Collaborative Filtering

Angshul Majumdar

INDRAPRASTHA INSTITUTE *of*
INFORMATION TECHNOLOGY
DELHI

over 2 million titles

amazon.com

over 100,000 titles

NETFLIX

hulu

nearly 50,000 titles

iTunes

over 11 million titles

AmazonPrime

over 100,000 titles

Recommendation

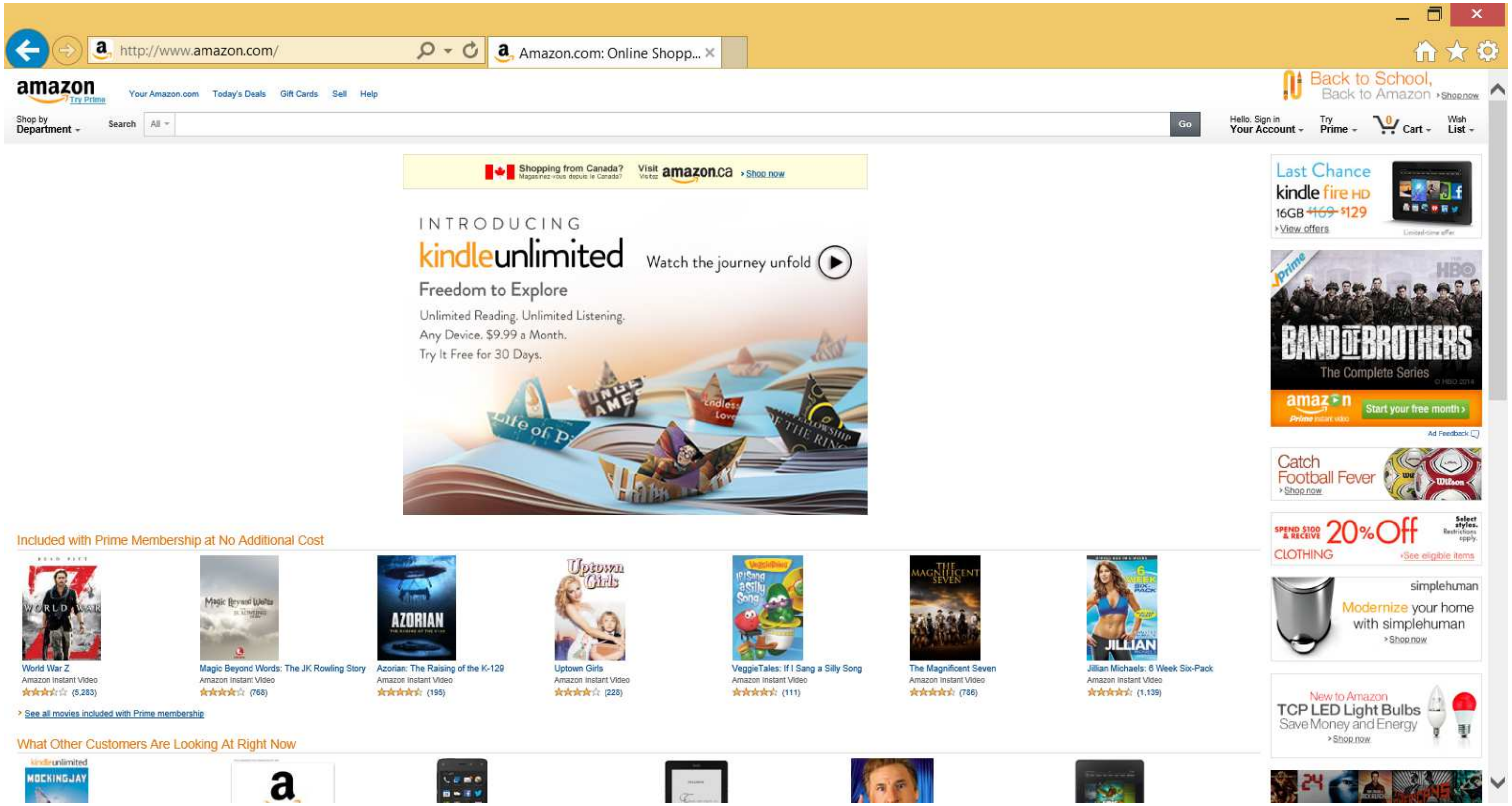
Business

- How to increase revenue?
- How to recommend items customers like?

Customer

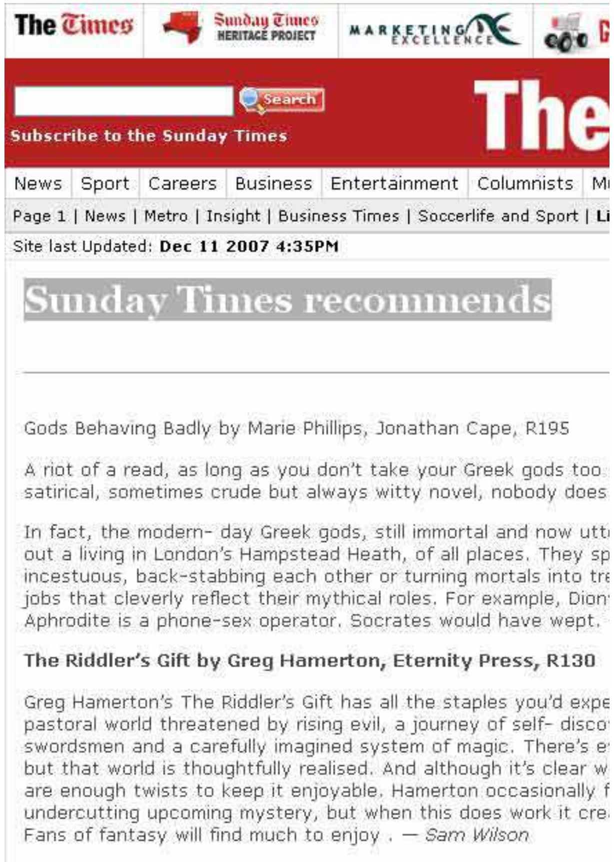
- Too many options.
- How to choose the right one?

Traditional – Advertisement (Business)



Amazon

Traditional – Recommendation (User)



The Times | Sunday Times Heritage Project | Marketing Excellence

Search

Subscribe to the Sunday Times

News | Sport | Careers | Business | Entertainment | Columnists | Mi

Page 1 | News | Metro | Insight | Business Times | Soccerlife and Sport | Li

Site last Updated: **Dec 11 2007 4:35PM**

Sunday Times recommends

Gods Behaving Badly by Marie Phillips, Jonathan Cape, R195

A riot of a read, as long as you don't take your Greek gods too satirical, sometimes crude but always witty novel, nobody does

In fact, the modern- day Greek gods, still immortal and now utt out a living in London's Hampstead Heath, of all places. They sp incestuous, back-stabbing each other or turning mortals into tre jobs that cleverly reflect their mythical roles. For example, Dion Aphrodite is a phone-sex operator. Socrates would have wept.

The Riddler's Gift by Greg Hamerton, Eternity Press, R130

Greg Hamerton's The Riddler's Gift has all the staples you'd expe pastoral world threatened by rising evil, a journey of self- disco swordsmen and a carefully imagined system of magic. There's e but that world is thoughtfully realised. And although it's clear w are enough twists to keep it enjoyable. Hamerton occasionally f undercutting upcoming mystery, but when this does work it cre Fans of fantasy will find much to enjoy . — *Sam Wilson*

P.08 movies

REVIEW: MILLION DOLLAR ARM

From left, Jon Hamm Madhur Mittal, Suraj Sharma and Pitobash in a scene from "Million Dollar Arm." (AP Photo/Disney, John Johnson)

"MILLION DOLLAR ARM IS AN ENTERTAINING FAMILY-FRIENDLY STORY THAT WHILE EFFECTIVE, SEEMS JUST A BIT BELOW EVERYONE INVOLVED IN IT."

By DAVE VOIGT
Criticize This!

One of those "Inspired by Real Events" movies, ***Million Dollar Arm*** is admittedly a little by the numbers as it tells your prototypical feel good sports story, but it works well enough thanks to some strong performances.

Failing sports agent JB Bernstein (*Jon Hamm*) concocts a scheme to find baseball's next great pitching ace. JB travels to India to produce a reality show competition called "Million Dollar Arm." With the help of a retired baseball scout (*Alan Arkin*) he discovers two 18-year-old boys who have no idea about playing baseball, yet have a knack for throwing a fastball. As the boys learn the finer points of baseball, JB learns valuable life lessons about teamwork, commitment and most importantly, family.

Million Dollar Arm is an entertaining family-friendly story that while effective, seems just a bit below everyone involved in it.

Director ***Craig Gillespie*** is a serviceable, yet unremarkable hand in the director's chair. He delivers a quick-moving narrative that never really lags and is ultimately a well balanced piece of storytelling.

Writer ***Thomas McCarthy***, who is building himself a solid little track record with films like ***The Station Agent*** and ***Win Win***, puts together a tight script with crisp dialogue that works from beginning to end, never giving us characters that feel forced or hackneyed which allows some solid performances to come out.

Jon Hamm works his magic as the steely jawed leading man and he works as the vapid agent who learns an important life lesson. Even when we aren't supposed to like him, he stills keeps us engaged and rooting for him. Unfortunately, after Hamm, the balance of the ensemble really just doesn't have a lot to do.

Alan Arkin, ***Aasif Mandvi***, ***Bill Paxton*** and ***Lake Bell*** all help to hammer home the ultimate feel good message about what is truly important in life. It's a nice message, but the film lacks emotional weight with no genuine stakes to make us feel like anything other than the happy ending that we are all expecting is coming down the pipe.

Ultimately, ***Million Dollar Arm*** works because of the talent involved and it provides the fluffy, feel good entertainment that it sets out to deliver, but it's not the kind of film that will stay with you long after you've left the theatre either, making it nice but not enough.

Criticize This!

For more movie news and reviews, visit criticizethis.ca

THE MONTHLY FASHION BOOK

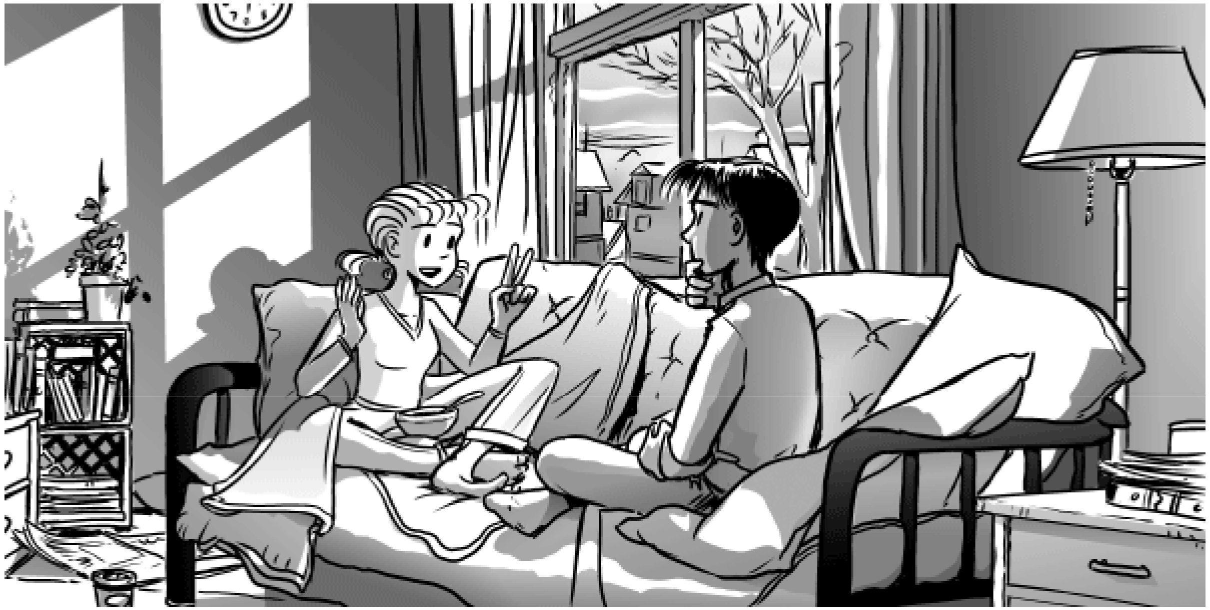
APRIL PATTERNS ISSUED IN MARCH, 1920

THE PICTORIAL REVIEW COMPANY

Don't Miss the Advance Spring Fashions in Pictorial Review for April

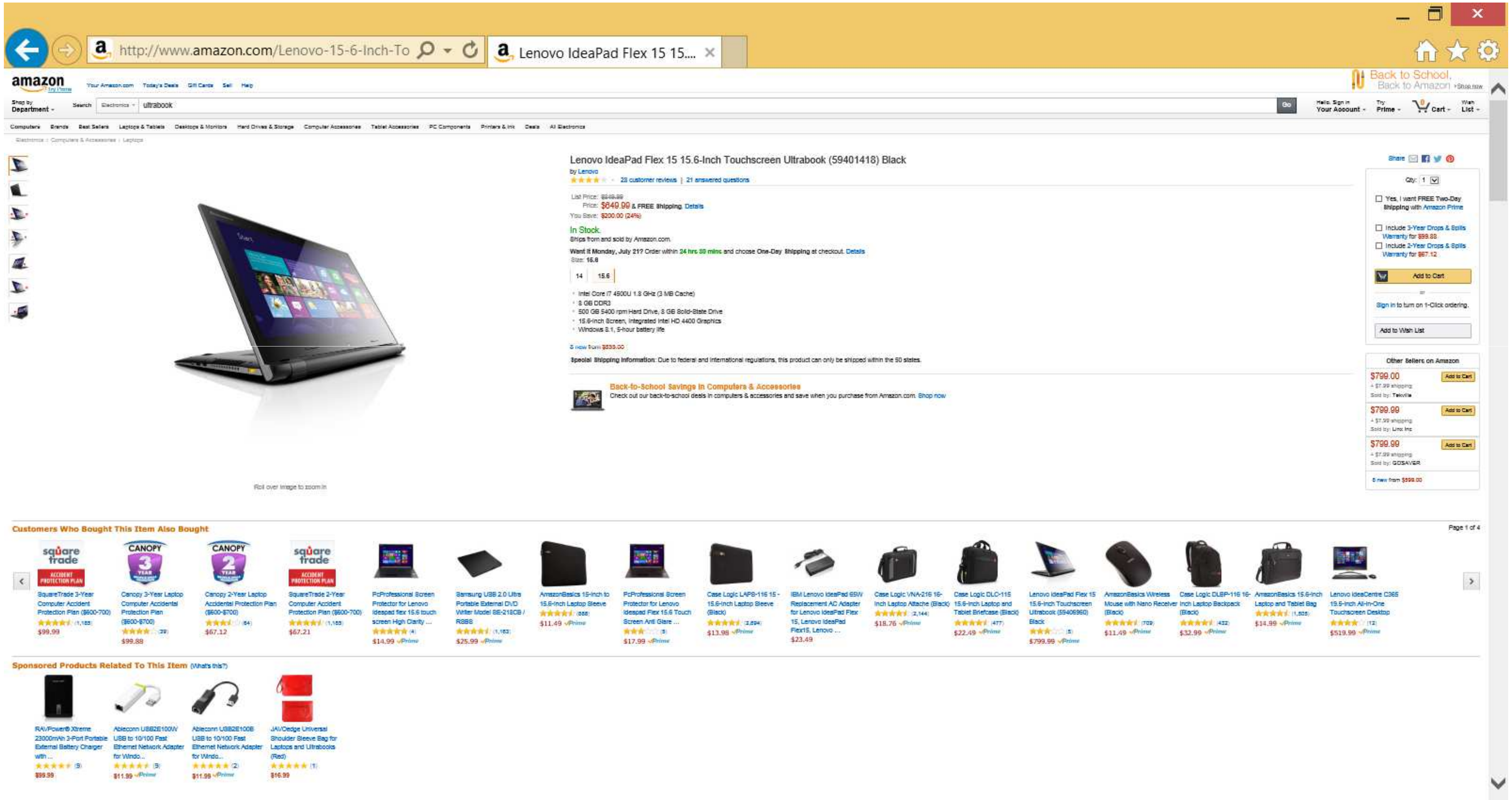
Newspaper reviews

Traditional – Recommendation (User)



Recommendations / suggestions from friends

Recommender Systems - Example



NEIGHBOURHOOD BASED METHOD

How to recommend?

- Emulate traditional ‘friendly’ recommendations on a large scale.

- Problem – to see whether to recommend you a particular item or not.

  1. Find similar users – friends; this is based on your prior choices.
  2. See if they have liked the item (movie, music, clothing, etc...) or not.
  3. Recommend based on their choice.

User-User Recommendations

From IMDB Top 10	U1	U2	U3	U4	U5	U6	U7	U8
Shawshank Redemption	3.5	2	5	3	-	-	5	3
Godfather	2	3.5	1	4	4	4.5	2	-
Pulp Fiction	-	4	1	4.5	1	4	-	-
The Good, the Bad and the Ugly	4.5	???	3	-	4	5	3	5
12 Angry Men	5	2	5	3	-	5	5	4
The Dark Knight	1.5	3.5	1	4.5	-	4.5	4	2.5
Schindler's List	2.5	-	-	4	4	4	5	3
The Lord of the Rings	2	3	-	2	1	4	-	-

The ratings of user's on popular movies

Want to find U2's rating on The Good, the Bad and the Ugly

User-User Recommendations

From IMDB Top 10	U1	U2	U3	U4	U5	U6	U7	U8
Shawshank Redemption	3.5	2	5	3	-	-	5	3
Godfather	2	3.5	1	4	4	4.5	2	-
Pulp Fiction	-	4	1	4.5	1	4	-	-
~~The Good, the Bad and the Ugly~~	~~4.5~~	~~???~~	~~3~~	~~-~~	~~4~~	~~5~~	~~3~~	~~5~~
12 Angry Men	5	2	5	3	-	5	5	4
The Dark Knight	1.5	3.5	1	4.5	-	4.5	4	2.5
Schindler's List	2.5	-	-	4	4	4	5	3
The Lord of the Rings	2	3	-	2	1	4	-	-

To find the 'similarity' between users – do not consider the item of interest

Finding Similarity (Cosine)

	U1	U2	U3	U4	U5	U6	U7	U8
U1	1	0.63	0.86	0.76	0.47	0.75	0.91	0.89
U2	0.63	1	0.56	0.89	0.47	0.86	0.55	0.47
U3			1	0.61	0.12	0.49	0.79	0.81
U4				1	0.68 -	0.91	0.80	0.72
U5					1	0.67	0.49	0.32
U6						1	0.69	0.64
U7							1	0.97

Finding Similarity (Cosine)

We want to find users similar to U2 ...

	U1	U2	U3	U4	U5	U6	U7	U8
U1	1	0.63	0.86	0.76	0.47	0.75	0.91	0.89
U2	0.63	1	0.56	0.89	0.47	0.86	0.55	0.47
U3			1	0.61	0.12	0.49	0.79	0.81
U4				1	0.68 -	0.91	0.80	0.72
U5					1	0.67	0.49	0.32
U6						1	0.69	0.64
U7							1	0.97

K-Nearest Neighbour

- 3-nearest neighbours – U6, U1 and U3.

- Ratings is estimated as a weighted mean (weighted by similarity scores), i.e.

$$\frac{0.86 \times 5 + 0.63 \times 4.5 + 0.56 \times 3}{0.86 + 0.63 + 0.56} = 4.3$$

User-User Recommendations

From IMDB Top 10	U1	U2	U3	U4	U5	U6	U7	U8
Shawshank Redemption	3.5	2	5	3	-	-	5	3
Godfather	2	3.5	1	4	4	4.5	2	-
Pulp Fiction	-	4	1	4.5	1	4	-	-
The Good, the Bad and the Ugly	4.5	???	3	-	4	5	3	5
12 Angry Men	5	2	5	3	-	5	5	4
The Dark Knight	1.5	3.5	1	4.5	-	4.5	4	2.5
Schindler's List	2.5	-	-	4	4	4	5	3
The Lord of the Rings	2	3	-	2	1	4	-	-

0.63 0.56 0.86

Summarizing …

- Filling an unknown rating is a linear interpolation problem

$$\hat{r}_{i,j} = \sum_{k \in R(i)} w_{i,k} r_{k,j}$$

  - $w_{i,k}$ – normalized similarity weight between user i & k
  - $r_{k,j}$ – user k's ratings on item j
  - R(i) – user's in the neighbourhood of i

RECOMMENDATION AS CLASSIFICATION

A Toy Example

Users	Sherlock	Game of Thrones	TBBT	Modern Family	Breaking Bad
Denny	Like	Dislike	Like	Like	Dislike
Allan	Dislike	Like	Like	Like	Dislike
Shirley	Like	Like	Dislike	Dislike	Like
Paul	Dislike	Dislike	Like	Like	???

Classification Problem

- Find Paul’s choice on ‘Breaking Bad’.

- Classification Problem – Supervised Learning
  - Features for each sample
  - Class labels for each sample

- Item based approach – each item is a sample.

Item Based Classification

- Labels – Paul's choice on each item

Sherlock	Game of Thrones	TBBT	Modern Family	Breaking Bad
-1	-1	+1	+1	???

- Features – Denny, Allan and Shirley's rating on each item.

Sherlock	Game of Thrones	TBBT	Modern Family	Breaking Bad
+1	-1	+1	+1	-1
-1	+1	+1	+1	-1
+1	+1	-1	-1	+1

Nearest Neighbour

- Given the test sample, find the nearest training sample(s).
- Assign the test sample to the class of the nearest test sample(s).

- Binary classification problem – Like / Dislike

Hamming Distance

- The Hamming distance between two strings of equal length is the number of positions at which the corresponding symbols are different.

- The Hamming distance between:
  - "ka**rol**in" and "ka**thr**in" is 3.
  - 1011101 and 1001001 is 2.

Working it out …

- Hamming Distance between Breaking Bad and others.

Sherlock	Game of Thrones	TBBT	Modern Family	Breaking Bad
+1	-1	+1	+1	-1
-1	+1	+1	+1	-1
+1	+1	-1	-1	+1
1	**1**	**3**	**3**	**0**

Solution

- Sherlock and GoT are the nearest to Breaking Bad.
- Based on this observation, Breaking Bad can be assigned a class label to either of these two. For our problem it is '-1'

Sherlock	Game of Thrones	TBBT	Modern Family	Breaking Bad
-1	-1	+1	+1	**-1**

- Paul will 'dislike' Breaking Bad

LATENT FACTOR MODEL

How do we choose a movie?

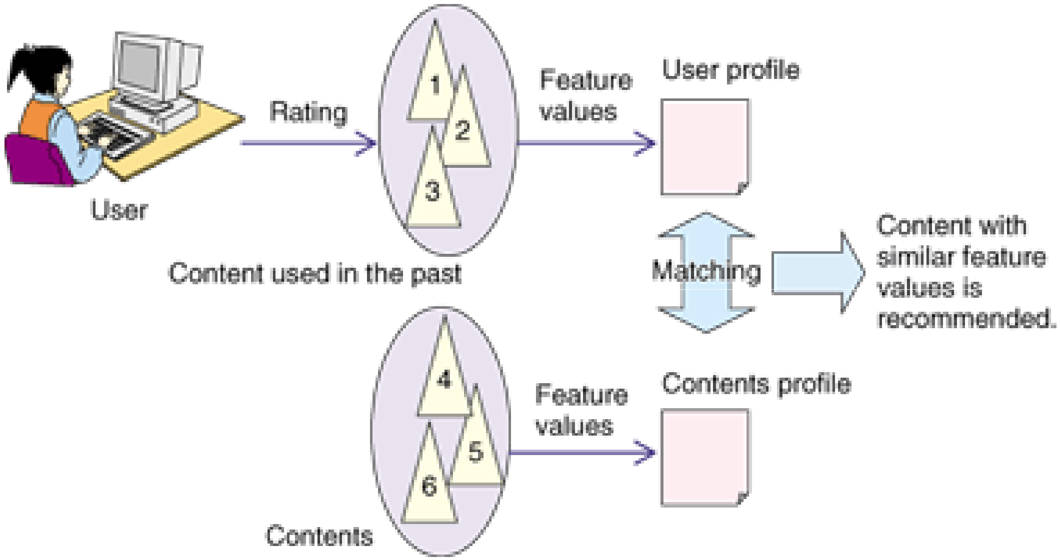
- Genre (Action, Thriller, Western, Drama ...)
- Actor
- Director (Tarantino, Nolan, Bergman ...)

- There are only a few factors that helps decide our choice.

- We may be able to find these factors and match them between users and items ...

Content Based Filtering

Preceded Collaborative Filtering.



- From previous (historical) data, find out the ‘preference’ of users.
- Match the preference with items contents for recommendation.

Cons …

- Content based filtering required the factors to be exactly known – privacy issues
- This is not exactly data driven and requires a lot of domain knowledge.
- Even then there is a possibility that ‘some factors’ were not being considered.

Latent Factor Model

- Assumes that the factors affecting the choices are hidden / latent.
- These factors need not be exactly known.
  - The item-j is characterized by m-factors

$$v_j = [v_j^{(1)}, v_j^{(2)}, \ldots . v_j^{(m)}]^T$$

  - The user-a is characterized by his / her affinity towards these factors

$$u_i = [u_i^{(1)}, u_i^{(2)}, \ldots . u_i^{(m)}]^T$$

Mathematical Formalism

- Latent factor model assumes that the rating of a user on an item is just an inner-product of the users' and items' latent factors.

$$r_{i,j} = u_i^T v_j$$

Biases

- Some user's are critics, some are over positive (e.g. U6). Critics tend to rate lower than the average
  - Negative user bias

- Similarly items like Titanic or Lord of the Rings, tend to get rated higher than normal.
  - Positive item bias

Baseline /Bias Modelling

$$r_{i,j} = \mu + b_i + b_j$$

- Global mean μ
- User bias b_i- observed deviation for user 'i'
- Item bias b_j- observed deviation for item 'j'

Baseline Estimation

- Potter Estimation

$$b_i = \frac{\sum_{i \in R(j)} (r_{i,j} - \mu)}{\lambda_1 + |R(j)|} \qquad b_j = \frac{\sum_{j \in R(i)} (r_{i,j} - \mu - b_i)}{\lambda_2 + |R(i)|}$$

- Koren Bell Estimation

$$\min_{b_i, b_j} \sum \left\{ r_{i,j} - (\mu + b_i + b_j) \right\}^2 + \lambda \left(\sum_i b_i^2 + \sum_j b_j^2 \right)$$

Formal Combined Model

- Combined Model = baseline + interaction

$$r_{i,j} = baseline(i,j) + u_i^T v_j = \mu + b_i + b_j + u_i^T v_j$$

- How do we use this model for prediction?

Estimation

- Estimation of various parameters is formulated as:

$$\min_{b,v,u} \sum \left(r_{i,j} - \mu - b_i - b_j - u_i^T v_j\right)^2 + \lambda\left(b_i^2 + b_j^2 + \|q_i\|^2 + \|v_j\|^2\right)$$

- This can be divided into following sub-problems ...

$$\text{P1:} \min_{b_i} \sum \left(r_{i,j} - \mu - b_i - b_j - u_i^T v_j\right)^2 + \lambda\left(b_i^2\right)$$

$$\text{P2:} \min_{b_j} \sum \left(r_{i,j} - \mu - b_i - b_j - u_i^T v_j\right)^2 + \lambda\left(b_j^2\right)$$

$$\text{P3:} \min_{v_j, u_i} \sum \left(r_{i,j} - \mu - b_i - b_j - u_i^T v_j\right)^2 + \lambda\left(\|q_i\|^2 + \|v_j\|^2\right)$$

SVD Algorithm (misnomer!)

- In each iteration, compute a prediction $(\hat{r}_{i,j})$
- Next, compute the prediction error $(e_{i,j} = r_{i,j} - \hat{r}_{i,j})$
- Now compute the different parameters,

$$b_i \leftarrow b_i + \gamma(e_{i,j} - \lambda b_i)$$
$$b_j \leftarrow b_j + \gamma(e_{i,j} - \lambda b_j)$$
$$u_i \leftarrow u_i + \gamma(e_{i,j} - \lambda u_i)$$
$$v_j \leftarrow v_j + \gamma(e_{i,j} - \lambda v_j)$$

- As you notice this is a stochastic gradient descent algorithm, $e_{i,j}$ is the gradient and λ is the step-size.

A holistic view

- The matrix of interactions

Items

Users

0.09	–	–	–	–	–	–	0.05	–	–
–	–	0.02	–	0.03	–	–	–	–	0.06
–	0.07	–	–	–	0.04	–	–	–	0.04
–	0.05	–	–	–	–	0.06	–	–	–
–	–	0.03	0.05	–	–	–	0.01	–	–
0.01	–	–	–	0.07	–	–	–	–	–
–	–	–	–	0.06	–	–	0.10	–	–
0.02	–	–	–	–	–	0.07	–	–	–
–	–	0.12	0.05	–	–	–	–	–	0.11
–	0.11	–	–	–	0.07	–	0.08	–	–

A low-rank model

- The matrix of ratings can be expressed as:

$$z_{i,j} = [u_i^{(1)}, u_i^{(2)}, \ldots. u_i^{(m)}] \begin{bmatrix} v_j^{(1)} \\ v_j^{(2)} \\ \ldots \\ v_j^{(m)} \end{bmatrix} \Rightarrow Z = UV^T$$

- According to our assumption, the matrix (Z – bias corrected) is of low rank (m).

Matrix Factorization

- SVD-CF is a crude one shot technique

- Better way to approach the problem ...

$$Y = M \odot Z + \eta = M \odot (UV) + \eta$$

- Solve it via Alternating Least Squares

$$Init: U_0$$

In iteration k

$$V_k = \min_V \left\| Y - M \odot (U_{k-1}V) \right\|_F^2$$

$$U_k = \min_U \left\| Y - M \odot (UV_k) \right\|_F^2$$

NNMF

- The ratings are always positive. So one can impose non-negativity constraints - NNMF
- The simplest algorithm for NNMF is to project onto the space of positive numbers in every iteration

$$Init : U_0$$

In iteration k

$$V_k = \min_V \left\| Y - M \odot (U_{k-1} V) \right\|_F^2 ; V_k = V_k^+$$

$$U_k = \min_U \left\| Y - M \odot (U V_k) \right\|_F^2 ; U_k = U_k^+$$

- However this does not apply after bias correction

RMF

- MF and NNMF solve the least squares problem

$$\min_{U,V} \left\| Y - M \odot (UV) \right\|_F^2$$

- This may result in over-fitting. The easiest way to prevent over-fitting is to add Tikhonov type regularization terms for each variable.

$$\min_{U,V} \left\| Y - M \odot (UV) \right\|_F^2 + \lambda \left(\left\| U \right\|_F^2 + \left\| V \right\|_F^2 \right)$$

- The sub-problems are still quadratic and hence easy to solve.

Sparsity in Item Matrix

- We have the matrix factorization model

$$Y - M \odot (UV) + \eta$$

- The user matrix is dense – human beings have interest in all factors.

- But the item matrix is sparse – an item cannot possess all qualities simultaneously.

BCS Type Formulation

- The prior model of RMF is not the best as it returns a dense item matrix

$$\min_{U,V} \|Y - M \odot (UV)\|_F^2 + \lambda\left(\|U\|_F^2 + \|V\|_F^2\right)$$

- We can impose sparsity on the item matrix:

$$\min_{U,V} \|Y - M \odot (UV)\|_F^2 + \lambda\left(\|U\|_F^2 + \|V\|_1\right)$$

- This is similar to the Blind Compressed Sensing formulation.

Elastic-Net BCS

- Some of the factors are always related, e.g. A Jason Statham movie (actor) is most likely to be a 'thriller' or 'action'.
- Such factors (actor / genre) are sometimes related.
- L1-norm fails to account for selection of related variables. An Elastic-Net formulation (additional L2-norm) accounts for that.

$$\min_{U,V} \|Y - M \odot (UV)\|_F^2 + \lambda_1 \|U\|_F^2 + \lambda_2 \left(\|V\|_F^2 + \|V\|_1 \right)$$

H. Zou and T. Hastie, "Regularization and variable selection via the elastic net", J. Royal Statist. Soc. B., Vol. 67 (2), pp. 301-320, 2005.

Some Results

Error Measures for 100K Dataset – 5 fold cross validation

Algo	MAE	RMSE	Time (in sec)
SGD – Koren Bell	0.7432	0.9421	150.34
BCS-CF	0.7215	09241	2.67
eNet-BCS - CF	0.7178	0.9162	2.67

Error Measures for 1M Dataset – 5 fold cross validation

Algo	MAE	RMSE	Time (in sec)
SGD – Koren Bell	0.6956	0.8763	1262.5
BCS-CF	0.6762	0.8697	31.36
eNet-BCS - CF	0.6757	0.8636	33.42

CF as Matrix Completion

- The ultimate goal is to fill the ratings matrix – we do not need the user and the item latent factor matrices.
- Indeed, we can directly solve

$$Y = M \odot R + \eta$$

- This is an under-determined problem with infinitely many solutions.
- But ... We know that Z is low-rank (rank m)

Solution ...

- Ideally one solves the rank minimization problem ...

$$\min_R rank(R) \text{ such that } \|Y - M \odot R\|_F^2 \leq \varepsilon$$

- However, this is an NP hard problem ...
- Instead one is almost guaranteed a solution by relaxing the problem to Nuclear Norm minimization

$$\min_R \|R\|_{NN} \text{ such that } \|Y - M \odot R\|_F^2 \leq \varepsilon$$

SVD Free Matrix Recovery

- The main challenge is to compute the SVD in every iteration of the SVS.
- Substitute the Nuclear norm by its equivalent Ky-Fan norm

$$\|X\|_{NN} = Tr(X^T X)^{\frac{1}{2}}$$

- Leads to a quadratic problem.
- Can be efficiently solved using Cholesky decomposition.

Some Results

Error Measures for 100K Dataset – 5 fold cross validation

Algo	MAE	Time (in sec)
SGD – Koren Bell	0.7432	150.34
Matrix Completion	0.7391	61.5
SVD Free MC	0.7400	61.5

Divide-and-conquer

- Split the ratings matrix into a number of column sub-matrices.
- Complete each column sub-matrix using some matrix completion / factorization technique.
- Combine these column sub-matrices into a full matrix by projecting them onto the column-space of a randomly chosen sub-matrix.

Some Results

Result for Divide and Conquer on 100K dataset

Algo	MAE	Time (in sec)
eNet-BCS	0.7178	2.67
eNet-BCS – D&C	0.7181	0.78 (4 partitions)

Results for 10M dataset

Algo	MAE	Time
eNet-BCS – D&C	0.6185	170.61
APG	0.6307	276.05
OptSpace	0.6437	1159.89
SVT	0.6645	265.74

Incorporating Metadata

- During the ‘sign up’ process the portal collects demographic information about age, sex, occupation etc.
- Similarly metadata is associated with items as well (actors, director, genre etc.)

- How to use this metadata information to improve collaborative filtering?
  - So far only used to address the ‘cold-start’ problem.

Similarity inducing penalty

- Remember Fisher Linear Discriminant Analysis – reduce within class scatter and increase between class scatter.
- One can assume that similar groups (say age, sex, occupation) will have similar tastes.
- Introduce a penalty that minimizes within class tastes.

$$\min_{Z} \left\| Y - M \odot R \right\|_F^2 + \lambda \left\| R \right\|_* + \sum_{G \in Groups} \left(\mu_G \sum_{u \in g} \left\| Z_{u,:} - m_{g,:} \right\|_2^2 \right)$$

Some Results

MAE for 100k Dataset

Algo	MAE
MC-group (Age)	0.7264
MC-group (Occu)	0.7310
MC-group (Age-Occu)	0.7206
BCD-NMF	0.7582
Graph-NMF	0.7577

MAE for 1M Dataset

Algo	MAE
MC-group (Age)	0.6772
MC-group (Occu)	0.6812
MC-group (Age-Occu)	0.6749
BCD-NMF	0.6863
Graph-NMF	0.7233

BCD-NMF is the state-of-the-art baseline
Graph-NMF is the only previous technique that accounted for metadata

Using Class Label Consistency

- Borrow ideas from supervised learning.
- We can group together users/items by assigning them class labels based on available metadata.
- Users belonging to same age group or occupation form one class; items sharing a genre clubbed together – Each can belong to multiple classes
- Introduce class label consistency terms in MF framework – ensuring recovered latent factor vectors consistent with the class label information

Using Class Label Consistency

$$\min_{U,V,C,A} \left\| Y - M\left(UV\right) \right\|_F^2 + \lambda_u \left\| U \right\|_F^2 + \lambda_v \left\| V \right\|_1 + \mu_u \left\| W - UC \right\|_F^2 + \mu_v \left\| Q - AV \right\|_F^2$$

- W capture class label information for users

  $W_{i,j} = 1$; iff user $i \in$ class j else $W_{i,j} = 0$

- Similarly, Q defined for items

Some Results

MAE for 100k Dataset

Algo	MAE
BCS-User	0.7316
BCS-Item	0.7253
BCS-User-Item	0.7239
BCD-NMF	0.7582
Graph-NMF	0.7577

MAE for 1M Dataset

Algo	MAE
BCS-User	0.6796
BCS-Item	0.6721
BCS-User-Item	0.6709
BCD-NMF	0.6863
Graph-NMF	0.7233

Item Metadata and grouping better able to capture the classification information than user metadata

Combining item and user information simultaneously improves accuracy further