Taobao Recommender System

Loh Si Jun Shauna | General Assembly DSI 11

Executive Summary

Problem Statement

The Dataset

Exploratory Data Analysis

Recommender System

Model evaluation and conclusion

Telegram Bot on Herokuapp

Problem Statement

"Consumers worldwide will spend nearly **$3.46 trillion** online in 2019, up from **$2.93 trillion** in 2018." *(Digital Commerce 360, Nov 2019)*

"Recommender systems help the users to get **personalized recommendations**, helps users to take **correct decisions** in their online transactions, **increase sales** and redefine the users web browsing experience, **retain** the customers, enhance their **shopping experience**." *(IEEE, Feb 2017)*

"Users **love** it when companies can **second-guess** their thoughts. Recommender systems, in particular, are the **new favorite** among **eCommerce** companies." *(Business2Community, Feb 2019)*

The Dataset

Alibaba Cloud

Taobao User Behavior Data

User_id
4831 users

Item_id
234,096 items

Category_id
4895 categories

Behavior_type
483,829 user behavioral actions

- **Page View**
- **Add to Cart**
- **Favorite**
- **Buy**

Timestamp
25 Nov'17 – 3 Dec'17

https://tianchi.aliyun.com/dataset/dataDetail?dataId=649

Data Cleaning

Change datetime format

```
pd.to_datetime(tb["timestamp"])

tb_new['timestamp'].dt.strftime('%d/%m/%Y')
```

Drop duplicates

```
tb.drop_duplicates()
```

Remove invalid dates

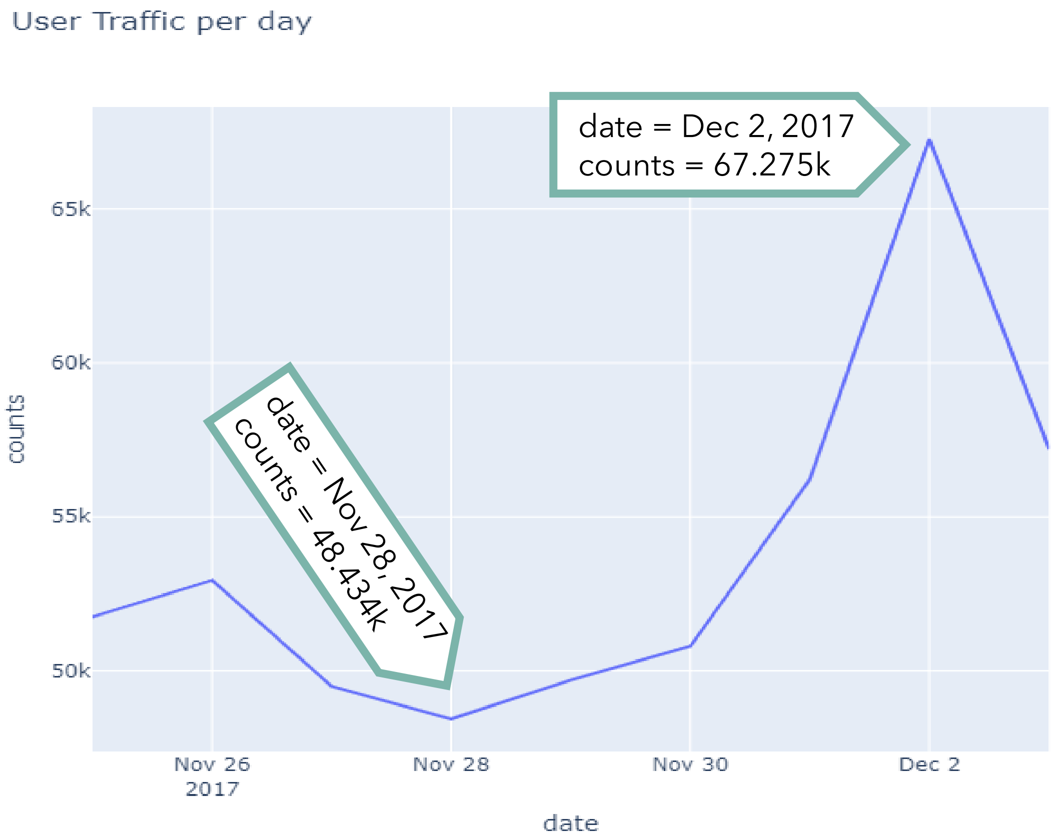
user_id	item_id	category_id	behavior_type	timestamp	date
170980	1500112	4145813	pv	1970-01-01 12:13:36	01/01/1970
419355	418492	5053508	pv	1970-01-01 03:08:57	01/01/1970

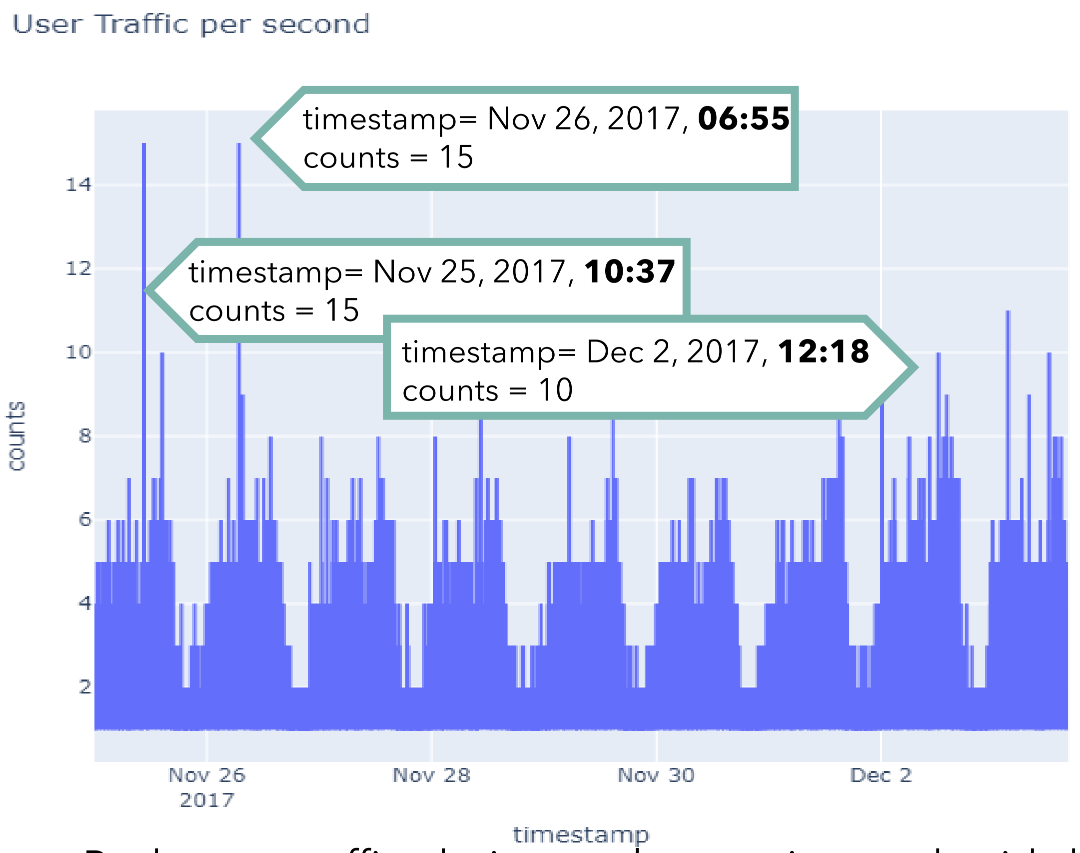
Scaling down dataset (by User ID)

```
tb = tb[tb['user_id']<5000]
```

Exploratory Data Analysis

Taobao could launch marketing campaigns or flash sales during peak user traffic as observed by user traffic trends below:

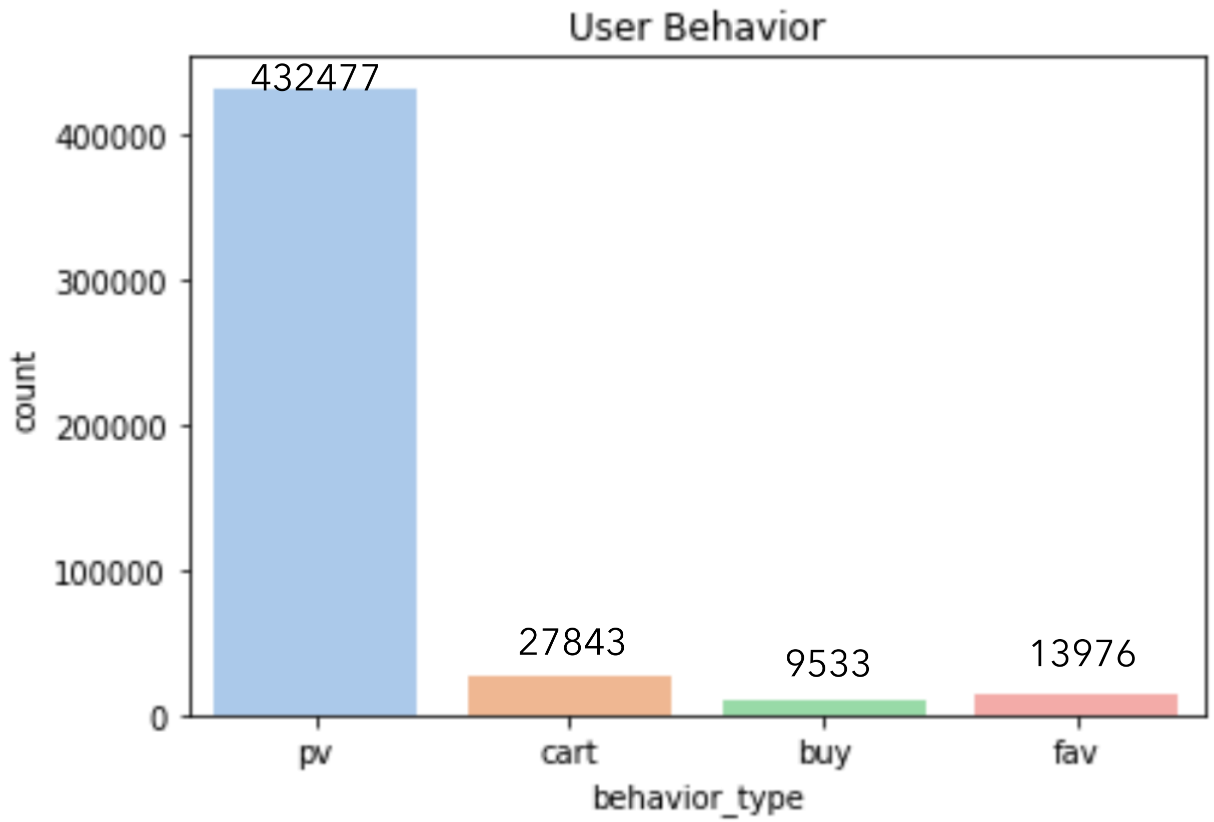




- Peak user traffic on Saturday – Users tend to have more free time on weekends for online shopping.
- Peak user traffic during early morning and mid-day – Users browse their shopping app either on the way to work or during lunch break.

Exploratory Data Analysis

Myth Buster: Most consumers on Taobao were 'window shopping':



1.97% of user activity = 'buy'

Out of 483,829 user behavior counts, 9533 counts were actual purchases.

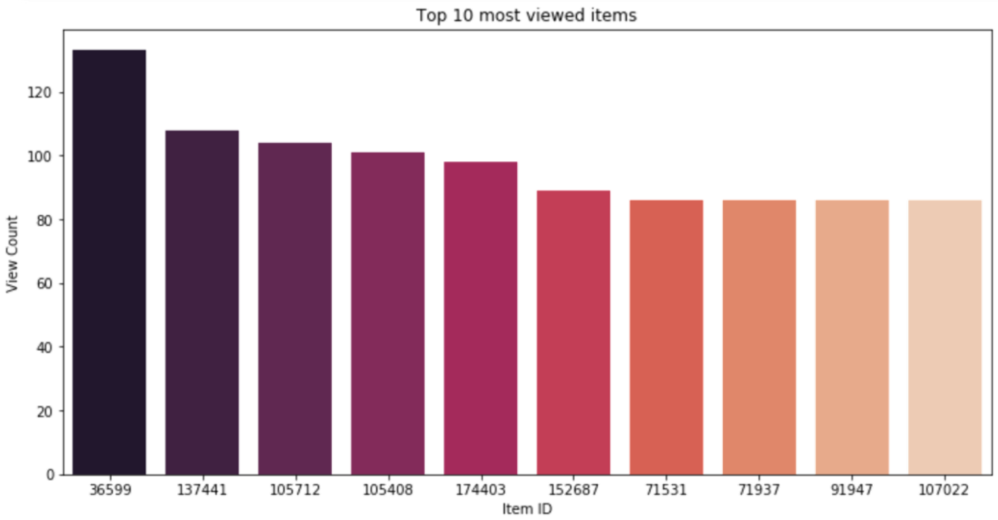
BUT

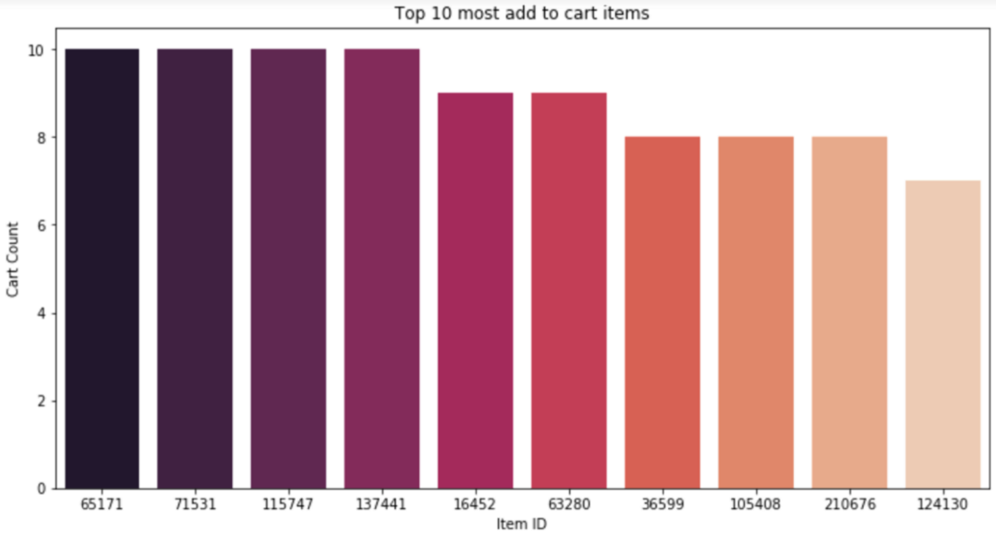
66.7% of users made purchase(s)

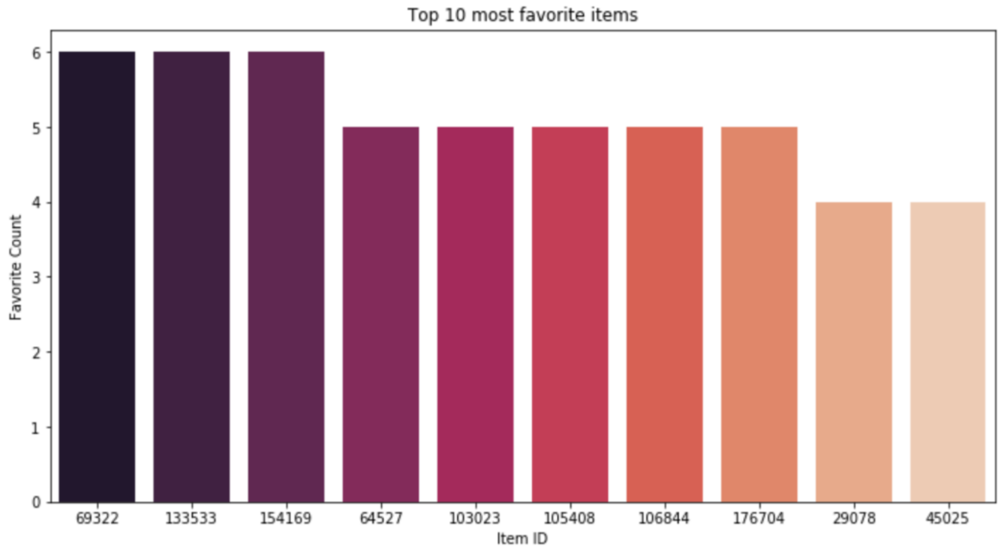
Out of 4831 users, 3224 users actually made at least one purchase from Taobao.

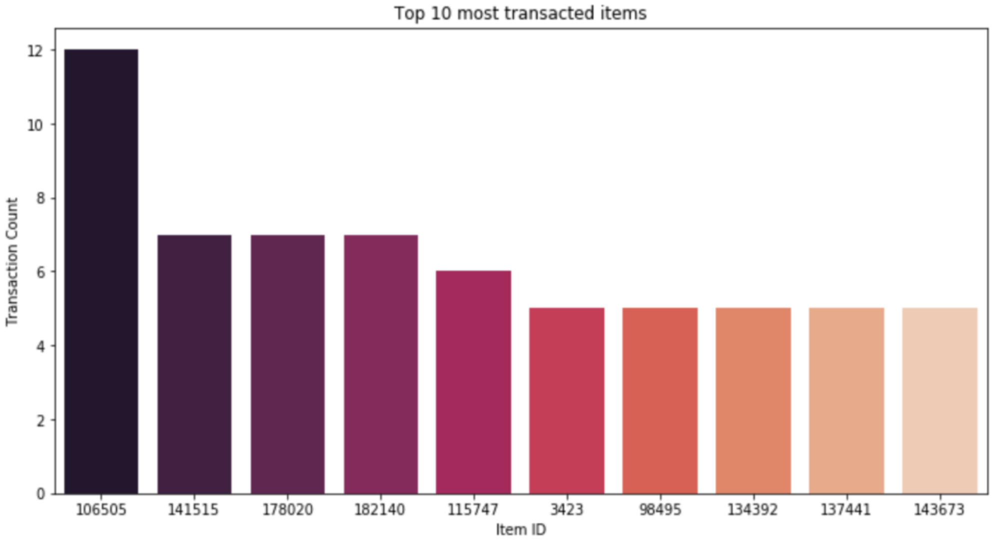
Exploratory Data Analysis

Top 10 most interacted items: *Popular purchases display different trends from popular viewed/ add to cart/favorite items. User interests may not lead to an actual purchase.*



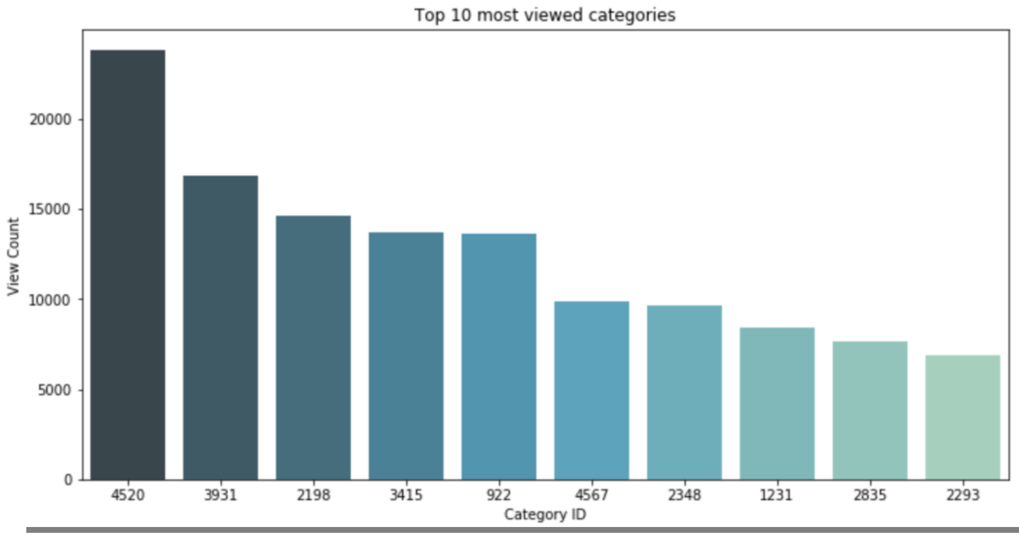


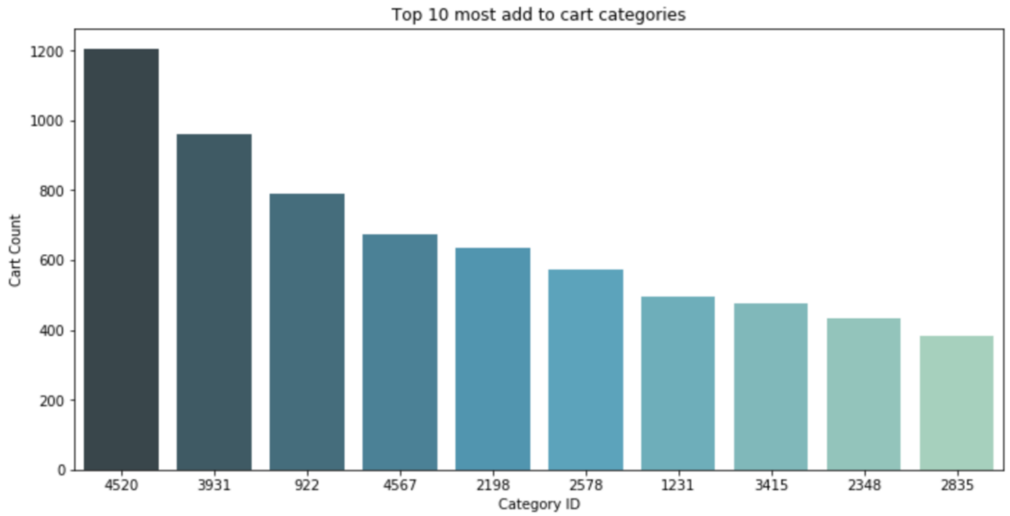


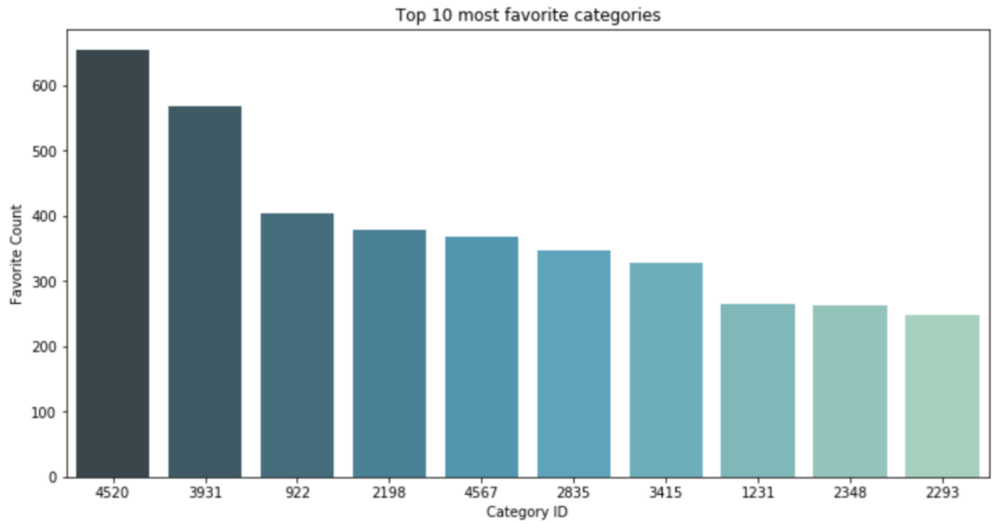


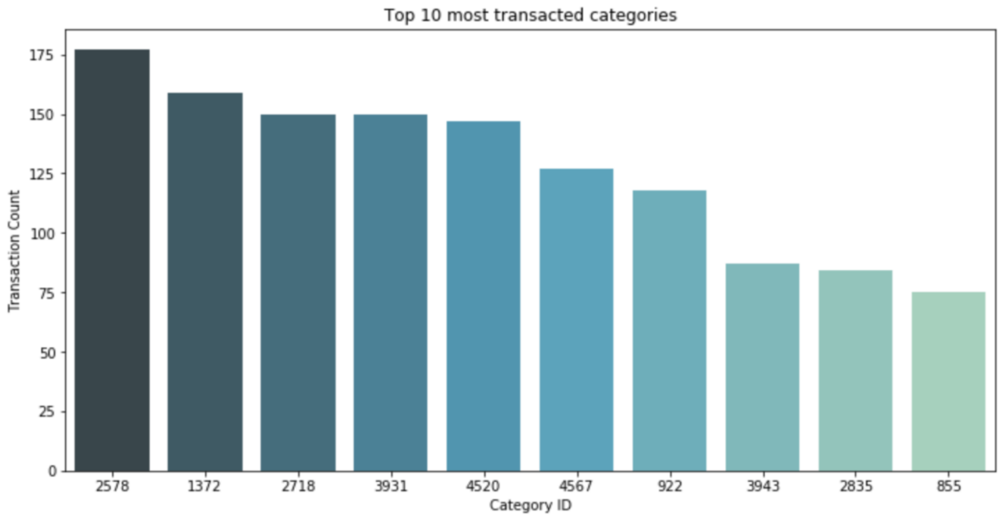
Exploratory Data Analysis

***Top 10 most interacted categories**: Popular purchases display different trends from popular viewed/add to cart/favorite categories. User interests may not lead to an actual purchase.*



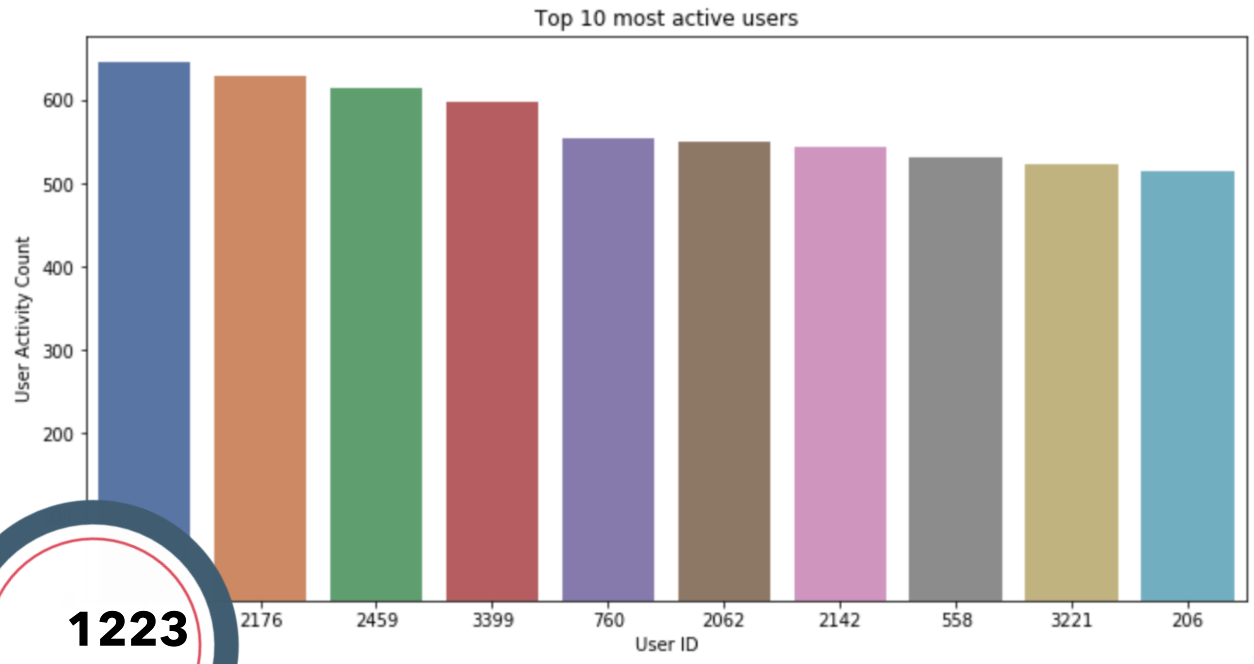


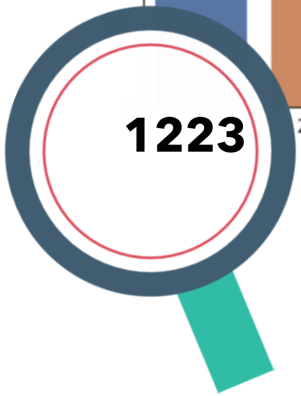




Exploratory Data Analysis

Are Taobao platinum users subscribing to the most highly interacted items?





The most active user does not subscribe to the top 10 most viewed/add to cart/favorite/bought items on Taobao.

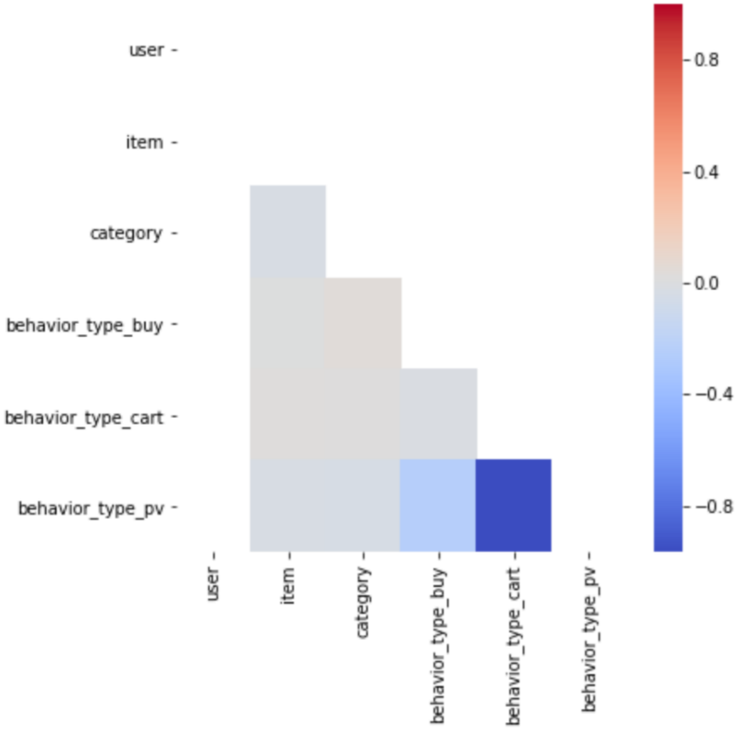
Most active user 1223:

Item ID	Item Value Counts	Included in Top 10 most interacted item list?
228348	9	×
88980	8	×
41567	8	×
192412	7	×
119765	7	×
35216	6	×
127345	5	×
3621	5	×
43124	5	×
67120	5	×

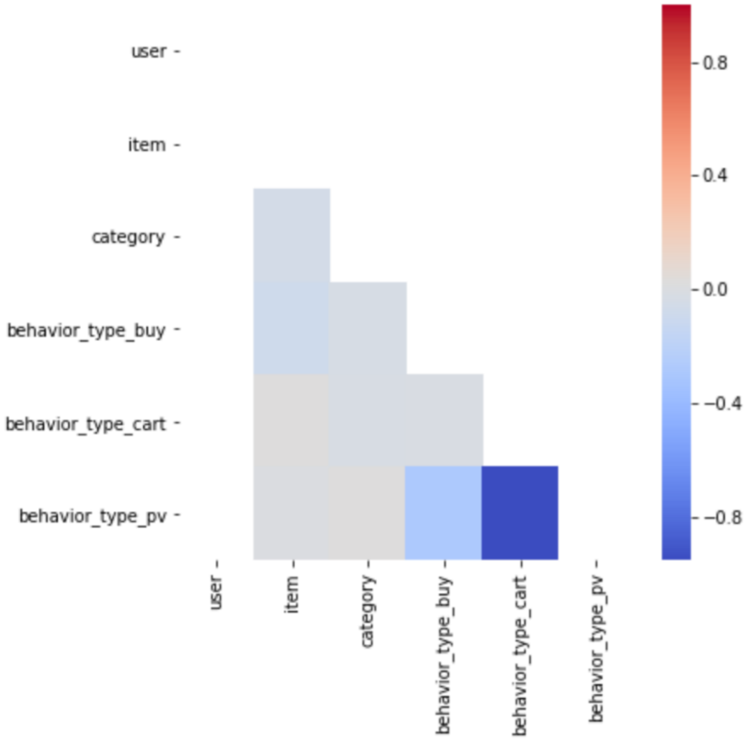
Exploratory Data Analysis

Heatmaps of Top 3 most active users:

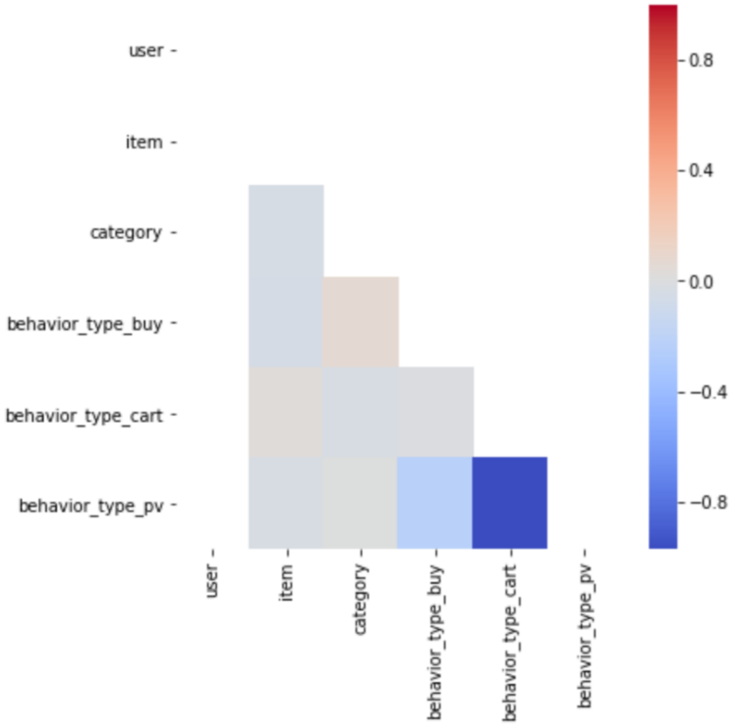
User ID: 1223



User ID: 2176



User ID: 2459



- Dummify feature: 'behavior_type'
- There is significant correlation between page view and cart; weak correlation between cart and buy.
- Most users are conservative buyers and may hesitate to purchase the items (by adding to cart first), and later decide not to purchase the items due to possible factors such as cost.

Recommender System

Implicit Collaborative Filtering Recommendation Engine

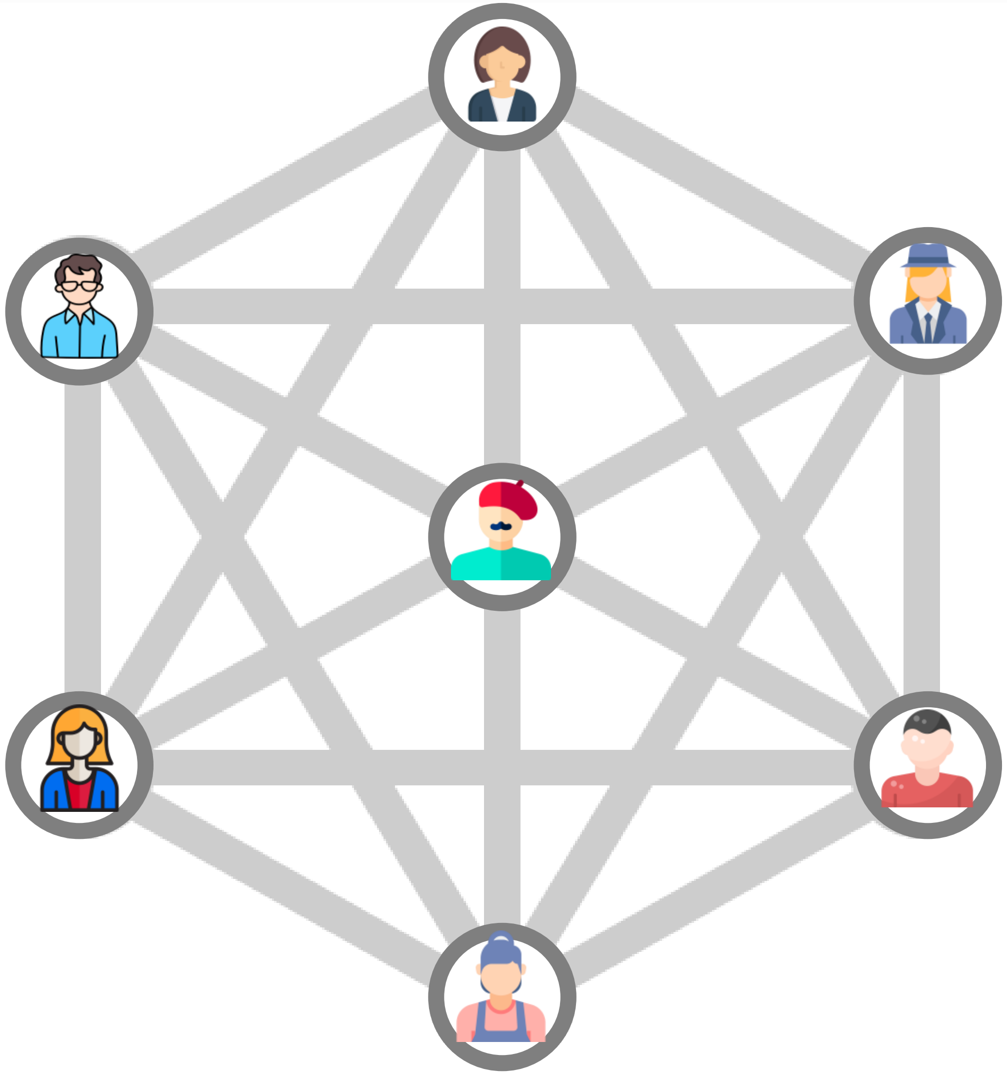
What is implicit data?

"Implicit data (the type of data we're using here) is data we gather from the users behavior, with no ratings or specific actions needed."
(Medium, Aug 2017)

Why implicit collaborative filtering?

"Explicit feedback is hard to collect as they require additional input from the users. The users give explicit feedback only when they choose to do so."
(towardsdatascience, Apr 2019)

"Implicit data gives us the ability to find connections between users who have no specific items in common but share common tastes." *(Medium, Aug 2017)*



Recommender System

Implicit Collaborative Filtering Recommendation Engine

How do we create a recommendation system based on implicit data?

- Matrix factorization with implicit feedback - user behavior data.
- User Vector: express each user as a vector of their taste preferences.
- Item Vector: express each item as a vector of what tastes they represent.
- Optimize **U** and fix **V**, vice versa. This is done iteratively to arrive closer to ***R* = *U* x *V***.

Illustration of User-Item Interaction:

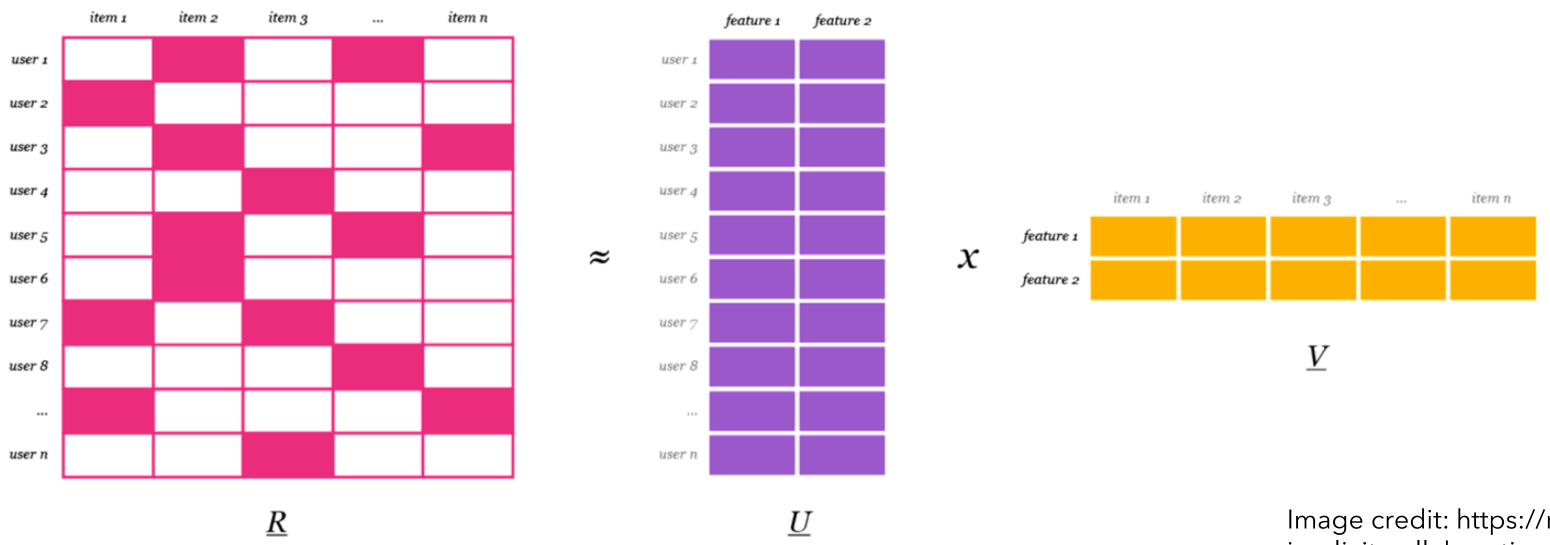


Image credit: https://medium.com/radon-dev/als-implicit-collaborative-filtering-5ed653ba39fe

Recommender System

Implicit Collaborative Filtering Recommendation Engine

Alternating Least Squares (ALS) Method

- ALS Model for each user behavior type: 'Page View', 'Add to Cart', 'Favorite' and 'Buy'.
- Model Fitting – Sparse Item by User Matrix.
- Recommendations – Spare User by Item Matrix.
- Recommendation score : dot product between the target user vector and transpose of the item vectors.

Baseline Model Recommender

- Item popularity – based on interaction counts.
- Each user behavioral action contributes to an interaction count.

Top picks for existing user:

	Item ID	Category ID	Score
0	60279	1840	0.829447
1	6949	3083	0.810875
2	206048	1252	0.780378
3	100288	1662	0.767859
4	135298	4007	0.756388
5	129876	3715	0.756388
6	178020	3415	0.750500
7	213295	4088	0.734380
8	125207	45	0.729198
9	208636	4298	0.719375

'Buy' Recommendations for User 1

Baseline Recommender for new users

Top picks for new user:

	Item ID	Category ID	Counts
0	141515	1427	7
1	3423	1791	5
2	137441	2578	5
3	142407	2326	4
4	115747	2578	4
5	208890	2612	4
6	133969	4678	4
7	47310	4557	4
8	134392	1178	4
9	120665	4359	4

Model Evaluation

Area Under the Curve (AUC) Values:

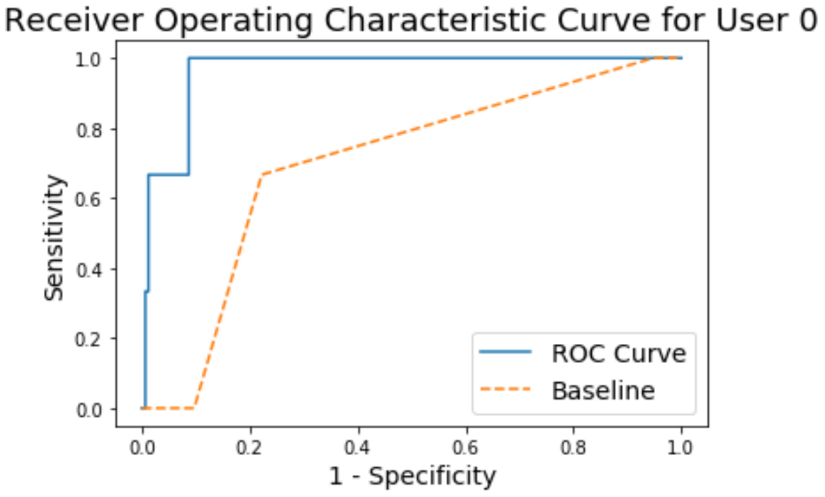
Mean AUC		
Behavior Type	**Recommender (ALS)**	**Baseline Recommender**
Page View	0.928	0.666
Add to Cart	0.986	0.951
Favorite	0.993	0.973
Buy	0.996	0.983

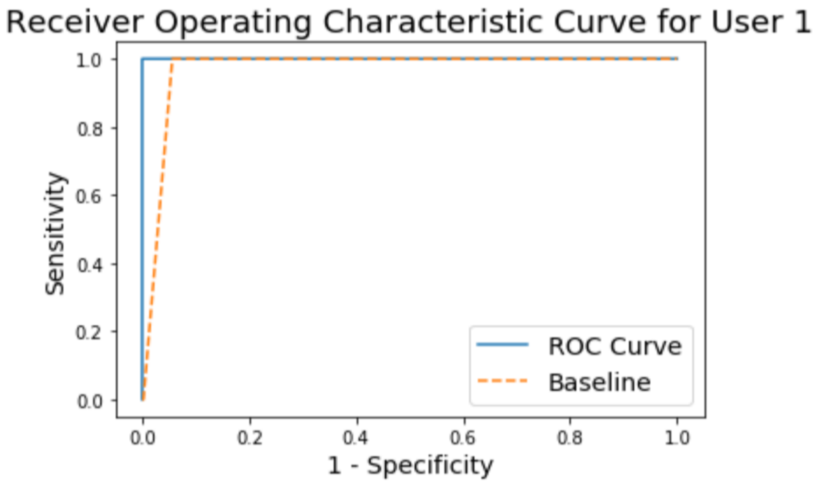
- Higher the AUC, the better model distinguishes its classes, hence the better the model is at it's recommendations.
- ALS Recommender has consistently higher AUC than the baseline recommender.

Model Evaluation

Plotting some examples of Receiver Operating Characteristic (ROC) curves:

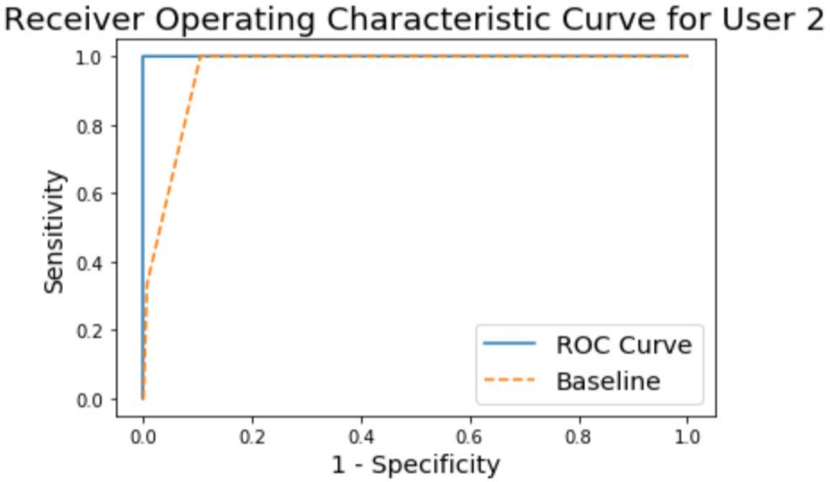
Page View

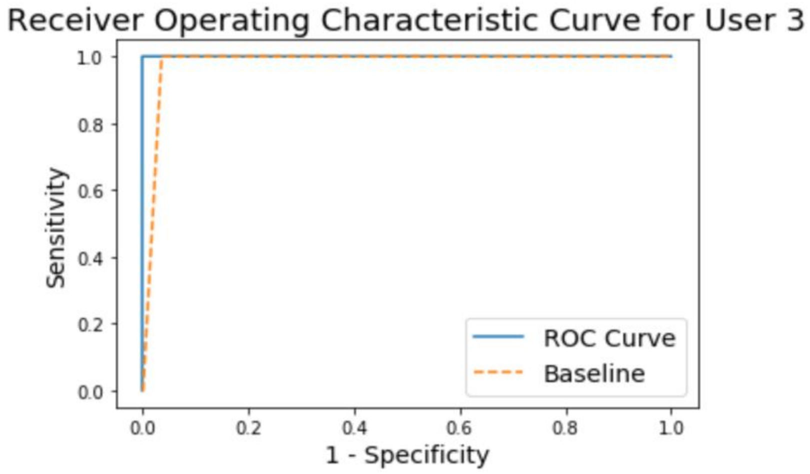




Add to Cart

Favorite





Buy

Conclusion

Based on a total of **4831 users** producing **483,829 user behavioral actions**, an **implicit recommendation** system was implemented to generate the top 10 most recommended to view/add to cart/favorite/buy items for each user, based on item and user similarity, as well as the interaction scores each user has with the items interacted. The **mean AUC score of each ALS model** was **higher** than the **mean AUC score of the baseline model**, thus suggesting the ALS model has positive and negative tastes perfectly separated and hence is more effective in its recommendations.

Conclusion

- **Relevant information for future implicit recommender systems:**
  - Price of each item
  - Name of each item and category
  - User behavior on special dates (e.g. 11.11, Black Friday, Valentine's Day)
  - Explicit data (i.e. user review and rating)

- **Recommender Technique: Dithering**
  - Re-order the recommendations list by adding random noise to the original recommendations.
  - Surface items further down the list to the top, to provide freshness to the recommendation list for subsequent user visits.

Telegram Bot on Herokuapp

Deployment:

Powered by:

Search ID:

@mytaobao_heroku

Telegram Bot on Herokuapp

User Guide:

Commands:

'/start': to start the bot.
'/register': registration is required for new users.
'/help': to provide assistance to the user in using the bot.
'/links': photos and URLs to the top 10 most popular items on Taobao.
'/exit': to leave the bot chat.

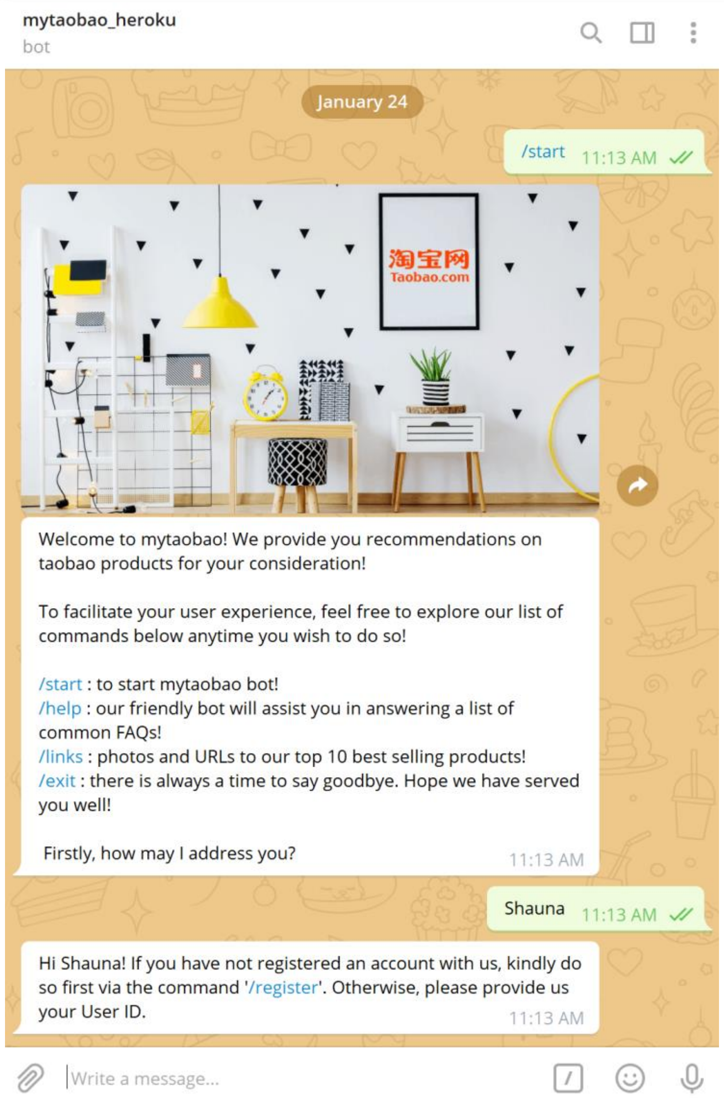
Recommendations:

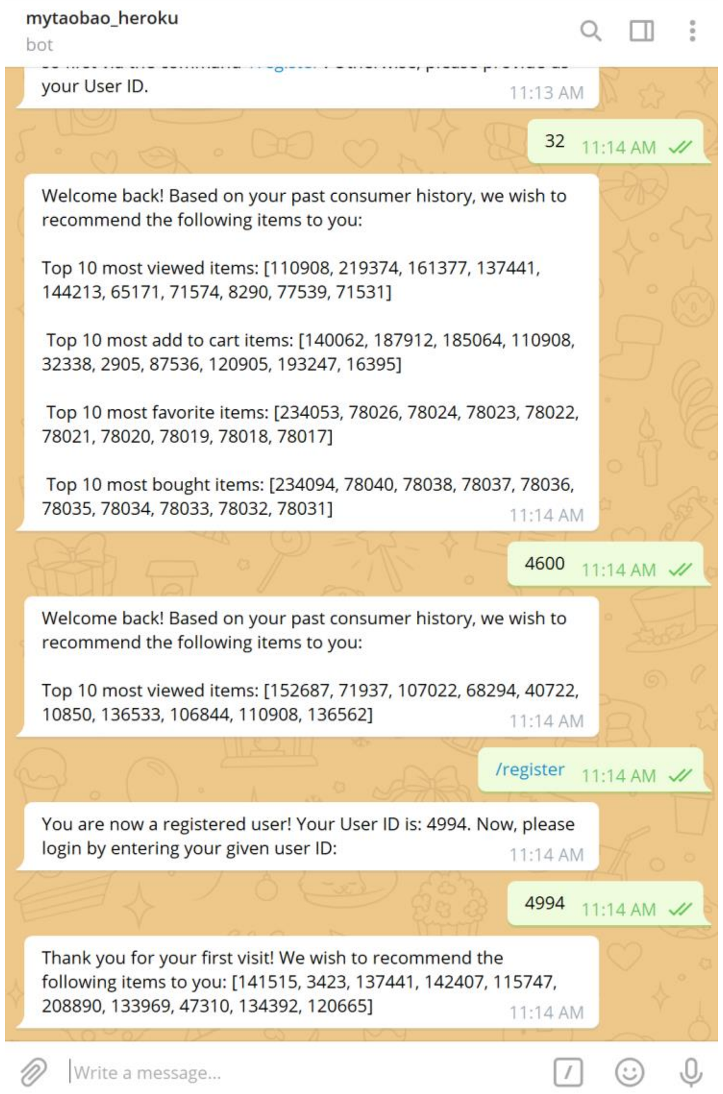
Existing User: recommended the top 10 items to view/add to cart/favorite/buy based on their user preferences history.

New User: recommended the top 10 items based on popularity.

Telegram Bot on Herokuapp

Demonstration: /start command and user recommendations





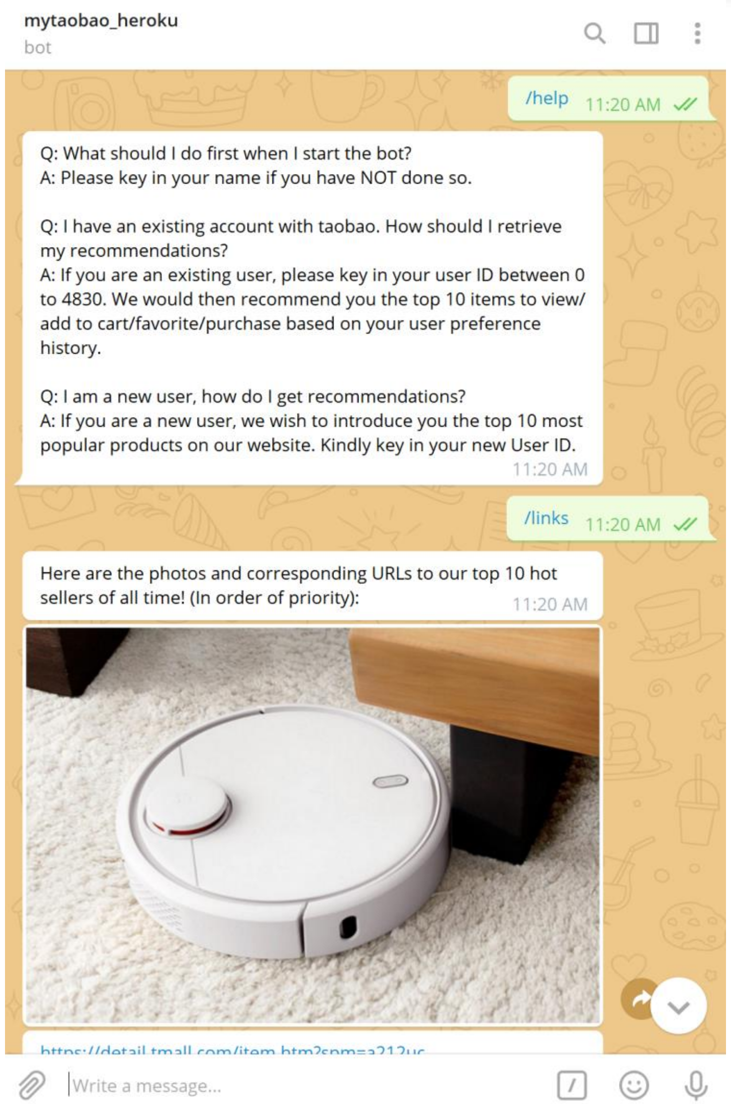
Existing User:
Completed all 4 user behavioral actions
(Recommended items on all behavior types)

Existing User:
Only viewed pages
(Recommended items on page view)

New User:
No user history
(Recommended popular items)

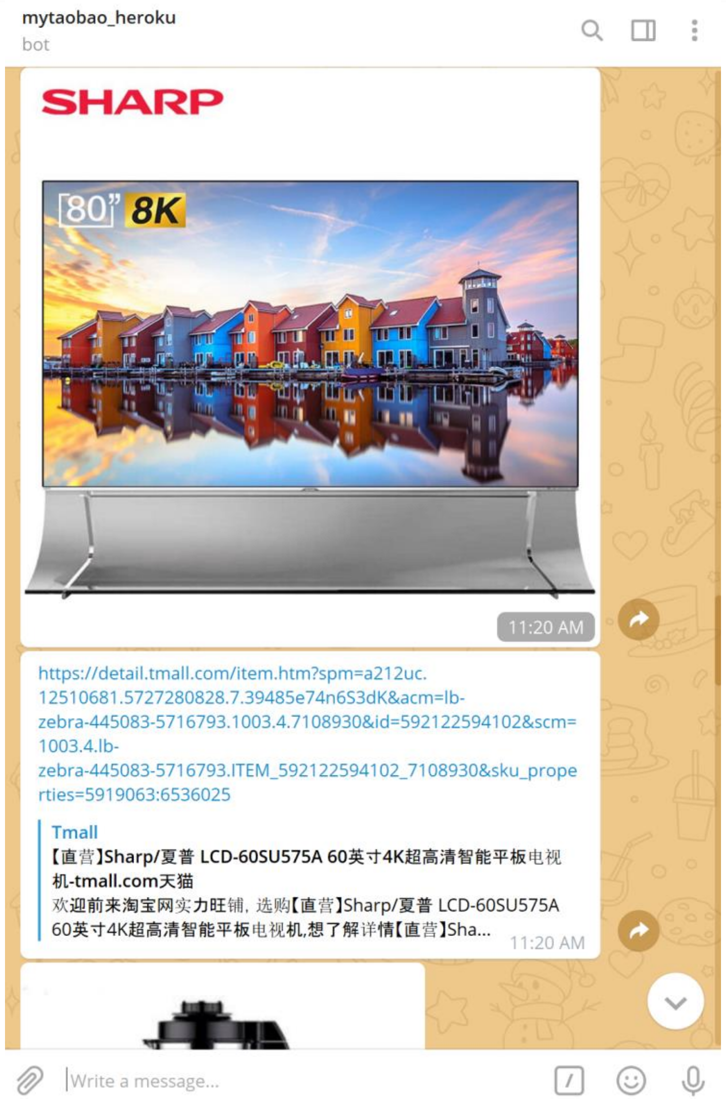
Telegram Bot on Herokuapp

Demonstration: /help and /links Commands



Telegram Bot on Herokuapp

Demonstration: /links and /exit Commands



Thank You!

Questions?