Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IMPACT FACTOR: 7.056

IJCSMC, Vol. 15, Issue. 2, February 2026, pg.72 – 79

Match Outcome Prediction in FPS Games Reflecting Sequential Match Flow: Focusing on Valorant and the Seq2Seq LSTM Model

Heesu Park[1]; Gyuseok Kang[2]; Taek Lee[3]

[1,2,3]Division of Computer Science and Engineering, Sun Moon University, Republic of Korea
[1] qkrgmltn0391@sunmoon.ac.kr; [2] qmffntmxms@sunmoon.ac.kr; [3] comtaek76@sunmoon.ac.kr
(Corresponding Author: Taek Lee)

DOI: https://doi.org/10.47760/ijcsmc.2026.v15i02.008

Abstract: With the rapid growth of the esports industry, game data analysis has emerged as a core technology for strategic planning and broadcasting system advancement. In particular, round-based First-Person Shooter (FPS) games involve not only combat performance but also economy management, inter-round momentum, and side-switching — sequential contextual factors that critically determine match outcomes. Existing studies have largely relied on regression models based on static snapshot data or single-timepoint classification models, failing to adequately capture the dynamic changes and causal relationships within a match.

In this study, we propose a Sequence-to-Sequence (Seq2Seq) LSTM framework — a time-series specialized model — to predict detailed second-half match patterns and final outcomes based on first-half data from FPS games. Experiments were conducted as a case study using actual match logs from Valorant, a representative round-based FPS game. The Seq2Seq model compresses the information from the first 12 rounds into a context vector to predict second-half win/loss outcomes.

Experimental results show that the proposed Seq2Seq LSTM model achieved an F1-Score of 85.56% and ROC-AUC of 92.88%, outperforming the CNN model (84.16%, 91.25%). Notably, the model demonstrated a higher Recall of 89.38%, reflecting stronger capability in capturing actual winning games. This suggests that an encoder-decoder architecture reflecting temporal flow demonstrates consistent performance improvements over fragmented data learning for modeling FPS game win/loss patterns.

Furthermore, this study implements real-time match data collection via web crawling and an automated processing pipeline, demonstrating potential for diverse esports industry applications including coaching system advancement, automated broadcast analysis, and player development tracking.

Keywords: FPS game, time-series analysis, Seq2Seq LSTM, Valorant, esports data analysis, temporal pattern, match outcome prediction

I. INTRODUCTION

The esports industry has been growing rapidly on a global scale, with the global esports market reported to have reached billions of dollars as of 2024 [1]. Esports data analytics has since emerged as a rapidly growing research domain, encompassing player performance evaluation, match outcome prediction, and broadcast enhancement [10]. In particular, with esports being adopted as an official medal event at the 2022 Hangzhou Asian Games, esports have gained industrial and social recognition comparable to traditional sports, transcending its status as mere entertainment [2].

Against this backdrop, VALORANT, a tactical shooter game developed by Riot Games, has shown rapid growth since its launch in 2020. Its official league, the Valorant Champions Tour (VCT), has recorded peak concurrent viewership of millions, establishing itself as a leading global esports title [3].

Round-based FPS games, including VALORANT, possess a unique structure in which side-switching, economy management, and inter-round win/loss momentum interact within a single match to determine the final outcome. This bears similarity to team sports such as basketball and volleyball, where continuous scoring processes directly influence match results. Accordingly, an approach that analyzes match flow as a time series — rather than relying on single-timepoint snapshot data — is essential [4].

However, existing win/loss prediction studies have largely focused on MOBA genres such as League of Legends (LoL) and Dota 2. Even in the case of FPS genre research, there has been a limitation in that studies were confined to static modeling — either classifying win/loss outcomes for a specific round or predicting only the final match result [6][12].

II. RELATED WORK

Prior studies on win/loss prediction in round-based FPS games can be broadly categorized into two methodological approaches: (1) static modeling using traditional statistical and machine learning techniques, and (2) deep learning-based modeling that directly learns time-series structures. This section reviews representative studies of each approach, analyzing their achievements and limitations to clarify the positioning of the present study.

A. *Statistical and Regression-Based Studies*

1) VALORANT Economy-Based Win Rate Prediction: Peng (2024) analyzed the influence of VALORANT's economy system on round win rates, applying a Logistic Regression model with credits, loadout value, and ultimate ability points as key variables. Training on 1,301 rounds yielded approximately 60.61% accuracy, demonstrating that overall team equipment has a greater impact on round win rate than ultimate ability possession, thereby validating the hypothesis that "economic advantage is the key factor determining round outcomes" [8].

2) League of Legends Role-Based Prediction: Bahrololloomi et al. (2023) set role-based performance metrics as core features in League of Legends and compared ten regression-based machine learning models including Random Forest, XGBoost, Gradient Boosting, Ridge Regression, KNN, SVR, CatBoost, MLP, Bagging, and Voting Regressor. Based on 29,010 player records from 2,901 matches, the study reported that feature engineering reflecting role information is a critical factor for improving prediction performance [5].

3) CS:GO Economy and Combat Statistics-Based Prediction: García-Méndez et al. (2025) applied multiple machine learning models including XGBoost, LightGBM, and Logistic Regression using features such as team's current money, loadout value, round K/D ratio, and damage dealt in Counter-Strike: Global Offensive (CS:GO). Results statistically confirmed that economic advantage — particularly equipment value — is the most important variable for predicting round outcomes [7].

B. *Deep Learning-Based Studies*

1) Foundational Studies on LSTM and Seq2Seq Models: For sequential pattern learning in time-series data, Hochreiter & Schmidhuber (1997) proposed the Long Short-Term Memory (LSTM) model, demonstrating outstanding performance across domains including weather forecasting, stock price prediction, and machine translation [13]. Subsequently, Sutskever et al. (2014) proposed the Sequence-to-Sequence (Seq2Seq) architecture with an Encoder-Decoder structure for learning complex mappings between input and output sequences, achieving groundbreaking results in machine translation and sequence generation [14].

2) Deep Learning Sequence Prediction in Sports Domains: Simpson et al. (2022) encoded event sequences in soccer matches as temporal information and applied a Transformer-based model (Seq2Event) to predict match progressions [15]. Miller & Johnson (2020) extracted player movement trajectories and scoring flows in basketball as time-series features and fed them into an LSTM model to predict temporal patterns of matches [16]. These studies demonstrated that time-series deep learning models effectively learn temporal dependencies in match flow, achieving higher performance compared to static models.

3) VALORANT Video-Based Multimodal Prediction: Hayakawa et al. (2025) extracted visual sequence information — including agent positions on the minimap, movement patterns, and tactical events — from VALORANT match footage and fed them into TimeSformer, a Transformer-based video understanding model. Experiments on professional match footage showed that the deep learning model exhibited far greater representational capacity than simple log-based models, recording approximately 81% accuracy after the mid-round phase [9].

C. *Summary and Differentiation of the Present Study*

Statistical and regression-based studies (Peng 2024, Bahrololloomi et al. 2023, García-Méndez et al. 2025) successfully identified the influence of key game features — such as economy indicators, equipment value, and ultimate ability status — on round outcomes, providing important theoretical foundations for the feature engineering in this study. However, since these studies rely on static modeling that treats each round as an independent event, they fail to capture the fact that a Full-Buy round after three consecutive losses and a Full-Buy round after three consecutive wins carry entirely different game contexts despite identical economic states.

Deep learning-based studies (Hochreiter & Schmidhuber 1997, Sutskever et al. 2014, Simpson et al. 2022, Miller & Johnson 2020, Hayakawa et al. 2025) demonstrate the ability to directly model time-series structures and learn sequential dependencies. Real-time prediction approaches such as Yang & Roberts (2016) have also shown the feasibility of in-game win probability estimation in MOBA genres [6]. In particular, the Seq2Seq model excels at learning complex mapping relationships between input and output sequences and has been successfully applied to match flow prediction in sports domains. However, sequence prediction based on round-level numerical log data remains relatively less explored from a many-to-many temporal modeling perspective in the esports FPS domain.

The present study utilizes the key features proven by statistical and regression-based research (economy state, buy type, K/D, ultimates, loadout value), while applying the methodology of deep learning-based research. Through a Sequence-to-Sequence LSTM model, it serially predicts round-by-round outcomes in the second half of VALORANT matches from a sequence of the first 12 rounds. This represents a fundamental differentiation in that it directly learns sequential inter-round dependencies — such as credit accumulation and momentum shifts — to model the dynamic structure of game flow that static models fail to capture.

III. Case Study on Valorant Data

A. *Research Overview*

Using the VCT (Valorant Championship Tour) dataset from Kaggle, this study conducts a case study to examine how much the continuity between rounds in a round-based FPS video game contributes to match outcome prediction through comparisons across multiple models. The study encompasses win/loss prediction research utilizing regression classification models and deep learning models, as well as additional research predicting round-by-round second-half outcomes using deep learning models based on first-half round data.

B. *Data Preprocessing and Feature Engineering*

1) *Dataset Structure*: The features provided by the dataset used in this study are summarized in Table 1. The VALORANT dataset consists of time-series data recording the game state at the round level, extracted from VCT official match records. Each game comprises a maximum of 24 rounds, with each round containing various features including economic conditions, kill/death counts, and round type. The original dataset contains a total of 24,965 game records, each consisting of approximately 30–50 original features.

TABLE I
DATASET FEATURE DESCRIPTION

Feature Name	Type	Description	Range
My_Kills_R	Integer	Team kills in the round	0–5
Opp_Kills_R	Integer	Opponent team kills in the round	0–5
Remaining Credits_R	Integer	Remaining team economy (credits)	0–9000
Win Count_R	Integer	Cumulative win rounds	0–24
Loss Count_R	Integer	Cumulative loss rounds	0–24
Outcome_R	Categorical	Round result (Win/Loss)	{Win, Loss}
Type_R	Categorical	Round type	{Pistol, Half-buy, Full-buy, Eco}
Method_R	Categorical	Round method	{Attack, Defense, ...}
Loadout Value_R	Integer	Total equipment value used	0–10000
Cumulative Score_R	Integer	Cumulative score (Win–Loss)	−24–24

The dataset structure reflects time-series characteristics, taking a two-dimensional form of (round, feature). Each round is treated as an independent observation while simultaneously possessing dependency in which information from the previous round accumulates into the next round. These time-series characteristics became an important consideration in the LSTM-based model design.

2) Missing Value and Outlier Handling: Missing values in the dataset arose due to missing initial round data and abnormal game terminations. At the game level, only games with a minimum of 13 recorded rounds were selected; of the total 24,965 games, 24,956 games (99.96%) were confirmed. For categorical features, missing values in Type and Method were replaced with the mode, while missing Outcome values were handled by record deletion. For numerical features, My_Kills and Opp_Kills were replaced with 0, and Remaining Credits along with cumulative values (Win Count, Loss Count, Cumulative Score) were processed using Forward Fill to maintain the previous round's value, thereby ensuring logical consistency of cumulative values. Since the maximum kill count per game rule is 5 and the maximum credits is 9,000, outlier clipping was applied — restricting My_Kills and Opp_Kills to a maximum of 5 and Remaining Credits to a maximum of 9,000 — to ensure the physical validity of the data. Clipped records accounted for less than 0.3% of the total. The handling methods applied are summarized in Table 2.

TABLE II
MISSING VALUE AND OUTLIER HANDLING METHODS

Target	Handling Method	Applied Column
Incomplete games	Filtering	All
Categorical missing	Mode substitution	Type, Method
Outcome missing	Record deletion	Outcome
Kill count missing	Zero substitution	My_Kills, Opp_Kills
Credits missing	Forward Fill	Remaining Credits
Cumulative missing	Forward Fill	Win/Loss Count, Score
Kill count outliers	Clipping	My_Kills, Opp_Kills
Credits outliers	Clipping	Remaining Credits

3) Categorical Feature Encoding and Numerical Normalization: Type (round type) consists of six categories — Pistol, Half-Buy, Full-Buy, Eco, Save, and Light-Buy — and was represented as a 6-dimensional binary vector using One-Hot Encoding (e.g., Pistol = [1, 0, 0, 0, 0, 0]). Method (round method) consists of five categories — Attack, Defense, Retake, Default, and Execute — and was encoded as a 5-dimensional binary vector. Outcome (round result) was binary-encoded as Win = 1 and Loss = 0 to ensure compatibility with the Sigmoid activation function and Binary Crossentropy loss function. Kill counts and credits were normalized to the [0, 1] range using Min-Max Scaling for neural network training stability — My_Kills and Opp_Kills with a maximum value of 5, and Remaining Credits with a maximum value of 9,000 (e.g., 3 kills → 0.6, 4,500 credits → 0.5). Win Count, Loss Count, and Cumulative Score were transformed to mean 0 and standard deviation 1 using Z-score normalization $\mathbf{X}_{\text{standardized}} = \frac{\mathbf{X}-\mu}{\sigma}$, where Win Count has statistics of $\mu \approx 8.2$ and $\sigma \approx 3.1$.

As a result, One-Hot Type (6) + One-Hot Method (5) + Binary Outcome (1) + Min-Max Kills (2) + Min-Max Credits (1) + Z-score cumulative values (3) yields a total of 22 features per round, and the input sequence for one game takes the form of (T, 22), as summarized in Table 3.

Regarding the target variable, a match outcome of Win was assigned when a team first reached 13 round wins, and Loss otherwise, resulting in a binary label used as the prediction target for model training. Rows with non-consecutive round records below 13 rounds were treated as missing and excluded. For rounds that were not played, a placeholder value of −1 was used to distinguish them from Win (1) and Loss (0) labels.

TABLE III
PREPROCESSING METHODS BY FEATURE

Preprocessing Method	Applied Feature	Dimensions	Total Features
One-Hot	Type	6	144
One-Hot	Method	5	120
Binary Encoding	Outcome	1	24
Min-Max	Kills (×2)	2	48
Min-Max	Credits	1	24
Z-score	Cumulative values (×3)	3	72

C. *Classification-Based Win/Loss Prediction Model Study*

1) *Classification Model Research Plan and Feature Summary*: The prediction models used in this study include Logistic Regression, Random Forest, Gradient Boosting, XGBoost, and LightGBM for regression-based classification model experiments. Based on the features from Table 1, features utilized in prior studies on FPS and round-based team game research were selected for use in both classification and deep learning model experiments. Through arithmetic operations on these features, the study aimed to identify which feature combination yields the optimal prediction performance, using the XGBoost model as the baseline for comparison against other models. The full list of derived features is presented in Table 4.

TABLE IV
FEATURES DERIVED FROM PRIOR RESEARCH

Feature Name	Data Source	Calculation Method
AvgCredits12R	R1–R12 Remaining Credits	Mean value
TotalCredits12R	R1–R12 Remaining Credits	Cumulative sum
R12Loadout	R12 start point	Direct extraction
R12Credits	R12 start point	Direct extraction
AvgKD12R	R1–R12 (My Kills − Opp Kills)	Mean value
TotalKillDiff12R	R1–R12 (My Kills − Opp Kills)	Cumulative sum
R12KD	R12 point	Direct extraction
R12KillDiff	R12 point	Direct extraction
EcoCount12R	R1–R12 purchase type	Number of Eco rounds
SemiEcoCount12R	R1–R12 purchase type	Number of Semi-Eco rounds
SemiBuyCount12R	R1–R12 purchase type	Number of Semi-Buy rounds
FullBuyCount12R	R1–R12 purchase type	Number of Full-Buy rounds
R12BuyType	R12 purchase type	Ordinal encoding
FinalScore	R12 end point	Direct extraction
R6Score	R6 end point	Direct extraction
R9Score	R9 end point	Direct extraction

TABLE V
FEATURE COMBINATION PERFORMANCE RESULTS

Comb_ID	Feature Configuration	No. of Features	F1-Score
C1	AvgCredits12R, TotalCredits12R, R12Loadout, R12Credits, R12KD, R12KillDiff, FinalScore	7	0.8360
C2	TotalLoadout12R, AvgCredits12R, R12Credits, TotalKillDiff12R, R12KillDiff, FullBuyCount12R, FinalScore, R9Score	8	0.8358
C3	TotalLoadout12R, R12Loadout, TotalKillDiff12R, R12KD, R12KillDiff, FullBuyCount12R, FinalScore	7	0.8349
C4	AvgCredits12R, R12Loadout, R12Credits, R12KD, R12KillDiff, FullBuyCount12R, FinalScore	7	0.8350
C5	AvgKD12R, R12KD, R12KillDiff, FullBuyCount12R, R12BuyType, FinalScore, R6Score	7	0.8359
C6	TotalKillDiff12R, R12KD, R12KillDiff, FullBuyCount12R, R12BuyType, R6Score	6	0.8357
C7	R12Loadout, TotalKillDiff12R, FullBuyCount12R, SemiEcoCount12R, FinalScore, R6Score	6	0.8341
C8	AvgCredits12R, R12Loadout, R12Credits, R12KD, R12KillDiff, FullBuyCount12R, FinalScore	7	0.8337
C9	AvgKD12R, R12KD, R12KillDiff, FullBuyCount12R, SemiBuyCount12R, FinalScore, R6Score	7	0.8339
C10	TotalCredits12R, R12Loadout, TotalKillDiff12R, SemiBuyCount12R, R12BuyType, FinalScore, R6Score, R9Score	8	0.8330

Based on the results of Table 5, the combination with the highest F1-Score was selected for subsequent model experiments.

2) Classification Model Experiment Results: Model training was conducted using GridSearchCV for hyperparameter optimization with the features identified through the above process. 80% of the preprocessed dataset was used for training and 20% for evaluation.

TABLE VI
CLASSIFICATION MODEL EVALUATION RESULTS

Model	Key Hyperparameters	F1-Score	ROC-AUC
Logistic Regression	C=0.01, penalty=L2, solver=lbfgs	0.8279	0.9098
Random Forest	n_estimators=100, max_depth=5	0.8324	0.9048
Gradient Boosting	n_estimators=100, lr=0.05, max_depth=2, subsample=0.8	0.8327	0.9102
XGBoost	n_estimators=100, lr=0.05, max_depth=3, subsample=1.0, colsample_bytree=1.0	0.8334	0.9100
LightGBM	n_estimators=100, lr=0.1, max_depth=10, num_leaves=15, subsample=0.8	0.8310	0.9093
MLP (Keras)	6-layer (128-128-64-64-32-32-16-16), Adam(lr=1e-3)	0.8356	0.9094

As shown in Table 6, the MLP and XGBoost models recorded the highest performance in terms of F1-Score, while Gradient Boosting and XGBoost demonstrated superior performance in terms of ROC-AUC. Overall, the

XGBoost model showed excellence in both stability and accuracy, achieving high performance across both metrics among all classification models.

D. *Deep Learning-Based Win/Loss Prediction Model Study*

1) Deep Learning Model Research Plan and Feature Summary: The prediction models used in this study include CNN and LSTM-based models for second-half sequence outcome prediction and win/loss prediction experiments. To enable performance comparison with classification-based models, an additional evaluation was conducted based on how accurately the overall match outcome — including both the first-half sequence and the predicted second-half sequence — was predicted. Referencing features used in prior deep learning-based studies on CNN and LSTM, the study aimed to identify the optimal feature combination using features derivable from or directly applicable to the currently secured dataset, as listed in Table 7.

TABLE VII
DEEP LEARNING-BASED DERIVED FEATURES FROM PRIOR RESEARCH

Feature Name	Calculation Method	Meaning
Cumulative_Kills_N	Sum of R1–RN My_Kills	Cumulative kill count (performance indicator)
Cumulative_Deaths_N	Sum of R1–RN Opp_Kills	Cumulative death count (defensive indicator)
KD_Ratio_N	(Cumulative Kills + 1) / (Cumulative Deaths + 1)	K/D ratio (offensive capability)
Economy_Efficiency_N	Cumulative Kills / (Total Economy / 1000 + 1)	Economic efficiency (resource utilization)
Economic_Trend_N	RN Economy − R(N−1) Economy	Economic trend (round N economy change)
Win_Rate_N	Win_Count_N / (Win + Loss_N + ε)	Cumulative win rate (match progress indicator)

A total of 64 feature combinations were evaluated, and the combination #29 — consisting of Cumulative_Kills_N, KD_Ratio_N, and Win_Rate_N — combined with the original dataset features recorded the highest accuracy. The top-performing combinations are presented in Table 8, and model training was conducted using the best-performing feature combination.

TABLE VIII
FEATURE COMBINATION PERFORMANCE RESULTS (DEEP LEARNING)

Comb_ ID	Derived Features	No. of Features	Accuracy
#29	Cumulative_Kills_N, KD_Ratio_N, Win_Rate_N	11	0.8365
#19	KD_Ratio_N, Win_Rate_N	10	0.8363
#16	Cumulative_Deaths_N, Win_Rate_N	10	0.8361
#40	KD_Ratio_N, Economy_Efficiency_N, Win_Rate_N	11	0.8359
#10	Cumulative_Kills_N, Economy_Efficiency_N	10	0.8357
#23	Cumulative_Kills_N, Cumulative_Deaths_N, KD_Ratio_N	11	0.8353
#31	Cumulative_Kills_N, Economy_Efficiency_N, Win_Rate_N	11	0.8342
#8	Cumulative_Kills_N, Cumulative_Deaths_N	10	0.8340
#17	KD_Ratio_12, Economy_Efficiency_12	10	0.8340
#33	Cumulative_Deaths_12, KD_Ratio_12, Economy_Efficiency_12	11	0.8340

2) Deep Learning Model Architecture: The rationale for adopting a Seq2Seq architecture, rather than a simple LSTM classifier, lies in the nature of the prediction target. Since the second half of a VALORANT match constitutes a 12-step outcome sequence rather than a single binary label, a many-to-many mapping is required. The Seq2Seq framework — comprising an encoder that compresses first-half round sequences into a context vector, and a decoder that generates second-half outcome sequences from that representation — is structurally suited to model such temporal evolution, enabling the capture of not only the final match outcome but also the round-by-round progression of the second half.

For the Seq2Seq-based LSTM model, performance evaluation differs from other models in that it is assessed based on the similarity of the predicted second-half sequence to the actual sequence. Accordingly, an additional evaluation process was incorporated — comparing the overall match outcome generated by combining the first-half sequence with the predicted second-half sequence against the actual match outcome — to ensure a fair performance comparison across all models on the same prediction target.

For the CNN model, a 2D-format dataset was constructed by arranging the 12 time-step first-half rounds and their corresponding features per round (11 derived features as selected in combination #29) into a matrix format, with the target being the binary classification of match win/loss. Both models were subsequently subjected to hyperparameter optimization via GridSearch to produce the final model evaluation results.

3) Deep Learning Model Evaluation Results:

TABLE IX
DEEP LEARNING MODEL EVALUATION SUMMARY

Metric	CNN	Seq2Seq LSTM
Accuracy	83.60%	84.45%
Precision	83.15%	81.92%
Recall	85.22%	89.38%
F1-Score	84.16%	85.56%
ROC-AUC	91.25%	92.88%

As presented in Table 9, the Seq2Seq LSTM model achieved an F1-Score of 85.56% and ROC-AUC of 92.88%, outperforming the CNN model (84.16%, 91.25%). In particular, with a Recall of 89.38%, the LSTM demonstrated stronger capability in accurately capturing actual winning games, which is attributed to LSTM's effective modeling of temporal dependencies between rounds. While CNN achieved higher Precision (83.15%) than LSTM (81.92%), LSTM demonstrated more consistent performance improvements in the overall balanced metrics of F1-Score and ROC-AUC. Accordingly, the Seq2Seq LSTM was selected as the primary model for subsequent analysis.

E. *Feature Importance Analysis of the Final Model*

SHAP (Shapley Additive Explanations) analysis was conducted on the best-performing Seq2Seq LSTM model to extract and analyze feature importance, and the results are presented in Table 10.

TABLE X
FEATURE IMPORTANCE VIA SHAP ANALYSIS

Rank	Feature Name	SHAP Value	Impact Type
1	Cumulative_Kills_N	0.184	Momentum
2	KD_Ratio_N	0.162	Momentum
3	Win_Rate_N	0.139	Momentum
4	Loadout Value	0.127	Economy
5	Remaining Credits	0.108	Economy
6	My_Kills	0.095	Performance

All top three features were momentum-related indicators, with Cumulative_Kills_N recording the highest importance. This indicates that continuous kill acquisition during rounds R1–R12 shows relatively higher association with second-half win/loss prediction in this dataset. Economy-related features, Loadout Value and Remaining Credits, ranked 4th and 5th respectively, showing comparatively lower SHAP values than momentum features. These results suggest that momentum-related indicators may play a more influential role than economy-related features within this experimental setting, though the differences in SHAP values are modest and should be interpreted with caution.

IV. CONCLUSIONS

A. *Key Research Findings:*

This study conducted win/loss prediction research based on the first-half 12-round data of VALORANT matches. Comparative experiments were performed across six regression classification models (Logistic Regression, Random Forest, Gradient Boosting, XGBoost, LightGBM, MLP) and two deep learning models (CNN, Seq2Seq LSTM). Results indicated that deep learning-based models demonstrated consistent performance improvements over regression classification models, with the Seq2Seq LSTM model achieving the best performance at F1-Score 85.56% and ROC-AUC 92.88%.

SHAP analysis revealed that all top three features were momentum-related indicators (Cumulative_Kills_N, KD_Ratio_N, Win_Rate_N), which showed comparatively higher SHAP values than economy-related features. This suggests that momentum-related indicators may play a more influential role than economy-related features in predicting match outcomes within this dataset, though the magnitude of difference warrants cautious interpretation.

While regression classification models and CNN process each round independently and fail to capture temporal context, the Seq2Seq LSTM learns the win/loss sequence of the first 12 rounds through an Encoder-Decoder structure and generates second-half patterns based on this learning. A Full-Buy round after consecutive wins and a Full-Buy round after consecutive losses carry entirely different game contexts despite identical economic states, and LSTM effectively learns such temporal dependencies, demonstrating stable and consistent performance improvements compared to other models.

In conclusion, this study empirically demonstrates that time-series deep learning models reflecting temporal dependencies show consistent performance improvements over static models for win/loss prediction in round-based FPS games, and suggests that momentum-related features may serve as more influential predictors than economy-related features within the scope of this study.

B. *Expected Effects and Applications:*

The Seq2Seq LSTM-based win/loss prediction model proposed in this study can be applied across various fields in the esports industry. In professional team coaching systems, it can provide second-half win rate prediction and momentum analysis at the mid-match point (end of R12), establishing a quantitative basis for tactical adjustments. In broadcasting and commentary systems, it can analyze round flow in real time and visualize win/loss probabilities to enhance viewer immersion, consistent with prior work demonstrating the value of predictive analytics in esports broadcasting [11]. In player development and growth tracking, it can quantify individual player momentum contributions to inform training directions.

The data used in this study can be crawled and extracted from VLR.gg, which can serve as the foundation for building a real-time prediction system that adapts to meta changes through continuous data collection and model updates [17]. This demonstrates the potential for esports data science to evolve beyond simple prediction into a core tool for strategic planning and player development.

References

[1]. Newzoo, Global Esports Market Report 2024, Newzoo, 2024.

[2]. International Olympic Committee, Esports at the 2022 Asian Games: Medal Events Announced, IOC Official Statement, 2022.

[3]. Riot Games, Valorant Champions Tour 2024 Statistics and Viewership Data, Riot Games Official Report, 2024.

[4]. A. C. Constantinou, N. E. Fenton, "Solving the problem of inadequate scoring rules for assessing probabilistic football forecast models," Journal of the Operational Research Society, Vol. 63, No. 5, pp. 589–595, 2012.

[5]. N. Bahrololloomi, M. Gabel, Y. Liu, "Role-based Performance Analysis and Match Outcome Prediction in League of Legends," SN Computer Science, Vol. 4, article 238, 2023.

[6]. M. Yang, S. Roberts, Real-time win prediction in Dota 2, Technical Report, Department of Engineering Science, University of Oxford, 2016.

[7]. A. García-Méndez, J. C. Burguillo, F. A. Mikic-Fonte, "Explainable e-sports win prediction through Machine Learning and feature engineering: A case study on CS:GO," Computer Communications, Vol. 199, pp. 32–47, 2025.

[8]. Y. Peng, "A Predictive Analysis of Valorant Esports: Win Probability Through Economy and Ultimate Ability," IEEE TechRxiv Preprint, 2024.

[9]. T. Hayakawa, S. Nakamura, K. Yamamoto, "Round Outcome Prediction in VALORANT Using Tactical Features from Match Footage," arXiv preprint arXiv:2510.17199, 2025.

[10]. P. P. S. Ayres, R. B. Mariño, A. C. L. Lopes, "Esports Analytics: A Systematic Review of Tools, Techniques, and Research Trends," arXiv preprint arXiv:2304.08765, 2023.

[11]. J. Smith, A. Johnson, "Applications of Predictive Analytics in Esports Broadcasting and Fan Engagement," IEEE Transactions on Games, Vol. 13, No. 2, pp. 145–159, 2021.

[12]. K. H. Kim, J. Park, S. Lee, "Machine Learning Approaches for Match Outcome Prediction in Multiplayer Online Battle Arena Games: A Survey," Pertanika Journal of Science & Technology, Vol. 33, No. 2, pp. 563–581, 2025.

[13]. S. Hochreiter, J. Schmidhuber, "Long Short-Term Memory," Neural Computation, Vol. 9, No. 8, pp. 1735–1780, 1997.

[14]. I. Sutskever, O. Vinyals, Q. V. Le, "Sequence to Sequence Learning with Neural Networks," Advances in Neural Information Processing Systems (NIPS), pp. 3104–3112, 2014.

[15]. I. Simpson, S. Magill, J. Davis, "Seq2Event: Learning the Language of Soccer using Transformer-based Models," Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), pp. 1715–1725, 2022.

[16]. A. R. Miller, P. K. Johnson, "Predicting Basketball Score Trajectories Using Recurrent Neural Networks," Journal of Sports Analytics, Vol. 6, No. 3, pp. 201–215, 2020.

[17]. VLR.gg, Valorant Competitive Match History and Statistics Database, https://www.vlr.gg, accessed 2024.

