Recommendations in Real-time

Ted Dunning

Who I am

Ted Dunning, Chief Applications Architect, MapR Technologies
Email tdunning@mapr.com tdunning@apache.org
Twitter @Ted_Dunning

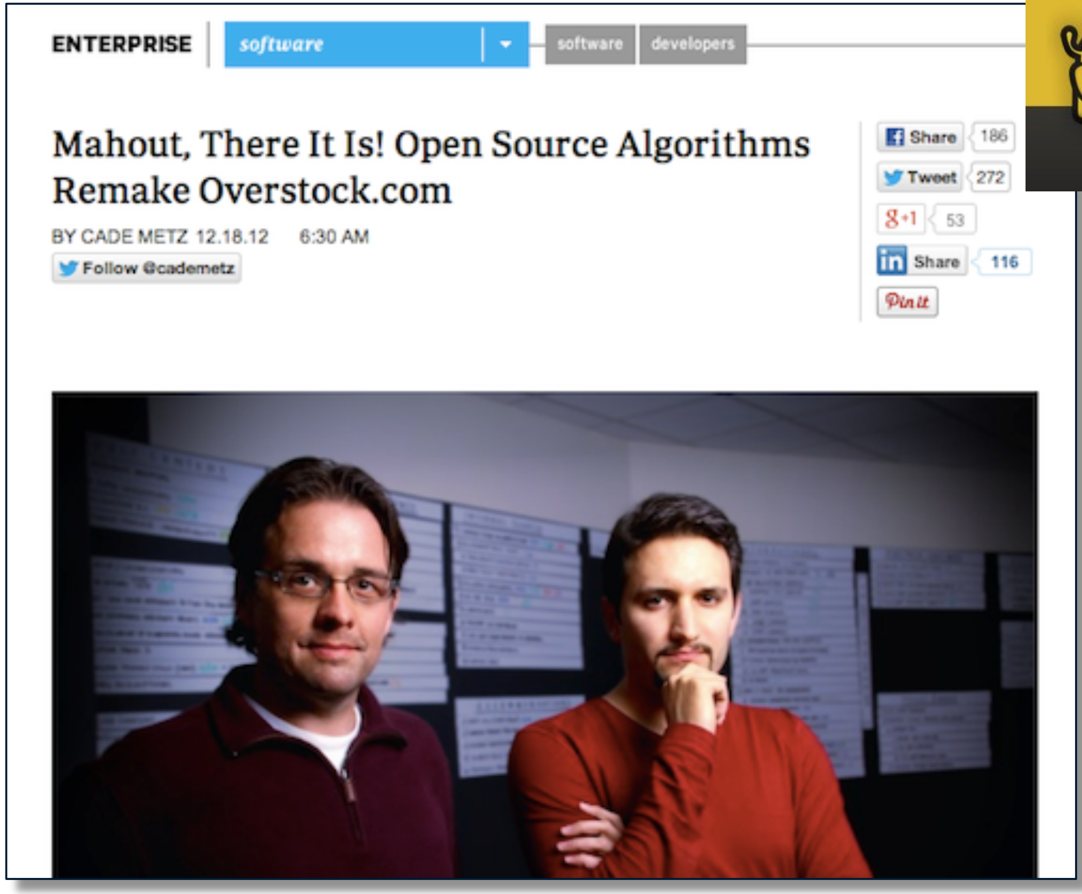
Apache Mahout https://mahout.apache.org/
Twitter @ApacheMahout

Big Data Everywhere Conference

Also involved with Apache Storm, Zookeeper, Kylin, Drill, Samoa and others

Recommendation: Widely Used Machine Learning

Example: Open source Apache Mahout used in production


ENTERPRISE software software developers

Mahout, There It Is! Open Source Algorithms Remake Overstock.com

BY CADE METZ 12.18.12 6:30 AM

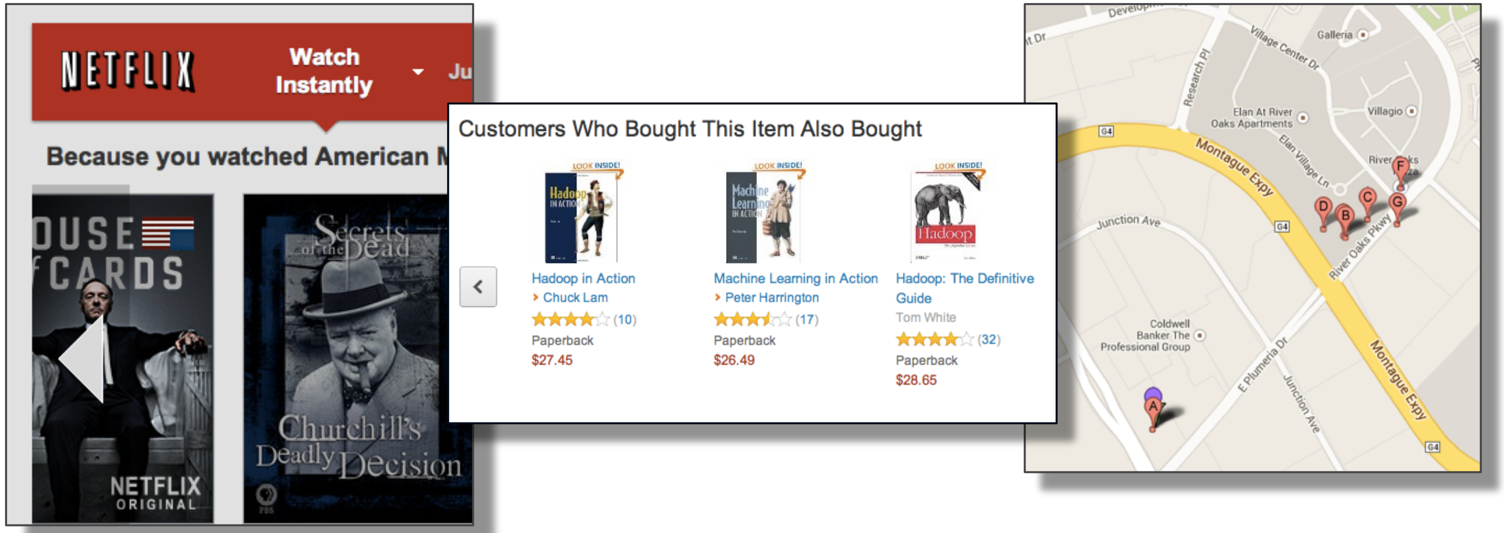
Follow @cademetz

Share 186
Tweet 272
+1 53
Share 116
Pin it

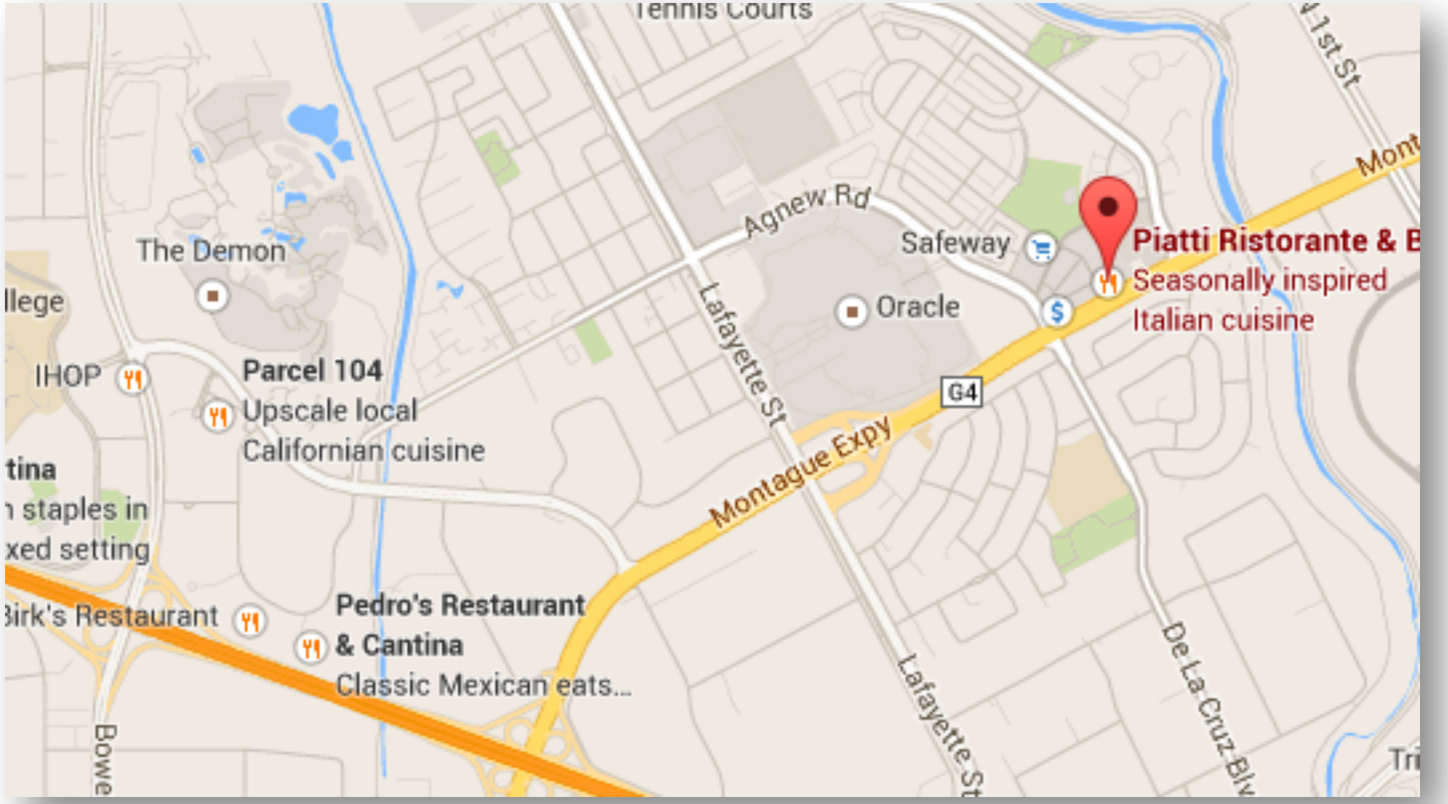
http://www.wired.com/wiredenterprise/2012/12/mahout/

Recommendations

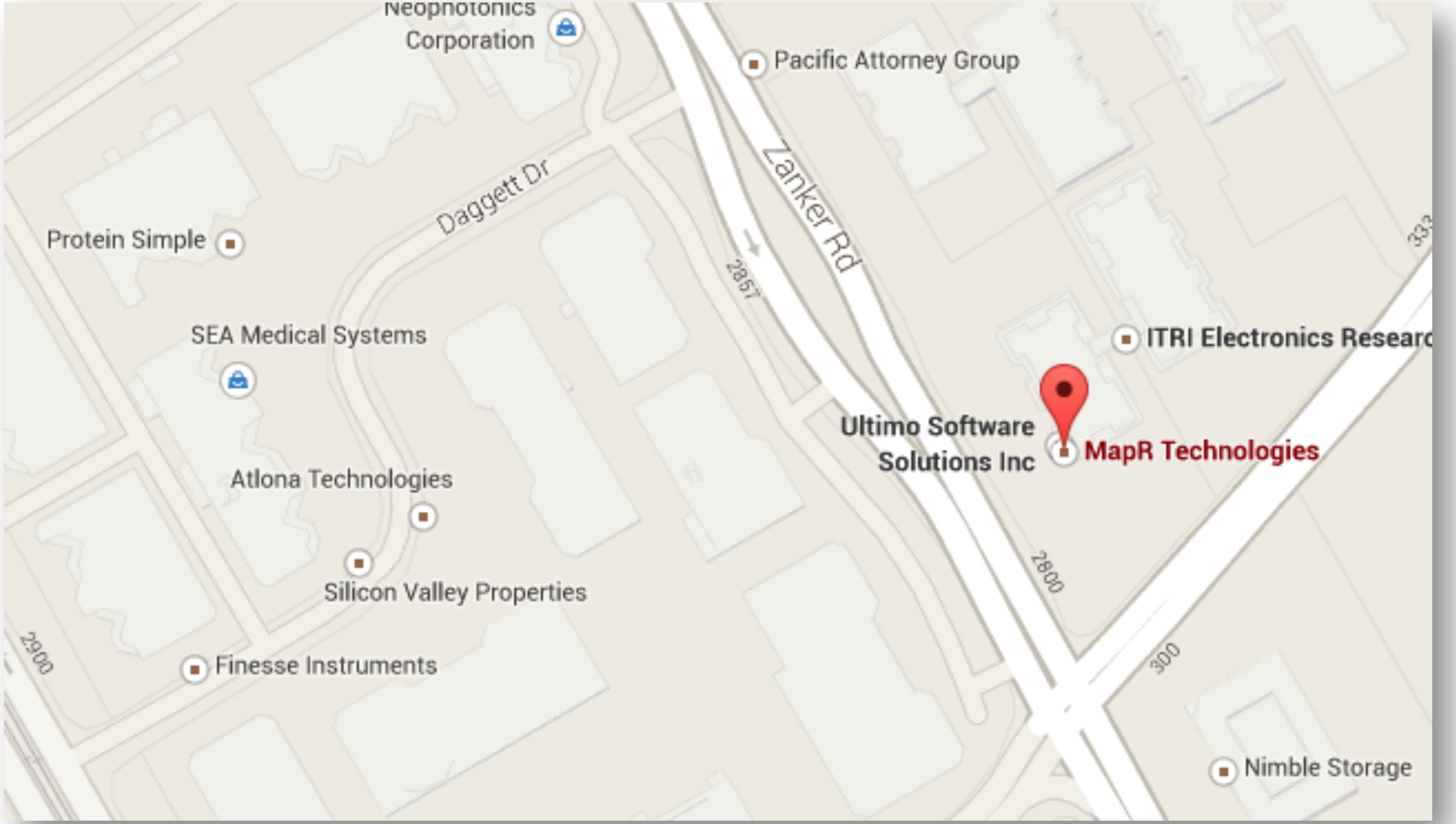
- Data: interactions between people taking action (users) and items
  - Data used to train recommendation model
- Goal is to suggest additional interactions
- Example applications: movie, music or map-based restaurant choices; suggesting sale items for e-stores or via cash-register receipts



Google maps: restaurant recommendations



Google maps: tech recommendations



Who I am

Ted Dunning, Chief Applications Architect, MapR Technologies

Email tdunning@mapr.com tdunning@apache.org

Twitter @Ted_Dunning

Apache Mahout https://mahout.apache.org/

Twitter @ApacheMahout

The outline

- Quick and fast recommendations

- Non-traditional recommendations

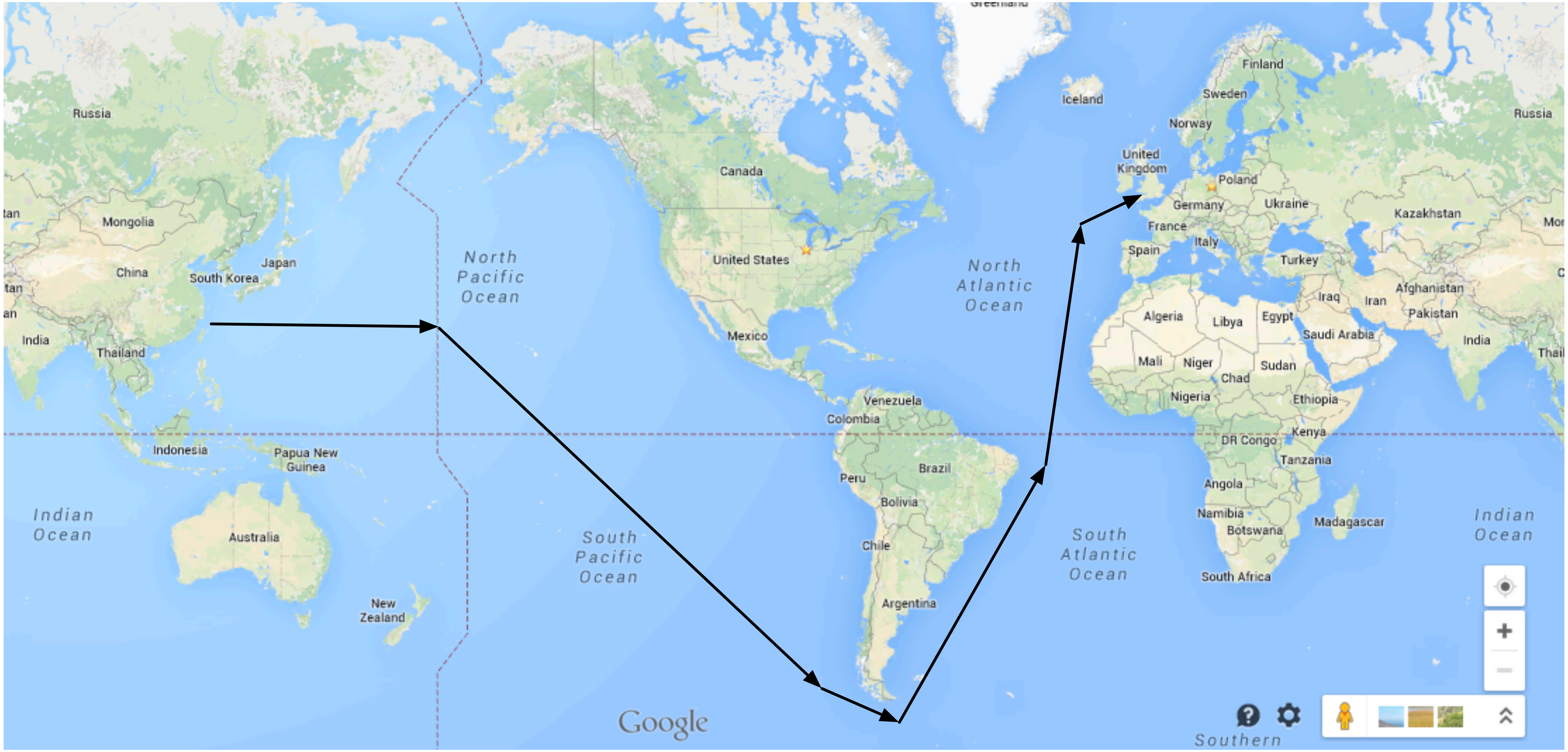
- Conclusions

But first:

An (small) apology for going off-script

Now, a story

In 1866, the top finishers in the tea race reached London in 99 days, within 2 hours of each other


Russia
Mongolia
China
South Korea
Japan
India
Thailand
Indonesia
Papua New Guinea
Australia
New Zealand
Indian Ocean
North Pacific Ocean
South Pacific Ocean
Canada
United States
Mexico
Venezuela
Colombia
Peru
Brazil
Bolivia
Chile
Argentina
North Atlantic Ocean
South Atlantic Ocean
Iceland
Finland
Sweden
Norway
United Kingdom
Poland
Germany
Ukraine
France
Italy
Spain
Turkey
Kazakhstan
Afghanistan
Pakistan
Iraq
Iran
Algeria
Libya
Egypt
Saudi Arabia
India
Mali
Niger
Chad
Sudan
Nigeria
Ethiopia
DR Congo
Kenya
Tanzania
Angola
Namibia
Botswana
Madagascar
South Africa
Indian Ocean
Southern
Google

Before 1850,
New York – San Francisco
took 150 days on average

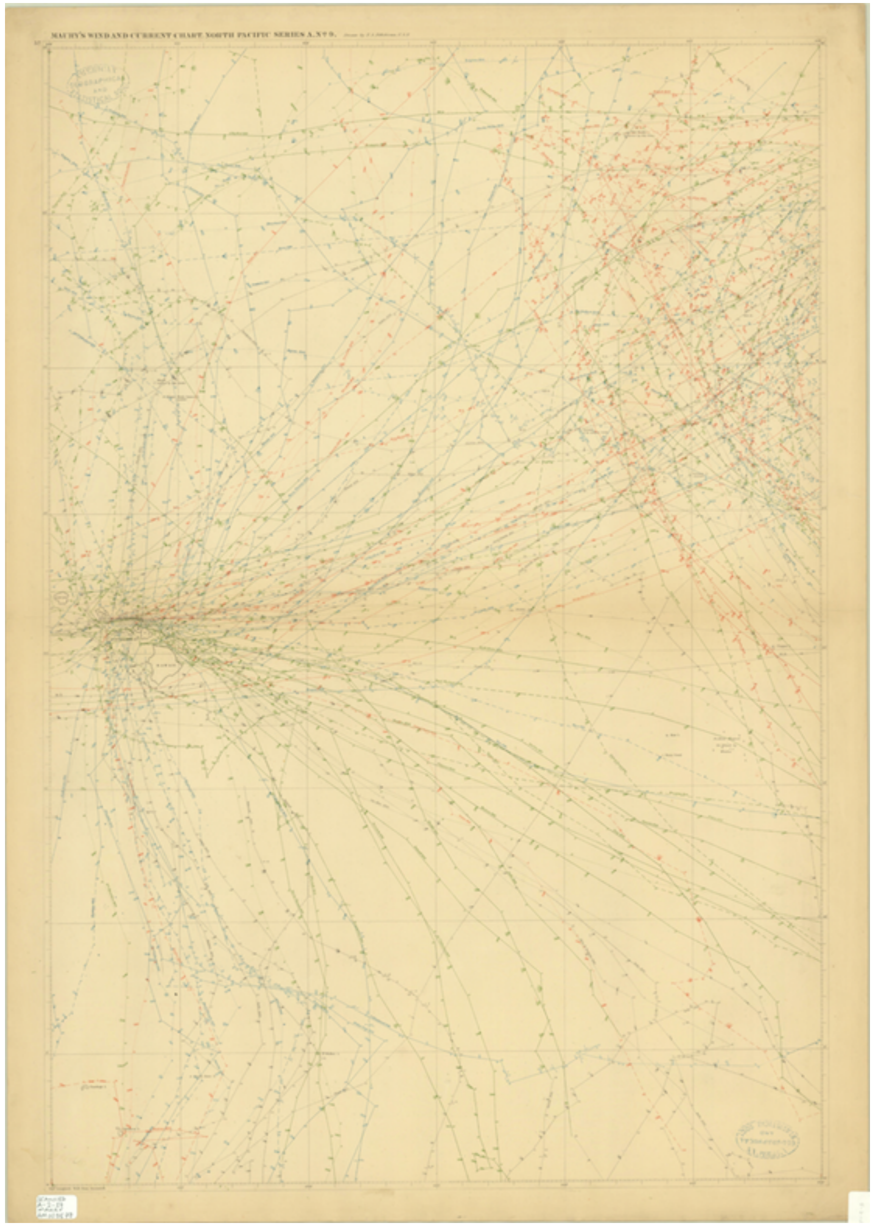
But in 1851, the record was set at 89 days by the Flying Cloud

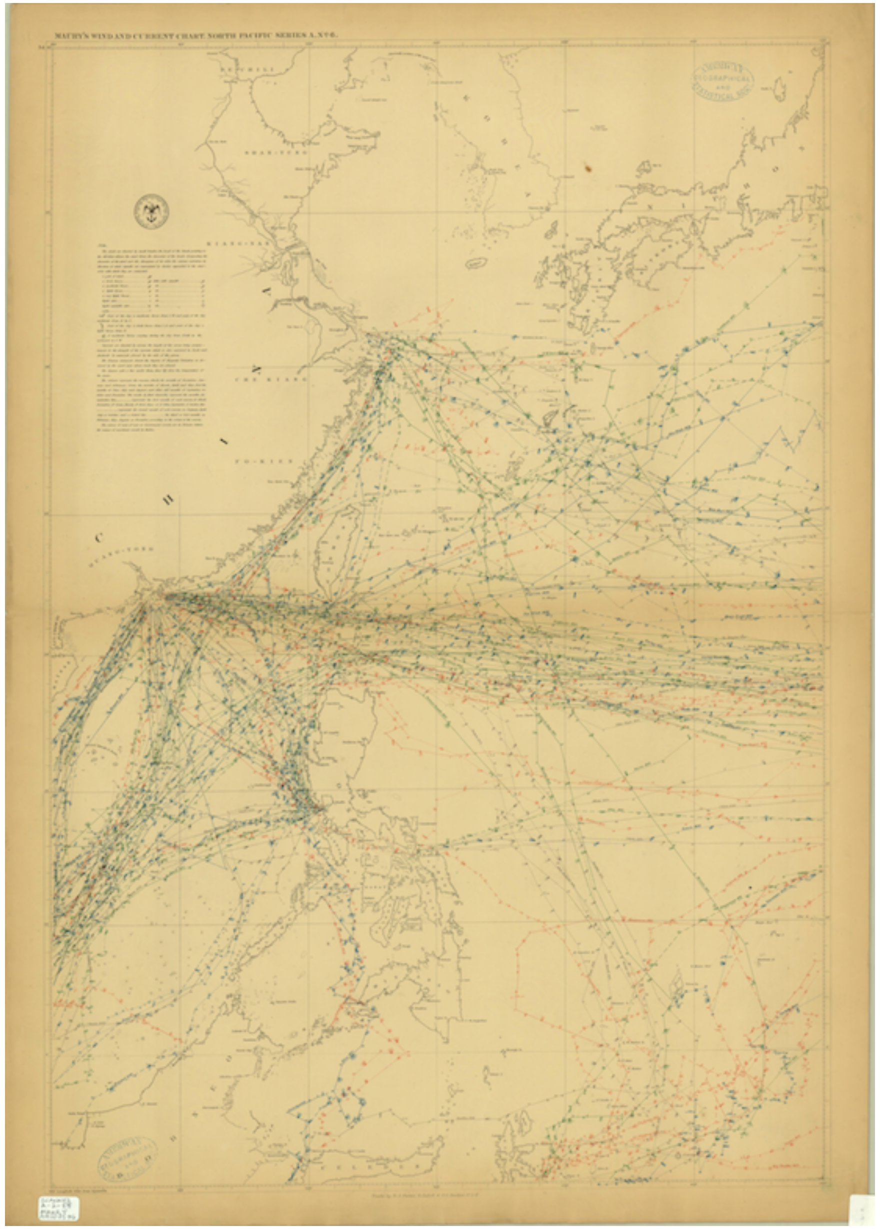
The difference was due
(in part)
to big data

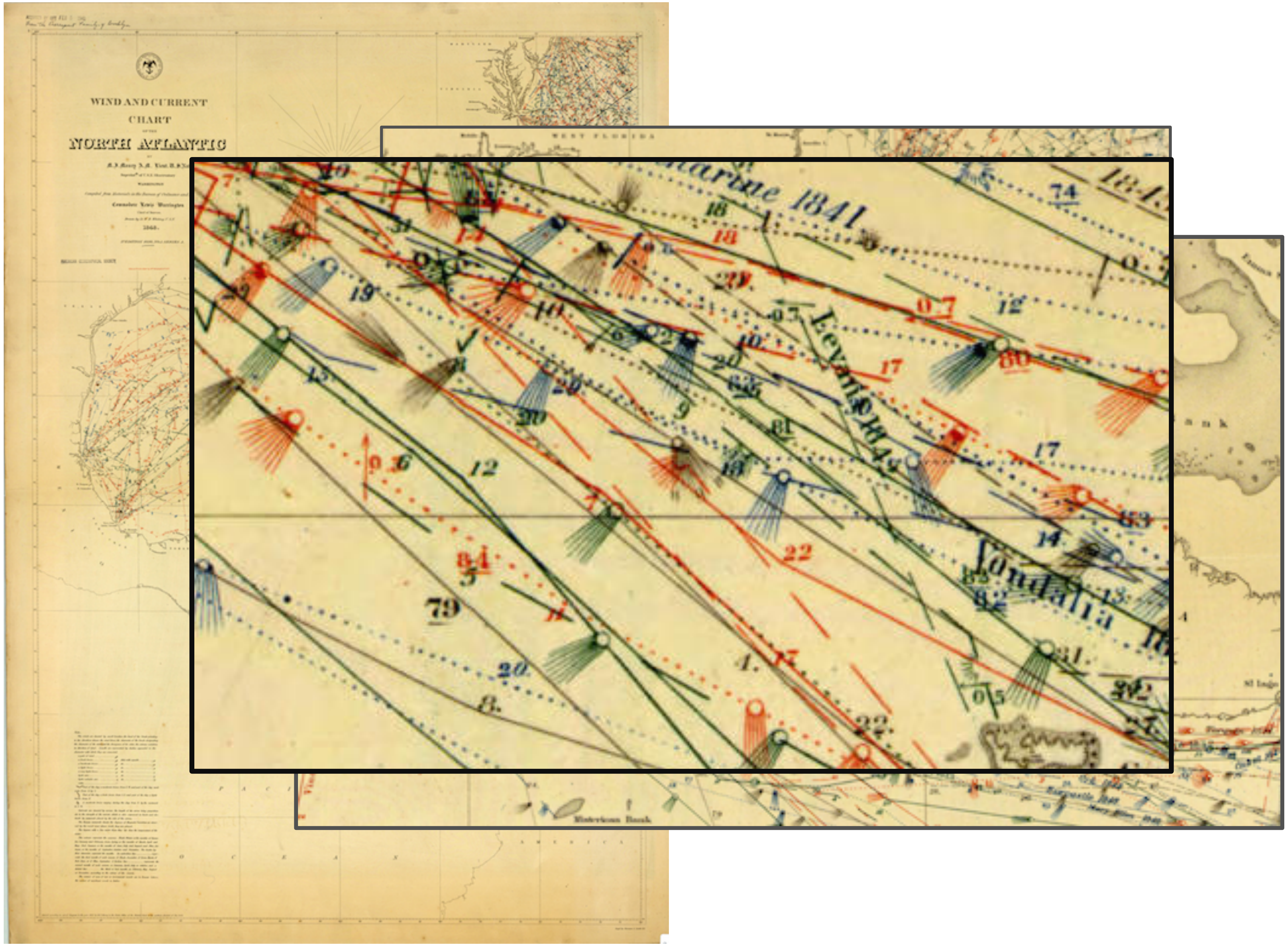
LOG of the UNITED STATES Steamer Bear, Rate, Guns,

Making passage from New York to St Johns N.F.

Hour.	Knots.	Fathoms.	Courses steered.	WINDS. Direction.	WINDS. Force.	Leeway	BAROMETER. Height in inches.	BAROMETER. Ther. att'd.	TEMPERATURE. Air Dry Bulb.	TEMPERATURE. Air Wet Bulb.	TEMPERATURE. Water at surface.	State of the Weather, by symbols.	Forms of Clouds, by symbols.	Prop. of Clear Sky, in 10ths.	State of the Sea.	Record of the sail the vessel is under at end of watch.
A. M.																
1	6	5	E x N N.E.	N. E.	2		29.60	53	44	44	45	O.C.m	Nimb	0	S	Steam alone
2	7	5	" "	N. W	2		" "	54	45	45	44	" "	"	0	"	
3	7	5	" "	West	1		" "	"	"	"	"	" "	"	0	"	
4	7	5	" "	"	1		" "	"	"	"	"	O.C.f	"	0	"	
5	7	5	" "	S. W.	1		" "	"	45	46	41	" "	"	0	"	
6	7	2	" "	S.W.	1		29.62	52	45	46	41	" "	"	0	"	
7	8	5	" "	N. W.	1		29.61	51	45	45	43	" "	"	0	"	
8	8	5	" "	"	1		" "	51	49	46	45	" "	"	0	"	7.25 Sail & Steam
9	7	4	E.N.E ½ E	"	2		29.62	53	47	46	45	O.C.m	"	0	"	
10	7	0	" "	N. E.	1		29.63	53	47	47	45	O.C	"	0	"	
11	7	4	" "	Calm	0		29.64	55	47	47	43	O.C. m	"	0	"	
Noon.	25	04	E.N.E ½ E East	"	0		" "	55	47	47	43	" "	"	0	"	


MAURY'S WIND AND CURRENT CHART NORTH PACIFIC SERIES A. N° 9.




WIND AND CURRENT
CHART
NORTH ATLANTIC
WEST FLORIDA
Vandalia

These charts were free …

If you donated your data

But how does this apply today?

Key Points of Maury's Work

- Give to get
  - Give the Abstract Log to captains, get data
- Data consortium wins
  - Merging data gives pictures nobody else can see
- Give back
  - Them that gives, also gets

- But this is just what every data driven web site does!
  - Just 150 years before everybody else

Now… back on topic

What You Will Hear About Recommendations

- Everybody knows that:
- You need ensembles of many models to do recommendations
- You need to use factorization models
- You predict what you observe
- (You should predict ratings)

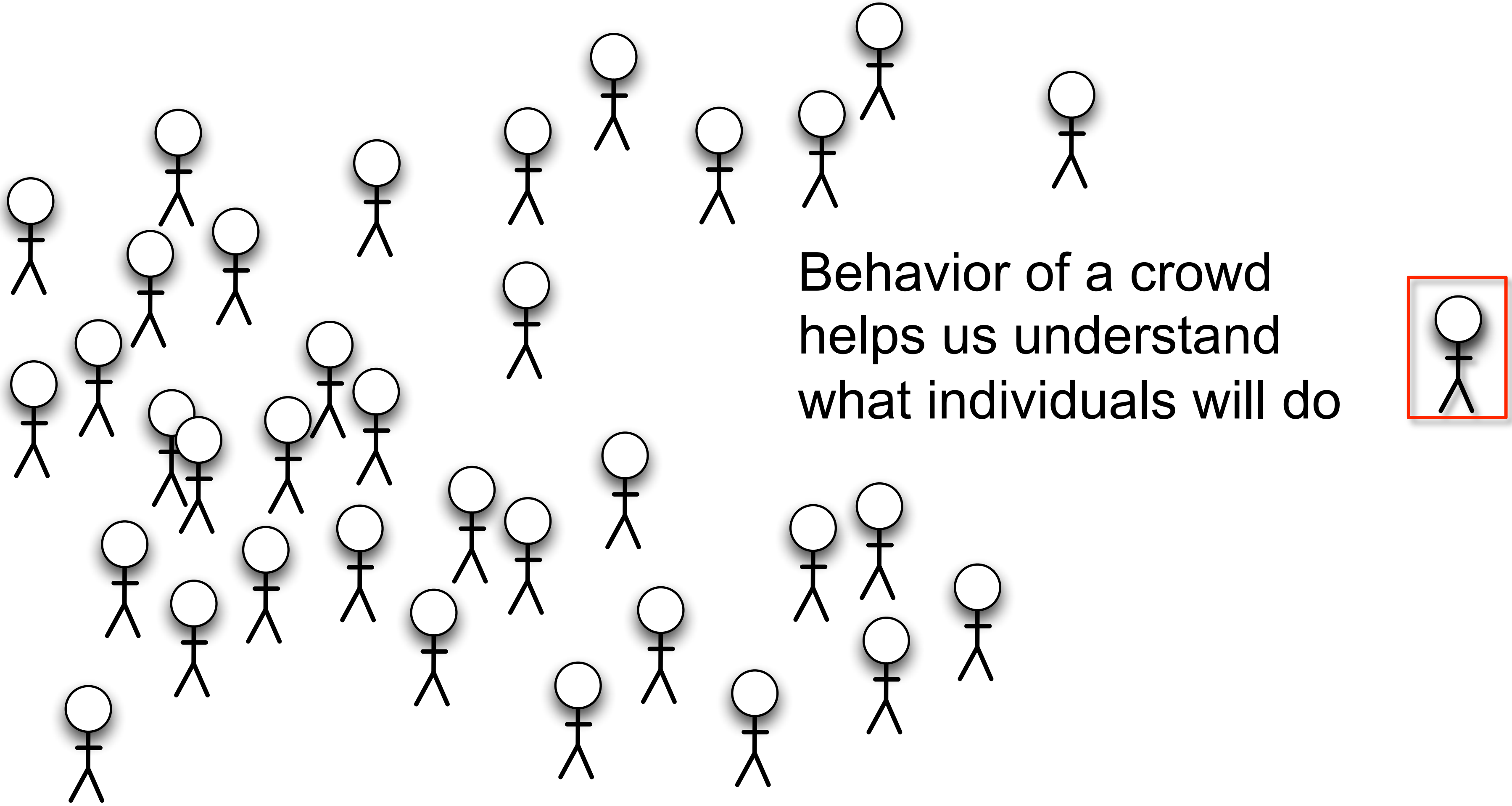
But …

none of this is really true

In fact,

- Fancy models are rarely useful expenditures of time
- Factorization can be good, but not much better (if at all)
- Ratings are disastrously bad data
- Cross-recommendation and multi-modal recommendations are much more interesting
  - Multiple kinds of input are far better than multiple models
- The UI has a far larger impact than the models
- The best algorithms combine simplicity with accuracy
  - So simple you can embed them in a search engine

Recommendation



Here’s how

Recommendations

Alice

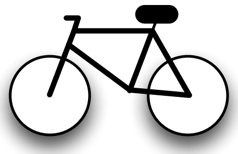
Alice got an apple and a puppy

Recommendations

Alice

Alice got an apple and a puppy

Charles



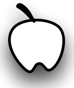
Charles got a bicycle

Recommendations

Alice

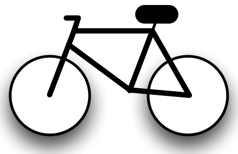
Alice got an apple and a puppy

Bob



Bob got an apple

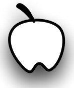
Charles

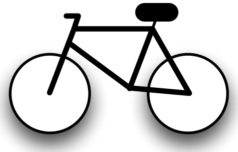


Charles got a bicycle

Recommendations

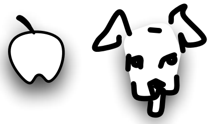
Alice  Alice got an apple and a puppy

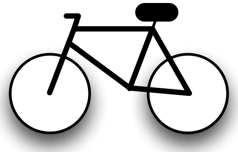
Bob  What else would Bob like?

Charles  Charles got a bicycle

Recommendations

Alice  Alice got an apple and a puppy

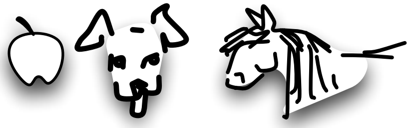
Bob  A puppy!

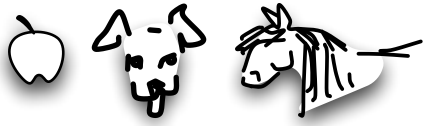
Charles  Charles got a bicycle

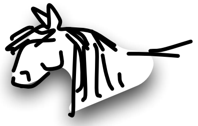
You get the idea of how recommenders can work…

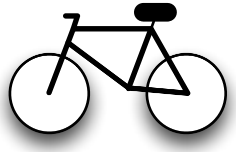
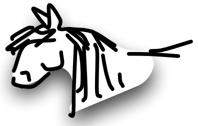
By the way, like me, Bob also wants a pony…

Recommendations

Alice 

Bob 

Amelia  ?

Charles  

What if everybody gets a pony?

What else would you recommend for new user Amelia?

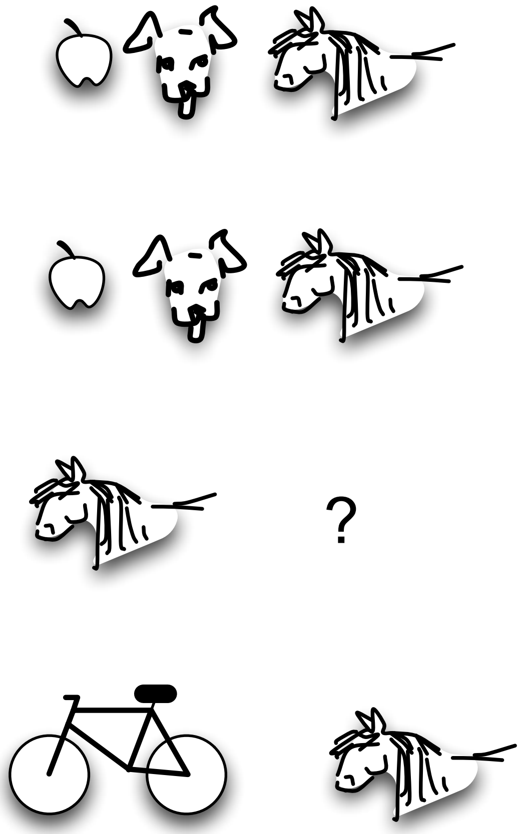
Recommendations

Alice

Bob

Amelia

Charles



If everybody gets a pony, it’s not a very good indicator of what to else predict...

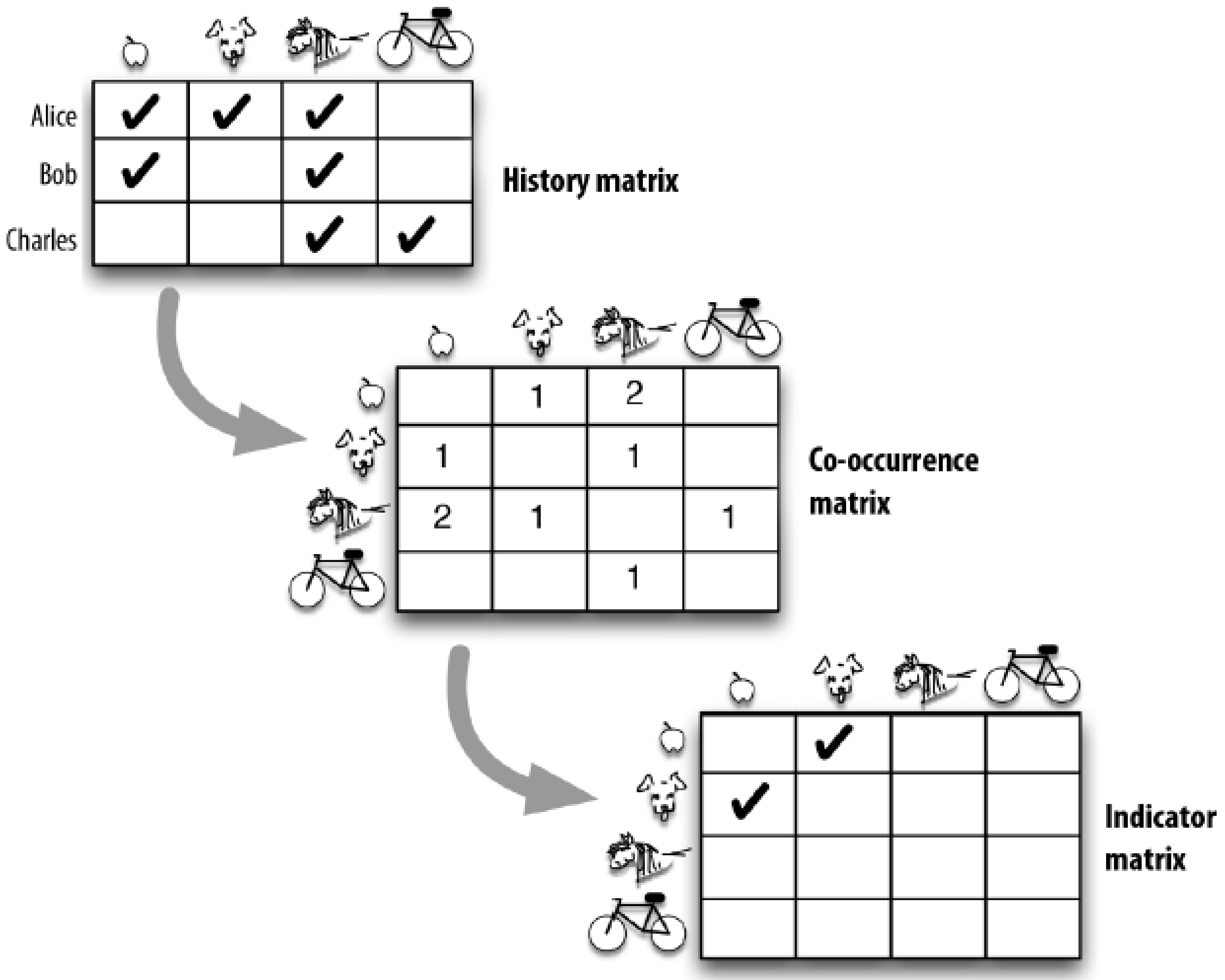
Problems with Raw Co-occurrence

- **Very popular items co-occur with everything** or why it's not very helpful to know that everybody wants a pony…
  - Examples: Welcome document; Elevator music
- **Very widespread occurrence is not interesting to generate indicators for recommendation**
  - Unless you want to offer an item that is constantly desired, such as razor blades (or ponies)
- What we want is ***anomalous*** co-occurrence
  - This is the source of interesting indicators of preference on which to base recommendation

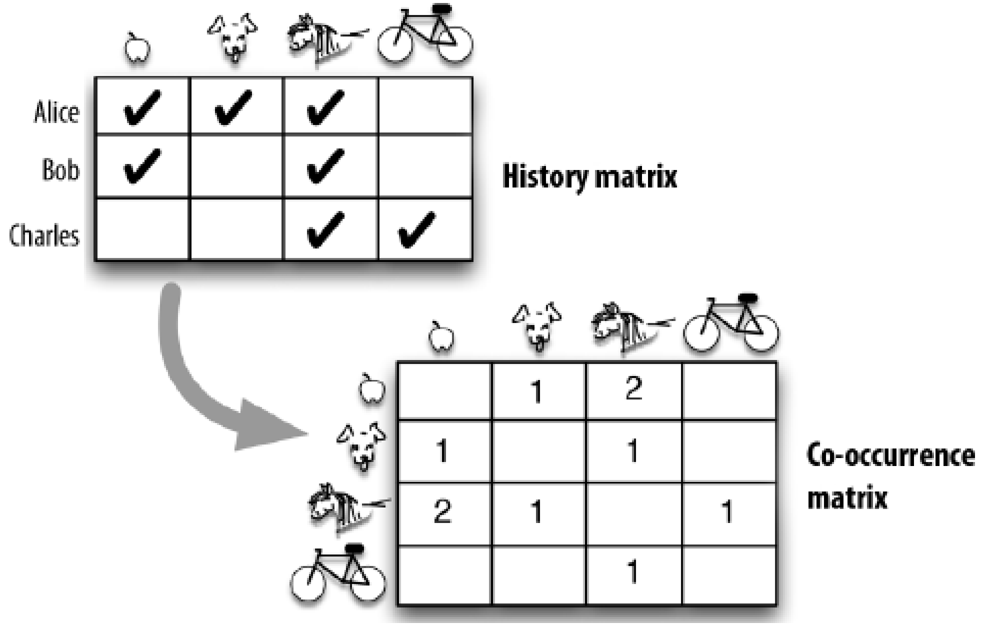
Overview: Get Useful Indicators from Behaviors

1. Use log files to build **history matrix** of users x items
   – Remember: this history of interactions will be sparse compared to all potential combinations
2. Transform to a **co-occurrence matrix of** items x items
3. Look for useful indicators by identifying anomalous co-occurrences to make an **indicator matrix**
   – **Log Likelihood Ratio (LLR**) can be helpful to judge which co-occurrences can with confidence be used as indicators of preference
   – `ItemSimilarityJob` in Apache Mahout uses LLR

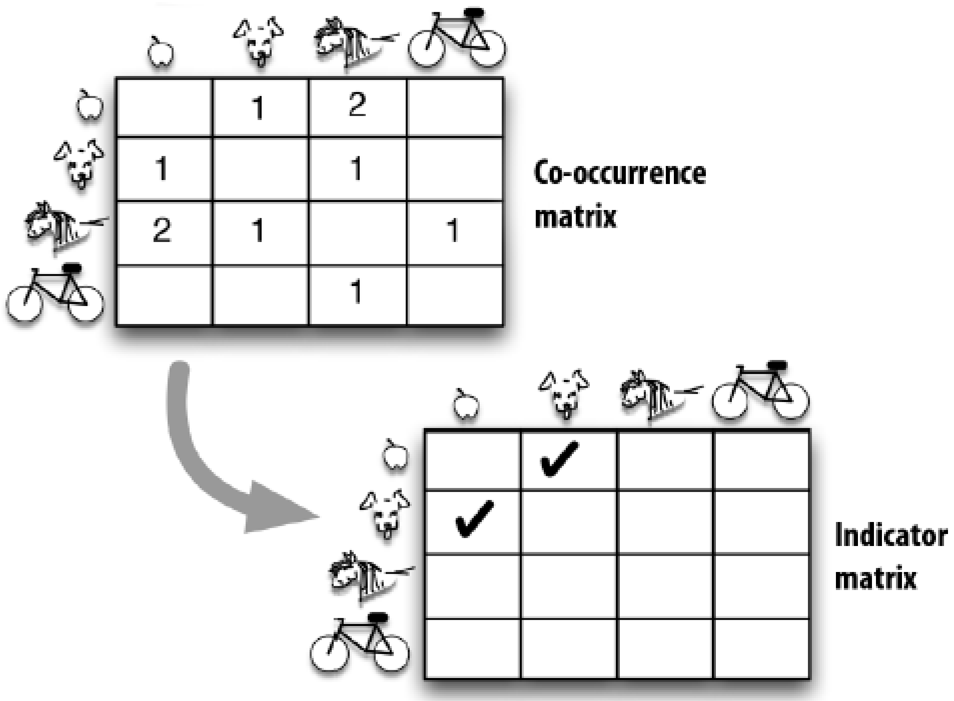
Cooccurrence Analysis



How Often Do Items Co-occur



Which Co-occurrences are Interesting?



Each row of indicators becomes a field in a search engine document

Which one is the anomalous co-occurrence?

	A	*not* A
B	13	1000
not B	1000	100,000

	A	*not* A
B	1	0
not B	0	2

	A	*not* A
B	1	0
not B	0	10,000

	A	*not* A
B	10	0
not B	0	100,000

Which one is the anomalous co-occurrence?

0.90

	A	*not* A
	13	1000
not B	1000	100,000

1.95

	A	*not* A
	1	0
not B	0	2

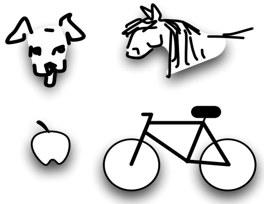
4.52

	A	*not* A
	1	0
not B	0	10,000

14.3

	A	*not* A
	10	0
not B	0	100,000

Collection of Documents: Insert Meta-Data



Item meta-data

Ingest easily via NFS

Search Technology

Document for "puppy"

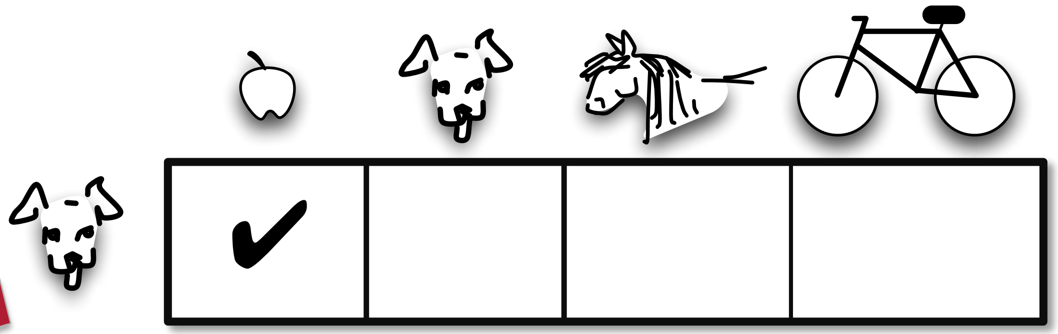
id: t4
title: puppy
desc: The sweetest little puppy ever.
keywords: puppy, dog, pet

Start With Search Index

Solr document
for "puppy"

id: t4
title: puppy
desc: The sweetest little puppy ever.
keywords: puppy, dog, pet

From Indicator Matrix to New Indicator Field



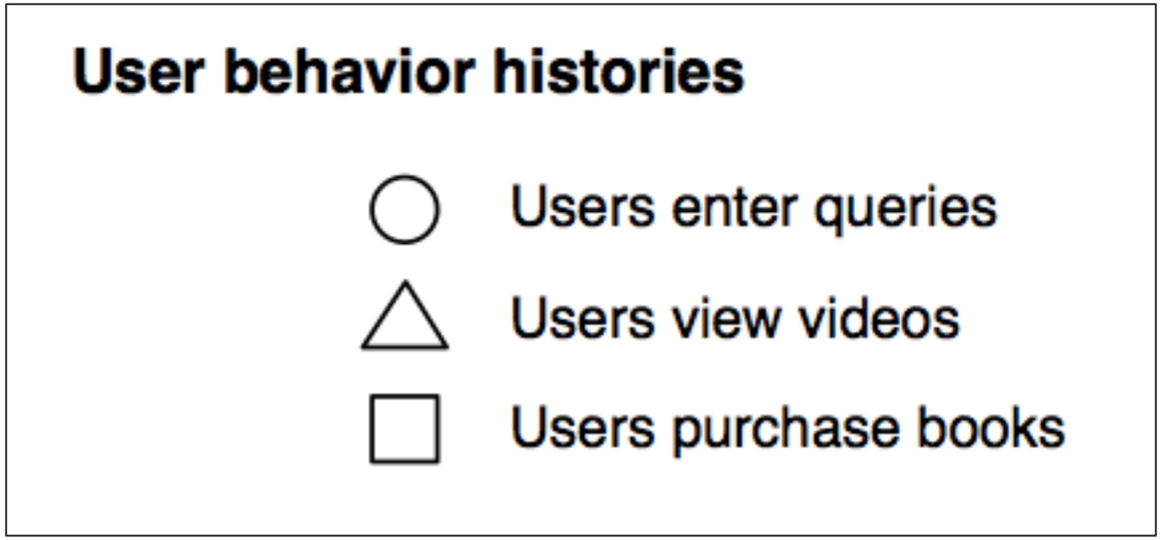
Solr document
for "puppy"

id: t4
title: puppy
desc: The sweetest little puppy ever.
keywords: puppy, dog, pet

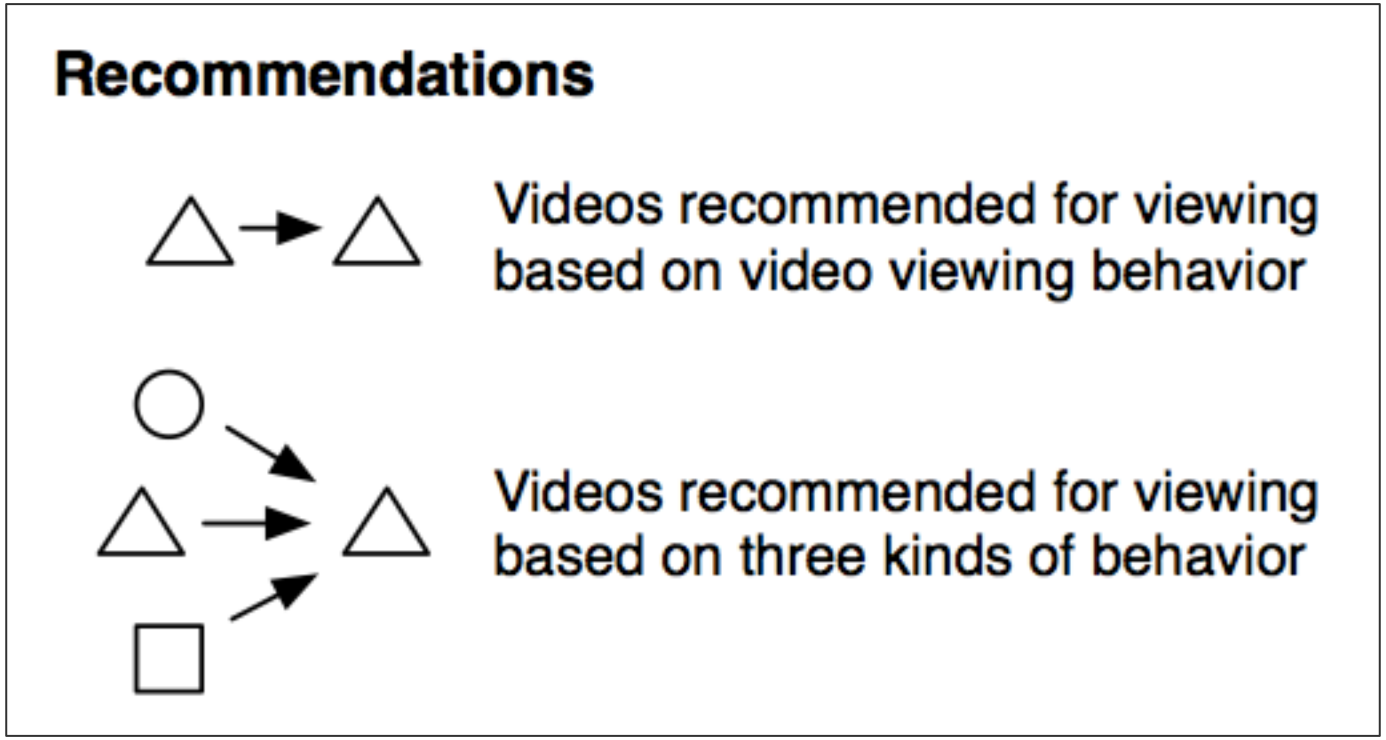
indicators: (t1)

Note: data for the indicator field is added directly to meta-data for a document in Apache Solr or Elastic Search index. You don't need to create a separate index for the indicators.

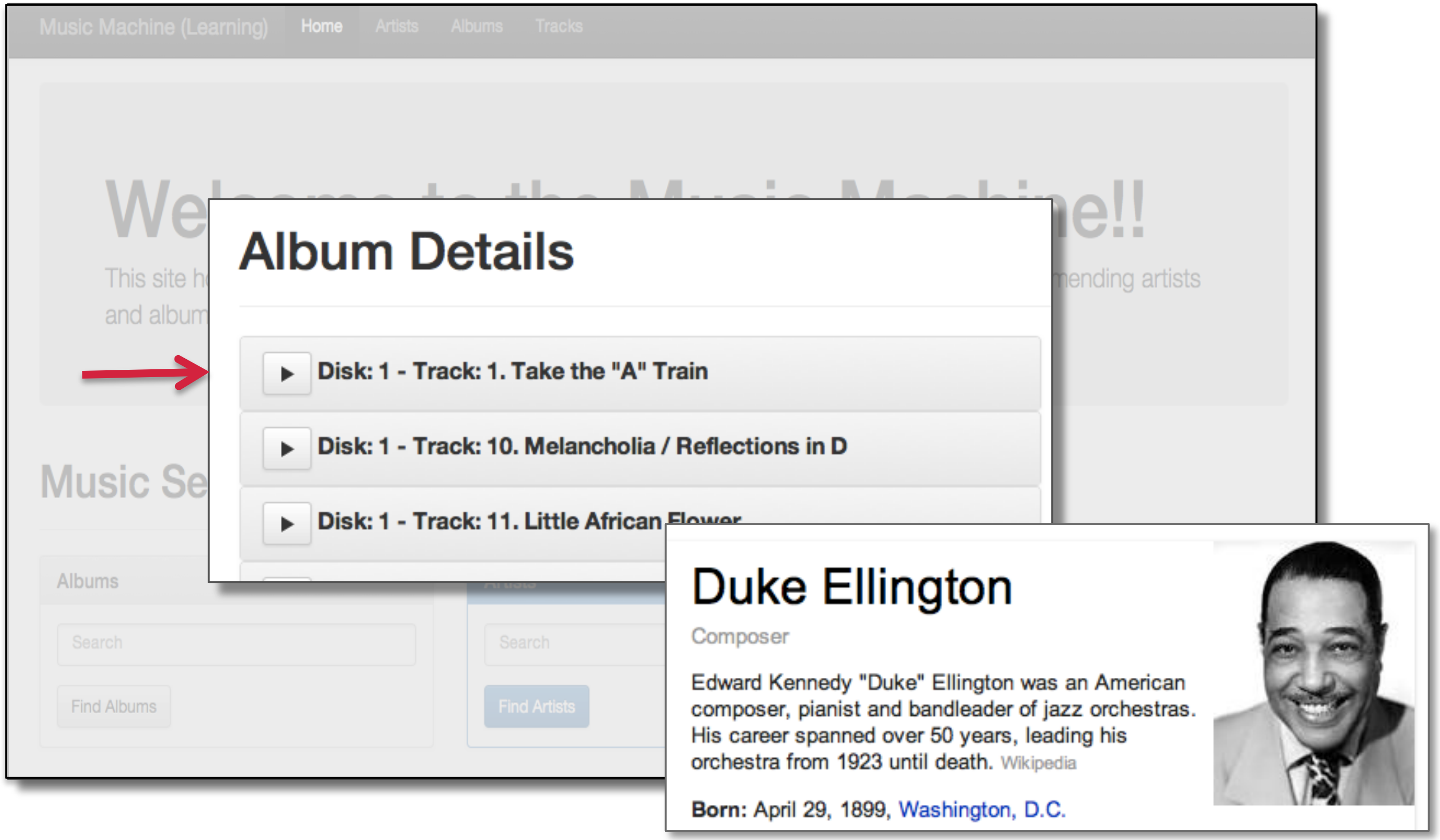
Going Further: Multi-Modal Recommendation



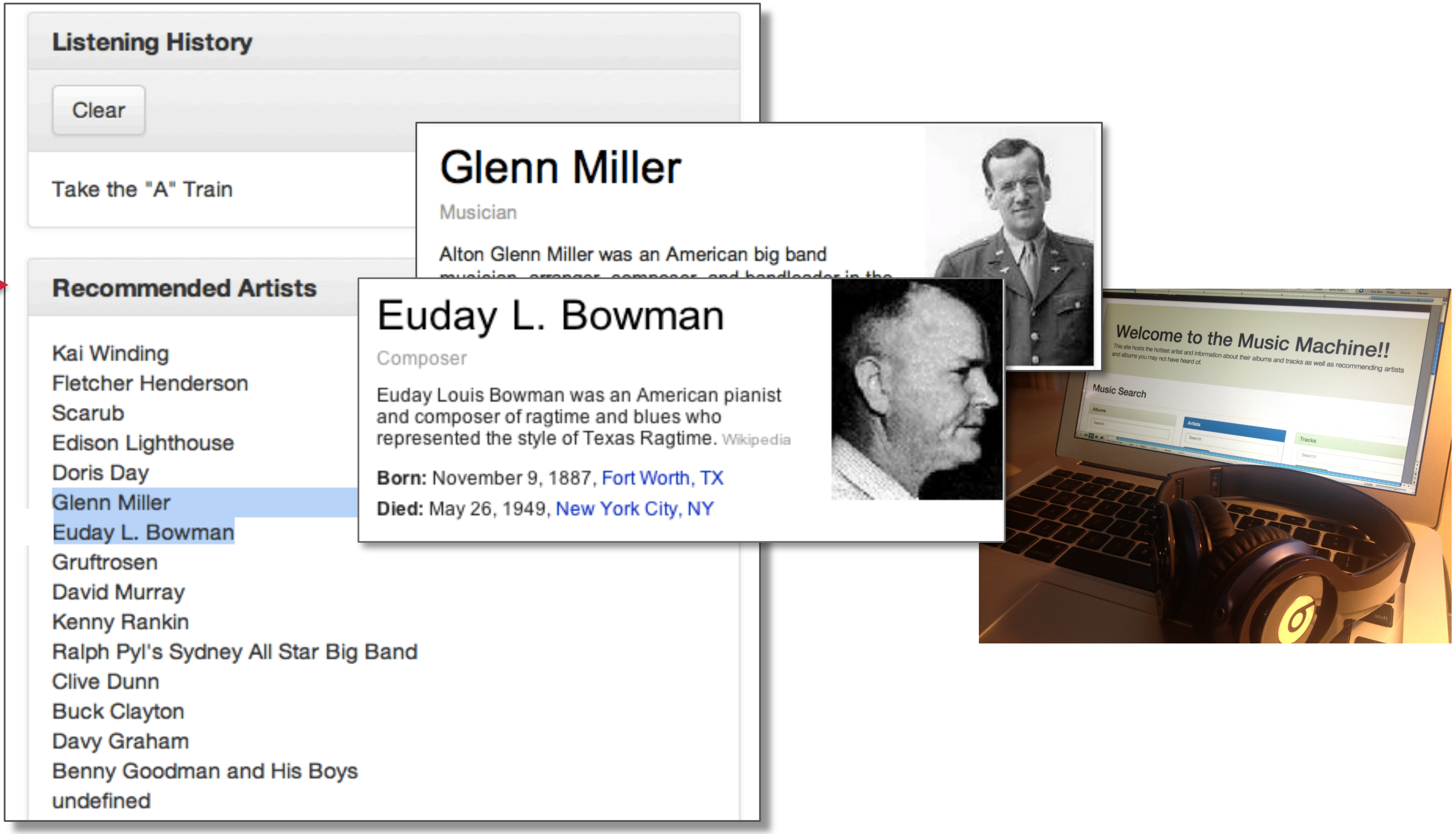
Going Further: Multi-Modal Recommendation



User activity: Listens to classic jazz hit "Take the A Train"



System delivers recommendations based on activity



Music Meta Data for Search Document Collections

- MusicBrainz data

```
259799  e891400b-b606-485c-9cd3-942e284e86b5    Ron Broder      Person  Unknown None
92252   147de9f1-2083-4fe9-82c2-2487ed626d9b    White Willow    Group   Norway  None
464647  382252b1-4c55-49c8-a36e-d968e2aa9fd9    Cosigner        Group   Unknown None
248525  aab74938-24ad-40b3-80e8-f6e891c35b45    Sarah McLeod    Person  Australia
       Female
307953  72906521-9d78-438f-866a-b2d741c38c90    Kolera  Group   Finland None
268138  467836a6-1a89-43d5-a4c0-78abb4828074    Lady Jane Grey  Group   Unknown None
191980  e246d7c2-f42a-4e17-9488-b7b065d641bc    Darkwood Dub    Group   Unknown None
60457   d7cb3da1-9d28-43aa-a7be-979678f53c2a    Spermbirds      Group   Germany None
536682  b6cc3141-ce39-4e81-9d80-cb8a6eb33329    Origami Group   Unknown None
883091  d236f9d0-364f-4a8c-9827-6fdd84248a04    山田裕介        Person  Japan   Male
628985  be4c7509-3db4-482e-ac5e-ac694b525ade    Marco Mzee      Person  Austria Male
398109  9127af94-581e-4d92-9369-46973408230d    Taj Mahal & The International Rhythm
 Band       Group   Unknown None
```

- Data includes Artist ID, MusicBrainz ID, Name, Group/Person, From (geo locations) and Gender as seen in this sample

Sample User Behavior Histories: Music Log Files

User ID

Time

Artist ID

Track ID

Event type

```
13      START     10113  2182654281
23      BEACON    10113  2182654281
24      START     10113  796006 11935028
34      BEACON    10113  796006 11935[illegible]
44      BEACON    10113  796006 11935[illegible]
54      BEACON    10113  796006 11935028
64      BEACON    10113  796006 11935028
74      BEACON    10113  796006 11935028
84      BEACON    10113  796006 11935028
94      BEACON    10113  796006 11935028
104     BEACON    10113  796006 11935028
109     FINISH 10113  796006 11935028
111     START     10113  589999 12011972
12[illegible]     BEACON    10113  589999 12011972
```

Sample Music Log Files

What has user 119 done here in the highlighted lines?

```
9669,BEACON,119,683689,10627847
9679,BEACON,119,683689,10627847
9689,BEACON,119,683689,10627847
9694,FINISH,119,683689,10627847
9694,START,119,2461,7020836
9694,BEACON,119,2461,7020836
9704,BEACON,119,2461,7020836
9714,BEACON,119,2461,7020836
9724,BEACON,119,2461,7020836
9734,BEACON,119,2461,7020836
9744,BEACON,119,2461,7020836
9754,BEACON,119,2461,7020836
9764,BEACON,119,2461,7020836
9774,BEACON,119,2461,7020836
9784,BEACON,119,2461,7020836
9794,BEACON,119,2461,7020836
9804,FINISH,119,2461,7020836
9804,START,119,2461,13767566
```

Artist ID for jazz musician Duke Ellington

Search Abuse for Music

Notice that data from indicator matrix of trained Mahout recommender model has been added to indicator field in documents of the artists collection

```
id              1710
mbid    592a3b6d-c42b-4567-99c9-ecf63bd66499
name    Chuck Berry
area    United States
gender          Male
indicator_artists 386685,875994,637954,3418,1344,789739,1460, …
```

```
id              541902
mbid    983d4f8f-473e-4091-8394-415c105c4656
name    Charlie Winston
area    United Kingdom
gender          None
indicator_artists 997727,815,830794,59588,900,2591,1344,696268, …
```


Another Example

- Users enter queries (A)
  - (actor = user, item=query)
- Users view videos (B)
  - (actor = user, item=video)
- A^TA gives query recommendation (query cooccurrence)
  - “did you mean to ask for”
- B^TB gives video recommendation (view cooccurrence)
  - “you might like these videos”

The Punch-line

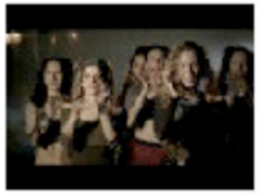
- $B^T A$ recommends videos in response to a query
  - (isn't that a search engine?)
  - (not quite, it doesn't look at content or meta-data)

Real-life Example

- Query: “Paco de Lucia”
- Conventional meta-data search results:
  - “hombres de paco” times 400
  - not much else
- Recommendation based search:
  - Flamenco guitar and dancers
  - Spanish and classical guitar
  - Van Halen doing a classical/flamenco riff

Real-life Example

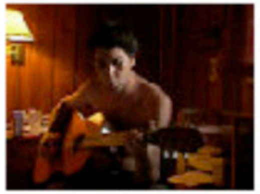
CONCIERTO CIUDAD DE LAS IDEAS PARTE FINAL
Music
58 views



Siudy / Buleria
Music
722 views

Vicente Amigo 2ª parte Ciudad de las Ideas
Music
124 views

Van Halen's Eruption
Music
4400 views



Freestyle Flamenco
Music
653 views

Hypothetical Example

- Want a navigational ontology?
- Just put labels on a web page with traffic
  - This gives A = users x label clicks
- Remember viewing history
  - This gives B = users x items
- Cross recommend
  - B' A = label to item mapping
- After several users click, results are whatever users think they should be

More Details Available

available for free at
http://www.mapr.com/practical-machine-learning

Bonus Round:

When worse is better

The Real Issues After First Production

- Exploration
- Diversity
- Speed

- Not the last fraction of a percent

Result Dithering

- Dithering is used to re-order recommendation results
  - Re-ordering is done randomly

- Dithering is *guaranteed* to make off-line performance worse

- Dithering also has a near perfect record of making actual performance much better

Result Dithering

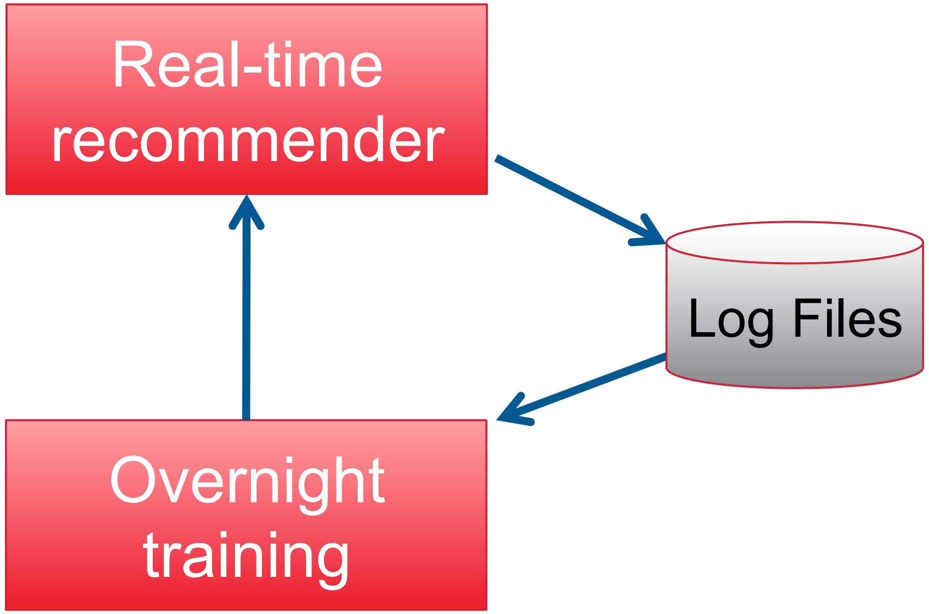
- Dithering is used to re-order recommendation results
  - Re-ordering is done randomly

- Dithering is *guaranteed* to make off-line performance worse

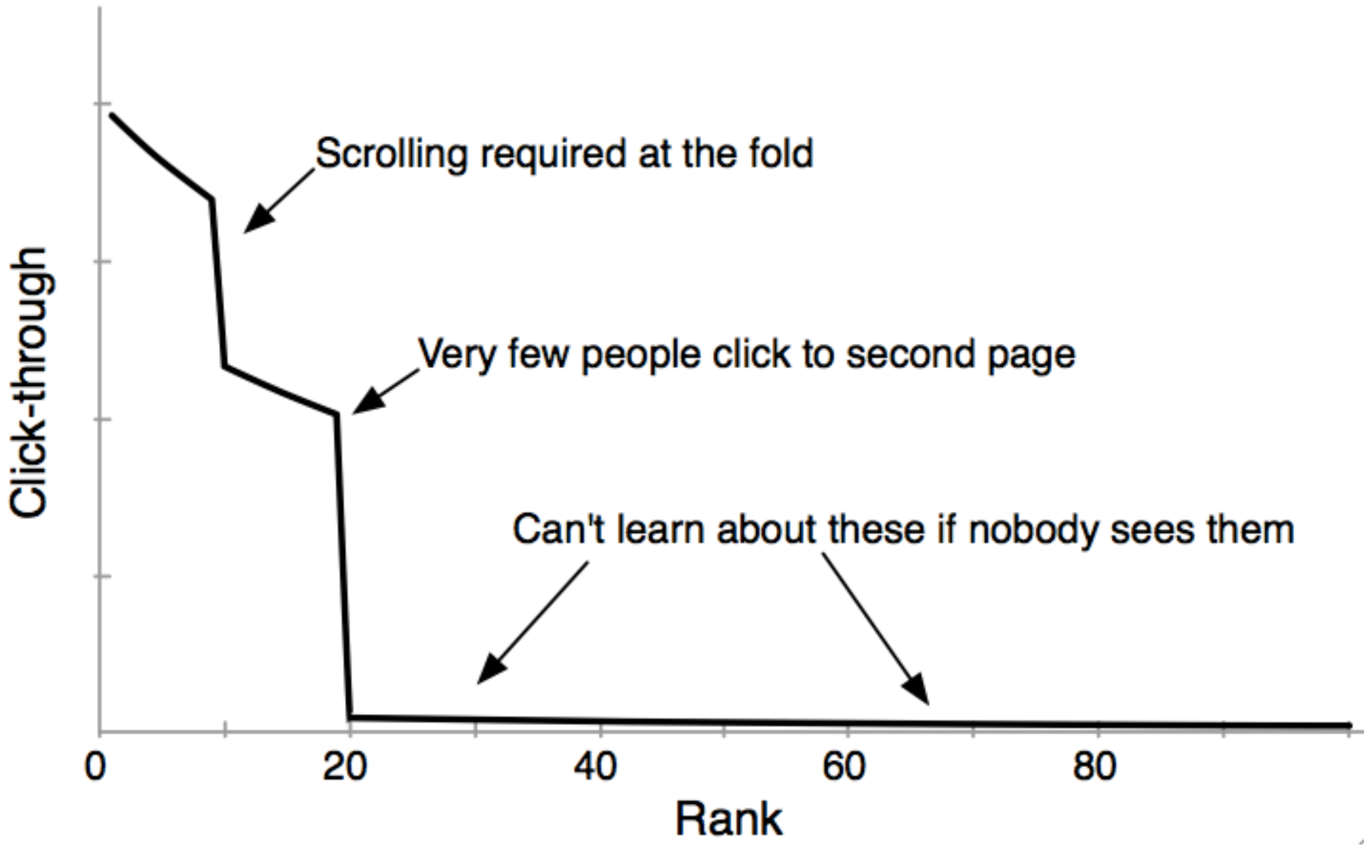
- Dithering also has a near perfect record of making actual performance much better

"Made more difference than any other change"

Why Dithering Works



Why Use Dithering?



Simple Dithering Algorithm

- Synthetic score from log rank plus Gaussian

$$s = \log r + N(0, \log \varepsilon)$$

- Pick noise scale to provide desired level of mixing

$$\frac{\Delta r}{r} \propto \varepsilon$$

- Typically

$$\varepsilon \in [1.5, 3]$$

- Also… use floor(t/T) as seed

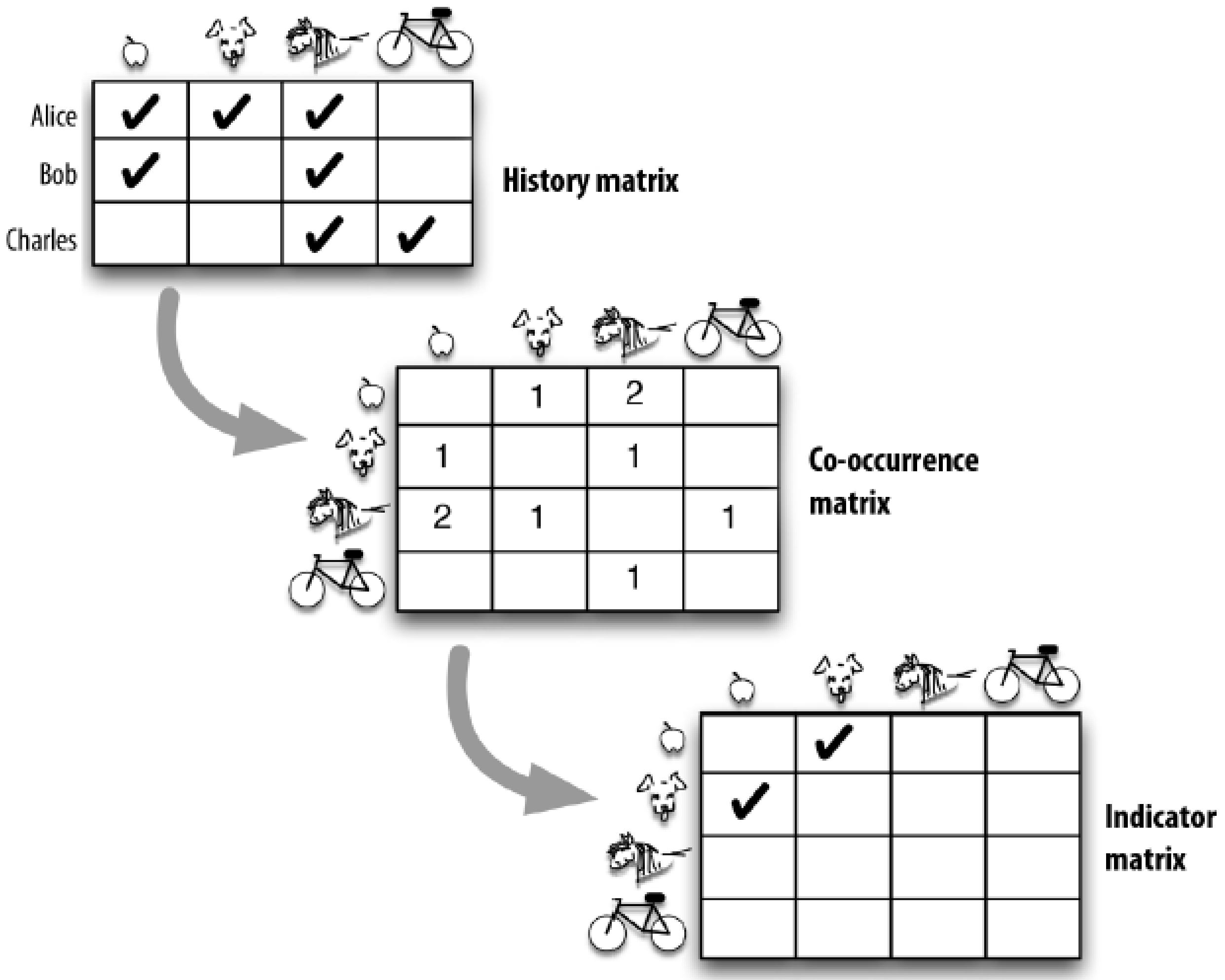
Example … ε = 2

1	2	8	3	9	15	7	6
1	8	14	15	3	2	22	10
1	3	8	2	10	5	7	4
1	2	10	7	3	8	6	14
1	5	33	15	2	9	11	29
1	2	7	3	5	4	19	6
1	3	5	23	9	7	4	2
2	4	11	8	3	1	44	9
2	3	1	4	6	7	8	33
3	4	1	2	10	11	15	14
11	1	2	4	5	7	3	14
1	8	7	3	22	11	2	33

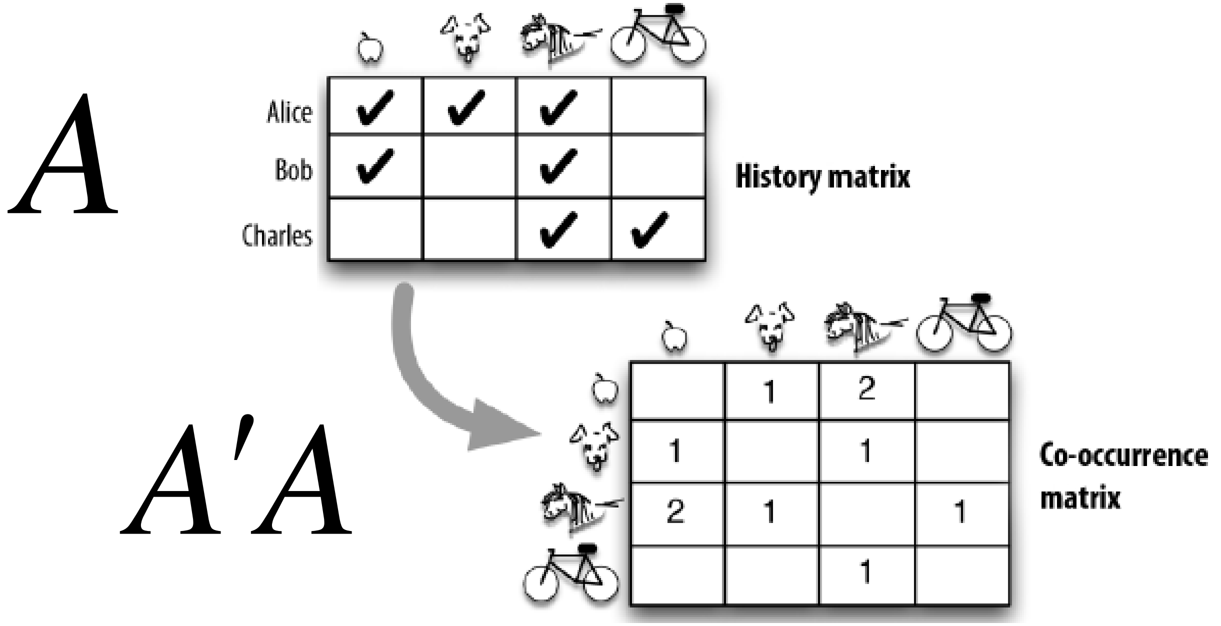
2nd Bonus Round:

Real-time puppies and ponies

Cooccurrence Analysis

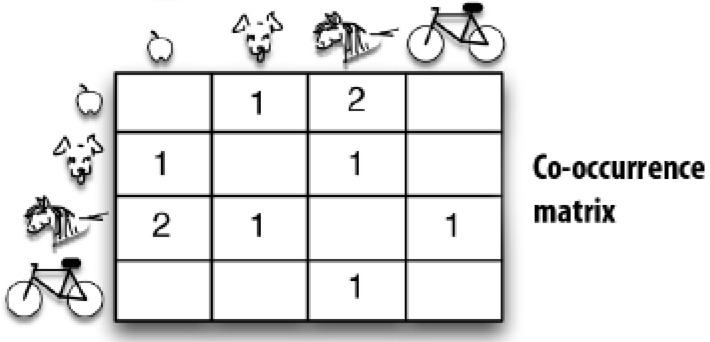


How Often Do Items Co-occur

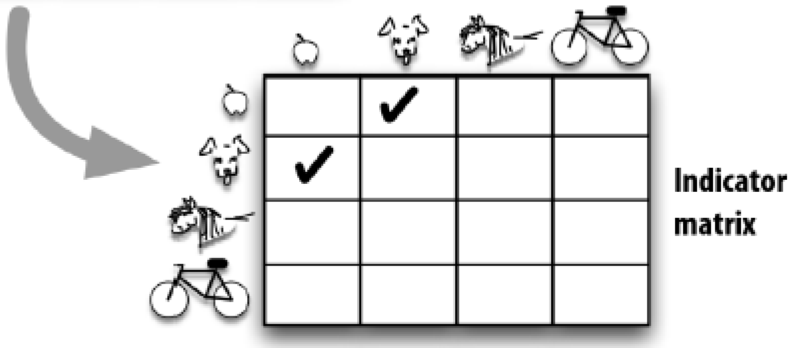


Which Co-occurrences are Interesting?

$$A'A$$



$$sparsify(A'A)$$



Each row of indicators becomes a field in a search engine document

Cooccurrence Mechanics

- Cooccurrence is just a self-join

```
for each user, i
    for each history item j₁ in A_i*
        for each history item j₂ in A_i*
            count pair (j₁, j₂)
```

$$\sum_i a_{ij_1} a_{ij_2} = A'A$$

Cross-occurrence Mechanics

- Cross occurrence is just a self-join of adjoined matrices

```
for each user, i
        for each history item j₁ in A_{i*}
          for each history item j₂ in B_{i*}
             count pair (j₁, j₂)
```

$$\sum_i a_{ij_1} b_{ij_2} = A'B \qquad [A \mid B]'[A \mid B] = \begin{bmatrix} A'A & A'B \\ B'A & B'B \end{bmatrix}$$

A word about scaling

A few pragmatic tricks

- Downsample all user histories to max length (interaction cut)
  - Can be random or most-recent (no apparent effect on accuracy)
  - Prolific users are often pathological anyway
  - Common limit is 300 items (no apparent effect on accuracy)

- Downsample all items to limit max viewers (frequency limit)
  - Can be random or earliest (no apparent effect)
  - Ubiquitous items are uninformative
  - Common limit is 500 users (no apparent effect)

Schelter, et al. Scalable similarity-based neighborhood methods with MapReduce. *Proceedings of the sixth ACM conference on Recommender systems*. 2012

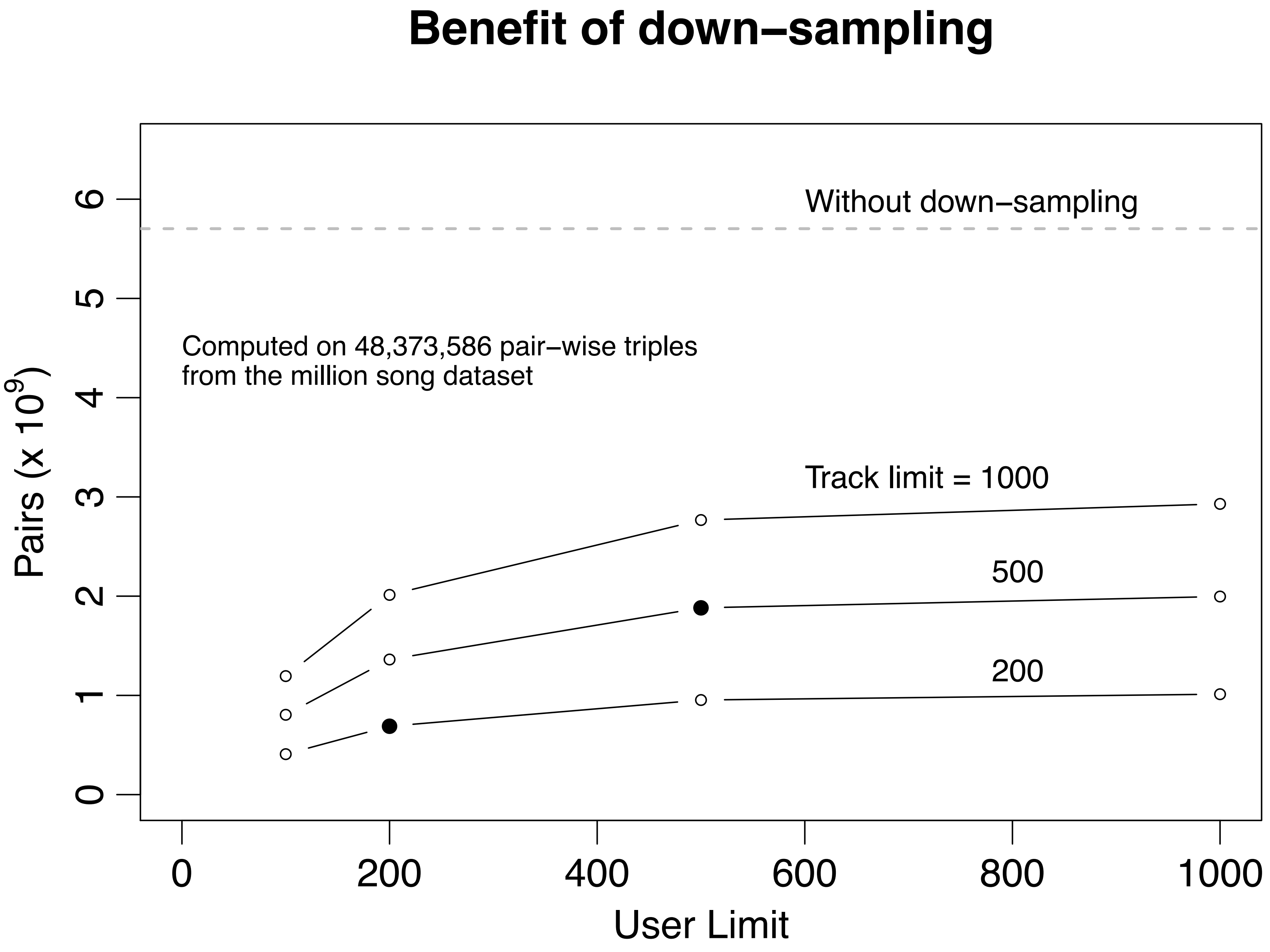
But note!

- Number of pairs for a user history with k_i distinct items is ≈ $k_i^2/2$
- Average size of user history increases with increasing dataset
  - Average may grow more slowly than *N* (or not!)
  - Full cooccurrence cost grows strictly faster than *N*

$$t \propto \sum_i k_i^2 \in o(N)$$

  - i.e. it just doesn't scale
- Downsampling interactions places bounds on per user cost
  - Cooccurrence with interaction cut is scalable

$$t \propto \sum_i \min\left(k_{\max}^2, k_i^2\right) < N k_{\max}^2 \in O(N)$$



Benefit of down−sampling
Without down−sampling
Computed on 48,373,586 pair−wise triples
from the million song dataset
Track limit = 1000
500
200
Pairs (x 10^9)
User Limit
0
200
400
600
800
1000
0
1
2
3
4
5
6

Batch Scaling in Time Implies Scaling in Space

- Note:
  - With frequency limit sampling, max cooccurrence count is small (<1000)
  - With interaction cut, total number of non-zero pairs is relatively small
  - Entire cooccurrence matrix can be stored in memory in ~10-15 GB
- Specifically:
  - With interaction cut, cooccurrence scales in size

$$\left|A'A\right|_0 \in O(N)$$

  - Without interaction cut, cooccurrence does not scale size-wise

$$\left|A'A\right|_0 \in \omega(N)$$

Impact of Interaction Cut Downsampling

- Interaction cut allows batch cooccurrence analysis to be *O(N) in time and space*
- This is intriguing
  - Amortized cost is low
  - Could this be extended to an on-line form?
- Incremental matrix factorization is hard
  - Could cooccurrence be a key alternative?
- Scaling matters most at scale
  - Cooccurrence is very accurate at large scale
  - Factorization shows benefits at smaller scales

Online update

Requirements for Online Algorithms

- Each unit of input must require O(1) work
  - Theoretical bound
- The constants have to be small enough on average
  - Pragmatic constraint
- Total accumulated data must be small (enough)
  - Pragmatic constraint

Space Bound Implies Time Bound

- Because user histories are pruned, only a limited number of value updates need be made with each new observation

- This bound is just twice the interaction cut k_{max}
  - Which is a constant

- Bounding the number of updates trivially bounds the time

Implications for Online Update

$$\left(A+e_{ij}\right)'\left(A+e_{ij}\right)-A'A=e'_{ij}A+A'e_{ij}+\delta_{jj}$$

$$=\begin{bmatrix} 0 \\ A_{i*} \\ 0 \end{bmatrix}+\begin{bmatrix} 0 & \left(A_{i*}\right)' & 0 \end{bmatrix}+\delta_{jj}$$

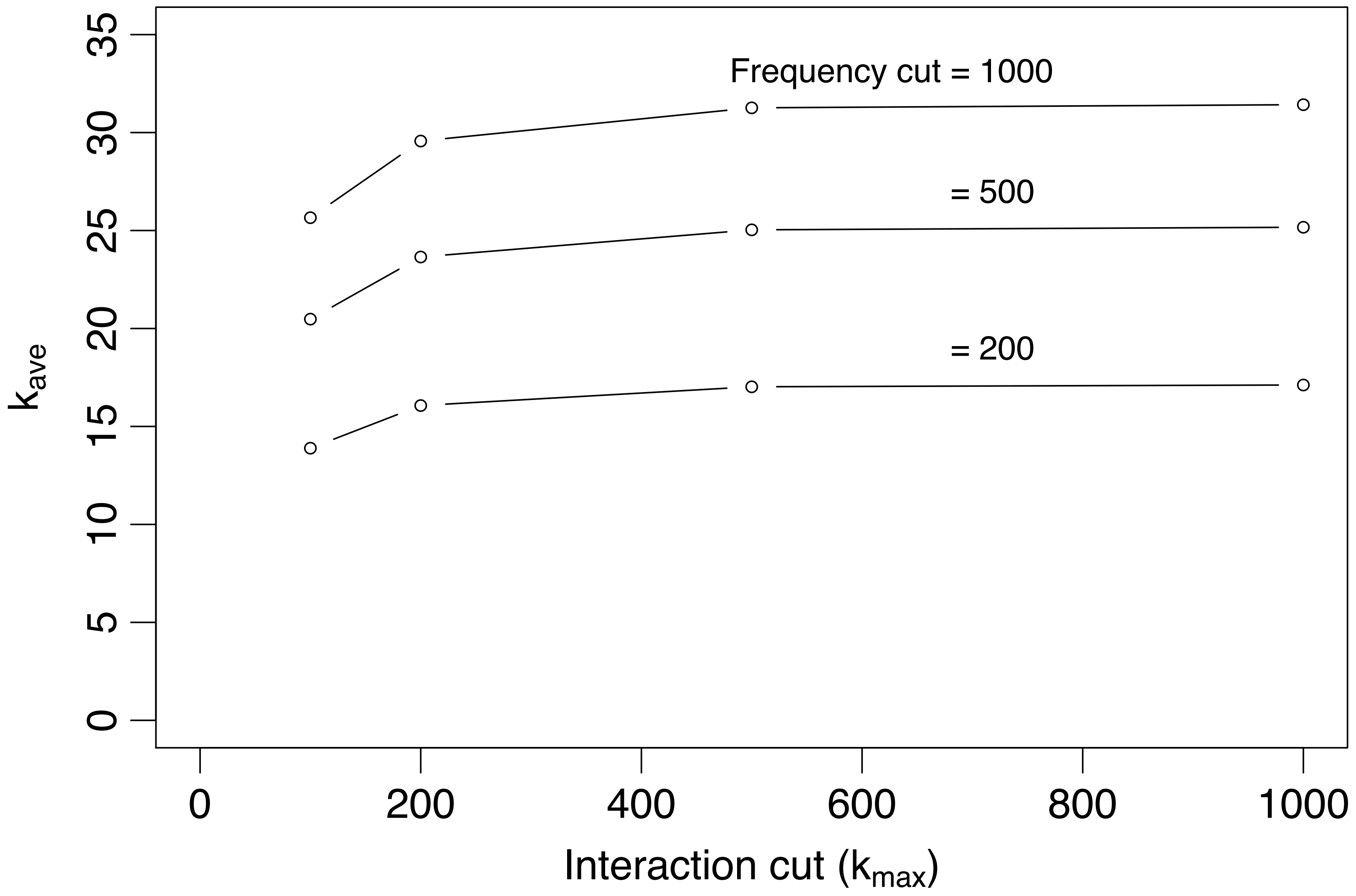
With interaction cut at $k_{\max}$

$$\left| A_{i*} + \left(A_{i*} \right)' + \delta_{ii} \right|_0 \leq 2k_{\max} - 1 \in O(1)$$

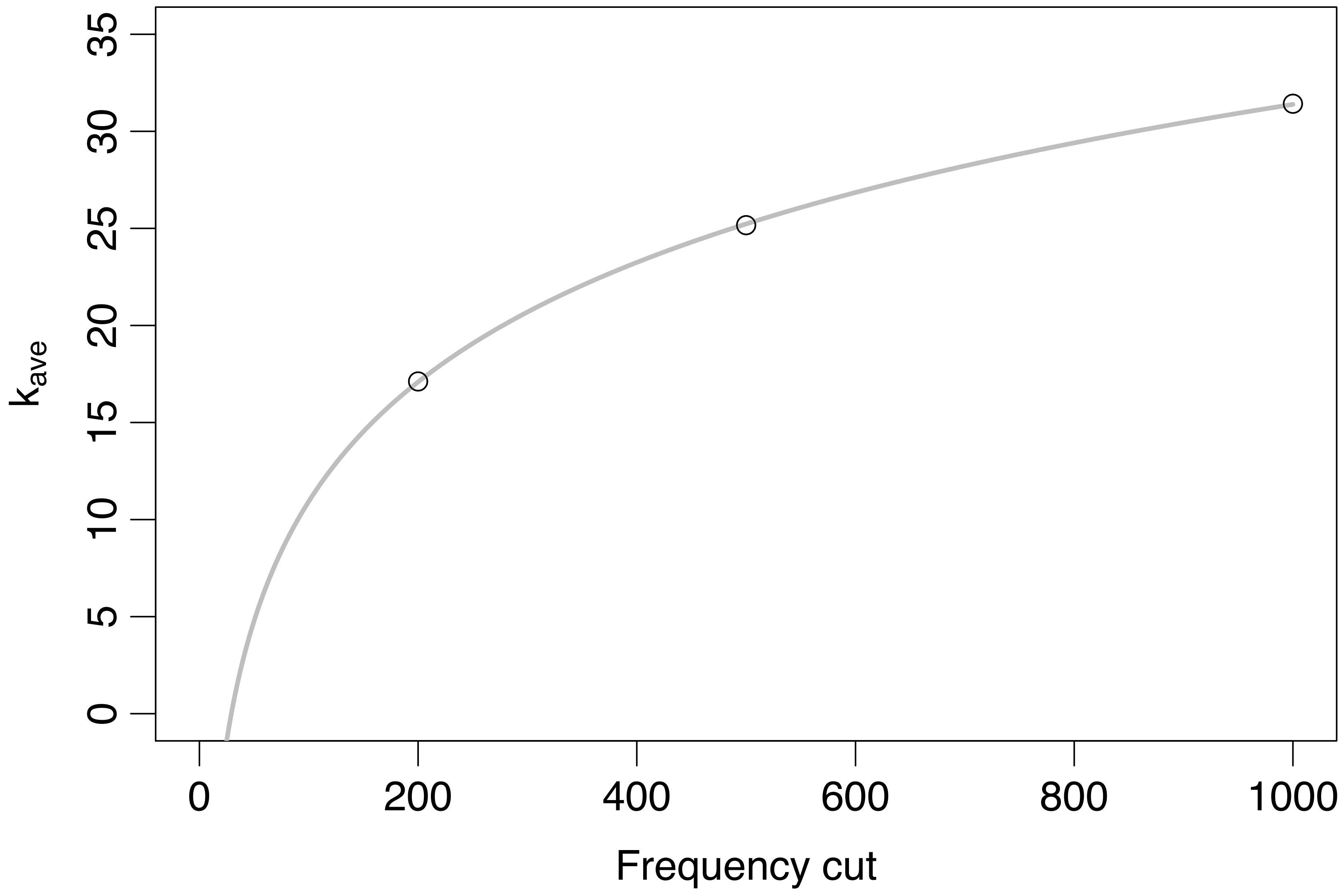
But Wait, It Gets Better

- The new observation may be pruned
  - For users at the interaction cut, we can ignore updates
  - For items at the frequency cut, we can ignore updates
  - Ignoring updates only affects indicators, not recommendation query
  - At million song dataset size, half of all updates are pruned
- On average k_i is much less than the interaction cut
  - For million song dataset, average appears to grow with log of frequency limit, with little dependency on values of interaction cut > 200
- LLR cutoff avoids almost all updates to index
- Average grows slowly with frequency cut

 MAPR.



Frequency cut = 1000
= 500
= 200
k_{ave}
Interaction cut (k_{max})
0
5
10
15
20
25
30
35
0
200
400
600
800
1000



k_{ave}
0
5
10
15
20
25
30
35
0
200
400
600
800
1000
Frequency cut

Recap

- Cooccurrence-based recommendations are simple
  - Deploying with a search engine is even better
- Interaction cut and frequency cut are key to batch scalability

- Similar effect occurs in online form of updates
  - Only dozens of updates per transaction needed
  - Data structure required is relatively small
  - Very, very few updates cause search engine updates

- Fully online recommendation is very feasible, almost easy

More Details Available

available for free at

http://www.mapr.com/practical-machine-learning

Who I am

Ted Dunning, Chief Applications Architect, MapR Technologies
Email tdunning@mapr.com tdunning@apache.org
Twitter @Ted_Dunning

Apache Mahout https://mahout.apache.org/
Twitter @ApacheMahout

Apache Drill http://incubator.apache.org/drill/
Twitter @ApacheDrill

Q&A

Engage with us!

@mapr	Twitter	Facebook	maprtech
mapr-technologies	LinkedIn	Google+	MapR
tdunning@mapr.com	Email	YouTube	maprtech