Sentiment Analysis of Steam Reviews: Informing Live Service Continuation, Sequel, or Prequel Development in Game Studios

Dimas Putra Aryawan
School of Computer Science
Bina Nusantara University
Jakarta, Indonesia 11480
dimas.aryawan@binus.ac.id

Michael Vincentius Ginting
School of Computer Science
Bina Nusantara University
Jakarta, Indonesia 11480
michael.ginting@binus.ac.id

Richard Bryan Antonius
School of Computer Science
Bina Nusantara University
Jakarta, Indonesia 11480
richard.antonius@binus.ac.id

Abstract*— User reviews have become a major influence in the success or failure of video games on platforms like Steam. These reviews shape how potential buyers view a game, often affecting whether they choose to make a purchase. But not all reviews are trustworthy—some can be biased, manipulated, or simply misleading. To better understand and evaluate the sentiment behind game reviews, this study explores the use of machine learning models, including Naive Bayes, Support Vector Machine (SVM), Random Forest, and Long Short-Term Memory (LSTM). Using real user review data from Steam, we applied natural language processing techniques like tokenization and TF-IDF to prepare the data for training and testing. Among the four models, SVM delivered the most balanced and accurate results, while LSTM showed strong potential for capturing deeper emotional context. The findings suggest that applying sentiment analysis to user reviews can give game developers valuable insight into what players genuinely enjoy or dislike, helping them improve future titles and make smarter development decisions. This approach can ultimately lead to better games, more satisfied players, and stronger trust between developers and their communities.

Keywords*— Machine Learning, Naive Bayes, SVM, Random Forest

I. Introduction

Customers who write online reviews play a crucial role in shaping consumer decisions, especially on digital distribution platforms like Steam, where users rely on feedback from others to determine whether a game is worth buying. Reviews provide insights into various aspects, from gameplay quality and technical performance to developer support and post-launch updates. They help potential buyers make informed choices and set realistic expectations about the games they are considering[9].

However, game descriptions on Steam do not always match the actual player experience. Many users purchase games based on trailers, marketing materials, or developer promises, only to find that the final product falls short. While reading reviews could help avoid disappointment, not all buyers take the time to do so[17]. This often leads to frustration, negative reviews, and a loss of trust in both the game and its developer.

Even though reviews are meant to guide players, they are not always reliable. Some games receive overwhelmingly positive ratings that suggest they live up to their promises, yet once played, they turn out to be disappointing. This has raised concerns about review manipulation, where developers or publishers inflate ratings, suppress negative feedback, or use fake aeccounts to boost their game's reputation[16]. When trust in reviews erodes, it becomes harder for players to make informed purchases. In extreme cases, failing to meet player expectations can have severe consequences—not just for a single game but for an entire studio. Many game companies, even those with years of experience, have been forced to shut down because they

couldn't keep players satisfied. No matter how long a studio has been around, losing player trust can be the beginning of the end[12].

To address these challenges, sentiment analysis has become an important tool for evaluating the authenticity and tone of player reviews[13]. A key aspect of sentiment analysis is building a lexicon that helps classify words as positive or negative, making it possible to analyze feedback more systematically. By applying different sentiment analysis methods, specifically Naive Bayes,SVM, Random Forest, and LSTM. We can better understand which approach provides the most accurate results. This not only helps players make more informed decisions but also gives developers valuable insights into what their audience truly wants[9]. A more transparent and reliable review system could help prevent misleading marketing, rebuild player trust, and ultimately, support the longevity of game studios in an increasingly competitive industry.

II. Literature review

A. User Reviews

With the growing market in video games the importance of peer user reviews is on the incline, as it has become an essential component in running a successful business venture[10,11]. Online reviews provide valuable information for consumers, particularly when purchasing goods like video games that rely on information about past user experiences[6]. Sales and image of a product is heavily impacted by online product reviews[11, 14]. Which put a big emphasis on the importance of user feedback in the form of online reviews.

B. Steam Platform

Personal computer based games are an up and coming powerhouse in the world of softwares. The major issue on user perceived experience in gaming is the distribution of games[15]. This is where steam takes part which itself is an online based digital distribution platform for products such as digital rights management, multiplayer experience, video streaming, and social networking services[7].

C. *Naive Bayes*

Bayesian network classifiers are widely used in supervised classification, with the Naïve Bayes classifier being one of the most well-known. This probabilistic classifier is based on Bayes' theorem and operates under the assumption of strong (Naïve) independence [2]. In paper [2], Naïve Bayes was applied to sentiment classification using hotel and film review datasets. However, not all datasets are suitable for the Naïve Bayes method in sentiment analysis. The study found that while Naïve Bayes performed effectively for film reviews, its performance was less reliable for hotel reviews.

D. Support Vector Machine

Support Vector Machines (SVM) belong to the category of supervised learning, meaning the model is trained beforehand to classify data by splitting it into training and testing sets. SVM works by identifying a separating function that distinguishes between two datasets belonging to different classes. The core idea behind SVM is to determine the optimal hyperplane that serves as a boundary between two classes in the input space while maximizing the margin between them[20]. Paper implements SVM to analyze the sentiments of customer reviews on online marketplaces, enhanced with the usage of Synthetic Minority Oversampling Technique (SMOTE) and Tomek Links which improves the overall classification accuracy of the model by a notable amount 20 .

E. Random Forest

Random Forest is a classification model introduced by Breiman to enhance the predictive performance of classification trees. According to Breiman, Random Forest consists of a collection of MMM tree-structured classifiers, where each tree is built from a smaller dataset containing n objects and p attributes.

The n objects are chosen using a Bagging approach, such as Bootstrap Sampling, while the p attributes are randomly selected without replacement. Each unique combination of these selections forms an individual decision tree. Once multiple trees are generated, their predictions are aggregated, with the final classification determined by the most frequently predicted class among the M trees.

Similar to classification trees, various Random Forest algorithms exist, differing in how samples are selected and which classification tree algorithm is employed[18]. Regarding the usage of the Random Forest method, paper [18] has come to a conclusion that using the Random Forest method exhibits an overfitting problem regardless of the amount of tree used. This may indicate that the dataset used was too specific and unique, such as using a pre-trained model that renders the model only being able to pinpoint the result on the specific data. Despite this limitation, Random Forest contributed to identifying the most important classification terms, with significant overlap in key terms across different classification methods.

F. LSTM

Long Short-Term Memory (LSTM) is a neural network architecture designed to handle, predict, and classify data over extended time periods, even when dealing with insurmountable challenges. As an improved version of the traditional recurrent neural network (RNN), LSTM is widely utilized in scientific research[19]. Paper [19] presents a sentiment classification approach for short social

media texts using LSTM. By leveraging word embeddings like Word2Vec, the model effectively captures contextual semantics. The study highlights that deep learning methods, particularly LSTM, achieve better sentiment classification performance with larger training datasets. Future research should explore "community"-specific sentiment lexicons and larger datasets to enhance classification accuracy for specific user behaviors.

III. METHOD

The stages of this research include:

A. Data Collection

The data used in this research was downloaded from https://www.kaggle.com/datasets/aqibrehmanpirzada/video-games-reviews/data as the dataset for model creation, and https://www.kaggle.com/datasets/andrewmvd/steam-reviews as the target

These 2 datasets contain game reviews that are officially registered in the steam platform.

The first dataset that used for training model contain variables about information from The Terraria game review as the target:

- app_id: Unique integers that represent each particular steam game that being reviewed
- app_name: String variable representing the steam game name that being reviewed
- review_text: String variable containing the content of the review
- review_score: Integer variable representing whether the reviewer recommends the game or not
- review_votes: Integer variable representing whether the reviewer was recommended by another user or not

The second dataset that used as a review model contain variables as a foundation to create the model:

- reviewText: String variable containing the content of the review
- overall: Float variable as an score from 1-5, whether the review is bad (1-2) or neutral (3) or a good review (4-5)

B. Data pre-Processing

Since this is a Natural language Processing (NLP) based model, we use the Natural Language Toolkit (NLTK). The steps that are involved for our data pre-processing are Remove Null, Remove Punctuations and Stopwords, Tokenize and Lowercasing, and lastly Label Encoding.

a) Remove Null Values: Any rows or entries in the dataset that have null or missing text data were removed to make sure data integrity and avoid any errors during further modelling.

b) Remove Punctuations and Stopwords: All punctuation marks were removed from the text to eliminate unnecessary data for processing. And also, common stopwords (e.g., “and,” “the,” “is”) that don’t have any meaningful information are also removed to only focus on significant and meaningful words.

c) Tokenization and Lowercasing: Tokenization is a process of breaking up a paragraph into words or can be called as tokens. It performs a crucial task in any natural language processing. It basically creates a list of arrays of words. For tokenizing the data, the NLTK library from Python is used in this work. And all tokens were converted to lowercase to maintain consistency and avoid treating the same word differently due to case difference (e.g., "Apple" vs. "apple").

d) Label Encoding: In the original dataset, game ratings range from 1 to 5. For our mode, we changed the labels to make it easier to visualize if a game is truly recommended or not. Ratings from 1 to 4 were changed to 0 (not recommended), and a rating of 5 was changed to 1 (recommended). This helps the model to only focus on reviews where people really liked the game. The goal is to help identify which key points / parts of the game should be removed, improved, changed, or kept in a future sequel, based on strong feedback from users.

C. Features extraction

In our model, we use TFIDF Vectorizer. TF-IDF means term frequency inverse document frequency. In this method, some semantic information is preserved as uncommon words are given more importance than common words. For Example: let’s take a sentence from the dataset that we use for the model. ‘This game is so fun!.’ Here ‘fun’ will have more importance than ‘This' or ‘game.’ This ‘game’ word can be present in a maximum number of the reviews in the documents. So that’s why this word frequency will be high. When calculating TF-IDF, the words that have high frequency will have a low value in the vectorizer. It only uses words that have a higher TF-IDF value as the input.

D. Training and Testing of Models

In this project, we trained and tested four different models—Naive Bayes, LSTM, Random Forest, and SVM—to classify whether a game review is positive or negative. I used a dataset of reviews from the game *Terraria* to train the models, where each review included the review text and whether the reviewer recommended the game. Before feeding the data into the models, I cleaned the text by removing things like punctuation and stopwords, then turned the words into numbers using TF-IDF for Naive Bayes, Random Forest, and SVM. For LSTM, which works better with sequences, I tokenized and padded the text. Each model brings a different strength: Naive Bayes is simple and fast, LSTM understands the context of words, Random Forest builds decisions from multiple trees, and SVM tries to draw the best line to separate positive from negative reviews. After training, I tested the models on a different set

of Steam reviews, converting scores into positive or negative labels.

E. Prediction using Naive Bayes Model

One of the classification models used in our machine learning model is Naïve Bayes. It is a simple yet powerful probabilistic model, especially effective for categorizing and classifying text. This model is one of the most common models because other people know how effective it is. Naïve Bayes is based on Bayes' Theorem and assumes that features are independent of each other, which simplifies the computation. It helps us calculate probabilities of an event occurring based on the data given. Bayes Theorem is stated as the following equation:

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$$

where

- A is called the proposition and B is called the evidence.
- P(A) is called the prior probability of proposition and P(B) is called the prior probability of evidence.
- P(A|B) is called the posterior.

F. Prediction using SVM

SVM is a classification method that works by finding the best boundary to separate data into classes, such as positive or negative sentiment. Even though it's a relatively simple algorithm, it's very effective, especially when combined with proper text preprocessing and feature extraction like TF-IDF. SVM is known for its robustness in high-dimensional spaces and its ability to generalize well, which makes it suitable for text classification tasks. Thanks to its focus on maximizing the margin between classes, SVM can make accurate predictions while avoiding overfitting, making it a strong method for sentiment analysis.

G. Prediction using Random Forest and LSTM

Random forest is a sensitive method, because just a small change to the data, it will change the growth of the random forest and makes it so different, but that make it this method is really accurate than other and do to the advantage of random forest that can handle complex data and its potential to mitigate overfitting, it's really good for classification. By leveraging LSTM neural networks, this method enables more complex text processing to identify whether a review is positive or negative.

IV. RESULT AND DISCUSSION

A. Performance Metrics

The four models—Naive Bayes, Support Vector Machine (SVM), Long Short-Term Memory (LSTM), and Random Forest—were evaluated on a held-out test set of Steam reviews. Table 1 summarizes their key performance metrics.

Model	Accuracy	Precision	Recall	F_1-Score
Naive Bayes	97 %	0.97	1.00	0.98
SVM	90.82 %	0.94	0.93	0.93
LSTM	89.48 %	0.92	0.93	0.92
Random Forest	85.73 %	0.85	0.96	0.90

B. Confusion Matrices

Naive Bayes:
[[0 807]
[7 24597]]

SVM:
[[4529 723]
[833 10856]]

LSTM:
[[4331 921]
[861 10828]]

Random Forest:
[[3265 1987]
[431 11258]]

C. Discussion

Overall Accuracy and Robustness

The updated Naive Bayes model significantly outperforms the other models with a striking accuracy of 97% and an F_1-score of 0.98. This suggests that under the right feature engineering or preprocessing pipeline, even a relatively simple probabilistic model can perform competitively against more complex architectures.

Precision–Recall Trade-offs

Naive Bayes achieves perfect recall (1.00), meaning it successfully identifies all positive sentiments in the test set. This comes with a very high precision (0.97), indicating a near-perfect ability to avoid false positives. Meanwhile, SVM maintains its strong balance (precision = 0.94, recall = 0.93), making it suitable when both error types matter. Random Forest continues to provide exceptional recall (0.96), ideal when minimizing false negatives is the priority.

Sequential Context in LSTM

LSTM's F_1-score of 0.92 reflects its ability to model the sequential nature of language. While it falls slightly behind SVM and the updated Naive Bayes, it remains a viable choice in applications where understanding word order is critical.

Baseline Re-evaluation with Naive Bayes

The performance leap in the updated Naive Bayes model highlights the importance of data preprocessing, feature extraction, or tuning. Although it traditionally suffers from independence assumptions, these limitations can be mitigated effectively, elevating the model from a simple baseline to a top performer.

Implications for Game Studio Decision-Making

Live Service Continuation: Naive Bayes and Random Forest offer strong recall, making them reliable tools to capture all positive feedback for iterative content updates.

Sequel/Prequel Development: SVM's balanced precision and recall make it a dependable model to gauge widespread player satisfaction for major investment decisions.

Sentiment Trends Over Time: High-accuracy models like the improved Naive Bayes can be used to monitor sentiment evolution across updates or post-release events.

Limitations and Future Work

Model Generalization: While Naive Bayes now performs well on this dataset, further testing on diverse platforms (e.g., Reddit, Metacritic) is needed.

Aspect-based Sentiment Analysis: Incorporating topic modeling or feature-level classification could improve interpretability for developers.

Hybrid Methods: Future work could explore ensemble or stacking models that combine the strengths of different classifiers for even more robust results.

V. CONCLUSION

This study shows how crucial trustworthy user evaluations are on digital distribution sites like Steam, where user ratings have a big impact on decisions to buy. Sentiment analysis is a useful tool for helping game producers to know what their players needed and/or wanted in their upcoming game sequel. This study compared many machine learning models, including Random Forest, Naive Bayes, Support Vector Machine (SVM), and Long Short-Term Memory (LSTM), in order to identify the best technique for categorizing user attitudes from game reviews.

The findings show that each model has distinct advantages: LSTM is excellent at capturing context in sequential data, Random Forest manages complexity but runs the risk of overfitting, SVM provides high accuracy in structured data, and Naive Bayes is quick and easy to understand. Of them, SVM scored out for striking a compromise between accuracy and generalization, while LSTM had the greatest promise for deciphering complex feelings in text provided sufficient data is available.

In the end, including strong sentiment analysis into review systems can give developers useful information for them to improve their upcoming games. Better game production, better marketing strategies, and better-informed consumer choices can result from this, which will help individual games and the studios that produce them succeed in the long run.

References

[1] H. ASH Basari, "Generate Contextual Insight of Product Review Using Deep LSTM and Word Embedding," 2020, doi: 10.1088/1742-6596/1577/1/012006.

[2] Dey, L., Chakraborty, S., Biswas, A., Bose, B., & Tiwari, S, "Sentiment analysis of review datasets using naive bayes and k-nn classifier", 2016, doi : https://doi.org/10.5815/ijieeb.2016.04.07

[3] R. B. T. Dhika Malita Puspita Arum, "Klasifikasi Rating Otomatis pada Dokumen Teks Ulasan Produk Elektronik Menggunakan Metode N-gram dan Naïve Bayes," Sep. 2020, doi: 10.32493/informatika.v5i3.6110.

[4] M. Lankes and A. Stöckl, "Game Reviews Reviewed: A Game Designer's Perspective on AI-generated Game Review Analyses," 2023, doi: 10.1109/CoG57401.2023.10333238.

[5] M. Viggiato, D. Lin, A. Hindle and C. -P. Bezemer, "What Causes Wrong Sentiment Classifications of Game Reviews?," Sep. 2022, doi:10.1109/TG.2021.3072545.

[6] V. Brodschneider and J. Pirker, "On the Influence of Reviews on Play Activity on Steam - A Statistical Approach," 2023, doi: 10.1109/CoG57401.2023.10333235

[7] R. Buettner, M. Blattner and W. Reinhardt, "Internet Gaming more than 3 Hours a Day is Indicative and more than 5 Hours is Diagnostic: Proposal of Playing Time Cutoffs for WHO-11 and DSM-5 Internet Gaming Disorder Based on a Large Steam Platform Dataset," 2020, doi: 10.1109/BigDataService49289.2020.00037

[8] M. Bilal, M. Marjani, M. I. Lali, N. Malik, A. Gani and I. A. T. Hashem, "Profiling Users' Behavior, and Identifying Important Features of Review "Helpfulness"," 2020, doi: 10.1109/ACCESS.2020.2989463

[9] S. Park, J. Cho, K. Park, and H. Shin, "Customer sentiment analysis with more sensibility" 2021, doi: 10.1016/j.engappai.2021.104356

[10] . Guo, Y. Liu, Y. Ouyang, V. W. Zheng, D. Zhang and Z. Yu, "Harnessing the Power of the General Public for Crowdsourced Business Intelligence: A Survey," 2019, doi: 10.1109/ACCESS.2019.2901027

[11] X. Li, C. Wu, and F. Mai, "The effect of online reviews on product sales: A joint sentiment-topic analysis," 2019, doi: 10.1016/j.im.2018.04.007

[12] Hothead Games, "Hothead closed its doors for the last time on Friday, December 13, 2024." *LinkedIn* , accessed Mar. 7, 2025.[Online]Available: https://www.linkedin.com/feed/update/urn:li:activity:7275748337667002368/

[13] H. Nguyen, A. Veluchamy, M. Diop, R. Iqbal, "Comparative Study of Sentiment Analysis with Product Reviews Using Machine Learning and Lexicon-Based Approaches", 2018, Available: https://scholar.smu.edu/datasciencereview/vol1/iss4/7/

[14] Nisar, Tahir M. & Prabhakar, Guru & Ilavarasan, P. Vigneswara & Baabdullah, Abdullah M. "Up the ante: Electronic word of mouth and its effects on firm reputation and performance," 2020, doi: 10.1016/j.jretconser.2018.12.010

[15] Lin, Dayi & Bezemer, Cor-Paul & Zou, Ying & Hassan, Ahmed E. An Empirical Study of Game Reviews on the Steam Platform. 2019, doi: 10.1007/s10664-018-9627-4.

[16] J. Coronado-Blázquez, “ A NLP Approach to "Review Bombing" in Metacritic PC Videogames User Ratings “ 2024, Available: https://arxiv.org/abs/2405.06306

[17] E. Maslowska, E. C. Malthouse , V. Viswanathan, “Do customer reviews drive purchase decisions? The moderating roles of review exposure and price” 2017, doi: 10.1016/j.dss.2017.03.010

[18] Jessica Kubrusly, Ana Luiza Neves, Thamires Louzada Marques, “A Statistical Analysis of Textual E-Commerce Reviews Using Tree-Based Methods”, June 2022, doi: https://doi.org/10.4236/ojs.2022.123023

[19] J.-H. W. Long Wang, “An LSTM Approach to Short Text Sentiment Classification with Word Embeddings,” 2018, [Online]. Available: https://aclanthology.org/O18-1021.pdf

[20] D. I. Sumantiawan, J. E. Suseno, and W. A. Syafei, "Sentiment Analysis of Customer Reviews Using Support Vector Machine and Smote-Tomek Links For Identify Customer Satisfaction," *Jurnal Sistem Informasi Bisnis*, vol. 13, no. 1, pp. 1-9, Jun. 2023. https://doi.org/10.21456/vol13iss1pp1-9