Predict Ratings of Mobile Strategy Games

Yuting Guo, Yao Ge

December 2019

Abstract

According to 2019 Global Games Market Report, the global gaming market is estimated to be worth billions of dollars, with 45 percent of that, coming directly from mobile games [1]. Among all kinds of mobile games, strategy games are one of the most popular types. With this dataset, the insights and analysis of strategy games can be conducted. In this project, our task is aim to figure out what factors contribute to the success of strategy games. To realize this idea, we implement various machine learning models and compare the results from them. Finally, we discuss and analyse the performance of models and conclude the factors which contribute more to the success of strategy games.

1 Introduction

According to 2019 Global Games Market Report, the global gaming market is estimated to be worth 152 billion dollars, with 45 percent of that, 68.5 billion dollars, coming directly from mobile games [1]. In fact, mobile games are growing at an alarming rate which lead to an increasing number of investments in the mobile games industry. One of the reasons for this phenomenon is that tremendous advancements in smart-phone technology render the mobile games more attractive. With this dataset, the insights and analysis of strategy games can be conducted. Our task is aim to figure out what factors contribute to the success of strategy games.

This dataset was collected on the 3rd of August 2019, using the iTunes API and the App Store sitemap. The input is a matrix corresponding to 17007 games and 17 columns. The output is a vector of the number of ratings corresponding to 17007 games.

One of the reasons we chose this dataset is that the insights on this dataset play an important role on the mobile strategy games market. On the one hand, by mastering the decisive factors of the popularity of strategy games and the preferences of players, game developers can discover the existing problems in the game, in addition to optimize its performance to meet the needs of players. On the other hand, by analysing this dataset, the Apple Store can foresee the future market of popular strategy games and decide on a recommended list of these games.

2 Related Work

There are several works related to predict ratings of video games using using classification model or regression models. Zhang et al. [8] propose a supervised learning technique to summarize videos by selecting key-frames. Glorot et al. [6] propose a deep learning model to obtain knowledge from reviews. Batchu et al. [2] propose some ideas about how to utilizing attributes of video games such as genres.
Also, many teams individuals shared their work on this dataset on Kaggle. However, the vast majority of their work is in the way of feature engineering, while a few of them stated the prediction results of various models. In [7], the author use Linear Regression, XGBoosting and Automated machine learning to predict the rating of testing data.

3 Our Approach

First, we check the missing value, merge the label of ratings and use feature engineering to pre-process the dataset. Then, we build different machine learning models to predict the number of ratings of testing data and analyse the difference of their performance. Finally, we use AUROC as evaluation metric.

3.1 Data Preprocessing

To train the Machine and Deep Learning methods multiple datasets with relevant features were created. Since the data does not contain information about something like apps downloaded count, ”Average User Rating” is the only way to decide whether a game is successful or not. By checking missing data and feature engineering, we make a new dataset from the original data in case we need the full dataset for future analytic.

3.1.1 Data Analysis

First, we check the simple data information and missing data. Since ”Average User Rating” is the target of this dataset, any NA data which is not really relevant or important to keep, are dropped. After dropping, 7561 games are remained.
The distribution of ”Average User Rating” is shown below:

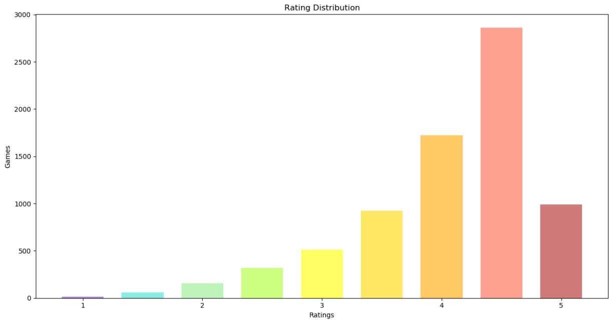


Figure 1: The distribution of rating

Figure 1 shows that 73.7 percent games have a rating greater than 4.0, which indicates the imbalance of this dataset.

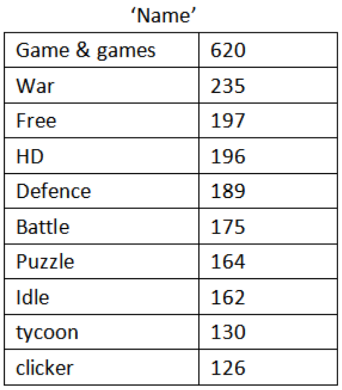
3.1.2 Label Merging

As Figure 1 shows, this dataset is very imbalanced on various ratings. Since the target of this task is to figure out what factors contribute to the strategy game ratings, labels of various ratings were merged into two labels, and the task was converted into a binary classification task. Based on the distribution of ”Average User Rating”, games with ratings greater than 4.0 are labeled as 1; otherwise, they are labeled as 0 in order to balance the size of data of each category.

3.1.3 Feature Engineering

Since 17 columns in this dataset have different types, we use various methods to analyse the relationship between features and rating, so that these features could be converted into real-values and become the input of models.
First, some features which have little impact on the success of strategy games such as ”URL”, ”ID” and ”Primary Genres” (96 percent of this feature are ”Game”), we drop these features from the dataset. Second, for the real-value features such as ”Price”, ”User Rating Count” and ”Size”, we use the original values directly. In addition, other features will be introduced the pre-process methods in detail.
”Name” In the features ”Name” it is significant to decide which words are most used. Due to this task, first the number of games whose ratings are greater than 3.5 are calculated, then top 3 keywords for ”Name” are selected respectively. After selecting, games are encoded with 1 (including top 3 keywords) or 0 (not including). Finally, word cloud, a novelty visual representation of text data, is used to depict keyword metadata (tags) on websites, or to visualize free form text. We also cut the words and calculate the frequency of words, the result is

shown below:



'Name'

Game & games	620
War	235
Free	197
HD	196
Defence	189
Battle	175
Puzzle	164
Idle	162
tycoon	130
clicker	126

Figure 2: Keywords of "Name"

According to the Figure 2, the top 3 keywords of feature "Name" are "game/games", "war" and "free".

"Subtitle" The method we use to pre-process feature "Subtitle" is similar to "Name". The top 3 keywords of feature "Subtitle" are "game/games", "strategy" and "puzzle".

"User Rating Count" The scatter plot for the relationship between "User Rating Count" and rating is shown below:

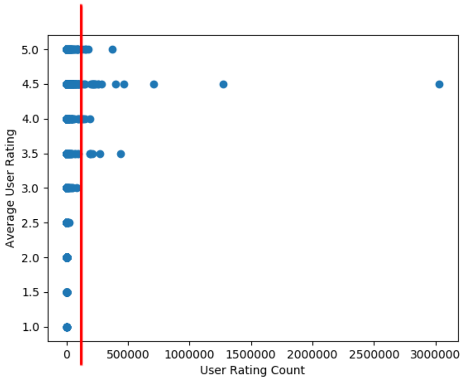


Figure 3: The relationship between "User Rating Count" and rating

Figure 3 shows that there might be some relationships between "User Rating

Count" and rating, since from around 10,000 to 50,000 counts, most of the games have the ratings greater than 3.5.

"Price" For the relationship between "Price" and rating, however, the result shows little relationship.

"In-app Purchases" We use 1 or 0 to label this feature, if the game is free, "0" is used, others are "1".

"Developer" For the feature "Developer", we are supposed to figure out whether the famous companies leads to higher rating of games or not. Therefore, we calculate the number of games of each developer, and we use these numbers as the feature to substitute "Developer". For judging its impact on rating, we also consider the results of feature selection based on different models.

"Age Rating" We remove "+" from this feature.

"Languages" Since the majority of the games are multilingual, the number of languages are used to represent this feature.

"Size" For analysing the months and years trends in games' size over time, we analyse the relationship between "Size" and "Original Release Date". The result indicates that the original release date of games highly effect the games' size, thus leading to the question about whether the date will effect the rating. The result is shown below:

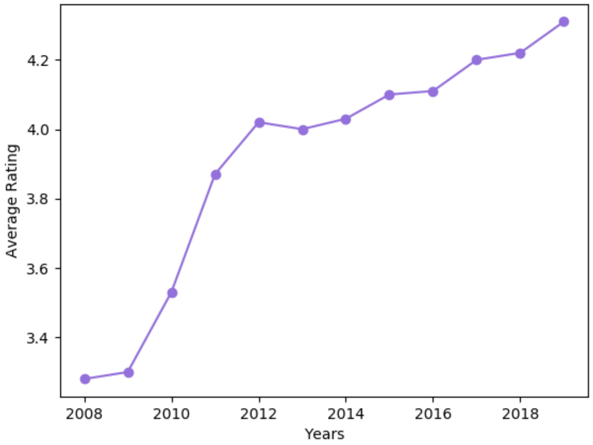


Figure 4: The relationship between date and rating

According to the Figure 4, it is concluded that with time going by, the average rating increase every year. However, in the previous work, we just consider the update Gap as the feature. In the future work, we will add "Original Release Date" as another feature of the dataset.

"Update Gap" Update Gap (days) = "Current Version Release Date" - "Original Release Date".

"Genres" Except the genres "Game", "Entertainment" and "Strategy" which are the labels of each game, the distribution of genres are shown below:

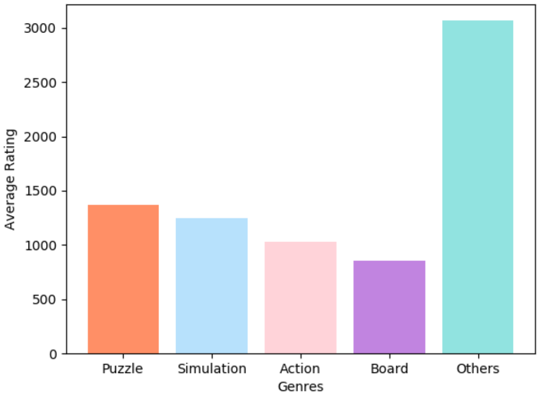


Figure 5: The distribution of Genres

The top 4 popular genres in this dataset are "Puzzle", "Simulation", "Action" and "Board", so we use one-hot encoding strategy to encode games with 1 (including one/several of top 4 genres) or 0 (not including) to represent the feature of "Genres", which means expand this one column "Genres" to 4 columns "Puzzle", "Simulation", "Action" and "Board".

3.2 Models

Logistic Regression The baseline model is Logistic Regression, which is one of the most widely used models for classification tasks [5]. Logistic regression is a statistical model using a logistic function as the loss function.

Support Vector Machine (SVM) SVM [3] is a non-linear classification model that maps inputs into high-dimensional space using kernel trick, which is also a popular model for classification tasks.

Boosting Tree Boosting Tree [4] is a boosting model that automatically build a tree in which each node combines a weak classifier, which is shallow decision tree in this case, using gradient boosting.

3.3 Metric

In order to measure the quality of classification models, we use the receiver operating characteristic curve (ROC) and compute the area under the curve (AUC) score, which is also called AUROC. One advantage is that we do not need to pay attention to the threshold using AUROC so that we can focus on the model analysis.

4 Experimental Results

The performance of the models is compared based on their AUC score of predicting correct strategy game ratings.

Model	AUC (train)	AUC (test)
Logistic Regression	0.58	0.61
SVM	1.00	0.52
Boosting Tree	0.78	0.72

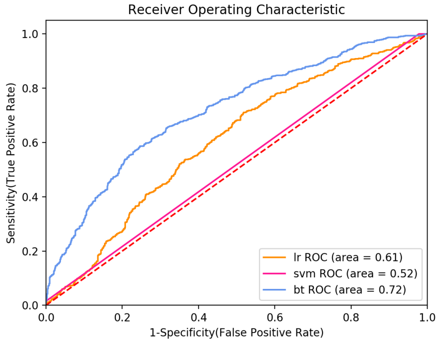


Figure 6: The ROC curves of Logistic Regression (lr), Support Vector Machine (svm), and Boosting Tree (bt) on testing data without filtering games with low user count.

Based on the above results, the observations that the Boosting Tree model is the model of highest AUC score, showing that the feature importance of the Boosting Tree model is mostly close to the important factors that contributes to strategy game ratings. Figure 7 below shows the feature importance of the Boosting Tree Model.

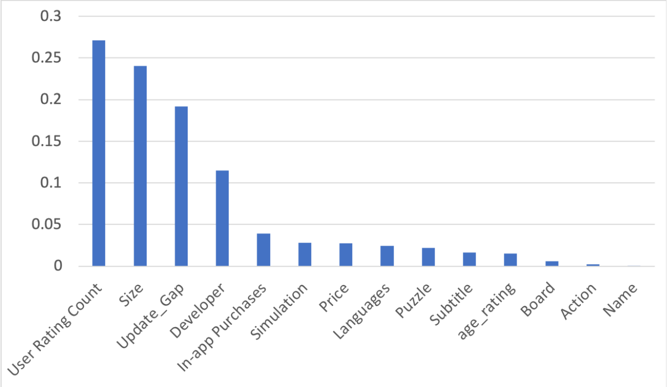


Figure 7: The feature importance of the Boosting Tree model without filtering games with low user count.

Intuitively, the feature of ”User Rating Count” can substantially effect the ratings, and the ratings of games rated by a few people can be biased. For example, if a game is just rated once, the rating of it is irrelevant to the game itself. In order to mitigate the effect of ”User Rating Count”, games with ”User Rating Count” less than 200 is filtered. The size of new training data is 1814, and the size new testing data is 429. To figure out whether the Boosting Tree model is still the best model, the three models were re-trained on the new data.

Model	AUC (train)	AUC (test)
Logistic Regression	0.63	0.68
SVM	1.00	0.52
Boosting Tree	0.89	0.70

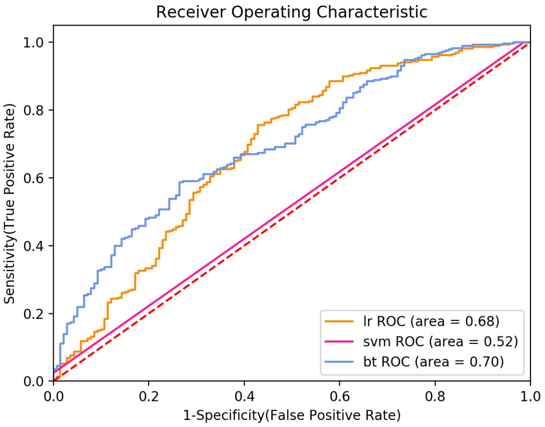


Figure 8: The ROC curves of Logistic Regression (lr), Support Vector Machine (svm), and Boosting Tree (bt) on testing data with filtering games with low user count.

Although the AUC score of the Boosting Tree model is lower than that of the Boosting Tree model trained on the entire data, the Boosting Tree model is the model of highest AUC score among these three models. Figure 9 below shows the feature importance of the Boosting Tree Model.

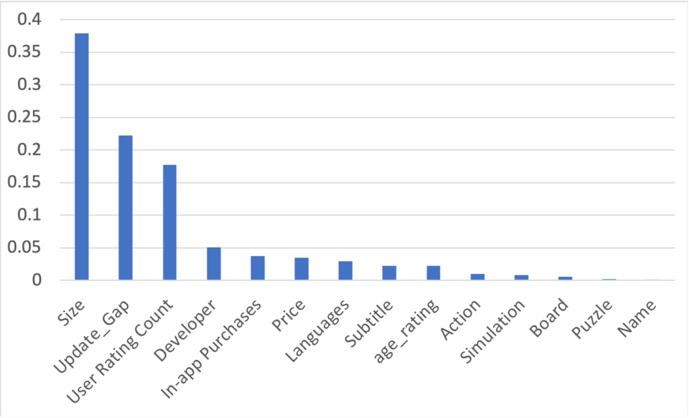


Figure 9: The feature importance of the Boosting Tree model with filtering games with low user count.

5 Discussions

5.1 Analysis

Based on the classification results, the Boosting Tree model is the model of best performance on this task. The reason is that the Boosting Tree model is using shallow decision trees, which means that it is a model of high bias and low variance. Because the data size is small, models of high variance like SVM are at a risk of over-fitting.

From Figure 7 and Figure 9, the following observations about what factors contribute to the ratings of strategy games can be made:

- For strategy games, the top 3 important factors for the ratings is sizes, the update frequencies, and scales of game developers or companies. The results are intuitively reasonable. Since the bigger the sizes are, the more complex the games are, which means that the games may contain more content. Similarly, the update frequencies may be relevant to the popularity today. Also, games released by famous developers and companies tend to obtain higher ratings.
- For strategy games, subcategories and names do not contribute to the game ratings. One explanation is that strategy games of various subcategories are not obviously different from each other, and names are too short to contain important information.

5.2 Future Work

In this project, we propose some ideas for feature engineering of strategy games. We compare the performance of different models including traditional machine learning models and analyze their performance. Also, we learn some conclusions about what factors contribute to strategy games.

Our future work is to conduct more feature engineering on icon data and date features. Although we do not utilize icon data because of some technical difficulties, icons of video games are intuitively influential for the popularity of games. Furthermore, since the results of our experiments show that the size of strategy games contribute substantially to the game ratings, and the size of strategy games has been increasing overtime, it seems that date features may also be important for game ratings.

References

[1] Global games market report. Technical report, newzoo, 2019.

[2] Vishal Batchu, Varshit Battu, Murali Krishna Reddy, and Radhika Mamidi. " how to rate a video game?"-a prediction system for video games based on multimodal information. *arXiv preprint arXiv:1805.11372*, 2018.

[3] Olivier Chapelle, Patrick Haffner, and Vladimir N Vapnik. Support vector machines for histogram-based image classification. *IEEE transactions on Neural Networks*, 10(5):1055–1064, 1999.

[4] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794. ACM, 2016.

[5] Stephan Dreiseitl and Lucila Ohno-Machado. Logistic regression and artificial neural network classification models: a methodology review. *Journal of biomedical informatics*, 35(5-6):352–359, 2002.

[6] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 513–520, 2011.

[7] Margesh Phirke. Regression scikit xgb h2o automl. Kaggle, 2019. https://www.kaggle.com/margesh/regression-scikit-xgb-h2o-automl.

[8] Ke Zhang, Wei-Lun Chao, Fei Sha, and Kristen Grauman. Video summarization with long short-term memory. In *European conference on computer vision*, pages 766–782. Springer, 2016.