Eötvös Loránd University

Faculty of Informatics

Dept. of Software Technology and Methodology

The usage of various AI models and techniques for cheat detection in online multiplayer games

Supervisor:
John Doe
Assistant Lecturer

Author:
Gémesi Szabolcs, Görcsös Gergely, Nagy Ri
Computer Science BSc

Budapest, 2024

Contents

Chapter 1

Abstract

If you like to spend your free time playing video games there is a high likelihood that you experienced one of the biggest problems in the gaming industry: cheating. Cheating ruins the competitive integrity of video games, and when it is widespread it can even damage the game by making the players leave the game, causing a dwindling player base. Game developers have tried numerous ways of fighting cheats, with varying success rates. A more innovative and less explored way of anti-cheat development involves artificial intelligence. This paper aims to further explore the possibility and effectiveness of using AI for cheat detection.

The problem covers a wide variety of different platforms and different types of video-games available on the market. In order to get the best results possible we try solving the problem with a combination of different AI powered techniques, namely DNNs, explainable AI and more.

By implementing, testing, and deploying our AI-driven anti-cheat solution we aim to demonstrate that AI can be more effective in identifying cheaters than traditional anti-cheating measures. Our finished system gives game developers a chance to significantly reduce cheating in their games, and maintain the integrity of competitive gaming.

Chapter 2

Introduction

After so many years and so many games, online games are still so popular that the end is nowhere in sight. Millions buy not just the latest software and games but also the environment, such as PCs and consoles just to play their favorite games at their best. For them, the most terrifying thing is when they encounter cheaters in the game. Of course it is bad if you lose in the game or cannot complete the mission, but the game itself can be destroyed if there are a lot of hackers and cheaters. This work shows how game developers can prevent cheaters from appearing in the game and thereby save their product.

Filtering out the cheaters is not new. There are a lot of work, research which process this problem and try to show how the developers can defend against it. In our paper, we will go through the best current methods available for game developers and using that information we will try to provide a new solution for filtering out the cheating users. We also show what is the biggest drawbacks of the aforementioned researches, why our solution is able to fill these holes and at the same time we prove why our innovation can do more than the others.

Our solution uses AI technologies such as [LIST OF AI MODELS/TECHNOLOGIES]. With our innovation we are making profiles and generate scores for the user actions, with the mentioned AI technologies, and filtering out the potentially engaged users in cheating activity. We also try to cover as many online game type as possible with our system. We are trying to provide a detector app that knows after a very short time if there is a cheater in the game.

With this research we are hoping to help the game developers to develop safer games and to provide more stable games for those, who are love games, who are

like to play online games. We presented this work to show that cheaters are still a problem to the gaming community, and to show that there are ways to prevent their negative impact on the gaming industry. We are hoping that with this work, we can help to reduce the cheating users in game, and also we hope that more and more new game developers and their new games will appear, because they can develop fast and effectively cheat detection to their system, which saves their product from the cheaters.

The structure of the paper is as follows: chapter 3 contains the most important papers of the topic with a short overview about their work. chapter 4 provides information about the current solutions in the industry. chapter 5 provides our theory for cheat detection and a few examples why it is works and why it is better than the current methods. chapter 6 concludes and depicts possible further future work.

Chapter 3

Literature review

3.1 Cheating and Detection Method in Massively Multiplayer Online Role-Playing Game: Systematic Literature Review [1]

The main topic of this review is the examination of possible cheat in MMORPG games, the exploration of its causes, introduction of its types and presentation of a possible detection method. In the reaserch, they discussed the structures of MMORPG games and described the types of cheating behaviours.

The authors show countermeasures to prevent cheating which comes from the structure of server and from the player's behavior. They classified the detecting methods into five groups: statistics analysis, data mining, similarity analysis, and network-based analysis and matched with the cheating behaviours. Finally, they showed that how they can detect cheaters with machine learning algorithms.

This paper provides extensive research on various occurrence cheating methods and solutions for detecting cheaters in MMORPGs.

3.2 Robust Vision-Based Cheat Detection in Competitive Gaming [2]

The online first-person shooter game, Counter-Strike 2 (Formerly known as Counter-Strike: Global Offensive) is infamous for its cheating problem and scandals. There have been several instances of professional players getting caught cheating on

stage. In the paper "Robust Vision-Based Cheat Detection in Competitive Gaming" we can explore an already existing solution that tries to tackle the cheating problem in Counter Strike - Global Offensive, and another first-person shooter game. The main idea of this method of cheat detection is using DNNs (deep neural networks) to look for rendered frames, that give away visual evidence of a cheating software being used. This can be information in the game that the player should not have (seeing otherwise invisible enemies through walls, or on the map), or the GUI (graphical user interface) of the cheat itself. The way it is implemented is very straightforward, yet novel. The GPU's frame buffer is captured right before it appears on the computer screen, and is analyzed by a trained DNN. The DNN is trained to look for altered and suspicious pixel patterns. The model's training took place on a large set of frames from the games with and without cheats. It calculates both the probability of a frame containing cheats and the confidence value, which is used both to avoid false positives and to help the developers know when the model needs to be retrained, for example when a cheat software got changed, to avoid detection. One of the major advantages of this method of cheat detection is its regard for privacy. Most anti-cheat software accesses the whole memory, leaving room for potential privacy breaches and data leaks. Some anti-cheats also operate on the kernel level (Vanguard - Valorant). If these are coded poorly and can be exploited they can create severe security vulnerabilities. Compared to these solutions visual cheat detection can be considered less intrusive, and more safe regarding data protection. The results were auspicious: The more visually prominent the cheating was, the better the system got at successfully detecting the presence of cheats. Different configurations were mentioned, and the exact results varied based on the configuration in use. The highest overall accuracy measured was 0.89. The developers noted, that this implementation should be used as an extension of already existing cheat detection methods. Overall in my opinion it is a great way to deal with cheaters in online games, and it is worth exploring further when it comes to AI cheat detection.

3.3 Deep learning and multivariate time series for cheat detection in video games [3]

This research introduces a novel cheat detection system for online video games that, instead of traditional in-game data, is based on human-computer interaction (HCI) data, such as keystrokes and mouse movements. Existing systems use game-specific customization to detect cheating, while this approach uses player interaction data as multivariate time series analyzed by convolutional neural networks (CNNs). This makes the system adaptable to various games and capable of detecting cheaters without any game-specific information. The method successfully identifies cheats like aimbots and triggerbots in first-person shooter (FPS) games by recognizing behavioral patterns that differ from typical human interactions.

The study was performed on data collected from players of Counter-Strike: Global Offensive, achieving detection rates of 99.2% for triggerbots and 98.9% for aimbots. Cross-validation was used to ensure the model's robustness and ability to adapt to new player inputs, demonstrating its effectiveness in real-world scenarios. The use of universal input data makes this system, compared to other systems, a flexible and scalable solution for cheat detection. This makes it easy to apply to other games without the need for any specific tuning. This research provides significant advancements in cheat detection methodologies by a game-independent approach that can also adapt to new and evolving forms of cheating.

3.4 Explainable AI for Cheating Detection and Churn Prediction in Online Games [4]

The paper explores the application of AI and XAI (eXplainable AI) in online gaming, focusing on issues such as game cheating detection and player churn prediction. It discusses how AI has historically been applied in games for tasks like matchmaking and behaviour monitoring. XAI specifically addresses the black-box nature of AI models, enhancing trust by providing explanations for AI decisions. The authors introduce datasets from NetEase Games, using a large amount of game logs, player behaviour sequences, client images, and social graphs to improve the transparency and accuracy of AI models.

The authors concluded that the use of XAI could be preferred over the black-box model having received more positive feedback from game operators and game designers alike for its' transparency in reasoning and explanation. These results are far better than the authors expected and they claim that this technology should be further examined and developed to achieve even better results.

Chapter 4

Background

Chapter 5

Theory

Chapter 6

Conclusion