RECOMMENDATION SYSTEM OF YELP RANK

Elizabeth Karpinski,
JingNan Xu, Xiaoshuai Li,
Qian Wang

1. Belmont Vegetarian Restaurant

147 reviews

$ · Vegan, American (New)

157 Belmont St
Worcester, MA 01608
(508) 798-8898

I would have preferred the pepper steak with a little heat, so I added some scotch bonnet pepper sauce I had on hand and that brought the flavor up two-fold. • He sensed my indecision and…

2. Mare E Monti Trattoria

106 reviews

$$ · Italian

19 Wall St
Worcester, MA 01604
(508) 767-1800

This restaurant is truly a hidden gem, but you will definitely enjoy! • Have tried the mushroom ravioli and the seafood pasta, both were delicious and had very complex flavors. • I was…

3. Pomir Grill

102 reviews

$$ · Afghan

119 Shrewsbury St
Worcester, MA 01604
(508) 755-7333

We all ended the night with the house made pistachio ice cream...incredible. • In addition to the complimentary bread, we also ordered a potato pastry called kachalu bolani...this also was…

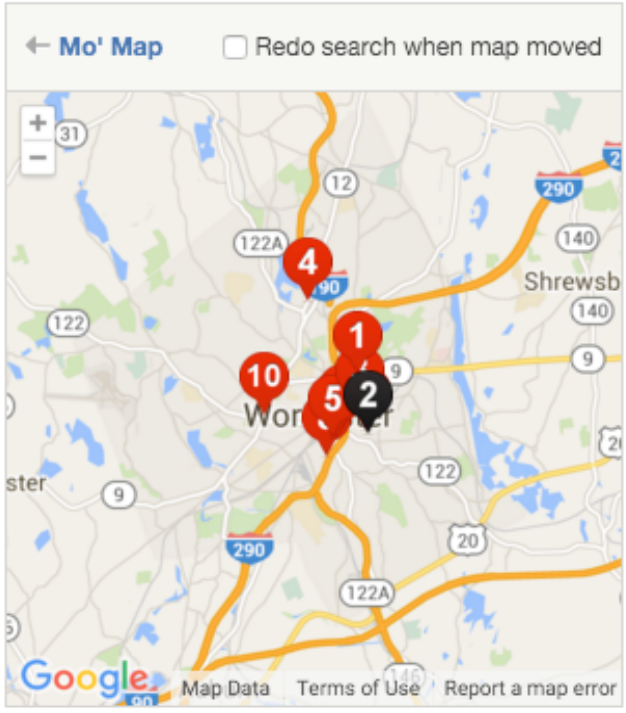
4. Fatima's Cafe

62 reviews

$ · African

43 W Boylston St
Worcester, MA 01605
(508) 762-9797

The samosas are probably the best ones I've ever had - I plan on catering with these for my next house party. • Everything on the platter was delicious, but my favorite was the goat stew.



amazon Prime All

WPI

More Top Picks for You

See more

Duty Calls
Download
and Play Today

Exclusive Coupons
for Prime members

Recommendations for You in Computers & Accessories

Recommendations for You in Electronics

WHY USING RECOMMENDER SYSTEMS?

Value for customer

- Find things that are interesting
- Narrow down the set of choices
- Discover new things

Value for provider

- Unique personalized service for the customer
- Increase trust and customer loyalty
- Increase sales, click trough rates, conversion etc.

REAL-WORLD CHECK

Myths from industry

- Amazon.com generates X percent of their sales through the recommendation lists ($30 < X < 70$)
- Netflix generates X percent of their sales through the recommendation lists ($30 < X < 70$)

There must be some value in it

- See recommendation of groups, jobs or people on LinkedIn
- Friend recommendation and ad personalization on Facebook
- Song recommendation at last.fm
- News recommendation at Forbes.com

MACHINE LEARNING AND PERSONALIZATION

Machine Learning can allow learning a *user model* or *profile* of a particular user based on:

- Sample interaction
- Rated examples

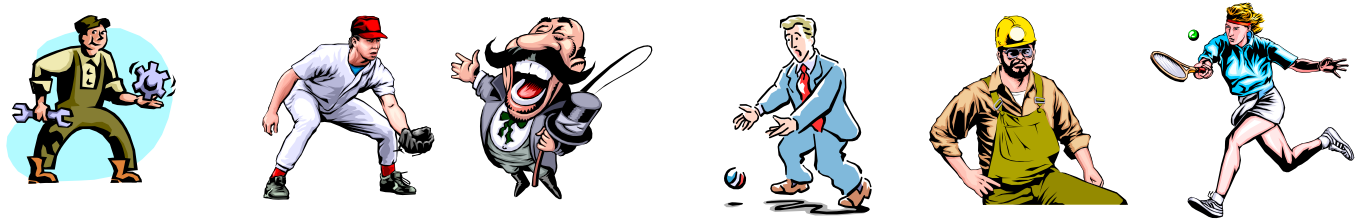
This model or profile can then be used to:

- Recommend items
- Filter information
- Predict behavior

RECOMMENDER SYSTEMS: BASIC TECHNIQUES

	Pros	Cons
Collaborative	No knowledge-engineering effort, serendipity of results, learns market segments	Requires some form of rating feedback, cold start for new users and new items
Content-based	No community required, comparison between items possible	Content descriptions necessary, cold start for new users, no surprises
Knowledge-based	Deterministic recommendations, assured quality, no cold-start, can resemble sales dialogue	Dnot react to short-term trends

COLLABORATIVE FILTERING



User Database

A 9
B 3
C
: :
Z 5

A
B
C 9
: :
Z 10

A 5
B 3
C
: :
Z 7

A
B
C 8
: :
Z

A 6
B 4
C
: :
Z

A 10
B 4
C 8
. .
Z 1

Correlation Match

A 9
B 3
C
: :
Z 5

A 10
B 4
C 8
. .
Z 1

Active User

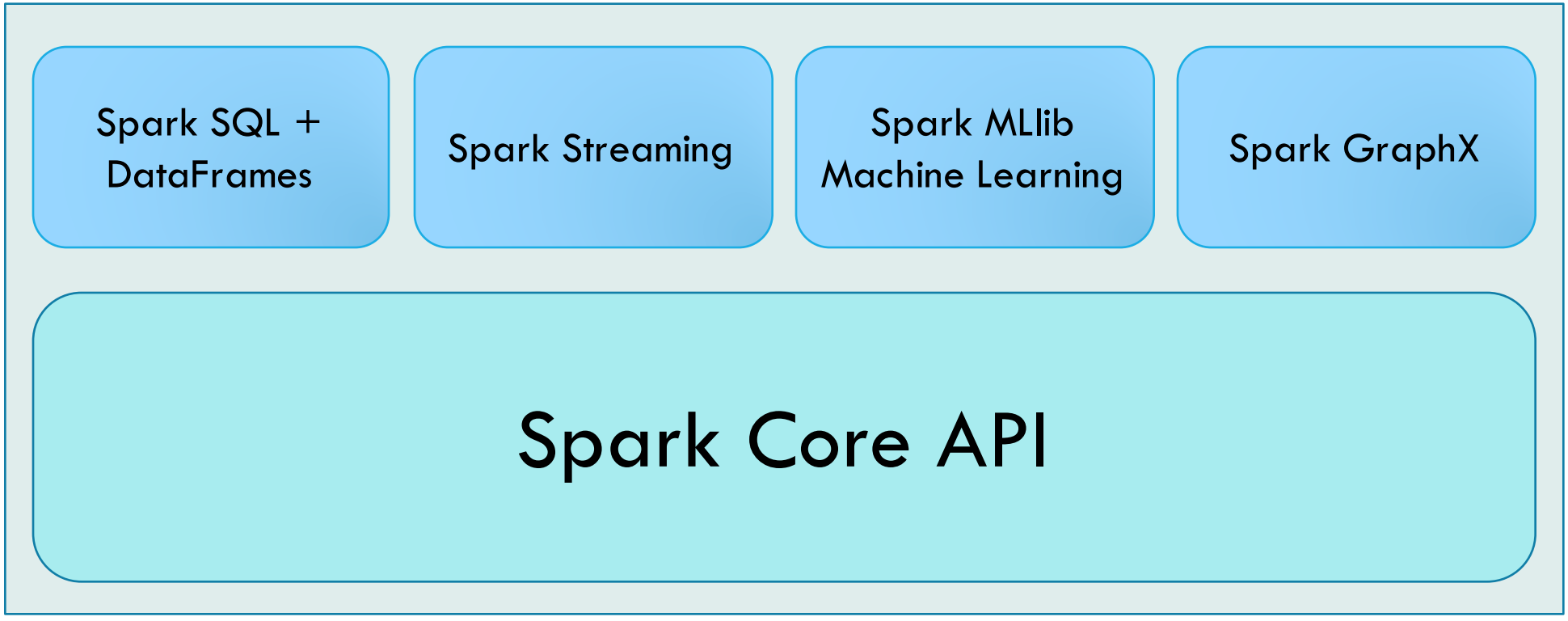
A 9
B 3
C
. .
Z 5

Extract Recommendations

C

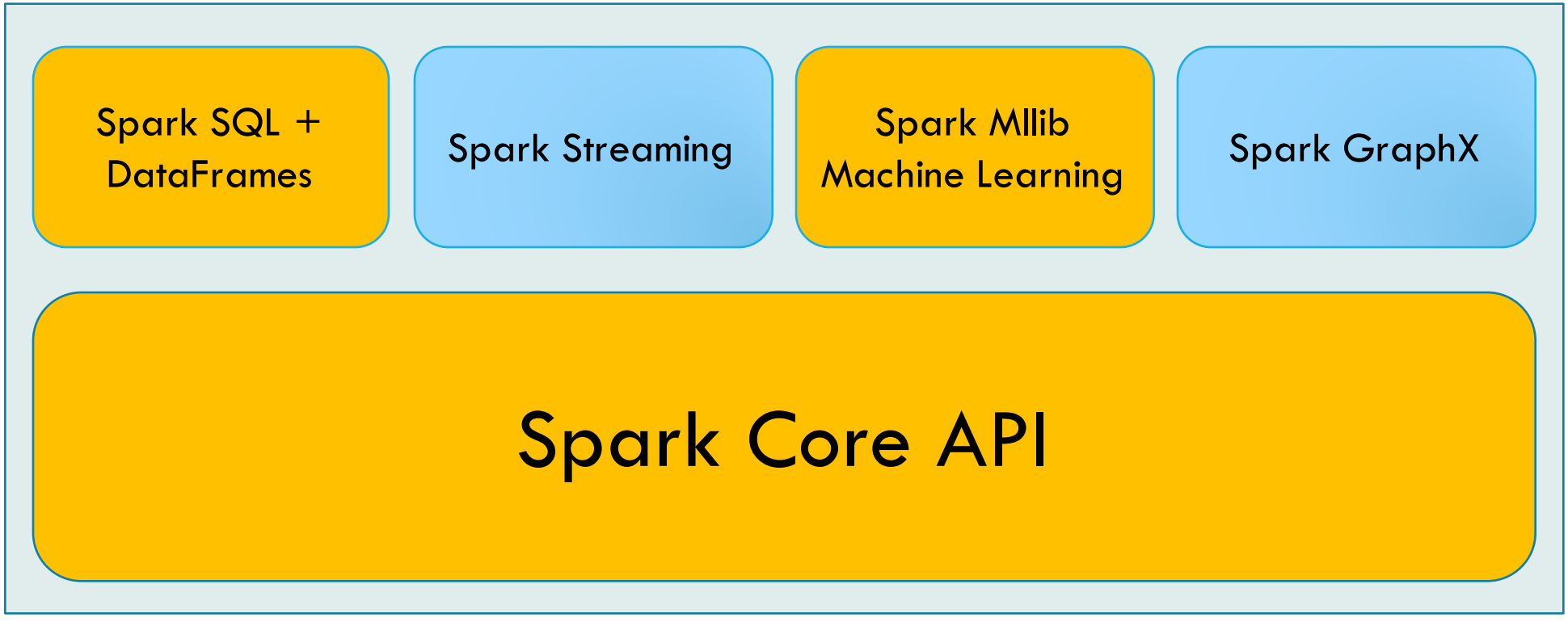
COLLABORATIVE FILTERING METHOD

SPARK ECOSYSTEM



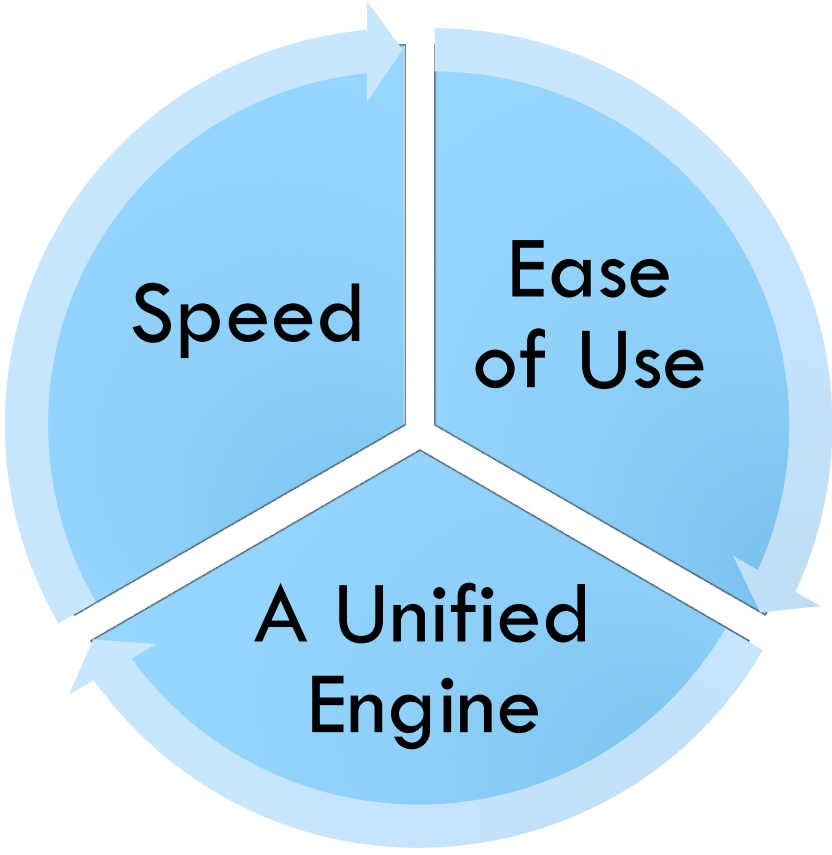
Apache Spark™ is a fast and general engine for large-scale data processing.

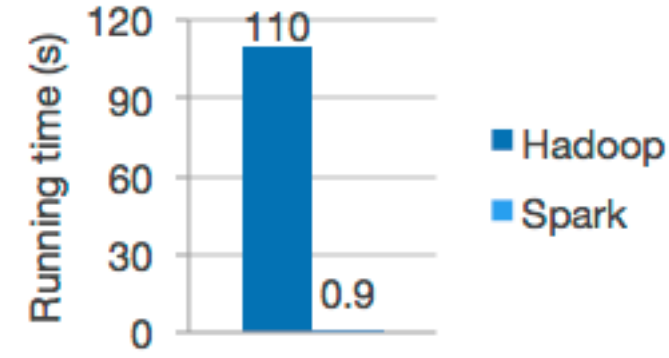
SPARK ECOSYSTEM



Apache Spark™ is a fast and general engine for large-scale data processing.

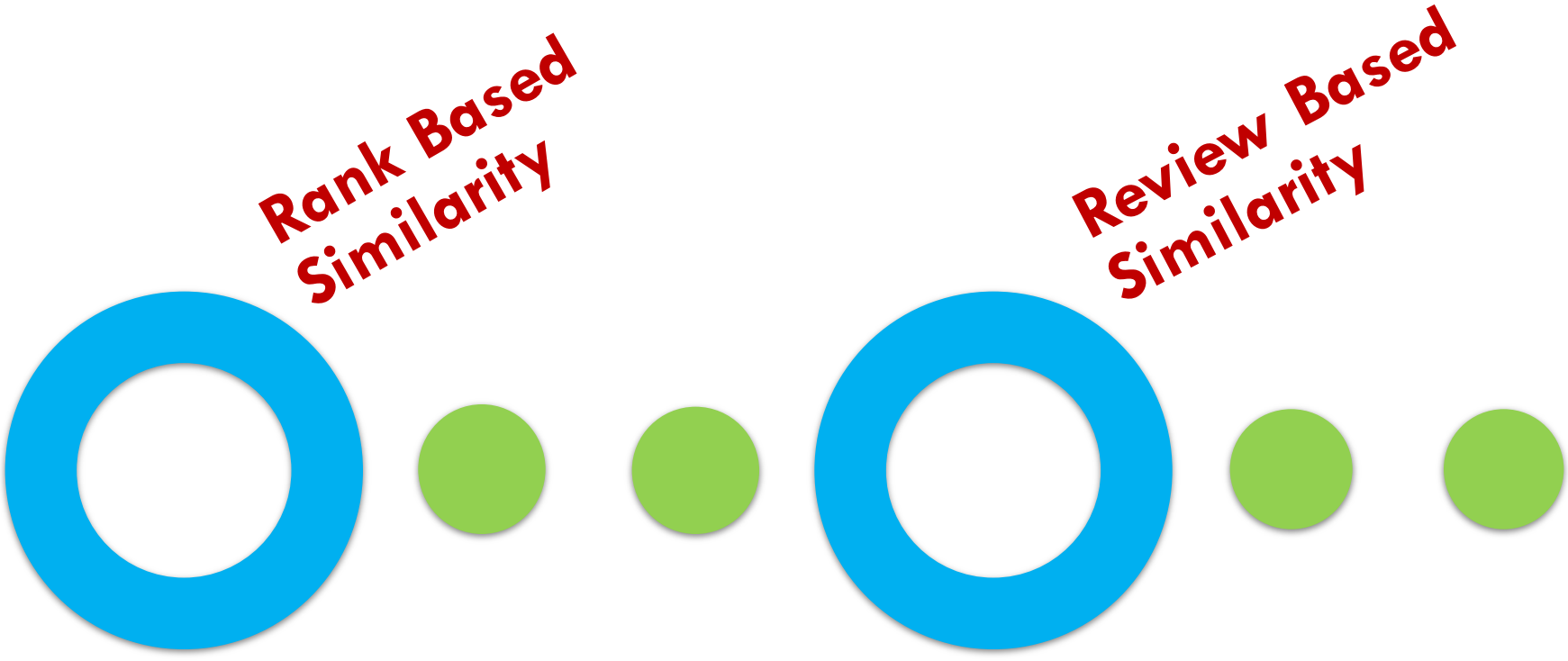
APACHE SPARK



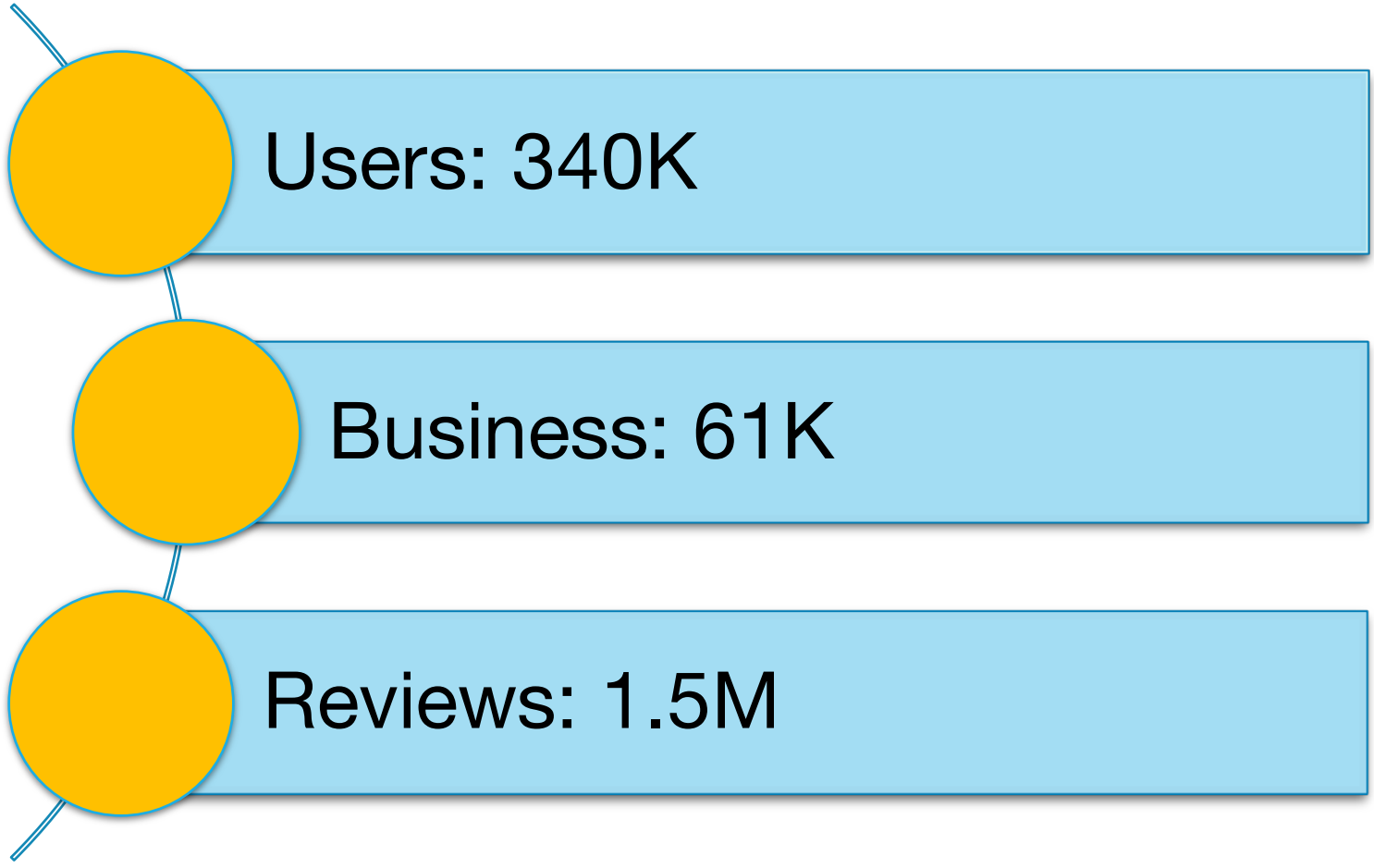


Logistic regression in Hadoop and Spark

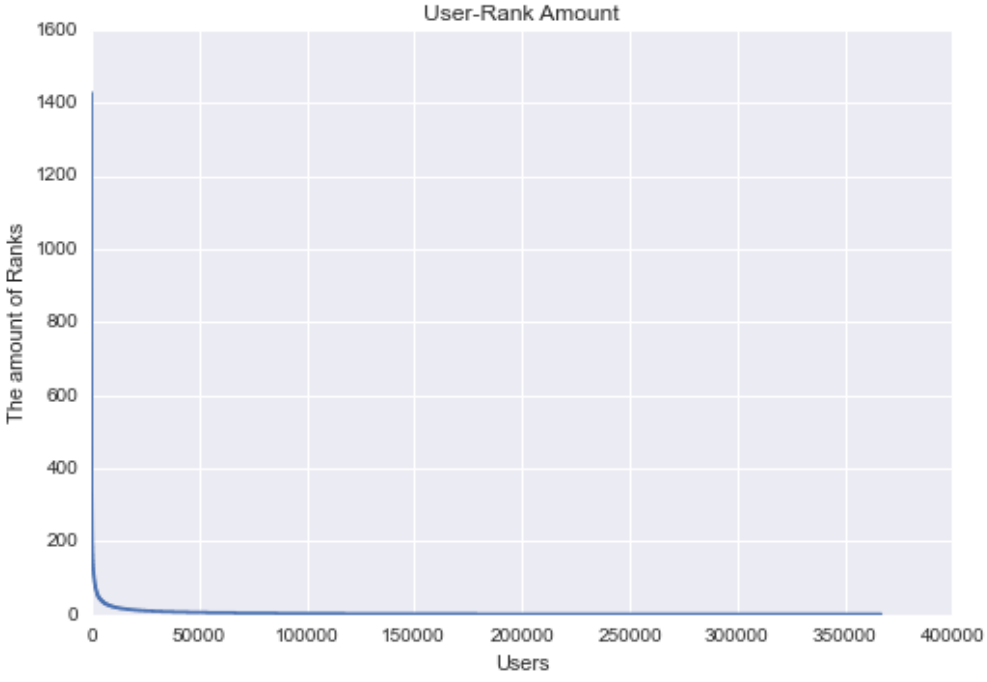
COLLABORATIVE FILTERING



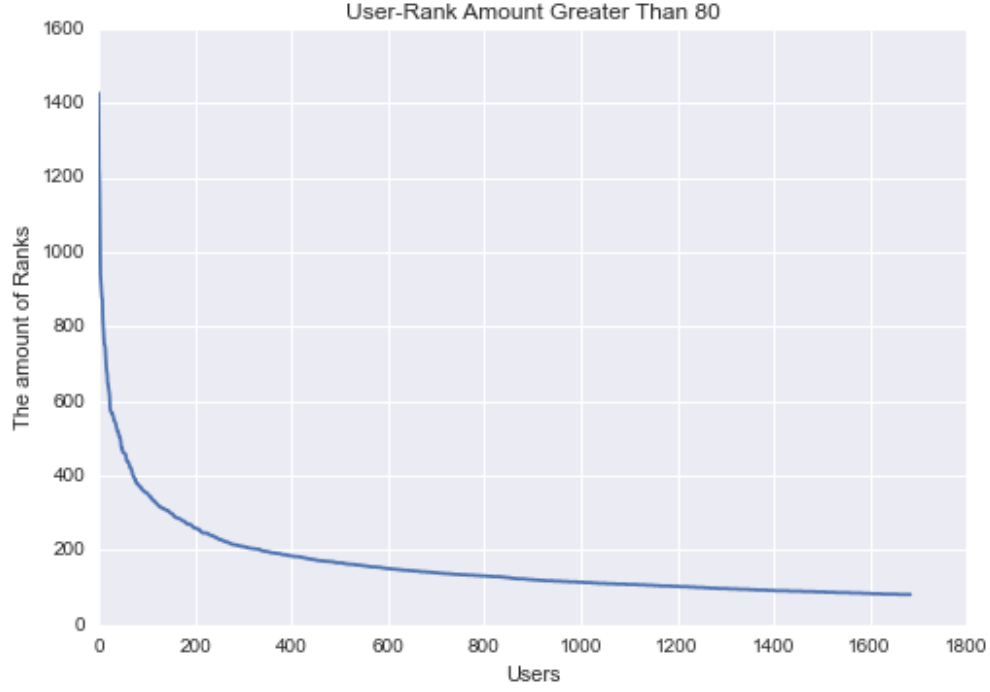
YELP ACADEMIC DATASET



USERS - RANKS AMOUNT

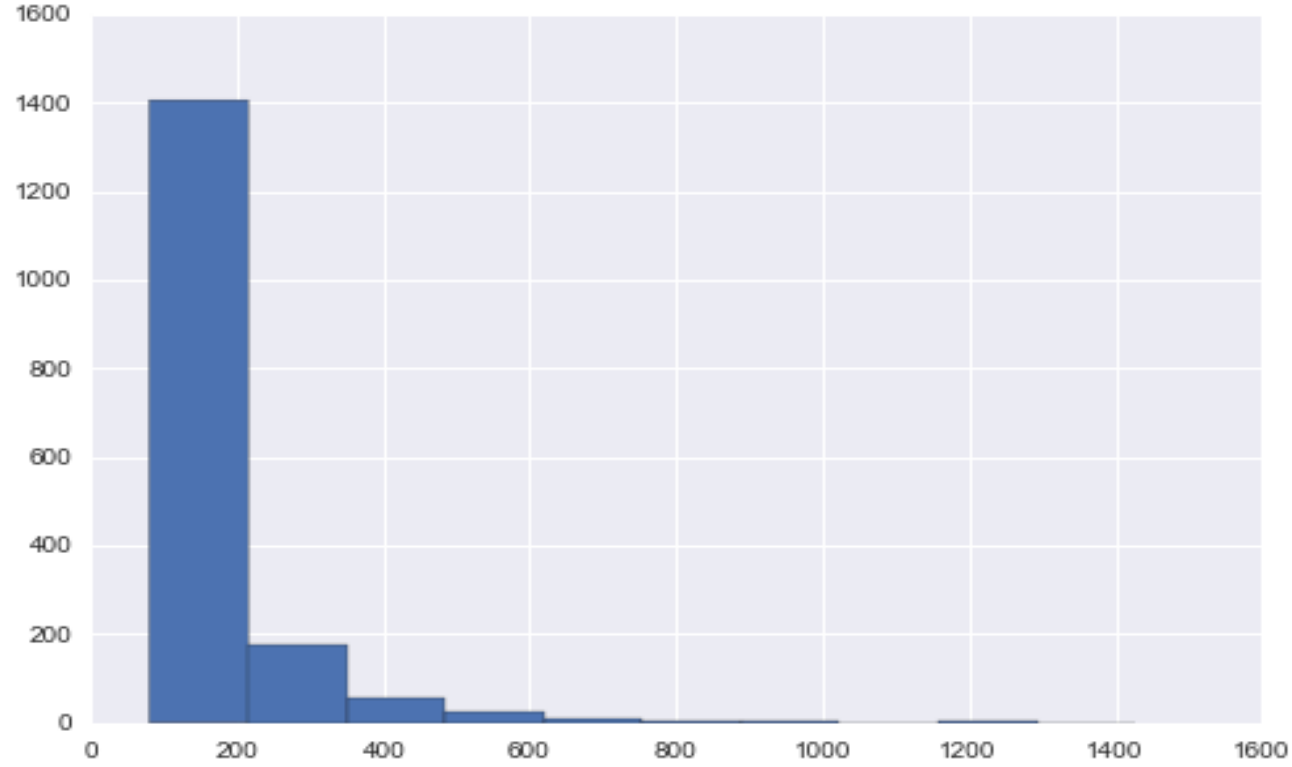


All Users



Users Who have more than 80 Reviews

THE DISTRIBUTION OF USERS' RANKS



We choose those users who have at least 80 reviews

MEASURING USER SIMILARITY

A popular similarity measure in user-based CF:

Pearson correlation

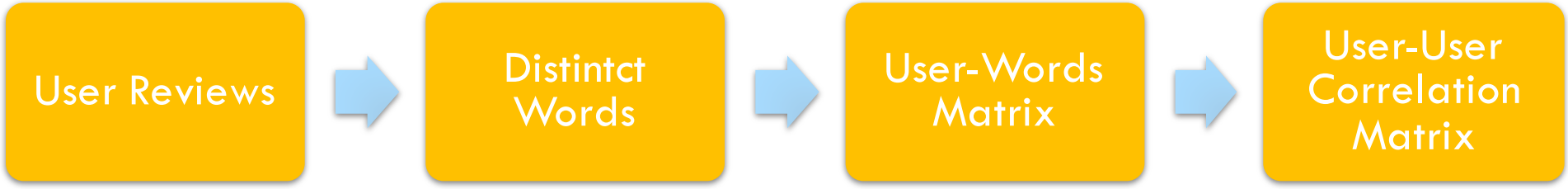
$$c_{a,u} = \frac{\text{covar}(r_a, r_u)}{\sigma_{r_a} \sigma_{r_u}}$$

	Business1	Business2	Business3	Business4	Business5
Alice	5	?	4	4	?
User1	3	?	2	?	3
User2	?	3	?	3	5
User3	3	?	1	5	?
User4	?	5	5	?	1

sim = 0,25
sim = -0.30
sim = -0.19

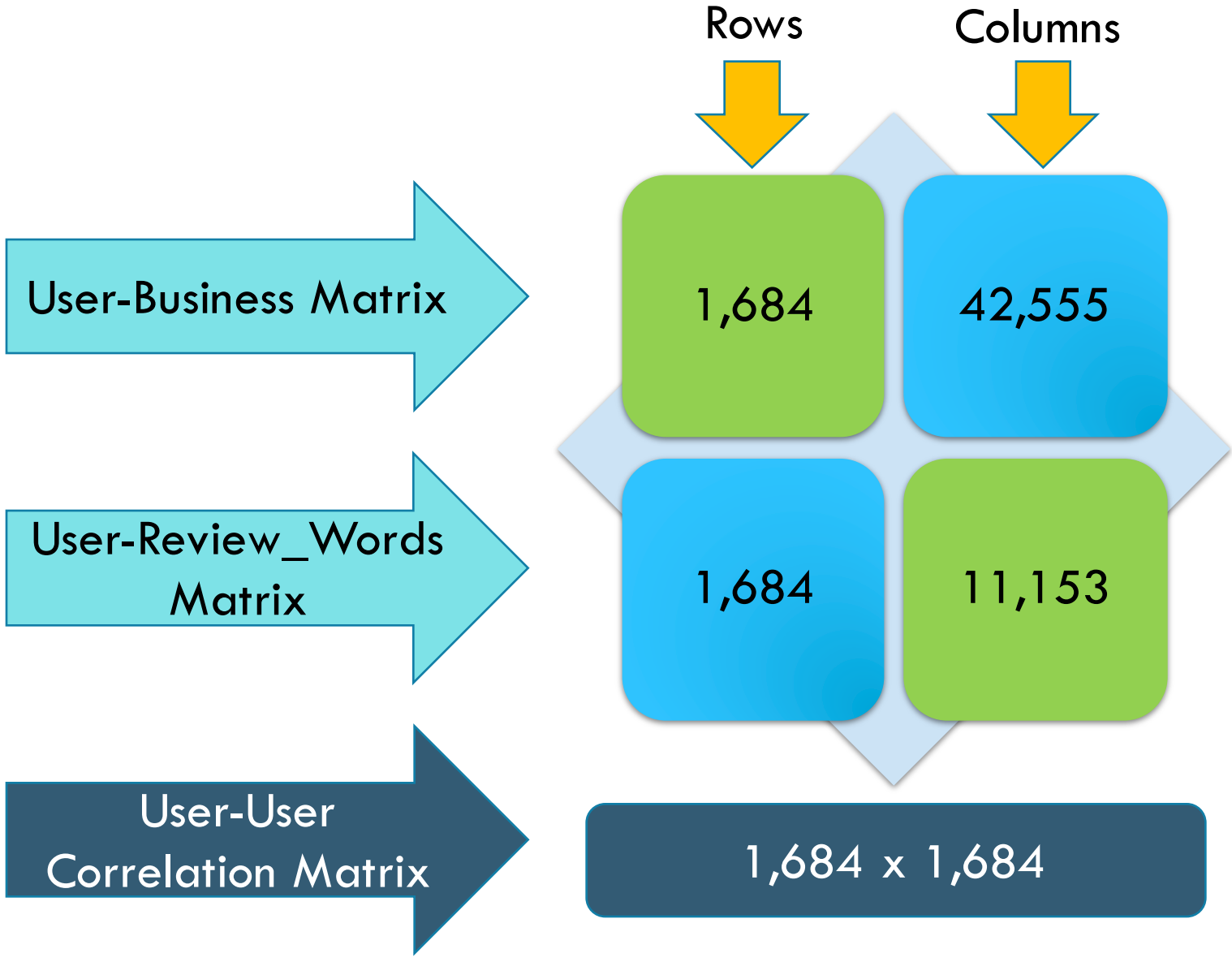
BASED ON REVIEW SIMILARIRY

Review Similarity Processes



	You	Great	Wow	Suck	Ooo~	zzzz	ome	The	……
User1	3	1	0	0	2	4	0	14	
User2	12	3	3	5	0	0	1	5	
User3	4	8	0	0	0	0	0	12	
User4	21	0	0	3	1	0	1	24	

USER-ITEM MATRIX



NEIGHBOR SELECTION

For a given active user, *u*, select correlated users to serve as source of predictions.

Standard approach

- Use the most similar *n* users' ranks based on similarity weights, $w_{a,u}$

Alternate approach

- Include all users whose similarity weight is above a given threshold

NEIGHBOR SELECTION

```
def getTopNeighbors(uid, n):
    aa_col = df_corr.orderBy(uid, ascending=0)\
                    .select(df_corr.user_id,uid).collect()
    top_neighbor = aa_col[1:n+1] # Because the first is itself
    return top_neighbor
```

```
def getStdRankofUser(uid):
    std_rank_user = np.std(df_review.filter(df_review.user_id == uid)\
                                    .select(df_review.stars)\
                                    .map(lambda a:a[0])\
                                    .collect())
    return std_rank_user
```

RATING PREDICTION

Predict a rating, $p_{a,i}$, for each item i, for active user, a, by using the n selected neighbor users, $u \in \{1,2,\ldots n\}$.

Weight users' ratings contribution by their similarity to the active user.

$$p_{a,i} = \bar{r}_a + \frac{\sum_{u=1}^{n} w_{a,u}\left(r_{u,i} - \bar{r}_u\right)}{\sum_{u=1}^{n} w_{a,u}}$$

DATA SPARSITY PROBLEMS

Cold start problem

- How to recommend new items?
- What to recommend to new users?

Straightforward approaches

- Ask/force users to rate a set of items
- Use another method (e.g., content-based, demographic or simply non-personalized) in the initial phase

METRICS MEASURE ERROR RATE

Mean Absolute Error (*MAE*) computes the deviation between predicted ratings and actual ratings

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|p_i - r_i|$$

Root Mean Square Error (*RMSE*) is similar to *MAE*, but places more emphasis on larger deviation

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(p_i - r_i)^2}$$

DATABRICKS SPARK CLUSTER

Attention: S3 and Library issues (read more)

Clusters

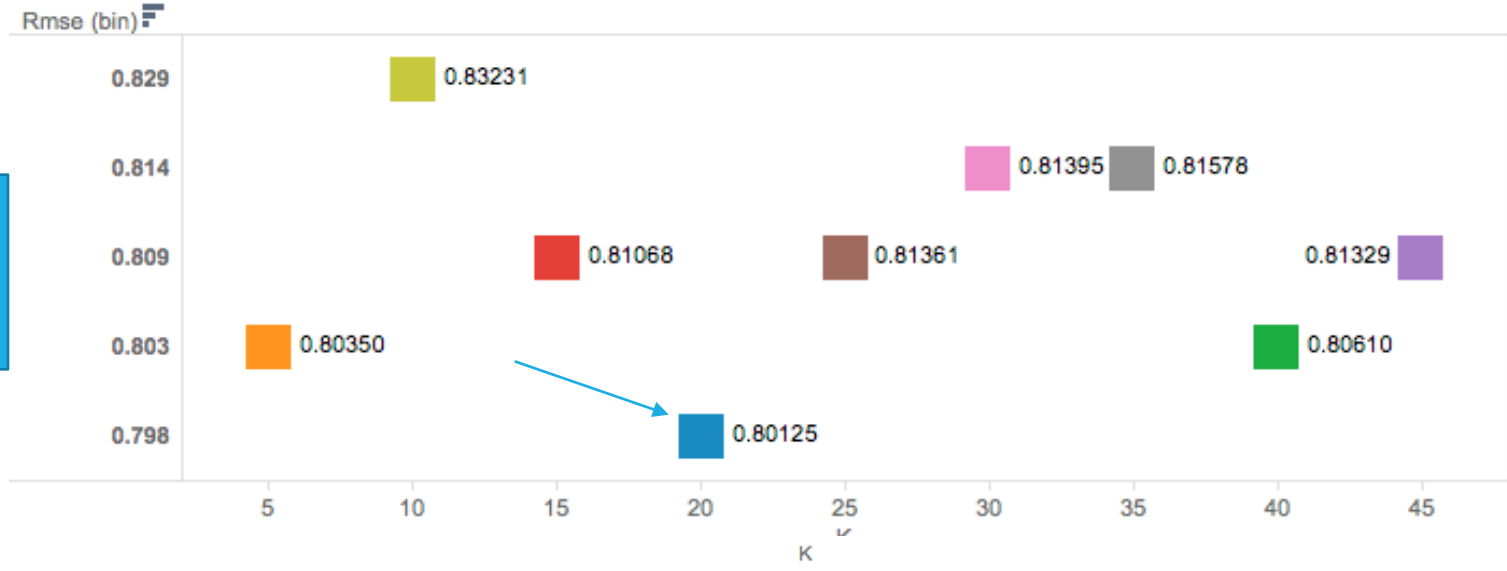
+ Create Cluster

Name	Memory	Type	State	Nodes
My Cluster	270 GB	Spot / Spark 1.5 ▾ Advanced Availability Zone: us-east-1b (default) Spark Config	Running	View Spark UI / Logs ▾ 9 Nodes Master Worker 0 Worker 1 Worker 2 Worker 3 Worker 4 Worker 5 Worker 6 Worker 7

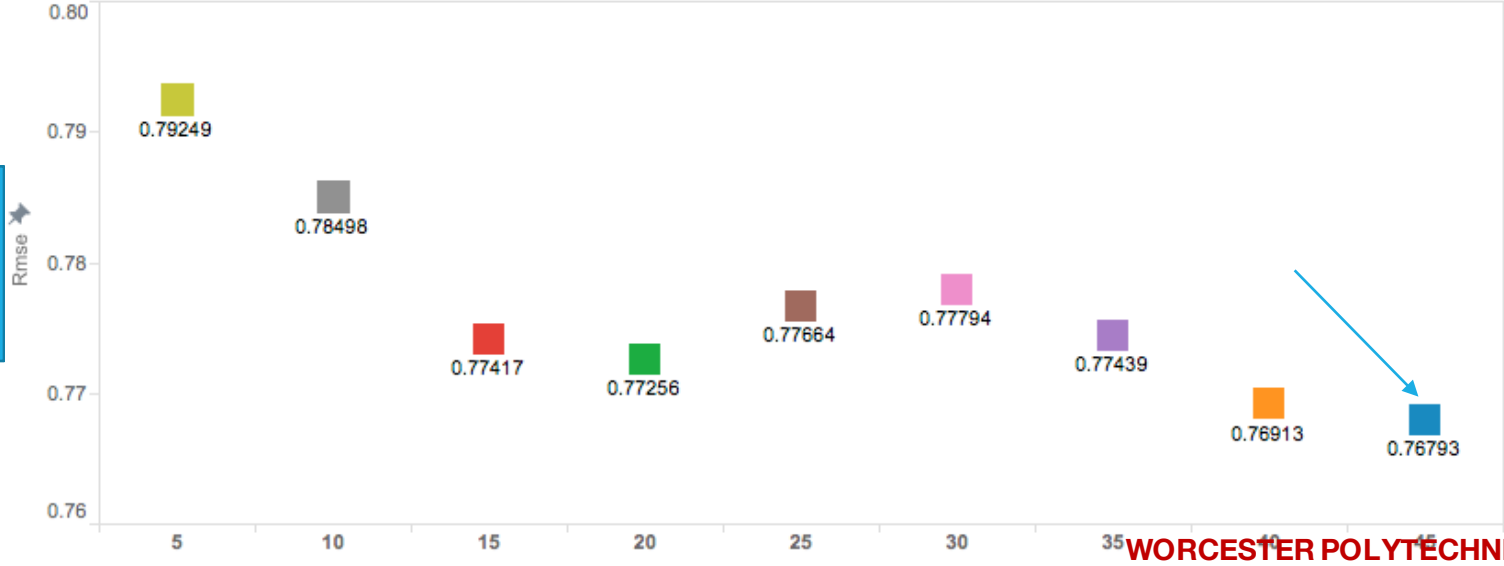
DATABRICS SPARK CLUSTER

COMPARE THE RESULTS

Review Based RMSE



Rank Based RMSE



RECOMMENDATION EXAMPLE

Using Rank-User based Collaborative Filtering
Top N Neighbors : 45

Be Recommended

User_ID = 'fczQCSmaWF78toLEmb0Zsw'
User_Name = 'Gabi'

Rank Prediction: 5.5

```
Name = "Rudy's Country Store And Bar-B-Q"
State = u'AZ'
City = u'Chandler'
Stars = 4.0
review_count = 391
```

CONCLUSIONS

I. Recommendation and personalization are important approaches to combating information over-load.

II. Machine Learning is an important part of systems for these tasks.

III. Collaborative filtering has problems

IV. Content-based methods (but have problems of their own)

V. Spark is powerful tool to do Big Data Analytics

QUESTIONS

Thanks!