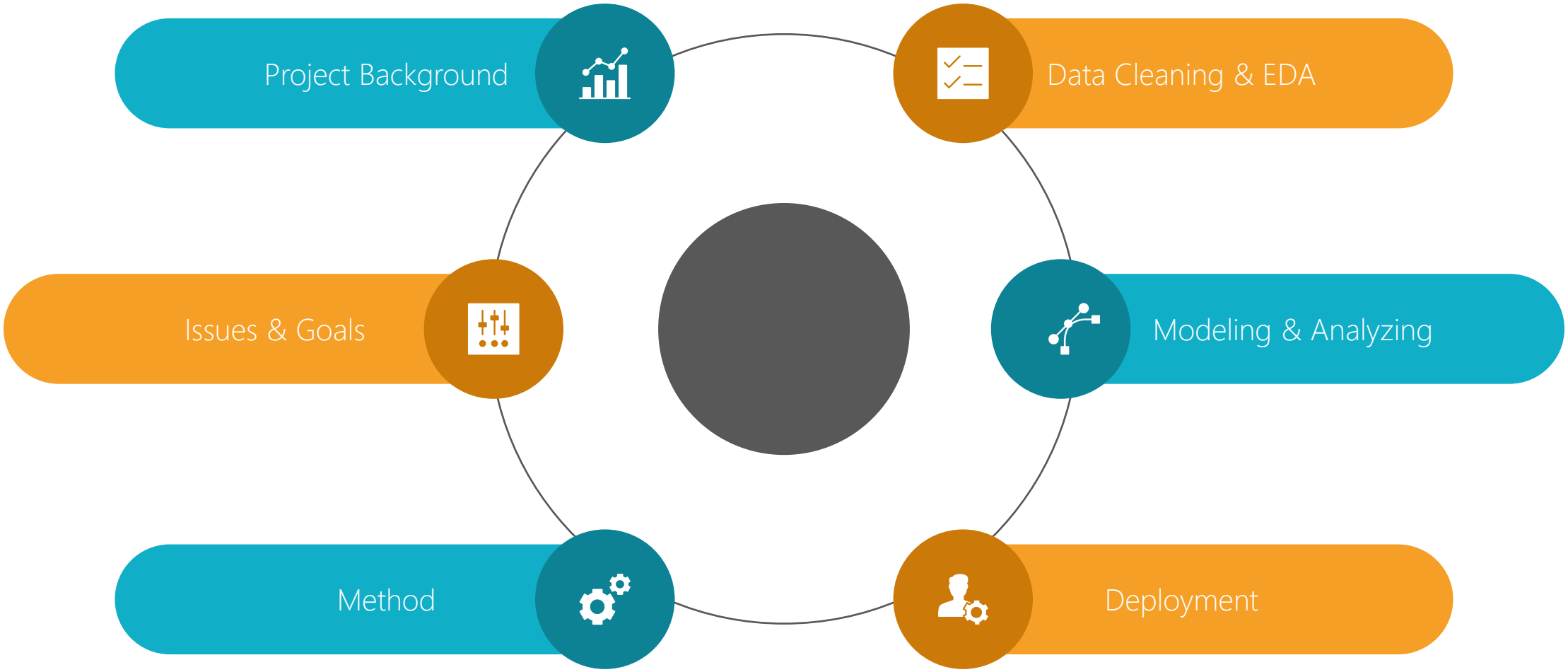
Analyzing Fall Guys Review in Steam

Laura Lesmana

Project Overview & Background

Overview


Project Background
Data Cleaning & EDA
Issues & Goals
Modeling & Analyzing
Method
Deployment

Background

- This project will analyze the sentiments of review and users' opinion for 'Fall Guys' game
- Fall Guys is developed by Mediatonic, a British video game developer based in London, established in 2015 with current staff of 230
- Fall Guys is released on 4 August 2020 for PC and PlayStation 4
- The game unexpectedly gained its popularity too fast, in the first week of its release, the servers were down due to million of players were playing it

MEDIATONIC

What is Fall Guys?

Fall Guys is a massively multiplayer party game with up to 60 players online. Player will play various random stages (usually 4 to 5), if they cannot pass it they will get eliminated. In the end, one player will remain to be the winner.

Stage 1

Stage 2

Stage 3

Final Round

Fall Guys Sales

August 26, 2020

PC

7 Million

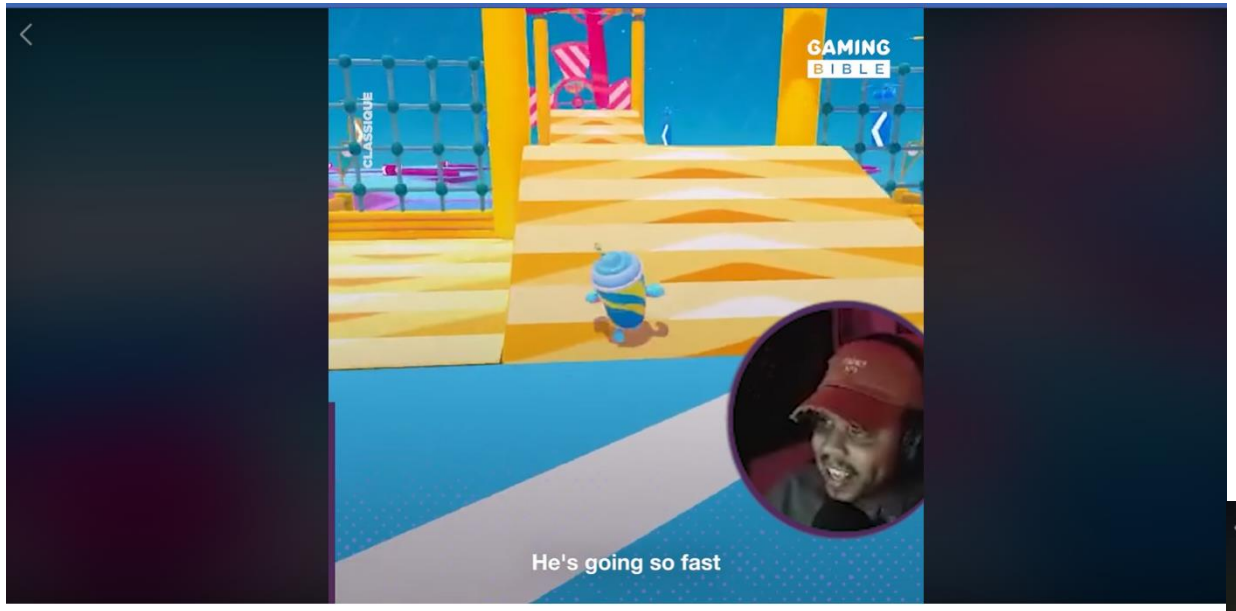
PlayStation 4

- Given free for PS Plus subscribers (there are 41 million PS Plus subscribers)
- ‘Most downloaded monthly PS Plus game of all time’

Issues for Developers

1. Mediatonic is a newly established game developers and its size also small. There are only 230 staff, compared with one of Japan's biggest developer Square Enix with 40,000 staff, or League of Legends developer, Riot Games with 2,500 staff.
    - Limited resources to maintain the user base that grows too fast
    - Unable to handle arising and running issues in the game
2. Fall Guys is just released in August, so with only two months since its release, it is a crucial period to maintain and increase the user base
3. Developing new features and maintain servers are costly
    - The cost to develop Fall Guys is estimated at $3 to $6 million

Known Issues



The World's Worst Fall Guys Hacker

GAMINGbible Follow

Like Comment Share

4.2K 1K Comments 417 Shares

IDR 108,999

US$ 19.99

X

600 Reactions +

200 Comments +

100 Shares +

Potential loss IDR 110,000,000

4K Reactions +

1K Comments +

400 Shares +

X

IDR 108,999

US$ 19.99

Potential loss IDR 436,000,000

Hackers On Fall Guys

GameByte Follow

Like Comment Share

663 265 Comments 148 Shares

Users Reactions

1. How to keep track the users reactions
2. How to know the users' sentiment in 200,000+ reviews
3. How to keep and increase the users' playtime

Project Goals & Method

Goals

1. Understand users' sentiments towards the game (positive/ negative)
2. Understand the game's brand image among the users
3. Efficiently gather users' opinion to find out possible future implementations
4. Prioritize which issues need to be handled as soon as possible in order to minimize users' uninstalling or to avoid the loss of new potential users due to game's bad issues.

Methods

1. Download data from Fall Guys' Steam review page (PC only)
2. Create topic label using LDA
3. Make unsupervised sentiment analysis model using VADER

Dataset & EDA

Dataset

Downloaded using SteamReviews, data per Sep 29

reviews

1. recommendationid - The unique id of the recommendation

author

1. steamid - the user's SteamID
2. num_games_owned - number of games owned by the user
3. num_reviews - number of reviews written by the user
4. playtime_forever - lifetime playtime tracked in this app
5. playtime_last_two_weeks - playtime tracked in the past two weeks for this app
6. playtime_at_review - playtime when the review was written
7. last_played - time for when the user last played
8. review - text of written review
9. timestamp_created - date the review was created (unix timestamp)
10. timestamp_updated - date the review was last updated (unix timestamp)
11. voted_up - true means it was a positive recommendation
12. weighted_vote_score - helpfulness score
13. steam_purchase - true if the user purchased the game on Steam
14. received_for_free - true if the user checked a box saying they got the app for free
15. written_during_early_access - true if the user posted this review while the game was in Early Access

[8]
```
df_raw.dtypes
```
```
recommendationid                 object
author_steamid                   object
author_num_games_owned            int64
author_num_reviews                int64
author_playtime_forever           int64
author_playtime_last_two_weeks    int64
author_last_played                int64
review                           object
timestamp_created                 int64
timestamp_updated                 int64
voted_up                           bool
weighted_vote_score              object
steam_purchase                     bool
received_for_free                  bool
written_during_early_access        bool
dtype: object
```

[9]
```
df_raw.shape
```
```
(14559, 15)
```

[11]
```
df_raw['steam_purchase'].value_counts()
```
```
True     12684
False     1875
Name: steam_purchase, dtype: int64
```

[12]
```
df_raw['received_for_free'].value_counts()
```
```
False    14108
True       451
Name: received_for_free, dtype: int64
```

[13]
```
df_raw['written_during_early_access'].value_counts()
```
```
False    14559
Name: written_during_early_access, dtype: int64
```

Checking NA & Change Data Types

```
df_raw['weighted_vote_score'] = df_raw['weighted_vote_score'].map(lambda x: float(x))
```

```
df_raw['author_playtime_forever']=df_raw['author_playtime_forever']/60
df_raw['author_playtime_last_two_weeks']=df_raw['author_playtime_last_two_weeks']/60
```

```
cols = ['author_playtime_forever', 'author_playtime_last_two_weeks', 'weighted_vote_score']
df_raw[cols] = df_raw[cols].round(2)
```

```
df_raw['author_last_played'] = pd.to_datetime(df_raw['author_last_played'],unit='s')
df_raw['timestamp_created'] = pd.to_datetime(df_raw['timestamp_created'],unit='s')
df_raw['timestamp_updated'] = pd.to_datetime(df_raw['timestamp_updated'],unit='s')
```

```
df_raw['timestamp_created'] = pd.to_datetime(df_raw['timestamp_created']).dt.normalize()
df_raw['timestamp_updated'] = pd.to_datetime(df_raw['timestamp_updated']).dt.normalize()
df_raw['author_last_played'] = pd.to_datetime(df_raw['author_last_played']).dt.normalize()
```

```
df_raw.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14559 entries, 0 to 14558
Data columns (total 10 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   author_num_games_owned      14559 non-null  int64
 1   author_num_reviews          14559 non-null  int64
 2   author_playtime_forever     14559 non-null  float64
 3   author_playtime_last_two_weeks  14559 non-null  float64
 4   author_last_played          14559 non-null  datetime64[ns]
 5   review                      14559 non-null  object
 6   timestamp_created           14559 non-null  datetime64[ns]
 7   timestamp_updated           14559 non-null  datetime64[ns]
 8   voted_up                    14559 non-null  bool
 9   weighted_vote_score         14559 non-null  float64
dtypes: bool(1), datetime64[ns](3), float64(3), int64(2), object(1)
memory usage: 1.0+ MB
```

EDA

```
df_raw.describe().round(2)
```

	author_num_games_owned	author_num_reviews	author_playtime_forever	author_playtime_last_two_weeks	weighted_vote_score
count	14559.00	14559.00	14559.00	14559.00	14559.00
mean	174.71	10.74	32.73	3.48	0.53
std	399.73	34.72	42.96	11.31	0.03
min	0.00	1.00	0.10	0.00	0.50
25%	22.00	2.00	10.05	0.00	0.52
50%	71.00	4.00	20.83	0.25	0.52
75%	190.00	10.00	41.76	2.43	0.53
max	14338.00	2304.00	1333.27	333.63	0.97

```
(pd.crosstab(df_raw.voted_up, df_raw.playtime_forever_cat, normalize = 'columns')*100).round(2)
```

playtime_forever_cat	low	medium	high
voted_up			
False	33.05	19.46	16.77
True	66.95	80.54	83.23

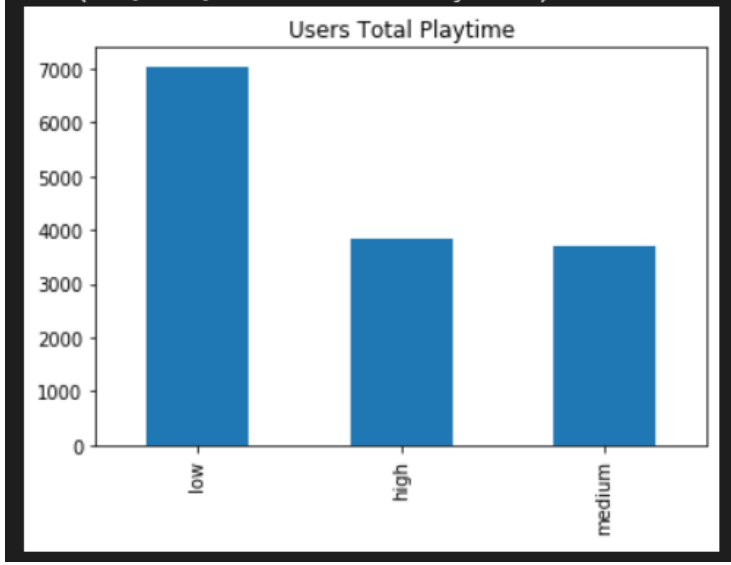
	Threshold	category
0	0.0	low
1	20.0	medium
2	40.0	high

```
df_raw['playtime_forever_cat'].value_counts()
```

```
low       7041
high      3828
medium    3690
Name: playtime_forever_cat, dtype: int64
```

```
df_raw['playtime_forever_cat'].value_counts().plot(kind= 'bar')
plt.title("Users Total Playtime")
```

```
Text(0.5, 1.0, 'Users Total Playtime')
```



EDA

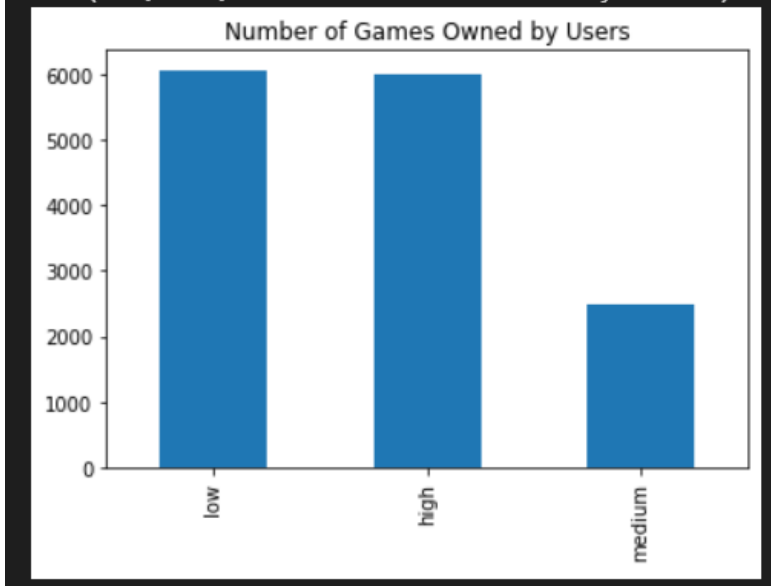
	Threshold	category
0	0	low
1	50	medium
2	100	high

```
df_raw['prod_cat'].value_counts()
```

```
low       6067
high      5996
medium    2485
Name: prod_cat, dtype: int64
```

```
df_raw['prod_cat'].value_counts().plot(kind= 'bar')
plt.title("Number of Games Owned by Users")
```

```
Text(0.5, 1.0, 'Number of Games Owned by Users')
```



[38]

```
(pd.crosstab(df_raw.voted_up, df_raw.prod_cat, normalize = 'columns')*100).round(2)
```

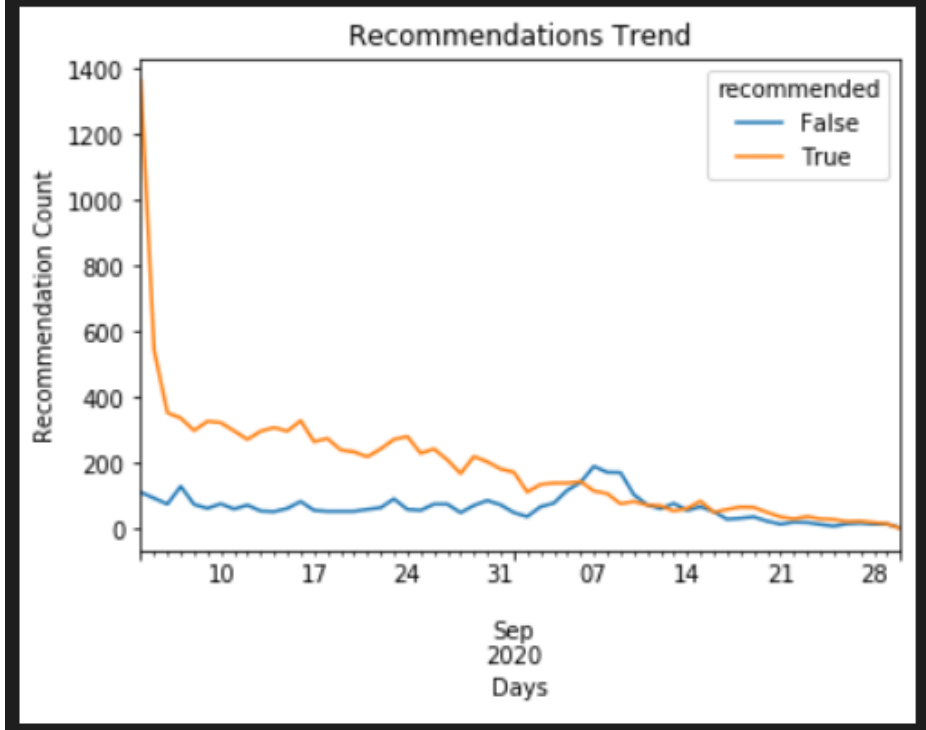
prod_cat	low	medium	high
voted_up			
False	19.85	27.0	30.15
True	80.15	73.0	69.85

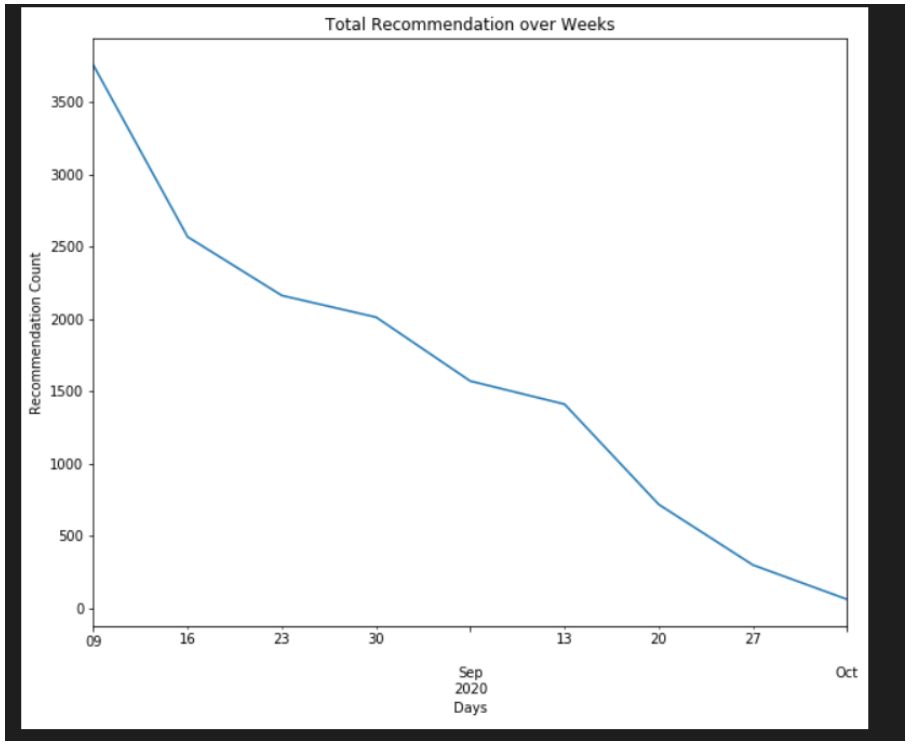
EDA

```
plt.figure(figsize=(10,8))
ct.plot.line()

plt.title('Recommendations Trend')
plt.xlabel('Days')
plt.ylabel('Recommendation Count')
plt.show()
```

```
&lt;Figure size 720x576 with 0 Axes&gt;
```






```
reviews_week = df_raw.groupby(pd.Grouper(key='timestamp_created', freq='W')).size()
reviews_week
```

```
timestamp_created
2020-08-09    3757
2020-08-16    2569
2020-08-23    2163
2020-08-30    2013
2020-09-06    1571
2020-09-13    1411
2020-09-20     717
2020-09-27     298
2020-10-04      60
Freq: W-SUN, dtype: int64
```

EDA

Take data with at least 45 minutes of total playtime for text analysis

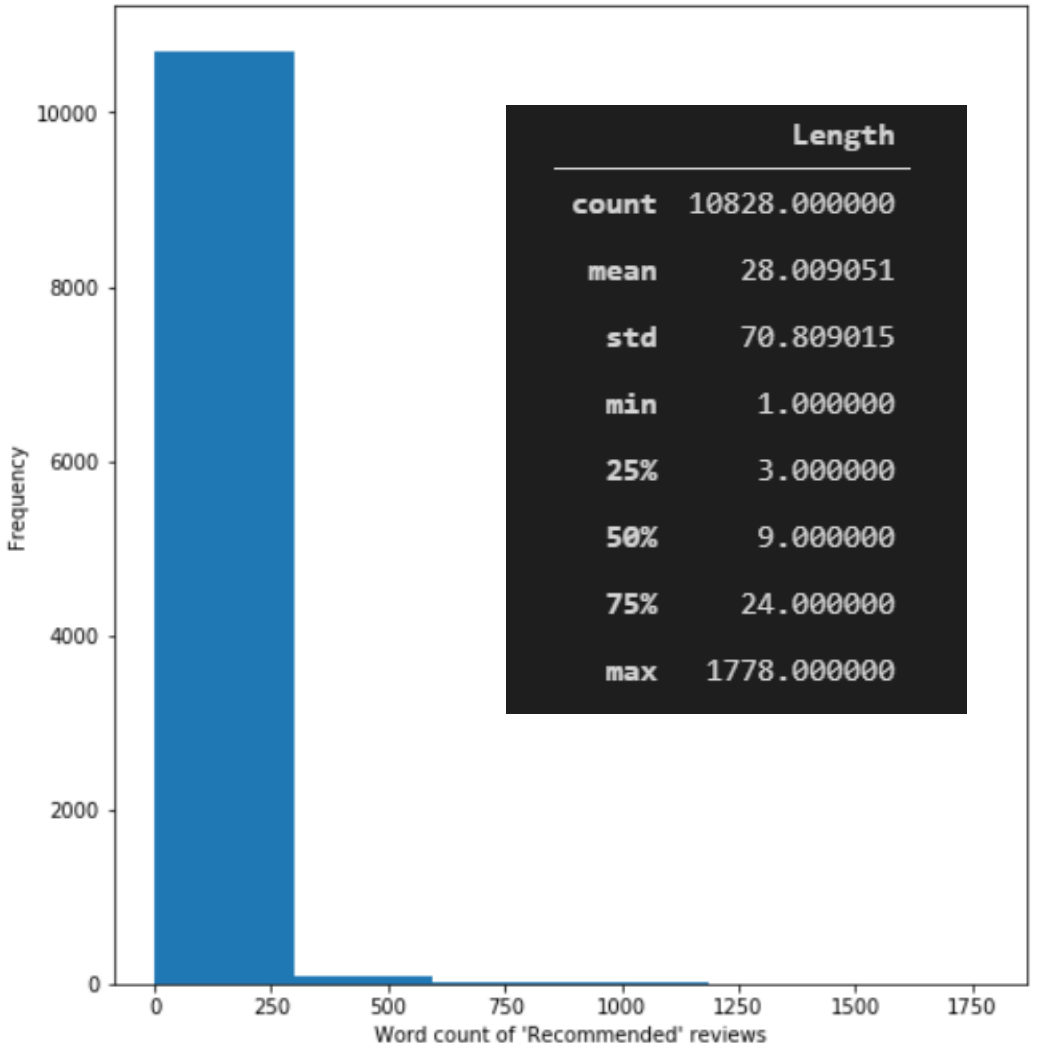
[61]

```
df_text = df_raw[(df_raw['author_playtime_forever']>=0.75)]
df_text
```

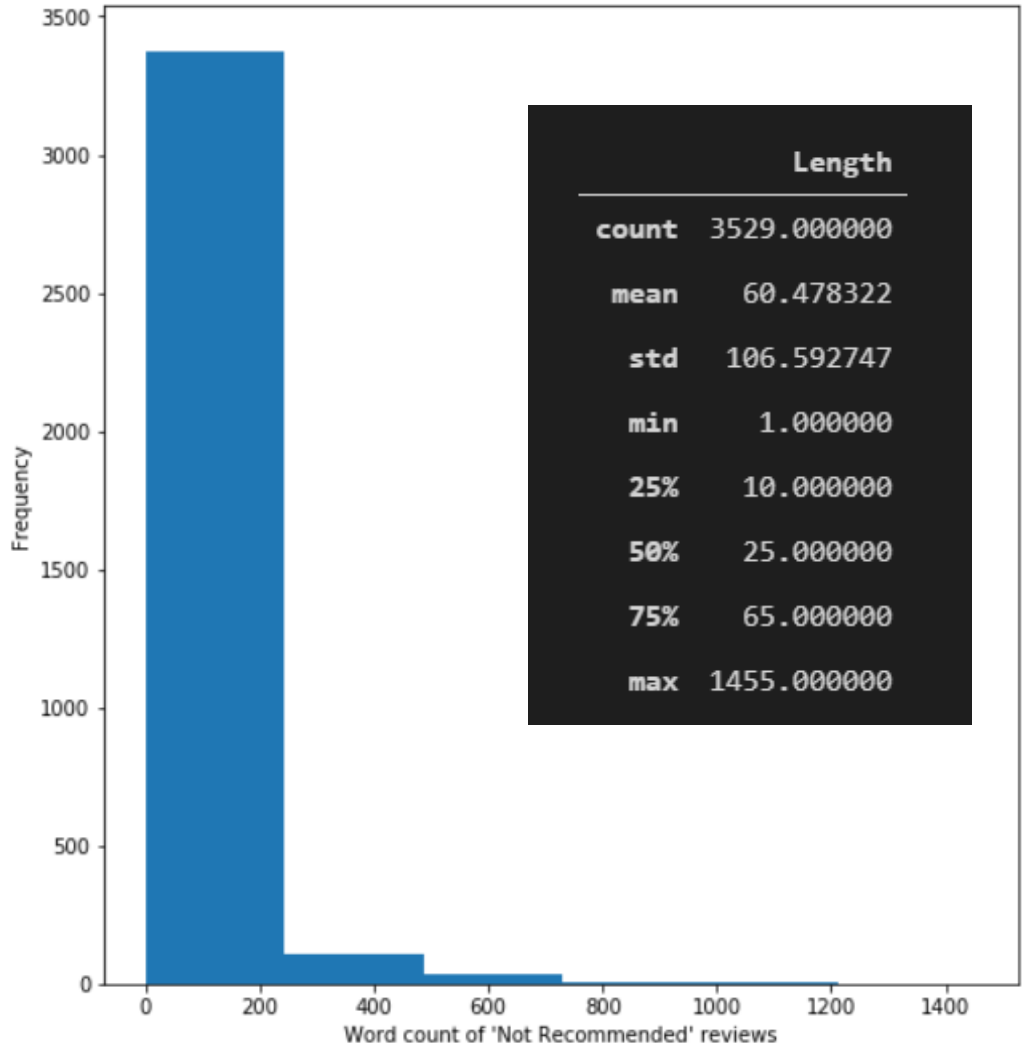
	author_num_games_owned	author_num_reviews	author_playtime_forever	author_playtime_last_two_weeks	author_last_played	review	timestamp_created	timestamp_updated	voted_up	weighted_vote_scor
0	348	34	16.18	4.12	2020-09-20	Fall Guys is a chill and fun game but it rea...	2020-08-17	2020-08-17	True	0.9
1	54	23	123.25	13.65	2020-09-28	I gave a little Hug to a player, he stared bac...	2020-08-07	2020-08-07	True	0.9
2	5	2	29.85	0.12	2020-09-17	I'd reccomend it so much i'd buy it for you	2020-08-15	2020-08-15	True	0.9
3	74	9	24.25	0.00	2020-09-06	you can hug people	2020-08-24	2020-08-24	True	0.9
4	1094	36	6.52	0.00	2020-09-05	When I was a little boy, I liked to see the Wi...	2020-08-09	2020-08-09	True	0.9

Exploring & Cleaning Text

Exploring Text



	Length
count	10828.000000
mean	28.009051
std	70.809015
min	1.000000
25%	3.000000
50%	9.000000
75%	24.000000
max	1778.000000



	Length
count	3529.000000
mean	60.478322
std	106.592747
min	1.000000
25%	10.000000
50%	25.000000
75%	65.000000
max	1455.000000

True	0.75
False	0.25

Cleaning Text

```
def clean_text(text):
    '''Make text lowercase, remove text in square brackets, remove punctuation and remove words containing numbers.'''
    text = text.lower()
    text = re.sub(r'\[.*?\]', '', text)
    text = re.sub(r'[%s]' % re.escape(string.punctuation), '', text)
    text = re.sub(r'\w*\d\w*', '', text)
    return text

df_text['clean_1'] =  df_text['review'].apply(lambda x: clean_text(x))
```

```
df_text = df_text.replace('\n',' ', regex=True)
```

```
#categorize reviews' language using fasttext
pretrained_model = "lid.176.bin"
model = fasttext.load_model(pretrained_model)

langs = []
for sent in df_text['clean_1']:
    lang = model.predict(sent)[0]
    langs.append(str(lang)[11:13])

df_text['langs'] = langs

df_text
```

```
en    13207
de      321
es       96
fr       84
pt       78
      ...
cy        1
nd        1
bn        1
lv        1
jv        1
```

Cleaning Text

```
ob = spell_checker(df_text, "clean_1")
ob.spell_check()
```

```
Analyzing suspected errors


Total suspected errors =  21906
```

```
df_text[df_text['clean_1'].str.contains("español")]
```

	review	voted_up	clean_1
63	~ THIS REVIEW IS IN ENGLISH AND IN SPANISH ~~ ...	True	this review is in english and in spanish est...

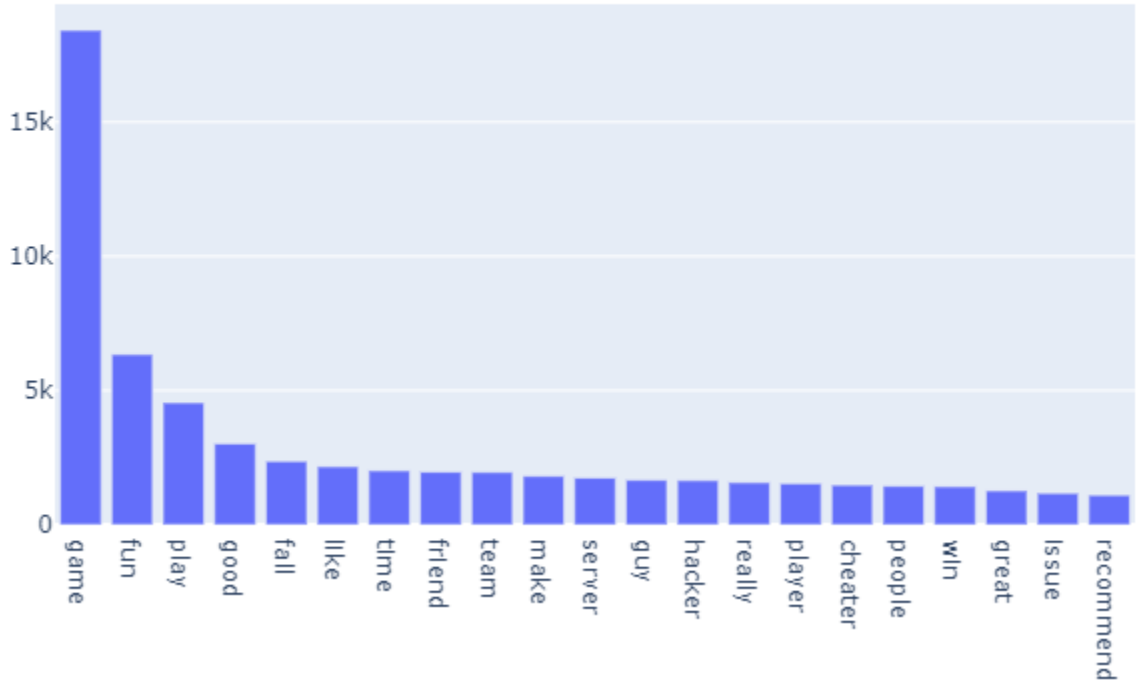
```
def remove_emoji(string):
    emoji_pattern = re.compile("["
                               u"\U0001F600-\U0001F64F"  # emoticons
                               u"\U0001F300-\U0001F5FF"  # symbols & pictographs
                               u"\U0001F680-\U0001F6FF"  # transport & map symbols
                               u"\U0001F1E0-\U0001F1FF"  # flags (iOS)
                               u"\U00002500-\U00002BEF"  # chinese char
                               u"\U00002702-\U000027B0"
                               u"\U00002702-\U000027B0"
                               u"\U000024C2-\U0001F251"
                               u"\U0001f926-\U0001f937"
                               u"\U00010000-\U0010ffff"
                               u"\u2640-\u2642"
                               u"\u2600-\u2B55"
                               u"\u200d"
                               u"\u23cf"
                               u"\u23e9"
                               u"\u231a"
                               u"\ufe0f"  # dingbats
                               u"\u3030"
                               u"\u2014"
                               u"\u2022"
                               "]+", flags=re.UNICODE)
    return emoji_pattern.sub(r'', string)
```

Cleaning Text

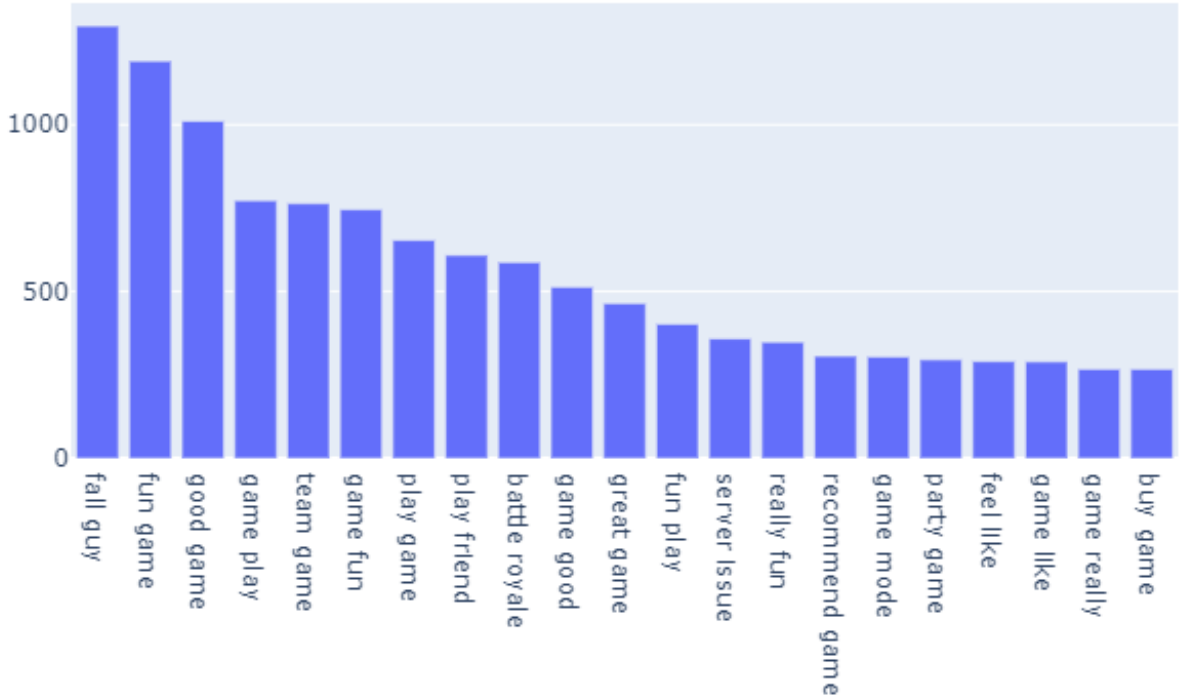
	review	voted_up	clean_1	review_lemmatize	tokenized	clean_token	clean_str
0	Fall Guys is a chill and fun game but it rea...	True	fall guys is a chill and fun game but it rea...	fall guy be a chill and fun game but real...	[fall, guy, be, a, chill, and, fun, game, but,...	[fall, guy, chill, fun, game, really, need, se...	fall guy chill fun game really need server imp...
1	I gave a little Hug to a player, he stared bac...	True	i gave a little hug to a player he stared back...	i give a little hug to a player stare back at...	[i, give, a, little, hug, to, a, player, stare...	[give, little, hug, player, stare, back, secon...	give little hug player stare back second jump ...
2	I'd reccomend it so much i'd buy it for you	True	id reccomend it so much id buy it for you	d reccomend so much d buy for	[d, reccomend, so, much, d, buy, for]	[reccomend, much, buy]	reccomend much buy
3	you can hug people	True	you can hug people	can hug people	[can, hug, people]	[hug, people]	hug people
4	When I was a little boy, I liked to see the Wi...	True	when i was a little boy i liked to see the wip...	when i be a little boy i like to see the wipeo...	[when, i, be, a, little, boy, i, like, to, see...	[little, boy, like, see, wipeout, tv, always, ...	little boy like see wipeout tv always dream ta...
...	...	...	...	...	...	...	...
13201	Well made game that i cant get enough of! Many...	True	well made game that i cant get enough of many ...	well make game that i can not get enough of ma...	[well, make, game, that, i, can, not, get, eno...	[well, make, game, get, enough, many, way, dif...	well make game get enough many way differentia...
13202	very fun and competitive especially with frien...	True	very fun and competitive especially with frien...	very fun and competitive especially with frien...	[very, fun, and, competitive, especially, with...	[fun, competitive, especially, friend, recomme...	fun competitive especially friend recommend bu...
13203	This game is extremely fun and a nice break fr...	True	this game is extremely fun and a nice break fr...	this game be extremely fun and a nice break fr...	[this, game, be, extremely, fun, and, a, nice,...	[game, extremely, fun, nice, break, usual, dro...	game extremely fun nice break usual drop loot ...
13204	Fun to play & they keep working on this game	True	fun to play they keep working on this game	fun to play keep work on this game	[fun, to, play, keep, work, on, this, game]	[fun, play, keep, work, game]	fun play keep work game
13205	[h1]The Turn Around[/h1] If you read my previo...	True	the turn around if you read my previous review...	the turn around if read previous review i ju...	[the, turn, around, if, read, previous, review...	[turn, around, read, previous, review, could, ...	turn around read previous review could recomme...

Exploring Clean Text

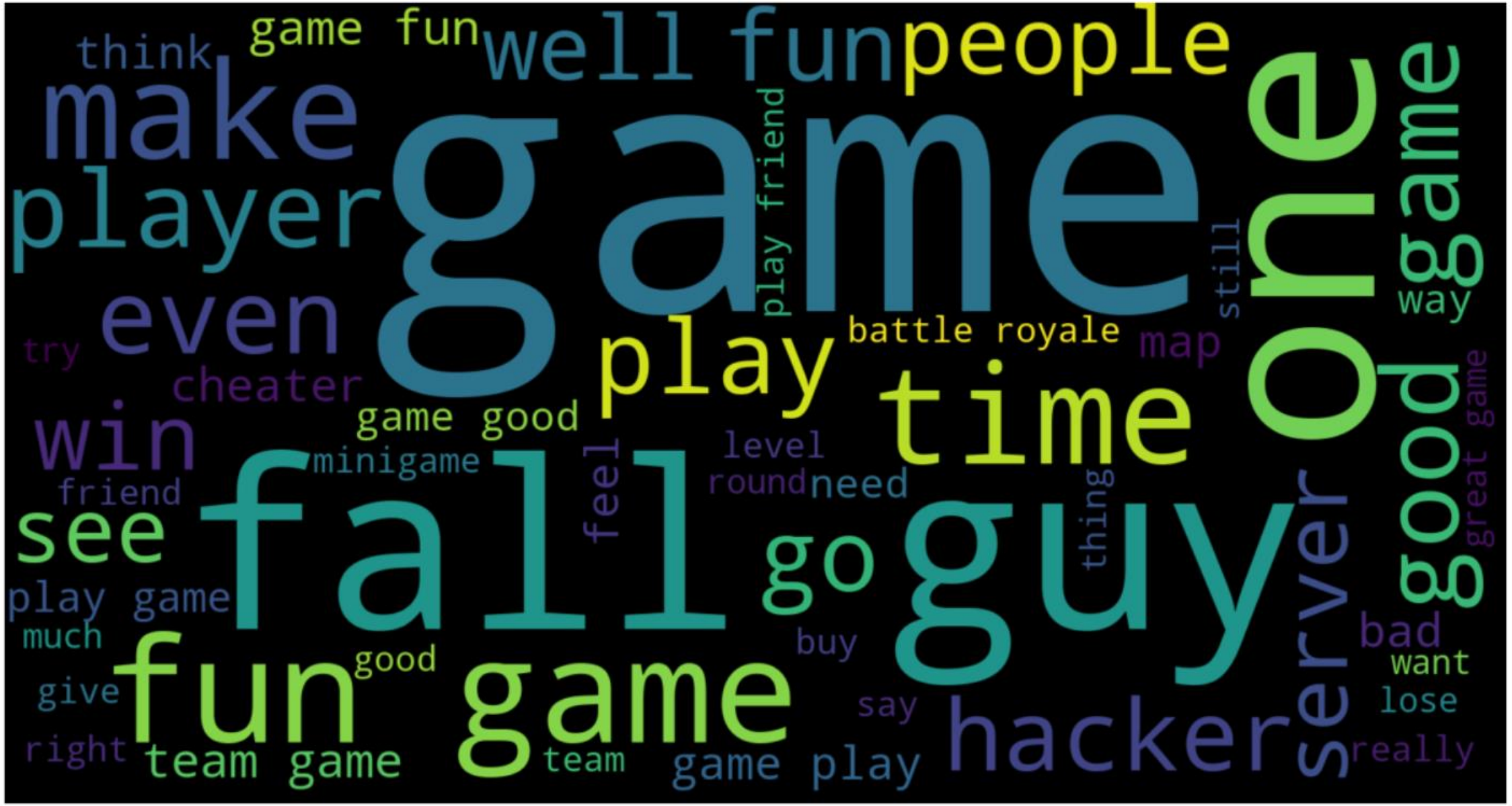
Top 20 unigrams in the review text



Top 20 bigrams in the review



Exploring Clean Text



Setting Topics

Recommended Reviews

	voted_up	clean_token	clean_str
0	True	['fall', 'guy', 'chill', 'fun', 'game', 'reall...	fall guy chill fun game really need server imp...
1	True	['give', 'little', 'hug', 'player', 'stare', '...	give little hug player stare back second jump ...
2	True	['reccomend', 'much', 'buy']	reccomend much buy
3	True	['hug', 'people']	hug people
4	True	['little', 'boy', 'like', 'see', 'wipeout', 't...	little boy like see wipeout tv always dream ta...
...	...	...	...
9779	True	['well', 'make', 'game', 'get', 'enough', 'man...	well make game get enough many way differentia...
9780	True	['fun', 'competitive', 'especially', 'friend',...	fun competitive especially friend recommend bu...
9781	True	['game', 'extremely', 'fun', 'nice', 'break', ...	game extremely fun nice break usual drop loot ...
9782	True	['fun', 'play', 'keep', 'work', 'game']	fun play keep work game
9783	True	['turn', 'around', 'read', 'previous', 'review...	turn around read previous review could recomme...

9784 rows × 3 columns

```
vectorizer = CountVectorizer(analyzer='word',
                             min_df=3,
                             stop_words=['game', 'fun', 'good'],
                             ngram_range=(2,2),
                             lowercase=True,
                             token_pattern='[a-zA-Z0-9]{3,}',
                             max_features=5000,
                            )

data_vectorized = vectorizer.fit_transform(df_yes['clean_str'])

lda_model = LatentDirichletAllocation(n_components= 6,
                                      learning_method='online',
                                      random_state= 0,
                                      n_jobs = -1
                                     )
lda_output = lda_model.fit_transform(data_vectorized)
```

Recommended Reviews

	Word 0	Word 1	Word 2	Word 3	Word 4	Word 5	Word 6	Word 7
Topic 0	server issue	negative review	launch day	yellow team	recommend anyone	review bomb	want play	wait see
Topic 1	play friend	would recommend	fall guy	battle royale	battle royal	great play	get eliminate	even well
Topic 2	battle royale	takeshis castle	mario party	jelly bean	friend play	feel like	obstacle course	get knock
Topic 3	big big	feel like	tail tag	final round	fall mountain	fall ball	slime climb	many people
Topic 4	fall guy	long time	battle royale	battle pass	get win	new map	play long	play hour
Topic 5	play friend	run run	guy fall	highly recommend	server issue	ever play	definitely worth	really play

Recommended Reviews

	topic	counts
0	server issue on launch day	4716
1	recommend to play with friend	1198
2	similar with other games and tv show	1162
3	need new map and battle pass	948
4	enjoy the game with friends but server issue	908
5	like some of the stages	852

Not Recommended Reviews

	voted_up	clean_token	clean_str
0	False	['hacker', 'destroy', 'game', 'use', 'hack', '...	hacker destroy game use hack final matchill ch...
1	False	['hacking', 'ruin', 'game', 'recommend', 'repo...	hacking ruin game recommend reporting antichea...
2	False	['update', 'dev', 'address', 'cheat', 'issue',...	update dev address cheat issue finally add eac...
3	False	['game', 'full', 'hacker', 'need', 'report', '...	game full hacker need report system something
4	False	['much', 'enjoy', 'game', 'rampant', 'hacker',...	much enjoy game rampant hacker discord remove ...
...	...	...	...
3417	False	['really', 'fun', 'first', 'time', 'play', 'ge...	really fun first time play get redundant annoy...
3418	False	['start', 'game']	start game
3419	False	['buy', 'game', 'day', 'release', 'wait', 'rel...	buy game day release wait release whole hour t...
3420	False	['really', 'fun', 'start', 'griefer', 'hacker'...	really fun start griefer hacker infest game pl...
3421	False	['fun', 'first', 'hour', 'gameplay', 'however'...	fun first hour gameplay however basic issue co...

3422 rows × 3 columns

```
vectorizer = CountVectorizer(analyzer='word',
                             min_df=3,
                             stop_words=['game'],
                             ngram_range=(2,2),
                             lowercase=True,
                             token_pattern='[a-zA-Z0-9]{3,}',
                             max_features=2000,
                            )

data_vectorized = vectorizer.fit_transform(df_no['clean_str'])

lda_model = LatentDirichletAllocation(n_components= 6,
                                      learning_method='online',
                                      random_state=0,
                                      n_jobs = -1
                                     )
lda_output = lda_model.fit_transform(data_vectorized)
```

Not Recommended Reviews

topic_theme	will change review if there is anti cheat	need cheater report system	remove team game	fun to play but server issue	hackers ruin the game	stages get repetitive and boring
Word 0	anti cheat	many cheater	final round	fall guy	many hacker	server issue
Word 1	really fun	cheater cheater	tail tag	feel like	full hacker	fun get
Word 2	way report	every match	remove team	play friend	hacker every	get repetitive
Word 3	fly around	almost every	get rid	battle royale	waste money	fall mountain
Word 4	worth price	report system	fun first	current state	cheater every	get boring
Word 5	chance win	even get	connection issue	fun play	every single	play hour
Word 6	full cheater	positive review	seem like	fun friend	report hacker	cheater ruin
Word 7	change review	get refund	team base	server issue	would recommend	get eliminate

Not Recommended Reviews

	topic	counts
0	will change review if there is anti cheat	1091
1	fun to play but server issue	556
2	hackers ruin the game	468
3	need cheater report system	459
4	remove team game	426
5	stages get repetitive and boring	395

Sentiment Modeling

Unsupervised Modeling

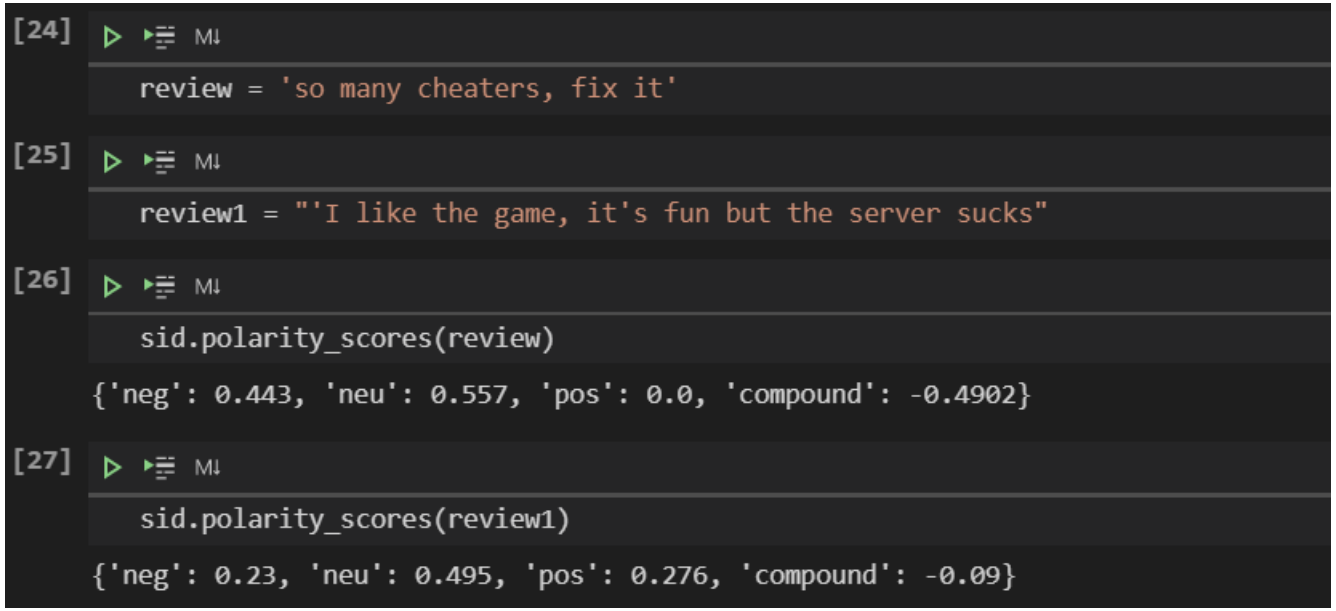
- Steam Reviews does not provide scale (1-5) for reviews, they only marked as Recommended and Not recommended
- Recommended and Not recommended label ≠ Positive & Negative
- The data will be unlabeled for modeling, and the used model is VADER Sentiment Analysis

VADER Sentiment Analysis. VADER (Valence Aware Dictionary and sEntiment Reasoner) is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media, and works well on texts from other domains.

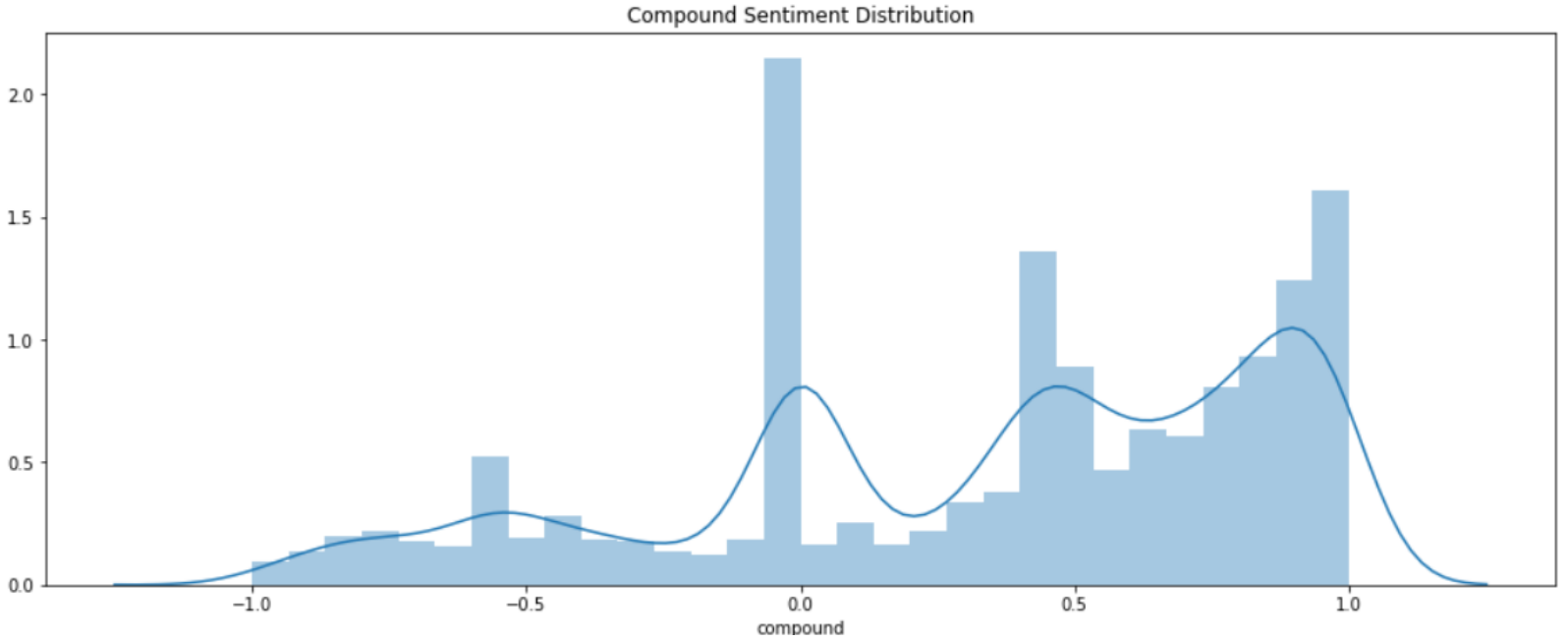
Unsupervised Modeling

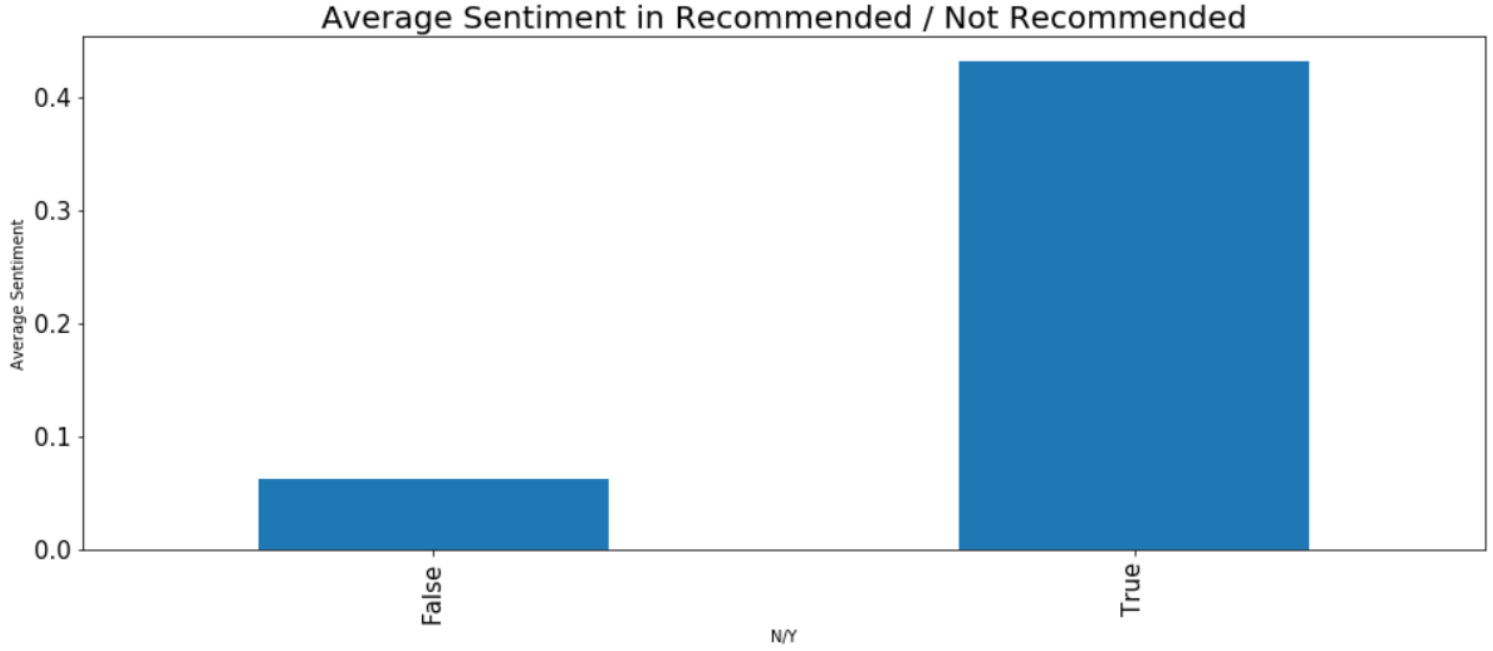
VADER has been found to be quite successful when dealing with social media texts, NY Times editorials, movie reviews, and product reviews. This is because VADER not only tells **about** the Positivity and Negativity score but also tells us about **how positive or negative a sentiment is**.[1]

The compound score is computed by summing the valence scores of each word in the lexicon, adjusted according to the rules, and then normalized to be between -1 (most extreme negative) and +1 (most extreme positive). This is the most useful metric if you want a single unidimensional measure of sentiment for a given sentence. Calling it a 'normalized, weighted composite score' is accurate.[2]



Unsupervised Modeling





Unsupervised Modeling

	voted_up	clean_str	scores	compound	comp_score
0	True	fall guy chill fun game really need server imp...	{'neg': 0.0, 'neu': 0.785, 'pos': 0.215, 'comp...	0.7430	pos
1	True	give little hug player stare back second jump ...	{'neg': 0.229, 'neu': 0.634, 'pos': 0.137, 'co...	-0.4391	neg
2	True	reccomend much buy	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...	0.0000	pos
3	True	hug people	{'neg': 0.0, 'neu': 0.244, 'pos': 0.756, 'comp...	0.4767	pos
4	True	little boy like see wipeout tv always dream ta...	{'neg': 0.0, 'neu': 0.549, 'pos': 0.451, 'comp...	0.8257	pos
...	...	...	...	...	...
13089	True	well make game get enough many way differentia...	{'neg': 0.06, 'neu': 0.492, 'pos': 0.447, 'com...	0.9460	pos
13090	True	fun competitive especially friend recommend bu...	{'neg': 0.0, 'neu': 0.21, 'pos': 0.79, 'compou...	0.8826	pos
13091	True	game extremely fun nice break usual drop loot ...	{'neg': 0.154, 'neu': 0.411, 'pos': 0.434, 'co...	0.8467	pos
13092	True	fun play keep work game	{'neg': 0.0, 'neu': 0.345, 'pos': 0.655, 'comp...	0.6908	pos
13093	True	turn around read previous review could recomme...	{'neg': 0.276, 'neu': 0.444, 'pos': 0.279, 'co...	-0.3944	neg

	voted_up	clean_str	scores	compound	comp_score
17	False	hacker destroy game use hack final matchill ch...	{'neg': 0.2, 'neu': 0.8, 'pos': 0.0, 'compound...	-0.5423	neg
19	False	hacking ruin game recommend reporting antichea...	{'neg': 0.284, 'neu': 0.373, 'pos': 0.343, 'co...	-0.0516	neg
20	False	update dev address cheat issue finally add eac...	{'neg': 0.184, 'neu': 0.682, 'pos': 0.135, 'co...	-0.9896	neg
47	False	game full hacker need report system something	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...	0.0000	pos
48	False	much enjoy game rampant hacker discord remove ...	{'neg': 0.147, 'neu': 0.669, 'pos': 0.184, 'co...	0.6652	pos
...	...	...	...	...	...
13071	False	really fun first time play get redundant annoy...	{'neg': 0.091, 'neu': 0.604, 'pos': 0.305, 'co...	0.9708	pos
13077	False	start game	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...	0.0000	pos
13083	False	buy game day release wait release whole hour t...	{'neg': 0.057, 'neu': 0.569, 'pos': 0.374, 'co...	0.9884	pos
13084	False	really fun start griefer hacker infest game pl...	{'neg': 0.25, 'neu': 0.568, 'pos': 0.182, 'com...	-0.8371	neg
13087	False	fun first hour gameplay however basic issue co...	{'neg': 0.111, 'neu': 0.679, 'pos': 0.209, 'co...	0.3832	pos

```
df_false = df.loc[(df['voted_up']==False)]
```

```
df_false.comp_score.value_counts()
```

```
pos    1987
neg    1421
Name: comp_score, dtype: int64
```

```
df_true = df.loc[(df['voted_up']==True)]
```

```
df_true.comp_score.value_counts()
```

```
pos    8528
neg    1149
Name: comp_score, dtype: int64
```

```
#Accuracy
(8528 + 1421) / (8528 + 1421 + 1149 + 1987)
```

```
0.7603362628964463
```

```
#False Positive Rate
1421 / (1421 + 1987)
```

```
0.4169600938967136
```

Conclusion & Improvements

- Conclusion
  - Reviews posted in Fall Guys mostly have positive sentiment
  - Users think that the game is fun and has potential but it has issues such as cheaters and server connection
  - Users mostly complains that the cheaters, server connection and some stages (the team game) ruin the game's fun
  - In general, the sentiment model can give acceptable result in average between recommended and not recommended reviews. But, there is several inaccuracies if it is in single review with polarities (e.g., I really like this game, but it gets redundant --> positive, because VADER emphasize the ‘really like’).

Conclusion & Improvements

- Future improvements
  - Improving the text cleaning so the topic labels and models can be more accurate (particularly on mistyped texts)
  - Evaluating the topic model accuracy
  - For sentiment analysis, it might be better to fix the dataset or setting threshold for the positive/negative sentiment or find entirely new models that can give better result.
  - Deploying the topic modeling for use in flask


THANK YOU!