Predict Players Engagement Level using KNN Method

Junyi Pan *

Nanning No.2 High School, Nanning, China

* Corresponding Author Email: zero_plfjy@outlook.com

Abstract. According to the video game industry development, players behavior analysis in game corporation becomes more essential than before, which not only helps game designer know the market needs, but helps company making future decision of the game as well. Therefore, in this experiment, the author built a model for machine learning about players behavior, a dataset found on Kaggle, which contains 40, 034 players with 13 features of each player such as Player ID, Age, Gender, Location, Game Genre, Play Time Hours, In Game Purchases, Game Difficulty, Sessions Per Week, Avg Session Duration Minutes, Player Level, Achievements Unlocked and Engagement Level. This experiment will use K-Neighbor (KNN) algorithm to make a model with random forest and some evaluation metrics to evaluate the model. The results of this research are not only valuable for predicting player engagement but can also provide actionable insights for game companies, such as optimizing monetization strategies, improving game design, and enhancing personalized user experiences.

Keywords: KNN, machine learning, game behavior.

1. Introduction

Nowadays, video game industry is increasing rapidly. For instance, the report on the global gaming industry delivered by the analytics and research company Newzoo shows that the value of the global gaming market in 2019 was USD 152.1 billion, 10% more than the year before [1]. Therefore, for video games companies, data of player's analysis is very important, which can help them to get more information and release new games to specific players. For example, video games companies can analyze existing player's behaviors and get predict a new player's favor games or his game engagement level and recommend proper games to this new player or help the video game companies to develop new games to cater for the market.

In 1950, Alan Mathison Turing published a paper which proposed Turing test to determine whether a machine is intelligent [2]. Five years later, Artificial intelligence (AI) was first introduced by John McCarthy and established as a subject at Dartmouth workshop [3]. In 2022, company Open AI published their latest chat bot called ChatGPT based on the large language model GPT-3.5 trained though Reinforcement learning. Machine learning as an essential part of AI, is growth sharply and it is a good way to train the computer provide a reliable prediction with big data. A lot of machine learning algorithms like K-Neighbors (KNN), Random Forest, Decision tree can serve as our tools to get efficient way to train the predict model.

According to the research of this paper [4], the author found the data analysis of player's behaviors not only can help game developers keep engagements but also can increase revenue and extend games lifecycle. So, it is essential for analysis gamers' data and find some law of the players.

In order to help game developers to get more information about this market. In this study, the author aims to analyze data of a dataset [5], do some visualization about this dataset, make a predictor using KNN algorithm to predict a new gamer's engagement level by corresponding behaviors.

2. Method

2.1. Dataset preparation

This study used the dataset from Kaggle [5], which contains 40, 034 players with 13 features of each player such as Player ID, Age, Gender, Location, Game Genre, Play Time Hours, In Game

Purchases, Game Difficulty, Sessions Per Week, Avg Session Duration Minutes, Player Level, Achievements Unlocked and Engagement Level. First of all, data visualization is important for analyzing data. These are some charts of the features' distributions in the dataset shown in Fig. 1

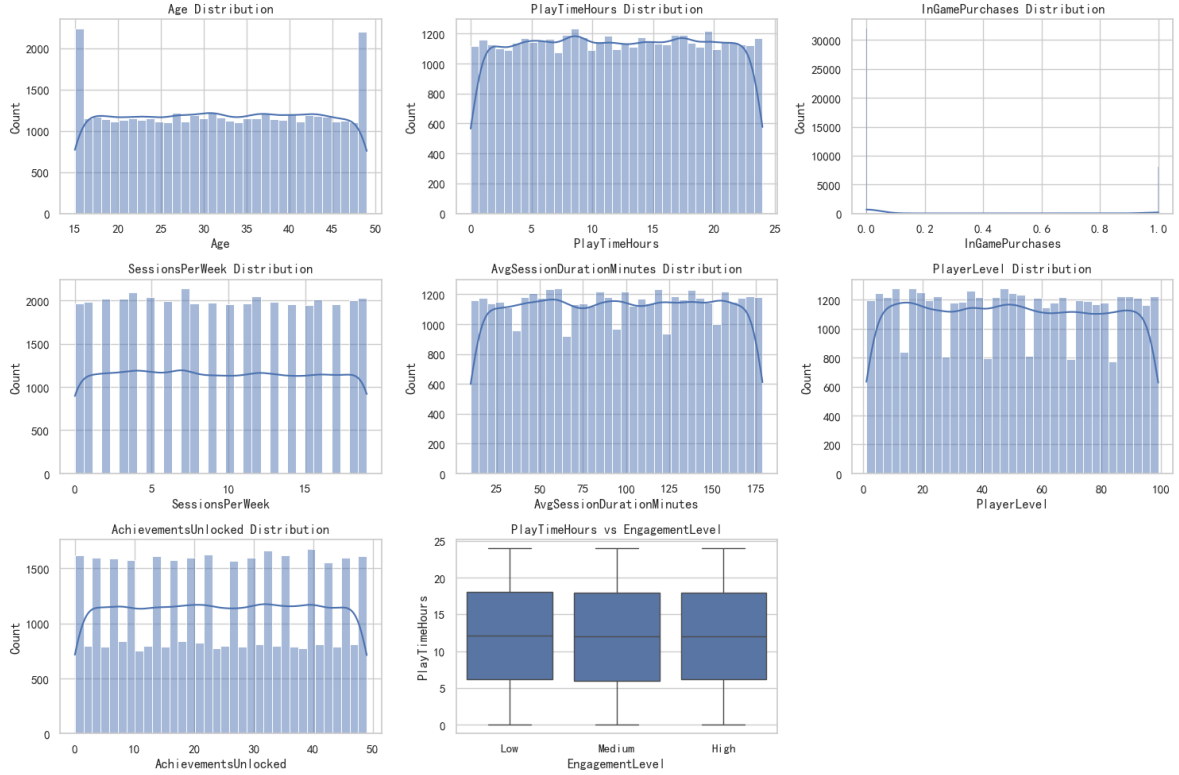


Fig 1. The distribution of some features (Picture credit: Original)

It is clear that these distributions are equally distributed, which contains enough machine-learning-needed data because of it uniformed from the author's perspective.

However not all columns (features) are useful and suitable for machine learning directly for example Player ID is a needless feature obviously. Therefore, the author needs to do some preprocessing of these data. First drop the Player ID column because it is needless as I mentioned. Then feature engineering is needed for Play Time Hours and Achievements Unlocked. Imagine that we cannot know a player who plays games 2 hours every day or only plays 5 hours on the weekend through the single data "Play Time Hours -- 5". The paper written by Olaoye indicated that feature engineering is a process of selection, conversion and new feature labeling from the original data and this is an important step of machine learning models performance improvement [6]. Therefore, feature engineering is important for machine learning especially for KNN and can solve this problem. In this experiment, the author first uses Play Time Hours divided by Sessions Per Week to get session efficiency values of each player as a new feature added into the dataset. The author also uses Achievements Unlocked divided by Player Level and gets the achievement rate values also added into the data. After the feature engineering, the boxplot of relationship between session efficiency or achievement rate and engagement level shown in Fig. 2.

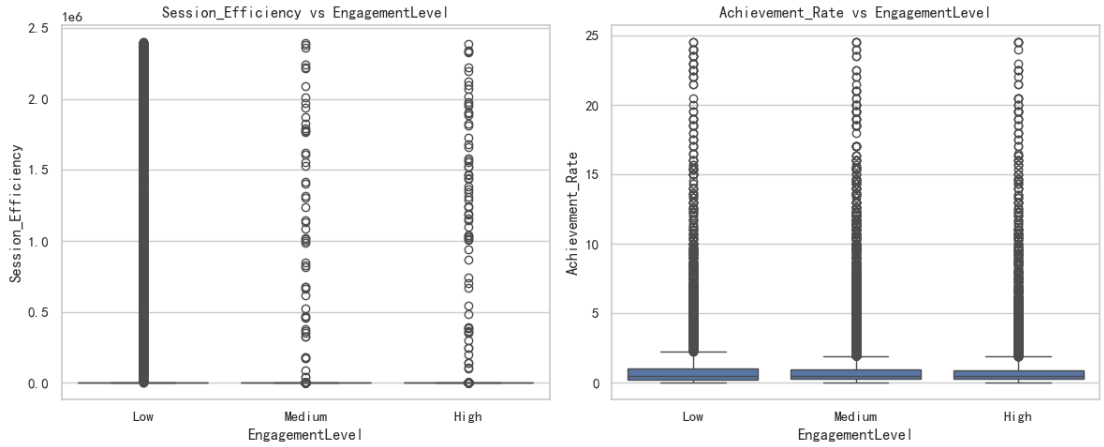


Fig 2. Initial boxplot of relationship between session efficiency and engagement level and relationship between achievement rate and engagement level (Picture credit: Original)

Obviously, the initial boxplot shows extreme values due to exception values in the initial dataset. There might be some extreme value that is not reasonable, for example, a player session efficiency reached 2,400 hours per week which is impossible. Therefore, the pre-process is needed and the author filtered extreme values in the new feature. The boxplot after filtered is shown as Fig. 3.

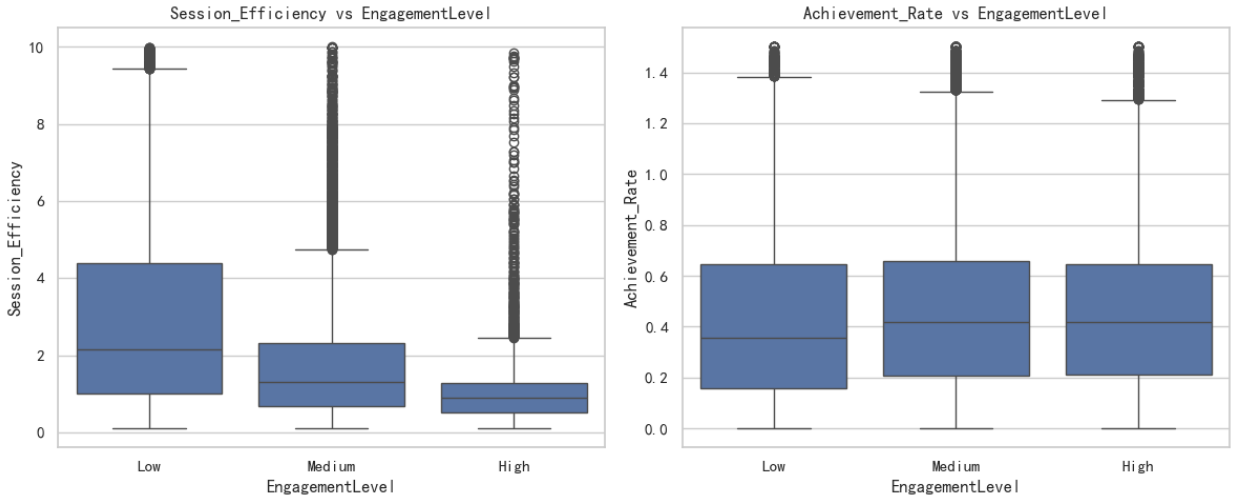


Fig 3. After filtering boxplot of relationship between session efficiency and engagement level and relationship between achievement rate and engagement level (Picture credit: Original)

In order to train the model, it is also needed to split the whole data into training sets and testing sets. So, splitting the data is essential for machine learning. In the experiment, the data is split into 80% as training sets and 20% as testing sets.

2.2. KNN

In this experiment, KNN algorithm is used in the model. According to the paper of learning KNN, the key idea of KNN is to predict data labels through the most used rule. Precisely, it uses the nearest k training data points' class to predict the testing data label [7]. Also finding the best k value of KNN during training is essential, it affects the model performance directly. For instance, there are several red and blue samples which are classified in a diagram. There is a new black sample, and it find it nearest 5 neighbors to know it belons to the red class shown in the Fig. 4.

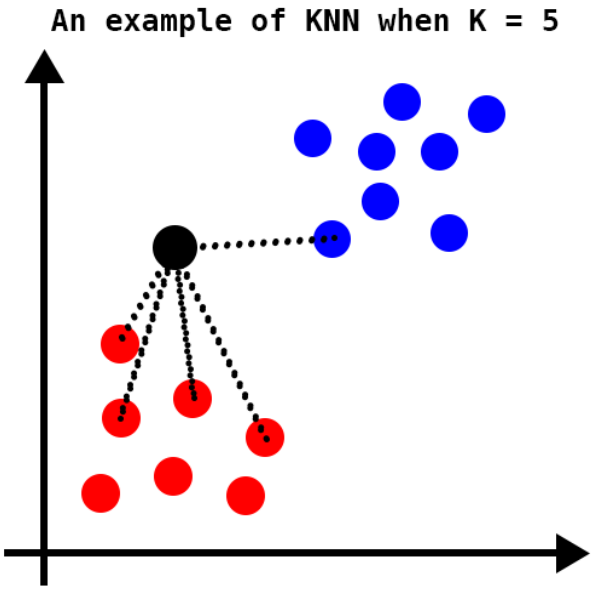


Fig 4. An example of KNN when K = 5 (Picture credit: Original)

2.3. Evaluation metrics

In order to measure the quality of the model, there are 4 metrics for evaluation in this experiment used: accuracy, precision, recall and F1-Score.

2.3.1 Accuracy

Accuracy refers to the ratio of right classified samples to the total samples, which is the most used metrics in machine learning.

2.3.2 Precision

Precision refers to ratio of right classified samples in a category to the total sample distributed to the category. It is usually indicated between [0, 1]. 1 refers prediction all right.

2.3.3 Recall

Recall refers to the ratio of the number of picked samples of the positive category to the total number of samples of the category. People dislike missing samples as possible as they like.

2.3.4 F1-Score

F1-Score refers to the mean of Precision and the recall, which means it will deal with any extreme value.

3. Results and Discussion

Because of the importance and effect of k value, picking the best k value is essential. This experiment uses grid search to get the best k value is 15. When the k value is 15, the model has the highest f1 score. After using 15 as the k value train the model, it could get the result as a table as follows:

Table 1. Core Final Report

Model	KNN
Best K	15.0000
Accuracy	0.8286
Precision	0.8501
Recall	0.7679
F1-score	0.7949

Table 2. Classification Report and Global Report

Classification Report				
	precision	recall	f1-score	support
High	0.88	0.81	0.85	1615
Low	0.87	0.55	0.68	1124
Medium	0.80	0.94	0.86	3088
		Global Report		
Accuracy		0.83		5827
Macro avg	0.85	0.77	0.79	5827
Weighted avg	0.84	0.83	0.82	5827

From the Table 1 and Table 2, it could be found that the prediction of this model of players in high engagement is the most precise at 88, which for game companies means when the company gives reward to 100 players in high engagement, there are 88 players worth to get this reward, just 12% loss. In addition, in these 1615 players who are in high engagement, it just misses layers it predicted. However, in these players who are marked as low engagement, it misses 45% of players, which may waste many resources in these players to launch retention campaign.

According to the result of the table, it can be drawn as a confusion matrix shown in Fig. 5.

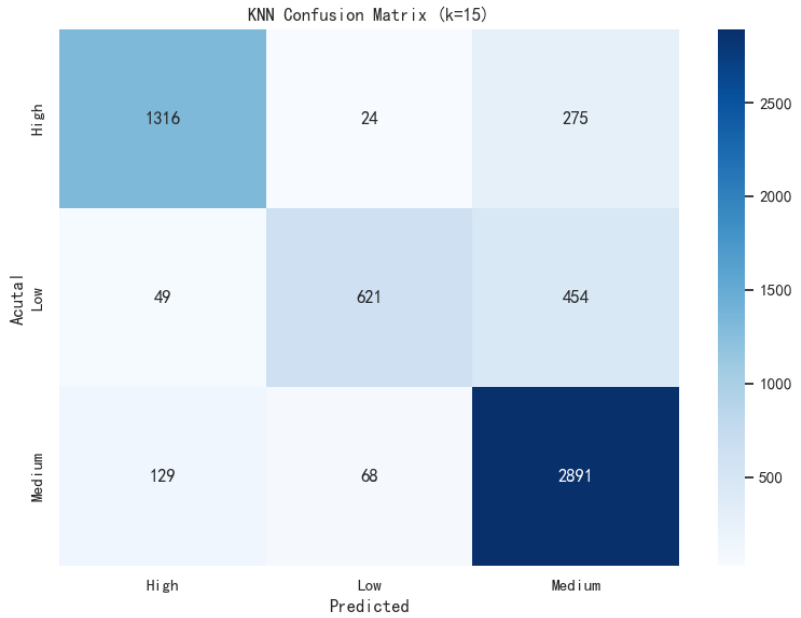


Fig 5. KNN confusion matrix (k = 15) (Picture credit: Original)

From this confusion matrix, it is clear that the performance of the model is well and correct predictions are far more than wrong ones.

There are some reasons for choosing KNN algorithm as follows. First, for game designers, results of KNN algorithm can be easy to understand and modify the strategies of the game like rewarding high engagement players. Second, there is no need to assume distributions, because the behavior of game players does not follow any standard distribution like Normal distribution. Finally, KNN is not affected by exceptions, which only needs more neighbors to affect the result.

There are also some disadvantages of KNN algorithm. The most problem is the cost of calculation, which means the players grow higher on the server, the slower result will get. However, for this static research, it does not matter. The second one is curse of dimensionality. In the high dimensional space, it might have same distance between every point which makes KNN unavailable to use. The author uses feature engineering to solve this problem and ensure every figure in the same dimension.

Among the results, the number of players who are medium engagement is too large. Therefore, it could be modified as using KNN to predict players who are in high or low engagement and using other models to predict medium engagement players and classify them into more categories.

4. Conclusion

In this experiment, the prediction model predicts player engagement based on KNN algorithm could recognize players who are in high recall rate in 81%, which means it just loss 19 players per 100 plyers in high engagement. Though session efficiency and achievement rate feature engineering, this experiment makes KNN break the limit of the original dataset. In the future work, the model could combine with these basic data and online behaviors such as chat rate and party rate to build a dynamic prediction system and explore the faster algorithm to modify this model. When AI could not only limit in predict player behavior but also create real profits, the target of technology help production could come true.

References

[1] Rykała P. The growth of the gaming industry in the context of creative industries. Biblioteka Regionalisty, 2020 (20): 131

[2] Turing A M. Computing machinery and intelligence. Parsing the Turing test: Philosophical and methodological issues in the quest for the thinking computer. Dordrecht: Springer Netherlands, 2007: 23-65.

[3] Russell S, Norvig P. Artificial Intelligence: a modern approach, 4th US ed. URL: https://aima. cs. berkeley. edu/(дата обращения: 26.02. 2023), 2021.

[4] Peng L. Research on Machine Learning Models for Predicting Player Churn. 2025.

[5] Kaggle. Predict Online Gaming Behavior Dataset

[6] https://www.kaggle.com/datasets/rabieelkharoua/predict-online-gaming-behavior-dataset/data, 2024

[7] Olaoye G, Fajinmi J. Role of Feature Engineering in Improving Machine Learning Predictions of Diabetes Mellitus in Healthcare Data. 2025.

[8] Zhang S, Li X, Zong M, et al. Learning k for knn classification. ACM Transactions on Intelligent Systems and Technology (TIST), 2017, 8(3): 1-19.