X-Informatics Case Study: e-Commerce and Life Style Informatics: Recommender Systems II: Examples and Algorithms

February 6 2013

Geoffrey Fox
gcf@indiana.edu

http://www.infomall.org/X-InformaticsSpring2013/index.html

Associate Dean for Research and Graduate Studies, School of Informatics and Computing
Indiana University Bloomington

2013

Big Data Ecosystem in One Sentence

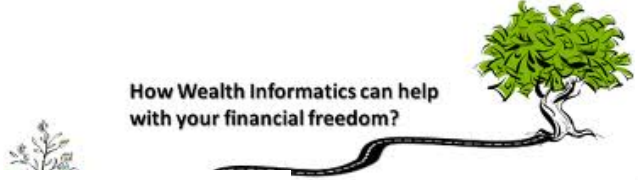
Use Clouds running Data Analytics Collaboratively processing Big Data to solve problems in X-Informatics (or e-X)

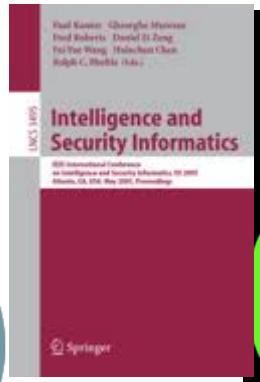
X = Astronomy, Biology, Biomedicine, Business, Chemistry, Climate, Crisis, Earth Science, Energy, Environment, Finance, Health, Intelligence, Lifestyle, Marketing, Medicine, Pathology, Policy, Radar, Security, Sensor, Social, Sustainability, Wealth and Wellness with more fields (physics) defined implicitly

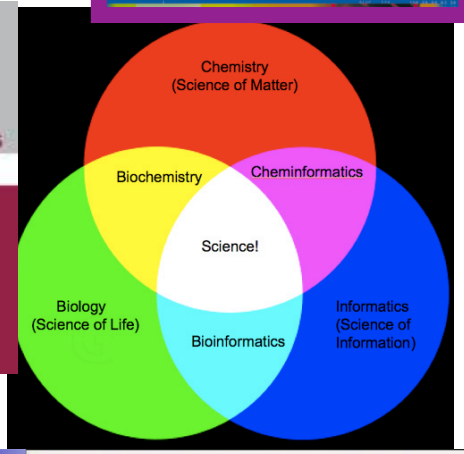
Spans Industry and Science (research)

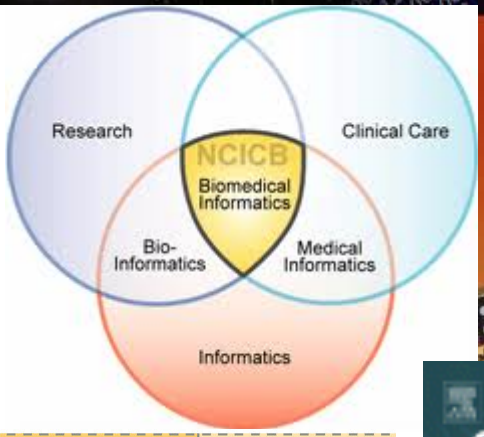
Education: Data Science see recent New York Times articles

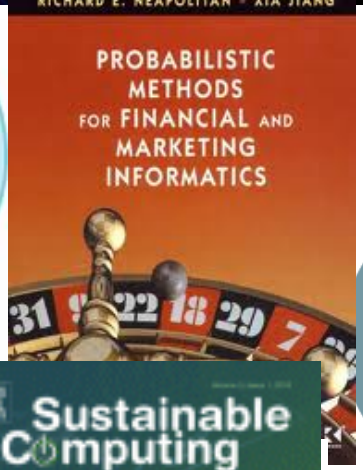
http://datascience101.wordpress.com/2013/04/13/new-york-times-data-science-articles/

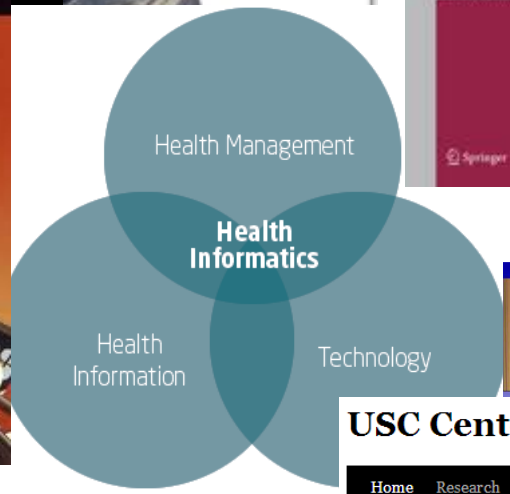


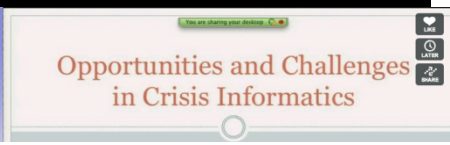


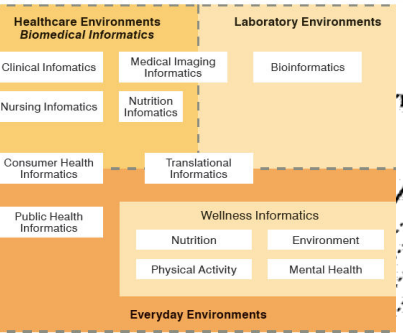


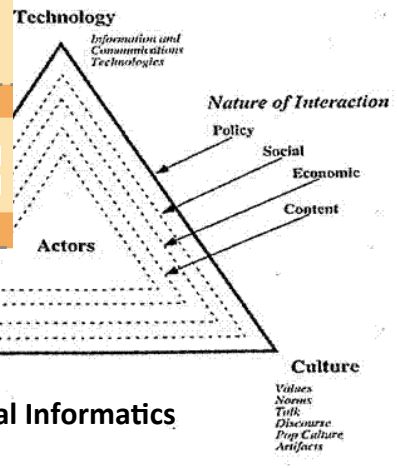








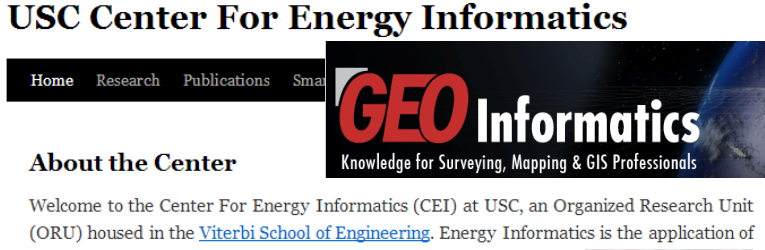


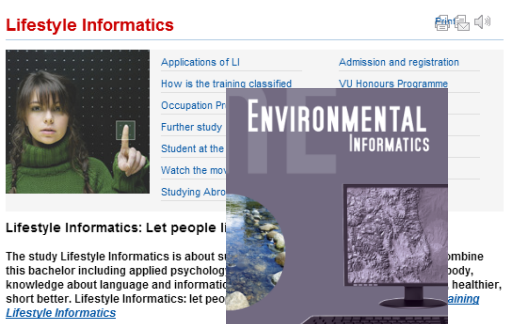


Noelia Penelope Greer (Ed.)

Business Informatics

Information technology, Management,





Recap of Recommender Systems

Overview of Problems

- Basic problem is personalized matching of items to people or perhaps collections of items to collections of people
- **People** to **products**: Online and Offline Commerce
- **People** to **People**: Social Networking
- **People** to **Jobs** or **Employers**: Job Sites
- **People+Queries** to the **Web**: Information Retrieval (search as in Bing/Google)

http://en.wikipedia.org/wiki/Recommender_system

- When viewing a product on **Amazon.com**, the store will recommend additional items based on a matrix of what other shoppers bought along with the currently selected item.
- **Pandora** uses the properties of a song or artist (a subset of the 400 attributes provided by the Music Genome Project) in order to seed a "station" that plays music with similar properties. User feedback is used to refine the station's results, deemphasizing certain attributes when a user "dislikes" a particular song and emphasizing other attributes when a user "likes" a song. This is an example of a content-based recommender system.
- **Last.fm** creates a "station" of recommended songs by observing what bands and individual tracks that the user has listened to on a regular basis and comparing those against the listening behavior of other users. Last.fm will play tracks that do not appear in the user's library, but are often played by other users with similar interests. As this approach leverages the behavior of users, it is an example of a collaborative filtering technique.
- **Netflix** offers predictions of movies that a user might like to watch based on the user's previous ratings and watching habits (as compared to the behavior of other users), also taking into account the characteristics (such as the genre) of the film.

Examples of Recommender Systems

The Google News personalization engine

News

U.S. edition | Modern

Top Stories
- Abdelaziz Bouteflika
- Silvio Berlusconi
- New Paltz
- Cory Monteith
- San Francisco 49ers
- Prescription drug
- Zach Braff
- Sarabjit Singh
- Stagecoach Festival
- Nicki Minaj

Bloomington, Indiana

source:_new_york_times_

World

Physics

California Institute...

Business

Entertainment

Sports

Spotlight

Health

Science

Barack Obama

Top Stories


NPR

With new arrest, ricin case takes a strange turn
Los Angeles Times - 4 hours ago
TUPELO, Miss. - Federal agents of all sorts invaded northeast Mississippi several days ago, on a mission: Find the man who sent a poison-laced letter to the president.


IBNLive

Gunshots fired near Italy prime minister's office, injuries
Reuters - 47 minutes ago
ROME | Sun Apr 28, 2013 6:01am EDT. ROME (Reuters) - Gunshots were fired in front of the Italian prime minister's office in Rome on Saturday as the new government of Enrico Letta was being sworn in at the president's palace around a kilometer away, RAI ...


ABC News

Supreme Court Justice Stephen Breyer breaks shoulder in bike crash
Los Angeles Times - 14 hours ago
Supreme Court Justice Stephen G. Breyer broke his shoulder in a fall from his bicycle and underwent surgery Saturday morning, according to a court spokeswoman.


New York D...

President Obama and Conan O'Brien joke during White House Correspondents ...
New York Daily News - 6 hours ago
Turning the tables, President Obama made a sea of journalists squirm in their seats Saturday night as he relentlessly mocked the media at the annual White House Correspondents' Dinner.


BBC News

Dhaka building collapse: Owner Mohammed Sohel Rana 'arrested'
BBC News - 11 minutes ago
The owner of a building that collapsed in the Bangladeshi capital Dhaka, killing hundreds of people, has been arrested, a government minister says.


Newsday

Barack Obama jokes about second term changes at White House ...
Newsday - 6 hours ago
WASHINGTON - President Barack Obama joked Saturday about his plans for a radical second-term evolution from "strapping young Socialist" to retiree golfer, all with a new hairstyle like first lady Michelle's.

World »

Bright lights, big city for transformed Pyongyang but rest of North Korea still cold ...
Washington Post - 5 hours ago
PYONGYANG, North Korea - The heart of this city, once famous for its Dickensian darkness, now pulsates with neon. Glossy new construction downtown has altered the Pyongyang skyline.

Personalize Google News

source:_new_york_ti...
World
Physics
California Institute of T...
Business
Entertainment
Sports
Health
Science
Barack Obama

U.S. +
Technology +

Add any news topic +
Examples: Astronomy, New England Patriots, White House
Advanced »

Adjust Sources

Adjust the frequency of any news source +

New York Times
Fox News
CNN
ESPN

Save | Settings | Reset | Help

Recent

http://recommenderbook.net/teaching-material/slides Chapter 2

Google News portal (1)

- Aggregates news articles from several thousand sources
- Displays them to signed-in users in a personalized way
- Collaborative recommendation approach based on
  - the click history of the active user and
  - the history of the larger community
- Main challenges
  - Vast number of articles and users
  - Generate recommendation list in real time (at most one second)
  - Constant stream of new items
  - Immediately react to user interaction
- Significant efforts with respect to algorithms, engineering, and parallelization are required

Google News portal (2)

- Pure memory-based approaches are not directly applicable and for model-based approaches, the problem of continuous model updates must be solved
- A combination of model- and memory-based techniques is used
- Model-based part: Two clustering techniques are used
  - Probabilistic Latent Semantic Indexing (PLSI) as proposed by (Hofmann 2004)
  - MinHash as a hashing method
- Memory-based part: Analyze story *co-visits* for dealing with new users
- Google's MapReduce technique is used for parallelization in order to make computation scalable

Example: Markdown Optimization in Retail

Department stores and fashion retailers clear apparel inventory by reducing prices

> "What prices do we set through the end of the season to meet inventory goals and maximize revenue?"

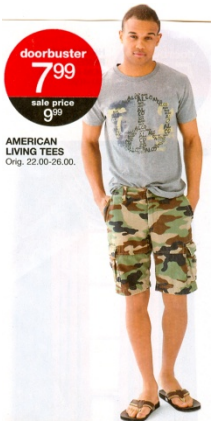
In practice

- Traditional pricing is based on business rules—not models
- Customers respond to prices, promotions at product-store level
- Weekly decisions are made at store level

http://isi2011.congressplanner.eu/pdfs/p_450457.pdf

Why an Analytical Solution?

- Huge number of decisions
  - 50,000 products/store x 2,000 stores = **100,000,000 decisions!**
- 1.5 - 3 terabytes of weekly data for two-year moving window
  - Number of units sold, price at point of sale, reference price
  - Starting and ending inventory in each store
  - Promotions, TV ads, circulars, …
- Building blocks
  - Statistical model for demand
  - Price optimization



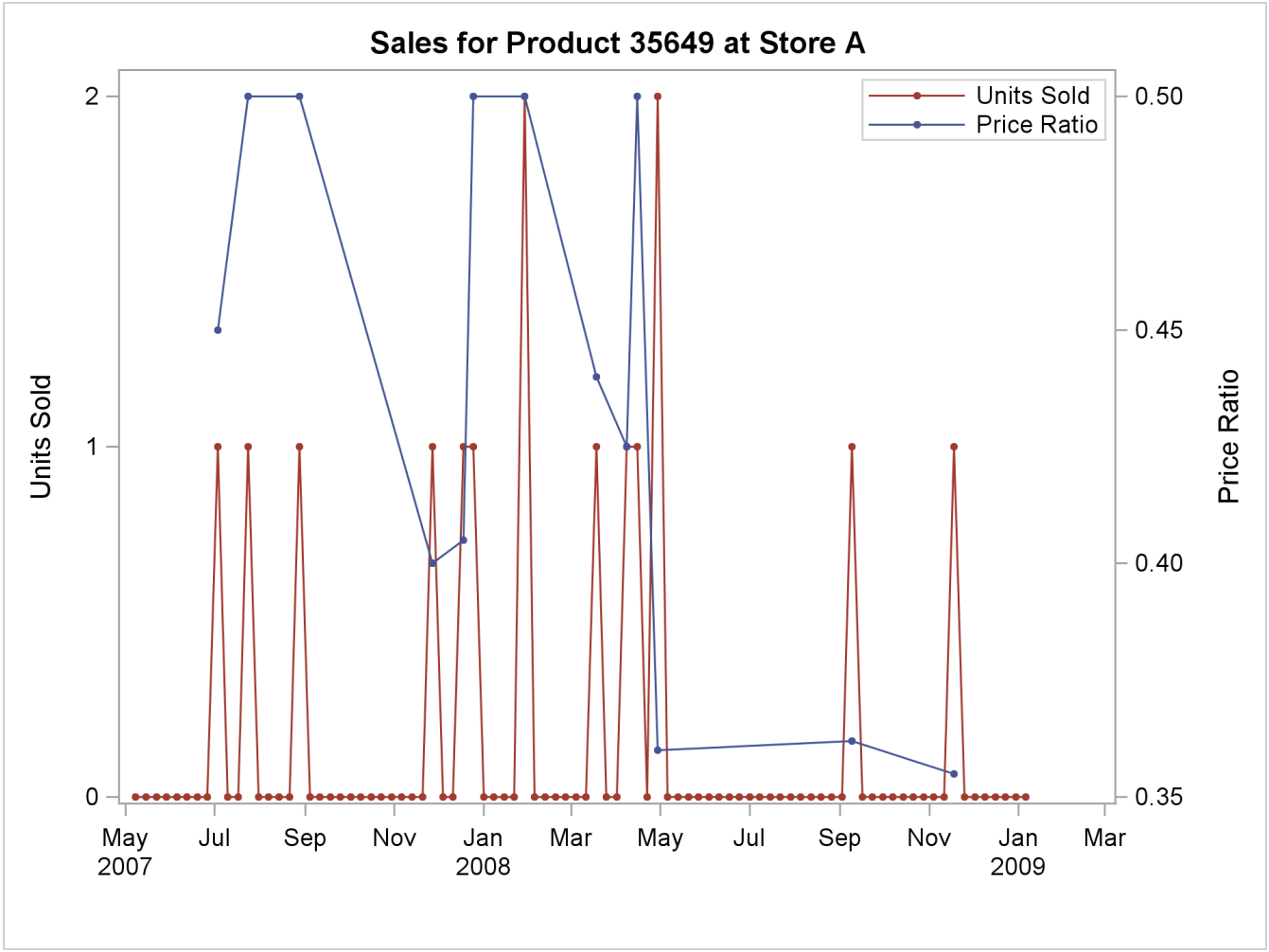
http://isi2011.congressplanner.eu/pdfs/p_450457.pdf

Modeling Considerations

- Demand depends on
  - Seasonal effects, holidays
  - “Marketing mix” effects: price, promotion, inventory
  - Product life cycle effect : introduction and phase-out
- Pricing rules and customer behavior are complex
  - Magic price points (80% is lowest markdown)
  - Discontinuities in response (difference between $1.99 and $2.00)
  - Holiday effects are confounded with promotions
- Sales data are sparse for a product within a store

http://isi2011.congressplanner.eu/pdfs/p_450457.pdf

Sparse, Noisy Data at Store-Product Level



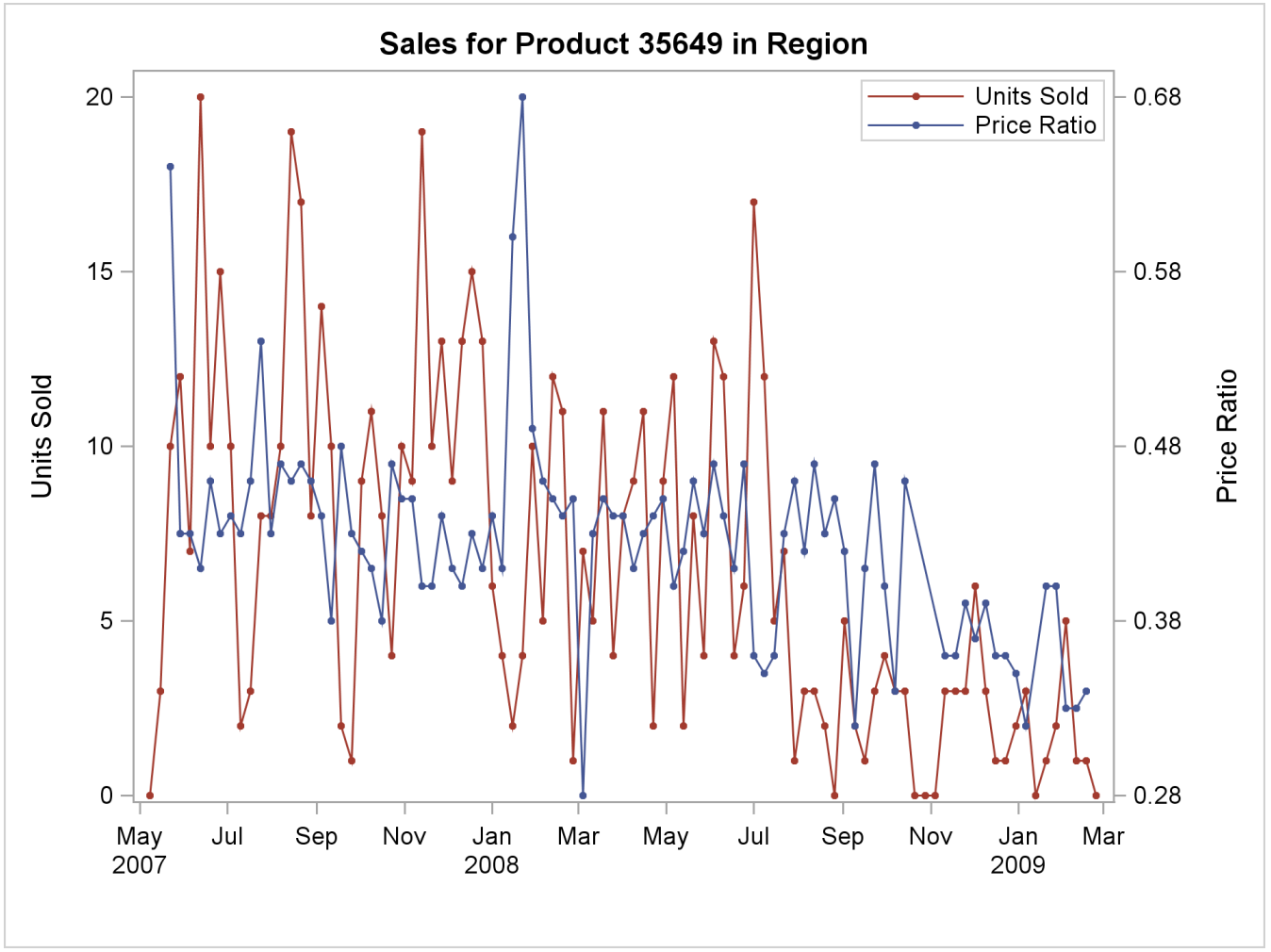
http://isi2011.congressplanner.eu/pdfs/p_450457.pdf

What Can You Do With this Data?

- How do you estimate the components of demand?
- The key lies in analytical hierarchies for
  - Products
  - Store locations

http://isi2011.congressplanner.eu/pdfs/p_450457.pdf

Data Aggregated at Region Level

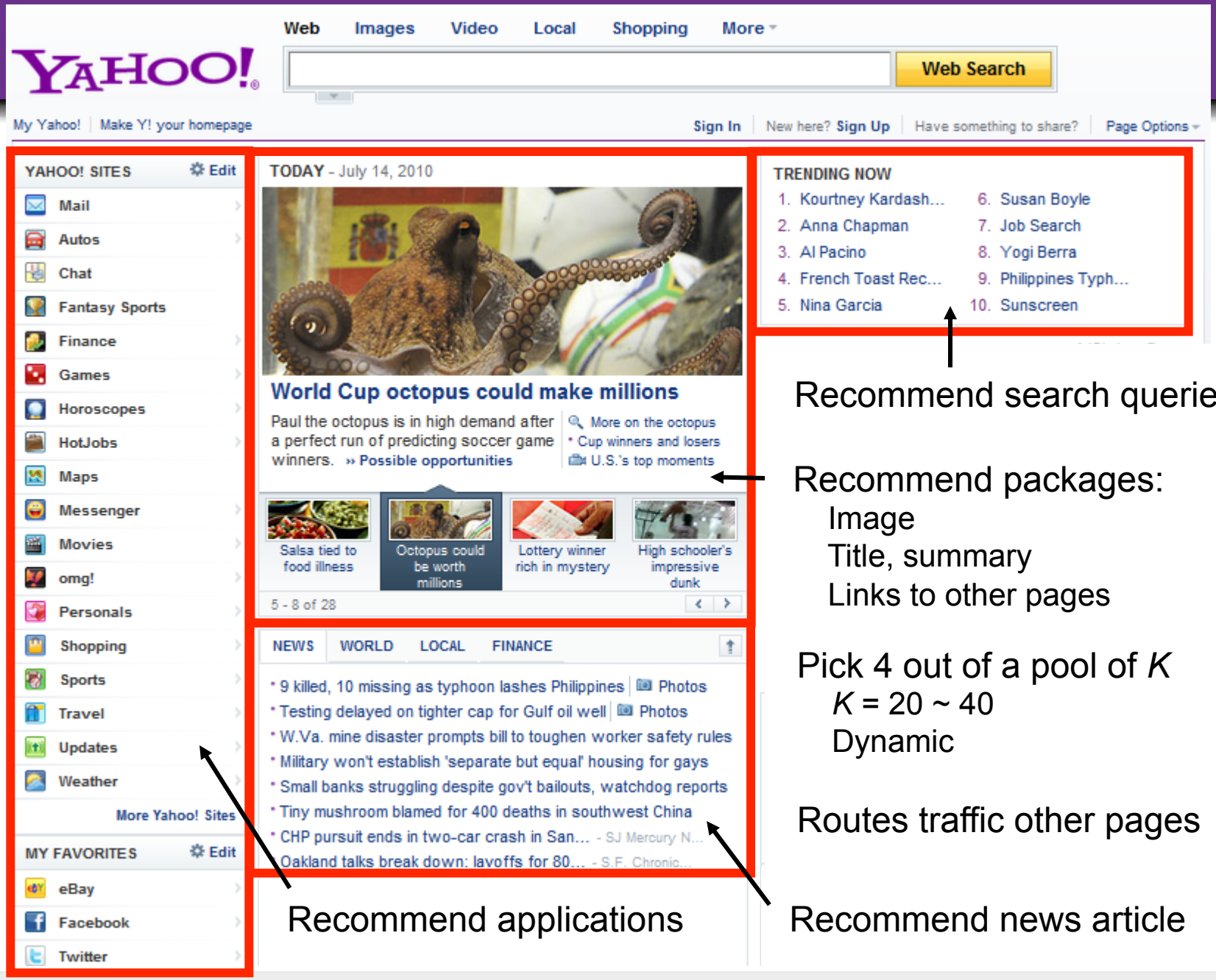


http://isi2011.congressplanner.eu/pdfs/p_450457.pdf

Case Study of Yahoo Recommender Systems

Modern Recommendation Systems

- Goal
  - Serve the right item to a user in a given context to optimize long-term business objectives
- A scientific discipline that involves
  - Large scale Machine Learning & Statistics
    - Offline Models (capture global & stable characteristics)
    - Online Models (incorporates dynamic components)
    - Explore/Exploit (active and adaptive experimentation)
  - Multi-Objective Optimization
    - Click-rates (CTR), Engagement, advertising revenue, diversity, etc
  - Inferring user interest
    - Constructing User Profiles
  - Natural Language Processing to understand content
    - Topics, “aboutness”, entities, follow-up of something, breaking news,…


Web Images Video Local Shopping More
YAHOO!
Web Search
My Yahoo! | Make Y! your homepage
Sign In | New here? Sign Up | Have something to share? | Page Options
YAHOO! SITES Edit
Mail
Autos
Chat
Fantasy Sports
Finance
Games
Horoscopes
HotJobs
Maps
Messenger
Movies
omg!
Personals
Shopping
Sports
Travel
Updates
Weather
More Yahoo! Sites
MY FAVORITES Edit
eBay
Facebook
Twitter
TODAY - July 14, 2010
World Cup octopus could make millions
Paul the octopus is in high demand after a perfect run of predicting soccer game winners. ›› Possible opportunities
More on the octopus
Cup winners and losers
U.S.'s top moments
Salsa tied to food illness
Octopus could be worth millions
Lottery winner rich in mystery
High schooler's impressive dunk
5 - 8 of 28
NEWS WORLD LOCAL FINANCE
9 killed, 10 missing as typhoon lashes Philippines | Photos
Testing delayed on tighter cap for Gulf oil well | Photos
W.Va. mine disaster prompts bill to toughen worker safety rules
Military won't establish 'separate but equal' housing for gays
Small banks struggling despite gov't bailouts, watchdog reports
Tiny mushroom blamed for 400 deaths in southwest China
CHP pursuit ends in two-car crash in San... - SJ Mercury N...
Oakland talks break down: layoffs for 80... - S.F. Chronic...
TRENDING NOW
1. Kourtney Kardash...
2. Anna Chapman
3. Al Pacino
4. French Toast Rec...
5. Nina Garcia
6. Susan Boyle
7. Job Search
8. Yogi Berra
9. Philippines Typh...
10. Sunscreen
Recommend search queries
Recommend packages:
Image
Title, summary
Links to other pages
Pick 4 out of a pool of K
K = 20 ~ 40
Dynamic
Routes traffic other pages
Recommend applications
Recommend news article

Some examples from content optimization

- Simple version
  - I have a content module on my page, content inventory is obtained from a third party source which is further refined through editorial oversight. Can I algorithmically recommend content on this module? I want to improve overall click-rate (CTR) on this module
- More advanced
  - I got X% lift in CTR. But I have additional information on other downstream utilities (e.g. advertising revenue). Can I increase downstream utility without losing too many clicks?
- Highly advanced
  - There are multiple modules running on my webpage. How do I perform a simultaneous optimization?

Problems in this example

- Optimize CTR on multiple modules
  - Today Module, Trending Now, Personal Assistant, News
  - Simple solution: Treat modules as independent, optimize separately. May not be the best when there are strong correlations.

- For any single module
  - Optimize some combination of CTR, downstream engagement, and perhaps advertising revenue.

Online Advertising

conversion

Response rates
(click, conversion, ad-view)

Bids

ML /Statistical model

Auction

Select argmax f(bid,response rates)

Click

Advertisers

Recommend

Ads

Best ad(s)

Page

Ad Network

User

Publisher

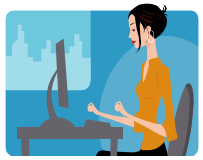
- *Examples:*
  Yahoo, Google, MSN, …
  Ad exchanges (RightMedia, DoubleClick, …)

Recommender problems in general

- Example applications
  - Search: Web, Vertical
  - Online Advertising
  - Content
  - …..

Item Inventory
Articles, web page, ads, …

Context
query, page, …



USER

Use an automated algorithm to select item(s) to show

Get feedback (click, time spent,..)
Refine the models

Repeat (large number of times)
Optimize metric(s) of interest
(Total clicks, Total revenue,…)

Important Factors

- **Items:** Articles, ads, modules, movies, users, updates, etc.
- **Context:** query keywords, pages, mobile, social media, etc.
- **Metric** to optimize (e.g., relevance score, CTR, revenue, engagement)
  - Currently, most applications are single-objective
  - Could be multi-objective optimization (maximize *X* subject to *Y*, *Z*,..)
- **Properties of the item pool**
  - Size (e.g., all web pages vs. 40 stories)
  - Quality of the pool (e.g., anything vs. editorially selected)
  - Lifetime (e.g., mostly old items vs. mostly new items)

Factors affecting Solution (continued)

- **Properties of the context**
  - Pull: Specified by explicit, user-driven query (e.g., keywords, a form)
  - Push: Specified by implicit context (e.g., a page, a user, a session)
    - Most applications are somewhere on continuum of pull and push
- **Properties of the feedback on the matches made**
  - Types and semantics of feedback (e.g., click, vote)
  - Latency (e.g., available in 5 minutes vs. 1 day)
  - Volume (e.g., 100K per day vs. 300M per day)
- **Constraints specifying legitimate matches**
  - e.g., business rules, diversity rules, editorial Voice
  - Multiple objectives
- **Available Metadata** (e.g., link graph, various user/item attributes)

Predicting User-Item Interactions (e.g. CTR)

- Myth: We have so much data on the web, if we can only process it the problem is solved
  - Number of things to learn increases with sample size
    - Rate of increase is not slow
  - Dynamic nature of systems make things worse
  - We want to learn things quickly and react fast

- Data is sparse in web recommender problems
  - We lack enough data to learn all we want to learn and as quickly as we would like to learn
  - Several Power laws interacting with each other
    - E.g. User visits power law, items served power law
      - Bivariate Zipf: Owen & Dyer, 2011

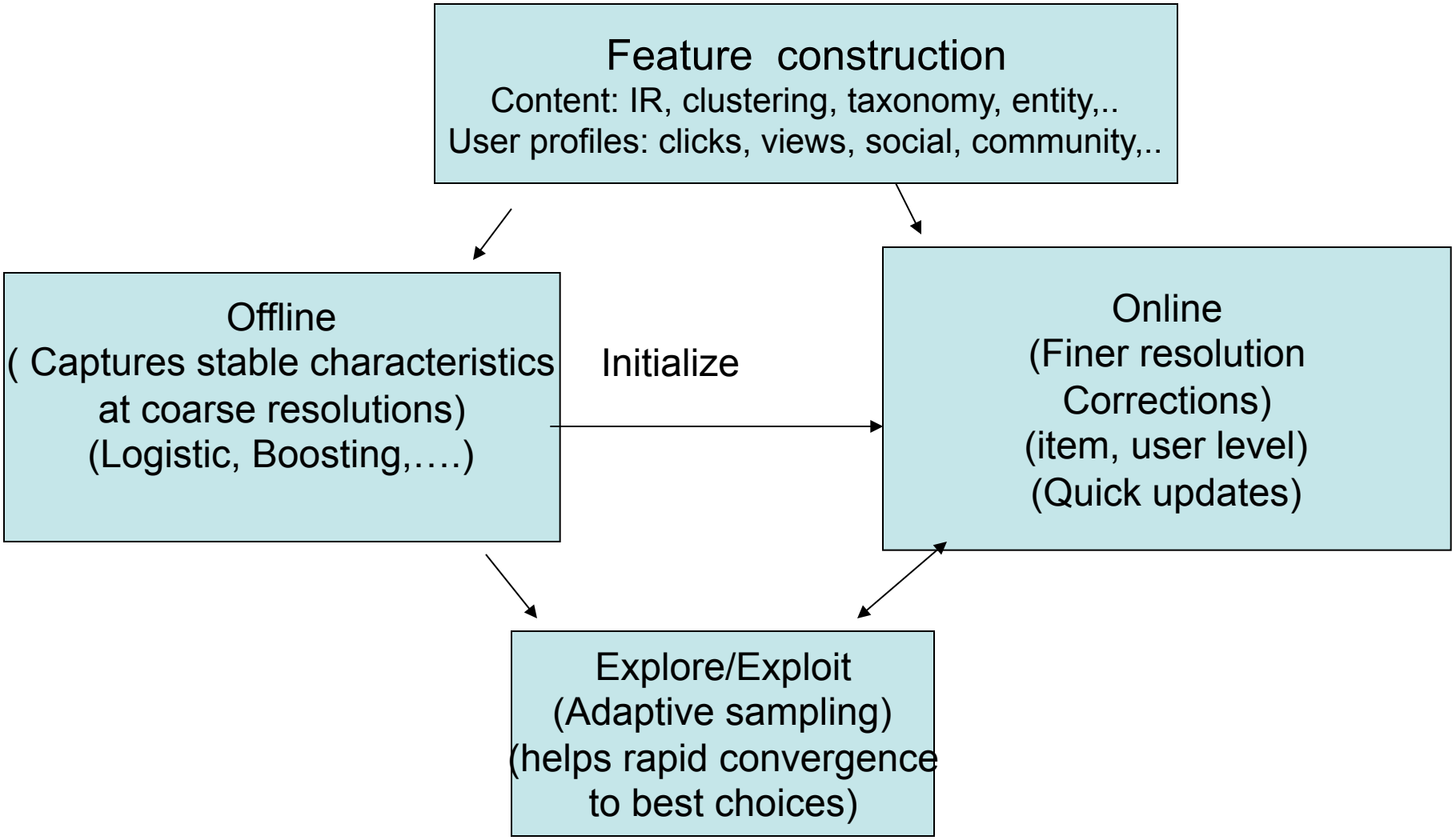
Can Machine Learning help?

- Fortunately, there are group behaviors that generalize to individuals & they are relatively stable
  - E.g. Users in San Francisco tend to read more baseball news

- Key issue: Estimating such groups
  - Coarse group : more stable but does not generalize that well.
  - Granular group: less stable with few individuals
  - Getting a good grouping structure is to hit the "sweet spot"

- Another big advantage on the web A/B Tests in last lecture
  - Intervene and run small experiments on a small population to collect data that helps rapid convergence to the best choices(s)
    - We don't need to learn all user-item interactions, only those that are good.

Predicting user-item interaction rates



Post-click: An example in Content Optimization

Recommender + EDITORIAL

content

Clicks on FP links influence downstream supply distribution

SPORTS

NEWS

OMG

FINANCE

AD SERVER

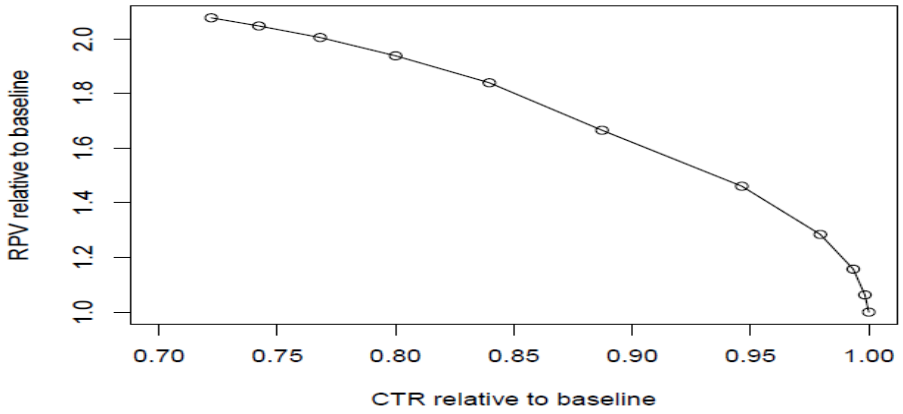
DISPLAY ADVERTISING Revenue

Downstream engagement

(Time spent)

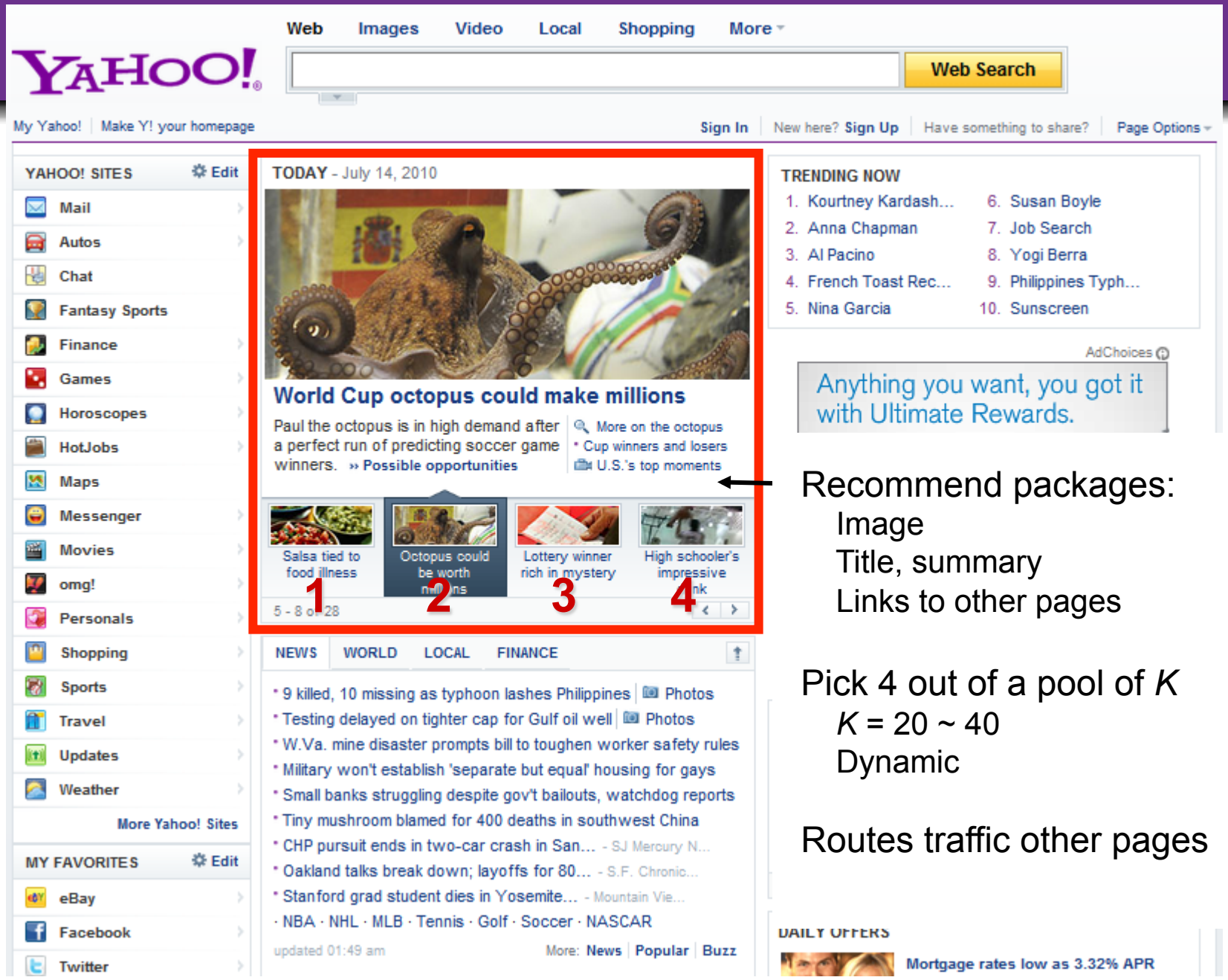
Serving Content on Front Page: Click Shaping

- What do we want to optimize?
- Current: Maximize clicks (maximize downstream supply from FP)
- But consider the following
  - Article 1: CTR=5%, utility per click = 5
  - Article 2: CTR=4.9%, utility per click=10
    - By promoting 2, we lose 1 click/100 visits, gain 5 utils
- If we do this for a large number of visits --- lose some clicks but obtain significant gains in utility?
  - E.g. lose 5% relative CTR, gain 40% in utility (revenue, engagement, etc)



Example Application: Today Module on Yahoo! Homepage

Currently in production, powered by some methods discussed in this tutorial



Recommend packages:

- Image
- Title, summary
- Links to other pages

Pick 4 out of a pool of K

- $K = 20 \sim 40$
- Dynamic

Routes traffic other pages

Problem definition

- Display “best” articles for each user visit
- Best - Maximize User Satisfaction, Engagement
  - BUT Hard to obtain quick feedback to measure these
- Approximation
  - Maximize utility based on immediate feedback (click rate) subject to constraints (relevance, freshness, diversity)
- Inventory of articles?
  - Created by human editors
  - Small pool (30-50 articles) but refreshes periodically

Where are we today?

- Before this research
  - Articles created and selected for display by editors
- After this research
  - Article placement done through statistical models
- How successful ?

"Just look at our homepage, for example. Since we began pairing our content optimization technology with editorial expertise, we've seen click-through rates in the Today module more than double. ----- Carol Bartz, CEO Yahoo! Inc (Q4, 2009)

Main Goals

- Methods to select most popular articles
  - This was done by editors before
- Provide personalized article selection
  - Based on user covariates
  - Based on per user behavior
- Scalability: Methods to generalize in small traffic scenarios
  - Today module part of most Y! portals around the world
  - Also syndicated to sources like Y! Mail, Y! IM etc

Vector Space Formulation of Recommender Systems

Distances in Funny Spaces I

- In **user-based collaborative filtering**, we can think of users in a space of dimension N where there are N items and M users.
  - Let *i* run over items and *u* over users
- Then each **user** is represented as a **vector $U_i(u)$** in "item-space" where ratings are vector components. We are looking for users u u' that are near each other in this space as measured by some distance between $U_i(u)$ and $U_i(u')$
- If u and u' rate all items then these are "real" vectors but almost always they each only rates a small fraction of items and the number in common is even smaller
- The "**Pearson coefficient**" is just one distance measure that can be used
  - Only sum over i rated by u and u'

Distances in Funny Spaces II

- In item-based collaborative filtering, we can think of items in a space of dimension M where there are N items and M users.
  - Let *i* run over items and *u* over users
- Then each **item** is represented as a **vector $R_u(i)$** in "user-space" where ratings are vector components. We are looking for items i i' that are near each other in this space as measured by some distance between $R_u(i)$ and $R_u(i')$
- If i and i' rated by all users then these are "real" vectors but almost always they are each only rated by a small fraction of users and the number in common is even smaller
- The "**Cosine measure**" is just one distance measure that can be used
  - Only sum over users u rating both i and i'

Do we need "real" spaces?

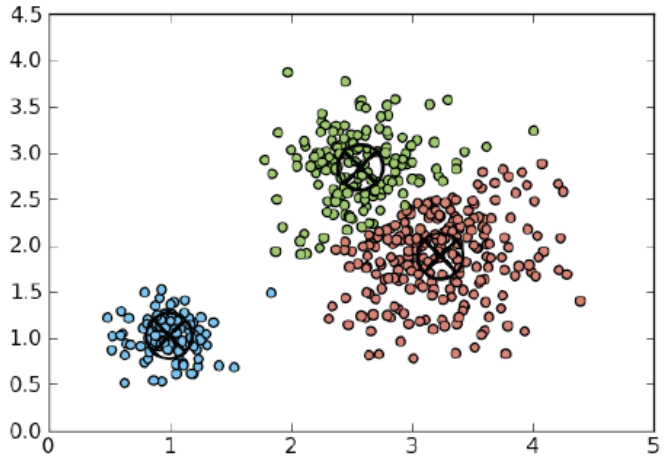
- Much of (eCommerce/LifeStyle) Informatics involves "points"
  - Events in LHC analysis
  - Users (people) or items (books, jobs, music, other people)
- These points can be thought of being in a "space"
  - Set of all books
  - Set of all physics reactions
  - Set of all Internet users
- However as in recommender systems where a given user only rates some items, we don't know "full position"
- However we can nearly always define a useful **distance d(a,b)** between points
- Always **d(a,b) >= 0**
- Usually **d(a,b) = d(b,a)**
- Rarely d(a,b) + d(b,c) >= d(a,c) **Triangle Inequality**

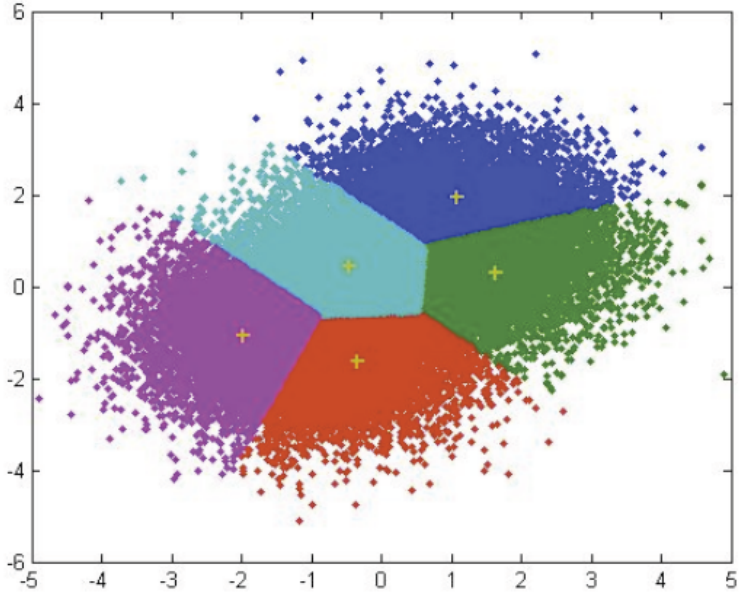
Using Distances

- The simplest way to use distances is "**nearest neighbor algorithms**" – given one point, find a set of points near it – cut off by number of identified nearby points and/or distance to initial point
  - Here point is either user or item
- Another approach is divide space into regions (topics, latent factors) consisting of nearby points
  - This is **clustering** described in following slide
- Also other algorithms like **Gaussian mixture models** or **Latent Semantic Analysis** or **Latent Dirichlet Allocation** which use a more sophisticated model

Clustering

- One can think of users and/or items as points in a space and try to find clusters
- Either clear clusters (as in left) or divisions of space into points close to each other (right picture)
- All entities in the same cluster are assigned the same recommendations which can dramatically speed up real time computation
  - i.e. all users in same cluster are assigned same recommendations for a given item
  - See 2003 Amazon article (http://www.cs.umd.edu/~samir/498/Amazon-Recommendations.pdf) for





Clustering discussed later

User-based nearest-neighbor collaborative filtering

Collaborative Filtering (CF)

- **The most prominent approach to generate recommendations**
  - 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 tastes in the future

Pure CF Approaches

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

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)

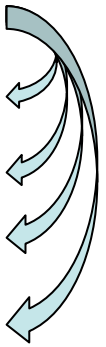
- **A popular similarity measure in user-based CF: Pearson correlation**
  - a, b : users
  - $r_{a,p}$: rating of user a for item p
  - P : set of items, rated both by a and b
  - Possible similarity values between -1 and 1

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

Measuring user similarity (2)

- **A popular similarity measure in user-based CF: Pearson correlation**
  - a, b : users
  - $r{\downarrow}a,p$: 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

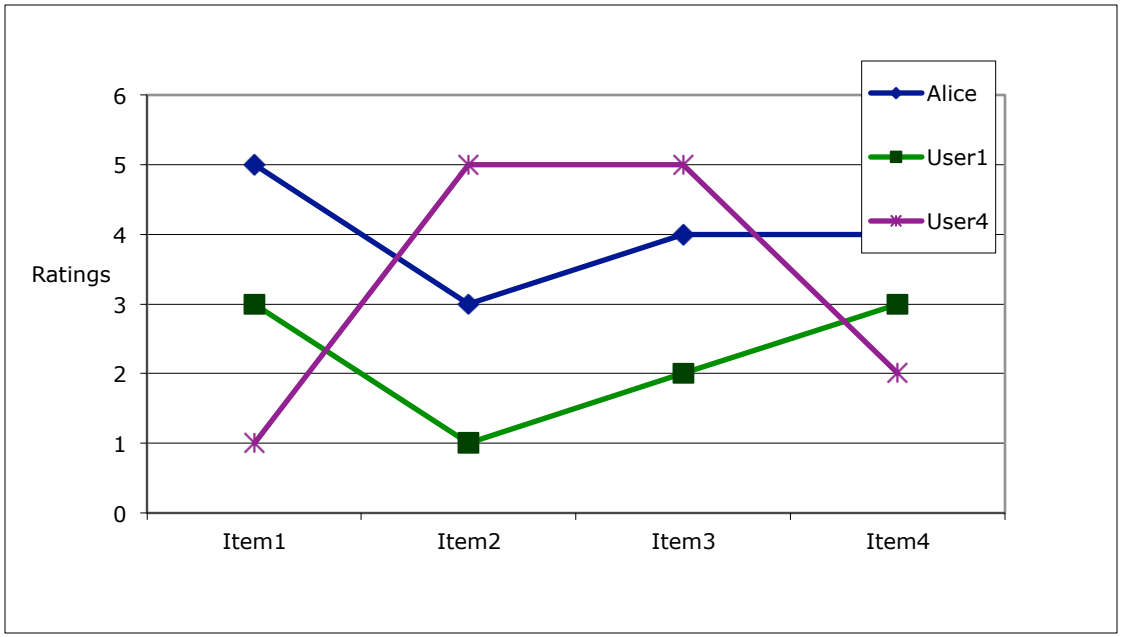
sim = 0,00

sim = 0,70

sim = -0,79

Pearson correlation

- **Takes differences in rating behavior into account**

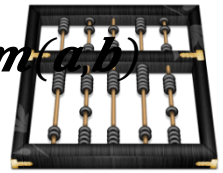


- **Works well in usual domains, compared with alternative measures**
  - such as cosine similarity

Making predictions

- A common prediction function:

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



- 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

Improving the metrics / prediction function

- **Not all neighbor ratings might be equally "valuable"**
  - Agreement on commonly liked items is not so informative as agreement on controversial items
  - **Possible solution**: Give more weight to items that have a higher variance
- **Value of number of co-rated items**
  - Use "significance weighting", by e.g., linearly reducing the weight when the number of co-rated items is low
- **Case amplification**
  - Intuition: Give more weight to "very similar" neighbors, i.e., where the similarity value is close to 1.
- **Neighborhood selection**
  - Use similarity threshold or fixed number of neighbors