

Capstone Project 2: Final Report

Introduction

What do people do when they are bored? Watch a movie, read a book, or play video games, etc. If you choose to play video games, it is not surprising you have used or heard of Steam before. Steam is one of the biggest digital game distributors owned by Valve. There are ninety million active users in Steam according to variety.com. Users pay a one-time fee for most games on Steam. They can comment and rate the game or even recommend games to their friends. The feature is also helpful for users that want to explore new games before purchasing. It is similar to purchasing products from Amazon; people look at the review and rating before they make their bet. In this project, I am going to build item-item collaborative filtering recommender system with five algorithms using the data downloaded from https://cseweb.ucsd.edu/~jmcauley/datasets.html#steam_data.

Data Wrangling

There are two datasets used in this project. Both datasets are downloaded directly from https://cseweb.ucsd.edu/~jmcauley/datasets.html#steam_data. The datasets contain content about Australian steam information. The first dataset (“australian_user_reviews.json”) contains user information such as user id, user url, user review, etc. The second dataset (“steam_game.json”) contains information on steam games such as item id, genres, specifications, publishers, etc.

Because I do not need all the features in both data, I had to extract what is necessary. From “australian_user_reviews.json” dataset, after transforming the data into dataframe, I extracted user_id, recommend, item_id, and review to form a new dataframe. The new dataframe contains 4 columns and 59305 in total with no null values. For the second dataset “steam_games.json”, I had to extract values of each key into lists and build a new dictionary due

to complications of file's quality. Once I got the dictionary I turned it into a dataframe using `pd.DataFrame`. The second dataframe contains 13 columns and 32135 entries with many null values. I left joined the two datasets together on column `item_id`. The new dataset contains 16 columns and 59305 entries in total. To deal with null values, I used values from column tags to fill missing value in column genres. For the price column, I fill the empty cell with the mean price. For the early access column, empty cells are filled with "False". The missing value at the column title is filled with values from the column app name. I also dropped the column discount price because it only contains about 200 values. Finally, I dropped all rows that exist empty values that cannot be otherwise filled in. The clean dataset contains 15 columns and 49704 entries. The final dataframe is saved as a csv file named "data" in the directory for later analysis.

Data Storytelling

After cleaning the data, I did exploratory analysis by plotting some graphs. The following graph showed that only 89.4% people gave games "would recommend" and 10.6% games received "would not recommend."

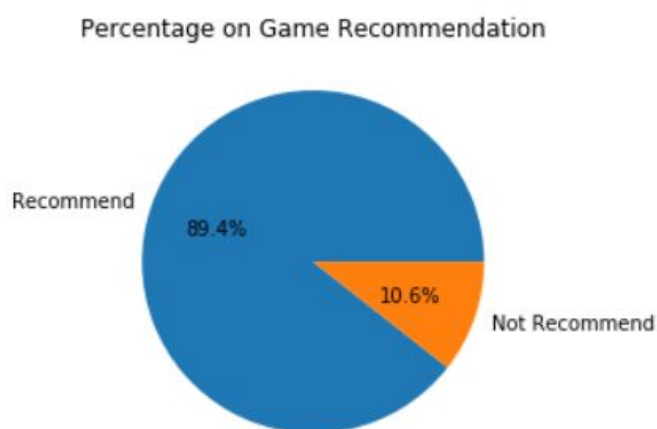


Figure 1. Percentage on Game Recommendations

It seems like Australian people are very happy about most of the games sold on steam. Among all the games, there's only 5.7% of the games on steam launch early access.

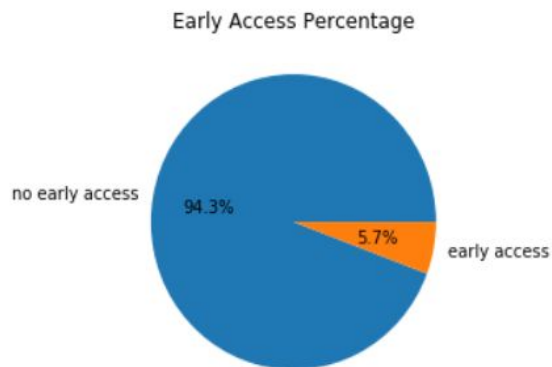


Figure 2. Early Access Percentage

For those that didn't launch early access, I assume the publisher either has a great team and is very confident at their product or they are just carefree. I also made some top charts on games, publishers, and genres. They are shown as following:

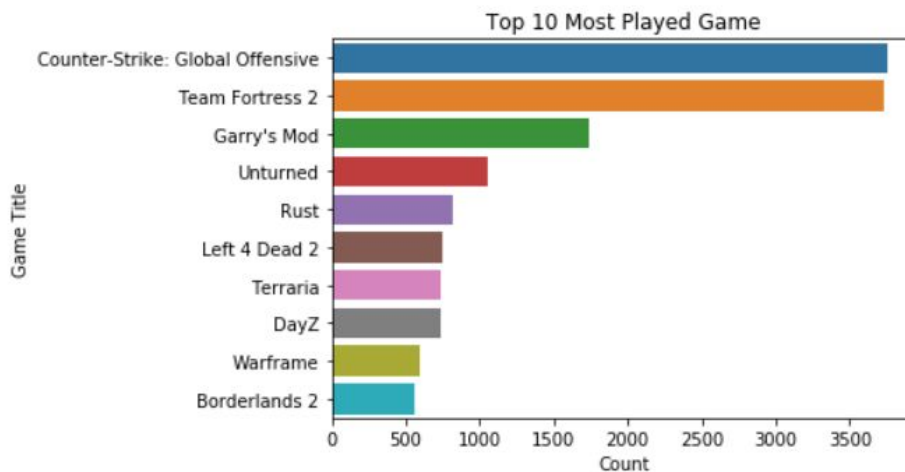


Figure 3. Top 10 Played Game

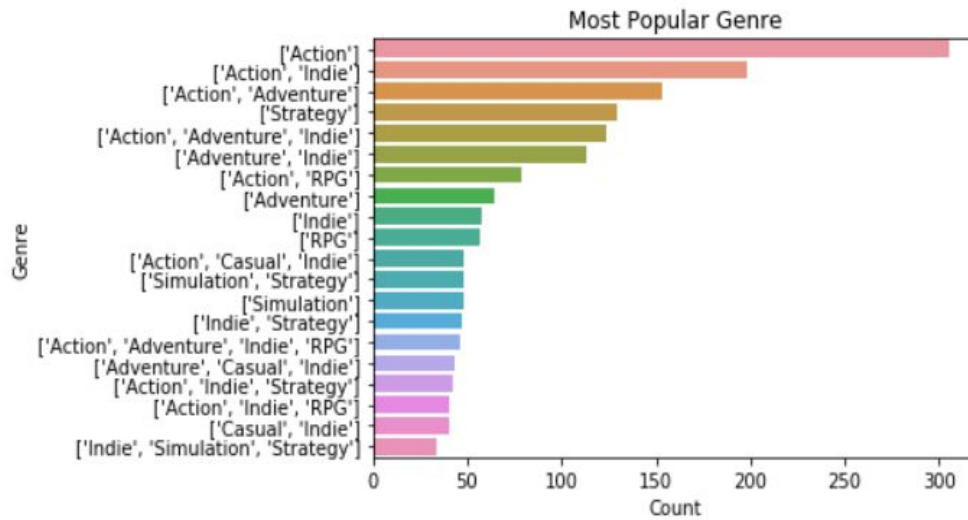


Figure 4. Top genres

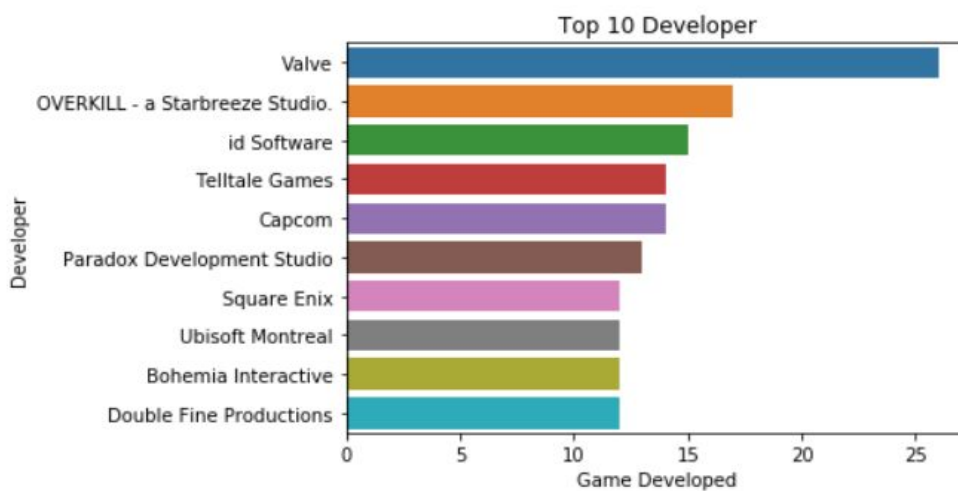


Figure 5. Top 10 Developers

What is interesting from figure 4 is that it seems like Australian people really don't play much games outside of action, strategy, and RPG games. It is surprising to see from figure 5 that many developer's names that I've never heard of. Finally, I plotted the price range of the games:

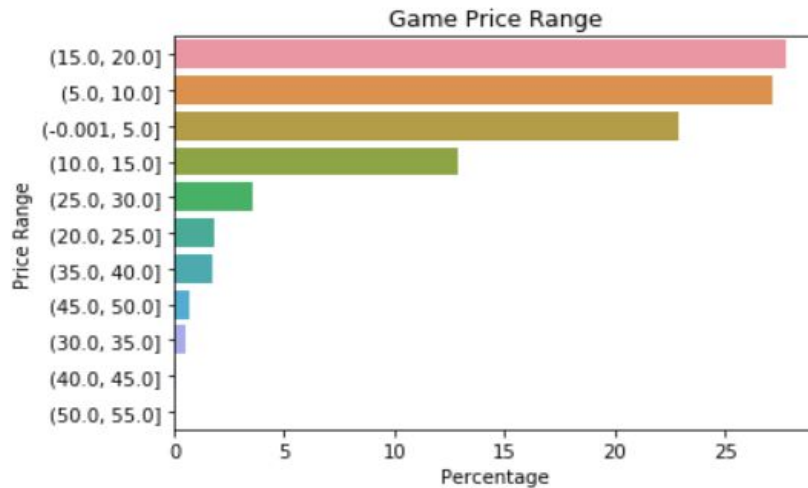


Figure 6. Game ranges

About 80% of the game is priced under \$20. It's rare to see games exceed \$20. Is it because that Australian people do not purchase games over \$20?

Statistical Analysis

From our storytelling project, there is a lot of interesting information. In this part of the process, I am going to further explore some of the information presents and perform A/B testing.

There are 3 questions that I want to answer:

1. Do ['Action'] games cost more than ['Action', 'Indie'] games?
2. Do multiplayer games cost less than single-player games?
3. Among the two most popular games (Counter-Strike: Global Offensive and Team

Fortress 2), which has a better reputation in terms of recommendation?

The null hypothesis for question 1 is that the average price of ['Action'] games greater or equals the average price of ['Action', 'Indie'] games? After filtering out the rows that contain specific genres, I used bootstrap to loop over 10000 cycles and calculated the confidence interval of the two groups. The 95% confidence interval is [5.79101534, 6.50657473]. I plotted the distribution graph as follows:

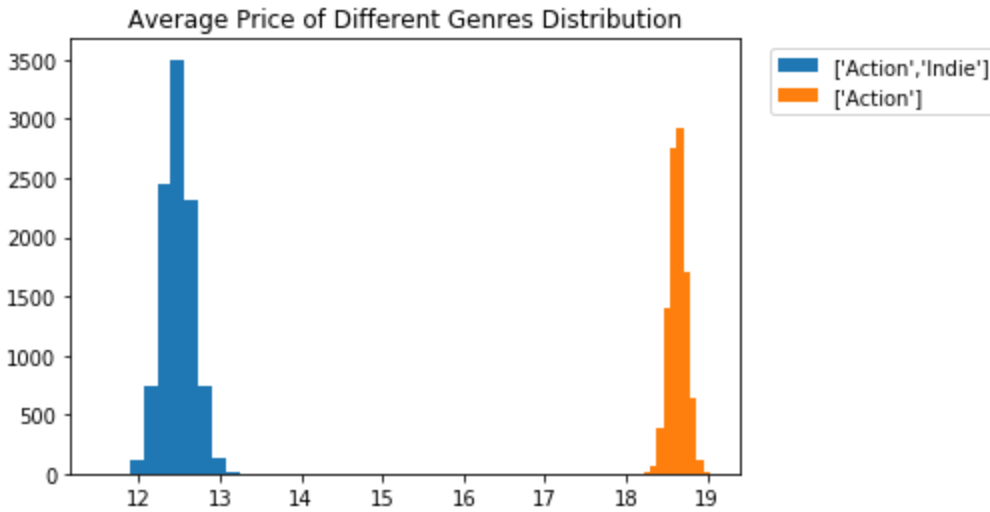


Figure 7. Average Price of Different Genres Distributions

Finally, I calculated the p-value which is less than the alpha level thus concluding that it is lucid that [‘Action’] games have higher average price than [‘Action’, ‘Indie’] games.

For question 2, the null hypothesis is that the average price of single-player games is greater or equal to the average price of multiplayer games. After filtering out the rows I used bootstrap to loop over 10000 cycles to find the confidence interval. The result came back as [-2.93274549, -2.60298355]. I plotted the distribution of the two sample groups:

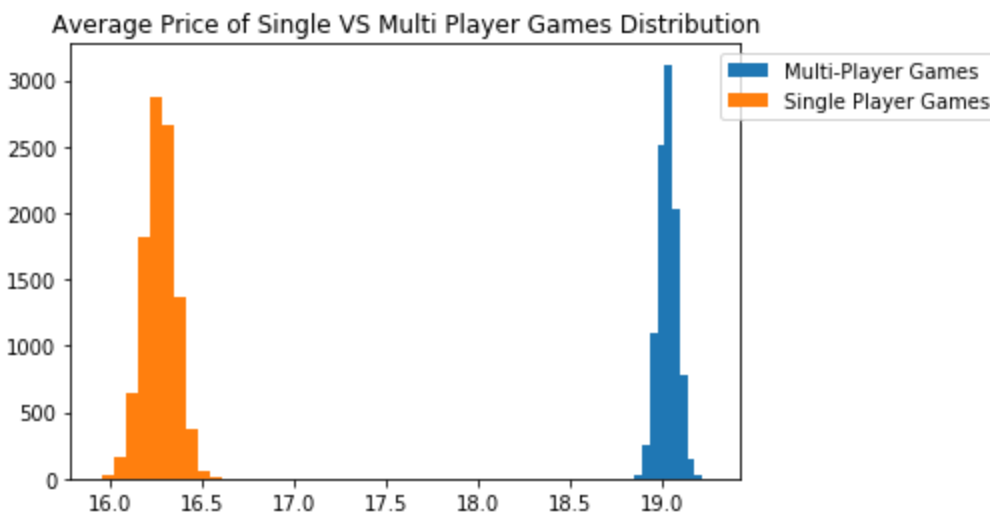


Figure 8. Average Price of Single VS Multi Player Games Distributions

Based on the confidence interval and distribution graph, it is clear that Single players games do not cost more than multiplayer games. Multiplayer games cost more than single-player games.

For question 3, the null hypothesis is that the percentage of recommended from Counter-Strike: Global Defense is less or equal to the percentage of recommended from Team Fortress 2. After selecting the appropriate rows, I performed bootstrap on the dfs over 10000 cycles and obtained the confidence interval of $[-4.98, -3.2795]$. I plotted the distribution graph as following:

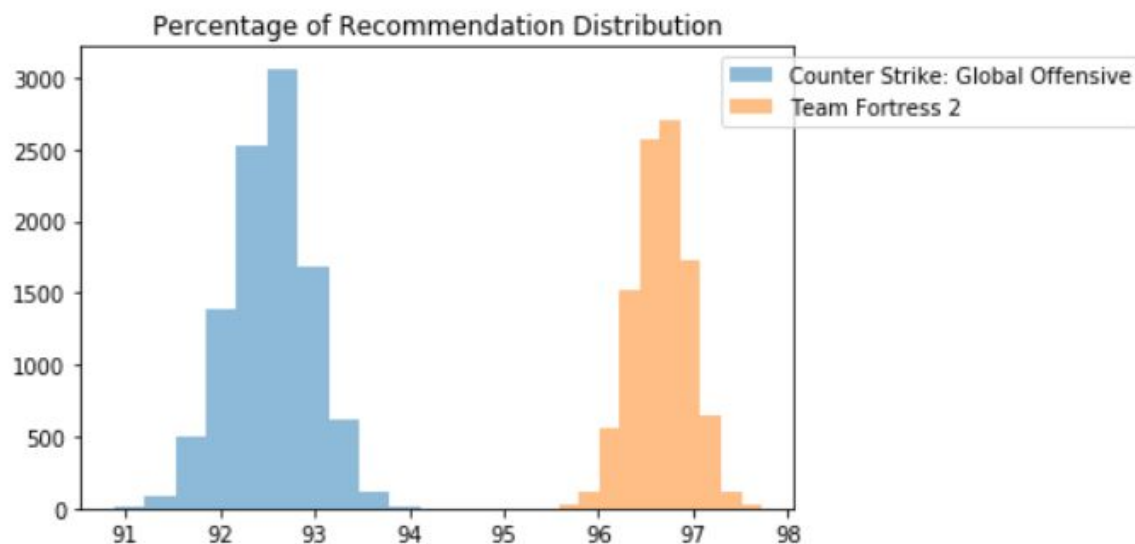


Figure 9. Percentage of Recommendation Distributions

Based on the evidence presented, CS:GO didn't have a greater recommendation rate than Team Fortress 2. In fact, Team Fortress 2 has a greater recommendation rate than CS:GO.

Building Recommender System

In this part of the project, I decided to build an item-item collaborative filtering after seeing that there are much more users than games (users:18875, games:2554). By doing item-item filtering, I could save a massive amount of time.

For building the recommendation system, I primarily use the Surprise library. After load in the dataset, I extracted the necessary columns to form a new data frame. The data frame is later loaded into Surprise's module: Dataset. I also set the rating scale to (0, 1) through Reader because I am working with a binary dataset. The dataset is then split into a train and test set for encountering overfitting and for evaluation.

The dataset is fed into five different models using cosine similarity. For KNN families, the models are tuned on two parameters: k and min_k. The SVD is also tuned on two parameters: n_epochs and lr_all. The following is the result:

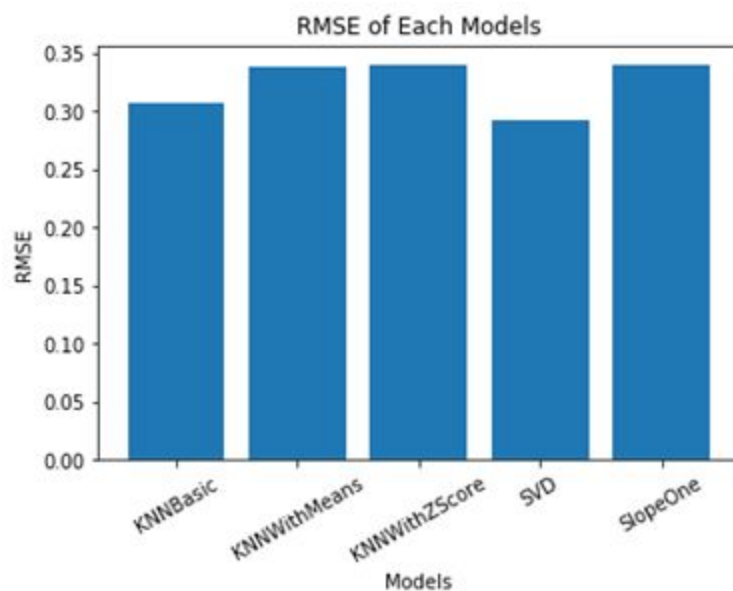


Figure 1. RMSE of Each Models

It is clear that SVD has the lowest RMSE out of the five models thus making the best model to use for the dataset.

Conclusion

There is no doubt that the recommender system played a significant role in users' online purchase experiment. Many people are willing to pay for products that the recommender system guided them to. According to <https://neoteric.eu/> , “59% of shoppers who have experienced [recommender system] say it has a big influence on their purchase decisions”. This is exactly the reason why many companies start to incorporate a recommender system into their online website.

Item-item collaborative filtering recommender system is just one of the many recommender systems people use. There are content-based, user-item, and hybrid recommender systems that can also be used on this dataset.

Dataset Contributors

Self-attentive sequential recommendation

Wang-Cheng Kang, Julian McAuley

ICDM, 2018

[pdf](#)

Item recommendation on monotonic behavior chains

Mengting Wan, Julian McAuley

RecSys, 2018

[pdf](#)

Generating and personalizing bundle recommendations on Steam

Apurva Pathak, Kshitiz Gupta, Julian McAuley

SIGIR, 2017

[pdf](#)