A Text-Based Analysis of Consumer Complaints in Tom Clancy's Rainbow Six Siege Game Reviews

Kyle Chu
kgc42@cornell.edu

Abstract

Online game reviews play a central role in shaping consumers' decision-making on digital marketplaces such as Steam, where users can provide a text-based review and indicate whether they "recommend" or "not recommend" the game. While negative sentiment is often equated with product dissatisfaction, many reviews exhibit a tension between critical language and positive endorsement. This paper explores this phenomenon in the context of Tom Clancy's Rainbow Six Siege by asking: among reviews that express negative sentiment, what distinguishes those that still recommend the game from those that do not? I am testing the hypothesis that negative, but recommended, reviews emphasize core gameplay quality and enjoyment, whereas negative non recommended reviews focus more on bugs, performance issues, and monetization concerns. To test our hypothesis, we model recommendation decisions within the negative subset using interpretable statistical models that relate complaint content to endorsement behavior. Statistical tests reveal significant differences, where recommended reviews contain 19% more gameplay language but 57% less monetization language and 42% less quality/bug language compared to non-recommended reviews. Logistic regression modeling confirms these patterns, with gameplay language associated with increased odds of recommendation and monetization language associated with decreased odds. These findings support the hypothesis and suggest that even when expressing negative sentiment, reviewers distinguish between fundamental gameplay enjoyment versus monetization and technical issues when making recommendation decisions. The results have implications for understanding how consumers evaluate complex digital products and how developers might prioritize improvements based on review content.

1 Introduction

Online reviews have become a primary mechanism through which consumers evaluate products. In the video game industry, platforms such as Steam host millions of user (and AI) generated reviews that combine textual feedback with a binary recommendation value. These reviews influence purchasing decisions, game developer reputation, and perception of game quality, making them a crucial source of data for studying consumer behavior.

A common assumption is that negative sentiment in review text corresponds directly to people not liking the game. However, this assumption overlooks a common pattern in gaming communities where many users express strongly negative opinions in their written reviews while still recommending the game overall. For example, players may criticize bugs, balance issues, or pay-to-win practices while simultaneously praising core gameplay principles and advising others to play the game. This apparent contradiction raises an important question about how consumers evaluate video games: when people complain, what determines whether they ultimately endorse or reject a product?

Previous research has examined how exposure to game reviews influences player perceptions and experience. For example, Livingston et al. conducted a controlled study in which participants read positive or negative reviews before playing a game and found that the vibe of review text significantly affected subsequent game ratings, independent of mood changes, suggesting that negative reviews bias player perceptions of game quality (Livingston et al., 2011). However, this study and many others focus on the broad effect that negative reviews have on consumers. This study aims to focus on what types of negative complaints are associated with whether users ultimately recommend a game or not, thereby unpacking the underlying the influence of negative reviews.

We hypothesize that different types of negative feedback carry different implications for recommendation decisions. Complaints centered on core gameplay quality and enjoyment are more likely to be linked with positive recommendations, whereas complaints emphasizing technical problems or monetization practices are more likely to result in non-recommendations. To test this hypothesis, we apply computational text analysis methods to a large corpus of Rainbow Six Siege reviews and estimate statistical models linking linguistic features to recommendation outcomes.

2 Data

This study uses the Steam Reviews 2021 dataset, a publicly available corpus of user game reviews collected from the Steam platform and hosted on Kaggle (Najzeko, 2021). The dataset contains over 8 million reviews across hundreds of games, with each review including the review text, a binary recommendation indicator, language, timestamps, and metadata about the reviewer and purchase status. We used this original data and applied some filters, such as keeping only English reviews, reviews where the reviewer purchased the game, and reviews that were longer than 10 words.

However, the collected data has some limitations. First, the dataset only includes reviews from 2013 to 2021, which may not capture the most recent trends/sentiments about the game's current state. Second, by filtering to English only reviews and Steam purchases only, the analysis excludes non-English players and users who obtained the game through other platforms such as the Ubisoft store, potentially limiting generalizability. Third, the 10-token minimum threshold, while necessary to filter out uninformative reviews, may exclude some legitimate but brief feedback. Despite these limitations, the dataset provides a substantial and representative sample of English-language Steam reviews for Rainbow Six Siege, suitable for examining linguistic patterns in negative reviews that recommend versus those that do not.

3 Methods

For this analysis, I filtered the dataset to focus on Tom Clancy's Rainbow Six Siege (Steam app ID: 359550), a popular tactical first-person shooter game. The initial dataset contained 371,154 English reviews for this game. I then removed reviews with null or empty text (998 removed), filtered to only include reviews from users who purchased the game on Steam, and removed reviews with fewer than 10 tokens to eliminate extremely brief or uninformative entries. Then, to isolate the "negative" sentiment reviews, I used NLTK's VADER sentiment analyzer. I chose NLTK's VADER because it is specifically designed for informal, user-generated text and incorporates general linguistic heuristics (like capitalization, punctuation, and negation words) that are common in online reviews and gaming discourse. VADER produces a continuous sentiment score that allows for threshold-based filtering and sentiment intensity, and it has been shown to perform competitively with supervised sentiment models on short, social-media-style text without requiring task-specific training data (Hutto and Gilbert, 2014). I kept all reviews with VADER sentiment score less than -0.2, making this our "negative" reviews dataset. I chose -0.2 to ensure that these reviews are definitively negative and not just slightly neutral/negative. After these filters, 20910 negative reviews remained. The dataset exhibited some class imbalance, with 11461 recommended negative reviews compared to 9449 not-recommended negative reviews. To ensure balanced analysis, I randomly sampled 9449 reviews from the "recommended" class, resulting in a final balanced dataset of 18898 total reviews.

For text preprocessing, I lowercased the review text, removed URLs, and lemmatized the words using spaCy's en_core_web_sm model. Lemmatization helps handle words with the same underlying concept, but slightly different syntax, such as plural words. This improves topic coherence and makes lexicon counts more robust and concise. After lemmatizing the text, I created three lexicons/dictionaries that represent the three categories stated in our hypothesis (bug/performance issues, monetization complains, and core gameplay features). For each review, I counted lemma matches from each lexicon and normalized by length to get rates per 100 tokens. I wanted length normalization similar to TF-IDF, where tokens are counted by relative frequencies rather than raw frequencies. However, I didn't use TF-IDF because I am only accounting for specific words, not all words. In addition, I computed other features such as review length in tokens, count of exclamation marks in the raw text, and profanity count (of a specific, manually selected set of words) to capture overall intensity and tone that might correlate with both sentiment and recommendation.

I also wanted to capture more open-ended themes, so I applied Latent Dirichlet Allocation (LDA) topic modeling to the lemmatized reviews. I used CountVectorizer with English stopword removal, min_df=5, max_df=0.8, and a vocabulary size capped at 5,000 to down-weight rare words/typos. I tested across multiple topic models from 5 to 15. I noticed that too few topics tended to mix unrelated topics while too many topics tend to be harder to interpret and redundant. The fitted LDA model yields, for each review, a topic proportion vector. I settled on using N_TOPICS = 7 for the best mix of interpretability and discrimination and appended these 7 continuous topic values (topic_0, . . . , topic_6) to the dataset. This allowed our feature set to consist of cleaned data like the cleaned text, normalized counts of the three categories' word frequencies, and token count while also including the proportion of how much the review fell under each topic.

Finally to test our core hypothesis, I fit a logistic regression model using scikit-learn with default L2 regularization. The outcome variable was the recommended column. The input vectors were the three normalized lexicon features, review token length, and the top 5 most group differentiating topic proportion values. I selected these top 5 topics by comparing the absolute difference in mean topic proportions between the recommended and non-recommended reviews for each topic. I chose Logistic regression because it produces signed coefficients and odds ratios for each feature that can be interpreted in terms of the hypothesis.

4 Results

Figure 1 shows the average number of lexicon matches per 100 tokens for three complaint categories (quality/bugs, monetization, and gameplay). Even within negative reviews, the two right-most subgroups look different: negative not-recommended reviews mention quality/bug and monetization language more often, while negative recommended reviews mention gameplay/fun language more often. This suggests that there are significant differences in language for reviews that recommend and don't recommend the game.

To check whether these differences are statistically reliable, I ran t-tests comparing the recommended and not-recommended negative reviews on each lexicon feature. Figure 2 summarizes both the t-statistics and the corresponding $-\log_{10}(p)$ values. All three features show significant differences between the groups: quality/bug and monetization are higher in not-recommended reviews, and gameplay is higher in recommended reviews. In other words, the patterns in Figure 1 are not just random noise.

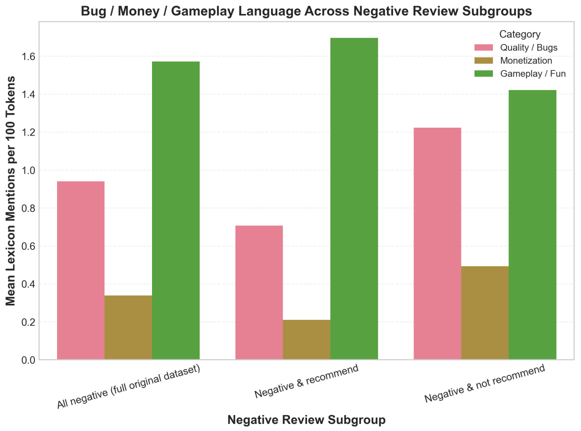

Figure 1: Mean lexicon mentions per 100 tokens across negative reviews. Negative not-recommended reviews contain more quality/bug and monetization language, while negative recommended reviews contain more gameplay/fun language.

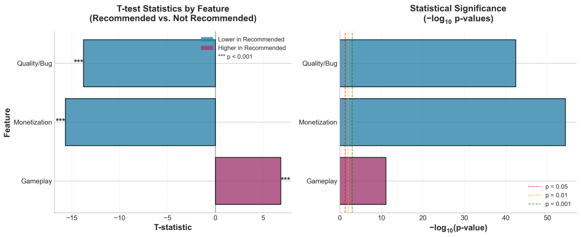

Figure 2: t-test results comparing negative recommended vs. negative not-recommended reviews. The left panel shows the t-statistics (direction indicates which group is higher), and the right panel shows statistical significance as $-\log_{10}(p)$.

Figure 3 shows the largest topic differences (Recommended − Not Recommended). Some topics are clearly more common in negative reviews that still recommend the game, while other topics are clearly more common in negative reviews that do not recommend it. More specifically, topics 1, 3, and 6 are more common in negative recommended reviews while topics 5, 4, and 2 are more common in not recommended reviews. Looking at the top words of these topics, topics 1, 3, and 6 has words such as "shoot", "kill", "team", "enemy", "game", "good", "fun", "player", "match", etc which are more representative of gameplay-related words. On the other hand, topics 5, 4, and 2 have words like

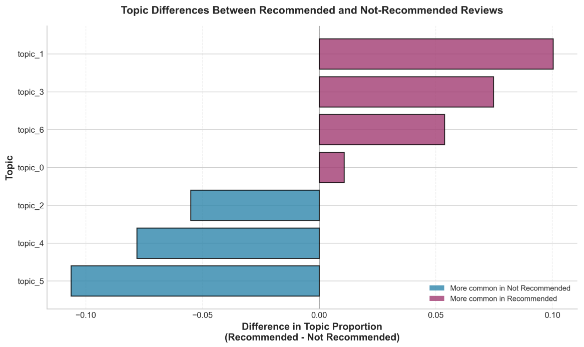

Figure 3: Differences in mean topic proportions between negative recommended and negative not-recommended reviews (Recommended − Not Recommended). Positive values indicate topics more common among negative recommenders; negative values indicate topics more common among negative non-recommenders.

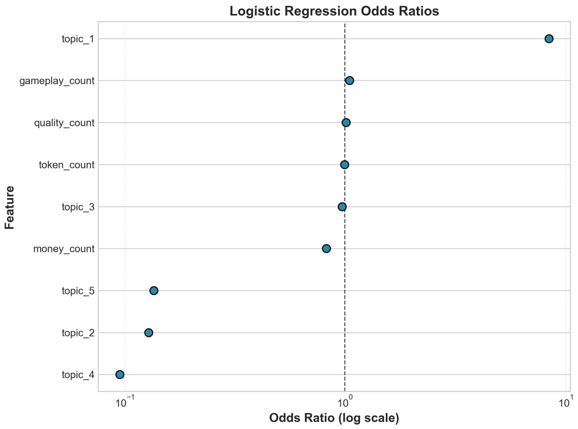

Figure 4: Odds ratios from logistic regression predicting whether a negative review recommends the game. Points to the right of 10^0 increase the odds of recommending. Points to the left decrease the odds.

"bad", "bug", "fix", "broken", "server", etc which align more with quality issues. Lastly, topic 0, which is the least differentiating topic, includes words like "edition", "money", and "pay" which indicate monetary issues with the game. This shows how money and pay to win mechanics may not be as prevalent of an issue in Rainbow Six Siege.

Finally, we fit a logistic regression model to see which features matter most when we consider them at the same time (lexicon counts, review length, and selected topic proportions). Figure 4 reports the model's odds ratios on a log scale. The results line up with the earlier comparisons, where gameplay-related language increases the odds that a negative review is still a recommendation (odds ratio > 1), while monetization language decreases those odds (odds ratio < 1). Several topic features have even larger effects than the lexicon counts, showing that the bigger themes in the review captured by topics separate the two groups strongly. Among the lexicon-based predictors, monetization language is associated with a lower chance of recommending: money_count has a negative coefficient (-0.194) and an odds ratio of 0.824, meaning that each additional monetization-related mention multiplies the odds of recommending by about 0.82 (roughly an 18% decrease). In contrast, gameplay language is associated with higher odds of recommending: `gameplay_count` has a positive coefficient (0.049) and an odds ratio of 1.050, implying about a 5% inc

All in all, these results show that negative reviews are significantly different in content. People can write negatively and still recommend the game when their complaints are more about gameplay frustrations, but they are less likely to recommend when their complaints are more about quality or the game being broken. However, reviews regarding money don't seem to have the same effect on whether the user recommend or doesn't recommend the game.

5 Discussion

The results support the hypothesis that negative reviews are not a rigid signal of dissatisfaction. Instead, satisfaction is dependent on what the reviewer is negative about. Specifically, the regression estimates imply a qualitative separation between complaints that preserve the game's perceived "core value" and complaints that undermine it. When negative language is tied to the core mechanical principles of the game (e.g., gunplay/mechanics), recommendation remains likely and that the product is still worth others' time or money despite frustrations of personal skill. In contrast, when negative language concentrates on systemic unreliability (servers/broken systems, bugs/issues) that affects all players regardless of skill, recommendation becomes substantially less likely.

6 Limitations

Firstly, the reviews used in this project may not be fully reliable. Sometimes humans would write reviews heavily criticizing the game in all different aspects, but then proceed to recommend the game. Therefore, its hard to find a true relationship between the words in the review and if its recommended because reviewers don't include all of their reasons in the review text when making their recommendation decision. Secondly, the VADER sentiment analysis is flawed because it isn't readily able to handle gaming-style vernacular and correctly analyze it. What may seem as positive to a "gamer" might come off as negative to an average person, and vice versa. Therefore a model trained strictly on labeled "gamer" reviews could result in better accuracy/sentiment compound scores for reviews of this nature. This dataset is also only limited to Steam reviews of Rainbow Six Siege, possibly failing to generalize for other games/genres. Some more pay to win games might have higher mean topic proportion differences for "money" related topics than Rainbow Six Siege's results. Lastly, this study is mostly correlational where we tend to notice higher counts of certain categories given recommended or not, rather than definitively establishing causality.

References

C. J. Hutto and Eric Gilbert. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media*.

Ian J. Livingston, Lennart E. Nacke, and Regan L. Mandryk. 2011. Influencing experience: The effects of reading game reviews on player experience. In *Proceedings of the International Conference on Entertainment Computing (ICEC)*, pages 89–100.