Pivotal

使用HAWQ和MADlib开发推荐应用

Hong Wu: hwu@pivotal.io
Ruilong Huo: rhuo@pivotal.io

Meetup @Shanghai, 2016/04/28

Agenda

- RecSys Review
- Recommendation with HAWQ
- Demo
- Q&A

Agenda

- RecSys Review

Pivotal

What is RecSys

- Recommendation
- More than search engine
  - explore the world without query
- Personalization
  - discover items based on preference
- Self-evolving

Industry Trends

Pivotal

System Design

- Content-based recommendation
  - features from item itself
- Collaborative filtering
  - user-user similarity & item-item similarity
- Model-based recommendation
  - factor model

System Design

- Content-based recommendation
  - features from item itself
- Collaborative filtering
  - user-user similarity & item-item similarity
- Model-based recommendation
  - factor model

Agenda

- RecSys Review
- Recommendation with HAWQ

System Overview

Model Pipeline

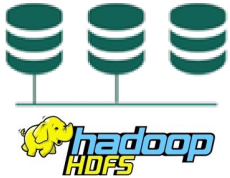
Preparation
Map original interactive data into user-item-rating tuples. Import into HAWQ Database.

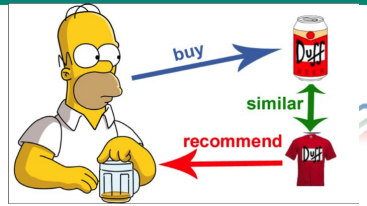
Latent Factor Model
Use **lmf_igd_run** function in MADLib to train user factors and item factors.

Similarity
Apply CF strategy to compute similarity.

Recommendation
Three ways to return(see later slides).

Import Training Data → **Matrix Factorization** → **Compute Item-Item Similarities** → **Make Recommendation**





Preparation

- User data
- Item data
- Interactive data

Preparation

- User data
- Item data
- Interactive data
  - {(user, item): preference weight}

```
1 1945809 15762 2005-03-20 2
2 2044720 15762 2003-12-30 5
3 337785 15762 2005-01-25 3
4 2099914 15762 2003-12-11 3
5 1788346 15762 2003-05-13 1
6 1873377 15762 2005-02-11 4
7 1386457 15762 2005-12-06 3
8 350560 15762 2004-04-26 2
9 1126245 15762 2005-06-21 4
10 2162676 15762 2005-08-25 1
11 562320 15762 2005-12-04 1
12 914958 15762 2003-10-23 4
13 657997 15762 2005-06-22 3
14 2195832 15762 2004-02-26 3
15 2590515 15762 2003-03-30 5
16 184813 15762 2004-10-25 4
17 829933 15762 2005-07-21 3
18 688306 15762 2003-09-23 3
19 2078484 15762 2004-10-27 3
20 1344624 15762 2005-10-04 1
```

History Matrix: Users by Items

Ted	✓	✓	✓
Carol	✓		✓
Bob			✓

	Troy (Action)	The Godfather (Crime)	Titanic (Romantic)	Forrest Gump (Comedy)
Fahime	5	⊘	5	1
Musi	5	⊘	⊘	1
Hamza	4	4	5	1
Paul	4	⊘	5	5
Adam	1	2	⊘	5

Latent Factor Model

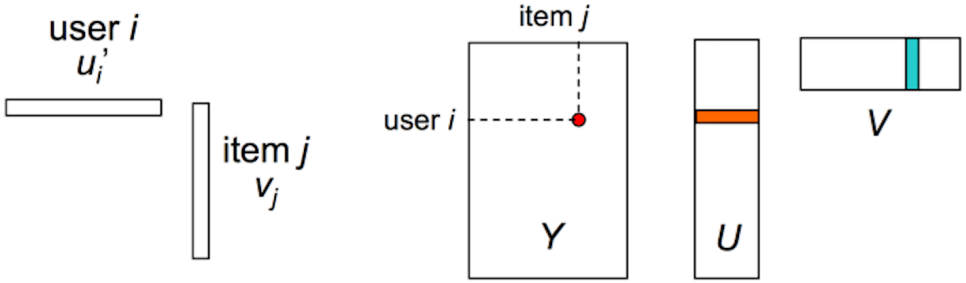
- Matrix factorization
  - model each user-item as a vector of factors

rating that user *i* gives item *j*

$K << M, N$
M = number of users
N = number of items

$$y_{ij} \sim \sum_k u_{ik} v_{jk} = u_i' v_j \quad \Leftrightarrow \quad \underset{M \times N}{Y} \sim \underset{M \times K}{U} \; \underset{K \times N}{V}$$

factor vector of user *i* — factor vector of item *j*



Latent Factor Model

Rating Matrix (N x M)

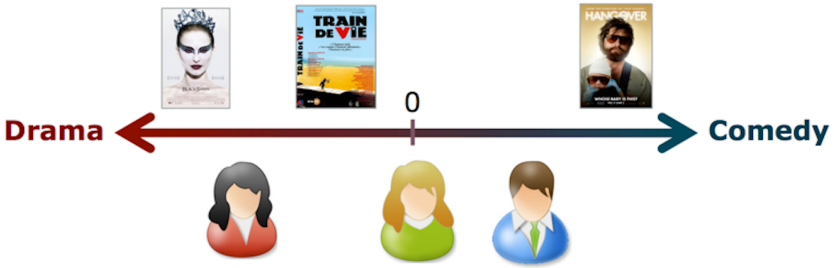
	5	3	5
	4	2	1
	0	3	3

User Feature Matrix (F x N)

f_1	1	-4	1
f_2	-2	0	-3
~~f_3~~	~~0~~	~~5~~	~~1~~

Movie Feature Matrix (F x M)

f_1	-1	0	-2
f_2	4	-4	1
~~f_3~~	~~0~~	~~2~~	~~2~~



Matrix Factorization in MADlib

```
SELECT madlib.lmf_igd_run(output,          — result table to store model
                          input,           — input table
                          row,             —  user attribute name
                          col,             — item attribute name
                          value,           — preference weight
                          row_dim,         — the number of users
                          col_dim,         — the number of items
                          max_rank,        — the number of factors
                          stepsize,        — learning rate
                          iterations,      — training rounds
                          tolerance);
```

Pivotal

Similarity

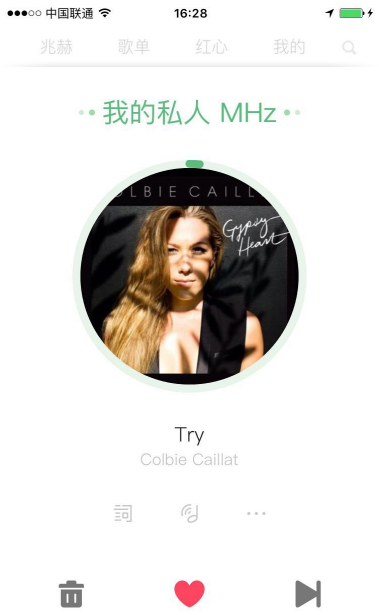
- We have user factors U and item factors V
- Apply collaborative filtering strategy
  - compute item-item similarities
  - less computation, large scale
  - good diversity vs original CF

Similarity

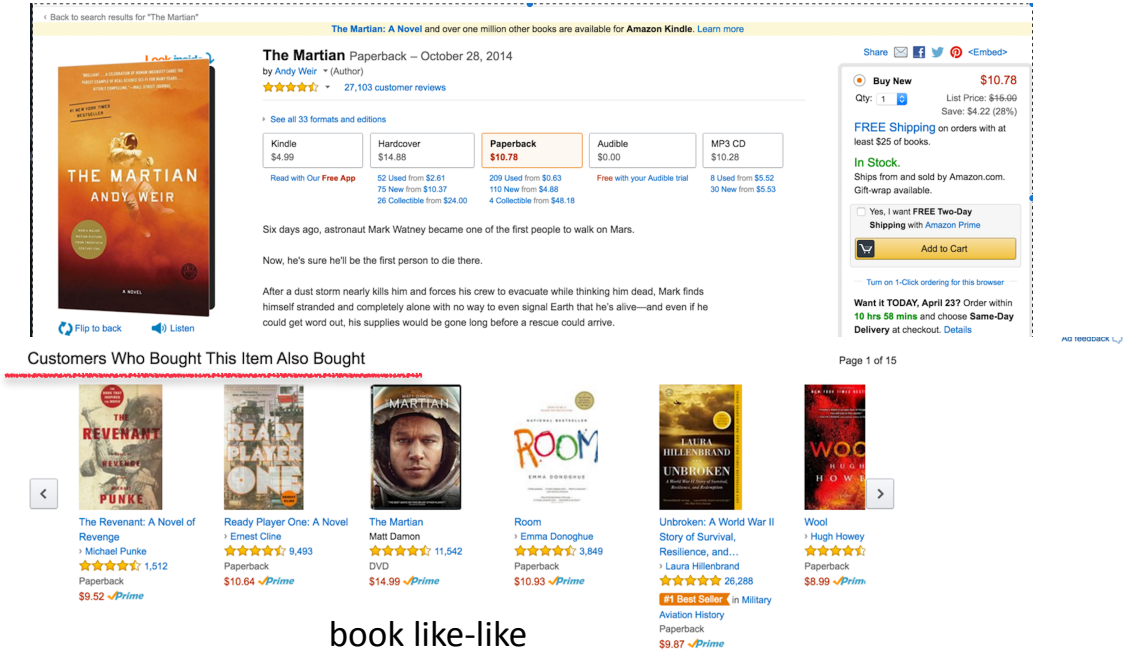
```
CREATE TABLE item_sim AS (
SELECT ifacA.iid,
       ifacB.iid,
       madlib.cosine_similarity(ifacA.rating,
ifacB.rating) as sim
FROM ifactor AS ifacA,
     ifactor AS ifacB
WHERE ifacA.iid < ifacB.iid
sim DESC LIMIT K
);
```

Recommendation

- Use item similarity directly



radio FM



book like-like

Pivotal

Recommendation

- Use user factor and item factor directly
  - dot product

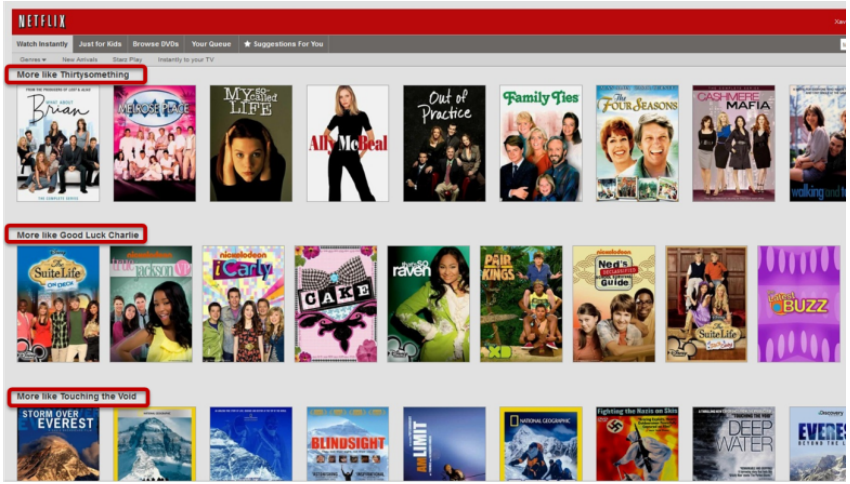
	Troy (Action)	The Godfather (Crime)	Titanic (Romantic)	Forrest Gump (Comedy)
Fahime	5	⊘	5	1
Musi	5	⊘	⊘	1
Hamza	4	4	5	1
Paul	4	⊘	5	5
Adam	1	2	⊘	5

Recommendation

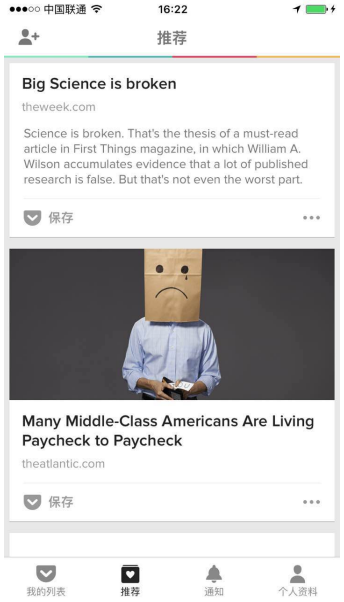
- Use item similarity to predict rating
  - weighted sum

	Troy (Action)	The Godfather (Crime)	Titanic (Romantic)	Forrest Gump (Comedy)
Fahime	5	⊘	5	1
Musi	5	⊘	⊘	1
Hamza	4	4	5	1
Paul	4	⊘	5	5
Adam	1	2	⊘	5

Playlist



Movie Guess



Feed Stream

Pivotal

Agenda

- RecSys Review
- Recommendation with HAWQ
- Demo

Try it Out

- https://github.com/xunzhang/xz_talk/tree/master/develop_recsys_with_hawq

Why Choose HAWQ to Build Machine Learning Application

- Simple & Efficient
  - high level abstraction, no transfer cost
  - focus more on application's own logic
  - dataflow thinking
- High scalability & performance
  - distributed machine learning with MADlib
  - parallel query engine with HAWQ

Summary

- Brief intro of RecSys
- A hybrid model pipeline based on HAWQ

Agenda

- RecSys Review
- Recommendation with HAWQ
- Demo
- Q&A

Thanks
Questions?

Pivotal®

Backup Slide

Solution of CF in large scale: AB_similarity.

Storage disaster of dot product based on factor model: Build Balltree index.