Abstract
This work addressed the problem of recognizing player personality as a first step in entertainment modelling. We describe our solution that is comprised of a task-based scenario and a bayesian network system that classifies personality based on sample data from the scenario. We also extensively describe our methodology for developing the task-based scenario, from initial task collection to integration with the bayesian network system. To validate our solution we applied our methodology to develop a Minecraft-based scenario and ran cross-validation tests over our system to obtain an average success rate of 68.97%. We have thus successfully developed the first part of an adaptive system, one that can identify personality and interests to adapt content and increase entertainment in games.

Keywords: Entertainment Modelling, Video Games, Personality Models, Machine Learning, Probabilistic Reasoning.

1 Introduction

1.1 Motivation
In the videogame industry the costs for development have grown and assets such as graphics, animations and scenarios require more and more manpower. These costs have made publishers avoid risk by looking for games with solid proven gameplay. This approach led to a mass production of games from similar genres (such as the first person shooter), leaving creativity somewhat stagnant and thousands of potential customers uninterested and unengaged.

One way to reduce risk would be to use entertainment modeling to adapt the game content in real time so it fits the players interests [1]. Unfortunately, everyone learns at different rates and in different ways[2], making player necessities complex and difficult to understand.

To help them understand players, developers have resorted to psychological theories such as flow theory to engage players[3][4][5]. International Hobo, a videogame consulting group, advises[6] to move beyond the dependence of personal judgements when designing games by viewing the process of game design from a psychological perspective.

In his thesis [7], Ricardo Dias has confirmed that adapting the content of videogames according to personality typologies can increase their entertainment value. As part of future work he recommends the extraction of personality before actual content adaptation. Our work intends to follow Dias’ recommendation and we have developed a prequel stage to his solution.

1.2 Problem Description
This work addressed the problem of how to infer, from data gathered at runtime, the personality preferences of players for content adaptation. To find a solution we presented the hypothesis that actions and choices could reflect the preferences of individuals.

Once personality has been gathered, using our
solution, the acquired information can be used to perform a more precise modelling and increase enjoyment value of games.

2 State of the Art

2.1 Personality Psychology

Our work is heavily based on previous studies done by the personality psychology community. Since personality psychology is a field based on empirical analysis, very different models have appeared. We begin by identifying some of the most important theories of the field.

**Meyers-Briggs Type Indicator (MBTI).** MBTI is one of the most used personality psychology models in the field of interactive entertainment[8]. It is based on the work of Carl Jung[9] that identified four functions of the human mind: Sensation (S) vs. Intuition (N) for perceiving, and Thinking (T) vs. Feeling (F) for judging. Each of Jung’s functions could be modified by two attitude types: Extraversion (E) vs. Introversion (I). MBTI underlines the need for a fourth dimension to explain interaction with the outside world - Judging (J) vs. Perceiving (P). All of the dimensions of MBTI appear as dichotomies and divide the human spectrum into sixteen possible outcomes.

**Keirsey Temperament Model (KTM).** Analysis of classical temperaments and MBTI allowed Keirsey to present his temperament theory[10]. This theory focuses on behaviour rather than thoughts and emotion. Keirsey extrapolates four archetypes from the sixteen MBTI junctions. The SJ or Guardians, the SP or Artisans, the NT or Rationals and the NF or Idealists. The nature of KTM behavior patterns is ideal to assess the skillset [8][11] by players when facing a challenge.

Other personality models such as Five Factor Model (FFM) and Cloninger’s Temperament and Character Inventory (TCI) were researched but did not contribute meaningfully to our solution.

2.2 Personality based player models

Besides psychology models, several game design studies have been made with the ambition of deriving player typefication from psychological theories.

**Bartle player types.** A study conducted by Richard Bartle on multi-user dungeons (MUDs) brought one of the first, and most enduring, player models[12]. Bartle analyzes the interaction patterns across players and identified two common axes. The Interacting - Acting axis and the Players - World axis. From these axis four player types emerge, the Socializer, the Killer, the Achiever and the Explorer. Socializers (Interacting, Players) derive enjoyment in other players and interacting with them. Killers (Acting, Players) become engaged while acting on others, usually causing grief or showing superiority. Achievers (Acting, World) act over the world and mastering it is their goal. Explorers (Interacting, World) want to be surprised by the world and find new things about it.

**Demographic Game Design (DGD).** DGD is based on MBTI and focuses on market oriented game design[6]. Out of the four dichotomies of MBTI, DGD uses only two, TF and PJ. These were shown as being the most discriminative when applied to gaming preferences.

Players can fall into one of four clusters, each with specific skillsets[8]: the Conqueror (TJ) has a strategic-logistical skillset; the Manager (TP) has a strategic-tactical skillset; the Wanderer (FP) has a diplomatic-tactical skillset; the Participant (FJ) has a diplomatical-logistical skillset. Additionally, they identified differences inside each cluster depending on whether player was casual or hardcore. Casual players possess the logistical or tactical skillsets, while hardcore players possess the strategic or diplomatic skillsets of their cluster.

**Lazzaro’s fun types.** Lazzaro’s work [13] into player motivation shows an important relation between play style and emotion. The work identifies four emotional keys to identify player experience - Hard fun, Easy Fun, Serious Fun and The People Factor. Hard Fun players seek challenges, strategy and problem solving. Easy Fun players become immersed in intrigue and curiosity. Serious Fun players gain enjoyment from internal experiences as reaction to visceral, behavior, cognitive and social aspects of the game. The People Factor players look for social experiences and use games as a mechanism to obtain them.
Unified Model. Comprehensive analysis was made in order to propose a unified model [14]. The unified model connects KTM, Bartle Mud types, DGD, MBTI, Lazzaro fun types and other game design models such as Gamist/Narrativist/Simulationist [15] and Mechanics/Dynamics/Aesthetics [16].

This extensive model suggests that Keirsey temperaments are a direct superset of Bartle MUD types and that Lazzaro fun types appear directly associated to each temperament and MUD type. A Killer is a type of Artisan who looks for Serious Fun. An Achiever is a type of Guardian who looks for Hard Fun. An Explorer is a type of Rational that looks for Easy Fun. A Socializer is a type of Idealist that looks for The People Factor.

DGD is related indirectly due to the MBTI dichotomies, and it fills the gaps between Temperaments. From this relation, the DGD skillsets can be assigned with each of the Temperaments; Artisans are Tactical, Guardians are Logistical, Rationals are Strategic and Idealists are Diplomatic.

2.3 Entertainment modelling

Entertainment modeling is a direct, practical application of our work. Modelling games can be done using different kinds of theories or empiric measures and it is crucial to understand previous work and research.

Research into entertainment modelling studies [7][17][18][19][20] shows that the work done is still focused on adapting challenge or difficulty. Flow appears as a recurrent due to its direct relation to game difficulty and balance. Affective data, that can be acquired from physiological signals, is also common as it is an interesting measure to complement performance.

For our work it is important to note that all presented studies suggest that extracting player personality can play an important part in increasing entertainment value. Dias’ thesis[7] revalidates the need for detecting personality traits previous to content adaptation. The work of Yannakakis et, al.[17], despite showing positive signals still fails 32% of the time in modeling children. Their conclusions indicate that potential data could be acquired from personality to increase its success rate. Chen[18] comments that at runtime games still do not understand what players are thinking, something that can be explained by personality theory. Rani et, al.[19] have shown how profiling is important in feature extraction, as well as physiological response to map affective states. PaSSAGE[20] used a rudimentary typology associated with tabletop role-playing games to model storytelling.

These studies reinforce our motivation for the need of creating a system that can infer in real time player personality before modelling and adapting content in games.

3 Solution

To characterize player typology we propose a solution comprised of a game scenario and an inferring system. The scenario is used to acquire player decisions, and it was previously designed by following our hypothesis that player actions can be a reflection of their preferences. The data acquired from the scenario is fed to the inferring system that uses it to classify players.

To develop this solution, we created our own design methodology based on Keirsey Temperaments. With our methodology, we intend to give tools that allow future researchers, or designers, to create their own version of the solution, adopting other types of player models into different games.

We choose Keirsey Temperament theory since it is a behavior oriented model. Additionally, due to its connection to MBTI dichotomies it can be associated to DGD and Bartle MUD types giving additional insight to expected temperament preferences in game.

In order to validate both our solution and its design methodology, we developed a sample scenario, that we entitled Rooms, for the game Minecraft. This scenario was then used to create the inferring system based on Bayesian networks. Having both the system and the scenario, we conducted an experiment to acquire player data using Rooms.

Minecraft was chosen due to its open attitude towards modding, complex world-based mechanics and ease of constructing ingame scenarios. Developing for Minecraft was done using the Minecraft Coder Pack (MCP) tools that allow decompilation and recompilation of the source code into the Java language.

The following subsections present our Design Methodology as it was used to develop Rooms, the inferring system model and the experiment conducted.
3.1 Design Methodology

In our work we intended to do more than exemplify the implementation of an inferring system for a particular game. We also wished to develop a methodology to design game scenarios that could be ported across genres, games and adapted to use different personality models. Our developed methodology is comprised of four chronological stages: Task Identification; Observation; Task Clustering; Scenario Design.

Task Identification. In this stage we established a list of potential meaningful tasks and temperament profiles. We looked at Minecraft mechanics, such as placing blocks, and identified tasks, such as building complex structures. We stopped identifying tasks once we found that our list was extensive enough to cover most of the possible game mechanics. It is crucial to state that this was not the final task list as it would be refined and improved over the next stages.

With the initial list finished, we still needed to understand the preference that temperaments gave to each task. For this, we looked at the relation between Keirsey temperaments and other personality models, especially Bartle MUD types and DGD, and created profiles for each temperament. Guardians are goal oriented individuals who try to validate their in-game duty by acquiring rare and powerful materials or achievements. Artisans are impulsive and enjoy combat and killing. Rationals seek to understand and explore both the world and game mechanics. Idealists enjoy artistic expression and are pulled away from straining dangerous situations such as combat. Using these profiles we gain some expectancies regarding the enjoyability of the temperaments over the task list.

Observation. The task list obtained in the previous stage was still crude and only an expectation of preference based on subjective interpretation. It was now required to corroborate our assumptions and reaffirm the usefulness of Keirsey temperaments for our work. To do this we set up an experiment to observe players and their actions while playing.

The experiment lasted between thirty to forty minutes and was done both in person and using the Skype platform. Before playing, players were asked to answer a personality survey that would give us their corresponding temperament. During play, the frequency of their actions was registered according to the task list. To help align their decisions with our task list, players were asked to maintain constant conversation.

At the end of the experiment, ten individuals were observed. Their data helped us verify the temperament preferences over each task. This information helped us rewrite our temperament profiles according to empirical data and create some rules of thumb to separate temperaments. The size of the sample led to the information of Artisan and Idealist temperaments to be scarcer than the remainder temperaments.

Task Clustering. Our task list for Rooms contained 24 tasks. These were far too excessive to classify personality typologies. Using the data from the previous step, we resorted to clustering algorithms to obtain clusters from tasks. These clusters aggregate tasks according to temperament preference making them smaller and more manageable information.

We used SPSS Statistics 2.0 software from IBM to calculate the k-means clustering algorithms. Ideally, we wanted to set $k=4$, meaning that we would have four clusters to work with.

In an ideal situation each temperament would have a specific preferred cluster. Unfortunately the scarcity of information for the Artisan temperament made the first cluster preferred by three temperaments, and made four clusters insufficient to distinguish temperaments.

Finally at $k=5$ we reach a group of clusters where the preferred cluster of each temperament is different. These clusters were manageable enough to be instantiated in our scenario.

Scenario Design. The goal of developing a scenario is to allow player choice between several clusters. At each stage the player is presented with tasks and he is free to choose which he enjoys best. This follows our hypothesis that decisions can indicate personality and that session data from the scenario can be used as evidence for our network to infer personality. Before achieving the final architecture of the scenario, we have to decide where the player choices are located and how player is transposed from choice to choice.
For our architecture we started assuming junctions of task rooms, represented in figure 1, each containing two tasks with which the player would interact before proceeding to the next room.

In terms of scenario architecture we began by assuming a binary tree architecture of rooms, where at the end of each room players would decide the next room based on a description. This architecture had an excessive number of rooms and made the system excessively dependent on the player interpretation of the description. Afterwards, we looked at a collapsed architecture that reduced the number of rooms but maintained the description issue.

Finally, we achieved the solution architecture showed in figure 2. This architecture forces decision on exit regarding the already experienced tasks. By deferring decision, the architecture is no longer dependent on the personal interpretation of descriptions. Another issue rises: the dependency on well implemented tasks. Rooms that contain badly implemented tasks, that do not accomplish their designed goals, can influence the inferring network negatively. These malformed rooms needed to be identified and removed to help the system achieve maximum efficiency.

Once the architecture was chosen, it was required to assign clusters to each room. By combining pairings of clusters and analysing which cluster each temperament would prefer, we were able to reach four patterns to separate temperaments. These patterns were: Separate Guardians and Artisans from Rationals; Separate Guardians and Artisans from Idealists; Separate Rationals from Idealists; Isolate Idealists. Each of these patterns forms a goal for a room. From the clusters that form the pairings of a pattern, we can select the tasks that better suit the goal, by looking at the already known preference of temperament over tasks.

The Artisan temperament appears tied to the Guardian temperament, likely due to the small sample from the experiment in Observation. To tackle this issue we created three additional goals, to try and separate the Artisan from each temperament. Once again we chose tasks that supported these goals before assigning them to the rooms.

At the end of this stage we posessed not only the final architecture of Rooms, but also its final seven rooms, each with two tasks assigned.

### 3.2 Inferring System

To build our inferring system we modeled a bayesian network based on the possible decisions of players in the scenario. To do this, we resorted to our hypothesis that player preferences and actions can be indicative of personality, or in other words, actions and preference are outcomes of personality. Using this hypothesis, we built a simple bayesian network.

We began by creating a parent node, a cause, that has four possible states, the Keirsey temperaments. Then we created a child node, an effect, for each player decision inside our scenario. This should lead us to the network shown in figure 3, where our total of N choices is 7.
With no sample data, our system assumes that all temperaments have the same probability of leading players to make a choice. This starting stage is uninteresting for classifying players but its probabilities can be increased by machine learning. As players interact with the scenario the data acquired from the decisions can be coupled with actual player temperament, acquired offline, and fed to the system. With samples, the system can update its knowledge base and improve its success rate for future classifications of temperaments.

Our actual implementation for Rooms was made using Decision Systems Laboratory SMILE (Structural Modelling, Inference, and Learning Engine) software, made in the University of Pittsburgh. This software allows the easy creation of bayesian networks and has jSMILE, a dynamic link library (DLL) for the Java language, the same language as Minecraft.

3.3 Experimental protocol

Having Rooms implemented in Minecraft, and our inferring system coupled to the scenario, we proceeded to obtain sample data. We ran an experiment to acquire samples of player decisions over the scenario.

The experiment began by acquiring player temperaments using an MBTI questionnaire. We used an MBTI questionnaire simply due to the ease to acquire a cost-free validated copy. Additionally, the connection between MBTI typologies and Keirsey temperaments allows the extrapolation of temperaments from the questionnaire.

Players were then allowed to play Rooms, crossing all seven rooms, from entrance to exit. They were asked to visit both tasks before deciding which they preferred. Decisions were made by leaving by the door adjacent to the preferred task.

At the end of the experiment, we possessed enough information to analyze the usefulness of our solution and improve the scenario.

4 Data Analysis

Our experiment collected data from thirty two individuals, mainly from different universities of Lisbon. Most of these belonged to courses related to Computer Science. The sample population was composed of 11 Guardians, 13 Rationals, 4 Idealists, 1 Artisan and 3 ambiguous individuals whose temperament could not be identified. The ambiguous samples were discarded from further testing.

For the remainder 29 identified temperament samples, we ran k-fold cross-validation tests, using a $k = 10$ value. These tests divided the population into sets of $n/k$ elements, where $n$ is our population size of 29. For our particular sample we obtained 9 sets of 3 elements and one set with 2 elements.

The test then ran $k$ steps of selecting a set, feeding the remainder sets to our inferring system and using the selected set to perform personality inferring. At the end of the $k$ steps we acquired an average hit rate of only 51.72%. The best predicted temperament was the Guardian with 72.73% hit rate. The system was unable to predict any Artisan or Idealist temperament.

To improve on these results we ran further k-fold cross-validation tests to try and identify malformed rooms. We began by looking at the influence each room had on the system’s success.

Once the influence of each room was ascertained, we discovered that removing rooms 1, 5, 6 and 7 led to a reduction of success rate, while removing rooms 2, 3 and 4 improved it. The same results showed true for the Guardian and Rational temperaments, while the system was again unable to predict any Artisan or Idealist temperaments.

When we understood that the set of rooms $\{1,5,6,7\}$ was potentially well formed, while the set $\{2,3,4\}$ was potentially malformed we ran further k-fold cross-validation tests, this time using only sample data from each room. Rooms 1, 5 and 6 appeared to remain well formed and rooms 2, 3 and 4 malformed, room 7, however, appeared
to have worse success rate when analyzed individually.

Having this insight on the rooms, we proceeded to our final k-fold cross validation tests. This time we ran tests only for the combinations of potentially malformed rooms, the sets \( \{2,3,4\} \), \( \{2,3,4,7\} \), and well formed rooms, the sets \( \{1,5,6,7\} \) and \( \{1,5,6\} \).

The results of the malformed rooms were in fact much lower than the original success rate. With the first set however, the system succeed in classifying one Idealist.

The results of the well formed rooms were the same for both sets, with an average success rate of 68.97%, 81.82% for Guardians, 84.62% for Rationals and 0.0% for Artisans and Idealists.

Having identified the malformed rooms led us to an increase in the systems success rate. Removing these rooms from the scenario also allows us to reduce its duration time making it lighter for players to interact with. Room 7 appears as a special case, neither entirely malformed nor well formed, but since it does not increase the average success rate, removing it makes the scenario smaller and faster to finish.

5 Conclusions

Entertainment modelling has been used to increase the enjoyment value and engagement of players in a game. Approaches to validate entertainment modelling systems have required a previous offline acquisition of player personalities, such as the case of Ricardo Dias[7]. This leaves the problem of classifying personality, prior to the adaption of game content and during play sessions.

Our work has contributed to a possible solution for classifying personality in a game. We presented a solution comprised of a game scenario to acquire player preference data and an inferring system to classify player typologies. Additionally, we have developed a design methodology to create scenarios and inferring systems for different types of games, genres and personality models.

To validate our methodology, we presented a practical example with the scenario Rooms for the game Minecraft, using Keirsey temperaments as a personality model. Our system was able to infer personality approximately 70% of the times, over 80% for the temperaments better represented in our sample.

We advise users of our methodology to gather enough user sample to represent each of the temperaments, as this might increase their system’s success. Having approached mainly players from the field of Computer Science led our experiment population to have a greater number of Guardians and Rationals. Users of our methodology, as well as other personality-based research, should be wary of this circumstance and vary the background of their experiment population.

As our work built upon Dias’ thesis[7] we notify of the possibility of joining both researches in a larger study. This study would first recognize personality using our solution and then adapt content according to player preferences in a seamless and smooth play session.

For future work, we recommend using our methodology to create systems to infer personality for other games besides Minecraft, and other genres of games besides the sandbox genre. We also leave the possibility of adapting our design methodology and inferring system to use other personality models besides the Keirsey temperaments.

References


