch Dark (NYRB Classics)
› Renata Adler
Paperback
$11.54

How Literature Saved My Life
› David Shields
★★★★☆ (60)
Hardcover
$18.08

Bleeding Edge
Thomas Pynchon
Hardcover
$18.05

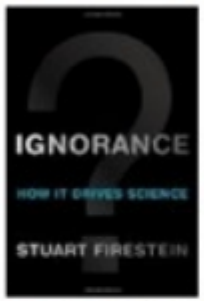
The Flamethrowers: A Novel
› Rachel Kushner
★★★½☆ (17)
Hardcover
$15.79

RECOMMENDATION SYSTEMS

Inspired by Your Wish List

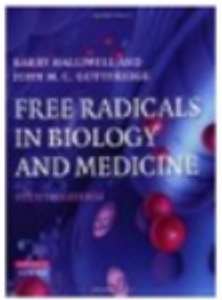
You wished for | Customers who viewed this also viewed

The Secret Anarchy of Science
› Michael Brooks
Paperback
★★★★☆ (6)



Ignorance: How It Drives Science
› Stuart Firestein
Hardcover
★★★★½ (31)
~~$21.95~~ $13.02

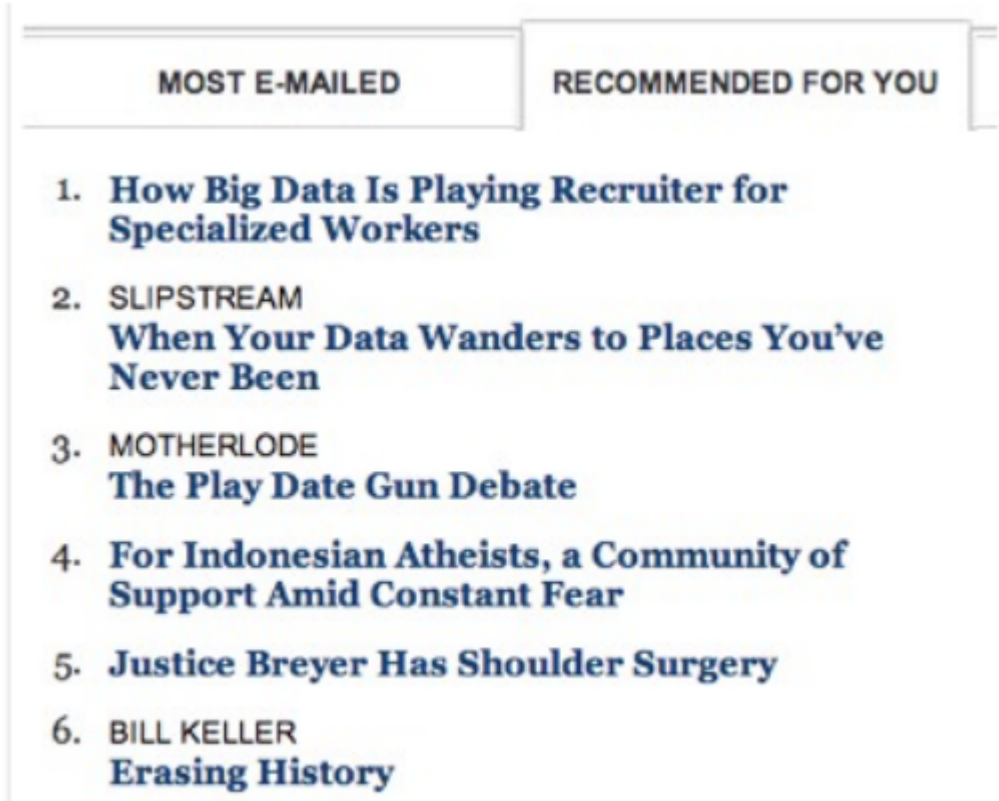
13 Things that Don't Make Sense: The...
› Michael Brooks
Paperback
★★★★☆ (65)
~~$15.95~~ $12.49



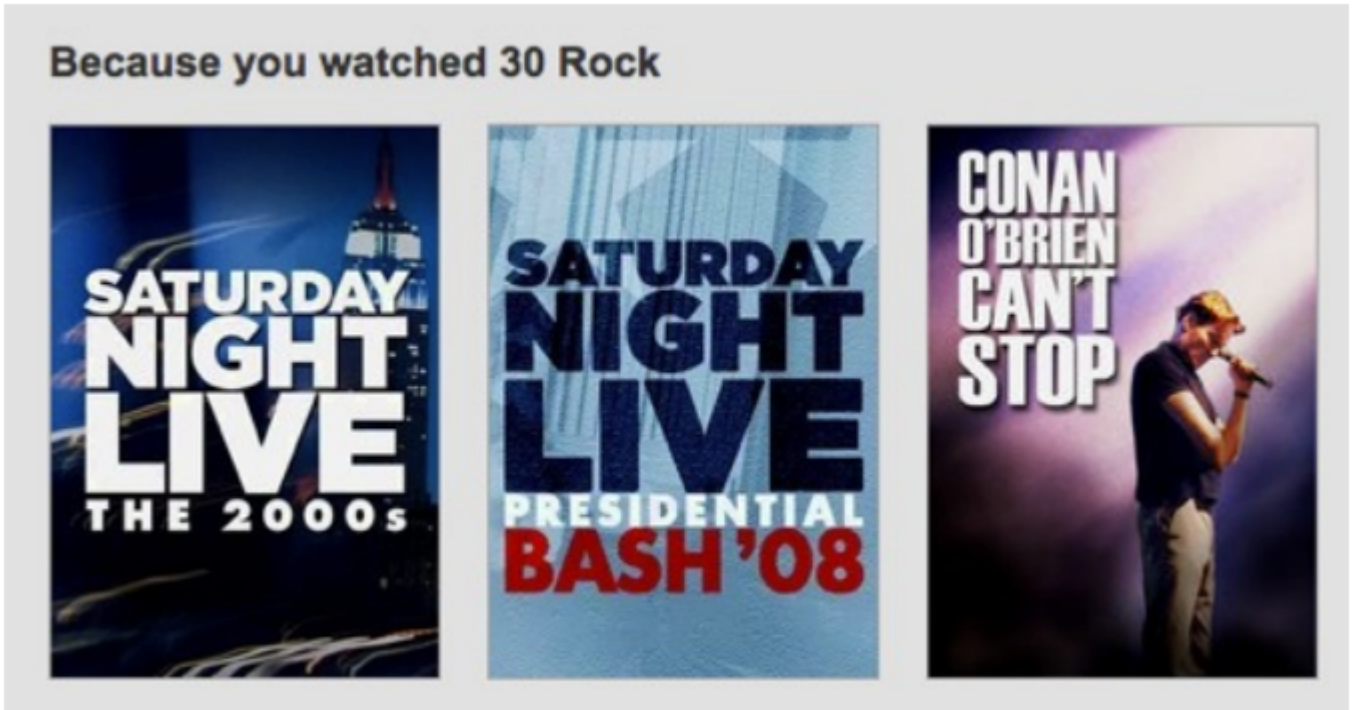
Free Radicals in Biology and Medicine
Barry Halliwell, John Gutteridge
Paperback
★★★★★ (6)
~~$90.00~~ $75.78

Nonsense on Stilts: How to Tell...
› Massimo Pigliucci
Paperback
★★★½☆ (35)
~~$20.00~~ $11.94

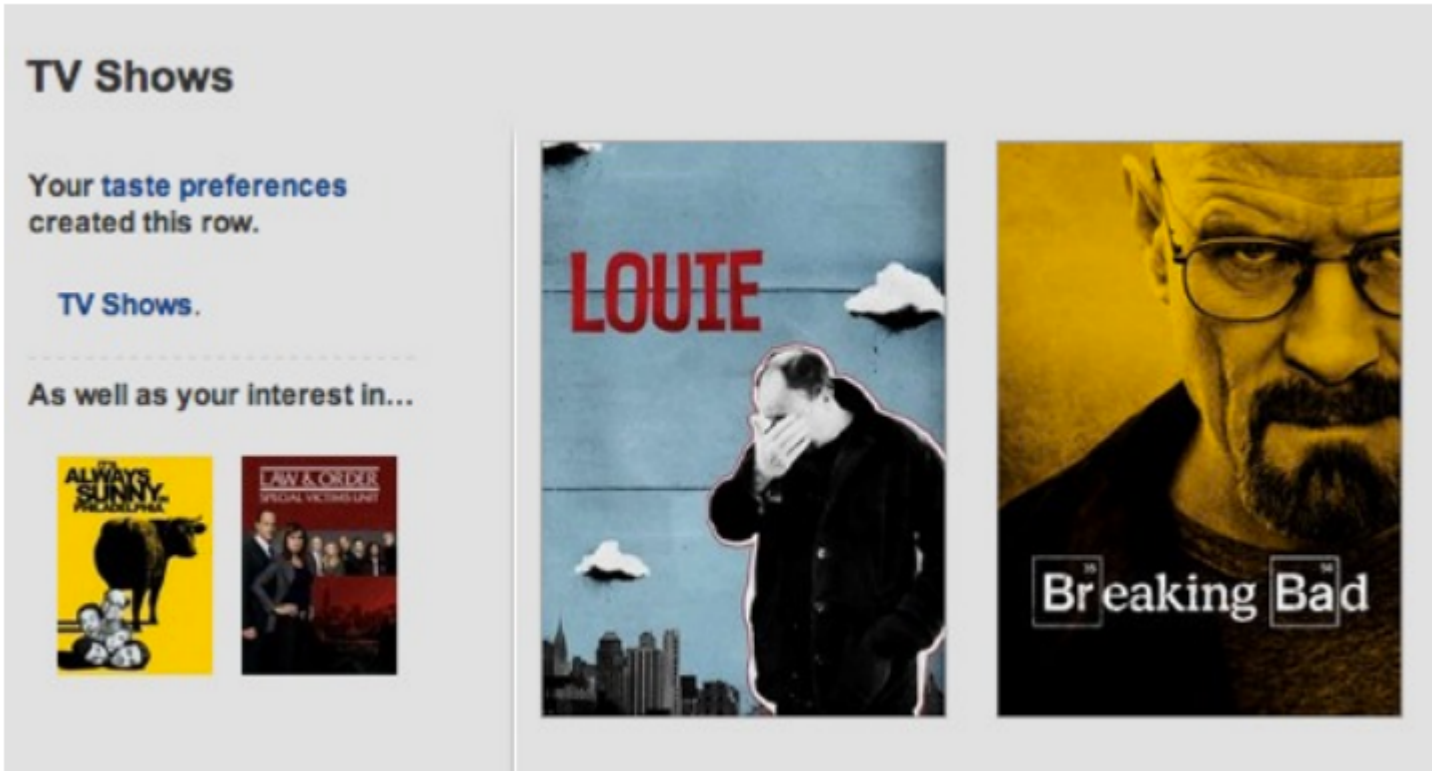
RECOMMENDATION SYSTEMS



RECOMMENDATION SYSTEMS



RECOMMENDATION SYSTEMS



RECOMMENDATION SYSTEMS

- The purpose of a recommendation system is to decide whether or not a user will be interested in a particular item

RECOMMENDATION SYSTEMS

- The purpose of a recommendation system is to decide whether or not a user will be interested in a particular *item*
- Items may be products, events, movies, songs, etc.

RECOMMENDATION SYSTEMS

- The purpose of a recommendation system is to decide whether or not a user will be interested in a particular item
- Items may be products, events, movies, songs, etc.
- Recommendation can be accomplished by predicting ratings of items for users, or by predicting whether or not the user will be interested in the item

RECOMMENDATION SYSTEMS

- The purpose of a recommendation system is to decide whether or not a user will be interested in a particular item
- Items may be products, events, movies, songs, etc.
- Recommendation can be accomplished by predicting ratings of items for users, or by predicting whether or not the user will be interested in the item
- Recommendation is ***not*** a machine learning task like multi-class classification or clustering

RECOMMENDATION SYSTEMS

There are two main approaches to building recommendation systems:

RECOMMENDATION SYSTEMS

There are two main approaches to building recommendation systems:

- In ***content-based filtering*** systems, items and users are mapped into a ***feature space***, and recommendations are learned from these feature representations

RECOMMENDATION SYSTEMS

There are two main approaches to building recommendation systems:

- In ***content-based filtering*** systems, items and users are mapped into a feature space, and recommendations are learned from these feature representations
- In contrast, the only data under consideration in ***collaborative filtering*** systems are the users' ratings for items

CONTENT-BASED FILTERING

CONTENT-BASED FILTERING

One of the main approaches to content-based filtering:

- Map users and items to same feature space, compute distance between a user and item

CONTENT-BASED FILTERING

Item vectors measure the degree to which the item is described by each feature, and user vectors measure a user's preference for each feature.

CONTENT-BASED FILTERING

Item vectors measure the degree to which the item is described by each feature, and user vectors measure a user's preference for each feature.

Toy Story -> (Comedy: 1, Animated: 1, Mafia: 0)
Godfather -> (Comedy: 0, Animated: 0, Mafia: 1)

CONTENT-BASED FILTERING

Item vectors measure the degree to which the item is described by each feature, and user vectors measure a user's preference for each feature.

Toy Story -> (Comedy: 1, Animated: 1, Mafia: 0)
Godfather -> (Comedy: 0, Animated: 0, Mafia: 1)

User 1 -> (Comedy: 1, Animated: 0, Mafia: 0)

CONTENT-BASED FILTERING

features = (big box office, intended for kids, famous actors)

Finding Nemo = (5, 5, 2)

Mission Impossible = (3, -5, 5)

Jiro Dreams of Sushi = (-4, -5, -5)

CONTENT-BASED FILTERING

features = (big box office, intended for kids, famous actors)

Finding Nemo = (5, 5, 2)

Mission Impossible = (3, -5, 5)

Jiro Dreams of Sushi = (-4, -5, -5)

Jason = (-3, 2, -2)

CONTENT-BASED FILTERING

features = (big box office, intended for kids, famous actors)

Finding Nemo = (5, 5, 2) (-3*5 + 2*5 -2*2) = -9

Mission Impossible = (3, -5, 5) (-3*3 - 2*5 -2*5) = -29

Jiro Dreams of Sushi = (-4, -5, -5) (3*4 - 2*5 + 2*5) = +12

Jason = (-3, 2, -2)

EXAMPLES OF CONTENT-BASED FILTERING

- Pandora uses content-based filtering.

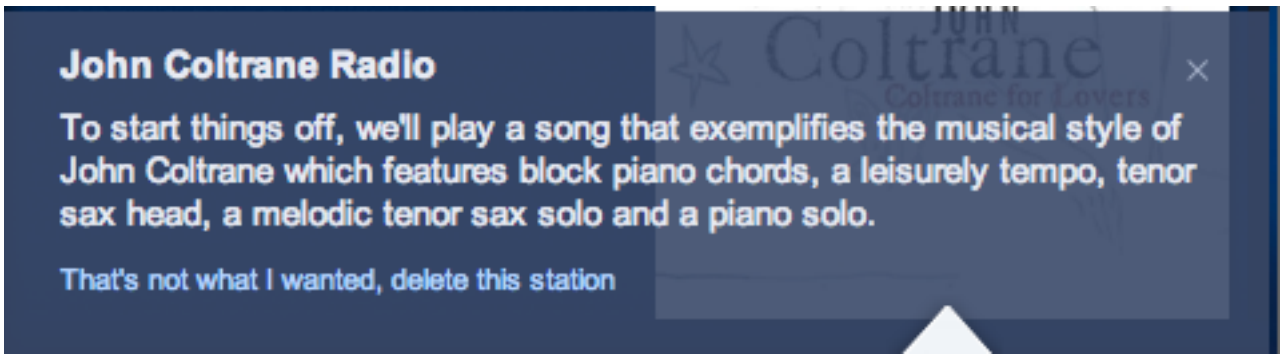
EXAMPLES OF CONTENT-BASED FILTERING

- Pandora uses content-based filtering.
- Pandora maps songs into a feature space using “genes” designed by the Music Genome Project.

EXAMPLES OF CONTENT-BASED FILTERING

- Pandora uses content-based filtering.
- Pandora maps songs into a feature space using “genes” designed by the Music Genome Project.
- Songs similar to the initial selection are assigned to the station.

Pandora


John Coltrane Radio
To start things off, we'll play a song that exemplifies the musical style of John Coltrane which features block piano chords, a leisurely tempo, tenor sax head, a melodic tenor sax solo and a piano solo.
That's not what I wanted, delete this station

ADVANTAGES OF CONTENT-BASED FILTERING

- Previous ratings are not required

DISADVANTAGES OF CONTENT-BASED FILTERING

- You must map each item into a feature space

DISADVANTAGES OF CONTENT-BASED FILTERING

- You must map each item into a feature space
- It is hard to create cross-content recommendations (e.g., books/music/films)

COLLABORATIVE FILTERING

COLLABORATIVE FILTERING

With Collaborative Filtering the goal is to predict how users will rate items that they have not yet rated.

COLLABORATIVE FILTERING

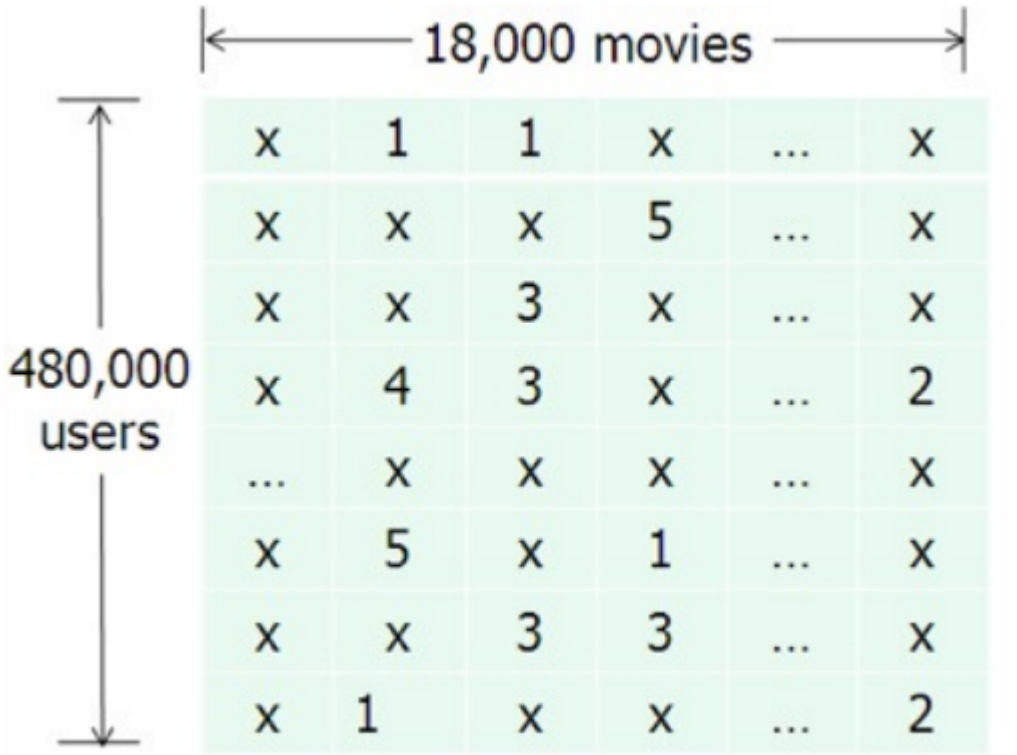
With Collaborative Filtering the goal is to predict how users will rate items that they have not yet rated.

Collaborative filtering methods do not represent the users and items in a feature space; they only use the existing user-item ratings.

COLLABORATIVE FILTERING

In the typical CF system, the dataset is a ratings matrix whose columns correspond to items, and whose rows correspond to users.

COLLABORATIVE FILTERING



COLLABORATIVE FILTERING

Collaborative filtering can be done in two different ways.

COLLABORATIVE FILTERING

Collaborative filtering can be done in two different ways.

Item-based (Neighbor or Memory-Based) CF uses an item-item similarity matrix.

COLLABORATIVE FILTERING

Collaborative filtering can be done in two different ways.

Item-based (Neighbor or Memory-Based) CF uses an item-item similarity matrix.

	Item1	Item2	...	ItemN
Item1	1	.8	0	0
Item2	0.1	1	0	0
...	0.6	0	1	0.1
ItemN	0	0	0.3	1

COLLABORATIVE FILTERING

Similarity could be as simple as the % of users who liked X who also liked Y.

COLLABORATIVE FILTERING

Similarity could be as simple as the % of users who liked X who also liked Y.

Recommendations are then made to a user for items most similar to those that the user has already rated highly.

COLLABORATIVE FILTERING

Item-based CF methods are popular and easy to understand, but they don’t scale well.

COLLABORATIVE FILTERING

Item-based CF methods are popular and easy to understand, but they don’t scale well. Why not?

COLLABORATIVE FILTERING

Item-based CF methods are popular and easy to understand, but they don't scale well. Why not?

→ An item-item similarity matrix can get pretty big!

e.g. for all of Amazon's products, the number of 2-combinations of a set of N items = N! / (2! * N-2!)

COLLABORATIVE FILTERING

Model-based CF techniques use matrix decomposition to find deeper structure in the ratings data.

COLLABORATIVE FILTERING

Model-based CF techniques use matrix decomposition to find deeper structure in the ratings data.

For example, we could decompose the ratings matrix via ***SVD*** to reduce the dimensionality and extract latent variables.

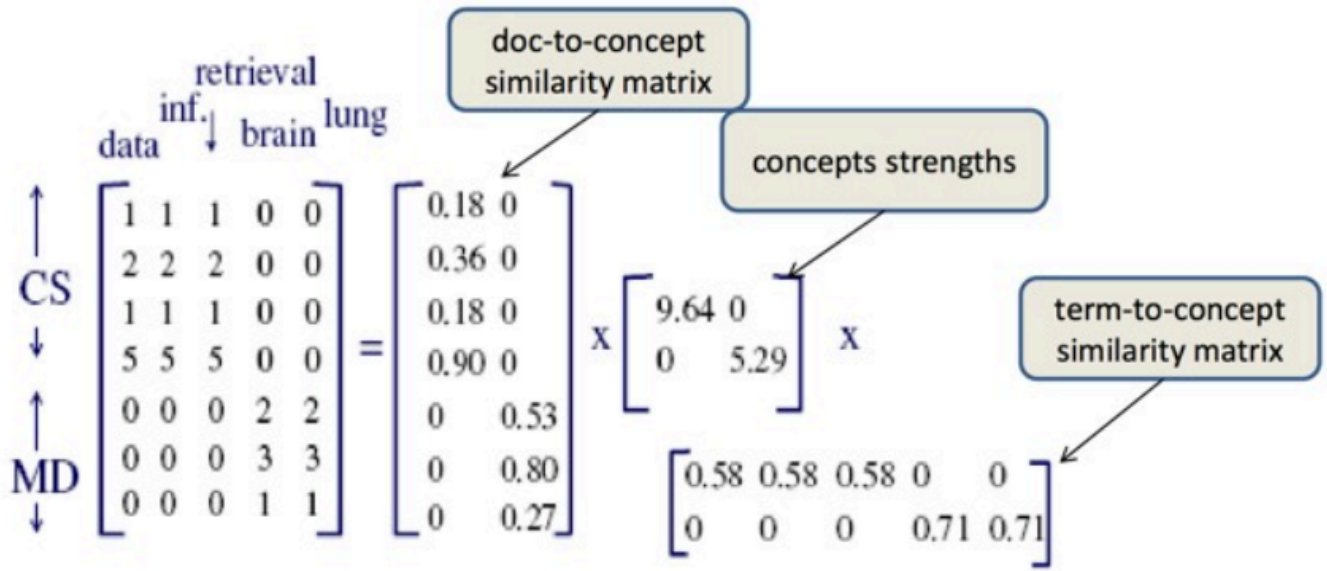
COLLABORATIVE FILTERING

Model-based CF techniques use matrix decomposition to find deeper structure in the ratings data.

For example, we could decompose the ratings matrix via ***SVD*** to reduce the dimensionality and extract latent variables.

$$\underset{(n \times d)}{A} = \underset{(n \times k)}{U} \; \underset{(k \times k)}{\Sigma} \; \underset{(k \times d)}{V^T}$$

COLLABORATIVE FILTERING



COLLABORATIVE FILTERING

Once we identify the latent variables in the ratings matrix, we can express both users and items in terms of these latent variables.

COLLABORATIVE FILTERING

Values in the item vectors represent the degree to which an item exhibits a given feature, and values in the user vectors represent user preferences for a given feature

COLLABORATIVE FILTERING

Values in the item vectors represent the degree to which an item exhibits a given feature, and values in the user vectors represent user preferences for a given feature

Ratings are constructed by taking dot products of user and item vectors in the latent feature space.

COLLABORATIVE FILTERING

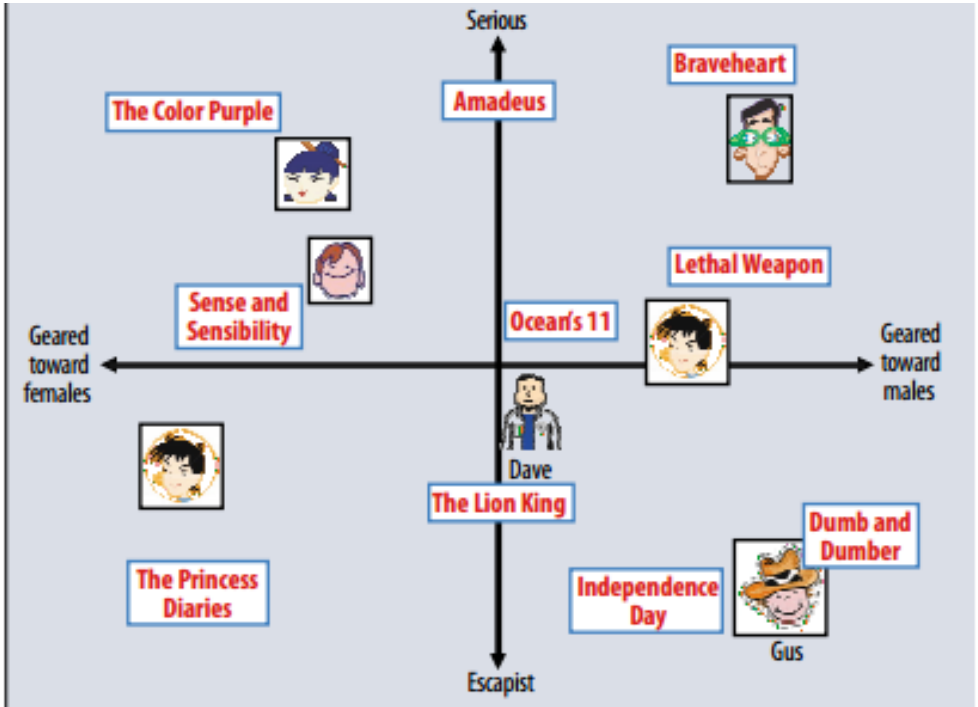


Figure 2. A simplified illustration of the latent factor approach, which characterizes both users and movies using two axes—male versus female and serious versus escapist.

ADVANTAGES OF COLLABORATIVE FILTERING

- CF is domain independent and requires no explicit user or item profiles to be created.

ADVANTAGES OF COLLABORATIVE FILTERING

- Domain independent and requires no explicit user or item profiles to be created
- Combines predictive accuracy and (relative) scalability

ADVANTAGES OF COLLABORATIVE FILTERING

- Won the Netflix prize!
- Collaborative filtering methods are generally regarded as the state-of-the-art in recommendation technology

DISADVANTAGES OF COLLABORATIVE FILTERING

- Lots of (high-dimensional) ratings data is needed

DISADVANTAGES OF COLLABORATIVE FILTERING

- Lots of (high-dimensional) ratings data is needed
- The data is typically very sparse (in the Netflix prize dataset, 99% of possible ratings were missing)

DISADVANTAGES OF COLLABORATIVE FILTERING

- Lots of (high-dimensional) ratings data is needed
- The data is typically very sparse (in the Netflix prize dataset, 99% of possible ratings were missing)
- The ***cold start problem***:

DISADVANTAGES OF COLLABORATIVE FILTERING

- Lots of (high-dimensional) ratings data is needed
- The data is typically very sparse (in the Netflix prize dataset, 99% of possible ratings were missing)
- The ***cold start problem***: need lots of data on new user or item before recommendations can be made

COLD START PROBLEM

The cold start problem arises because we've been relying only on ratings data, or on explicit feedback from users

Until a user rates several items, we don't know anything about his/her preferences!

COLD START PROBLEM

We can get around this by enhancing our recommendations using implicit feedback, which may include things like item browsing behavior, search patterns, purchase history, etc.

Implicit feedback leads to less accurate ratings, but the data is much denser (and less invasive to collect)

Implicit feedback can help to infer user preferences when explicit feedback is not available, therefore easing the cold start problem

HYBRID METHODS

Hybrid filtering methods provide another way to get around the cold start problem by combining filtering methods (e.g., by using content-based info to “boost” a collaborative model)

This content-based info can be item-based as above, or even user-based (e.g., demographic info)

Hybrid methods can also make the data sparsity issue easier to deal with, by broadening the set of features under consideration

DEMO