Towards explainable prediction of player frustration in video games.

Max Wolterink
Utrecht University, Department of Information and Computing Sciences
Utrecht, The Netherlands

Sander Bakkes
Utrecht University, Department of Information and Computing Sciences
Utrecht, The Netherlands

ABSTRACT

Frustration is a key concept in retaining a player interest in both commercial and applied games. In a HCI context, frustration is often seen as a purely negative phenomenon. However, for games to be interesting some amount of frustrating has to be present. As such, dynamically adjusting game elements to ensure optimal frustration levels can be a valuable way to increase player retention. A first step towards such a system is an accurate classifier of frustration. To date, most attempts at frustration classification use models that are relatively hard for a human to understand. In this paper an attempt will be made at creating an explainable predictor of player frustration. To accomplish this, the frustration-aggression theory was used to identify a number of key components that determine the severity of a frustrated response. 135 participants were asked to play a series of Pac-Man levels while being asked about the frustration components. Gameplay features, participant behaviour and participant responses were gathered and used as a dataset to train a number of random forest classifiers. The classifiers were trained to predict player frustration, with accuracy ranging from 66.3% to 83.1% depending on the amount of frustration classes used. Accuracy dropped significantly when excluding participant responses on frustration component questions from the dataset. Furthermore, feature importance analysis revealed the overwhelming importance of the Repeated Failures component, as well as the relatively low importance of all in-game variables. These results suggest that the currently used variable set might not accurately represent the components of frustration. A possible avenue for future research could be the discovery of accurate metrics for these internal component perceptions.

CCS CONCEPTS

• **Human-centered computing** → **User centered design**; **User models**; • **Computing methodologies** → *Classification and regression trees.*

KEYWORDS

Player modelling, Frustration, User Experience, Random Forest Classifier

FDG'21, August 3–6, 2021, Montreal, QC, Canada

ACM ISBN 978-1-4503-8422-3/21/08…$15.00
https://doi.org/10.1145/3472538.3472566

ACM Reference Format:
Max Wolterink and Sander Bakkes. 2021. Towards explainable prediction of player frustration in video games.. In *The 16th International Conference on the Foundations of Digital Games (FDG) 2021 (FDG'21), August 3–6, 2021, Montreal, QC, Canada.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3472538.3472566

1 INTRODUCTION

Digital games are one of the largest modern entertainment formats [24]. An estimated 42% of adults play video games on a weekly basis [9], with those numbers continuing to rise in recent years. Furthermore, game industry revenue continues to reach new heights, reaching a record of 152.1 billion dollars in global revenue in 2019[20]. Games sell because they appeal in some way to their players. To achieve economic success it is often a good idea to attract a large variety of individuals as to maximize the potential buyer base. However, video games are currently designed with a model of an intended player group in mind which corresponds to the designer's idea of how a player will interact, respond and adapt to their design. It is inherently difficult to design for such a large number of players due to the individual variance present in this group. In an attempt to combat this, some games have started to implement dynamic systems that respond to player behaviour to provide a more tailored user experience. To achieve this, player modelling is used to create computational representations of players that are generated based on player input. These models are then used to predict various internal states that influence the user experience, such as perceived fun, challenge and frustration.

Of these factors, frustration is one of the most interesting. In psychological and human-computer interaction theory, frustration is often seen as a purely negative phenomenon. However, in video games some amount of frustration might be necessary. A game without obstacles that generate frustration tends to be boring and low difficulty can be as much of a reason for disinterest as one that is too high. An explanation for this might be found in Flow theory, specifically the GameFlow model [25]. This model introduces the state of Flow [11], which is an enjoyable experience that can arise when certain conditions are met. The most important of those conditions being a delicate balance between challenge and personal skill level [15].

It might thus be useful to model a player's frustration level if the goal is to dynamically alter a game to influence this frustration level for optimal enjoyment. In this paper an attempt will be made at modelling player frustration in the game of Pac-Man. Section 2 will cover previous research, known issues of frustration, Flow theory and player modelling. This section will end with a description of the contribution of this paper. Section 3 will describe the testbed game, participants, procedure and model choice. Sections 4 will

describe the results of statistical testing and section 5 will analyze and discuss the findings.

2 BACKGROUND AND CONTEXT

2.1 Frustration

To be able to make any model of frustration a definition of frustration has to be given first. There does not seem to be any overarching definition in psychological literature that is used in all contexts. Instead, a distinction is made between external conditions ('a' frustration) [7] and an emotion that exists within an organism in reaction to outside events [2].

The Frustration-Aggression hypothesis[13] provides a useful definition of frustration in an human-computer interaction context. Dollard and his colleagues defined a frustration as *'an interference with the occurrence of an instigated goal-response at its proper time in the behavior sequence'*[13]. In this theory, a frustration is always a block to a certain goal. It is useful to explain the individual parts of this definition in more detail to get a full understanding of what the authors meant. For something to be classified as a frustration, two aspects have to be present. First, an individual needs to have a goal. Without a goal, nothing can be impeded and thus no frustration can be present[7]. Secondly, the individual needs to make anticipatory goal-enjoying responses[7]. What is meant by this is that it is not enough for an individual to just have a goal, but some type of satisfaction derived from attaining that goal needs to be anticipated. This satisfaction might be extrinsic or intrinsic in nature and can be anything from autotelic pleasure to monetary rewards[7]. In this, Dollard and his colleagues differ from his predecessors. He does not consider just deprivation of a reward as a frustration. It becomes a frustration only when this deprivation is of an anticipated reward due to thwarting of a goal for which the individual was genuinely motivated.

According to the original Frustration-Aggression hypothesis, aggression follows frustration, hence the name. There are several factors in this process that influence the strength of this aggressive response. Berkowitz[13] identified three main aspects. First, the extent to which an individual expected satisfaction from the attainment of the goal. The stronger the anticipated the reward, the stronger the aggressive response. Secondly, the extent to which the goal was blocked, with limited goal-reaching reducing the aggressive response. Finally, the number of in-sequence frustrated responses. Here, frustrating events will become more frustrating the more they are repeated within a given time frame. Since frustration generates aggression, it makes sense that the factors influencing the extent of the aggressive response are all facets determining the 'success' and satisfaction of goal attainment. Since these factors determine the presence and severity of the frustrations.

Later research revised this hypothesis to accommodate new research. While the general idea of the hypothesis was validated ([7] [19], for an overview of empirical evidence), it was found that frustrations only give rise to aggressive responses to the degree that they generate negative affect [5][6]. In this reformulation, frustrations are seen as aversive events that produce negative affect, with the extent of the negative affect determining the appropriate level of aggressive response. Negative affect is defined as *"any feeling that people typically seek to lessen or eliminate"* [7]. This is typically seen as the feeling of frustration or anger. Thus, the previously mentioned factors that determine the height of the aggressive response do so because of their negative affect generating traits. Furthermore, this reformulation attempts to incorporate thought processes into the factors determining the strength of the response. By doing this, Berkowitz[7] added three factors to the mediating aspects formulated by Dollard[13]. Firstly, the legitimacy of the block, where an obstacle that is perceived as being there for legitimate reasons causing less aggression than one that is seen as illegitimate. As an example, one could look at a failing teammate dragging the performance of the entire team down. If this teammate is failing for a perceived legitimate reason, such as a sudden illness, it would inhibit the frustrated response in comparison to a perceived illegitimate reason, such as laziness. Secondly, there is the deliberation factor, where an act that is seen as deliberately blocking someone's progress towards a goal causes more aggression than a block that is perceived as accidental. One gets less frustrated having to drive slowly and perhaps not reaching your destination in time if it is raining and the road is slippery compared to someone purposefully blocking the road just to spite you. Lastly, the anticipation of failure mediates the strength of the response, with unanticipated failures contributing significantly more to the aggressive response than anticipated ones. Researchers found that this sudden failure has a resemblance to punishment [14], which generates a larger amount of negative affect and thus corresponds to a larger aggressive response.

The reformulated frustration-aggression thus gives us 6 components that determine the extent of negative affect generation, which in turn determines the measure of the aggressive response. These 6 components are respectively; Personal Satisfaction, (partial) Goal Completion, Repeated Failures, Block Legitimacy, Block Deliberateness and Block Anticipatability. These 6 components can thus be used as a guideline to evaluate the frustration a feature will cause.

So far frustration has been described as a purely harmful phenomenon that causes negative emotions and many problems. In a human-computer interaction context this is often the dominant viewpoint. The goal of most devices is to complete tasks, and anything getting in the way of efficient task completion should therefore be reduced as much as possible. However, for games this might not be the case. Most games contain a sequence of obstacles that block a player's progression and it can be argued that the whole point of most games is overcoming these obstacles. Even games that most players fail in time and time again and most would describe parts of the experience as highly frustrating (such as the Dark Souls series, famed for its difficulty) are also enjoyed by many players. On the other hand, games lacking in difficulty and containing nearly no obstacles are often seen as 'boring' if there are no other aspects present to keep a player entertained. Clearly, there is a balancing act of frustration, skill and challenge that is needed to keep a player engaged.

2.2 Flow Theory

An often used explanation for this balancing act is found in Flow Theory [11]. Flow Theory states that there is an experiential state (called 'Flow') in which a person is completely absorbed by the

activity one is partaking in. In this state, actions 'flow' from one to the other without need for conscious intervention on the user's part. This flow experience is defined by five elements. The merging of action and awareness, the centering of attention, the loss of the sense of self, the feeling of control of action and environment and the demand for action. This phenomenon is highly enjoyable for most and is often observed in instances of play, but has also been reported surfacing during work-, creative- or religious activities [11].

However, a state of flow is not always experienced whenever anyone engages in these activities. According to Csikszentmihalyi[11] flow emerges when a player is able to focus on a game and the game presents obstacles that stay challenging enough throughout the learning process. Indeed there is some evidence that this seems to be the case [16]. This suggests that the ratio of challenging obstacles and skill might be the key to providing an enjoyable experience.

2.3 Player Modelling

It is useful to model a player's internal frustration state to be able to maintain an optimal level of frustration for a given player as to facilitate a flow experience. This can be done by player modelling, which aims to detect and predict cognitive, affective and behavioural patterns by using a computational model[28]. This is usually done through some form of analysis of the player's behaviour in a given game. In this regard it is different from player profiling, which attempts to place a player into a typology based on static information[28]. In contrast, player modelling is based on dynamic information acquired during gameplay. This definition is not universal though as other researchers state that player modelling refers to modelling external player behaviour while player profiling attempts to model internal player states[26]. For the purposes of this paper, Yannakakis'[28] definition will be referred to whenever player modelling is mentioned.

Some attempts at frustration modelling have already been made. Pedersen, Togelius and Yannakakis [21] attempted to train a neural network that used level design parameters and player behaviour to predict three reported emotions, including frustration. They found that even a relatively simple single-neuron network resulted in a high accuracy prediction of frustration labels (88,66%). However, other emotions were harder to predict.

Yu and Trawick [29] expanded on this and tried to build models that attempted to predict frustration or boredom labels from a set of in-game aspects and the same characteristics of player behaviour as Pedersen et al's[21] research. They categorized players based on input from a questionnaire and in-game behaviours using a soft clustering algorithm. Two models were then built per category, one of player frustration and one of player boredom. These models were based on player feedback and gameplay features, which includes both player behaviour as well as controllable features. A ranking algorithm was used to predict the final label given to a specific level. Their frustration prediction accuracy was 87.5%.

2.4 Model Evaluation

Player modelling needs some way of checking the goodness of fit of their model. Usually this is done by comparing model predictions to some form of real world data. Most player models attempt to model an internal state, such as frustration or enjoyment, which is often hard to directly measure. This means that alternative measures need to be used as an indication of the presence of the internal state. Three types of data are commonly used to infer this presence. The most frequently used measures of internal states are physical indicators, self-report and behavioural cues.

Physical measures are things like skin conductivity [27], mouse pressure [18] and facial expressions [4]. These indicators can be used to infer certain internal states [17][10][12], although the extent to which these signals truly indicate a certain emotional state is debated. For example, skin conductivity will change with the amount of perspiration or the tightening of muscles. Both of these factors will be present when playing a high intensity game, which will lead to a prediction of higher arousal. However, the higher skin conductivity might be attributed solely to these physical aspects instead of the internal emotional state.

In contrast, self-reports are generally valid in measuring the desired internal state. However, they might be inconsistent and inaccurate [29]. Common problems include inaccurate ranking (e.g: a player could rank the frustration of two very similar levels as differently frustrating depending on what he experiences directly before. A phenomenon known as anchoring) and inconsistent ranking (e.g: a frustration rating of 3 might mean something different to player A than it does to player B). These deficiencies can be reduced through data manipulation, such as the use of a ranking algorithm in Yu and Trawick[29].

Finally, behavioural cues can often be used to predict player characteristics. Shaker, Yannakakis & Togelius [23] used in-game behaviour such as his movement behaviour or jumping frequency to change the parameters of automatically generated levels to optimize fun, challenge and frustration. A large problem with behavioural cues is that, much like physical measures, these cues often indicate the presence of a certain internal state but are no direct measures of this state. Therefore you need a good theory on what type of behaviours indicate which state. Alternatively, as Shaker, Yannakakis & Togelius[23] have done, you can see which statistical correlations exist between the data and self-reported internal states.

2.5 Contributions of this paper

Some progress towards an accurate predictor of player frustration has been made. However, most attempts to date have used model-free approaches with a fairly low explainability. This limits the usefulness of such a tool, as game designers would want to know which variables to tweak to influence the amount of experienced frustration. Additionally, most approaches use no assumptions on the origin of frustration in their prediction. Using pre-existing theories about frustration might narrow the search for influential variables. These factors are so far relatively unexplored.

Thus, in this paper an attempt will be made at creating an explainable predictor of player frustration using a pre-existing theory of frustration. Data will be collected in the game Pac-Man. Data collected from the game consisted of 3 categories. Static gameplay features (ghost speed, lives, etc), participant behaviour (number of pellets collected, number of ghosts eaten, etc) and participant responses on a questionnaire with a number of questions related

to frustration and the 6 components as defined in the frustration-aggression hypothesis. By combining self-report and behavioural measures, higher performance might be achieved. Physical measures were neglected due to the perceived difficulty at obtaining them in real-world scenarios and due to the ambiguity of their results. Variants of the resulting dataset were used as input for a number of random forest classifier models. The random forest learning method was used because of its relative explainability and the ability to easily generate variable importances, which might give an insight into the causes of frustration. In the final step, these models were tested and evaluated as predictors of frustration.

3 METHODS

3.1 Overview

Randomly selected participants played 5 levels of a custom built, modified Pac-Man game online. Various variables relating to gameplay features, in-game participant behaviour and participant response were collected during the experiment to serve as a foundation for the frustration classifier. All these factors will be illustrated in greater detail below.

3.2 Testbed Game

Pac-Man was chosen as the testbed game to collect data from. Gameplay in Pac-Man consists of the player moving Pac-Man in a tile-based 2d maze, trying to collect all the pellets in the level. Pac-Man can only move up, down, left or right. The player is victorious if Pac-Man collects all pellets in a level. There are 4 ghosts in the maze that serve as the player's obstacles. If Pac-Man touches a ghost, the player loses a life and resets the position of both Pac-Man and all the ghosts, but not the collected pellets. Upon losing 3 lives, the player is defeated and the level ends.

At the start of a level only the red ghost (Blinky) is active in the maze, with the rest of the ghosts trapped in the ghost house. However, the other 3 ghosts are released during a level at set time intervals. The ghosts all have slightly different behavioural patterns, determined by their target tile. For a full overview of ghost pathfinding mechanics see the Pac-Man Dossier [22]

Pac-Man was chosen as a testbed because of its relative familiarity to most participants, and its simplicity in both rules and control. This made it more likely that every participant was able to play and understand the game, reducing the influence of at-game frustration in the measurements, which was not being measured. This kept the focus on the in-game frustrations that served as data for the model. Furthermore, Pac-Man was relatively easy to recreate, allowing total control over every aspect of the experiment.

3.3 Participants

Since the experiment did not focus on any particular population group, participants were recruited using random selection from various websites, with the bulk of the participants coming from the Utrecht University paid participants Facebook page. As such, the participant population skewed towards college students that participated for student credit.

168 Participants were randomly selected in this way. After filtering out those that only partially completed the experiment 135 participants were left with a mean age of 23.7. Of these participants, 59 were male (mean age: 25.5), and 76 were female (mean age: 22.4). Zero individuals chose 'other' or preferred not to state their gender.

Finally, participants were asked how many hours they spent playing video games in an average week. 54 participants reported almost never playing videogames, 44 reported between 1-5 hours of use per week, 8 reported between 6-10 hours of use per week, 11 reported between 11-15 hours of use per week, 7 reported between 16-20 hours of use and 11 reported a usage of 20+ hours every week.

3.4 Materials

The Pac-Man implementation was a custom-build version of Pac-Man developed by the researcher in Unity 2019.3 specfically for this research. The playable build was a webGL build deployed to a private server and thus playable in the browser. The Pac-Man build was identical to the traditional Pac-Man game, with minor differences. The 'ghost house' mechanic where ghosts go after they are eaten to reform was removed, replaced with a mechanic where the ghosts would flee to the other side of the map. This was done to ensure a consistent time where ghosts were frightened. Some gameplay elements were randomized to research the effect of these gameplay features on frustration levels. The aspects that varied between levels consisted of: Ghost Speed (Range: 50%-110% of Pac-Man's speed), Powerpellet number (Ranging between 0 and 8), Powerpellet location (8 different locations), Powerpellet Frightened time (between 4-10 seconds), Scattertime (Between 4-12 seconds) and Chasetime (Between 10-30 seconds). Although not originally planned as an online experiment, the experiment want administered online to conform to the Covid-19 pandemic quarantine regulations present on the university at time of writing. As such, devices on which participants played varied, although all were regular computers as the experiment did not function on mobile devices.

3.5 Procedure

After recruitment, participants followed a link to the website on which the experiment was playable. The participants were greeted with a main menu screen were they could either start the experiment, or quit. After starting, participants were briefly informed of the procedure and purpose of the research, after which they could sign a consent form. The participants were then shown a brief instruction on the objective and controls of the Pac-Man game, to make sure every participant knew how to play. They then played 5 levels of the modified Pac-Man game.

While playing, a survey would pop up at set time intervals. This survey would pause the game and ask the participant 6 questions. The participant would be asked about his overall experienced frustration level in the last 30 second gameplay segment. Furthermore, the participant was asked about the absence or presence of 5 (of the 6) components of frustration. It was judged that Personal Satisfaction was unable to be measured in a simple questionnaire. It was thus dropped from the components of frustration and it was assumed that participants had similar amounts of motivation.

Experiences of frustration components as well as total amounts of frustration were gathered through simple 1-question questionnaires per component. These questions directly asked about the presence or absence of a certain component in the last gameplay segment. For example, the Legitimacy component question was;

'Did the last gameplay segment feel Legitimate?'. For the components, a clarification was given below the given on the meaning of these components terms. For example, the clarification for the Legitimacy question was; *'Clarification: Legitimate gameplay feels reasonable, proper and fair, illegimate gameplay does not.'* These 1-question questionnaires were chosen over validated question batteries due to the volume of labels needed for the random forest models. Ideally, we would have a frustration label for every second of gameplay data. Realistically however, this is not viable since a player getting interrupted every second leads to an unplayable game and participants' frustration will not fluctuate from moment to moment. Thus a 30 second interval between questionnaires and a simple questionnaire was chosen as a tradeoff between the optimal amount of labels needed for the model to function and internal validity.

Participants could answer the questions by clicking one of five buttons that formed a Likert scale. The program would then stay paused for 3 more seconds to allow the player time to prepare before resuming play. These surveys popped up every 30 seconds and at the end of each level. 5 Levels were played this way by each participant. After these 5 levels, a demographics survey would be administered, as well as an opportunity to report bugs and glitches. Finally, the participants were thanked and an email-address was presented in case the participants had any more unanswered questions. Screenshots of every screen the participant encountered in order can be found online at [3].

3.6 Data Collection

A mixture of behavioural measures and self-report was chosen as indicators of a frustrated state. Physical measures were not used due to the ambiguity of their results. Furthermore, behavioural measures and self-report are data that a game designer has easier access to, increasing real world viability of the system. 41 variables were measured during play, dividable into 2 categories; gameplay variables and participant response data. Gameplay variables were measured every second and involved either controllable gameplay features (such as ghost speed) or in-game participant behaviour (such as number of pellets collected). These variables were chosen due to their perceived effect on the 5 components of frustration based on expert opinion. The gameplay variables can be classified into 6 categories: Time, Location, Progress, Deaths, GhostBehaviour and Miscellaneous.

Time: The time of player death; The time it took to complete a level; The current time in the round; The percentage of time spent standing still.

Location: Pac-man's location (in X and Y coordinates); Blinky's Location, Inky's Location, Pinky's Location and Clyde's Location; The location of deaths; The distance to all ghosts; the distance to the nearest ghost.

Progress: The current level; The number of pellets collected and powerpellets collected; the number of ghosts eaten.

Deaths: The number of deaths; The number of kills of each respective ghost; The number of lives remaining.

GhostBehaviour: The time ghosts are in scattermode; The number of ghosts in chase behaviour; The time ghosts are in chasemode; The time ghosts are in frightened mode; The speed of all ghosts.

Miscellaneous: The number of teleporters used; the number of functionally useless inputs.

Furthermore, the participant's responses on the questionnaire were added to the data after the 30 second mark. This concerned 6 variables related to frustration or frustration components, namely; LegitimacyScore, DeliberatenessScore, AnticipationScore, RepeatedFailures, (partial) Goal Completion and TotalFrustration. Appending these scores to the previous data created data blocks of 30 seconds of gameplay data followed by participant labels. These data blocks would serve as the input for the random forest model.

3.7 The Random Forest Classifier

To predict frustration, a function has to be approximated between gameplay features, participant behaviour and participant response. A random forest classifier was judged to be a good fit for this problem. A random forest classifier is a method of ensemble-learning where a number of decision trees are constructed based on a dataset that all vote on an outcome. The outcomes in this case being the 5 frustration classes. In this way it can classify frustration from a number of variables. For a full overview of the workings of a random forest, see (Breiman, 2001).

There were a number of reasons for this choice. Firstly, a random forest has relatively robust ways of calculating feature importance. This could give some insight into not only when a participant is frustrated but also why. This increases the explainability of the model and makes it more useful as input for game designers. Secondly, it allows for a large array of input variables, making it scalable to more complicated games. Finally, the mapping between gameplay variables and frustration is probably non-linear, which a random forest can handle.

3.8 Data Processing

After data was collected, some processing was required to make the data suitable for use with the random forest model. For most variables, averages were calculated for a given gameplay interval. This was done to variables for which an average was more representative for that given gameplay interval such as GhostDistance or NumberOfGhostsOnChaseBehaviour (e.g: an average distance to all ghosts in the past 30 seconds is a better indication how close the ghosts were in in comparison to just the last value).

Other variables (Deaths, Kills etc) were added to a total for a given gameplay interval. This was done so a block of data accurately reflects what happened in a certain gameplay interval. Due to a difference between the locally tested Pac-Man build and the deployed online build, the Useless_Inputs variable was not properly tracked with the deployed online build of the experiment and reported only zeroes. This variable was therefore removed from the dataset.

The dataset was then divided into a training and testing set, with a 80/20 train/test split percentage. Because the classes that are aimed to be predicted were imbalanced, a stratification process was used to make sure both the training and test datasets contain examples of all the classes in the same proportion as the original dataset.

Table 1: Overview of participant responses

Component	Mean	Standard Deviation
Frustration	2.22	1.27
Legitimacy	3.62	1.16
Deliberateness	2.49	1.28
Repeated Failures	2.10	1.19
Anticipation	3.38	1.22
Goal Completion	3.50	1.26

Table 2: Frequency of frustration class responses

Class	Frequency
1	568
2	355
3	253
4	177
5	98

4 RESULTS

4.1 Participant Reponse Overview

An overview of participant responses on the questionnaire questions can be found in table 1. On average we see means between 2.22 and 3.62. The standard deviation of all components is similar, ranging from 1.16 to 1.28. Furthermore, there was a significant effect of amount of hours spent playing video games as determined by a 1-way ANOVA ($F(5) = 8.011$, $p < 0.001$). Participants that played more reported less frustration than those that barely played video games, although the effect was very small.

4.2 Frequency of Frustration Classes

As previously mentioned, participants could respond to any frustration of frustration component question with an answer between 1 and 5. With regards to responses in the frustration category, an unbalance between frustration classes can be seen in figure 3, with a clear trend downwards. Only 91 instances of maximum frustration (class 5) out of 1451 total observations were seen, compared to 568 observations of class 1. To illustrate this difference, class 1 accounted for over 40% of all observations while class 5 only accounted for 6.3%.

4.3 The Random Forest Classifier

A number of random forest models were tried to predict user frustration, starting with a default Random Forest Classifier. To see how this model would perform, the default set of hyperparameters was selected as a first trial. This model was trained on two versions of the dataset. One version of the dataset contained participant responses on each of the five components of frustration, while they were absent in the other version. This was done to see if user perception of components captured something that gameplay variables could not. The models using component response data will be known as the Component Response Data Classifier (CRD Classifier), while those that do not use component response data will be known as noCRD Classifier models.

4.4 Random Forest Classifier

Table 3: Hyperparameters

Hyperparameter	Value
N-estimators	1600
criterion	Gini
max_depth	10
min_Samples_Split	12
max_features	18
max_samples	2

The random forest classifier was trained on the two datasets. For these two datasets, four metrics of performance were calculated. These metrics are Accuracy, Precision, Recall and the F1-score. Accuracy refers to the percentage of correct predictions. Precision is the number of correctly identified members of a class, divided by the total amount of class predictions. Recall refers to the number of members of a certain class that the model identified divided by total members. The F1 score is a measure of both of the previous metrics combined. Additionally, the support for each category was tracked. Finally, a confusion matrix was produced to provide further insight into where the model might make incorrect statements. Some overfitting was found thus randomized search cross validated was used to tune the hyperparameters. The used hyperparameters can be found in table 3.

4.5 CRD Classifier Performance

Generally, we see an accuracy with an average of 66.3% on the test set. Training set accuracy was reduced to 87.1%, which is closer to test set accuracy. When looking closer at the data in table 4, a split can be seen with classes in the middle of the spectrum. The model performs best on the first class, with decreasing performance seen on the higher classes. Recall seems to gradually decline on the higher classes while Precision dips on class 3 and then climbs up again. The model has the greatest difficulty recalling class 4, and has the greatest difficulty overall with class 3.

The decrease in performance manifests itself in two ways, as seen in figure 1 and table 3. Firstly, the model tends to make additional mistakes the further down the classes you go. Secondly, the magnitude of error (e.g: the distance between predicted class and true class) increases as well. The only exception is frustration class 5, which the model relatively frequently (55%) recalls correctly, but also relatively frequently predicts to be in class 1 (20% of the time),

Table 4: CRD classification report

Class	Precision	Recall	F1-Score	Support
1	0.78	0.89	0.83	114
2	0.59	0.62	0.60	71
3	0.48	0.47	0.48	51
4	0.67	0.34	0.45	35
5	0.65	0.55	0.59	20
Accuracy		66.3%		

Table 5: noCRD Classifier report

Class	Precision	Recall	F1-Score	Support
1	0.58	0.80	0.67	114
2	0.41	0.41	0.41	71
3	0.26	0.24	0.25	51
4	0.46	0.17	0.25	35
5	0.40	0.10	0.16	20
Accuracy		48.1%		

which is the largest error the model can make. Errors this large are rare while predicting other classes.

4.6 noCRD Classifier Performance

Performance without the component response data is significantly reduced compared to the CRD classifier, with an average accuracy of 48.1%. As seen in table 5 and figure 2, we see a similar reduction in Recall as with the previous model the further you get from class 1 and a similar dip in Precision around class 3. In the previous model this performance drop was mostly seen in the middle classes. However, without participant response, this drop carries over to class 5, which is seen especially in the Recall rate. Furthermore, the model seems to be biased to incorrectly predict observations to be in class 1.

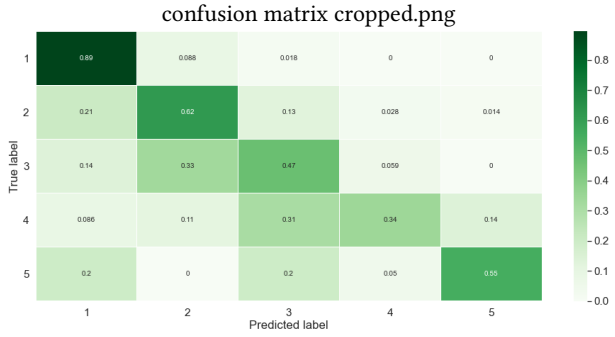

Figure 1: CRD Classifier confusion matrix trained on participant responses and gameplay variables

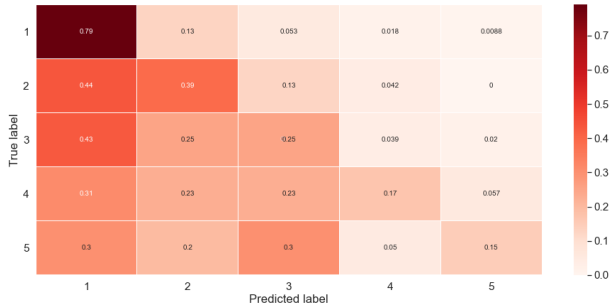

Figure 2: noCRD Classifier confusion matrix trained on just gameplay variables.

Table 6: Classifier accuracies with 3 frustration classes

Model	Accuracy
3-Cat AMS Classifier	81.4%
3-Cat AES Classifier	73.5%

4.7 Aggregated Categories

Since model performance varies so much between classes and observations of higher frustration classes are limited, two models were trained on a dataset with aggregated frustration categories. Participant responses of total frustration were aggregated into three categories, low, middle and high. This was done in two different ways. First, classes 2, 3 and 4 were aggregated into a single class, becoming the new 'middle' class (new class 2). Meanwhile, the classes on the edge (classes 1 and 5) became classes 'low' and 'high' (new class 1 and 3) respectively. This aggregated dataset will henceforth be known as the Aggregated Middle Set (AMS). Alternatively, a second dataset was created where the first and second class were aggregated, becoming the 'low' class. In this dataset the fourth and fifth class were aggregated to become the new 'high' class. This aggregated dataset will be known as the Aggregated Ends Set (AES). The classifier was retrained on this new dataset, with the results shown in table 6 and figures 3 and 4.

As can be expected, reducing the categories from five down to three significantly increases the performance of the classifier (Accuracies between 73.5% and 81.4% compared to 66.3%) with the AMS trained classifier leading to a better performance than the AES trained one. Note that the AES classifier not only performs worse than the AMS classifier, but also seems to have a bias towards category 1, which is the largest category by far. Even more so since the Aggregated Ends Set now combines the largest 2 classes of the previous dataset. This could artificially inflate its accuracy as always predicting category 1 would result in a high accuracy but questionable usefulness due to the skewedness of the classes.

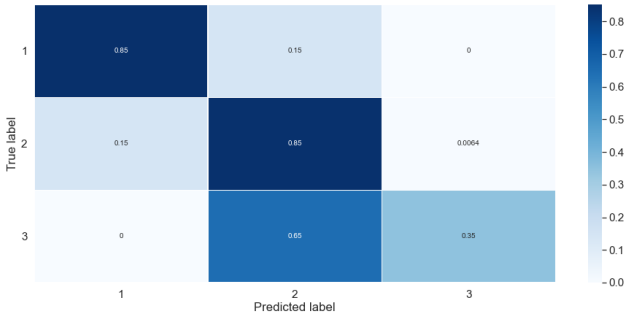

Figure 3: AMS Classifier confusion matrix trained on the aggregated middle dataset.

4.8 Frustration Component Prediction

Finally, a random forest classifier was built to try and predict the five components of frustration. In the same manner as the previous random forest models, 2 versions of dataset were used to train these

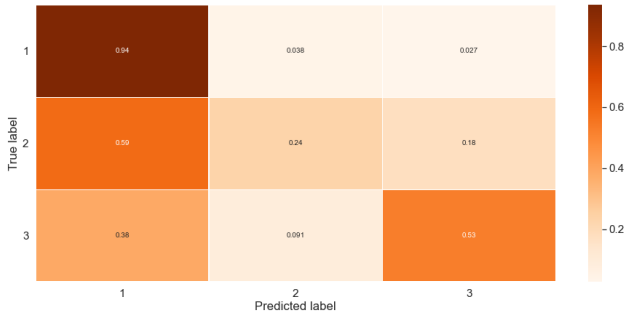

Figure 4: AES Classifier confusion matrix trained on the aggregated ends dataset.

models. One without component response data (noCRD) and one including component response data (CRD). These models attempted to predict 5 classes of the frustration components without class aggregation as seen in the AES and AMS Classifiers. This was only mildly succesful, with accuracies ranging from 50.2% on the low end to 60.1% on the high end of the spectrum with component data. Similar to the noCRD classifiers, performance is significantly worse without component data, ranging from 31.6% to 43.3%.

4.9 Feature Importance

To see which feature might best predict frustration, feature importance analysis was performed. The standard metric for feature importance is Mean Decrease in Impurity (MDI). Mean Decrease in Impurity is the average of a feature's total decrease in node impurity, weighted by the proportion of samples reaching that node in each individual decision tree in the random forest. For a full overview of the math behind this metric, see Breiman (2001). However, MDI is computed on the training set. As we have seen, the model tends to learn the entire training set, and perform worse on the test set. Even after parameter tuning some degree of overfitting is present. MDI therefore might give a biased view of variable importance.

To combat this, Permutation Importance (PI) per feature was used as a metric. A PI function randomly shuffles the values of a certain feature, therefore breaking the relationship between feature and target frustration label. This drops a model's accuracy score and thus indicates the importance of that variable for the model's performance. For a full overview of the math behind this function, see Breiman (2001).

Feature importance can be seen in figure 5. Four of the five components of frustration rank relatively high (top 10) in feature importance, with the most important one being the RepeatedFailures score. This feature is the score participants gave themselves on how often they failed in the last gameplay segment.

Interestingly enough, a number of features that determine difficulty such as GhostSpeed, FrightenedTime and Scattertime (PI only) are near the bottom of the importance list. One would think increasing the difficulty would be strongly linked to failure rate and thus increase the frequency of frustrating events. However, this data indicates otherwise.

5 DISCUSSION

5.1 Can frustration be predicted?

This research set out to find an explainable way to predict user frustration from game features, user behaviour and participant response. To this end, a number of random forest models were constructed and trained on a dataset consisting of controllable gameplay features, user behaviour and user response data on a number of questions regarding frustration.

The current approach of predicting frustration was mildly successful. The tuned CRD classifier reached a 66.3% accuracy predicting 5 classes and 81.4% accuracy when predicting 3 classes. These accuracies are significantly higher than chance (20% and 33% respectively, assuming equal class distributions). Furthermore, most errors fall within 1 class difference and extreme errors are rare, with the exception of the highest frustration class. This means the model usually predicts a value close enough to give a useful indication of frustration state. While still far from ideal, prediction of user frustration with a random forest model seems to have some potential.

The tuned CRD classifier seems to have the least success with the ambiguous classes in the middle of the frustration spectrum. These classes might be hard to predict because what causes smaller amounts of frustration differs greatly, while very frustrating events might be more uniformly experienced. The idea that individuals experience frustration differently has some basis in the literature, as the 6th component of the frustration-aggression hypothesis, Personal Satisfaction, is an internal factor that differs from person to person. The idea of a universally frustrating experience is less pronounced in the literature. However, seeing the importance of the

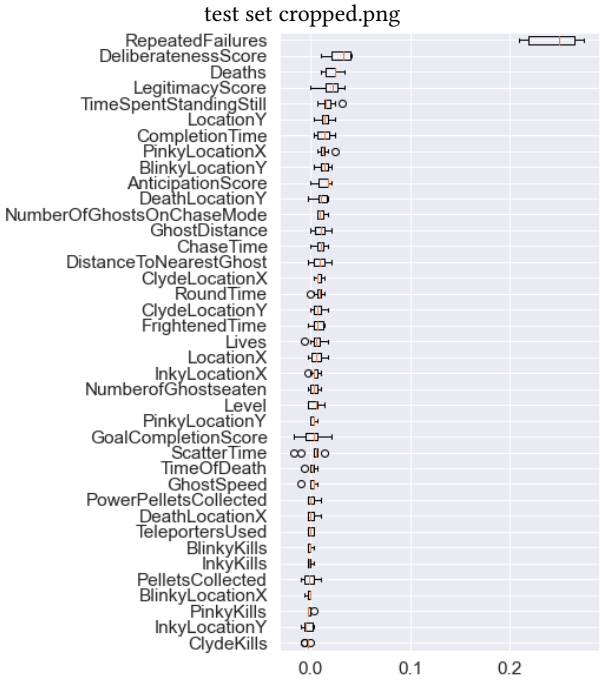

Figure 5: Permutation Importances calculated on the test set

RepeatedFailures component it is plausible that repeatedly failing is universally seen as very frustrating.

However, model performance is usually mostly dependent on quality and quantity of available data. That can be seen in this model as well, as model performance decreases with the amount of data available. Class 1 accounts for nearly 40% of all observations with class 2 accounting for another 24%. Coincidentally, these are the classes where we see the best performance. On the other side of the spectrum, class 4 only accounts for 12% of the observations. This might force the model to learn patterns from a very limited amount of observations, and be tested on a small set which is vulnerable to outliers. This is especially true with regards to the noisiness of the signal that is to be predicted.

The only exception to this seems to be class 5, the highest frustration class. This class only accounts for 6.3% of all observations However, it might be that there are very strong indicators that serve as a sort of 'cut-off' point of frustration. For example, someone dying 3 times within 30 seconds might always experience high frustration as these deaths compound the amount of felt frustration as seen in the importance of the RepeatedFailures feature. These indicators will then be a very good predictor of class 5, which the model can use. These predictors might not exist for the more ambiguous classes, as what causes small amounts of frustration might vary more between subjects than what causes large amounts of frustration. This again harkens back to the idea of a universally frustrating experience.

Additionally, the performance of both models heavily depends on the user response to the frustration component questions. Performance drop significantly if you exclude these datapoints from the learning process. Furthermore, the classifier starts to gain a heavy bias towards predicting class 1, which leads to an overestimation of actual accuracy as class 1 is the largest class. This is unfortunate, since one of the aims of this research was to see if it was possible to predict user frustration without prompting the user. It seems that the used method, with the used variables, is unsuitable for predicting user frustration from just game data, at least with the amount of data collected. This might be corrected by finding a better way to measure frustration than these 30 second segments. It might be that frustration develops in longer-term interactions that these segments did not capture. Furthermore, the interruptions caused by the questionnaire might be a cause of frustration in and of itself.

5.2 Can this prediction be explained?

Feature importance analysis reveals some interesting findings. Again we see the importance of the frustration components. RepeatedFailures is overwhelmingly the most important feature, with Deliberateness and Legitimacy as second and fourth. This explains why the model loses much of its predicting power when these features are dropped out of the training dataset. A random forest model prediction of the RepeatedFailures component resulted in a average accuracy (60.1%), with no single feature standing out as contributing much to performance. This can suggest two things. One, the current dataset of gameplay features does not contain the necessary variables to accurately reflect when Repeated Failures happen in game. Two, it is not necessarily actually repeatedly failing in game that causes frustration, but just the perception of repeatedly failing. In the first case, efforts could be made to try to capture the RepeatedFailures component with a different variable set. Perhaps one where different types of failure or some sort of failure similarity metrics are incorporated. In the second case one should try finding variables that correspond with this mental state, which might be different from those that common sense would assume are correlated.

Additionally, the random forest classifiers that predicted frustration component responses yielded a relatively mediocre accuracy. The CRD accuracy was slightly lower on 5 class predictions than the CRD predictions of player frustrations. This slight reduction can be explained by the relative importance of frustration components. Since the CRD Frustration Classifier has all components to learn from, and frustration components are a relatively good predictor of frustration, the model has a slightly higher accuracy. Components predictors however have one less component to learn from, and do not have access to TotalFrustration to compensate, leading to a slightly worse performance. Additionally, this finding seems to reinforce the idea that the used variable set does not accurately reflect when these frustration components are triggered. It seems different variable sets are needed to explain where the frustration is coming from.

The tuned classifier has a lower accuracy with participant data than other methods used by researchers when predicting 5 classes. For example, Yannakakis et al's[21] approach based on evolutionary learning attained a 88.66% accuracy on the frustration dimension. Kapoor et al[17] used a assessment method based on Gaussian process classification and Bayesian interference and reached an accuracy of 79%. However, it must be noted that both these studies had significantly higher chance conditions so higher accuracies are to be expected. All in all it seems that the currently used method of frustration prediction has some merit, but requires further research to unlock its full potential.

5.3 Conclusion

In conclusion, a study was conducted in which a number of participants played variations of Pac-Man. While playing, these participants were asked about their experience in relation to five components of frustration as mentioned in the frustration-aggression hypothesis. Gameplay features, participant behaviour and participant responses were captured and used as a dataset to train a number of random forest classifiers. These classifiers were trained to predict frustration, with results ranging from average (66.3%) to good (83.1%) accuracy when predicting 5 or 3 frustration classes respectively.

Accuracy dropped significantly when not incorporating participant responses in the dataset. Correlation analysis revealed significant correlations between all frustration components and frustration, as well as additional correlations between certain gameplay features and frustration. Feature importance analysis revealed the influence of the Repeated Failures component, as well as the relatively low importance of all other gameplay features. These results suggest that the frustration components can be useful as an explanation for the cause of frustration. Furthermore, the used

method of frustration prediction leads to useable models that predict a near-enough value to be useful. However, efforts need to be made to find a variable set that captures the frustration components to predict without prompting the user.

These results are a first step in the direction of a game that dynamically adjusts to player frustration. Furthermore, the results can be used to inform researchers to create better models of user frustration, as well as further the search in what exactly creates a frustrating game.

5.4 Future Work

There are a number of avenues for future work. Player types can investigated, to see if there is a relationship between player type and finding certain variables frustrating. Additionally, more factors than those described in the frustration-aggression hypothesis can be incorporated in the model, which could improve prediction accuracy. Once satisfactory accuracy is reached, efforts can be made at creating an adaptive game A.I that detects and adapts to player frustration. These advances can then be used to enhance the gameplay experience for users, allowing for greater player retention in commercial games, and a lower dropout chance in educational ones.

REFERENCES

[1] A Amsel. 1958. The role of frustrative nonreward in noncontinuous reward situations. *Psychological Bulletin* 55(2) (1958), 102.

[2] A. Amsel. 1992. *Frustration theory: An analysis of dispositional learning and memory.* Cambridge University Press.

[3] Anonymous. 2020. Appendix. *at: https://drive.google.com/file/d/1wg3nwBRG0E3t1d5OdlmuTDf0fA9vJQxW/view?usp=sharing* (2020).

[4] M. S. Bartlett, G. Littlewort, I. Fasel, and J. R. Movellan. 2003. Real time face detection and facial expression recognition: Development and applications to human computer interaction. *Paper presented at the 2003 Conference on Computer Vision and Pattern Recognition Workshop, (Vol. 5, pp. 53-53)* (2003).

[5] L. Berkowitz. 1978. Whatever happened to the frustration-aggression hypothesis? *American Behavioral Scientist* 21, 5 (1978), 691–708.

[6] L. Berkowitz. 1983. Aversively stimulated aggression: Some parallels and differences in research with animals and humans. *American Psychologist* 38, 11 (1983), 1135.

[7] L. Berkowitz. 1989. Frustration-aggression hypothesis: Examination and reformulation. *Psychological Bulletin* 106, 1 (1989), 59.

[8] L. Breiman. 2001. Random forests. *Machine learning* 45, 1 (2001), 5–32.

[9] R. Cade and J. Gates. 2017. Gamers and video game culture: An introduction for counselors. *The Family Journal* 25, 1 (2017), 70–75.

[10] C. Conati. 2002. Probabilistic assessment of user's emotions in educational games. *Applied Artificial Intelligence* 16, 7-8 (2002), 555–575.

[11] M. Csikszentmihalyi, S. Abuhamdeh, and J. Nakamura. 2014. Flow. In *Flow and the foundations of positive psychology.* Springer, 227–238.

[12] J. A. DeFalco, J. P. Rowe, L. Paquette, V. Georgoulas-Sherry, K. Brawner, B. W. Mott, and J. C. Lester. 2018. Detecting and addressing frustration in a serious game for military training. *International Journal of Artificial Intelligence in Education, 28(2), 152-193.* 28, 2 (2018), 152–193.

[13] J. Dollard, N. E. Miller, L. W. Doob, O. H. Mowrer, and R. R. Sears. 1939. Frustration and aggression. *New Haven, CT: Yale University Press* (1939).

[14] C.B Ferster. 1957. Withdrawal of positive reinforcement as punishment. *Science, 126, 509* (1957).

[15] S. A. Jin. 2011. I feel present. therefore, I experience flow: A structural equation modeling approach to flow and presence in video games. *Journal of Broadcasting & Electronic Media* 55, 1 (2011), 114–136.

[16] S. A. Jin. 2012. Toward integrative models of flow : Effects of performance, skill, challenge, playfulness, and presence on flow in video games. *Journal of Broadcasting & Electronic Media* 56, 2 (2012), 169–186.

[17] A. Kapoor, W. Burleson, and R. W. Picard. 2007. Automatic prediction of frustration. *International Journal of Human-Computer Studies* 65, 8 (2007), 724–736.

[18] H. M. Mentis and G. K. Gay. 2002. Using touchpad pressure to detect negative affect. Paper presented at the Proceedings. In *Fourth IEEE International Conference on Multimodal Interfaces.* 406–410.

[19] A. Newhall, W. C. Pedersen, M. Carlson, and N. Miller. 2000. Displaced aggression is alive and well: A meta-analytic review. *Journal of Personality and Social Psychology* 78, 4 (2000), 670.

[20] Newzoo. 2019. Global games market report (lite version). *at: https://resources.newzoo.com/hubfs/2019_Free_Global_Game_Market_Report.pdf?utm_campaign=Games* (2019).

[21] C. Pedersen, J. Togelius, and G. N. Yannakakis. 2009. Modeling player experience in super mario bros. In *Paper presented at the 2009 IEEE Symposium on Computational Intelligence and Games.* 132–139.

[22] J Pittman. 2009. The pac-man dossier. *found at https://www.gamasutra.com/view/feature/3938/the_pacman_dossier.php?print=1* (2009).

[23] N. Shaker, G. Yannakakis, and J. Togelius. 2010. Towards automatic personalized content generation for platform games. *Paper presented at the Sixth Artificial Intelligence and Interactive Digital Entertainment Conference* (2010).

[24] S. E. Siwek. 2007. Video games in the 21st century: Economic contributions of the US software entertainment industry. *Entertainment Software Association Report* (2007).

[25] P. Sweetser and P. Wyeth. 2005. GameFlow: A model for evaluating player enjoyment in games. *Computers in Entertainment (CIE)* 3, 3 (2005), 3.

[26] G. Van Lankveld, S. Schreurs, and P. Spronck. 2009. Psychologically verified player modelling. *Paper presented at the Gameon* (2009), 12–19.

[27] J. H. Westerink, Van Den Broek, L. Egon, M. H. Schut, J. Van Herk, and K. Tuinenbreijer. 2008. Computing emotion awareness through galvanic skin response and facial electromyography. In *Probing experience.* 149–162.

[28] G. N. Yannakakis, P. Spronck, D. Loiacono, and E. André. 2013. Player modeling. *Artificial and computational intelligence in games, DFU, 6 (pp.45-59).* (2013).

[29] H. Yu and T. Trawick. 2011. Personalized procedural content generation to minimize frustration and boredom based on ranking algorithm. *Paper presented at the Seventh Artificial Intelligence and Interactive Digital Entertainment Conference* (2011).