Project Report

Group Members: Cindy Huang, Lan Bowen, Manoji Kamalumpundi

BAIS: 6100 Text Analytics

Professor Qihang Lin

Project Goals:

a. What game was the most highly recommended game
b. Find keywords that gives a game a good review
c. Find out the sentiment of the video games, whether a game was overall positive reviews, or negative reviews
d. Find out the average number of hours played for each game.

Description of Dataset:

The dataset is from (https://www.kaggle.com/luthfim/steam-reviews-dataset) that contains a list of video game reviews from the popular video game site STEAM. The size of the file is 121 MB and the license is listed as a public domain. This dataset contains reviews from STEAM's best selling games as of February 2019. The file has 434,891 rows and 8 columns. The columns included in the dataset are:

- Date_posted
- Funny - how many other players think the review is funny
- Helpful - how many other players think the review is helpful
- Hours played - how many hours a reviewer played the game before making a review
- Is early access review
- Recommendation - whether reviewer recommended the game or not
- Review - the text of user review
- Title - game title being reviewed

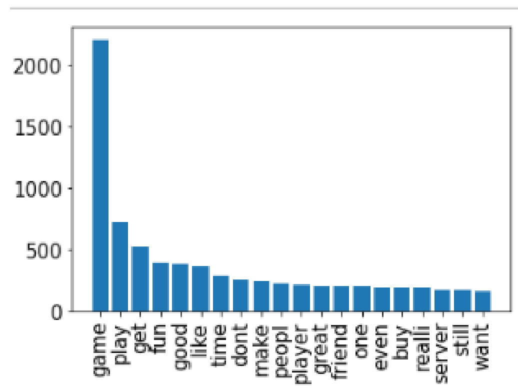
The techniques Used to create the Project Report:

1. Review data:
   - Import .csv data using encoding “utf8” and pipe separate
   - df = pd.read_csv("steam_reviews2.csv",encoding="utf8",sep='|', delimiter=None)
   - Because the dataset was so large, we took a random sample of 2000 reviews except for sentiment scores.
2. Clean up data:
   - Replace all non-English letters with empty string
   - Utilized SQL within Python to remove any rows with NaN
3. Check DTM Matrix before Text Visualization:
   - Apply stemming using the Snowball stemmer from nltk and remove stop words in the stop word list of ntlk library without changing the size of vocabulary.
   - Select the most 20 common words in the reviews. They are “game, play, get, fun, good, like, time, don’t, make, peopl, player, great, friend, one, even, buy, realli, server, still, want”. Seem like positive words from the reviews. (Answer question b in project goal)
   - The below wordcloud contains both positive and negative words from the reviews.

Text Visualization

Created a DTM in term frequencies using CountVectorizer, StemmedCountVectorizer and vectorizer.fit_transform before doing Visualization. This is shown here. Next, we created a bar chart for these most common words, which is also shown below.

	Term	Frequency
0	game	2207
1	play	722
2	get	523
3	fun	394
4	good	379
5	like	358
6	time	288
7	dont	256
8	make	243
9	peopl	225
10	player	217
11	great	206
12	friend	205
13	one	201
14	even	194
15	buy	190
16	realli	189
17	server	174
18	still	173
19	want	163



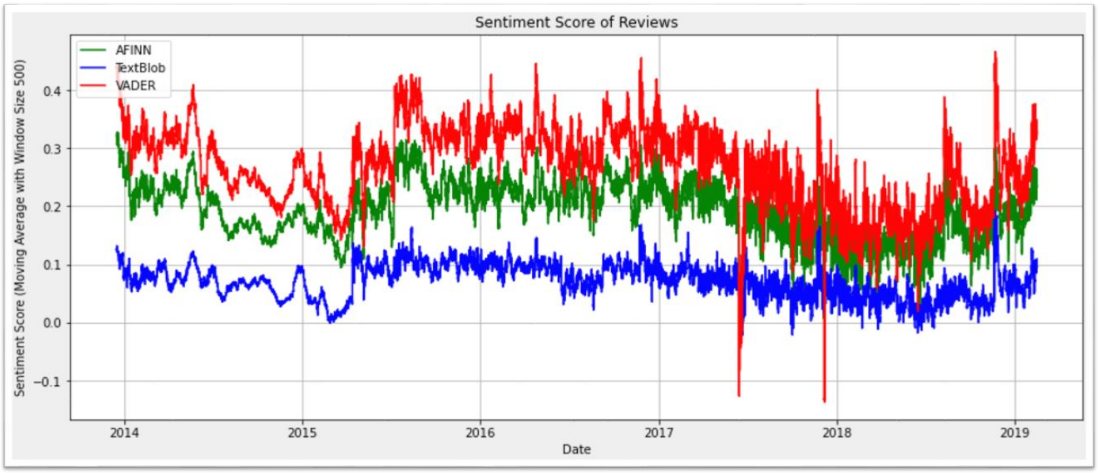
We then applied AFINN, TextBlob and VADER and generate the correlations matrix to see how the sentiment scores of these reviews.

	date_posted	funny	helpful	hour_played	is_early_access_review	recommendation	review	title	review1	AFINN	TextBlob	VADER
0	6/28/2017	0	0	243	1	Recommended	Enjoy the game a lot really recommend it to ev...	PLAYERUNKNOWN'S BATTLEGROUNDS	Enjoy the game a lot really recommend it to ev...	7.0	0.225	0.8357
1	7/27/2015	0	0	60	0	Recommended	Awesome game!	Rocket League®	Awesome game	4.0	0.300	0.6249
2	3/12/2017	0	0	2261	1	Recommended	is gud	Rust	is gud	0.0	0.000	0.0000
3	11/25/2018	0	0	189	0	Recommended	the game i waiting for.. all in 1 game.. mcm n...	MONSTER HUNTER: WORLD	the game i waiting forall in gamemcm neslo	0.0	-0.400	0.0000
4	5/8/2015	0	0	324	0	Recommended	The best GTA yet. There are so many details an...	Grand Theft Auto V	The best GTA yetThere are so many details and ...	7.0	0.500	0.8176

Shown below is the correlation matrix of different sentiment scores

	funny	helpful	hour_played	is_early_access_review	AFINN	TextBlob	VADER	AFINN_scaled
funny	1.000000	0.000185	0.000129	0.000689	0.000529	-0.001842	0.001342	0.000654
helpful	0.000185	1.000000	0.006336	-0.000093	0.007271	-0.002119	0.001834	0.002512
hour_played	0.000129	0.006336	1.000000	0.039009	0.009470	0.025374	0.031621	0.015289
is_early_access_review	0.000689	-0.000093	0.039009	1.000000	0.000830	-0.001853	0.017231	0.012159
AFINN	0.000529	0.007271	0.009470	0.000830	1.000000	0.229683	0.472324	0.626004
TextBlob	-0.001842	-0.002119	0.025374	-0.001853	0.229683	1.000000	0.455876	0.419020
VADER	0.001342	0.001834	0.031621	0.017231	0.472324	0.455876	1.000000	0.778839
AFINN_scaled	0.000654	0.002512	0.015289	0.012159	0.626004	0.419020	0.778839	1.000000

To further show the sentiment scores, we created a time series of the sentiment score of reviews based on the review date. AFINN scores were scaled by dividing the score by 10. Scores that were still above 1 or below -1 were scaled to 1 or -1 respectively. Due to sheer number of reviews, a moving average of 500 reviews was used.



Summary Statistics

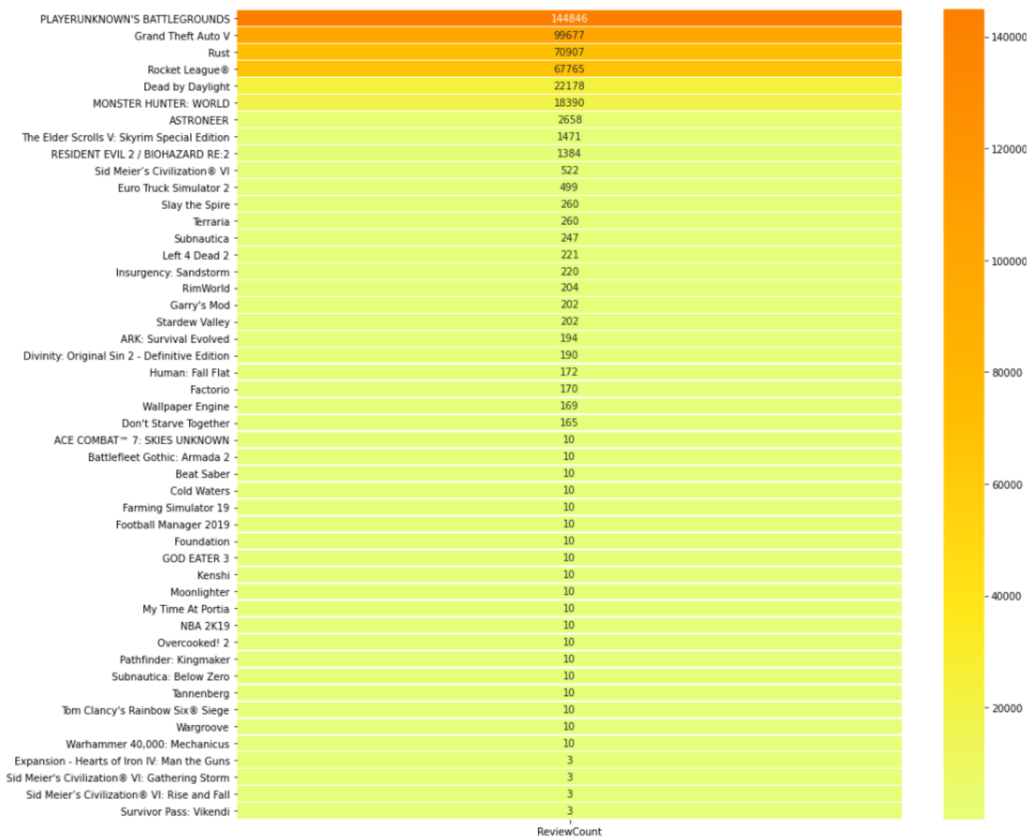
There are 2 reviews do not have recommendations. There are 633 Not Recommended and 1365 Recommendations out of 2000 reviews.

	recommendation	RecommendationCount
0	Not Recommended	633
1	Recommended	1365

The Recommendations Ratio was calculated by scoring reviews as 1 if they were recommended and 0 if they were not. The total of this number was calculated and was divided by the total number of reviews for that game.

title	RecommendationRatio
ASTRONEER	1.000000
RESIDENT EVIL 2 / BIOHAZARD RE:2	1.000000
Terraria	1.000000
Garry's Mod	1.000000
Euro Truck Simulator 2	1.000000
Don't Starve Together	1.000000
Factorio	1.000000
RimWorld	1.000000
Subnautica	1.000000
Wallpaper Engine	1.000000
Wargroove	1.000000
Rocket League®	0.897959
Rust	0.793510
MONSTER HUNTER: WORLD	0.779221
Dead by Daylight	0.736842
Sid Meier's Civilization® VI	0.666667
Grand Theft Auto V	0.593074
PLAYERUNKNOWN'S BATTLEGROUNDS	0.557751
Divinity: Original Sin 2 - Definitive Edition	0.500000
The Elder Scrolls V: Skyrim Special Edition	0.400000
ARK: Survival Evolved	0.000000

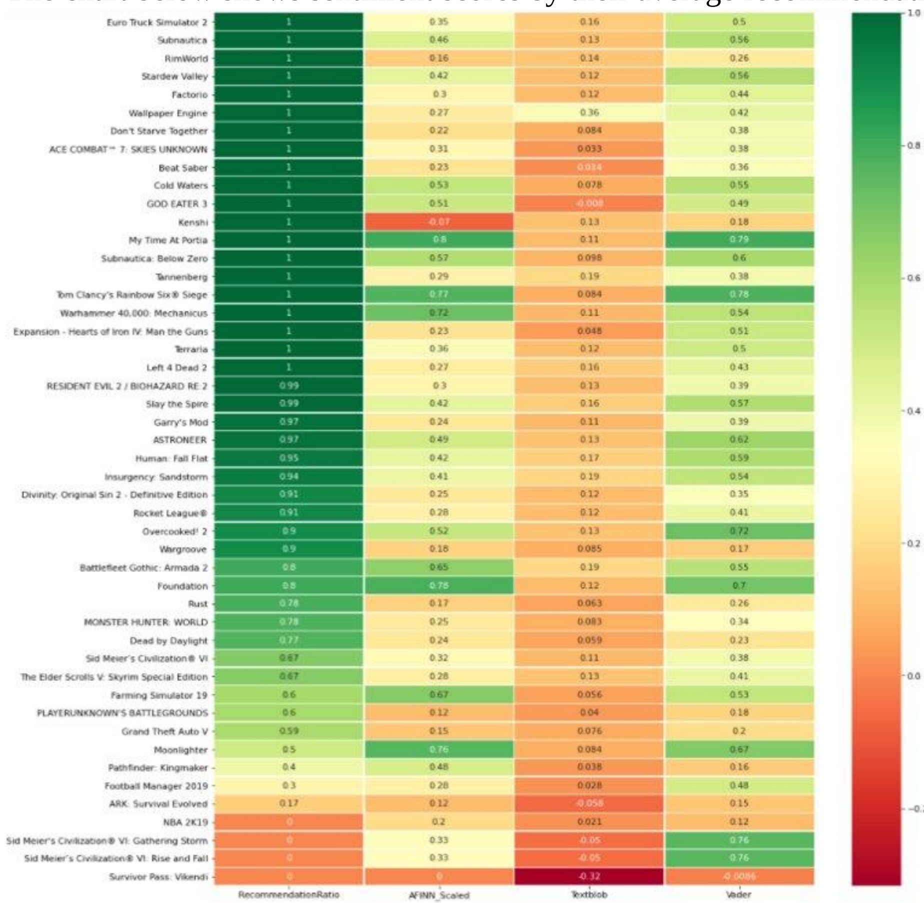
Use seaborn to create a heatmap for the number of reviews in each game. Note for the heat maps below, the complete dataset was used. Games with fewer reviews are more varied in their overall sentiment scores and didn't present consistent patterns.



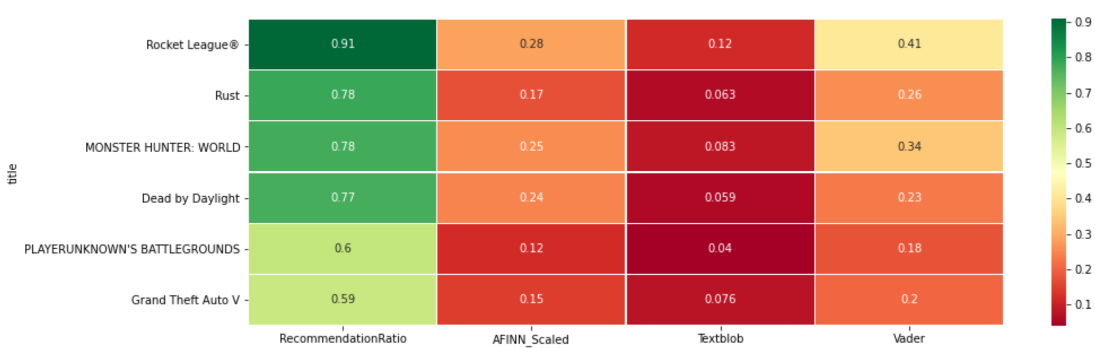
Sentiment Score (Answer question c in the project goal)

Found out the sentiment of the video games, whether a game was overall positive reviews, or negative reviews:

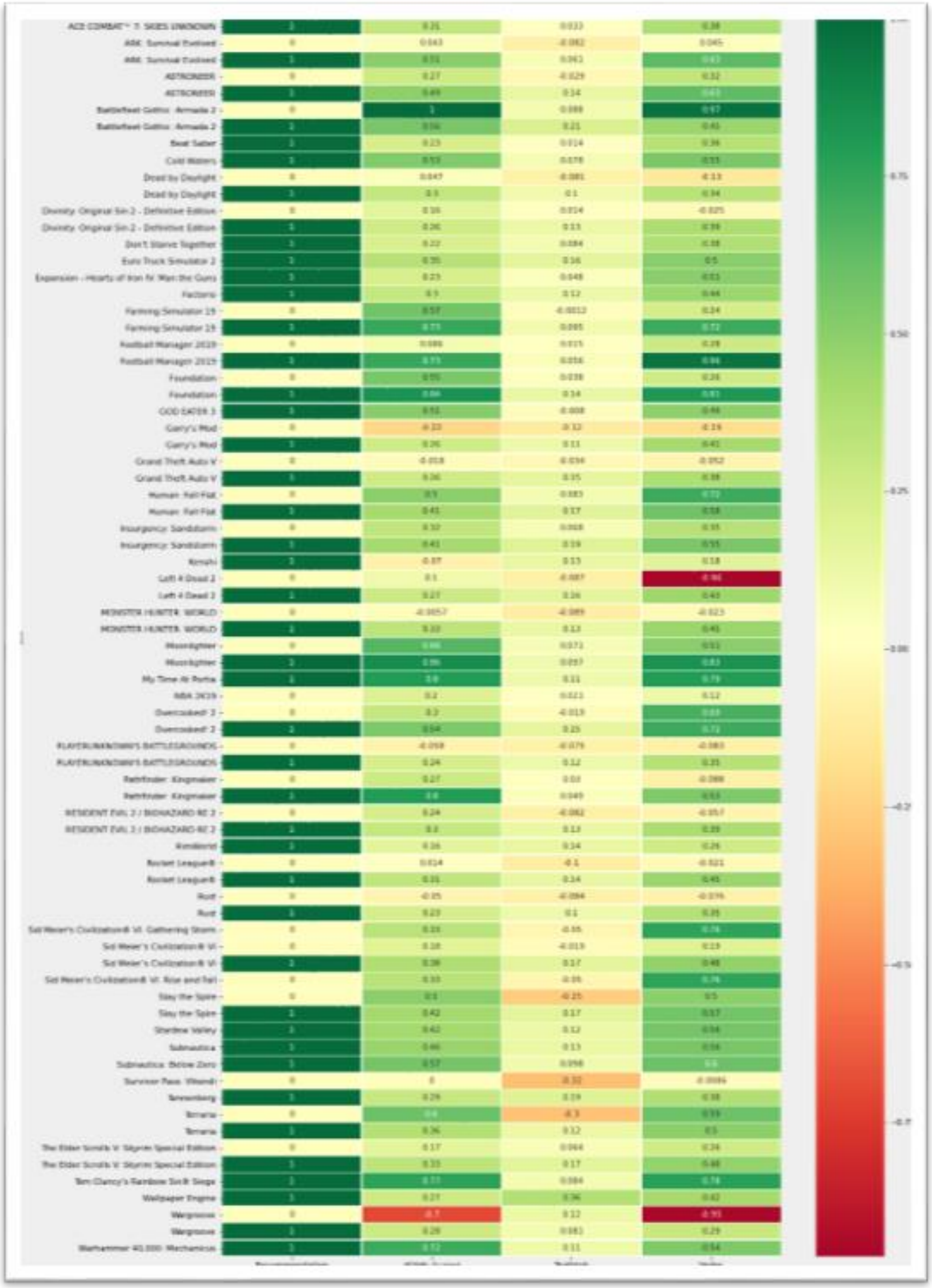
The chart below shows sentiment scores by their average recommendation ratio.



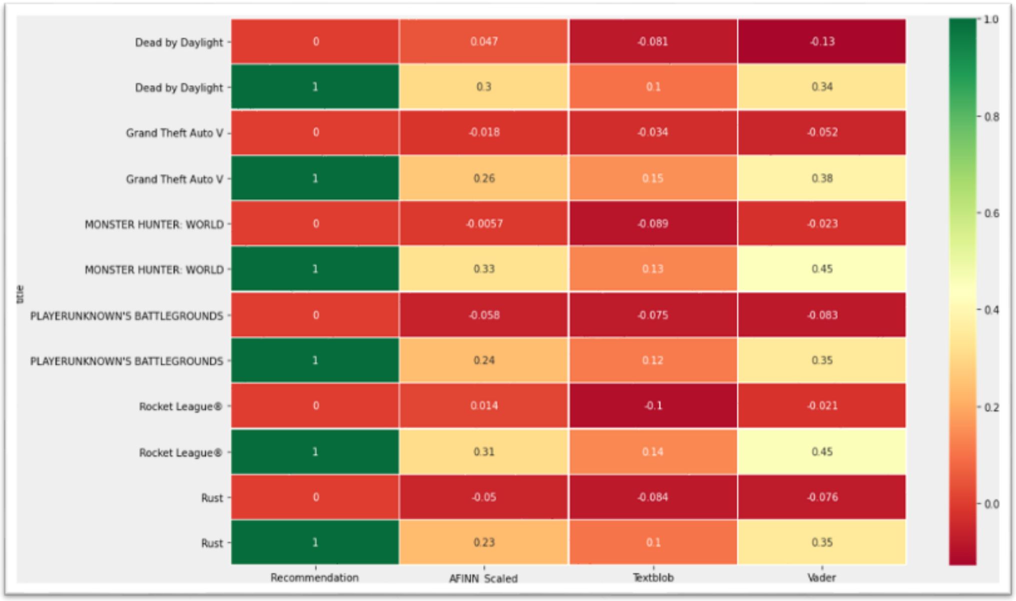
The secondary chart below only shows the top 6 games by their average recommendation ratio. A more consistent pattern can be observed between recommendation rating and sentiment score.



The chart below shows sentiment scores by their recommendation rating:



The secondary chart below only shows the top 6 games by their recommendation rating. A more consistent pattern can be observed between recommendation rating and sentiment score.



We analyzed sentiment scores to see if the content of the review agreed with the overall recommendation of the reviewer. This is important because the recommendation attribute on Steam is a binary variable; we want to quantify how good a review is. Once sentiment scores were generated on the cleaned dataset, we generated a correlation matrix with the metrics. Overall, there was no significant correlation between the sentiment scores and where or not people found the review funny, helpful, or the amount of play time the reviewer had. In terms of the sentiment scores themselves, VADER had the highest total correlation scores; however, in terms of distinguishing if a review was positive or negative Textblob had the highest resolution. ANFINN was the worst of the three-scoring methods as it marked too many reviews as having a positive sentiment.

From the heatmaps, it is easy to distinguish between reviews that had a recommend vs not recommend rating from higher/lower sentiment scores for the most part. The exceptions were mostly observed in cases where there were fewer reviews. Because Steam has a binary variable, the threshold for someone to recommend might be 80% compared to 50% for someone else. With a larger number of reviews this disparity is downed out. Games like Battlefleet Gothic: Armada 2 where the not recommended sentiment was higher than the actual sentiment reflect this, in that review the not recommended viewer praised how beautiful the game was, but they didn't like the playstyle of the game. The positive reviews weren't as eloquent in describing why they liked the game and had a lower sentiment score than the not recommend review.

Found out the average number of hours played for each game: (Answer question d in the project goal)

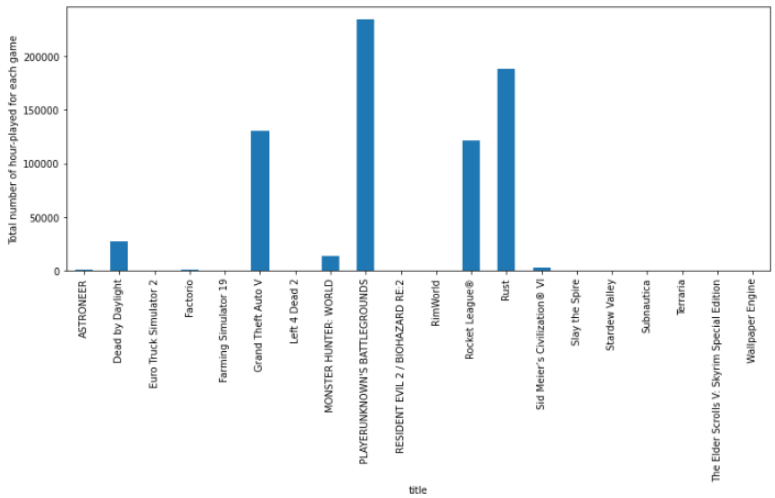
title	count	mean	max	sum
PLAYERUNKNOWN'S BATTLEGROUNDS	686.0	341.169096	2570.0	234042
Rust	327.0	574.296636	7196.0	187795
Grand Theft Auto V	429.0	303.312354	4532.0	130121
Rocket League®	316.0	383.101266	4289.0	121060
Dead by Daylight	99.0	277.898990	2104.0	27512
MONSTER HUNTER: WORLD	86.0	156.581395	1394.0	13466
Sid Meier's Civilization® VI	3.0	814.000000	2010.0	2442
ASTRONEER	23.0	51.478261	183.0	1184
Factorio	2.0	590.000000	1071.0	1180
RimWorld	1.0	422.000000	422.0	422
The Elder Scrolls V: Skyrim Special Edition	7.0	52.000000	140.0	364
Euro Truck Simulator 2	1.0	192.000000	192.0	192
RESIDENT EVIL 2 / BIOHAZARD RE:2	6.0	31.666667	80.0	190
Left 4 Dead 2	3.0	59.000000	152.0	177
Stardew Valley	1.0	146.000000	146.0	146
Terraria	1.0	120.000000	120.0	120
Wallpaper Engine	1.0	111.000000	111.0	111
Subnautica	1.0	65.000000	65.0	65
Farming Simulator 19	1.0	12.000000	12.0	12
Slay the Spire	1.0	11.000000	11.0	11

The Player Unknown Battlegrounds has the largest number of reviews which is 686 out of 2000 sample. That is over 34.3% of 21 games reviews. There are 424 reviewers that played the game and recommended. The total number of hour-played by reviewers is 234,042 hours. One reviewer has played 2,470 hours. The average hour-played is 341.2. It is the most popular game and prediction is number 1 sale and it was the most highly recommended game. (Answer question a in project goal)

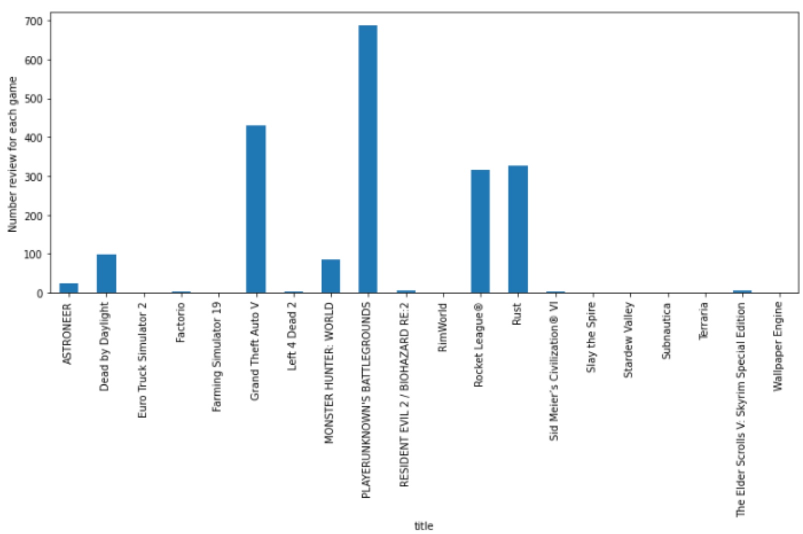
The next best game is Grand Theft Auto V which had 429 reviews out of 2000 sample. That is over 21.45% of 21 games. There are 269 reviewers that recommended the game. The total number of hour-played by reviewers were 130,121 hours. The average hour-played was 303.3 hours. One reviewer has played 4,532 hours. This game was predicted to be the second popular.

title	Recommendation	TotalReviews
PLAYERUNKNOWN'S BATTLEGROUNDS	424	686
Grand Theft Auto V	269	429
Rust	251	327
Rocket League®	287	316
Dead by Daylight	76	99
MONSTER HUNTER: WORLD	68	86
ASTRONEER	23	23
The Elder Scrolls V: Skyrim Special Edition	6	7
RESIDENT EVIL 2 / BIOHAZARD RE:2	6	6
Left 4 Dead 2	3	3
Sid Meier's Civilization® VI	3	3
Factorio	2	2
Euro Truck Simulator 2	1	1
Farming Simulator 19	0	1
RimWorld	1	1
Slay the Spire	1	1
Stardew Valley	1	1
Subnautica	1	1
Terraria	1	1
Wallpaper Engine	1	1

Total number of hour-played for each game



Number reviews for each game:



Topic Model

Next, we wanted to sort the terms into clusters to see if we can get any patterns out of the reviews. In order to do that, we utilized the topic model method. We set the number of topics to 5 by following the “elbow method”. We then summarized the topics into categories and those categories are shown in the graph below.

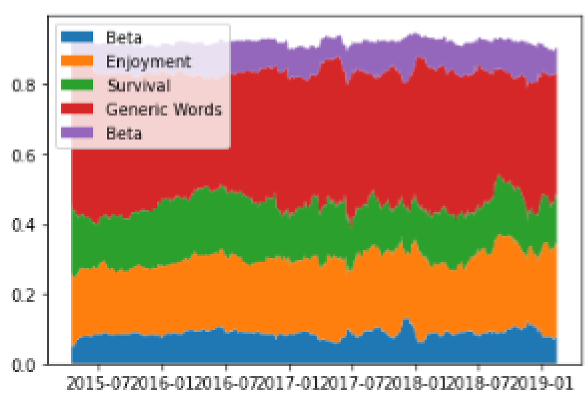
	0	1	2	3	4
0	best	game	game	game	best
1	worth	better	player	like	worth
2	time	server	youll	just	time
3	play	rock	food	ing	play
4	rock	play	cloth	great	rock
5	great	awesom	resourc	surviv	great
6	beta	want	surviv	build	beta
7	amaz	overal	best	definet	amaz
8	game	good	play	log	game
9	like	far	rock	fallout	like

Topic 0 and Topic 4 can be summarized as talking about the beta version of the games. This means they review the game before the game is officially released.

Topic 1 has words that are overall positive and can be summarized as words of enjoyment.

Topic 2 can be categorized with words dealing with survival type games. This is hinted at from words such as ‘cloth’, ‘resource’, surviv’ , etc.

Topic 3 can be categorized into generic words.



Prediction Model

In running a prediction model, we decided to use the decision tree model. We used the XGBoost package from python to build our model. From our model, we found the 10 most important terms to indicate how useful the term was in the construction of the decision tree. Those terms are shown here:

	Term	Importance
0	two	0.019010
1	money	0.017209
2	issu	0.015544
3	state	0.015502
4	server	0.015458
5	ing	0.014664
6	pc	0.014366
7	help	0.013489
8	wors	0.013193
9	best	0.013075

After that, we then took a look at evaluation tools to evaluate our model. Our model had a accuracy score of 0.9463 and a area under the cure (AUC) score of 0.9835. Lastly, the confusion matrix was shown as followed:

```
[[334  68]
 [  7 987]]
```