Sentiment Analysis of Game Reviews

Daniel Wertz
Computer Science
Western Washington University
Bellingham, WA
wertzd@wwu.edu

***Abstract*—The aim of this study is to perform sentiment analysis on game review text to identify characteristics most likely to result in games receiving positive or negative reviews. We also attempt to predict whether a review is positive or negative based on the review text, genre, and release date information. To achieve this goal, we will explore training machine learning models based on both logistic regression and decision tree algorithms. In this report, we will provide a detailed outline of the procedures undertaken to explore the dataset and extract knowledge from it. Our findings reveal potential factors influencing the sentiment of game reviews and the effectiveness of the prediction models.**

Keywords—Machine Learning; Sentiment Analysis; Video Games; Data Mining

I. Introduction

In recent years, the popularity of online gaming has soared, resulting in a vast amount of user-generated game reviews available on various platforms. Understanding the sentiment expressed in these reviews and identifying the factors that contribute to positive or negative sentiments can provide valuable insights to game developers and publishers. The aim of this project is to perform sentiment analysis on game review text and investigate the characteristics that are most likely to influence whether a game review is positive or negative. Additionally, we seek to predict the sentiment of reviews based on the textual content, genre, and release date information. By employing machine learning techniques, specifically logistic regression and decision tree algorithms, we aim to develop accurate prediction models that can assist in assessing the sentiment of game reviews.

II. Data Preperation

A. Datasets

In our study, we utilized two datasets in our analysis. The first dataset consisted of ~6.4 million Steam reviews along with corresponding review scores. The second dataset contained comprehensive information about various aspects of most games listed on Steam, including genre and release year. By leveraging the unique Steam IDs present in both datasets, we were able to merge and consolidate the relevant data, focusing on the necessary columns for our analysis. The resulting data contained information on each game's genres, release date, and the text and score of each review.

B. Missing Attributes

Both datasets used had no missing attributes in the relevant columns, so a missing attribute policy was not needed.

C. Discretization

Since the features we utilized from our datasets only contained string and binary features, we did not perform discretization on the numerical features. One way discretize was used it through the bag-of-words technique, which involves converting each string into a vector of word frequencies. We applied this technique to the review text strings and the resulting data consists of numerical feature vectors representing the frequency of each word in the reviews. This approach helped to simplify the text data and enabled the use of logistic regression algorithms, which require numerical inputs.

D. Preprocessing

Our initial review text data contained over 3 million unique "words", so we took several steps to reduce the feature space and eliminate potential noise. This involved extensive preprocessing of the text data in the stream reviews, including stemming/lemmatizing, removal of punctuation and special characters, and tokenization. We also removed stop words such as "the", "is", and "so". Furthermore, we removed all words that did not occur 100 times or more in our set of reviews. By implementing these preprocessing steps, the resulting feature representation was made more robust to variations in language and style. This also helped to reduce the dimensionality of the review text feature space down to ~17 thousand words, thereby improving computational efficiency, and reducing the risk of overfitting.

III. Exploretory Analysis

A. Genre Distribution

To gain insights into the genre distribution of our dataset, we examined the frequency of different genres in the merged data. Fig 1. presents a bar graph illustrating the top 10 genres based on their occurrence. This visualization provides a clear overview of the prevalent genres in our dataset, highlighting the most common types of games represented.

GRAPH I. TOP 10 GENRE DISTRIBUTION

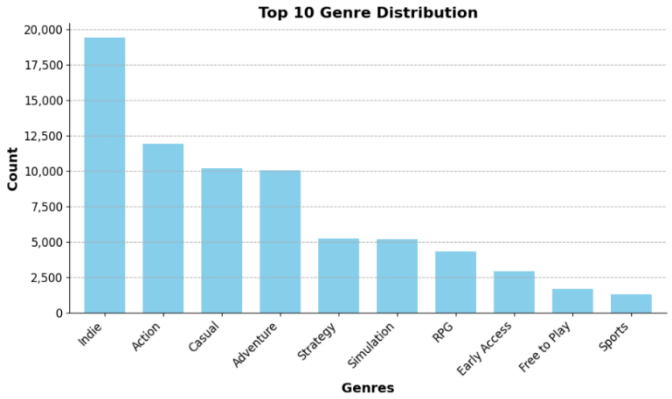

Fig. 1. Distrubtion of the top 10 most common genres in our dataset

B. *Release Year Distribution*

We also examined the frequency of different release dates in our merged data. This visual indicates a reasonable spread of release years and shows that the cutoff point for our data was in 2019.

GRAPH II. RELEASE YEAR DISTRIBUTION

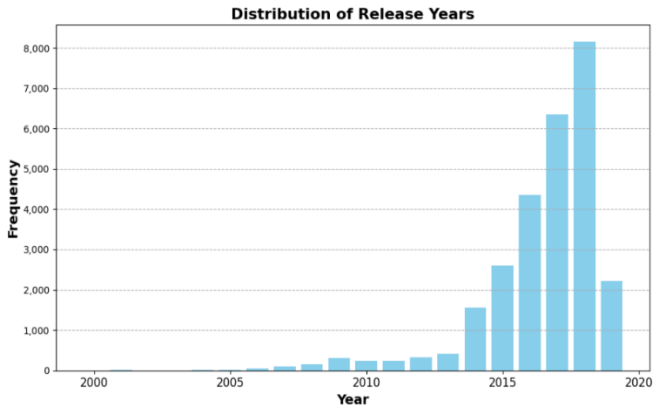

Fig. 2. Distribution of game release years in our dataset

IV. SENTIMENT ANALYSIS OBSERVATIONS

A. *Method*

Our methodology for investigating sentiment analysis of individual words in Steam reviews involved several key steps. Initially, we calculated the PMI value for words in relation to positive and negative sentiment, quantifying their association with each sentiment category. Next, we determined the Positive PMI/Negative PMI ratio to assess the relative importance of words in expressing positive or negative sentiment.

To ensure the reliability of our findings, we applied a minimum occurrence threshold, excluding words that appeared less than 25 thousand times. This filtering focused our analysis on frequently occurring words, which are more representative of common language used in Steam reviews. We then manually categorized the identified words into terms representing general quality and terms representing more specific characteristics.

B. *Positive Associations*

Many of the words with the highest ratio of positive PMI to negative PMI were vague terms relating to the quality of the game. These words such as "Best," "Awesome," and "Great." While not particularly surprising, these findings support the effectiveness of our methodology in capturing and identifying words indicative of positive sentiment.

TABLE I. QUALITY WORDS ASSOCIATED WITH POSITIVE REVIEWS

Base Form Word	***PMI Ratio***	***Total Word Occurrences***
Best	1.30	520,783
Awesome	1.27	185,688
Amaze	1.26	237,977
Fantastic	1.21	82,342
Great	1.21	840,629
Excellent	1.20	73,974
Perfect	1.13	80,254
Epic	1.13	37,247
Solid	1.09	72,572
Nice	1.06	255,681

Fig. 3. Base form of words regarding quality that are associated with positive reviews

More interestingly, we found several words that appear to imply user preferences for specific game characteristics, such as "Addict," "Beautiful," and "Soundtrack." These terms appear to reflect aspects that players value in games, potentially shedding light on user preferences regarding game characteristics.

TABLE II. CHARACTERISTIC WORDS ASSOCIATED WITH POSITIVE REVIEWS

Base Form Word	***PMI Ratio***	***Total Word Occurrences***
Addict	1.30	78,136
Beautiful	1.16	110,596
Soundtrack	1.14	70,359
Unique	1.13	108,589
Challenge	1.10	146,708
Replay	1.08	88,108
Atmosphere	1.08	64,433
Realistic	1.07	42,901
Immersive	1.07	43,627
Storyline	1.06	49,923

Fig. 4. Base form of words regarding game characteristics that are associated with positive reviews

The results from Table II provide potential insights into user preferences regarding various aspects of games. Firstly, there are words that suggest a preference for the artistic qualities of a

game, exemplified by terms like "Beautiful," "Soundtrack," and "Storyline." It is also noteworthy that the words "Realistic" and "Immersive" have relatively high PMI ratios. This indicates that players appreciate games that provide a sense of realism and immersion.

Interestingly, the term "Challenge" exhibits one of the highest PMI ratios among the listed words. This suggests that players value games that offer a level of difficulty and require strategic thinking or skill.

C. Negative Associations

Like our analysis of words with a positive review association, several of the words with the lowest ratio of positive PMI to negative PMI encompassed vague terms pertaining to the quality of the game. Examples of such words include "Worst," "Waste," and "Terrible." While the presence of these words might not come as a surprise, their inclusion in our findings reaffirms the validity of our methodology in identifying words indicative of negative sentiment.

TABLE III. QUALITY WORDS ASSOCIATED WITH NEGATIVE REVIEWS

Base Form Word	*PMI Ratio*	*Total Word Occurrences*
Worst	0.75	52,494
Waste	0.75	77,311
Terrible	0.79	54,152
Horrible	0.79	36,418
Poor	0.80	41,444
Crap	0.84	28,943
Stupid	0.85	42,094
Bad	0.86	248,894
Suck	0.87	52,703
Lack	0.88	106,399

Fig. 5. Base form of words regarding quality that are associated with negative reviews

We also discovered a range of words that seem to imply user dissatisfaction towards specific game characteristics. Examples of such words include "Crash," "Bore," and "Grind." These terms suggest that players may have encountered issues or experienced a lack of fulfillment in certain aspects of the game. The presence of these words provides potential insights into the aspects that players find problematic or unsatisfactory.

TABLE IV. CHARACTERISTIC WORDS ASSOCIATED WITH NEGATIVE REVIEWS

Base Form Word	*PMI Ratio*	*Total Word Occurrences*
Crash	0.82	103,138
Money	0.83	173,799
Bore	0.83	122,261
Fix	0.84	144,475
Pay	0.85	127,895
Bug	0.86	115,754
Patch	0.87	54,578
Server	0.88	77,858
Grind	0.89	51,387
Issue	0.89	127,168

Fig. 6. Base form of words regarding game characteristics that are associated with negative reviews

The words in Table IV reveal aspects that players may find frustrating or disappointing within a gaming experience. Notably, terms like "crash," "bug," and "issue" highlight technical problems that players encounter, indicating the significance of smooth gameplay and a bug-free experience in shaping review sentiment.

Additionally, words like "money" and "pay" suggest potential frustrations related to the upfront cost or monetization models of games. On the other hand, words like "bore" and "grind" indicate dissatisfaction with gameplay that involves repetitive and tedious tasks, underscoring the need for engaging and varied game mechanics to mitigate negative sentiment and maintain player interest.

V. MACHINE LEARNING EXPERIMENTS

A. Tools

- Python: We used Python (version 3.10.10) to perform our data analysis and modeling.
- Pandas: We used the Pandas library for data manipulation and analysis.
- NumPy: We used the NumPy library for numerical computation and analysis.
- Scikit-learn: We used the Scikit-learn library for machine learning modeling and evaluation.
- NLTK: We used the Natural Language Toolkit library for natural language processing tasks such as tokenization, stemming, lemmatization.

B. Machine Learning Models

Our machine learning experiments focused on the classification of game reviews as positive or negative based on the text, genre, and release date information. We employed logistic regression and decision tree models to accomplish this task. By comparing the performance of these two models, we aimed to assess their effectiveness in sentiment classification tasks and their suitability for analyzing game review data. Through these experiments, we aimed to gain insights into the strengths and weaknesses of each model in capturing the nuances of sentiment expressed in game reviews.

C. Baseline

We chose to use the majority class to establish a baseline performance. The majority class refers to the class that appears most frequently in the dataset. In our case, the majority class is positive, which makes up approximately 81% of the dataset.

Since the majority class is much larger than the other classes, it is a good choice for the baseline, as any model that performs worse than the majority class baseline is not useful.

D. Evaluation

To evaluate the performance of our models, we used standard evaluation metrics such as accuracy, precision, recall, and F1-score. These metrics enabled us to measure the effectiveness of each algorithm and compare their performance to determine which algorithm produced the most accurate and reliable predictions. We significantly prioritized F1-score because we prioritized our model's capability to predict both positive and negative reviews accurately. Furthermore, we used cross-validation techniques to validate the performance of our models and ensure they were not overfitting to the training data.

To gain a deeper understanding of the predictive performance of our models, we also made use of confusion matrices as part of our evaluation process. Confusion matrices allowed us to examine the distribution of predicted labels against the true labels, providing insights into the types of errors made by our models. This information provided a more comprehensive assessment of the models' predictive capabilities.

VI. Logistic Regression

A. Reason for choosing this model

Logistic regression is a well-suited model for the task of predicting sentiment because it handles binary classification problems effectively. Additionally, logistic regression is known for its computational efficiency, making it a practical choice for training models with larger datasets such as ours. Overall, it offers a balance between simplicity and performance, making it a valuable choice for sentiment classification in this context.

B. Training Process

We utilized the LogisticRegression class from scikit-learn to train our models. During the training phase, we explored the impact of the class_weight parameter on the performance of the logistic regression model. This was done to prevent overpredicting the majority class. By iterating through different values of the class_weight parameter, we systematically tested and evaluated the model's performance across a range of weightings assigned to the positive and negative classes. This allowed us to identify the optimal weight configuration that yielded the best F1-score, ensuring a balanced consideration of precision and recall in our sentiment classification task. The class_weight values that yielded the best F1 score were 0.38 and 0.62 for positive and negative reviews, respectively.

C. Best Results Achieved with Logistic Regression

Our best logistic regression model achieved solid performance with an accuracy of 85%, slightly surpassing our baseline accuracy of 81% and indicating some ability to correctly classify sentiment. The F1-score, which harmonizes precision and recall, stands at 91%, indicating a good balance between correctly identifying positive and negative sentiments.

TABLE V. Logistic Regression Confusion Matrix

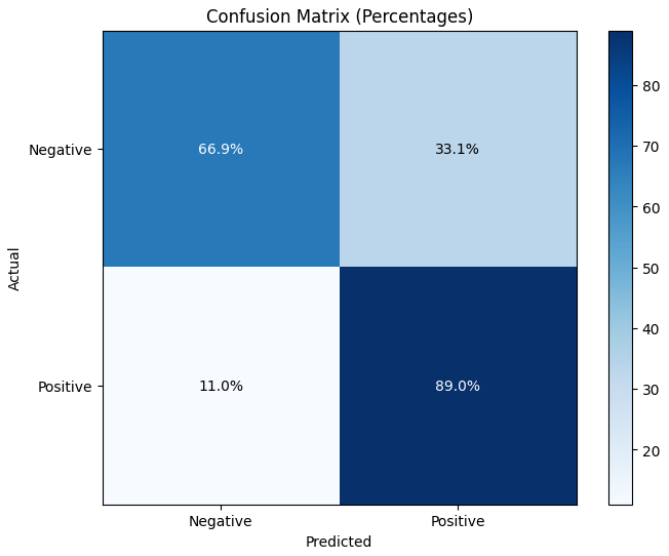

Fig. 7. Confusion Matrix representing accurcy of our logistic regression model with respect to each class

As seen in Table V, we were able to achieve considerable accuracy in predicting both the majority and minority class. These results indicate substantial improvement of the logistic regression model over the baseline, highlighting its efficacy in accurately predicting sentiment based on the review text, release date, and genre information.

VII. Decision Tree

A. Reason for choosing this model

Decision Trees can capture complex patterns between features, which should allow them to effectively represent relationships between our textual features and the sentiment labels. Decision Trees also provide interpretability by generating intuitive and easily interpretable decision rules. Furthermore, Decision Trees can handle large datasets efficiently, making them practical for sentiment classification tasks with substantial amounts of data. Overall, Decision Trees offer a balanced combination of interpretability, flexibility, and performance.

B. Training Process

For our decision tree model, we employed the DecisionTreeClassifier class from scikit-learn. During the training phase, we conducted a systematic exploration of the model's performance by iterating over different values of the max_depth and class_weight parameters. The max_depth parameter controls the maximum depth of the decision tree, while the class_weight parameter was adjusted to address the class imbalance and ensure accurate predictions for both positive and negative classes.

Systematically varying the max_depth and class_weight values allowed us to identify the optimal combination that

resulted in the best F1-score. In our experiments, the decision tree model with a max depth of 30 and class weights of 0.20 and 0.80 for positive and negative reviews, respectively, achieved the highest F1-score. This configuration optimized the model's ability to handle the class imbalance and make accurate predictions for both positive and negative sentiments in our dataset.

C. Best Results Achieved with Decision Tree

Our decision tree model resulted in lackluster performance. The accuracy of this model was 81%, roughly equal to the accuracy of our baseline. On the other hand, we were able to correctly classify most reviews for both the majority and minority class. This model resulted in an F1-score of 88%, slightly below what we achieved with our logistic regression model. Overall, our decision tree model scored lower in every metric than our logistic regression model.

TABLE VI. DECISION TREE CONFUSION MATRIX

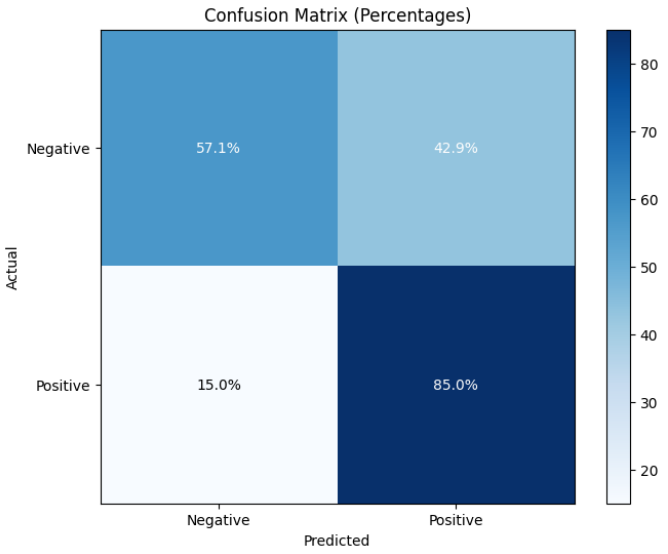

Fig. 8. Confusion Matrix representing accurcy of our decision tree model with respect to each class

One possible reason for the relatively poor results of this model is that decision trees are prone to overfitting, especially when dealing with complex datasets. Since our data contained many features and possibly had intricate relationships between them, the decision tree might have struggled to capture the underlying patterns effectively.

VIII. CONCLUSION

In this study, we examined words implying potential characteristics associated with positive and negative game reviews. Our findings revealed valuable insights into user preferences, highlighting the appreciation for artistic qualities, the importance of realism and immersion, and the value of challenging gameplay. Conversely, our analysis also shed light on potential frustrations and disappointments, such as technical issues and gameplay mechanics that may lead to negative sentiments. Furthermore, our logistic regression model demonstrated success in accurately predicting review sentiments.

Nevertheless, it is important to acknowledge the limitations of our study. Due to constraints in computing power and time, we focused on examining individual word associations with review sentiment. Further research is warranted to explore genre and release year specific dynamics in sentiment. Additionally, future research should consider exploring multi-word sequences, as this may uncover more nuanced patterns and provide a deeper understanding of player preferences. By addressing these limitations, future studies can expand upon our findings, leading to more nuanced insights into user preferences and trends in the gaming industry.

Overall, this study provides insights into both positive and negative aspects of game reviews. These findings have practical implications for developers striving to create engaging and positively received games while mitigating potential frustrations expressed by players.

REFERENCES

[1] Maranhão, A., Marjani, A., Marpaung, D., Pooya, P. (2022). Steam Reviews [Data set]. Kaggle. https://www.kaggle.com/datasets/andrewmvd/steam-reviews

[2] Davis, N. (2019). Steam Store Games [Data set]. Kaggle. https://www.kaggle.com/datasets/nikdavis/steam-store-games