Predicting Video Game Publisher Stock Returns Using Game-Derived Features: A Machine Learning Approach

Torrin Pataki
ID: 77268944

Abstract

The rise of alternative data has introduced new possibilities for financial forecasting, particularly in leveraging high-frequency, industry-specific metrics. This research explores the predictive capability of video game-derived performance metrics for forecasting short-term stock returns of publicly traded game publishers. We introduce a hierarchical feature engineering pipeline that transforms granular game data into publisher-level features, capturing engagement, monetization, sentiment, and portfolio dynamics. Our methodology extends the triple-barrier labelling method with a weighting scheme prioritizing high-impact trades and implements purged cross-validation to prevent forward-looking bias. Through comparative analysis across six major publishers, we demonstrate that neural networks consistently outperform traditional machine learning approaches, achieving validation errors as low as 0.1000 in specific cases. While technical indicators provide limited additional value for most publishers, we identify interesting cases where their inclusion improves model performance, suggesting potential synergies under specific market conditions. Our work contributes both methodological innovations and practical implementations, with source code for the enhanced triple-barrier labelling and purged cross-validation frameworks made publicly available to support future research.

1 Introduction

The video game industry is an industry that produces an extensive amount of high-granularity data about player behaviour and interaction with the game environment [Kim et al., 2023]. Such data presents an opportunity to derive various consumer metrics that may serve as leading indicators of company performance. Traditional financial analysis relies heavily on quarterly reports and aggregated metrics, which often lag behind real-time market dynamics. Our research investigates whether alternative data sources—specifically, detailed game performance metrics—can enhance our ability to predict stock returns for video game publishers.

Recent research by Dessaint et al. [2024] demonstrates that alternative data sources excel in short-term financial forecasting by providing timely insights into market behavior. Building on this foundation, we develop a comprehensive framework for aggregating and analyzing game-specific metrics as potential predictors of publisher stock performance. Our work makes several key contributions:

- **Feature Engineering Pipeline:** We develop a hierarchical system that transforms granular game-level metrics into meaningful publisher-level features, incorporating both absolute performance measures and relative competitive dynamics.
- **Enhanced Triple-Barrier Labeling:** We extend Lopez de Prado's [2018] methodology by introducing weighted labels based on price movement magnitude, providing a more nuanced approach to classifying trading signals.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

- **Model Comparison Framework:** We implement and evaluate multiple machine learning architectures, demonstrating the superiority of neural networks in capturing complex relationships between game metrics and stock returns.
- **Implementation Contributions:** We provide public implementations on Github [1] of our purged cross-validation methodology and triple-barrier labeling system, contributing to the reproducibility and advancement of financial machine learning research.

2 Related Work

Our research bridges three primary streams of literature: financial machine learning methodology, alternative data applications, and video game analytics. This synthesis creates a novel framework for predicting publisher stock returns using game-specific metrics.

2.1 Financial Machine Learning Foundations

Lopez de Prado [2018] introduced the triple-barrier method for labeling financial data, providing a robust framework for generating trading signals based on dynamic price movements. This approach addresses the limitations of fixed-threshold methods by adapting to market volatility. Our work extends this methodology by incorporating weighted labels that reflect the magnitude of price movements, particularly for cases where the vertical barrier is reached.

Advances in cross-validation techniques for financial data, particularly the purged k-fold method [Lopez de Prado, 2018], have highlighted the importance of preventing forward-looking bias in model evaluation. We implement and extend these techniques while adapting them to our weekly prediction horizon.

Htun et al. (2020) emphasize the importance of feature selection and dimensionality reduction techniques, such as PCA, for the typical high-dimensionality of financial prediction tasks. In our study, we incorporate PCA for logistic regression models to address the high dimensionality of game-derived features.

Deep learning methods, particularly LSTMs, have shown promise in financial time-series tasks due to their ability to capture temporal dependencies (Fahd et al., 2020). However, our results reveal that neural networks outperform LSTMs, challenging the conventional assumption that sequence models are inherently superior for time-based data.

2.2 Alternative Data in Financial Forecasting

Recent work by Dessaint et al. [2024] demonstrates that alternative data sources provide superior predictive power for short-term financial outcomes compared to traditional indicators. Their analysis shows that granular, high-frequency metrics often capture market dynamics before they manifest in conventional financial data. However, their work does not explore industry-specific alternative data sources.

2.3 Video Game Analytics

Kim et al. [2023] established strong relationships between player behavior metrics and game financial performance. Their work demonstrates that engagement statistics serve as leading indicators for monetization success, though their analysis remains at the individual game level. We scale these insights to publisher-level prediction by developing aggregation strategies that preserve signal while reducing noise.

2.4 Integration and Novel Contributions

Our work advances the current state of literature in several ways:

[1]https://github.com/Torr1n/Predicting-Video-Game-Publisher-Stock-Returns-Using-Game-Derived-Features

- We extend the triple-barrier method by incorporating magnitude-weighted labels for vertical barrier hits, providing more nuanced training signals for machine learning models.
- We develop a comprehensive feature engineering pipeline that transforms game-level metrics into publisher-level predictive features while maintaining temporal consistency.
- We demonstrate that neural networks can effectively capture non-linear relationships between game performance metrics and stock returns, outperforming traditional approaches across multiple publishers.

3 Problem Formulation

Given a set of publisher features $X_t \in \mathbb{R}^d$ at time t, we aim to predict the direction of stock returns $y_t \in \{-1, 1\}$, where:

$$y_t = \begin{cases} 1 & \text{if take-profit barrier is reached} \\ -1 & \text{if stop-loss barrier is breached} \\ \text{sign}(r_t) & \text{if vertical barrier is hit} \end{cases}$$

Here, r_t denotes the log return of the stock at time t.

4 Methodology

Our approach combines granular game data collection, hierarchical feature engineering, and robust model validation techniques. We detail each component of our methodology, providing theoretical justification and implementation specifics.

4.1 Data Collection and Processing

We analyze six major video game publishers listed on different global exchanges: SEGA (6460.T), Electronic Arts (EA), Ubisoft (UBI.PA), Square Enix (9684.T), Frontier Developments (FDEV.L), and CD PROJEKT RED (CDR.WA). The study spans 88 weeks from January 2023 to September 2024, with this span chosen to ensure access to hourly stock data granularity through yfinance.

Through the Gamalytic API, we collected comprehensive hourly data for each publisher. The data encompassed player engagement metrics such as concurrent user counts and average playtime, financial indicators including revenue and sales figures, user sentiment measured through review scores and volumes, and forward-looking metrics that captured follower growth and wishlist additions.

4.2 Feature Engineering Pipeline

Our hierarchical feature engineering process transforms granular game-level data into publisher-level predictive features through three distinct stages. This process is outlined visually in Appendix A and begins with the computation of base game-level metrics capturing key performance indicators. These metrics encompass player engagement measured through concurrent users and playtime statistics, monetization effectiveness quantified by revenue per active player and sales momentum, sentiment indicators derived from weighted review scores, and lifecycle stability metrics focusing on player retention and revenue sustainability.

The second stage introduces relative performance metrics through two complementary approaches. First, we compute genre-relative performance by calculating z-scores against games sharing similar genres. This calculation requires a minimum of 30 peer games to ensure the statistical validity of applying the central limit theorum, and uses Jaccard similarity between genre and tag sets to identify appropriate comparisons. Second, we evaluate audience-overlap performance by computing weighted z-scores based on player-base similarity, considering the top 100 games with the highest audience overlap as identified through the Gamalytic API.

The final stage aggregates these metrics to the publisher level through three distinct strategies. The first approach calculates revenue-weighted means and standard deviations across all features, ensuring that economically significant games receive appropriate emphasis in our publisher metrics. The second strategy computes age-adjusted metrics for base features, incorporating square root decay to balance the influence of mature titles against recent releases. The third method focuses on relative features, calculating outperformance ratios that capture the proportion of publisher revenue generated by games exceeding their peer benchmarks.

To capture temporal dynamics, we incorporate lagged features, moving averages, and exponential moving averages for each publisher-level metric. Additionally, we compute portfolio diversity measures using Herfindahl indices applied to both genre distribution and game revenue contribution. This comprehensive approach results in a rich feature set of 1367 dimensions per publisher, capturing both static and dynamic aspects of publisher performance.

4.3 Enhanced Triple-Barrier Labeling

We extend Lopez de Prado's triple-barrier method by introducing a novel weighting scheme for vertical barrier hits. The implementation follows:

Dynamic Barrier Placement:

- Upper Take-Profit barrier: $\mu + 1.5 \cdot \sigma$
- Lower Stop-Loss barrier: $\mu - 1.5 \cdot \sigma$ where μ and σ are weekly estimates of expected returns and volatility estimated using a rolling window over hourly price data

Weight Calculation: For cases where the vertical barrier is reached first, we compute weights based on the proximity of the terminal price to the nearest horizontal barrier. The weight is calculated as the ratio of the observed return r_t to the return that would have been required to hit the nearest barrier $r_{barrier}$, capped at 1: $w_t = \min(1, \frac{|r_t|}{r_{barrier}})$ This weighting scheme ensures that price movements that come close to breaching a horizontal barrier are given more importance during model training than those that remain near the starting price, offering 0 returns.

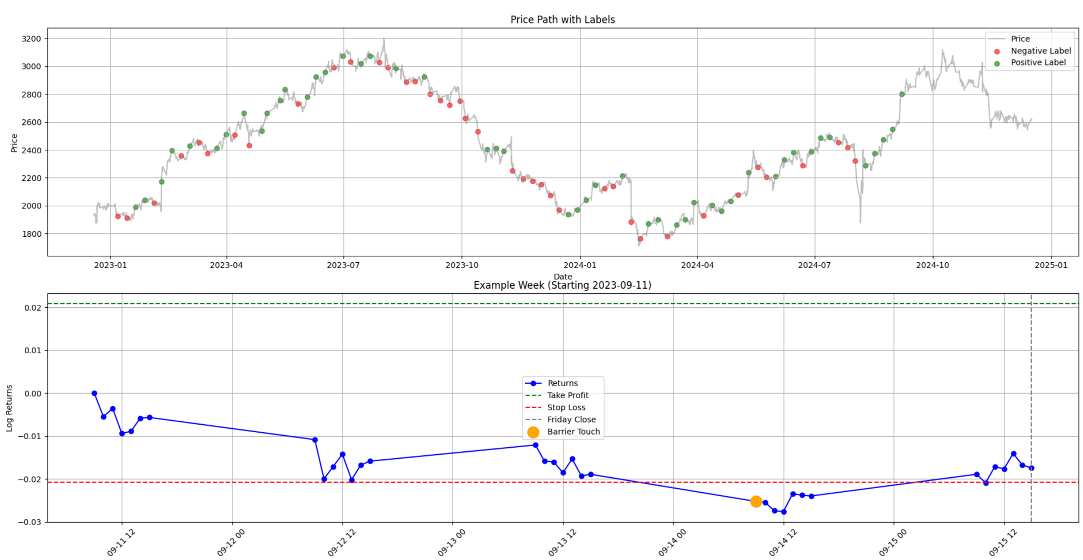


Figure 1: Triple-barrier labeling, where the bottom barrier is hit and we label the example -1.

4.4 Model Architectures

Our research evaluates five distinct machine learning architectures, each selected to test different hypotheses about the relationship between game-derived features and stock returns.

A Random Forest classifier serves as our first baseline model, implementing an ensemble of 500 decision trees with bootstrapped feature sampling. We optimize the maximum tree depth search by searching for a depth that balances model complexity with generalization capability, finding that relatively shallow trees with a maximum depth of 4 achieve optimal performance on our validation sets.

The second baseline employs logistic classification with L2 regularization, optimized through stochastic gradient descent. This model tests the hypothesis that a linear decision boundary might sufficiently capture the relationship between our features and stock movements. We searched over regularization strengths ranging from 10^{-4} to 10^{2} finding that a strength of 10^{-1} provided the optimal balance between bias and variance.

To address the high dimensionality of our feature space while maintaining non-linear modelling capability, we implement an RBF kernel logistic classifier with PCA preprocessing. This approach first reduces our feature space to capture 95% of variance, then applies the RBF transformation to model non-linear relationships. Grid search optimization identified optimal kernel width and regularization strength of 10.

Our neural network architecture consists of three hidden layers with 64 units each, employing ReLU activation functions and dropout regularization with a rate of 0.3. This configuration emerged from extensive experimentation with different architectures, showing superior ability to capture complex relationships in our high-dimensional feature space. We train the network using the Adam optimizer with a learning rate of 0.001, which provides stable convergence across different publishers. Detailed hyperparameter grid-search optimization graphs are presented in Appendix B.

The final architecture implements a three-layer LSTM network designed to capture temporal dependencies in our weekly data. Each layer contains 64 hidden units and incorporates dropout regularization. The model processes sequences of 12 weeks, balancing the need to capture long-term patterns against the limitations of our dataset size. We implement inverse square root learning rate decay, initializing at 0.1, to stabilize training and improve convergence.

4.5 Validation Framework

Our validation methodology implements two distinct approaches tailored to the temporal nature of financial data. For non-sequential models, including the Random Forest, Logistic Classification, and Neural Network architectures, we employ purged cross-validation following Lopez de Prado's [2018] framework. This approach uses 5-fold splitting with an 8-week purge window, chosen to match our longest feature lookback period. The purging process eliminates samples from the validation set that overlap with training data, preventing the information leakage that often compromises financial forecasting models.

For the LSTM model, where maintaining temporal ordering is crucial for capturing sequential dependencies, we implement a chronological validation scheme. This approach splits the data into an 80% training set and a 20% validation set, strictly preserving temporal order. While this method provides fewer validation samples than k-fold cross-validation, it better reflects the model's real-world performance characteristics by maintaining the temporal structure of the data.

For model evaluation, weighted classification error serves as our primary metric, incorporating the magnitude-based weights from our enhanced triple-barrier labeling scheme. This metric ensures that models are evaluated not just on directional accuracy but also on their ability to identify high-impact trading signals. We complement this with precision and recall metrics to understand the balance between false positives and false negatives, particularly important in financial applications where the cost of errors may be asymmetric. The F1 score provides a harmonic mean of precision and recall, offering a single metric that captures both aspects of model performance.

5 Results

Our experimental analysis examines model performance across multiple publishers and feature sets, using SEGA as our primary case study with full results provided on the public GitHub under the Results folder. We present detailed performance metrics, analyze the impact of technical indicators, and investigate cross-publisher generalization patterns.

5.1 SEGA Case Study

The neural network architecture demonstrates superior predictive capability for SEGA, achieving a weighted validation error of 0.2388 and an F1 score of 0.7843 on temporal validation splits. This performance significantly surpasses traditional machine learning approaches, with Random Forest and Logistic Classification models achieving validation errors of 0.4578 and 0.4708 respectively.

The RBF Logistic Classification with PCA preprocessing achieves modest improvement over standard logistic classification, reaching a validation error of 0.4657. This improvement, while statistically significant ($p < 0.05$), remains substantially behind the neural network's performance. The reduced gap between training and validation errors in the RBF model suggests better generalization despite similar absolute performance.

LSTM performance falls between traditional methods and the feed-forward neural network, with a validation error of 0.3272. This intermediate performance likely reflects the challenges of applying sequence models to our limited dataset, where the benefits of capturing temporal dependencies compete with the increased model complexity and reduced effective sample size due to sequence requirements.

5.2 Cross-Publisher Performance Analysis

The relative performance patterns observed for SEGA generally hold across publishers, though with varying magnitudes of effect. Table 1 presents the neural network architecture across all publishers, highlighting it's consistently strong performance.

Publisher	Validation Error	F1 Score
SEGA	0.2427	0.7541
CD PROJEKT RED	0.1000	0.9269
Square Enix	0.3200	0.6500
Ubisoft	0.2742	0.7383
EA	0.3120	0.6472
Frontier Developments	0.3400	0.6000

Table 1: Cross-publisher performance metrics for neural network models

CD PROJEKT RED exhibits particularly strong predictability, with neural network validation error reaching 0.1000 when including technical indicators.

5.3 Impact of Technical Indicators

The inclusion of technical indicators produces nuanced effects across our publisher set. For CD PROJEKT RED, technical indicators significantly improve neural network performance, reducing validation error from 0.1253 to 0.1000 and increasing the F1 score from 0.9066 to 0.9269. However, this pattern does not generalize across publishers. For LSTM models, technical indicators often degrade performance, suggesting potential interference with the model's ability to learn temporal patterns from game metrics alone.

6 Discussion

Our experimental results reveal several important insights about the relationship between game performance metrics and publisher stock returns. The consistent superiority of neural networks across multiple publishers suggests the presence of complex, non-linear relationships between game-derived features and stock price movements that simpler models cannot capture. This finding challenges traditional approaches to financial forecasting that rely primarily on linear models or technical indicators.

The effectiveness of our feature engineering pipeline demonstrates the value of incorporating domain-specific knowledge into financial prediction models. By transforming granular game metrics into publisher-level features through carefully designed aggregation strategies, we capture both absolute performance measures and relative competitive dynamics. The strong predictive performance achieved using these features suggests that game performance metrics serve as meaningful leading indicators for publisher financial performance.

The variation in model performance across publishers of varying sizes and focuses warrants careful consideration, suggesting that the effectiveness of game-derived metrics as predictive signals varies across publisher portfolios and market structures.

Our enhancement of the triple-barrier labeling method through weighted labels represents a contribution to financial machine learning. The strong performance across models indicates that incorporating magnitude information in the labeling process provides valuable additional signal for prediction tasks.

The mixed impact of technical indicators across different models and publishers raises important questions about feature interaction effects. While technical indicators improved neural network performance for CD PROJEKT RED, their interference with LSTM performance suggests potential redundancy with temporal patterns already captured by game metrics. This observation highlights the importance of careful feature selection in financial prediction tasks.

6.1 Limitations

Several limitations of our current approach warrant discussion. First, our 88-week study period, while providing sufficient data for model training, may not capture the full range of market conditions affecting publisher stocks. Second, the reliance on daily aggregation of game metrics potentially obscures intraday patterns that could provide additional predictive power. Third, our feature engineering pipeline's computational complexity could present challenges for real-time deployment.

7 Conclusion and Future Work

This research demonstrates the predictive value of game-derived metrics for forecasting publisher stock returns while contributing multiple methodological innovations to financial machine learning. Our hierarchical feature engineering pipeline effectively transforms granular game data into meaningful publisher-level signals. The enhanced triple-barrier labeling method improves the capture of price movement magnitude, while our implementation of purged cross-validation ensures robust model evaluation.

Neural networks consistently outperform traditional approaches, achieving validation errors as low as 0.1000 for certain publishers. This performance suggests that sophisticated non-linear models can effectively leverage alternative data sources for financial prediction. The varying effectiveness of technical indicators across publishers indicates potential for further optimization of feature combinations under specific market conditions.

Several promising directions emerge for future research:

- First, extending our approach to intraday prediction horizons could capture shorter-term market responses to game performance changes. This would require modifications to both the feature engineering pipeline and labeling mechanism to handle higher-frequency data.
- Second, exploring transformer architectures might better capture long-range dependencies in our feature set compared to current LSTM implementations. These models could potentially identify complex temporal patterns that our current architectures miss.
- Third, investigating the impact of market microstructure and exchange-specific factors could explain the variation in predictability across publishers. Understanding these effects might lead to more targeted prediction strategies for different market contexts.

The public release of our purged cross-validation and triple-barrier labeling implementations provides valuable tools for future research in financial machine learning. These contributions support reproducibility and encourage further methodological innovations in the field.

References

Dessaint, O., Foucault, T., & Fresard, L. (2024). Does alternative data improve financial forecasting? The Horizon effect. *The Journal of Finance*, *79*(3), 2237–2287. `https://doi.org/10.1111/jofi.13323`

Fahd, K., Badr, H., Eddine, L. S., & Mouad, A. (2024). Forecasting stock market returns using Deep Learning and time series techniques: A comparative and empirical study using technical indicators. *Multimedia Tools and Applications*. `https://doi.org/10.1007/s11042-024-20321-9`

Htun, H. H., Biehl, M., & Petkov, N. (2023). Survey of feature selection and extraction techniques for stock market prediction. *Financial Innovation*, *9*(1). `https://doi.org/10.1186/s40854-022-00441-7`

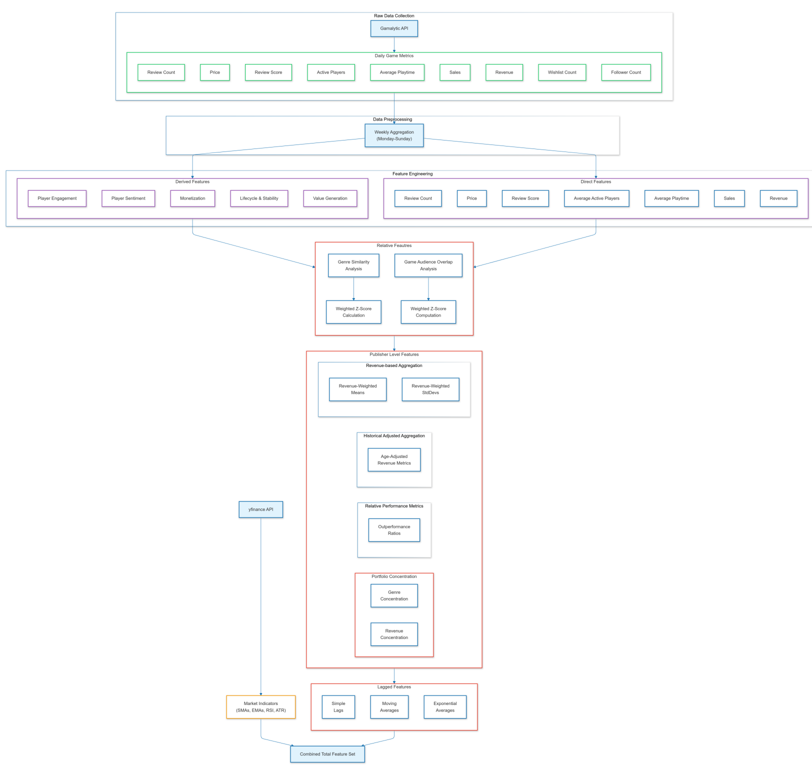
Hu, Z., Ding, Y., Wu, R., Li, L., Zhang, R., Hu, Y., Qiu, F., Zhang, Z., Wang, K., Zhao, S., Zhang, Y., Jiang, J., Xi, Y., Pu, J., Zhang, W., Wang, S., Chen, K., Zhou, T., Chen, J., … Fan, C. (2023). Deep learning applications in games: A survey from a Data Perspective. *Applied Intelligence*, *53*(24), 31129–31164. `https://doi.org/10.1007/s10489-023-05094-2`

Kim, S., Choi, D., Lee, E., & Rhee, W. (2017). Churn prediction of Mobile and online casual games using play Log Data. *PLOS ONE*, *12*(7). `https://doi.org/10.1371/journal.pone.0180735`

López de Prado, M. (2018). *Advances in financial machine learning*. John Wiley & Sons, Inc.

Shen, J., & Shafiq, M. O. (2020). Short-term stock market price trend prediction using a comprehensive deep learning system. *Journal of Big Data*, *7*(1). `https://doi.org/10.1186/s40537-020-00333-6`

A Appendix A: Feature Engineering Pipeline Visualization



B Appendix B: Grid Search Results for Neural Network

Figure 2: Hidden Dimension Size vs Learning Rate

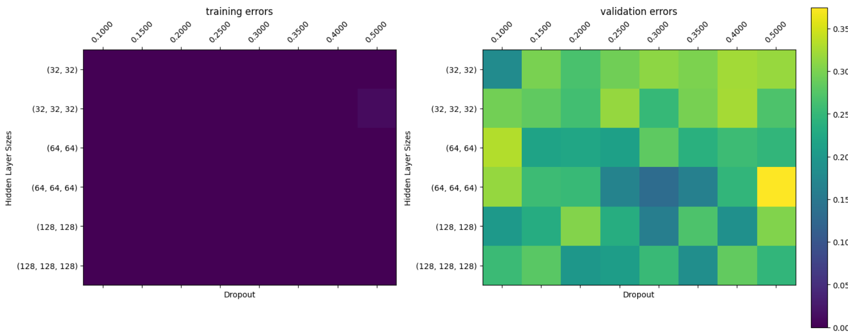


Figure 3: Hidden Dimension Size vs Dropout