Smart Outfit Finder

Team ‘Untitled’

Xiaoyi Wang, Xinyuan Shi, Jianfeng Yin, Xinyue Li

**Introduction**

As online shopping becomes part of us everyday life, an increasing importance has been attached to electronic commerce. Especially in the fashion industry, all the retailers try to invest more on the online marketing and shopping experience. However, finding a suitable cloth is always hard by only exploring the lookbook and reviews. As a result, we decided to build a smart outfit finder that could help people find items they may like and the possible right size.

We tried to Improving online shopping experiences for customers and reducing cost on advertising and return for retailers.

**Data Cleaning and Exploratory Analysis**

Our recommending system is developed using the transaction data from ModCloth[1], which is a comprehensive shopping website for women’s clothing. The data[2] comes from a researcher's web-page, which had been collected for recommender system development.

The dataset contains 82,790 unique transactions. There are three types of variables: purchased item information, customers’ body measurements, purchase feedback. After reviewing the data type and missing percentage of each column, we decided to do cleaning steps including:

* Adding underscores where white space exists in column names.
* Dropping columns that have too many missing rows.
* Dropping null values in columns required for later analysis (quality, review text).
* Convert some variable types to numeric.
* Handling outliers.
* Using median or grouped median to do data imputation.

After cleaning the data, we conducted some exploratory analysis. We got a rough distribution of the customers. For example, the average customer of ModCloth is: 65 inches tall, wears US size 12, wears US size 8 shoes, rates the item quality 3.95/5. Apart from that, the number of purchased items varies in category. Tops, bottoms, and dresses are the main purchased categories. And the customers’ feedback about whether the item fits is imbalanced: the number of “fit” much larger. Also, we visualized each item’s purchase count and each user’s purchase count. They both have long tails, so we might need to filter out low-purchase items in later analysis. The plot also highlights that the customer base has a small number of repeat buyers given the median transaction per user is 1. This might be something to review with a business audience in an attempt to develop strategies to increase repeat customers, such as providing subscriptions on the website.

**Text Mining**

According to the review text in the dataset, we implemented the word frequency analysis and sentiment analysis that are groupby the fit conditions (fit, small, large). For frequency analysis, we produced the word clouds and tried to find out the different among them. However, most of them are very similar and only a few descriptive word like, waist, which may indicate the reason of fitness.

For sentiment analysis, we encounter the challenge of that the review text are relatively short and distributed unevenly. The vador score of three classes are all around 0.38. Most of the records are shown as fit, then it is hard to learn the cases that are unfit. This will lead to poor prediction force in small and large classes. Moreover, customers’ review text somehow relates to the personal and subjective characteristics of style rather than size match only. As a result, we use the vador score as an indicator of item recommendation rather than the size recommendation.

**Recommendation**

Based on the customers’ purchase history and their comments, we can recommend a customer with the fitting size for each item and the items they may be interested in.

For different recommendation system we use content based filtering and collaborative filtering. The content based filtering matches users to the content or items they liked or bought and the attributes of the users and the items are important. The collaborative filtering is based on the assumption that if A and B buy similar items, A is more likely to buy an item that B has bought than a random item. There is no features corresponding to user or item in collaborative filtering.

-- Size Recommendation

* Collaborative Filtering

collaborative filtering models based on the assumption that people like things similar to other things they like, and things that are liked by people with similar taste. In this method, we group the data by item and size and find similar users based on their ratings on the specific item and specific size. Then, we build a pivot table with user\_id as row, item\_id and size as column and quality as value. Next, We build the correlation matrix between each item and each size based on similar users so that when we input an item\_id and size and the recommendation system will recommend the items and sizes with highest correlation to users.

-- Item Recommendation

* Content Based Filtering

Content based filtering makes recommendations based on user preferences for product features and in this project, we combined review summary and review text as the content. To use this method, we dropped observations without text summary and text review and we split the data into 80% train data and 20% test data and we stratified by user\_id so that every user is in both train data and test data in order to test the accuracy and we combine the review text and review summary variables in order to gather as much as information about the review of each item. We use Tf-idf to calculate the weight of a word to the whole review document and we used cosine similarity to match similar reviews based on counting the maximum number of common words. So we calculated the correlation of items based on their reviews. And we built a correlation matrix to see the correlation between each item so that if a user purchase one item, we can recommend other items based on their correlation. We used the test dataset to test our recall rate, which is around 21%, which is overall good.

* Collaborative Filtering

In this part we will use the result of vader sentiment score from the text mining part. We transform the vader score (from -1 to 1) to ratings from 1 to 5. Thus, we can avoid the 0 in vader score bring confusion to machine learning matrix.

In this model, we only require users ID, item ID and the rating based on the purchase history. We use memory based method which we define no model for user-item interactions and rely on the similarities between users or items in terms of observed interactions.

We use pairwise distance to calculate the similarity and build user similarity matrix and item similarity matrix. The user similarity matrix can be used to perform user-based filtering, means that make recommendations to a user based on the fact that the products have been liked by users similar to the user. The item similarity matrix can be used to perform item-based filtering, which identify similar items based on users’ previous ratings.

For evaluation, we use RMSE (rooted mean squared error) as the indicator, and the result is acceptable and is not overfitted (if the training set RMSE is bigger than the test set, the model is overfitted).

**Conclusion**

* We find out some characteristics of the customers’ purchasing habits on Modcloth. We suggest Modcloth expand their inventory of popular items and popular sizes.
* We build a recommendation system using text mining and machine learning including size recommendation and item recommendation. Customers can be easier to find the fit item and fit size for themselves which will improve their experience at Modcloth.

**References**

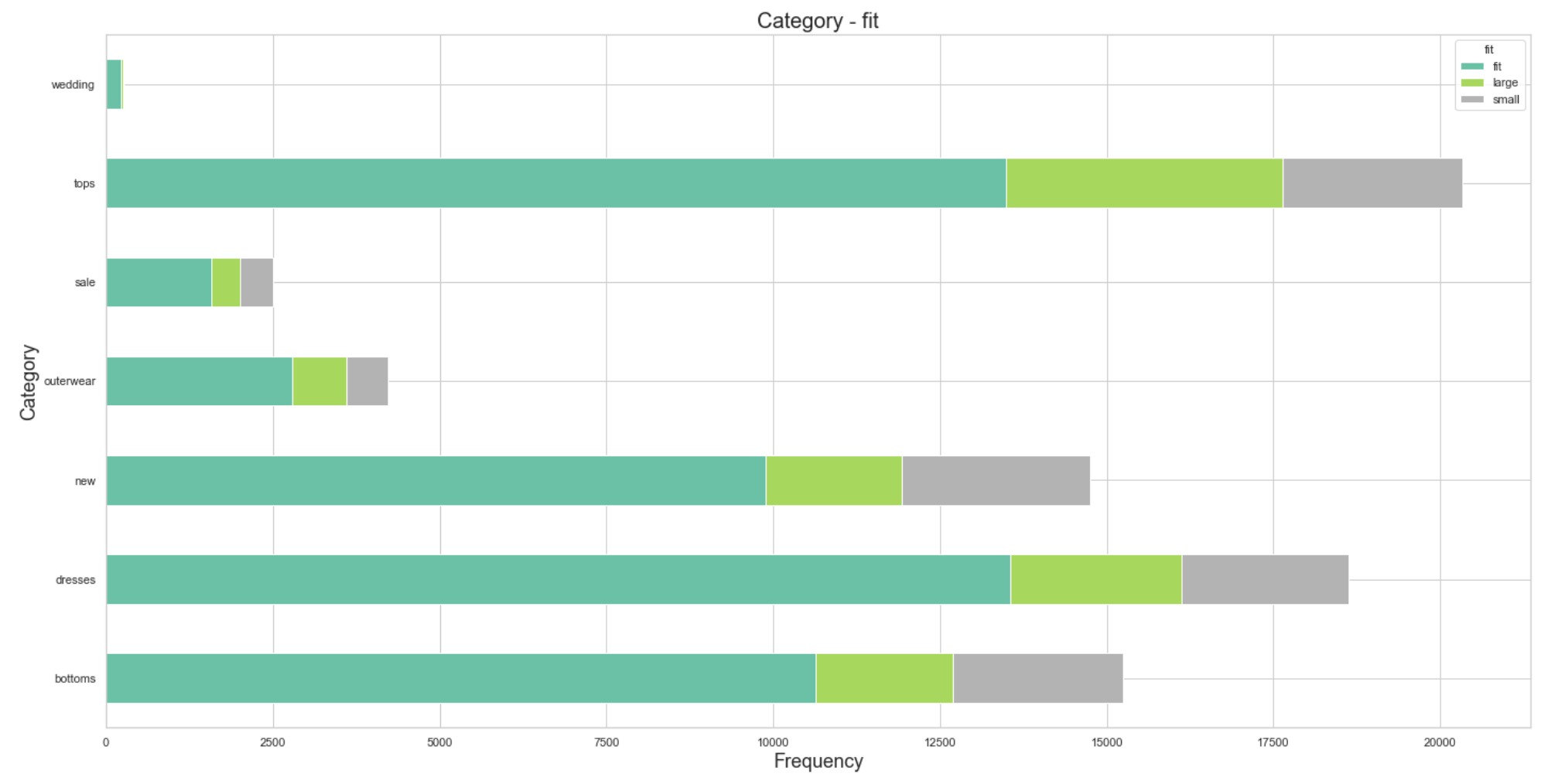
[1] <https://www.modcloth.com/>

[2] Decomposing fit semantics for product size recommendation in metric spaces

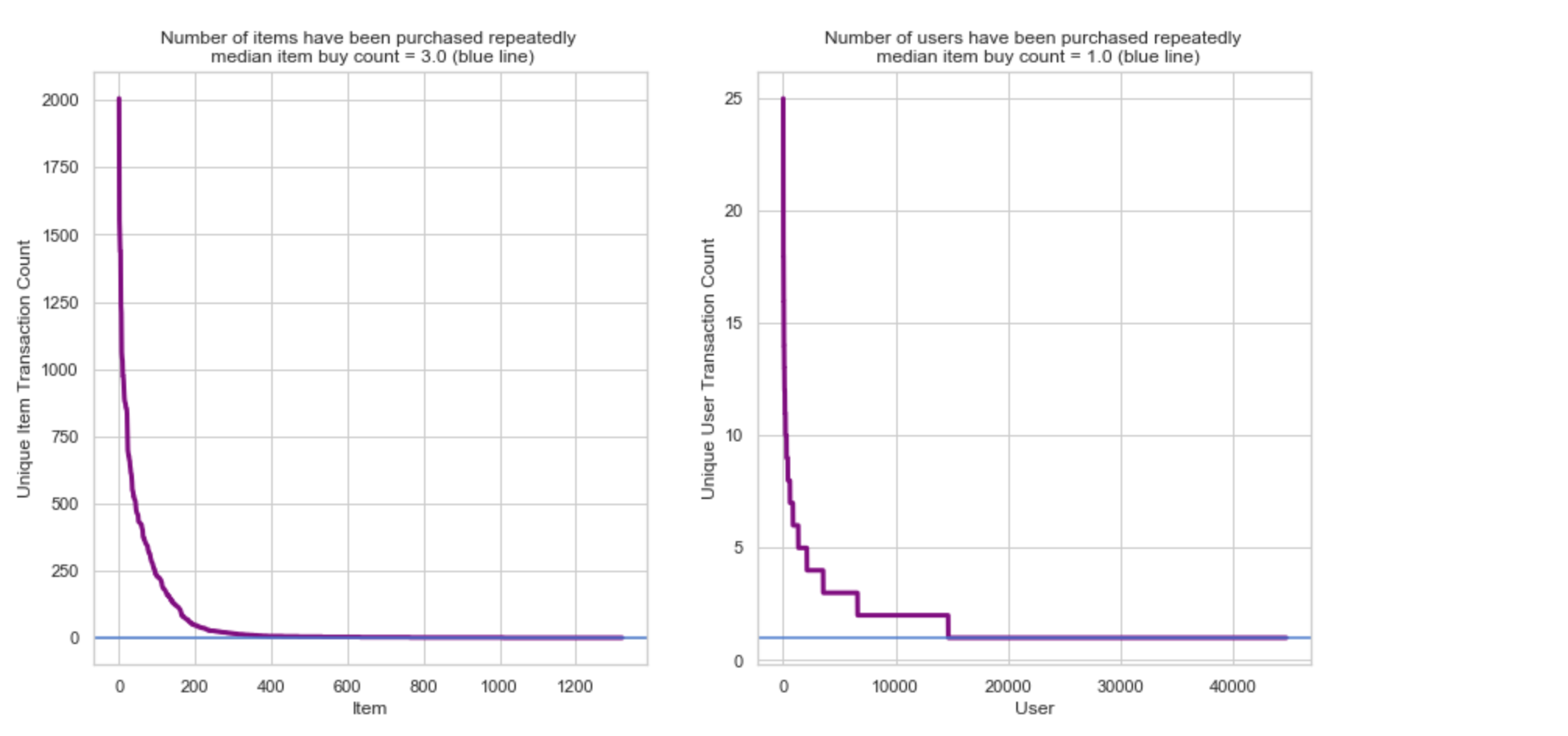
Rishabh Misra, Mengting Wan, Julian McAuley

RecSys, 2018

**Appendix**

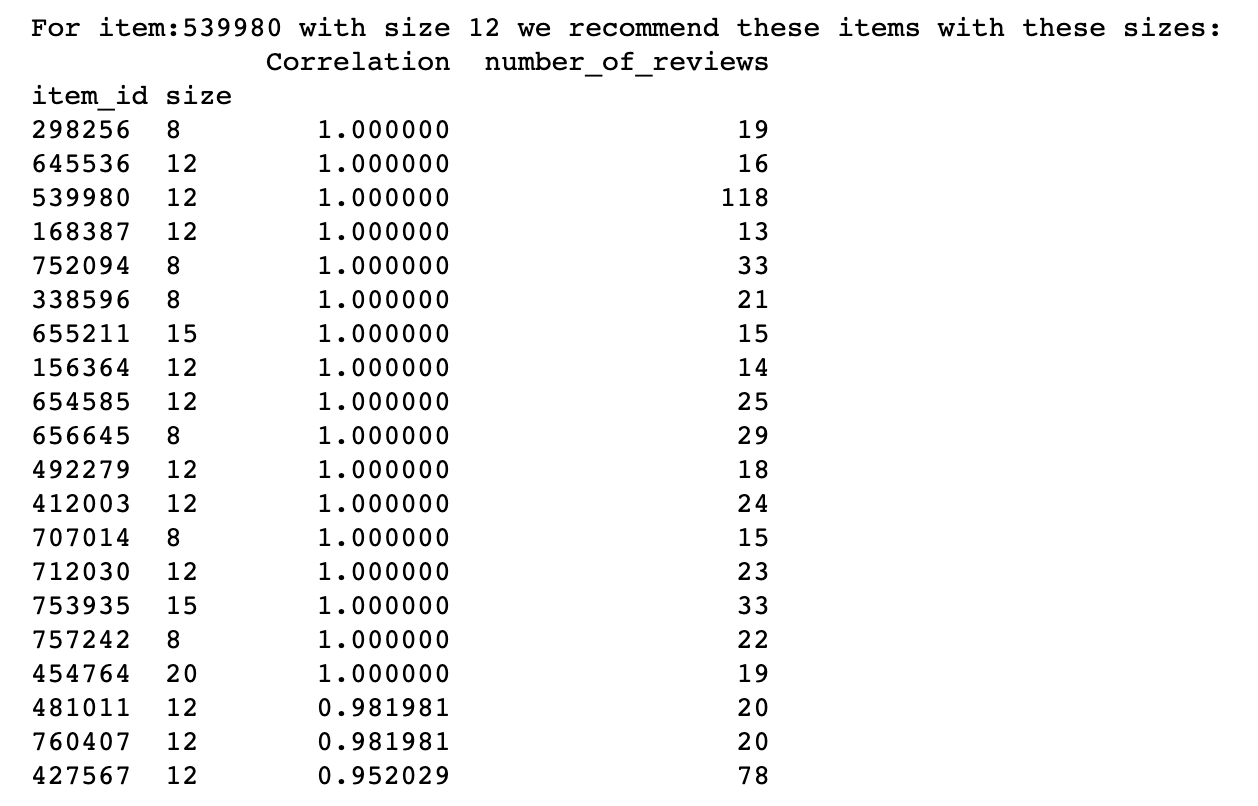


Fit Condition among Categories of Clothing

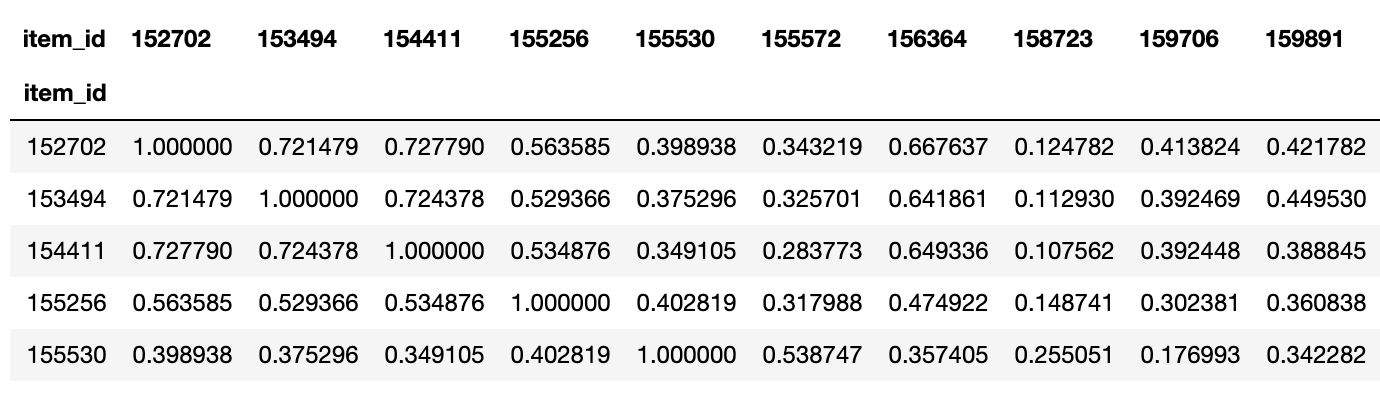


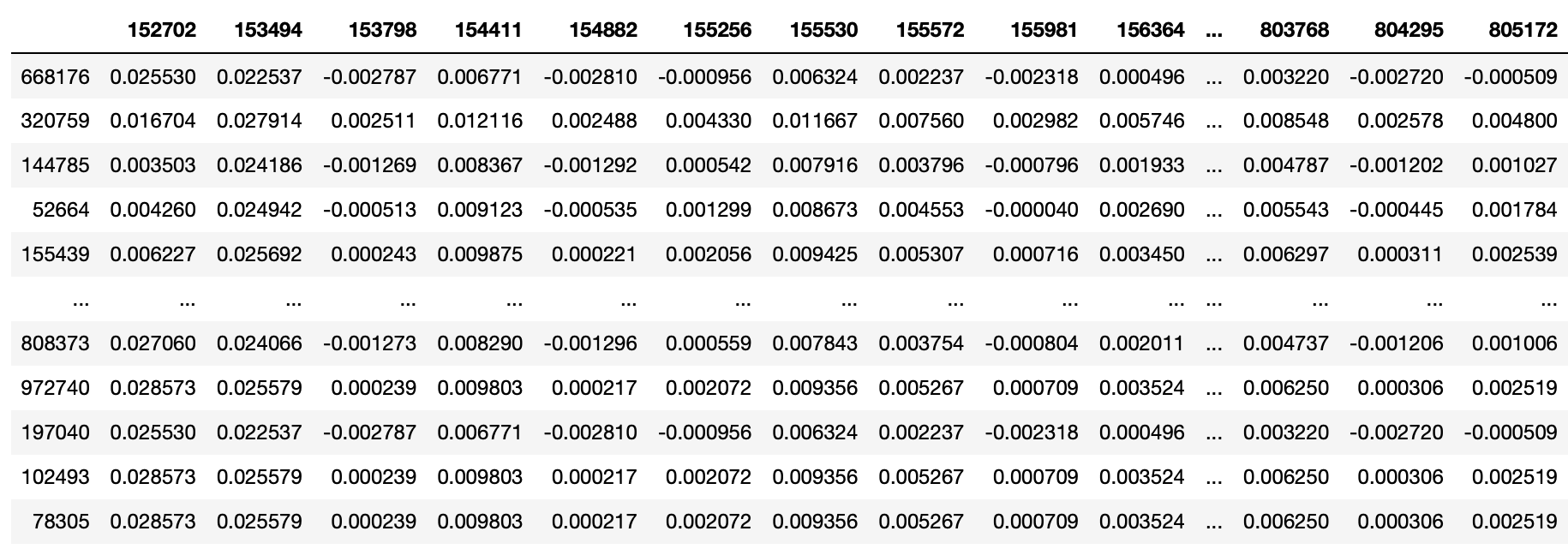
Purchase Count of Each Item and User

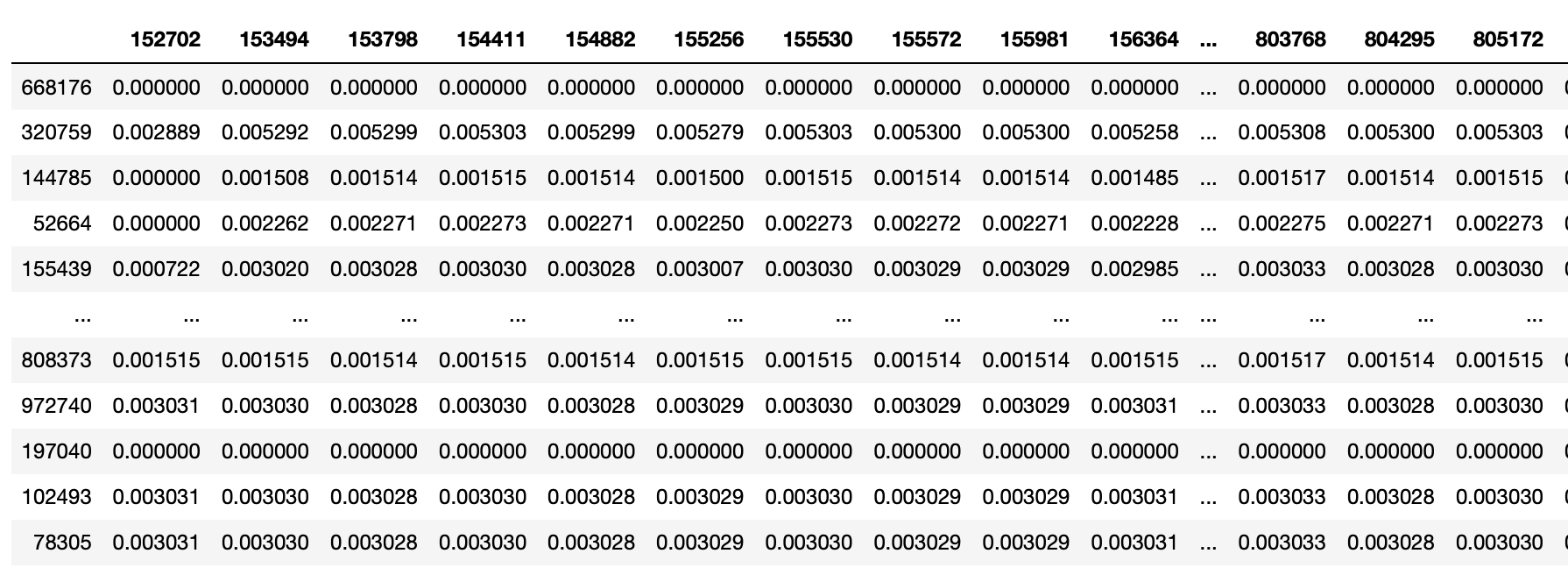
Word Cloud from Text Review of Fit Class 



Example of size recommendation used collaborative filtering

Example of item recommendation used Content based Filtering 

User-base matrix of item recommendation using collaborative filtering 



Item-base matrix of item recommendation using collaborative filtering