Anomaly Detection in eSport Games Through Periodical In-Game Movement Analysis with Deep Recurrent Neural Network

Mhd Irvan, Franziska Zimmer, Ryosuke Kobayashi, Maharage Nisansala Sevwandi Perera, Roberta Tamponi and Rie Shigetomi Yamaguchi

Graduate School of Information Science and Technology, The University of Tokyo, Japan

{*irvan, zimmer, kobayashi, perera.nisansala, roberta.tamponi*}*@yamagula.ic.i.u-tokyo.ac.jp, yamaguchi.rie@i.u-tokyo.ac.jp*

Keywords: Machine Learning, Deep Neural Network, Behavioral Analysis, Game AI.

Abstract: Detecting anomaly in online video games is important to ensure a fair and secure gaming session. This is particularly crucial in eSport games, where competitive fairness is crucial. In this paper, we present an approach to anomaly detection in gaming sessions, using a variant of Deep Recurrent Neural Network, called Long Short-Term Memory (LSTM) network. Recurrent Neural Networks (RNNs) and their variant, LSTMs, are well-suited for this kind of task due to their ability to capture sequential patterns in gameplay data. The proposed system learns from normal gameplay patterns to identify anomalous behaviors such as impersonation. To confirm the feasibility of our approach, we use a game called Counter-Strike: Global Offensive (CSGO) serving as a case study. We utilize a public CSGO dataset containing in-game movement data, including coordinates, timestamps, and other contextual information. To test the model's detection capabilities, synthetic data representing anomalous behaviors was injected into the dataset. The data was preprocessed and segmented into sequences, simulating the dynamics of player movements. Our LSTM model was trained to learn temporal dependencies within these sequences, enabling it to distinguish between normal and anomalous behaviors. Performance evaluation demonstrated the model's robustness and effectiveness in detecting anomalies. The results indicate that our approach is able to detect anomalous activities, highlighting its potential for application in online gaming platforms to foster a more enjoyable gaming experience for all participants.

1 INTRODUCTION

The gaming industry has evolved rapidly over the past decades, with online multiplayer experiences becoming more prevalent than ever. However, alongside the growth in popularity, there has been a corresponding rise in disruptive behaviors such as cheating, impersonation, and exploitation of game mechanics (Rosell, 2017).

Anomaly detection in online video games is essential due to the impact it has on the gaming experience, competitive integrity, and overall security of gaming environments. Online video games are rising in popularity. The number of players engaging in competitive and cooperative gameplay across various platforms is increasing (Funk, 2018). Ensuring a fair and secure environment has become a critical concern for game developers. Anomalies in online gaming can manifest in various forms, including cheating, hacking, bot-assisted playing, and other unauthorized behaviors that disrupt the fairness of the game (Chen, 2018). Such activities not only destroy the experience for legitimate players but also undermine the competitive nature of esports, where fairness and skill are paramount. Cheating in online games can lead to significant losses for game developers and publishers, as it could deter new players from joining, drive away existing ones from staying, and tarnish the reputation of the game itself (Ghoshal, 2019).

The popularity of esports has further demand the need for effective anomaly detection. In professional gaming tournaments, where large sums of prize, sponsorships, and reputations are at stake, keeping the integrity of the overall competition is crucial (Conroy, 2021). Any form of cheating or unfair advantage can have damaging consequences, potentially leading to disqualification and a loss of trust in the competitive scene. Therefore, robust mechanisms to detect and mitigate anomalies are essential to maintain the integrity and credibility of the competitions.

Traditional methods of anomaly detection (Dinh, 2016), often relying on rule-based systems and manual monitoring, have proven to be insufficient against sophisticated cheating techniques. These methods

Irvan, M., Zimmer, F., Kobayashi, R., Perera, M., Tamponi, R. and Yamaguchi, R.
Anomaly Detection in eSport Games Through Periodical In-Game Movement Analysis with Deep Recurrent Neural Network.
DOI: 10.5220/0012978100003837

In *Proceedings of the 16th International Joint Conference on Computational Intelligence (IJCCI 2024)*, pages 430-437
ISBN: 978-989-758-721-4; ISSN: 2184-3236

can be circumvented and are not scalable to the large volumes of data generated in modern online games. Consequently, there is a pressing need for advanced, data-driven approaches that can automatically and accurately identify anomalous behaviors.

In recent years, the advent of deep learning techniques, particularly deep neural networks (DNNs), has opened up new avenues for addressing complex problems in various domains. One such area is anomaly detection, where DNNs have shown promise in identifying deviations from normal patterns within large and diverse datasets (Irvan, 2021). By analyzing vast amounts of gameplay data, these techniques can learn complex patterns and detect deviations that hint anomalous behavior. By leveraging the rich and dynamic data generated during gameplay, including player movements, interactions, and decision-making processes, DNNs offer a powerful tool for detecting anomalous behaviors in video game environments.

A specialized type of DNNs, called Recurrent Neural Networks (RNNs) are well-suited for this kind of task due to their ability to capture temporal dependencies in sequential data (Goh, 2017). In particular, variations of RNNs, called Long Short-Term Memory (LSTM) networks, have been found success in capturing patterns in very long sequential data (Lindemann, 2021). This research explores the application of LSTM networks for anomaly detection in esports, using "Counter-Strike: Global Offensive" (CSGO) (Rizani, 2018) as a case study. We use player movements information throughout the game as the input data for our model. To simulate anomaly, we inject synthetic data representing various types of anomalies into the data. By implemeting this, we aim to enhance the model's ability to detect anomalous behaviors. To validate the efficacy of our approach, we conduct extensive experiments using a dataset generated from CSGO gameplays. We then evaluate the performance of anomaly detection system in accurately identifying abnormal behaviors.

Our approach not only addresses the limitations of traditional methods but also provides a scalable and effective solution for maintaining fairness of online gaming environments. Through this study, we aim to contribute to the development of advanced anomaly detection systems that can be applied across different online gaming platforms and genres, ultimately fostering a more secure and enjoyable gaming experience for all players.

The rest of this paper is structured as follows. In section 2, we review existing approaches to anomaly detection, highlighting the limitations. In section 3, we describe the use of Long Short-Term Memory (LSTM) networks, offering a detailed explanation of the data preprocessing and model architecture. In section 4, we present the experimental setup and evaluates the model's performance. In section 5, we discuss our findings from interpreting the results. In section 6, we suggest potential improvements for future work. Finally, in section 7, we provide concluding remarks, emphasizing the potential of deep learning for improving fairness in online gaming environments.

2 RELATED WORKS

The domain of security in online video games has garnered significant attention in recent years. This section reviews the relevant literature and methodologies that have been applied to anomaly detection, with a focus on machine learning and deep learning approaches.

2.1 Traditional Anomaly Detection Techniques

Early efforts in anomaly detection for primarily relied on rule-based systems or heuristic approaches (Fernandes, 2019), which may struggle to capture the complexity and variability of player behaviors. Rule-based systems use predefined rules and thresholds to identify suspicious activities. For instance, an abnormal increase in a player's score within a short time frame could trigger an alert. While straightforward, these methods suffer from high false positive rates and can be easily circumvented by sophisticated cheats.

Statistical methods have also been used to detect outliers (Akoglu, 2015) in data. These methods model the normal behavior of players and identify deviations as potential anomalies. However, they often fail to capture the complex and dynamic nature of player behaviors in modern online games.

2.2 Machine Learning Approaches

The advent of machine learning introduced more advanced techniques for anomaly detection in gaming. Supervised learning methods, including decision trees, support vector machines (SVMs), and ensemble methods like random forests, have been applied to classify normal and anomalous behaviors (Hosseinzadeh, 2021). These approaches require labeled datasets for training, which can be a limitation due to the scarcity of labeled anomalous data.

Unsupervised learning methods have been explored to detect anomalies without labeled data (Nguyen, 2015). These algorithms learn to catego-

rize data, and anomalies are identified based on categorical densities. While effective, these methods often require extensive feature engineering and may not fully capture the temporal dependencies in sequential gameplay data.

2.3 Deep Learning and Recurrent Neural Networks

Deep learning approaches have shown great promise in addressing the limitations of traditional methods. Convolutional Neural Networks (CNNs) have been used to analyze spatial patterns in data (Alabadi, 2020). By training the model on a large dataset, the system was able to identify deviations indicative of anomalous actions However, CNNs are not well-suited for capturing temporal dependencies, which are crucial for understanding sequential player actions.

In addition to CNNs, recurrent neural networks (RNNs) have also been explored for anomaly detection (Ackerson, 2021). RNNs have been demonstrated as powerful tools for sequence modeling. They are capable of learning long-term dependencies and capturing the dynamics of player behaviors. Previous studies have applied RNNs for various anomaly detection tasks, such as identifying fraudulent activities in financial transactions and detecting network intrusions.

While these studies demonstrate the potential of deep learning techniques for anomaly detection, there remain several challenges and opportunities for further research (Pang, 2021). The interpretability of deep learning models in the context of anomaly detection remains an ongoing area of investigation, as understanding the underlying reasons for model predictions is crucial for effective decision-making and intervention strategies.

3 PROPOSED APPROACH

Our proposed approach to anomaly detection in video games through in-game behavioral analysis of players revolves around the utilization of Long Short-Term Memory (LSTM) network (Sherstinsky, 2020) to capture intricate patterns of normal gameplay behavior and identify deviations indicative of anomalous actions. LSTM networks have found success in anomaly detection for other domains due to their ability to capture temporal dependencies in sequential data (Lindemann 2021). Our model detect anomalous player behaviors by capturing sequential dependencies in gameplay data. To enhance the model's detection capabilities, synthetic data representing various types of anomalies is injected into the dataset.

The process begins with data preprocessing, where in-game location data from players is gathered, including coordinates (x, y, z) sampled at regular intervals, along with timestamps and contextual information. This raw data is then subjected to normalization to ensure consistency across different maps and sessions. The normalized data is segmented into fixed time intervals or game rounds, creating manageable sequences that reflect the dynamics of player movements.

These segmented sequences serve as inputs to our model architecture, which is designed to capture and analyze the patterns in the movement data. Our model incorporates LSTM units, which are well-suited for handling long-range dependencies and subtle temporal variations in player behavior. The units are followed by fully connected (dense) layers that map the learned features to detect anomaly.

3.1 Data Preprocessing

The foundation of our approach is a dataset of in-game player behavior data. The data can be collected using a logging program that records player actions during gameplay sessions. Each entry in the dataset should include the following attributes:

- Player ID: A unique identifier for each player.
- Timestamp: The time at which the action was recorded.
- Coordinates (X, Y): The two-dimensional position of the player in the game environment.

This dataset provides a source of general information for analyzing player behavior and detecting anomalies. The coordinate data is scaled into a standardized range, which helps in speeding up the training process and improving model performance. The continuous gameplay data is divided into temporal sequences. Each sequence represents a series of player actions over a specific period, capturing the dynamics necessary for anomaly detection.

3.2 Model Architecture

The core of our proposed approach is an LSTM network, chosen for its ability to model long-term dependencies in sequential data. The components of LSTM models capture the sequences of movements over time. The architecture of the LSTM model (figure 1) is as follows:

1. Input Layer - Receives the preprocessed sequences of player behavior data.

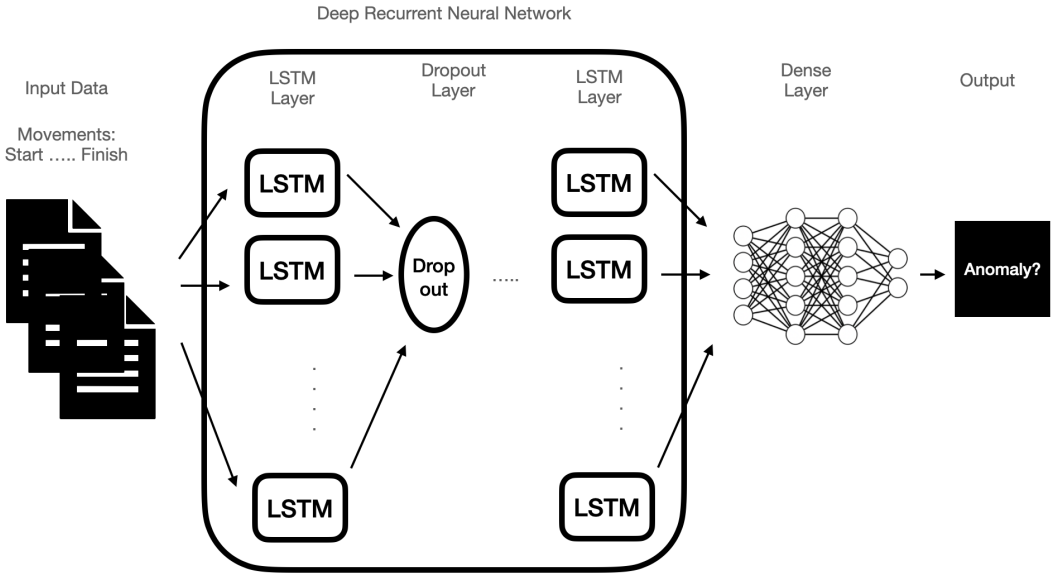

Figure 1: Proposed architecture for anomaly detection.

2. LSTM Layers - Multiple LSTM layers are stacked to capture the complex temporal dependencies in the data. These layers are configured with appropriate hidden units to balance model complexity and computational efficiency.
3. Dropout Layers - Incorporated between LSTM layers to prevent overfitting by randomly setting a fraction of units to zero during training.
4. Fully Connected Layer - A dense layer that maps the LSTM outputs to a lower-dimensional space, facilitating anomaly detection.
5. Output Layer - Produces a probability score indicating the likelihood of the sequence being anomalous.

The model is trained using a supervised learning approach, where normal and synthetic anomalous sequences are labeled appropriately. By integrating these two normal gameplay and synthetic anomaly, our model can effectively capture the dynamic nature of player behaviors.

Training the model involves preparing the data by labeling it with indicators of normal or anomalous behavior, which can be derived from known instances of cheating or synthetic anomalies. The dataset is then split into training, validation, and test sets. The model is trained using the training set, with validation conducted on the validation set. The training process exposes the model to labeled examples of normal and anomalous gameplay behaviors, allowing it to learn the underlying patterns that distinguish between the two classes.

3.3 Anomaly Detection

The final component of our proposed approach is the deployment of the LSTM model for anomaly detection. Once trained, the model is deployed within a simulated environment to perform anomaly detection on incoming gameplay data streams. The data incoming is dynamically segmented into temporal sequences suitable for the LSTM model. As players perform their gameplay activities, their behaviors are continuously analyzed by the model, which flags any deviations from normal patterns as potential anomalies. The data sequences are fed into the trained LSTM model to detect anomaly. For anomaly detection, the trained model is used to predict anomaly scores on unseen player data. A threshold is set for these scores to classify movements as normal or anomalous. The threshold is based on the mean and standard deviation of the anomaly scores from the normal data points.

4 EXPERIMENT

In this section, we present the experimental setup, methodology, and results of our proposed anomaly detection approach. The experiments aim to evaluate the performance and effectiveness of our model in identifying anomalous behaviors. To validate the effectiveness of our proposed approach for anomaly detection in online and esports games, we conducted the experiments using public data from ”Counter-Strike:

Global Offensive" (CSGO) (Xenopoulos, 2022). We selected CSGO as our testbed due to its popularity and the availability of detailed player movement data. The dataset included player positions, timestamps, velocities, and many other contextual information such as player states and in-game events. The data was logged at regular intervals to capture continuous player behaviors. Only data necessary for our model (mentioned in the section of proposed approach) is being used in our experiments. The continuous gameplay data was divided into sequences, each representing a series of player movements over a specified time window. This segmentation is important to capture the dynamics of player behavior.

These sequences then serve as input to the model, training it to learn the sequential dependencies and patterns within the data. Each sequence is labeled as either normal or anomalous. Normal sequences are derived from typical player behavior, while anomalous sequences are identified from instances generated using synthetic methods to simulate abnormal behavior. These sequences are crucial for training a model that can accurately distinguish between normal and anomalous behaviors.

4.1 Synthetic Anomaly Injection

To simulate anomalous behaviors within the gameplay data, we inject synthetic anomalies into the dataset at predefined intervals and durations. These synthetic anomalies mimic common disruptive behaviors observed in online gaming environments, such as impersonation and teleportation. By injecting anomalies of varying severity and complexity, we evaluate the model's ability to detect and classify various anomalous behaviors.

Synthetic anomalies were injected into the dataset using two different methods (figure 2):

1. Sequential Injection (all data after a particular point in gameplay is replaced with synthetic data, simulating a change of player mid-session)
2. Random Injection (data at various points is replaced with synthetic data, simulating multiple players pretending to be one player)

By incorporating these synthetic anomalies, we ensure that the model is exposed to different types of anomalous patterns during training, enhancing its generalization capabilities. By following these preparation steps, we create a diverse dataset suitable for training and evaluating our model. The synthetic anomalies include:

- Erratic Movements - Simulating unnatural or random movements that deviate from typical player behavior.
- Teleportation - Introducing sudden changes in position that mimic teleportation or hacking.
- Bot-like Behavior - Generating movements that resemble automated scripts or bots, which can be detected by their lack of human-like variability.

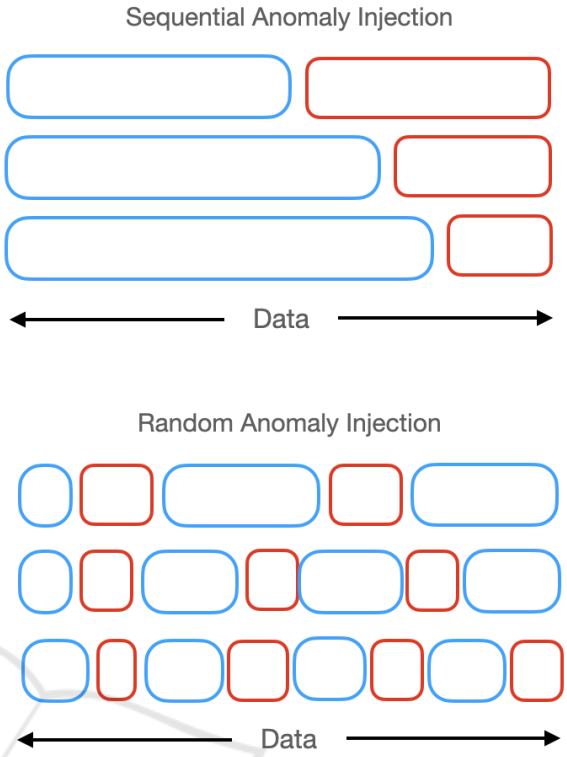

Figure 2: Injecting anomaly into the dataset (blue bars represent real data and red bars represent synthetic data).

For Sequential Injection, five different cases were created where a certain portion of the sequence is real data followed by synthetic data. These cases are : (1) 50% real data followed by 50% synthetic data, (2) 60% real data followed by 40% synthetic data, (3) 70% real data followed by 30% synthetic data, (4) 80% real data followed by 20% synthetic data, and (5) 90% real data followed by 10% synthetic data,

While for Random Injection, another five different cases were created where synthetic data is inserted randomly between real data. These cases are: (1) 10% synthetic data, (2) 20% synthetic data, (3) 30% synthetic data, (4) 40% synthetic data, and finally (5) 50% synthetic data

4.2 Experimental Settings

We partition the dataset into training, validation, and test sets, ensuring that each set contains a balanced

distribution of normal and anomalous gameplay behaviors. The training set is used to train the anomaly detection model, the validation set is used to tune hyperparameters and to prevent overfitting, and the test set is reserved for final evaluation.

The deep learning model is configured with Long Short-Term Memory (LSTM) units to capture the temporal dependencies in the movement data. The architecture consists of four LSTM layers followed by fully connected (dense) layers, which map the learned temporal features to anomaly scores. We limit the number of LSTM layers to four due to the potential of overfitting (Merity 2017). We use binary cross-entropy as the loss function, due to the binary nature of our classification task (normal vs. anomalous behavior), and Adam optimizer for efficient training.

During training, we implement various techniques to enhance the model's performance and generalizability. Early stopping is employed to prevent overfitting by monitoring the validation loss and halting training when performance no longer improves. Model checkpoints are used to save the best performing model based on validation metrics, ensuring that the optimal model configuration is retained.

To further validate our model, cross-validation techniques are employed. The dataset is partitioned into five folds and the model is trained on different combinations of these folds, ensuring that the evaluation results are consistent. Through this model training process, we ensure that our model is well-optimized and capable of detecting anomalies.

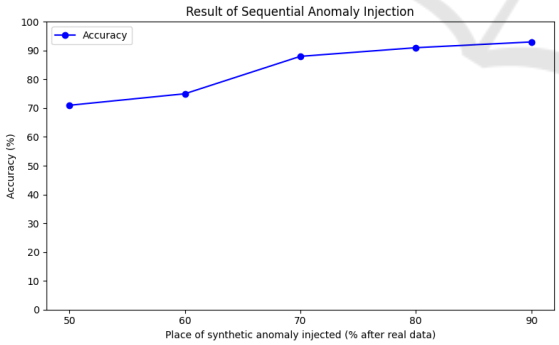

Figure 3: Results showing the accuracy of sequential anomaly detection.

5 ANALYSIS

In this section, we analyze the experimental results obtained from our proposed anomaly detection approach. This includes evaluating the performance of the LSTM model under different synthetic anomaly injection methods, interpreting the findings, and discussing their implications.

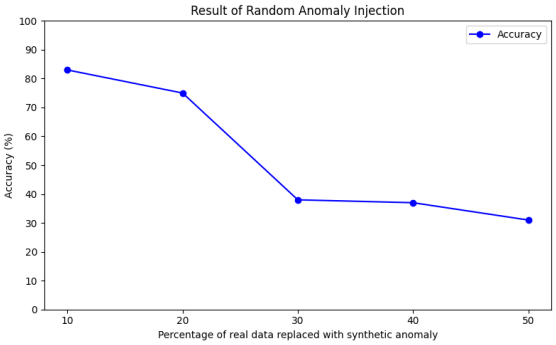

Figure 4: Results showing the accuracy of random anomaly detection.

The trained model was evaluated on the test set with injected synthetic anomalies. Each experiment was repeated 10 times to account for random variations, and the average accuracy was calculated.

The model's performance was primarily evaluated using accuracy metric, which measures the proportion of correctly identified sequences (both normal and anomalous) out of the total sequences. The evaluation was done across two synthetic anomaly injection methods: sequential and random. High accuracy indicates that the model effectively distinguishes between normal and anomalous player behaviors.

5.1 Detection Accuracy of Sequential Anomaly Injection

In this method, synthetic anomalies were injected after a certain proportion of real data. Five cases were tested, with the proportion of real data increasing by 10% increments, starting from 50% real data followed by 10% synthetic data, up to 90% real data followed by 10% synthetic data. Results for the experiment with sequential anomaly injection are summarized in figure 3.

- Case 1 (From 50% Real, 50% Synthetic) - The model achieved an accuracy in the range of 70%.
- Case 2 (From 60% Real, 40% Synthetic) - The accuracy improved slightly.
- Subsequent Cases - A similar trend was observed, with the accuracy gradually increasing as the proportion of real data increased, reaching a peak accuracy above 90% in the 90% real data scenario.

5.2 Detection Accuracy of Random Anomaly Injection

In this method, synthetic anomalies were inserted randomly between segments of real data. Five cases were tested, with the proportion of synthetic data increasing by 10% increments, starting from 10% synthetic data, up to 50% synthetic data injected. Results for the experiment with random anomaly injection are summarized in figure 4.

- Case 1 and 2 (10% Synthetic and 20% Synthetic) - The model achieved an accuracy in the range of 70% 85%.
- Case 3 (30% Synthetic) - The accuracy dropped to the range of near 40%.
- The accuracy continued to show a slight decreasing trend, with the model maintaining accuracy in the range of 30% to 40% in the 50% synthetic data scenario.

5.3 Interpretation of Results

For sequential anomaly injection, the observed trend of increasing accuracy with higher proportions of real data suggests that the model becomes more confident and accurate as it processes longer sequences of normal behavior before encountering anomalies. The results showed that the model needs to see at least 70% of real data before it achieves approximately 90% accuracy.

On the other hand, in the case of random anomaly injection, the slight decrease in accuracy with increasing proportions of synthetic data indicates the challenge posed by random noise. However, the model's performance remains robust, demonstrating its capability to handle at least 20% levels of noise and still able to detect anomalies.

The model's performance across the above scenarios highlights both its potentials and limitations. Results from both experiments, showed that even with similar amount of synthetic anomaly injected, accuracy varies depend on where the anomaly is injected. Our model works best when anomalies are injected after a long sequence of real data. This proves that LSTM networks are able to reliably detect patterns within the sequences of players movement. When anomalies are introduced in a random manner, the LSTM network still effectively distinguishes between normal and anomalous behaviors, but the performance start to drop after significant noise (anomaly) is inserted. This is because the more random data is inserted, the shorter the sequences of real data are seen by the model.

Finally, the high accuracy achieved in synthetic anomaly injection methods underscores the potential for real-time anomaly detection. The ability to promptly identify anomalies during live gameplay sessions can significantly enhance the integrity of online games by preventing cheating and ensuring fair play.

The framework can be generalized to various online games, providing a scalable solution for anomaly detection across different esports platforms. This generalizability is crucial for game developers and administrators aiming to maintain fair play and enhance the player experience across multiple games.

6 FUTURE WORK

While the results are promising, it is important to acknowledge certain points. The use of synthetic anomalies provides a controlled testing environment, but real-world anomalies may exhibit different characteristics. Future research should involve validating the model with real-world anomalous data to confirm its effectiveness in practical scenarios.

While CSGO serves as a useful case study, the model's performance may vary when applied to other games with different dynamics and player behaviors. Further experiments are needed to ensure the model's broad applicability across various gaming environments.

Future work should involve testing the model with other games' real-world anomalous data to validate its effectiveness beyond synthetic scenarios. Integrating additional contextual features such as player interactions and in-game events could provide a more comprehensive understanding of player behavior and improve anomaly detection accuracy. Additionally, exploring transfer learning techniques could enable the application of the trained model to different games with minimal retraining, broadening its applicability across various gaming platforms. Finally, ensuring the model's scalability for real-time deployment in large-scale gaming environments is crucial. Optimizing the model for efficient processing and handling high data throughput will be important steps in this direction.

7 CONCLUDING REMARK

The exploration of anomaly detection in video games through in-game behavioral analysis using deep recurrent neural networks present a promising avenue for enhancing fairness, and enjoyment within online

gaming communities. Our proposed approach, making use of advanced machine learning techniques, has demonstrated effectiveness in identifying anomalous behaviors within a gaming environment.

By harnessing the power of deep learning, we have developed an adaptable anomaly detection system capable of analyzing player behaviors and flagging deviations indicative of disruptive actions such as impersonation. The experimental results highlight the potential of our approach to promote a fair and enjoyable gameplay experience free from the effects of disruptive behaviors.

Furthermore, fostering collaboration between researchers, game developers, and players is essential for ensuring the responsible deployment and ethical use of anomaly detection technologies. By engaging the whole community in discussions surrounding privacy, autonomy, and transparency, gaming industry can collectively strive towards creating an environment that prioritizes fairness and mutual respect.

In conclusion, while there are challenges and opportunities, our approach towards applying deep neural networks for anomaly detection in video games represents a step forward in advancing the state-of-the-art models, fostering a more enjoyable gaming experience for all.

ACKNOWLEDGEMENTS

This work was supported by JST Moonshot R&D, Grant Number JPMJMS2215.

REFERENCES

Ackerson, J. M., Dave, R., and Seliya, N. (2021). Applications of recurrent neural network for biometric authentication and anomaly detection. Information, 12(7), 272.

Akoglu, L., Tong, H., and Koutra, D. (2015). Graph based anomaly detection and description: a survey. Data mining and knowledge discovery, 29, 626-688.

Alabadi, M., and Celik, Y. (2020, June). Anomaly detection for cyber-security based on convolution neural network: A survey. In 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA) (pp. 1-14). IEEE.

Chen, V. H. H., and Ong, J. (2018). The rationalization process of online game cheating behaviors. In Information, Communication and Society, 21(2), 273-287.

Conroy, E., Kowal, M., Toth, A. J., and Campbell, M. J. (2021). Boosting: Rank and skill deception in esports. In Entertainment Computing, 36, 100393.

Dinh, P. V., Nguyen, T. N., and Nguyen, Q. U. (2016). An empirical study of anomaly detection in online games. In 2016 3rd National Foundation for Science and Technology Development Conference on Information and Computer Science (NICS) (pp. 171-176). IEEE.

Fernandes, G., Rodrigues, J. J., Carvalho, L. F., Al-Muhtadi, J. F., and Proença, M. L. (2019). A comprehensive survey on network anomaly detection. Telecommunication Systems, 70, 447-489.

Funk, D. C., Pizzo, A. D., and Baker, B. J. (2018). eSport management: Embracing eSport education and research opportunities. In Sport Management Review, 21(1), 7-13.

Ghoshal, A. (2019). Ethics in esports. In Gaming Law Review, 23(5), 338-343.

Goh, J., Adepu, S., Tan, M., and Lee, Z. S. (2017, January). Anomaly detection in cyber physical systems using recurrent neural networks. In 2017 IEEE 18th International Symposium on High Assurance Systems Engineering (HASE) (pp. 140-145). IEEE.

Hosseinzadeh, M., Rahmani, A. M., Vo, B., Bidaki, M., Masdari, M., and Zangakani, M. (2021). Improving security using SVM-based anomaly detection: issues and challenges. Soft Computing, 25(4), 3195-3223.

Irvan, M., Thao, T. P., Kobayashi, R., Nakata, T., and Yamaguchi, R. S. (2021). Learning from Smartphone Location Data as Anomaly Detection for Behavioral Authentication through Deep Neuroevolution. In ICISSP (pp. 723-728).

Lindemann, B., Maschler, B., Sahlab, N., and Weyrich, M. (2021). A survey on anomaly detection for technical systems using LSTM networks. Computers in Industry, 131, 103498.

Merity, S., Keskar, N. S., an d Socher, R. (2017). Regularizing and optimizing LSTM language models. arXiv preprint arXiv:1708.02182.

Nguyen, T. T., Nguyen, A. T., Nguyen, T. A. H., Vu, L. T., Nguyen, Q. U., and Hai, L. D. (2015, December). Unsupervised anomaly detection in online game. In Proceedings of the 6th International Symposium on Information and Communication Technology (pp. 4-10).

Pang, G., Shen, C., Cao, L., and Hengel, A. V. D. (2021). Deep learning for anomaly detection: A review. ACM computing surveys (CSUR), 54(2), 1-38.

Rizani, M. N., and Iida, H. (2018, October). Analysis of counter-strike: Global offensive. In 2018 international conference on electrical engineering and computer science (ICECOS) (pp. 373-378). IEEE.

Rosell Llorens, M. (2017). eSport gaming: the rise of a new sports practice. In Sport, ethics and philosophy, 11(4), 464-476.

Sherstinsky, A. (2020). Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. Physica D: Nonlinear Phenomena, 404, 132306.

Xenopoulos, P., and Silva, C. (2022). Esta: An esports trajectory and action dataset. arXiv preprint arXiv:2209.09861.