How can we increase revenue from Catch the Pink Flamingo?

JUNA LUZI

Problem Statement

To help Eglence identify new revenue opportunities in their “Catch the Pink Flamingo” game we need to use the data Eglence collects to understand various aspects of user behavior.

In this analysis we will focus on the following data source:

- *chat-data* directory
- *flamingo-data* directory
- *combined-data.csv* file

To help Eglence identify new revenue opportunities in their “Catch the Pink Flamingo” game we need to use the data Eglence collects to understand various aspects of user behavior.

In this analysis we will focus on the following data source:

- *chat-data* directory which contains 6 csv files and will be used to create the graph model and implement graph analytics in Neo4j
- *flamingo-data* directory which contains 8 csv files which contain information on ad-clicks, buy-clicks, game-clicks, level-events, team-assignments, team and user-session; we will explore this data in Splunk
- *combined-data.csv* file which we will use in KNIME to run a decision tree model to classify users as high spenders or low spenders

The purpose of this report is to utilize Eglence user data to produce better models; therefore make higher prediction results so we can increase Eglence’s revenue. The real value of this data comes from the integration of the diverse data streams and analysis at scale. This will drive our ability to gain new insights.

Data Exploration Overview

Figure 1. Most purchased items

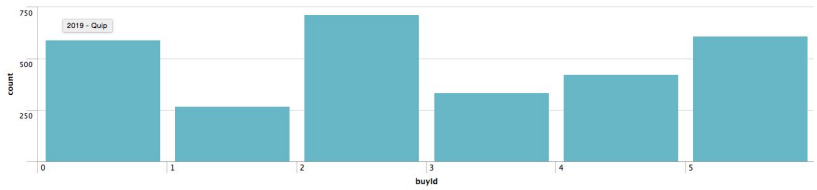
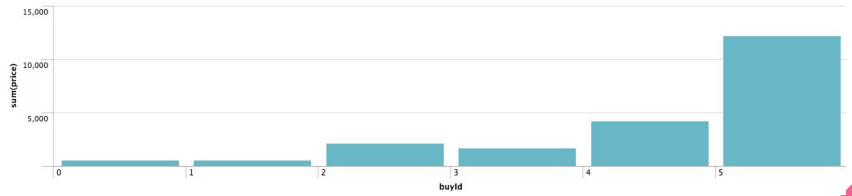


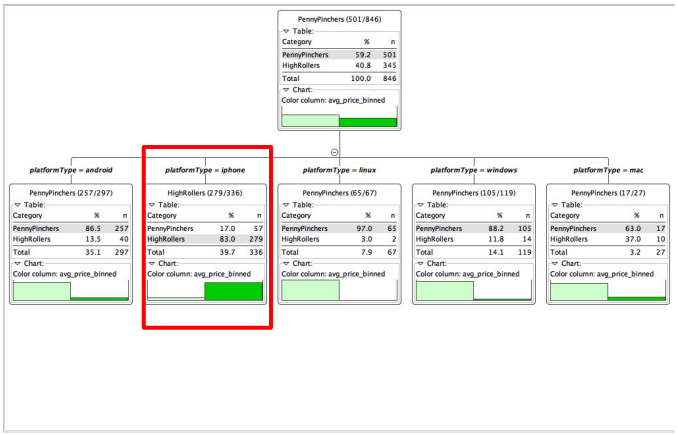
Figure 2. Revenues by item



Before we analyze the datasets it is helpful to look at the data to get a general sense of what the data looks like.
Figure 1 shows which items are popular and which items are not. It is evident from the image that item 2 is the most popular as it is the most purchased item while item 1 is the least popular purchased item.

Next, we look at how much revenue each item brought in through the image shown in Figure 2. It is evident from the image that item 5 brought the most revenue while items 0 and 1 brought in the least amount of revenue. We can infer from both images that while item 5 is the second most purchased item, it also brought the most revenue. Lastly, it is interesting to see that the most purchased item, item 2, brought very little revenue compared to other items.

What have we learned from classification?



PennyPinchers are more likely to be users on platforms such as such as android, linux, windows and mac.

HighRollers are more likely to be iPhone users.

As our first analysis, we used KNIME to run a decision tree algorithm to classify the data in the *combined-data.csv* file to identify which users would be big spenders (HighRollers) or low spenders (PennyPinchers).

HighRollers are users who purchase items that cost more than $5.0, while PennyPinchers are users who purchase items that cost $5.0 or less. We partitioned that data into a train and test datasets. The train dataset was used to create the decision tree model, which was then applied to the test dataset to make sure that our classifier is not overfitting or underfitting.

The image to the right shows the results from the decision tree. It is evident from the picture that users are grouped by *platformType*. PennyPinchers are more likely to be users on platforms such as such as android, linux, windows and mac; whereas HighRollers are more likely to be iPhone users.

What have we learned from clustering?

Three attributes used during the clustering:

- Total ad clicks per user
- Sum of money spent by users
- Number of ad categories per user

Cluster #	Total revenue	Total ad clicks	Number of ad categories	Conclusion
1	67.448	34.144	8.128	Users on the margin
2	145.511	41.066	8.488	High-level spending users
3	17.126	26.364	7.367	Low-level spending users

We then continued our analysis by clustering the data utilizing Spark MLlib and Jupyter Notebook. The three attributes I decided to cluster are:

- Total ad clicks per user - this attribute would allow us to see if there are groups of users that are high ad-click users or low ad-click users. This provides information on user ad click behavior
- Sum of money spent by users - this attribute would allow us to see if there are varying degrees on user preferences since we can capture the total cost per user
- Number of ad categories per user - this attribute would allow us to see if users prefer varying ad categories or if they focus on a few since we can capture the number of ad categories per user

From the clustering analysis we can conclude the following:

- Cluster 1 is different from the others in that they are not necessarily the highest or lowest spenders from the two clusters, their total ad clicks fall in between the ranges of clusters 2 and 3, and the number of ad categories that they also click on is between the ranges of the other two clusters. It seems like these users are on-the-margin users.

- Cluster 2 is different from the others in that it seems to be the cluster with users that buy the most items, have higher total ad clicks and click on a higher variety of ad categories (although the magnitude is not substantially different from the other two clusters, it is still a bit higher).

- We can categorize these users are high spending users.
- Cluster 3 is different from the others in that the total revenue, total ad clicks per user and number of ad categories per user are all less than the rest of the cluster; which indicates that these users are low level spending users. Their ad-click ratio is smaller than the rest of the users and their number of ad categories is less than the other users -although not by much - while it looks like the cost is a lot less for cluster 3 users than the other two clusters.

From our chat graph analysis, what further exploration should we undertake?

Top 10 chattiest users

u.id
394
2067
209
1087
554
516
1627
999
668
461

Top 10 chattiest teams

t.id
82
185
112
18
194
129
52
136
146
81

u.id	t.id
394	63
2067	7
209	7
1087	77
554	181
999	52
516	7
1627	7
461	104
668	89

Lastly, to gain more insights from the datasets we used the *chat-data* files to create a graph from chat data utilizing Neo4j.

- We found out that the longest conversation chain in the dataset consisted of 10 chats and 9 conversations chains. Furthermore, we found out that there were only 5 unique users in the longest conversation chat.
- We also analyzed the relationship between the top 10 chattiest users and top 10 chatties teams. We were able to conclude that only one user (user with id 999 who is part of the team with id 52) is one of the top 10 chattiest users who is also part of the top 10 chattiest teams.
- Lastly, by calculating a clustering coefficient for each user we were able to identify how active groups of users are. We concluded that the most active users are users with id 209, 554, 1087 with corresponding clustering coefficients 0.9523, 0.9047, 0.8.

Recommendation

1. Target iPhone users with high price items
2. Increase the number of low price items for other platform users
3. Target users in cluster 1 with promotional offers
4. Target users in cluster 2 with more diverse products

1. Target iPhone users with high price items

   The decision tree classification created in KNIME indicated that iPhone users are more likely to be HighRollers - users who spend more than $5.0 on items - therefore, one way to increase revenue could be by targeting these users with more high price items or items with higher prices.

2. Increase the number of low price items for other platform users

   The decision tree implemented in KNIME also indicated that other platform users such as Android, Linux, Windows and Mac users are more likely to be PennyPinchers - users who spend $5.0 or less on items - therefore, while the price of items can remain low, we can increase the number of items shown to these users.

3. Target users in cluster 1 with promotional offers

   The implementation of the k-means clustering algorithm in PySpark indicated that users in the first cluster with cluster center [67.448, 34.144,   8.128] should be targeted with promotional offers to bring the users to buy more items and click more ads since they are users on the margin. Figuring out what

incentivizes these users to become as “active” (high ad clickers and item buyers) users as cluster 2 users could change their behavior and eventually shift them to cluster 2 users.

4. Target users in cluster 2 with more diverse products

   The implementation of the k-means clustering algorithm in PySpark also indicated that users in the second cluster with cluster center [145.51111111, 41.06666667, 8.48888889] should be targeted with more products since they are high spenders and have higher ad-click ratio and their ad category preference is higher. Maybe diverting some of the resources from cluster 3 users to cluster 2 users could benefit more since we will be allocating resources to higher spending users.