“All Press is Good Press?” : Analysis on the Relation Between Subreddit Reviews and Steam Player Counts

Team Members: James Clark (JAC692), Alexander Hertadi (AFH78), Katherine O’Conner (KSO25)

Introduction and Problem Statement:

There is a famous saying “all press is good press”. Is this actually true? The aim of our project is to analyze the association between the sentiment and count of subreddit engagement of a few specific video games and their Steam player counts. We will attempt to measure video game sales via proxy through Steam player counts. There are two overarching goals:

1. We want to explore how the play-style design of the game (single-release, single release and DLC content, single-release and updates) correlates to reddit traffic and steam player counts of the games over time.
2. We want to determine the most important features for predicting Steam player counts. Specifically, we’re interested in how negative vs. positive subreddit comments and posts correlate with Steam player count.

With these goals in mind, we chose three games that each represented the play-style designs described above. We also opted for games released in 2016 as the Reddit torrent data is much smaller and more feasible to work with from that time frame. Also, since these games were released so long ago, we are better able to capture play trends and internet traffic patterns of the games over their most relevant timelines. We chose *Dark Souls III, No Man’s Sky, and Stardew Valley*. From our own background knowledge, *No Man’s Sky* is infamous for its negative review upon release, yet was still wildly popular.

Hypothesis:

We expect to find that single release video games will have an initial surge in sales/player count and taper off with time, whereas video games that get patches or additional downloadable content will experience more stable sales/player count. We expect this same trend for the count of subreddit comments and posts. We expect no strong relationship between time of release and sentiment of reviews.

Generally, we expect the number of Reddit comments and posts to be the most important predictor of Steam player counts. However, we predict that more positive sentiment reviews will be associated with more stable Steam player counts, whereas more negative sentiment reviews will be associated with sporadic steam player counts and an initial spike upon release.

Data Set:

Our dataset consists of comments and posts from Reddit for the year 2016, specifically for the games Dark Souls III, No Man’s Sky, and Stardew Valley. The specific subreddits we will pull data from are:

- r/DarkSouls3

- r/NoMansSkyTheGame
- r/StardewValley

Our data is sourced from an academic torrents website and because of its size, is stored in .zst format. Our first step in EDA will be to download, decompress and extract all data relating to the above 3 subreddits.

The data in question is very detailed and contains information about:

- Subreddit of the post
- Engagement with the post (upvotes and downvotes)
- Text of post/comment
- Date posted

There are a large number of fields available, but for our analysis we will focus on the above. A full list of features available can be found in the info.md files here.

Our initial data set contained 363,227 reddit posts and 4,475,317 comments from the specified subreddits with 53 columns ranging from the actual text of the post, when it was created, the number of upvotes, and less useful details such as media embeddings and html assets. From the available information, we filtered the data and chose to focus only on the number of posts, number of comments per day, average score of posts per day (upvotes - downvotes), average length of the post, the date, and a sentiment score from NLTK (Natural Language Toolkit).

The other datasets were very simply downloaded from SteamCharts. We will be focusing on the daily average number of players to approximate sales and overall popularity of the game.

Refer to Fig 1.

Process:

Upon finding the appropriate dataset, the data will be filtered to the desired subreddits associated with the video games of interest, and posts/comments will be categorized to be positive or negative. The resulting count and ratio of positive to negative post/comments will be compared to the sales/player count from the steam database.

For the Reddit post and comments dataset, a script was written to unpack the initial .zst file into a python readable .json file. From there the data was loaded and the following steps were made to make the dataset ready for further analysis:

- The time variable, “created_utc”, was converted from its initial format of UNIX timestamps (number of seconds elapsed since 00:00:00 UTC, 1 January 1970) to Python standardized datetime format.
- The data was filtered out for comments that did not have any content value, which in this case is where the “self_text” column contained one of the three following:
    - “[removed]” : comment removed by subreddit moderator or admin
    - “[deleted]” : comment removed by user

    - “” : empty comment
- Dataframe was reindexed with “.reset_index()” to account for the rows removed

For the SteamCharts database, we converted the DateTime variable to be of the standardized date format for easy processing.

Sentiment Analysis:

NLTK-VADER Sentiment Analyzer:

Natural Language ToolKit(NLTK) is a popular library in Python used for Natural Language Processing (NLP). It helps to simplify text data processing tasks such as classification, tokenization, stemming, tagging, parsing, and semantic reasoning.

Sentiment analysis in NLTK is done through the “nltk.sentiment” module. For this project, we decided to run VADER(Valence Aware Dictionary and sEntiment Reasoner) Sentiment Analyzer through the dataset. VADER, being developed based on social media text data, is a sentiment analysis tool that is created to analyze mainly informal texts through a curated set of rules and a lexicon—a list of lexical features (e.g., words, phrases, emojis) each tagged with sentiment scores, namely positive, neutral, or negative.

In addition to the general sentiment classifications, VADER also assigns an “intensity of sentiment” scores ranging from -1 to 1, where -1 to -0.05 is considered negative, -0.05 to 0.05 neutral, and 0.05 to 1 positive, on the input as a compounded sentiment score. This compound sentiment score is calculated by scoring not only the individual words, but also uses grammatical and syntactical rules of everyday language to adjust the scores. Some of these features are as follows:

- Capitalization : Increase of intensity in all-capitalized lettered words
- Punctuation : Exclamation points, questions marks, or multiple periods can change sentiment
- Conjunctions : Sentiment shifts due to words like “but”
- Modifiers : Intensity increases with adverbs such as “really”, “very”, etc.
- Negations : Phrases/words that have a preceding “not” will be classified correctly.

In addition, slang and emoticons are also taken into account, since they are prevalent in internet social media text. For our project, we defined a function “get_sentiment” which computes sentiment scores for text data contained within a pandas DataFrame, and then saving the results in a .csv file. Within the function, an instance of “SentimentIntensityAnalyzer” from NTLK’s “sentiment” module is created to utilize VADER.

Lists are then instantiated to store sentiment data such as:

- “Positive”, “negative”, “neutral”, “compound” : Sentiment score for each text input
- “Length” : Length of each text input
- “Subreddit” : Subreddit in which the text input originated

A loop is then run over each text entry where sentiment scores and other relevant data is entered to their respective dataframes, and at the end the dataframes are compiled into a .csv file to be read and used in visualizations and further analysis.

Ex: From r/StardewValley

- “"Don't get me wrong, I love farming and trading.. but gimme a sword and I'll be the happiest monster hunter you've ever seen. I'm so excited to gain skill and go through caves and mine through everything! GAH I'm so excited!!" → Compound Score: 0.9786
- "Fishing is the only thing I hate in this game and I can't do it" → Compound Score: -0.5719

Refer to Fig 2.

From the three video game subreddits, we were able to extract and plot a time series graph showing the average sentiment score of Reddit posts per week. The X-axis shows the date starting January 2016 up to the end of June 2017, while the Y-axis shows the average compound sentiment score, which typically ranges from -1 to 1. It can be seen for all these games that the average compounded sentiment score does not drop below zero, with No Man’s Sky and Stardew Valley both generating scores that are considerably above the positive sentiment threshold. We can also see comments tend to be less positive on average than our posts, but follow similar movements over time.

We can see significant drop-offs in the sentiment for each game over our time frame, most notably Dark Souls and Stardew Valley. We will investigate to see how this correlates with average players.

Refer to Fig 3.

We then plotted a stacked density histogram of the three games’ sentiment scores. We can see that the majority of posts and comments from all three games tend to have sentiment scores that fall within the positive range (0 to 1) and have the highest density around 0. From the three games, it can be seen that Stardew Valley has exceptionally high density in the positive sentiment score region, with its score density in the negative sides being very low, while Dark Souls’ negative sentiment score density is the highest of the three, and having the most balanced distribution of sentiment scores. No Man’s Sky falls in between the two games in terms of distribution.

Exploratory Analysis:

After cleaning the data, we initially created correlation heatmaps to get a better sense of the strength of positive and negative associations between variables and daily player steam counts. We also analyzed the correlation at an individual level per game, and for all the subreddit comments and player counts as a whole.

Refer to Fig 4.

Our first row of correlation matrices are done on posts while our second row is on comment data. As can be seen from the graphs, we created the correlation heatmaps based on the same variables for each game:

- num_posts (only present in posts) : Number of posts from the respective game’s subreddit
- num_comments : Number of comments from the respective game’s subreddit
- average_score : Average score of each post/comment, denoted by number of upvotes subtracted by number of downvotes
- sentiment : Average sentiment score of the posts/comments

- average_length : Average length of posts/comments
- created_utc : Time post/comment was uploaded
- Players : Game player counts from the steam dataset

Interesting insights from the heatmaps:

- We can see that ‘num_posts’ and ‘num_comments’ have very strong positive correlations with each other, which makes sense since more posts published should mean more comments.
- For all three games, ‘average_length’ and ‘sentiment’ have a very high correlation. This suggests that it is likely quite discernable over long text blocks whether a comment is negative or positive.
- There is a moderate level of correlation between ‘Players’ and both ‘num_posts’ and ‘num_comments’ for Stardew Valley and No Man’s Sky, while showing a very high correlation for Dark Souls 3. This could indicate subreddit activity increasing as the active players increase.
- For Stardew Valley, there is a weak negative correlation between ‘created_utc’ and ‘num_posts’ and ‘num_comments’. Although the scores are quite low, at -0.3 and -0.27 respectively, it may indicate a trend of the subreddit having less engagement with time
- For Dark Souls 3, there is a moderate positive correlation between ‘created_utc’ and ‘average_score’ (0.56). This could mean more positive engagement on the game subreddit with time

Overall, initial correlations suggest that the number of comments and posts could be an indicator of player count, regardless of the type of game. We can see sentiment is very weakly correlated with players regardless of game, which suggests the actual sentiment of the player base could be irrelevant as a predictor for its active player count.

Exploratory Modeling:

We wrote a custom function to randomly split the data into training test splits and apply an sklearn linear regression, ridge regression, k-nearest neighbors regressor, decision tree and random forest models to the data. It then produces a scatter plot of the predicted values against the actual values. The values from the training dataset are plotted in orange and the test points are in blue.

Refer to Fig 5.

Refer to Fig 6.

The scatter plots (Fig 6 in appendix) and the testing MSEs (Fig 5 in appendix) show that Random Forest and Decision Tree yielded the lowest MSE out of the models tested, with the exception being the spike in MSE for Decision Tree on comments for No Man’s Sky. While initially this may seem unusual, the better performance of the Decision Tree indicates that we may be able to better approximate our daily players with fewer features and that the random forest may be more complex than what we need.

However, the unusually high MSE for our comments Decision Tree suggests that solely using the number of comments might be a cause for underfitting in some cases. The exact reason for this is unclear, but it should be noted that No Man’s Sky had a very negative press reaction on release, and thus it is likely that

despite there being a large number of comments on the subreddit, the player count may have dropped significantly due to poor reviews.

Looking at our individual graphs we can see fairly high MSE values across the board, though they are generally much higher in our simple linear and ridge regressions, with KNN for the most part being not too far behind. Our predictions for Linear, Ridge and KNN deviate significantly at higher player count values, whereas the spread in our Decision Tree and Random Forest models is much more contained across our range of possible values.

Because of this, we opted to employ the Random Forest model to determine what, and if any, of our features were good predictors of daily player counts. While we could have opted for the Decision Tree model, we wanted to investigate how the model ranked feature importance and whether we could improve the Random Forest through tuning. Additionally, the underfitting in the Comments data by the Decision Tree suggests Random Forest could be a more stable predictor across the board.

Hyper-parameter Tuning of Random Forest Regressors

Refer to Fig 7.

In an attempt to improve model performance, we wrote a custom function to choose the optimal tree depth for the Random Forest Model regressors. For every specified depth, this function creates 10 random forest model objects on different random splits of the data, then averages the test mean squared error of those models. The function then returns the depths with the lowest average of the testing mean squared errors. Additionally the graph plots the averages and ranges of the testing mean squared errors as depicted below.

Unfortunately, there is no apparent significant model improvement from tuning of the depth parameter of the random forest regressor objects, though interestingly we can see that our random forest for Stardew Valley had a much lower average tree depth than our other games, which coincides with it being the most effective of the Random Forest models.

Analysis of Feature Importance via Mean Decrease in GINI Impurity

To determine what features our Random Forest treats as the most important, we plotted the feature importance across our 3 games and across our entire dataset for both posts and comments.

Refer to Fig 8.

For our posts we can see that the ‘num_posts’ variable is clearly the most important feature across all random forest models and that ‘sentiment’ is largely irrelevant. We can also see that ‘created_utc’ and ‘num_comments’ are our second and third most important features, differing slightly by game.

The low depth of the Random Forest and the low MSE of both Decision Tree and Random Forest for Stardew Valley is likely due to the lack of importance in all other features, and thus approximated better

by a simpler, lower depth forest. For Dark Souls 3 and No Man's Sky, 'num_comments' and 'created_utc' are more evenly distributed, with 'average_score' being more significant for both than in Stardew Valley.

If we look at our comment data which does not have num_posts, we can see that num_comments is likewise our most important feature and sentiment is largely irrelevant.

Refer to Fig 9.

Comments paints an even clearer picture; 'num_comments' is our most important feature with 'created_utc' as second and all other features as largely irrelevant.

The uniformity of feature importance across all games and the strength of both 'num_posts' in our post data and 'num_comments' in our comment data suggest that not only does the type of game (in terms of release) likely not matter in predicting the average player count over a period of time, but the sentiment of the reddit fanbase also has little to no effect. The single largest predictor of Steam player counts is solely the amount of activity on a games' subreddit, regardless of whether that content is positive, negative, long or short.

Conclusion and Summary

In our analysis, we sought to investigate whether the type of game and the sentiment of the games' subreddit could be used as a predictor for the player count of the game over time. We sought to apply several models to our data to see which one, if any, offered the best predictor.

There appears to be little impact, outside of the number of comments and posts, of a subreddits behavior on the player count of a video game. We found that sentiment was largely irrelevant, having low correlation with player counts and a very low feature importance in our Random Forest models. This suggests that our sentiment scores were not accurate or indicative of the true sentiment of the gaming playerbase and/or that the sentiment of a games' subreddit has no bearing on the active player base.

Our only real variable of importance was the number of comments and posts, which largely acts as a proxy for the activity in the subreddit. It is logical that this would have a relationship with the number of active players, however it is not certain whether the number of active players acts as a predictor of the activity on the subreddit or vice versa. Given that most of our models had a very large MSE, it is likely that the amount of active players is what drives the amount of activity on a subreddit, and that the activity on a subreddit has very little bearing or influence on the number of active players, even if that subreddit has a negative or positive sentiment.

In conclusion, the activity on a subreddit does not strongly influence the amount of active players playing the game on Steam. The number of comments and posts is correlated with the number of active players, but it cannot be ascertained what way this relationship flows, and whether it is an active player base that drives an active subreddit, or an active subreddit that encourages people to play the game.

Future Analysis

There are several improvements to our analysis that we could not implement due to timing and lack of data. Our model is not a WMD by any means, though our use of active player base as a proxy for steam sale counts could lead to a statistically weak outcome. It could also be argued from a fairness perspective that Reddit data is biased and represents a small segment of the gaming community. However, the purpose of our project was to specifically measure whether Subreddits had an effect on the game itself, and thus this bias is not relevant.

In future work we would look to alternative ways of measuring sentiment. Our sentiments were all largely neutral, and while this may not be untrue for our subreddits, our expectation, particularly for No Man's Sky, was for much more divisive sentiment values. With more time and resources, we would look into alternative sentiment analysis techniques, such as an API into a large language model such as ChatGPT or Mistral, or fine-tuning a smaller LLM such as BERT on our data.

We would also seek to look at a wider range of games to more distinctly measure the impact of game type, as our analysis was limited to just 3 games and a wider selection of games might uncover potential relationships between the player counts and the subreddit activity. This is a potential source of bias as we are limiting our analysis to a small subset of games that likely all have very different communities, and thus the impact of their subreddits might not be comparable.

Lastly, it would be worth investigating how Steam reviews tracked over the same period, and whether the sentiment in a games subreddit tracked or influenced the sentiment on the Steam store page. This is likely a better measure of how impactful a games' subreddit can be on the game itself, and could help address our aforementioned proxy issue.

References

stuck_in_the_matrix, Watchful1, RaiderBDev (2024), *Reddit comments/submissions 2005-06 to 2023-12*. Academic Torrents, Accessed 5 May 2024. https://academictorrents.com/details/9c263fc85366c1ef8f5bb9da0203f4c8c8db75f4.

Bird, Steven, Edward Loper and Ewan Klein (2009), *Natural Language Processing with Python*. O'Reilly Media Inc.

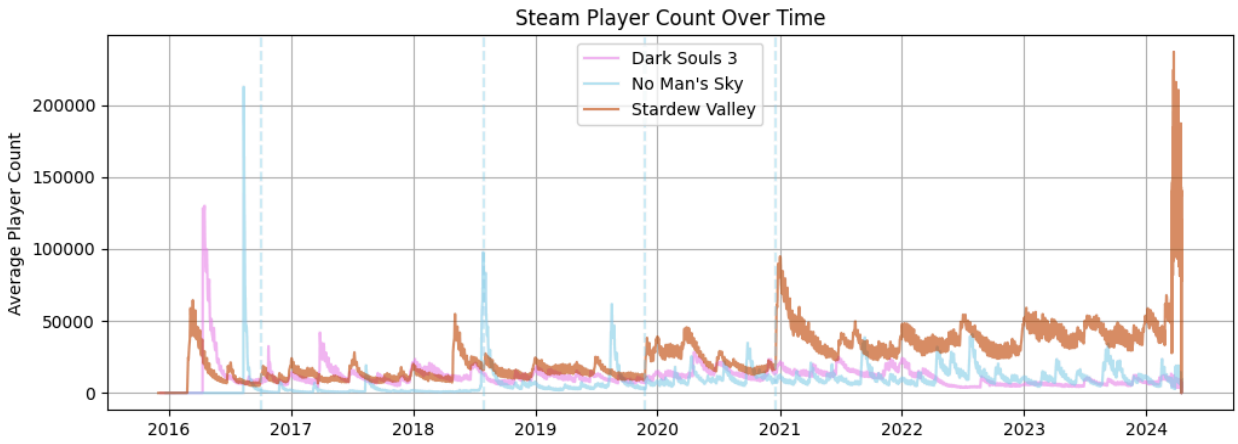
Course Code: ORIE5741
Github Link: https://github.com/JACProjec/ORIE5741PJ/tree/main

Team member contributions: All team members worked on exploratory data analysis, modeling and writing of the report and slides

- James Clark: Data cleaning
- Katherine O'Connor: hyperparameter optimization, video editing
- Alex Hertadi: Github Management

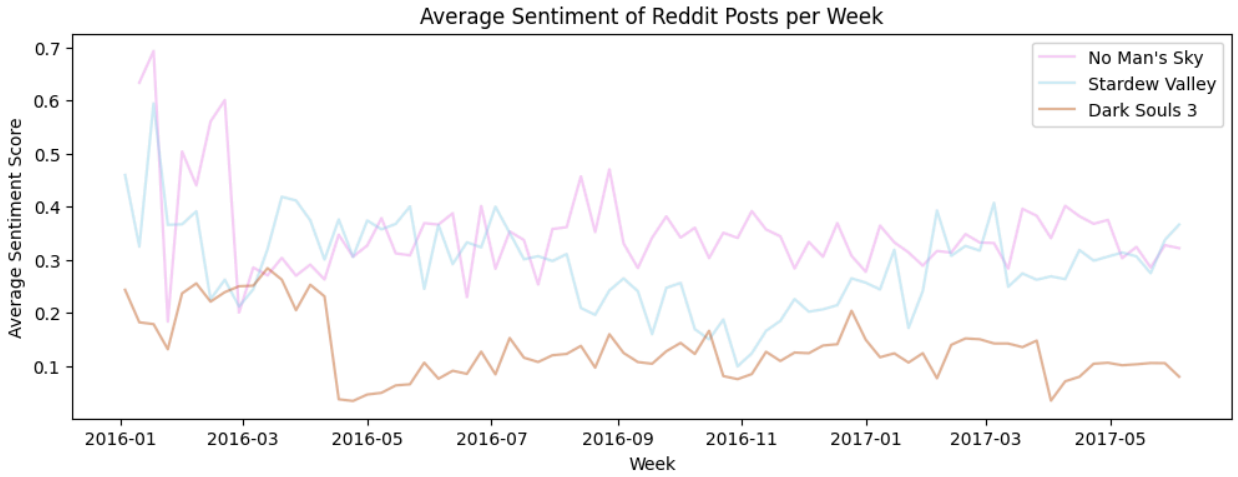
Figures

Fig 1: Steam player counts for our 3 games since release



The dotted lines represent major game updates.

Fig 2: Average sentiment of comments and posts for our 3 games over the Jan-2016 - June 2017 period



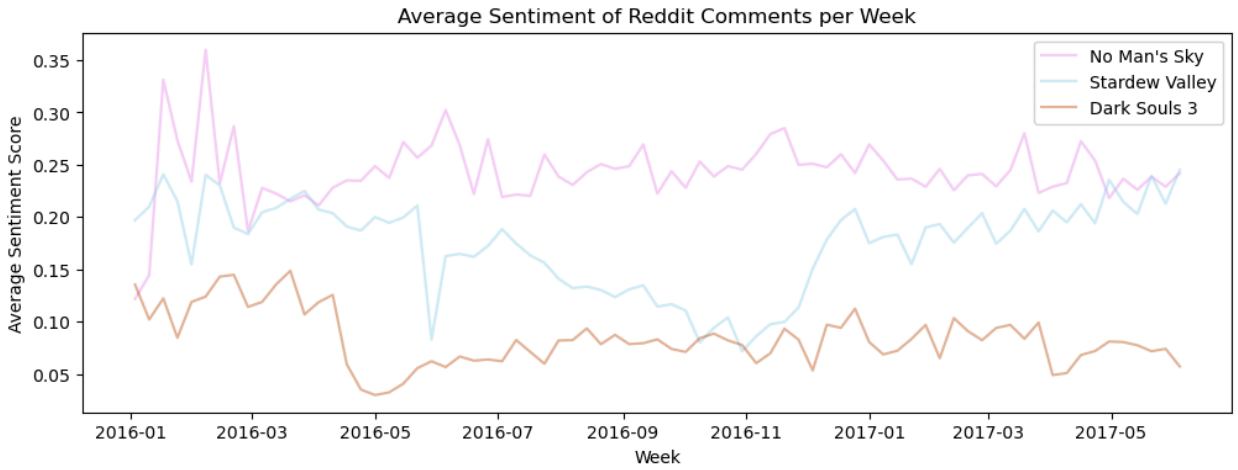


Fig 3: Distribution of sentiment for posts and comments of our 3 games

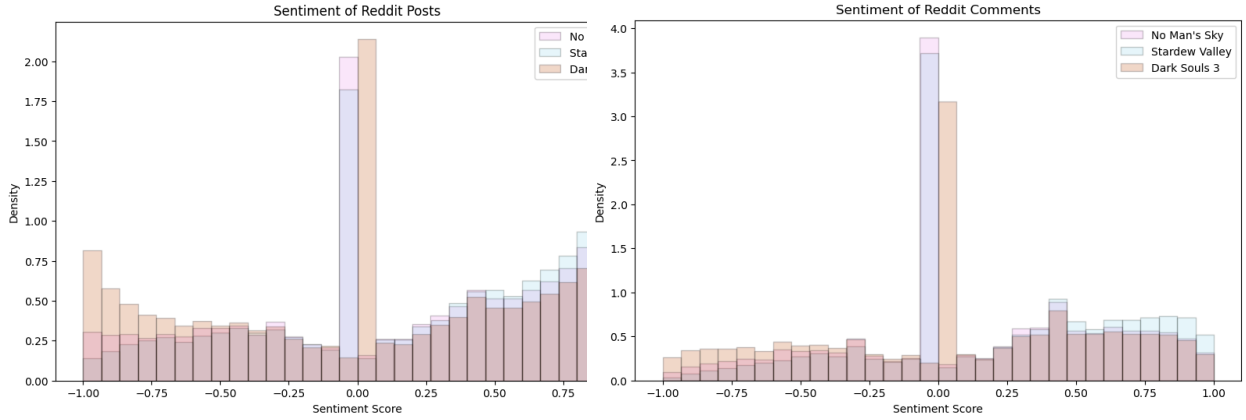


Fig 4: Correlations for comment and post data across our 3 games

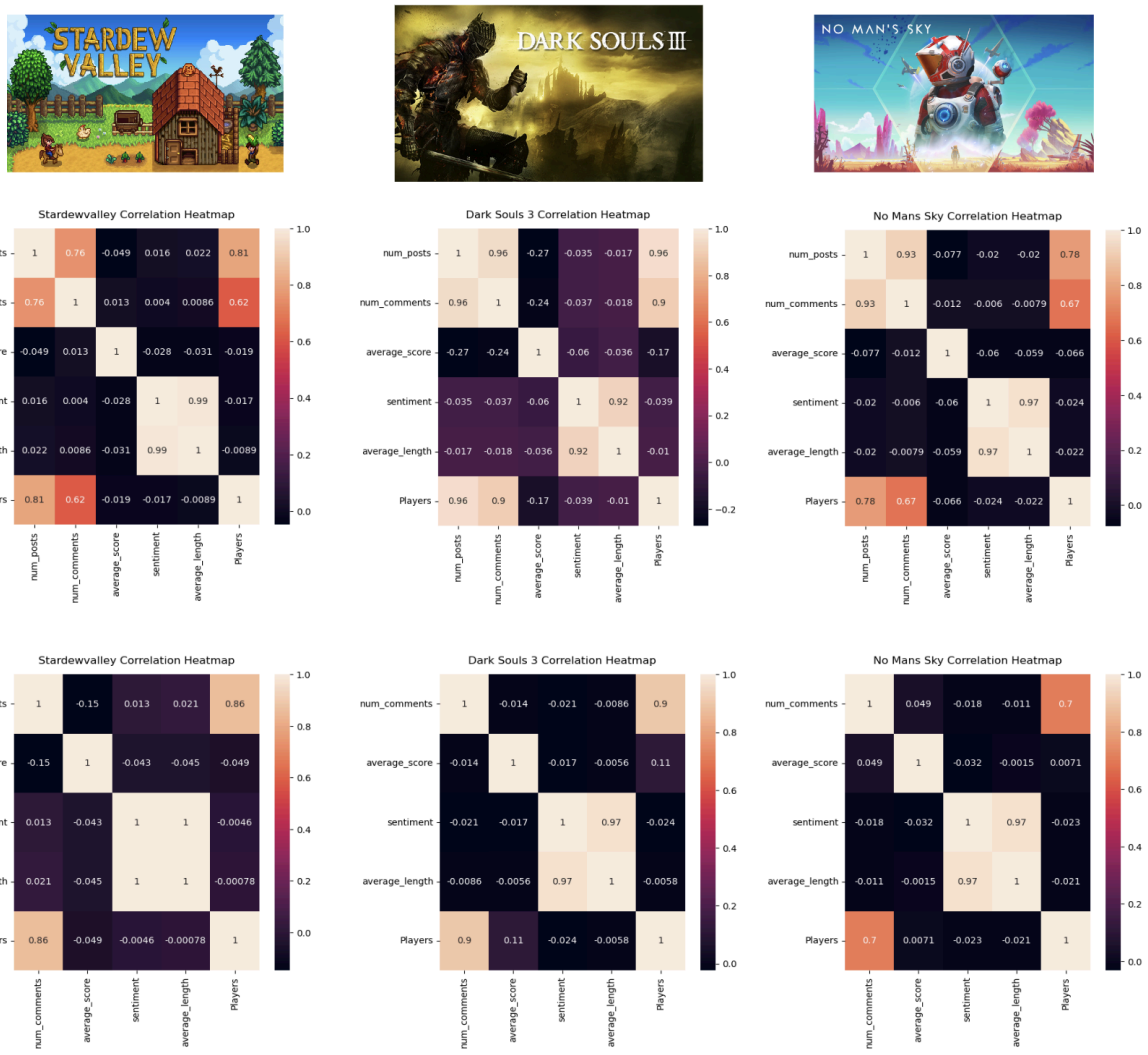
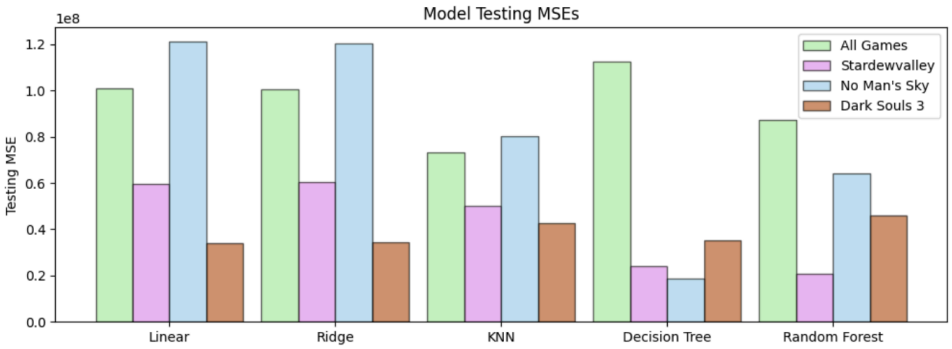


Fig 5: MSEs per model per game for posts and comments

Posts



Comments

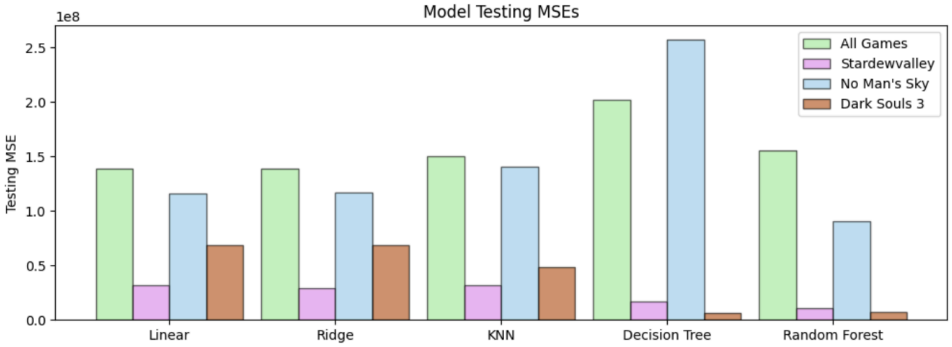


Fig 6: MSE per model per game for post data

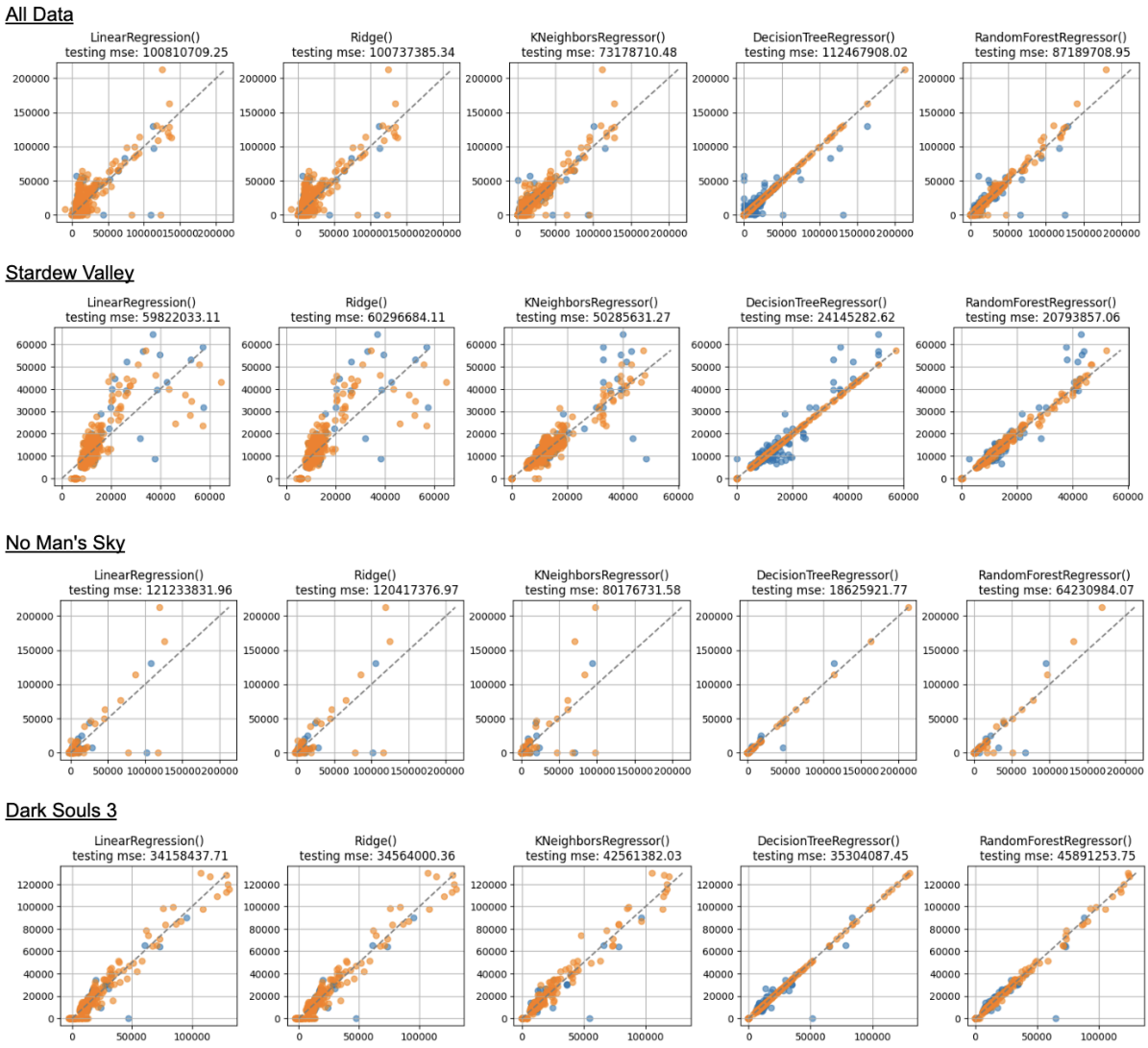


Fig 7: Testing MSE by Random Forest depth for our post data

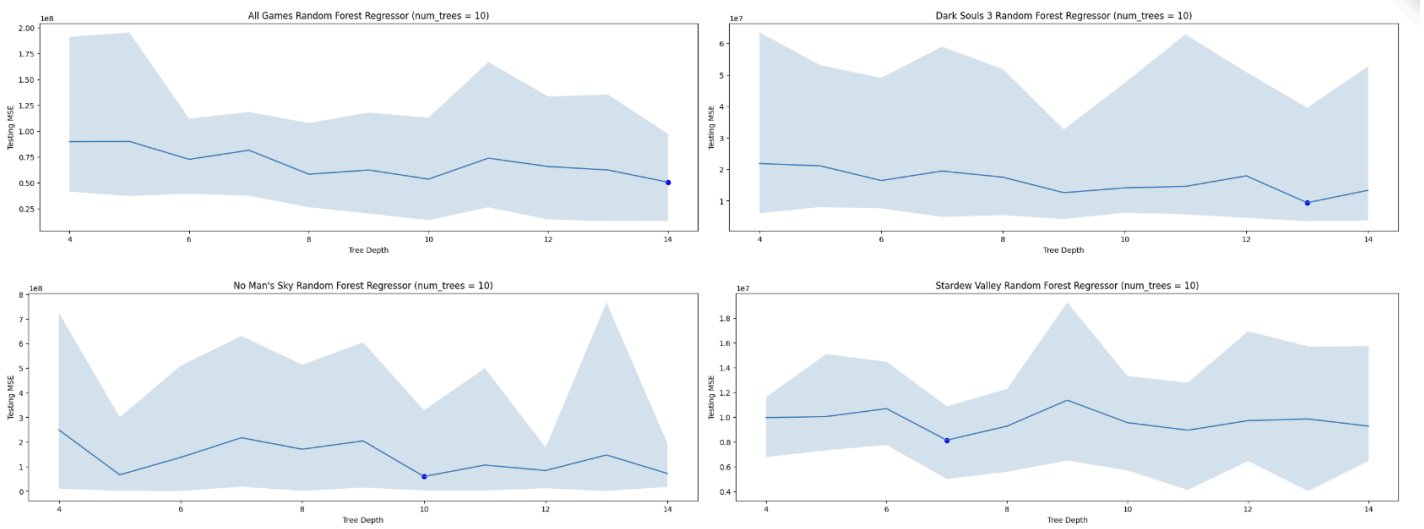


Fig 8: Feature importance for our Posts data

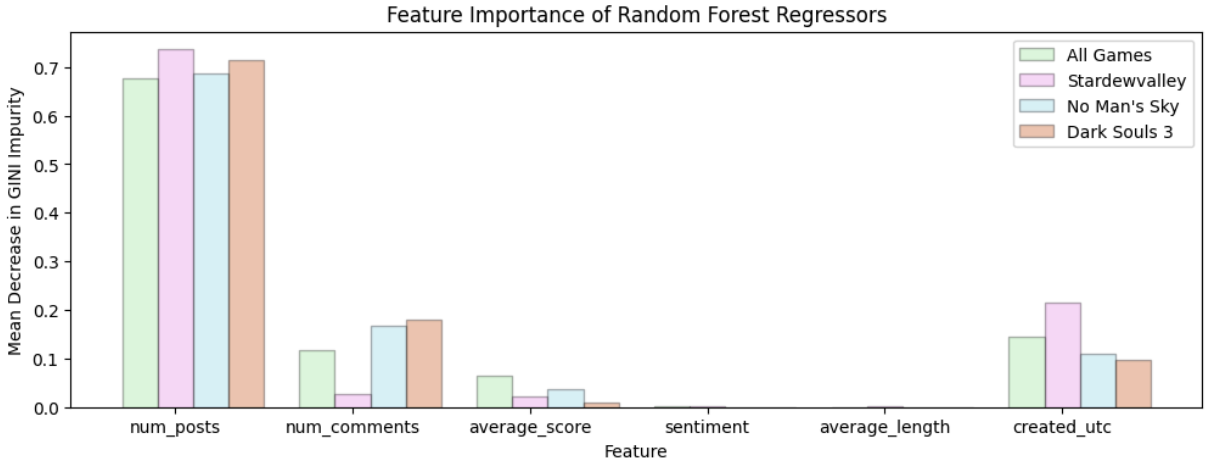


Fig 9: Feature importance for our Comments data