Proceedings of the Twenty-First AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE 2025)

Signals of Struggle: Detecting Player Difficulties Using Machine Learning

Nabeeha Ali, David Thue

School of Information Technology, Carleton University, 1125 Colonel By Drive, Ottawa, ON, K1S 5B6, Canada
nabeeha.ali@cmail.carleton.ca, david.thue@carleton.ca

Abstract

Struggle is an inevitable part of gameplay, and it's often what makes games meaningful, rewarding, and fun. Still, some moments of difficulty spiral into frustration or confusion, and can cause players to quit entirely. Being able to detect and interpret struggle is thus essential for designing better player experiences, but identifying these moments remains challenging. We present a machine learning approach for detecting player struggle in real time using gameplay telemetry. Using three quests built in *Terraria* that each emphasize a different set of game mechanics – gathering, combat, or crafting – we collected data on how players interact with different systems and had them annotate where they encountered difficulty. Using this dataset, we trained Random Forest classifiers and evaluated model performance across different feature sets, window sizes, and step sizes. Our results show that such a model can successfully identify whether unseen players are experiencing struggle in the crafting quest, while the other quests proved more difficult. We also tested whether a model could classify the type of struggle as cognitive or performative, and found promising results for the crafting and combat quests. Our findings demonstrate the potential of using player telemetry to detect struggle, laying the groundwork for future adaptive systems that offer real-time, context-aware support tailored to individual player needs.

1 Introduction

Not everyone experiences a game in the same way; what feels like the perfect challenge for one player might seem impossible – or boring – to another. Csikszentmihalyi's (2014) flow theory suggests that optimal experiences occur when the game's challenges match the ability of a player, fostering intrinsic motivation and sustained engagement. However, players bring diverse abilities, preferences, and play-styles that shape how they interact with a game's mechanics and objectives. Because of this diversity, crafting a one-size-fits-all experience is extremely difficult. Even when a game is well balanced for its target audience, there will always be players who fall outside that range, risking frustration, boredom, or early disengagement. Achieving the right balance is a critical first step, but it may not always be enough. Supporting and retaining as many players as possible means going beyond static difficulty design and exploring more dynamic, personalized approaches to challenge.

To support broader accessibility and improve player experiences, designers require a strong understanding of when and how players struggle. In this context, *struggle* refers to moments of heightened effort or challenge that require the player to adapt, reflect, or persist. These might not necessarily be negative experiences, but signs that the game is pushing the boundaries of the player's current ability. Struggle is an inherent part of gameplay and can stem from a wide range of sources. Denisova et al. (2020) developed the Challenge Originating from Recent Gameplay Interaction Scale (CORGIS) to address this complexity. CORGIS provides a systematic framework that captures multiple dimensions of difficulty that players may face, offering a clear way to distinguish between different types of challenges that can be applied across various games and player skill levels. These include *cognitive challenges*, which arise from the need for preparation, planning, memorization, effort, and multitasking; and *performative challenges*, which emerge from the game requiring rapid and accurate action from the player. Unlike general measures of engagement or satisfaction, CORGIS focuses specifically on struggle, making it especially useful for balancing gameplay and designing adaptive support systems. Understanding the different ways players experience difficulty sheds light on not only how struggle manifests, but also what struggle means from the player's perspective. While it is a relatively recent tool, CORGIS has already begun to be adopted in other studies, highlighting its growing relevance in game research (Frommel, Klarkowski, and Mandryk 2021; Foffano 2023; Hegedues et al. 2023; Cuerdo, Baskaran, and Melcer 2024).

Detecting moments of struggle remains difficult, as player behaviour is inherently unpredictable. Traditional game development relies heavily on prototyping and playtesting, which can be very resource-intensive. As a result, there is growing desire for player models that spot signs of difficulty early in the design pipeline (Ariyurek, Betin-Can, and Surer 2021; Ali 2025). Such models can provide designers with actionable insights about when and where players might encounter problems, leading to improved game design.

In this paper, we explore a way to automatically detect whether a player is currently struggling based on their recent gameplay telemetry, and, if so, whether that struggle

is cognitive or performative. Our contributions include a refined methodology for constructing a labelled dataset for this task (Ali 2025), a data gathering platform to support collecting such data through a player study, a method for preparing the data and training classifiers to identify moments of struggle from gameplay telemetry, and empirical results from training and evaluating several such classifiers. By following our process, developers can train their own models to detect player struggle using playtest data, and then use those models to iteratively improve their designs. We aim to advance the field of adaptive game design by providing designers and developers with tools to better understand and respond to player needs across diverse gameplay scenarios.

2 Related Work

Previous research has explored the use of machine learning to detect player struggle in video games. Liu et al. (2023) applied predictive models to gameplay logs from *Wake: Tales from the Aqualab* (Field Day Learning Games 2023), collecting anonymous telemetry data from 501 students across 3,859 hours of gameplay. Although their best performing model demonstrated some ability to identify moments of struggle (achieving an AUC score of 0.635), the approach faced several limitations. The definition of "struggle" in their study was broad, without clearly distinguishing between issues rooted in user interface design and those tied to skill-based challenges. They also derived labels for struggle from researcher annotations, which may not have fully captured the players' actual experiences. To help address these challenges, we adopt a more structured definition of player struggle using the Challenge Originating from Recent Gameplay Interaction Scale (CORGIS) (Denisova et al. 2020). We also have players label their own moments of struggle, which reduces ambiguity and aligns the model's labels more closely with how players perceive difficulty in real time.

This paper directly follows our recent research (Ali 2025), which outlined how to create an annotated dataset related to players' struggles in games. Our method involved asking players to review their recorded gameplay footage after each play session and describe each moment where they struggled. Unfortunately, this method suffers from a key limitation: finding the parts of the footage that include a player's struggles requires reviewing most or all of the footage, much of which contains no struggle. We developed a way to address this limitation, which we describe in Section 3. We also previously proposed using the game *Terraria* (Re-Logic 2011) to collect player data and designed three quests with varied mechanics to do so, focused on Gathering, Combat, and Crafting respectively. We used these quests and their telemetry-recording system as the foundation of our data gathering platform (Section 3). This paper extends beyond our previous work by explaining how we collected annotated data from a group of human participants, trained machine learning models to predict players' struggles, and analyzed the results to offer new insights about this challenge.

Hegedues et al. (2023) investigated how both mechanical and visual changes in game design affect player challenge, using EEG bandpower analysis to measure brain activity. Twelve participants played four different minigames, where either the mechanics or visuals were altered to increase difficulty. The results showed that frontal lobe activity increased across all mechanically challenging versions and in two of the four visually altered versions, but these activities were not consistent with players' self-reported experiences. The use of EEG in real-time gameplay also presents practical limitations, including the need for specialized equipment and the potential for the apparatus to interfere with natural play. Our work leverages gameplay telemetry data to identify moments of struggle, which avoids disrupting the player's experience and needs no specialized equipment.

Dynamic difficulty adjustment (DDA) (Hunicke 2005) has been widely studied as a means of enhancing player experience by automatically adapting game difficulty in response to player performance. One approach by Frommel et al. (2018) introduced emotion-based DDA, where self-reported levels of frustration and boredom were used to trigger difficulty adjustments. A user study with 66 participants showed that their method improved player experience and perceived competence compared to static or linearly increasing difficulty. While this work highlights the value of emotional feedback in adaptive systems, emotions alone may not fully capture the underlying reasons why players struggle. Emotional states like frustration or boredom are often symptoms of deeper issues, such as cognitive overload or difficulty with execution, which may go unaddressed if the system lacks awareness of the root cause (Cao, Sweetser, and Zhu 2023; Bopp, Mekler, and Opwis 2016). Our work builds on this idea by shifting focus from general emotional cues to detecting concrete moments of player struggle, as inferred from gameplay behaviour. By detecting whether a player is struggling, our approach aims to provide a clearer, more actionable signal that adaptive systems can use to intervene or support the player in real time.

The notion of struggle has been explored in various game genres. Puzzle games often frame difficulty as a function of cognitive uncertainty (Chen, White, and Sturtevant 2023), solution structure (Linehan et al. 2014), or player learning curves (Kristensen, Valdivia, and Burelli 2021). VR games introduce physical and spatial elements that add a new layer of embodied challenge (Cuerdo et al. 2023), while turn-based games emphasize mental effort and risk assessment under constrained conditions (Cao, Sweetser, and Zhu 2023). These differences demonstrate the importance of context when analyzing struggle and suggest that adaptive systems may need to be genre-aware to be effective. While some approaches focus on modelling player behaviour quantitatively, others emphasize design patterns or emotional responses to difficulty. However, a notable gap remains in research on struggle in open-ended sandbox or quest-driven games – genres where player goals are more emergent and gameplay is less linear. This paper begins to address this gap by exploring how struggle can be detected and interpreted in these more flexible and player-directed environments.

Whitby, Deterding, and Iacovides (2019) surveyed 101 players on the struggles they encountered while playing a game and found that most players experience endo-transformative reflection – a form of critical thinking focused on their own strategies, decisions, or understanding

Figure 1: Gameplay footage from our *Terraria*-based Crafting quest. Our data gathering platform automatically embeds a red warning icon in the footage (bottom right) whenever a player indicates that they are struggling.

of the game system. While that research offers insights into how players perceive their struggles after the fact, it does not address the timing of such struggles during gameplay, which is arguably more important for enabling effective, in-game support. Our work addresses this need by building a machine learning model that detects moments of struggle as they happen, allowing for real-time intervention.

3 Data Gathering Platform

To collect annotated data on player struggle, we used a set of three custom quests for the game *Terraria* (Re-Logic 2011) that we previously developed for this purpose (Ali 2025).

Terraria is a 2D sandbox game where players explore, build, fight, and survive in a procedurally generated world filled with diverse biomes (see Figure 1). Players begin with basic tools and guidance from an NPC, then collect resources to craft equipment, build shelters, and fight enemies. *Terraria* is well-suited to gathering data about player struggle because it has a broad set of mechanics that reflect those found in many game genres, supporting the applicability of our work. Compared to the more linear structure of puzzle or narrative-driven games used by related work (Ang and Mitchell 2017), *Terraria*'s open-ended gameplay offers the flexibility to deploy controlled scenarios that still feel natural to the game's environment.

Each of our quests focuses on a distinct type of game mechanic – gathering, combat, and crafting – which helps us consider how players engage and struggle through different forms of interaction. The structure of each quest follows Yu, Sturtevant, and Guzdial's (2021) quest definition framework, including: a set of tasks completed according to a partial order, each defined by the number of players that can complete that task; conditions for task completion; a way to present each task to the player; a system to monitor the completion of each task; rewards for completion; and a way to distribute those rewards among players. Each quest is completable in under 10 minutes by players of varying skill levels, while still presenting meaningful gameplay challenges.

To improve the efficiency of our proposed data annotation method in utilizing each participant's time (Section 4.3), we added a feature to the footage-recording part of our platform. Specifically, whenever a participant pressed the LEFT SHIFT key on the keyboard during gameplay, a highly visible red icon would be added as an overlay in the gameplay footage (see Figure 1). By asking each participant to use this feature whenever they experienced struggle, we made it much simpler to identify those periods while the footage was being reviewed. We built this functionality using the Advanced Scene Switcher plugin for Open Broadcasting Software (OBS) (Warmuptill 2016; Lain 2025).

4 Data Collection

After receiving ethics clearance from our university's Research Ethics Board, we recruited participants through email distribution and posts in the student and gaming Discord servers of groups located in Ottawa, Ontario. For eligibility, participants needed to be at least 18 years old, have normal or corrected-to-normal vision, be proficient in English, and have some prior experience with video games, regardless of skill level. They also needed to be comfortable using computers and digital interfaces. Each participant received an Amazon e-gift card worth $20 CAD as compensation. We anonymized the data to protect participants' privacy. Each session took at least one hour per person.

4.1 Demographics

We first asked participants to complete a demographic questionnaire that collected information on their age range, gender, race, and disability status. To assess their gaming background, we also inquired about how many hours per week they typically spend playing video games, which helped us gauge their general experience level (e.g., beginner: $<$5 hours, intermediate: 5-10 hours, or advanced: $>$10 hours). We also asked participants whether they had played *Terraria* before and, if so, to rate their familiarity with the game. We also gathered information about participants' preferred game genres to better understand individual play styles and preferences. Lastly, we asked about how participants typically respond to in-game challenges, offering options such as quitting due to frustration, taking breaks before retrying, or pushing through to continue playing. Together, these responses provided important context for understanding the diversity of player experiences represented in the data.

We collected data from twelve participants. Of these, ten identified as a woman, one as a man, and one as gender-fluid. Ten participants were 18–25 years old, one was 25–30, and one was 31–40. Four participants were White, one was Middle Eastern, one was Latino, one was Southeast Asian, and five participants were South Asian. Gaming experience varied, as nine participants reported playing fewer than 5 hours per week, two played 5–10 hours, and one played 11–20 hours. Two participants had prior experience with *Terraria*, with one identifying as a beginner and the other as an advanced player. Most (7) participants reported that when encountering difficulty in games, they tend to push through and continue playing, rather than quitting or taking breaks.

4.2 Playing the Quests

Before they started playing, participants had access to a printed instruction sheet outlining the game's keybindings, including how to perform tasks such as movement, inventory access, and crafting. The instructions also provided step-by-step descriptions on how to collect plants, craft items, and engage in combat within the game. We informed them that they could refer to this sheet at any point during the session to ensure stronger familiarity with the game's controls. Participants also received an explanation of the quests' objectives through NPC dialogue, verbally by the researcher, and via the instruction sheet, which helped us ensure that any struggles observed could be attributed to gameplay mechanics rather than confusion about the task.

Throughout the session, participants played each of the three custom quests in a fixed order: Gathering, Combat, then Crafting, each lasting up to 10 minutes (without pausing) or until the quest was completed. We screen-recorded gameplay using OBS (Lain 2025), and automatically captured gameplay telemetry including game-state information and player actions. The logging system added a new entry 60 times per second (matching the game's frame rate), resulting in up to 36,000 log entries per participant for each of the three quests. We instructed each participant to press the LEFT SHIFT key whenever they experienced a moment of struggle, to mark their gameplay footage as shown in Figure 1. We also modded the game interface to display real-time progress updates in white text at the bottom left of the screen. Whenever no updates were visible, an on-screen message instructed the player to press ENTER to open the chat log and review their progress. This setup helped ensure a consistent gameplay experience across participants.

4.3 Footage Annotations

After completing each quest, participants reviewed their recorded gameplay footage and used an Excel spreadsheet to annotate their moments of struggle. We involved participants in this process to ensure that their personal experiences of struggle were directly reflected in the dataset. We provided a consistent definition of struggle to guide their annotations, with the goal of capturing how each player interpreted and recognized struggle in the context of their own gameplay. To streamline the process and reduce the cognitive load of retrospective reflection, we instructed participants to locate the red markers that had been automatically added to the footage whenever they signaled a moment of struggle. They then reviewed the surrounding gameplay footage (spanning a few seconds around each marker) to identify and define the start and end of the struggle that they experienced. We also asked them to provide details about each period of struggle, including a written description of their challenge, the mechanic they struggled using (options varied depending on the quest), and the type of struggle (cognitive or performative).

While participants were encouraged to annotate as much of the footage as possible, time constraints limited their ability to review every moment. We instructed them to proceed through the footage from start to end, to preserve consistency in what was considered a completed struggle annotation when processing the data.

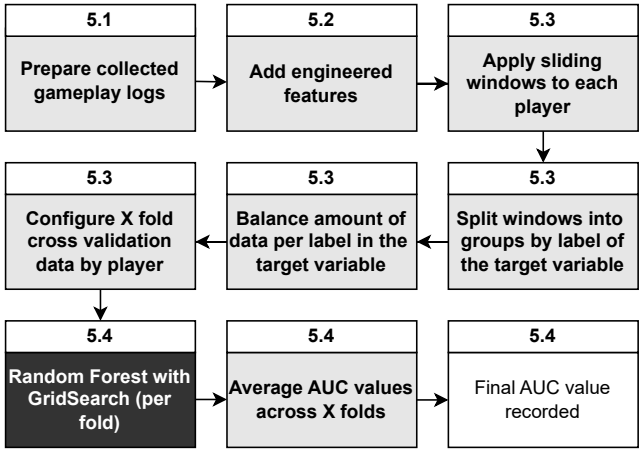

Figure 2: Overview of our experimental pipeline for training Random Forest classifiers with target window and step sizes. We applied this process separately for each of the two target variables (struggle vs. non-struggle, and cognitive vs. performative), and for each quest individually. The number of cross-validation folds (X) varied depending on the amount of data available in each case, as reported in Tables 2 and 3.

Following the annotation process, participants completed a short post-experiment questionnaire to provide feedback on their experience. This included reflections on the clarity of the tasks, the appropriateness of the quest challenges, and any issues they encountered during their participation.

5 Building the Models

We now present our method for training and testing models to detect two target variables from gameplay telemetry: struggle versus non-struggle, and performative struggle versus cognitive struggle; Figure 2 shows a high-level overview.

5.1 Dataset Preparation

The result of our footage annotation process is a dataset that marks the start and end times of multiple periods of player struggle, along with the mechanic that the player struggled with and whether the struggle was cognitive or performative. To ensure label quality, we validated each player's annotations by checking whether they had selected valid values for the mechanic. If a participant selected "other", we manually reviewed their written description to classify it into an existing mechanic or, if necessary, introduced a new one. If a player's annotation referred to a struggle that was unrelated to player-controllable mechanics (e.g., confusion about enemy behaviour), we re-labelled it as a non-struggle, as our focus was on detecting struggles through player actions. Two participants reported no struggles in the Combat quest, and one participant reported no struggles in the Crafting quest, stating that the mechanics were intuitive and that more time might have made challenges more apparent. For these sessions, we labelled every log entry as a non-struggle.

We combined the annotations with the gameplay telemetry as follows. Using Python, we converted every player's

log file into a structured dataframe (one row per log entry) and then separated the result into the three distinct quests (Gathering, Combat, and Crafting). To build label columns for our two target variables, we used timestamp matching to automatically align each player's quest telemetry with their gameplay footage, taking into account any offsets between when the screen recording began and when the player initiated a quest. Since some periods of struggle at the end of each quest lacked completed annotations (Section 4.3), we discarded all telemetry and footage that came from times after each player's last completed annotation for each quest. Doing so allows us to safely assume that any retained data that falls *outside* of an annotated struggle period is a non-struggle period. For each retained log entry, we set its struggle/non-struggle and cognitive/performative labels based on the annotated struggle periods (non-struggle entries get no second label). Finally, we exported the resulting datasets as a single, structured CSV file.

5.2 Feature Engineering

To augment our dataset, we engineered features based on the gameplay telemetry. One feature was "Time Elapsed", which we extracted directly from logs. We also introduced a numerical feature, "Progression", to indicate how much of the quest had been completed at a given time. To engineer this feature, we created user journeys for each quest, mapping key gameplay milestones to percentage markers along a linear progression scale (Figure 3). This feature lets us quantify how far along a player was in the intended quest flow.

This process closely parallels the work of Teng et al. (2025), who introduced "player journeys" as interactive node-edge graphs that represent sequences of player actions, for visualizing and identifying problem-solving strategies in games. Similarly, our user journeys represent the intended paths through the quest space, providing a reference structure against which we can interpret real-time player actions. While Teng et al.'s (2025) work used these journeys primarily for visual exploration and segmentation (e.g., highlighting patterns specific to experts or atypical subgroups), we extend the idea to feature engineering for classification. By providing information about how far a player has progressed, our Progression feature could help the model infer deviations from expected progressions, which might suggest confusion, exploration, or alternate strategies – behaviours that might otherwise be dismissed as noise.

We also filtered the available telemetry features differently for each quest, to ensure that its final dataset only included features that pertained to its unique objective (see Table 1). For example, we excluded "Weapon items" from the Gathering quest dataset because weapons were unavailable.

5.3 Data Splitting

To prepare the dataset for machine learning, we loaded each player's CSV file and tagged it with a source file identifier to maintain player context. We standardized all Boolean values to TRUE/FALSE and encoded all categorical features using scikit-learn 1.6.1's LabelEncoder (Pedregosa et al. 2011).

To convert (effectively) continuous gameplay telemetry and labels into sequences suitable for supervised learning,

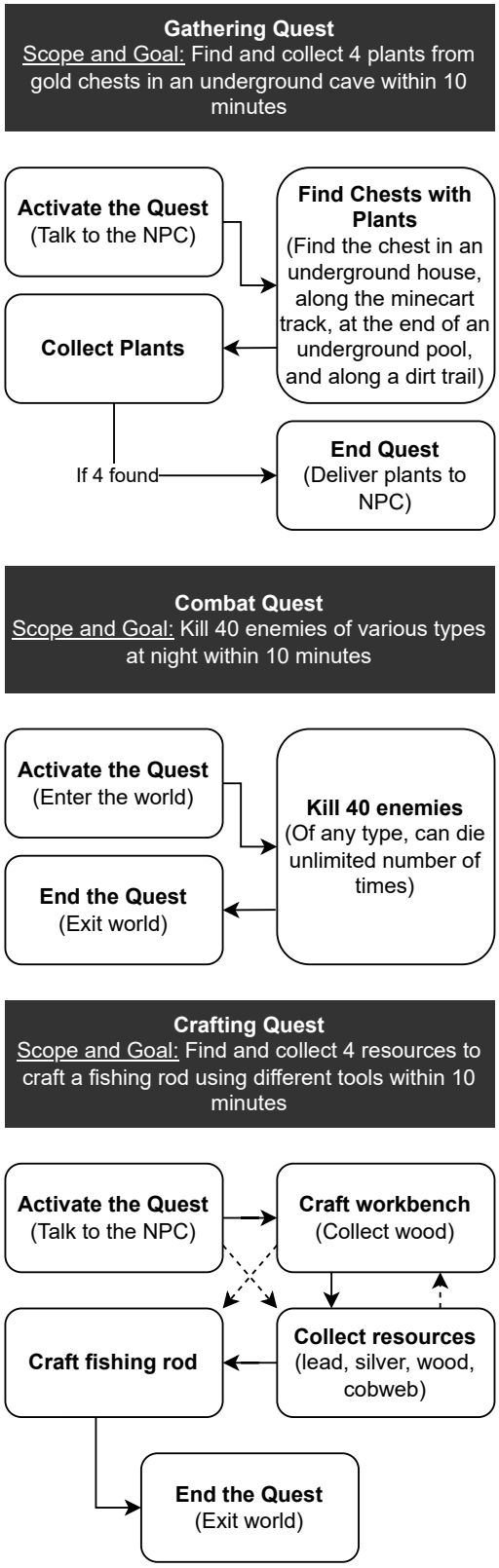

Figure 3: User flow diagrams illustrating the sequence of actions that players were expected to take in each quest, following Teng et al. (2025). Dashed lines show alternate paths.

Features	Gathering	Combat	Crafting
Raw from Gameplay			
Time stamp	✓	✓	✓
Quest name	✓	✓	✓
Quest started	✓	✓	✓
Quest completed	✓	✓	✓
Player location	✓	✓	✓
Tile at location	✓	✓	✓
Control binding used	✓	✓	✓
Item held by player	✓	✓	✓
Plant items	✓		
Tool items			✓
Weapon items		✓	
Current health	✓	✓	✓
Times died	✓	✓	✓
Chatting with NPC	✓		✓
Player hitting enemy		✓	
Using minecart	✓		
UI interaction	✓	✓	✓
Engineered			
Time elapsed	✓	✓	✓
Progression	✓	✓	✓
Labels			
Struggled	✓	✓	✓
Struggle Type	✓	✓	✓

Table 1: List of features used for training, separated by quest.

we applied a sliding window technique (Figure 4). This method transforms each player's data into multiple fixed-length segments, enabling the model to detect temporal patterns associated with player struggles. Detecting temporal patterns is important because a struggle is rarely isolated to a single moment; instead, it unfolds over a series of interactions. Each sliding window is defined by a *window size* (the number of data rows per segment) and a *step size* defines the offset between consecutive windows. We flatten the data within each window into a feature vector, and the corresponding labels – Struggled (Yes or No) and Struggle Type (Cognitive or Performative) – are determined by the labels attached to the final row in the window.

In cases where the number of available non-struggle windows was lower than the number of struggle windows, we down-sampled the player's set to preserve data balance; this occurred for one player's data in the Gathering Quest, three in the Combat Quest, and three in the Crafting Quest. Although balancing our data reduced the total number of samples that we used for training, doing so can support better generalization by preventing bias toward the majority class. To support generalization to new players and balance struggle versus non-struggle examples, we separated the data into training and testing sets by player, dropping the column that identified each player to avoid data leakage.

Our goal was to test using leave-one-out cross-fold validation, but not every player's data was rich enough to use for this purpose. As a result, we varied the number of folds per quest and target label, based on the total number of players with eligible data for that quest/label pair. To be used for testing the struggle/non-struggle models, a player needed to have struggled and not struggled at least once (each) during that quest. To be used for the cognitive/performative models, a player must have exhibited at least one struggle of each type. When training and testing models that predicted the type of struggle, we only used windows that were labelled TRUE for struggle/non-struggle. We report these varying numbers of folds alongside our results in Section 6.

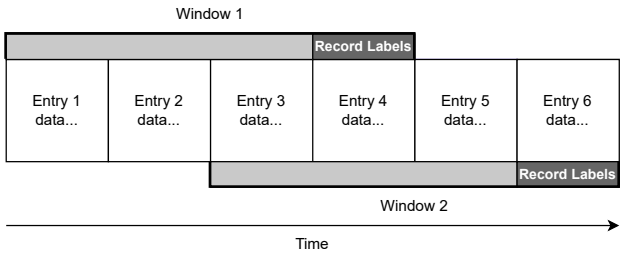

Figure 4: Visualization of our sliding window technique for segmenting gameplay logs. Each window captures a fixed-length sequence of data entries (here, window size = 4), with the labels of the final entry in the window used as the labels for that window. In this example, windows overlap by 50% of their length (step size = ½).

5.4 Model Selection and Training

We trained binary Random Forest classifiers for each quest and each target variable: player struggle and type of struggle (cognitive or performative). We chose Random Forests based on their general capacities to handle mixed feature types, capture nonlinear interactions, and perform well with noisy datasets, all of which are relevant given the complexity of gameplay telemetry (Parmar, Katariya, and Patel 2019). Random Forests are also relatively fast to train and easy to interpret, making them suitable for iterative tuning, feature analysis, and implementation in the context of games.

We initialized a value for the window size (600, 300, 120, 60, 30) and step size (either the same as the window size (1) or half of that (½)). We chose these parameters to understand how data representation (overlapping or not, via step size) and temporal granularity (via window size) affect a model's ability to detect struggle. For every tested model, we tuned its hyperparameters automatically using GridSearchCV (scikit-learn 2025), exploring combinations of n estimators (80, 100), max depth (None, 10, 20), and min samples split (2, 5).

We used Area Under the receiver operating characteristic Curve (AUC) as the scoring metric to determine which parameter configuration yielded the best-performing model at each fold. We used AUC because it offers a more informative assessment of model performance than accuracy, particularly for detecting player struggle. Unlike accuracy, which measures performance at a fixed decision threshold, AUC evaluates how well the model ranks struggle over non-struggle (or cognitive vs. performative struggle) across all possible thresholds. This is especially useful in our con-

Window	Step	Gathering	Combat	Crafting
With Engineered Features				
600	½	0.59 (12)	0.59 (10)	0.77 (11)
300	½	0.60 (12)	0.48 (10)	0.79 (11)
120	½	0.64 (12)	0.51 (10)	0.78 (11)
60	½	0.63 (12)	0.47 (10)	0.77 (11)
30	½	0.63 (12)	0.48 (10)	0.78 (11)
600	1	0.64 (12)	0.55 (10)	0.76 (11)
300	1	0.62 (12)	0.57 (10)	0.78 (11)
120	1	0.60 (12)	0.53 (10)	0.72 (11)
60	1	0.64 (12)	0.52 (10)	0.77 (11)
30	1	0.64 (12)	0.47 (10)	0.77 (11)
Raw Data Only				
300	1	0.57 (12)	0.55 (10)	0.78 (11)

Table 2: Struggle vs. Non-Struggle AUC scores, split by quest and training parameters: window size, step size, and including or excluding engineered features. Brackets show number of folds used for cross-fold validation.

Window	Step	Gathering	Combat	Crafting
With Engineered Features				
600	½	0.16 (6)	0.66 (3)	0.73 (8)
300	½	0.42 (6)	0.64 (3)	0.63 (8)
120	½	0.34 (6)	0.70 (3)	0.45 (8)
60	½	0.31 (6)	0.71 (3)	0.54 (8)
30	½	0.34 (6)	0.71 (3)	0.51 (8)
600	1	0.73 (4)	0.79 (3)	0.86 (7)
300	1	0.11 (6)	0.60 (3)	0.71 (8)
120	1	0.32 (6)	0.67 (3)	0.68 (8)
60	1	0.38 (6)	0.64 (3)	0.53 (8)
30	1	0.28 (6)	0.71 (3)	0.57 (8)
Raw Data Only				
600	1	0.54 (4)	0.81 (3)	0.86 (7)

Table 3: Cognitive vs. Performative AUC scores, split by quest and training parameters: window size, step size, and including or excluding engineered features. Brackets show number of folds used for cross-fold validation.

text, where struggle annotations may be noisy, subjective, or vary in severity. While we balanced our dataset to ensure an equal number of windows within each variable, accuracy still fails to capture how confidently or consistently the model is identifying meaningful patterns. For example, a model might be correct but for the wrong reasons, or only be confident in easy cases. AUC, however, captures the trade-off between true and false positives, making it better suited for applications that require flexible or player-sensitive responses, such as triggering an in-game tutorial only when confidence is high. By prioritizing AUC, we align our evaluation with the needs of adaptive game systems, where both over-detecting and under-detecting struggle can negatively impact the player experience.

We then calculated the final model performance by averaging the AUC scores obtained from each fold. To assess the value of our engineered features, we additionally trained three more models (one per quest) with both engineered features left out, using the same window and step sizes as the best overall parameterization from when the features were included. We computed each parameterization's overall score by summing AUC scores of the three models that used it. This resulted in 33 models for classifying struggle versus non struggle, and another 33 models for classifying cognitive and performative struggles.

6 Results and Analysis

We evaluated two main classification tasks in our analysis: struggle detection, which involved identifying whether a player was struggling (see Table 2), and struggle type detection, which focused on distinguishing between cognitive and performative struggles, but only when a struggle was already present (see Table 3). Our models showed notably different performance between the two classification tasks. Struggle detection was relatively stable across window sizes and quests, with AUC scores generally falling in the mid-0.6 to 0.7 range. In contrast, struggle type classification was far more variable, with some models reaching AUC scores as high as 0.86, while others dropped as low as 0.11. This inconsistency likely stems from two key factors: first, the type classification task used a smaller subset of data, since it only included windows already labeled as struggles; and second, the behavioural differences between cognitive and performative struggles are often more subtle and could be difficult for the model to distinguish consistently.

The Influence of Window Size and Fold Count. The number of folds used in cross-validation varied across tasks and quests, partly due to how different window sizes affected the availability of usable data (Section 5.3). As window size increases, fewer windows are generated, and an entire struggle period can end up "sandwiched" within a window between periods of non-struggle, causing the window to be labelled as non-struggle despite containing struggle data. This reduces the number of windows labelled as struggles and, in turn, limits the number of folds that can be used for training and evaluation. This reduction in fold count had noticeable effects on model performance. In the Gathering Quest's struggle type classification task, for example, the AUC score rose to 0.73 (from 0.42 as the next highest score) when the number of folds dropped to 4 (Table 3). One possible explanation is that the omitted players – including the only advanced Terraria player in our dataset – introduced gameplay patterns that were harder to infer based on the data of less-advanced players. When their data was excluded because they lacked sufficient examples of struggle (Section 5.3), the model was able to more easily learn patterns from the remaining, more consistent player data. This occurrence emphasizes the need to not only ensure demographic diversity when building models, but also consider how player characteristics (e.g., skill level) affect model performance.

We also observed that the variance in AUC scores across folds could be quite high. As a result, interpretations based

solely on cross-fold AUC values may overlook the instability introduced by small sample sizes or outlying player behaviours. Collecting more data from each player could help, as more periods of struggle could be captured.

Larger window sizes also caused data sparsity, especially for detecting cognitive vs. performative struggles, where fewer labelled windows were available. This sparsity can reduce the reliability of cross-fold AUC scores, meaning that scores from larger window sizes should be considered with increasing caution. Going forward, it would help to develop a way to estimate confidence intervals around cross-fold AUC scores. Carefully analyzing outlier contributions may also improve the robustness and interpretability of gameplay-based prediction models.

Raw Data vs. Feature-Engineering. To estimate the effect of including our two engineered features (Section 5.2) in the models, we removed them from the data and trained three new models (one per quest) for each target variable. We chose the window and step size by summing the AUC scores across all quests (*with* engineered features included) and picking the combination with the highest total. Overall, the Gathering and Combat quest models trained on feature-engineered data outperformed those trained on raw telemetry for struggle detection, while the Crafting quest model remained the same. This difference suggests that features such as Time Elapsed and Progression may have helped highlight the buildup to struggle events by providing clearer, higher-level behavioural signals. For struggle type detection, the value of the engineered features varied for each quest. In the Gathering quest, feature-engineered data worked substantially better (0.73 vs. 0.54), which demonstrates the importance of the added features. In the Combat quest, performance was very close (0.79 vs 0.81), possibly because the detailed action sequence already captured signs of struggle related to combat. For the Crafting quest, both types of data yielded the same score (0.86). Overall, it appears that the usefulness of the engineered features depends on the specific mechanics and challenges in each quest.

The Impact of Overlapping Windows. Overlapping windows (step size = ½) were not consistently better than non-overlapping ones (step size = 1). In struggle detection, some configurations with overlap slightly outperformed their non-overlapping counterparts, particularly in the Gathering Quest (e.g., 120:½ = 0.64 vs. 120:1 = 0.60). However, in the Crafting Quest, overlapping and non-overlapping windows produced almost identical results. Overall, this suggests that overlapping windows offer only marginal benefits at best, and their effectiveness may depend more on the nature of the quest than on the windowing strategy itself.

In the struggle type classification task, overlapping windows generally led to lower performance, particularly in the Gathering Quest, where the lowest AUC scores came from step sizes of ½. This might be due to data leakage, as overlapping windows often share similar sequences, which could potentially cause the model to memorize rather than generalize. They also introduce many near-duplicate inputs, which can amplify the model's weaknesses at classifying labels for non-overlapping data.

Quest-specific Differences. The models trained for the Crafting Quest consistently yielded the strongest AUC values in both tasks, with struggle detection scores reaching up to 0.79 and struggle type classification peaking at 0.86. This suggests that behavioural signals related to crafting mechanics might be more structured and easier for models to learn from. The stability of the struggle detection scores across window sizes (Table 2) also indicates that player struggles in the Crafting Quest were captured with similar reliability, regardless of temporal resolution. In contrast, the models trained for the Combat Quest performed the worst in the struggle detection task, never exceeding an AUC of 0.59. However, the Combat Quest models performed reasonably well in struggle type classification, with a maximum AUC of 0.81 (Raw Data) and several other scores above 0.70. This suggests that while struggle events may be hard to detect in this quest, the distinction between struggle types (once detected) is relatively clear. Models trained for the Gathering Quest fell in the middle for struggle detection (up to 0.64), but performed poorly in the struggle type task, with all but one of the AUC values falling between 0.11 and 0.42. As AUC scores below 0.5 indicate performance that is worse than random guessing, the stand-out, 4-fold result of 0.73 for this quest (600:1) highlights the strong effect that including too much divergent data (e.g., from players with different demographics) can have on classification results.

7 Discussion

This work demonstrates the feasibility of using gameplay telemetry to detect player struggle in real time, offering an adaptable approach for games without requiring extensive changes to core systems. Using Random Forest classification and a sliding window technique, we have demonstrated that temporal patterns in gameplay data can be leveraged to detect both the presence and type of player struggle, particularly in quests focusing on crafting mechanics. We proposed an efficient way for players to annotate their gameplay struggles; this method could be useful for future studies that explore affective or behavioural modelling in games. We also offered insight into how self-reports can be used effectively in training machine learning models, despite challenges around construct validity and annotation consistency.

Using our data gathering platform, we created a structured, labelled dataset linking player gameplay logs to self-reported experiences of struggle using the CORGIS framework. This dataset captures nuanced differences between cognitive and performative difficulties in the context of real gameplay scenarios, and we share it to support future work[1]. To offer further context in understanding our models, we also share additional performance data[2].

Our methods for collecting data, training models, and analyzing the results are designed with flexibility in mind, allowing others to adapt this approach to different game genres, mechanics, and player populations. By experimenting with parameters such as window size, step size, and feature engineering, future researchers and developers can build

[1] See: https://tinyurl.com/terrariadataset.

[2] See: https://tinyurl.com/terrariamodelperformance.

models that suit their own use cases. Our work moves toward scalable player support tools that can detect when, how, and why a player is struggling, contributing to more accessible and responsive game experiences.

7.1 Limitations

While this research offers promising results in identifying player struggle using gameplay logs, several limitations should be acknowledged.

First, our participant pool was limited to 12 individuals, which is not sufficient to ensure the reliability or generalizability of the models' performance. While the results show promising trends – particularly in detecting and classifying struggle across different gameplay contexts – the small sample limits our ability to draw firm conclusions. Our sample also lacked gender diversity, as 10 of the 12 participants were women. This imbalance may have influenced the types of gameplay behaviours and struggles observed, further limiting our ability to generalize across broader populations.

Due to the constraints of our data collection timeline, the dataset contained fewer instances of player struggle than planned. This limited sample restricts the strength of our conclusions and makes it difficult to assess whether additional data would have improved model performance. This work should therefore be considered an early-stage feasibility study. We offer a proof of concept that struggle can be detected in real time, but our results should be considered preliminary. To test generalization, we used leave-one-player-out cross-validation, though the small sample made it difficult to run meaningful statistical comparisons. Future studies with larger and more diverse participants are needed to confirm and extend these findings.

Another important limitation lies in the design of the Combat Quest. While it was intentionally challenging, many players experienced repeated deaths, which led to fatigue and frustration. This may have reduced the accuracy of self-reported struggle annotations, as players became less motivated or too overwhelmed to record struggles in the moment. Complicating this further, *Terraria* enforces a 10-second respawn countdown after death, which created a pause in gameplay that players often used to reflect and mark their (prior) experience of struggle. These annotations were sometimes disconnected from the actual moments of struggle, leading to potential misalignment between the gameplay data and the labels used for training.

We relied exclusively on Random Forest classifiers for all detection tasks. While this model performed reasonably well across most configurations, it may not fully capture the temporal dependencies inherent in gameplay data. Random Forests are well-suited for structured data and provide insights into feature importance, but they treat each input instance independently. This may limit their effectiveness in modelling struggles that unfold over time. Since this is still a relatively new research area, our aim was not to benchmark against alternative models, but rather to explore whether annotated gameplay telemetry can be used at all to train models that detect player struggles.

Finally, while our machine learning approach showed potential within the context of the three custom-designed quests in *Terraria*, its generalizability to other games or digital experiences may be limited without adaptation. Different games introduce unique mechanics, pacing, feedback systems, and player expectations, which can affect how struggle manifests and is captured in telemetry data. However, the goal of this work is not to create a one-size-fits-all model, but rather to provide a flexible foundation that others can build upon. With access to similar player data and domain-specific knowledge, researchers and designers could adapt our approach by tweaking features or model parameters to suit their own game environments and target player behaviours.

8 Future Work

While this research focused on binary classification of player struggle (struggle vs. non-struggle) and the type of struggle (cognitive vs. performative), we designed our data collection process to also support exploring multi-dimensional struggle classification. Specifically, participants also annotated the severity of their struggle (e.g., mild, moderate, or severe). Future work could extend our framework to support this expanded classification task. Doing so has the potential to enable more adaptive and context-sensitive feedback systems in games, not only detecting that a player is struggling, but also responding in ways that are tailored to the type and intent of that struggle.

It would also be interesting to compare our Random Forest approach against simple heuristics, alternative classifiers, or more sequence-aware models (e.g., LSTMs or Temporal Convolutional Networks) to better capture temporal dependencies in gameplay behaviour.

Our model could be implemented directly within *Terraria*'s core gameplay, to power dynamic hint systems tailored to the type of struggle a player is facing. For instance, cognitive struggles might trigger brief on-screen popups that clarify objectives or explain missing steps, while performative struggles could be addressed by optional side quests designed to reinforce specific mechanics through guided practice. One could evaluate the effectiveness of these interventions through not only gameplay outcomes, but also the lens of learning and retention, drawing on principles from instructional design. We see our approach as offering value beyond entertainment games, with potential applications in serious games or educational environments where personalized feedback can support user learning and engagement.

Acknowledgments

The authors acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC), Grant No. 2020-06502.

References

Ali, N. 2025. Towards Custom Quest Tutorials: Identifying and Addressing Players' Cognitive and Performative Struggles in Games. In *Proceedings of the 20th International Conference on the Foundations of Digital Games*, FDG '25. New York, NY, USA: Association for Computing Machinery. ISBN 9798400718564.

Ang, D.; and Mitchell, A. 2017. Comparing Effects of Dynamic Difficulty Adjustment Systems on Video Game Experience. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '17, 317–327. New York, NY, USA: Association for Computing Machinery. ISBN 9781450348980.

Ariyurek, S.; Betin-Can, A.; and Surer, E. 2021. Automated Video Game Testing Using Synthetic and Humanlike Agents. *IEEE Transactions on Games*, 13(1): 50–67.

Bopp, J. A.; Mekler, E. D.; and Opwis, K. 2016. Negative Emotion, Positive Experience? Emotionally Moving Moments in Digital Games. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, 2996–3006. New York, NY, USA: Association for Computing Machinery. ISBN 9781450333627.

Cao, Y.; Sweetser, P.; and Zhu, X. 2023. Exploration of Player Emotions, Behaviours, and Individual Differences across Game Difficulty Levels in a Turn-Based Strategy Game. In *Companion Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY Companion '23, 143–148. New York, NY, USA: Association for Computing Machinery. ISBN 9798400700293.

Chen, E. Y. C.; White, A.; and Sturtevant, N. R. 2023. Entropy as a measure of puzzle difficulty. In *Proceedings of the Nineteenth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, AIIDE '23. AAAI Press. ISBN 1-57735-883-X.

Csikszentmihalyi, M. 2014. *Play and Intrinsic Rewards*, 135–153. Dordrecht: Springer Netherlands. ISBN 978-94-017-9088-8.

Cuerdo, M.; Baskaran, D.; and Melcer, E. 2024. Exploring how Emotional Challenge and Affective Design in Games Relates to Player Reflection. In *Proceedings of the 19th International Conference on the Foundations of Digital Games*, FDG '24. New York, NY, USA: Association for Computing Machinery. ISBN 9798400709555.

Cuerdo, M. A. M.; Mahajan, A.; Mao, J.; and Melcer, E. F. 2023. Try Again?: A Macro-Level Taxonomy of the Challenge and Failure Process in Games. In *2023 IEEE Conference on Games (CoG)*, 1–8.

Denisova, A.; Cairns, P.; Guckelsberger, C.; and Zendle, D. 2020. Measuring perceived challenge in digital games: Development & validation of the challenge originating from recent gameplay interaction scale (CORGIS). *International Journal of Human-Computer Studies*, 137: 102383.

Foffano, F. 2023. When Games Become Inaccessible: A Constructive Grounded Theory on Stuckness in Videogames. In *Companion Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY Companion '23, 333–336. New York, NY, USA: Association for Computing Machinery. ISBN 9798400700293.

Frommel, J.; Fischbach, F.; Rogers, K.; and Weber, M. 2018. Emotion-based Dynamic Difficulty Adjustment Using Parameterized Difficulty and Self-Reports of Emotion. In *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '18, 163–171. New York, NY, USA: Association for Computing Machinery. ISBN 9781450356244.

Frommel, J.; Klarkowski, M.; and Mandryk, R. L. 2021. The Struggle is Spiel: On Failure and Success in Games. In *Proceedings of the 16th International Conference on the Foundations of Digital Games*, FDG '21. New York, NY, USA: Association for Computing Machinery. ISBN 9781450384223.

Hegedues, C.; Dias Constantino, J. P.; Dixen, L.; and Burelli, P. 2023. Investigating Perceived and Mechanical Challenge in Games Through Cognitive Activity. In *2023 IEEE Conference on Games (CoG)*, 1–4.

Hunicke, R. 2005. The case for dynamic difficulty adjustment in games. In *Proceedings of the 2005 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*, ACE '05, 429–433. New York, NY, USA: Association for Computing Machinery. ISBN 1595931104.

Kristensen, J. T.; Valdivia, A.; and Burelli, P. 2021. Statistical Modelling of Level Difficulty in Puzzle Games. In *2021 IEEE Conference on Games (CoG)*, 1–8. IEEE.

Lain. 2025. Open Broadcasting Software (OBS) Studio. https://obsproject.com/.

Linehan, C.; Bellord, G.; Kirman, B.; Morford, Z. H.; and Roche, B. 2014. Learning curves: analysing pace and challenge in four successful puzzle games. In *Proceedings of the First ACM SIGCHI Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '14, 181–190. New York, NY, USA: Association for Computing Machinery. ISBN 9781450330145.

Liu, X.; Slater, S.; Andres, J. M. A. L.; Swanson, L.; Scianna, J.; Gagnon, D.; and Baker, R. S. 2023. Struggling to Detect Struggle in Students Playing a Science Exploration Game. In *Companion Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY Companion '23, 83–88. New York, NY, USA: Association for Computing Machinery. ISBN 9798400700293.

Field Day Learning Games. 2023. Wake: Tales from the Aqualab. https://fielddaylab.wisc.edu/play/wake/.

scikit-learn. 2025. GridSearchCV. https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html.

Parmar, A.; Katariya, R.; and Patel, V. 2019. A Review on Random Forest: An Ensemble Classifier. In Hemanth, J.; Fernando, X.; Lafata, P.; and Baig, Z., eds., *International Conference on Intelligent Data Communication Technologies and Internet of Things (ICICI) 2018*, 758–763. Cham: Springer International Publishing. ISBN 978-3-030-03146-6.

Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; and Duchesnay, E. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830.

Re-Logic. 2011. Terraria.

Teng, Z.; Holmes, J.; Dominguez, F.; Pfau, J.; Junior, M. E.; and El-Nasr, M. S. 2025. Identifying Player Strategies Through Segmentation: An Interactive Process Visualization Approach. In Plass, J. L.; and Ochoa, X., eds., *Serious Games*, 77–90. Cham: Springer Nature Switzerland. ISBN 978-3-031-74138-8.

Warmuptill. 2016. Advanced Scene Switcher. https://obsproject.com/forum/resources/advanced-scene-switcher.395/.

Whitby, M. A.; Deterding, S.; and Iacovides, I. 2019. "One of the baddies all along": Moments that Challenge a Player's Perspective. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '19, 339–350. New York, NY, USA: Association for Computing Machinery. ISBN 9781450366885.

Yu, K. K.; Sturtevant, N. R.; and Guzdial, M. 2021. Towards Disambiguating Quests as a Technical Term. In *Proceedings of the 16th International Conference on the Foundations of Digital Games*, FDG '21. New York, NY, USA: Association for Computing Machinery. ISBN 9781450384223.