

# yelp<sub>Lp</sub> Recommendation

Zhejing Hu  
Yang Lyu  
Lan Jiang  
Chi Ma

# Motivation

- Many people use yelp today for convenience
- Yelp make recommendation based on location rather than your taste
- Serve as an alternative to yelp core functionality

# Project Outline

1

## Data Preprocessing

Remove irrelevant columns and filtering.

2

## Recommendation Engine

Building different recommendation algorithms and evaluate performance

3

## Ui Design

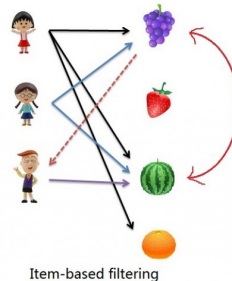
Front-end deployment and design



web development,  
one drop at a time



## SVD



$$\mathbf{M} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^*$$

## Back-end



## Database

## Recommendation



# MySQL database

```
5      ( SELECT user_id
6        FROM df_review2
7        GROUP BY user_id
8        HAVING COUNT(*) > 10
9      )
10 ORDER BY user_id
11 #import df_review
12
13 CREATE TABLE df_review(business_id varchar(50),cool int, date date, funny int, review_id varchar(50),stars int, text text, useful int, user_id varchar(50));
14
15
16 #import df_business
17
18 CREATE TABLE df_business(address varchar(50), attributes text, business_id varchar(50), categories text, city varchar(20), hours text, is_open int, latitude float, longitude float, name varchar(100),
19 neighborhood varchar(500), postal_code varchar(50),review_count int,stars float,state varchar(2), PRIMARY KEY(business_id));
20
21 #check drop null df_business("categories")
22
23 SELECT * FROM df_business WHERE categories IS NULL OR categories = "";
24
25 #Create new df_vegas
26 CREATE TABLE df_vegas LIKE df_business
27
28 INSERT INTO df_vegas(address, attributes, business_id, categories, city, hours, is_open, latitude, longitude, name, neighborhood, postal_code, review_count, stars, state)
29 SELECT address, attributes, business_id, categories, city, hours, is_open, latitude, longitude, name, neighborhood, postal_code, review_count, stars, state
30 FROM df_business
31 WHERE city = "Las Vegas"
32 AND categories LIKE "%Restaurants%"
33 AND review_count > 10;
34
35 #Create table df_left like df_vegas with features: business_id,name,stars,address,longitude,latitude
36
37 CREATE TABLE df_left(business_id varchar(50), name varchar(100),avg_stars float,address varchar(50),longitude float, latitude float, PRIMARY KEY(business_id));
38
39 # fill values
40
41 INSERT INTO df_left(address,business_id,latitude,longitude,name,avg_stars)
42 SELECT address,business_id,latitude,longitude,name,stars
43 FROM df_vegas;
44
45 #Create Table df_right like df_review with features: business_id, review_id, stars, text, user_id
46
47 CREATE TABLE df_right(business_id varchar(50), review_id varchar(50), stars float,text text,user_id varchar(50));
48
49 #fill values
50
51 INSERT INTO df_right(business_id,review_id,stars,text,user_id)
52 SELECT business_id,review_id,stars,text,user_id
53 FROM df_review;
54
```

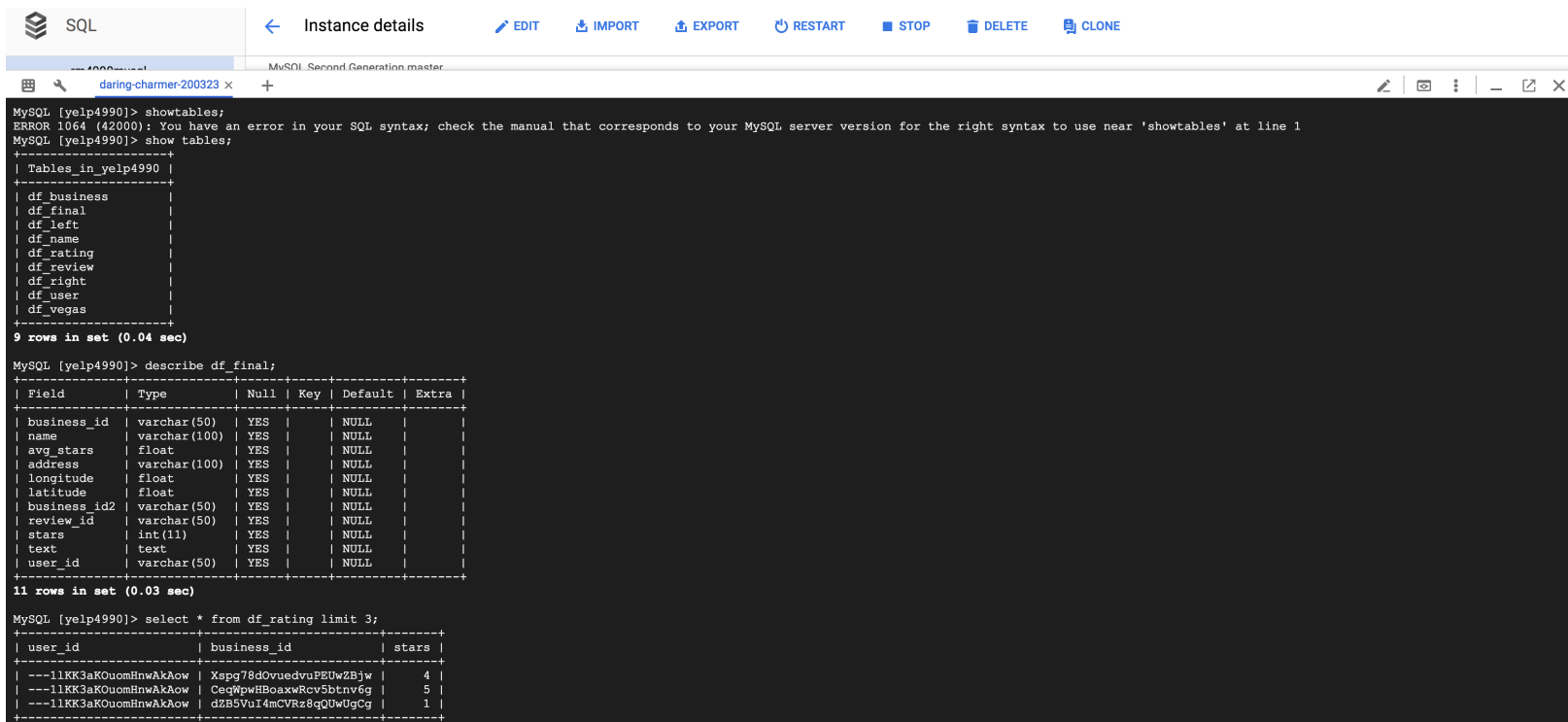
Type to enter a caption.

# MySQL database

```
49
50 INSERT INTO df_right(business_id,review_id,stars,text,user_id)
51 SELECT business_id,review_id,stars,text,user_id
52 FROM df_review;
53
54
55 #CREATE df_final inner join df_left & df_right;
56 CREATE TABLE df_final(business_id varchar(50), name varchar(100), avg_stars float, address varchar(100), longitude float, latitude float, business_id2 varchar(50),review_id varchar(50),stars int,text
57 text,user_id varchar(50));
58
59 INSERT INTO df_final(business_id, name, avg_stars, address, longitude, latitude, business_id2, review_id, stars, text, user_id)
60 SELECT *
61 FROM df_left
62 INNER JOIN df_right
63 On df_left.business_id = df_right.business_id2;
64
65 #CREATE df_rating
66 CREATE TABLE df_rating(user_id varchar(50),business_id varchar(50),stars float);
67
68 INSERT INTO df_rating(user_id, business_id, stars)
69 SELECT user_id, business_id, stars
70 FROM df_final
71 WHERE user_id IN
72 (
73     SELECT user_id
74     FROM df_final
75     GROUP BY user_id
76     HAVING COUNT(*) > 10
77 )
78 And business_id IN
79 (
80     SELECT business_id
81     FROM df_final
82     GROUP BY business_id
83     HAVING COUNT(*) > 15
84 )
85 ORDER BY user_id
86
87 #CREATE df_name
88 CREATE TABLE df_name(business_id varchar(50),name varchar(100));
89
90 INSERT INTO df_name(business_id, name)
91 SELECT business_id, name
92 FROM df_final
93 WHERE user_id IN
94 (
95     SELECT user_id
96     FROM df_final
97     GROUP BY user_id
98     HAVING COUNT(*) > 10
99 )
100 And business_id IN
101 (
102     SELECT business_id
103     FROM df_final
104     GROUP BY business_id
105     HAVING COUNT(*) > 15
106 )
107 ORDER BY business_id
```

Type to enter a caption.

# MySQL database



The screenshot shows a MySQL database interface with a terminal window. The terminal displays the following SQL commands and their results:

```
MySQL [yelp4990]> showtables;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'showtables' at line 1
MySQL [yelp4990]> show tables;
+-----+
| Tables_in_yelp4990 |
+-----+
| df_business         |
| df_final            |
| df_left             |
| df_name             |
| df_rating           |
| df_review           |
| df_right            |
| df_user             |
| df_vegas            |
+-----+
9 rows in set (0.04 sec)

MySQL [yelp4990]> describe df_final;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| business_id | varchar(50)   | YES  |     | NULL    |       |
| name        | varchar(100)  | YES  |     | NULL    |       |
| avg_stars   | float         | YES  |     | NULL    |       |
| address     | varchar(100)  | YES  |     | NULL    |       |
| longitude    | float         | YES  |     | NULL    |       |
| latitude    | float         | YES  |     | NULL    |       |
| business_id2 | varchar(50)   | YES  |     | NULL    |       |
| review_id   | varchar(50)   | YES  |     | NULL    |       |
| stars       | int(11)       | YES  |     | NULL    |       |
| text        | text          | YES  |     | NULL    |       |
| user_id     | varchar(50)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
11 rows in set (0.03 sec)

MySQL [yelp4990]> select * from df_rating limit 3;
+-----+-----+-----+
| user_id | business_id | stars |
+-----+-----+-----+
| ---11KK3aK0uomHnwAkAow | Xspg78dOvuedvuPEUw2Bjw | 4 |
| ---11KK3aK0uomHnwAkAow | CeqWpwHBoaxwRcv5btnv6g | 5 |
| ---11KK3aK0uomHnwAkAow | dZB5VuI4mCVRz8qQUwUgCg | 1 |
+-----+-----+-----+
```

Type to enter a caption.

Demo