A Methodology for Approximating Motivation-Related Latent States in Large Scale Scenarios

And its Role in Engagement Prediction within a Video Game Setting

Valerio Bonometti

Doctor of Philosophy

University of York

Computer Science

December 2022

Abstract

Motivation is a fundamental psychological process guiding our everyday behaviour. For doing so, it heavily relies on the ability to attribute relevance to potentially rewarding objects and actions (i.e., incentives). However, despite its importance, quantifying the saliency that an individual might attribute to an object or an action is not an easy task, especially if done in naturalistic contexts. In this view, this thesis aims to outline a methodology for approximating the amount of attributed incentive salience in situations where large volumes of behavioural data are available but no experimental control is possible. Leveraging knowledge derived from theoretical and computational accounts of incentive salience attribution, we designed an Artificial Neural Network (ANN) tasked to infer a latent representation able to predict duration and intensity of future interactions between individuals and a series of video games. We found video games to be the ideal context for developing such methodology due to their reliance on reward mechanics and their ability to provide ecologically robust behavioural measures at scale. We developed and tested our methodology on a series of large-scale ($N > 10^6$) longitudinal datasets evaluating the ability of the generated latent representation to approximate some functional properties of attributed incentive salience. The present work opens with an overview of the concept of motivation and its interconnection with engagement in a video-game setting. It proceeds by formulating the theoretical and computation foundations on which our methodology is built upon. It then describes the iterative process of model building, evaluation and expansion underlying the implementation of our methodology. It continues by analysing the latent representation generated by the ANN and comparing its functional characteristics with those of attributed incentive salience. The manuscript ends with a general overview of the potential applications of our methodology with a particular focus on the area of automated engagement prediction and quantification in videogames settings.

Declaration

I declare that this thesis has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree other than Doctor of Philosophy of the University of York. This thesis is the result of my own investigations, except where otherwise stated. Other sources are acknowledged by explicit references. All sources are acknowledged as References. I hereby give consent for my thesis, if accepted, to be made available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations. The work presented in this thesis is the result of a partnership between University of York and Square Enix Ltd. The PhD project has been developed full-time within the analytics and data science team of Square Enix Ltd. It was supported by the EPSRC Centre for Doctoral Training in Intelligent Games & Games Intelligence (IGGI) through grant [EP/L015846/1]. All data employed in this work were obtained and processed in compliance with the European Union's General Data Protection Regulation [1] and Square Enix Ltd. data protection policies. Chapter 3 is based on the work carried out in [2, 3]. Chapter 4 is based on the work carried out in [2–4]. Chapter 5 is based on the work carried out in [3]. Other contributions that were not included in this doctoral thesis can be found in [5] and [6].

Acknowledgements

This work was supported by the EPSRC Centre for Doctoral Training in Intelligent Games & Games Intelligence (IGGI) through grant [EP/L015846/1].

The partner company Square Enix Ltd. provided all the data used for the development of this thesis. We want to highlight how this work would have not been possible without them. Therefore, we are profoundly grateful and thankful to the whole Square Enix Analytics and Insight team and to its leadership in particular.

I would like to thank my supervisors Alex Wade and Anders Drachen for their presence and support as this is something that undoubtedly shaped my PhD journey.

Finally, a special thanks goes to all the colleagues and friends that accompanied me during these challenging but surely exciting years.

To Mum and Dad
To Simona

Contents

List of Tables

List of Figures

Chapter 1

General Introduction

1.1 Introduction

When we see a professional athlete competing at an event we often find ourselves thinking "how much work they must have done to reach such level of performance". Similarly we might be surprised discovering the effort miners were putting in finding even small amount of gold nuggets during the 19^{th} century gold rush. Looking at something closer to our everyday experience, we might widen our eyes noticing how many hours we sank watching the latest TV-series or playing our favourite videogames.

But what do all these activities have in common? They are rather different in nature but are nevertheless able to elicit prolonged and vigorous behavioural responses. Indeed, it appears that human beings are capable of remarkable feats when trying to achieve goals that lead to positive and pleasurable outcomes for them. From a psychological point of view, we say that in all those instances a common set of cognitive and affective processes, which go under the umbrella of "motivation", are involved in the generation of goal-directed behaviour.

This implies that knowing the motivational state of an individual during a particular activity, puts us in a favourable position for understanding qualitative aspects of their current and past behaviour as their intensity and future likelihood. Within this general framework lies the aim of this thesis: we attempted to develop a methodology for deriving an approximate quantification of the motivational drive of individuals in situations where large volumes of observational data are present but no direct contact is possible. Indeed, without direct access to individuals, more traditional (and potentially more accurate) methods for inferring their motivational state are simply not feasible.

1.2 Thesis Outline

The work carried out for this thesis, although it originated in the need to address a series of academic challenges (e.g. inferring the motivational states of individuals from large scale behavioural data) has been developed completely within an industrial setting. It is therefore the product of two separate although complementary goals: the academic desire to advance knowledge and tackle novel problems and the constrains imposed by the industry to use this knowledge for developing concrete and actionable tools and products. I therefore present data and models that derive from scientific analysis but I also include extended discussions of how these contributions can be used in the future in industrial applications.

1.2.1 Academic Outline

From an academic point view we found that despite there has been noticeable interest in the estimation of the internal states generated by various psychological processes (e.g. motor control [7], memory [8, 9], visual [10, 11] and olfactory [12] perception), less work has taken into consideration the construct of motivation [13, 14].

In addition, most of efforts in this area have leveraged data generated in laboratory or simulation settings [10, 15–23] leaving the estimation of internal states in observational settings a less explored venue. When attempts in this direction were made, the focus was mostly on animal behaviour and the adopted methodologies relied largely on completely data-driven approaches [20–22].

In this view this thesis focuses in the first place on creating a theoretical framework that bridges computational accounts of motivation with data-driven approaches for its inference. It then leverages this theoretical framework to design, develop and validate a methodology for approximating the motivational states of individuals using large volumes of behavioural data acquired in naturalistic settings.

1.2.2 Industrial Outline

One major factor limiting the inference of psychological states in naturalistic settings has been the difficulty in acquiring sufficient amounts of data describing individual behaviour in a relatively unbiased manner.

This issue has been partially resolved by the relatively recent ability to acquire these types of data through the use of telemetry systems [24, 25]. A practice, that has seen an dramatic increase recently, especially in industrial settings [1, 24, 25]. For this reason, a collaboration between academia and industry appears a promising venue for tackling some of the challenges outlined in paragraph 1.2.1.

However, despite the fact that there might be an overlap between the needs of research and industrial institutions, commercial entities are often laser-focused on delivering practical and actionable tools

serving the ultimate purpose of improving or optimizing operations and, driving revenues. This thesis highlights how motivation is deeply connected with the concept of **engagement**, a well known construct in the videogames industry with widespread implications ranging from user experience quantification to revenue optimization. It also illustrates how the methodology introduced in section 1.2.1 can be directly applied for solving tasks related to automated assessment, quantification and prediction of user engagement at large scale.

1.3 Hypotheses

A more detailed description of the hypotheses and assumption that drove our work can be found within each chapter of the thesis. A brief overview of them is reported here:

Chapter 2 : we hypothesized that engagement could be better understood as the behavioural manifestation of changes in the latent motivational state of individuals.

Chapter 3 : we hypothesized that the motivational state of an individual could be approximated by the solution to a specific supervised learning problem. In particular:

- The problem should be concerned with predicting the intensity of future interactions between an individual and a rewarding object (a videogame in this thesis) based on the history of past interactions.
- The model used for tackling such problem should be able to represent a large class of functions, embed different types of rewarding object, simultaneously predict multiple indices of behavioural intensity and also model temporal dynamics.
- When information about the characteristics of the rewarding objects and the environment in which the interactions occurred are available, the model is able to produce more accurate predictions and more stable representations.

Chapter 4 : we hypothesized that a multi-task artificial neural network with recurrent connections would have been an ideal candidate for tackling the type of of supervised learning problem we were aiming to solve. In particular:

- A model equipped with such characteristics would achieve a higher degree of predictive performance when compared with approaches with similar computational power and expressiveness.
- The addition of contextual information about the rewarding object and the environment in which an individuals interacted with such object would grant the model additional predictive accuracy.

Chapter 5 : we hypothesized that the representation inferred by the approach proposed in our work would have showed characteristics comparable to that of a motivational latent state. In particular:

- The ability to distinguish individuals based on the expected intensity of future interactions that they would have with a rewarding object.
- The ability to embed and distinguish multiple rewarding objects withing a unified representation.
- The ability to maintain these characteristics consistently over time.

1.4 Thesis Structure

Chapter 2 The thesis will begin with an overview on the literature on motivation and engagement highlighting how the latter can be seen as a behavioural derivative of the motivational state of an individual. Next, it will illustrate various approaches for characterizing and quantifying both engagement and motivation highlighting their strengths and weaknesses. The overview closes with a review of data-driven approaches for inferring the motivational state of individuals and predicting engagement in large scale scenarios.

Chapter 3 illustrates how theoretical and computational models derived from the field of behavioural neuroscience can be used to estimate the latent states produced by motivation. It opens by introducing the idea that latent states produced by motivational processes can be represented as a manifold on which observable behaviour resides. It continues by illustrating a computational model of incentive salience, a particular theoretical account of reward-driven motivation, and how its underlying principles can be used for defining the architecture of an artificial neural network (ANNs). The chapter closes by proposing the idea that the amount of motivational drive that an individual exhibits at a given point in time can be approximated by the manifold structured inferred by an ANN tasked to solve a particular type of supervised learning problem: predicting the intensity of future behavioural responses given the history of interactions between an individual and a potentially rewarding object (i.e., video-games).

Chapter 4 focuses on translating the theoretical insights derived from chapter 2 into a concrete model implementation. This is done through an iterative approach: starting from a simple but sub-optimal model additional components are added during three separate cycles of model specification, validation and expansion.

Chapter 5 analyzes the latent representations generated by the models defined in chapter three, and compares their functional characteristics with those of attributed incentive salience.

Chapter 6 shows a potential application of the work presented in the previous chapters in the context of automated engagement prediction and qualification in large scale scenarios, with a particular focus on applications for the videogame industry.

The thesis closes with a summary of the findings illustrated in each chapter, their limitation as well as recommendations and venues for future research.

1.5 Contribution Overview

A more detailed description of the contribution of each chapter can be found in chapter 7.

Chapter 2 the contribution of this chapter has been to provide an overview of the psychological process of motivation while creating a bridge with the construct of engagement, both from a theoretical and modelling perspective.

Chapter 3 the major contribution of this chapter has been to lay out a theoretical framework justifying the use of Artificial Neural Networks with specific characteristics for generating latent representations that well approximate the characteristics of motivational latent states.

Chapter 4 the contribution of this chapter has been to validate some of the assumptions that were proposed in chapter 3: the importance of non-linearity, recurrency, multi-tasking and global modelling for deriving representations with good predictive capabilities.

Chapter 5 the contribution of this chapter has bee to illustrate how the representations generated by our approach showed some similarities, both from the functional and behavioural point of view, with motivational latent states.

Chapter 6 the contribution of this chapter has been to sketch the prototype of a system leveraging the modelling approach presented in this thesis for large-scale automated engagement prediction.

1.6 List of Publications

Part of the work reported in Chapter 2 has been published in [4] and presented in a pre-print work in [2]. The material outlined in Chapter 3, Chapter 4 and Chapter 5 is based in small part on the work published in [4] and in large part on the manuscript published in [3]. The content of Chapter 6 has been informed by the industrial context in which this work was conducted as the PhD project was developed full-time within the analytics and data science team of our partner company Square Enix Ltd.

In all the listed publications my contribution has been in the conceptualization, methodology, software development, formal analysis, investigation, data curation, writing of the original draft and production of the visualizations.

1.7 Table of Notation, Special Symbols and Abbreviations

In this section we report a summary of the notation, abbreviations and special symbols used inside the thesis.

Table 1.1 reports the definition of common notation symbols. Additional notation specific to formulae and equations can be found in Appending A.

Table 1.1: **Table of Notation**

Symbol	Definition
θ	Generic indicator for a sub-set of parameters in a model.
Θ	Generic indicator for the collection of all the parameters in a model.
α and β	Intercept and the slope of a linear model.
w and b	Weight and bias of an Artificial Neural Network.
$\hat{.}$	Estimate for a random variable.
f	Generic function.
$f(.;\theta)$	Generic single-argument function parametrized by θ
$\approx$	Approximately equal to.
$\mapsto$	Maps to.
$\in$	Belongs to.
$\mathcal{N}$	The normal distribution.
μ	The mean of a normal distribution.
σ	The standard deviation of a normal distribution.
$\sim$	Is sampled from.
$p(.)$	Unconditional probability.
$p(.\|.)$	Conditional probability.
$\mathbb{E}[.]$	Expectation of a random variable.
T	Length of an ordered sequence.
$_t$	Index of elements in an ordered sequence.
$\{1,\ldots,N\}$ and $1:N$	The finite set ranging from 1 to N
$\{1,\ldots,T\}$ and $1:T$	The finite ordered sequence ranging from 1 to T
X	Upper case letters indicate matrices.
h, z	Latent representation inferred by a model.
$\mathbf{x}$	Bold lower-case letters indicate vectors.
X and y	The input and target of a model.
π	Policy used by an agent for selecting the optimal actions in a reinforcement learning context.
$^\top$	The transpose operation.
d	A generic distance function.
$\langle . \rangle$	A tuple of values.
γ	A discount factor.

Table 1.2 reports the definition of special symbols used in chapter 3 for describing how inferring motivation-related latent states can be casted into a reinforcement (RL)([13, 14] and supervised [3] learning problem. It is important to highlight that despite the notation might resemble in places the one used in standard RL literature [26], it shows some substantial differences (mostly motivated by the reliance on the framework proposed by McClure [13] and Zhang [14].

Finally, Table 1.3 reports the description of various abbreviations used inside this work. All abbreviations are nevertheless presented in extended form the first time are mentioned inside the thesis.

Table 1.2: **Table of Special Symbols**

Symbol	Definition
O	Object with which an individual can interact.
I	Individual who can interact with an object.
B	Frequency, intensity and ammount of behavioural responses generated during the interaction between an individual and an object.
Env	State of the environment in which the interaction between an individual and an object occours.
r	Signal produced by the reward-generating function of an object with which an individual is interacting. It has the **potential** to be rewarding for the individual.
$\hat{r}$	Transformation of the potentially rewarding signal generated by the object. It is carried out by the individual during an interaction with an object and it will determine if the interaction is perceived as rewarding.
S	State of the dynamical system underlying the interaction between an individual and an object. It can be understood as an holistic representation of the state of all the entities at play during the interaction at a specific point in time.
V	Function through which an individual is able to generate valued representation of an object while interacting with it.
G	State of the videogame object with which an individual is interacting. A videogame object indicates a specific type of O employed in the empirical work carried out in this thesis.

1.8 Core Machine Learning Concepts Overview

In this section we will provide a general and very brief overview of some of the core machine learning concepts used throughout the thesis. For a more extensive and detailed examination of the topic, we recommend consulting Murphy (2022) [27] and Bengio (2017) [28] as they have been used as sources for the current section.

Supervised Learning in the most general sense supervised learning can be seen as a set of problems within machine learning aiming to estimate the parameters (although not all supervised learning models are parametric) θ of a function f such that

$$f(X;\theta) \mapsto y \tag{1.1}$$

When $y \in \{1,\ldots,K\}$, with K being mutually exclusive labels, the problem is conventionally called classification. When $y \in \mathbb{R}$ the problem is conventionally defined as regression [27]. In probabilistic terms this equates to estimating

$$p(y|X,\theta) \tag{1.2}$$

Unsupervised Learning in the most general sense unsupervised learning can be seen as a set of problems within machine learning aiming to estimate the generative process underlying some input

Table 1.3: **Table of Abbreviations**

Abbreviation	Definition
ANN	Artificial Neural Network
MLP	Multi-Layer Perceptron
PCA	Principal Component Analysis
UMAP	Uniform Manifold Approximation and Projection
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
GAM	Generalized Additive Model
NAM	Neural Additive Model
HMM	Hidden Markov Model
MDP	Markov Decsion Process
LR	Logistic Regression
RL	Reinforcement Learning
BCE	Binary Cross-Entropy
SMAPE	Symmetric Mean Percentage Error
GLM	Generalized Linear Model
API	Application Program Interface
CPU	Central Processing Unit
GPU	Graphical Processing Unit
RAM	Random-Access Memory
VRAM	Video Random-Access Memory
LMM	Linear Mixed Model

data X [27]. In probabilistic terms this equates to estimating

$$p(X|\theta) \tag{1.3}$$

Classical examples of unsupervised learning tasks are clustering and latent factor analysis.

Reinforcement Learning in the the most general sense reinforcement learning is concerned with dynamically inferring the parameters of a policy π indicating to an artificial agent which actions a to take while interacting with an environment [27].

Usually the agent sequential decision-making can be described by a Markov Decision Process (MDP) defined by the tuple $\langle S, A, T, R\rangle$ (bare in mind these differ from the notation we use in our work). With S we define a finite set of states in which the agent can end up in after interacting with the environment [26, 29]. With A we imply the set of actions available to the agent. With T we define a function governing the transition to a new state, it can be seen in probabilistic terms as $T(s_{t+1}|a_t, s_t) \in [0,1]$ for $a \in A$ and $s \in S$. The function R provide a mapping of the type $f(a_t, s_t) \mapsto \mathbb{R}$ and $f(a_t, s_t, s_{t+1}) \mapsto \mathbb{R}$ connecting action and states to reward signals when an action or transition is performed [26, 29].

In this context the role of π is to perform the mapping $S \mapsto A$ conditional on the output of a value

function V mapping $S \mapsto \mathbb{R}$, where S here defines the state in which the agent ends up in after performing the action proposed by π [26, 29] and $\mathbb{R}$ is the expected ammount of future reward given by

$$\int_{t0}^{\infty} \gamma^t R(s_t, \pi(s_t))|s_{t-1} \tag{1.4}$$

with γ being a discount factor diminishing the impact of distant rewards. The role of RL in this setting is to solved the MDP by finding an optimal π such that V is maximized for every $s \in S$.

Linear Models are a set of methods used for estimation or predictions tasks where a vector of covariates $\mathbf{x}$ is used for predicting the expected value of a scalar (or vector) target y. These models can be used for solving supervised learning problems and extended for inferring both linear and non-linear mappings. In probabilistic terms this equates to estimating

$$p(y|\mathbf{x}, \theta) = \mathcal{N}(y|\alpha + \beta^\top \mathbf{x}, \sigma^2) \tag{1.5}$$

in particular the equation reported the standard case of a linear regression with normally distributed errors.

Clustering and Partitioning are a set of methods used for aggregating elements in a data matrix X. These can be used within an unsupervised learning context for inferring labels for each $\mathbf{x} \in X$ [27]. Clustering approaches can be distinguished based on the type of algorithm (partitioning or hierarchical) and objective (distance, similarity or density) used. This work will make use of partitioning algorithms, that as the name suggest, infer labels by dividing the data space in C mutually exclusive regions defined by a prototypical example $\mathbf{c}$ and assigning points to each region according to

$$c_i^* = \arg\min_{\mathbf{c} \in C} d(\mathbf{x_i}, \mathbf{c}) \tag{1.6}$$

in particular when $d = \|\mathbf{x_i} - c\|_2^2$ the objective function is that of the K-Means algorithm extensively used in chapter 5 [27].

Linear and non-linear Dimensionality Reduction can be understood as a form of unsupervised learning where the aim is to infer a mapping of the form

$$f(X; \theta) \mapsto z \tag{1.7}$$

with $z \ll X$. The mapping will project the input into a lower dimensional space relying on a function f which could be either parametric or non-parametric [27]. The mapping can be carried out by means of linear or non-linear transformations. The most common example of parametric linear dimensionality

reduction is Principal Component Analysis (PCA) which given the functions

$$f_{encode}(X;\theta_{encode}) = \theta_{encode}^{\top} X \qquad (1.8)$$
$$f_{decode}(z;\theta_{decode}) = \theta_{decode} z$$

will attempt to minimize the following objective

$$\|X - f_{decode}(f_{encode}(X;\theta_{encode});\theta_{decode})\|_2^2 \qquad (1.9)$$

producing lower dimension orthogonal components able to describe most of the variation in X.

Non-linear dimensionality reduction techniques rely on the idea of learning a lower dimension manifold underlying the higher dimensional data X. We will introduce this concept in chapter 2 and expand it in chapters 3 and 5).

The Uniform Manifold Approximation and Projection (UMAP) algorithm [30] used in this work is an example of a non-linear and non-parametric dimensionality reduction technique that can be used for learning the structure of a lower dimensional manifold, we will introduce this algorithm in chapter 5.

Artificial neural networks can also be used as a parametric alternative for learning lower dimensional manifolds as we will outline in chapters 3 and 5. In the particular case of unsupervised learning, this requires to parameterize the functions expressed in equations 1.8 with an appropriate ANN architecture obtaining what is called an Autoencoder model [27].

Latent Variable Models are closely connected with ANN and dimensionality reductions and can be used both in a supervised and unsupervised learning setting.

In the context of supervised learning we can re-formulate the problem of predicting a target y given a set of covariates X as

$$H \sim \mathcal{N}(\mu, \Sigma_h) \qquad (1.10)$$
$$y \sim \mathcal{N}(\alpha + \beta^{\top} + \theta^{\top} H)$$

with H being a latent variable from which we assume y is sampled from and that we bias with additional covariates X.

If we extend this to sequential data $X_{t:T}$ and $y_{t:T}$ we obtain what is defined as a State Space Model (SSM).

$$H_t = \phi^{\top} H_{t-1} + \mathcal{N}(0, \Sigma_H) \qquad (1.11)$$
$$y = \theta^{\top} H_t + \beta^{\top} X + \mathcal{N}(0, \sigma_y)$$

with H evolving over time through a transition matrix ϕ. This generic formulation can also be used for

describing dynamic latent variable models with non-continuous state space, such as Hidden Markov Models (HMM) where H is assumed to come from a multinomial distribution.

The idea behind latent variables models is that the target y is a noisy manifestation of an unobserved (often lower-dimensional) latent variable H. Artificial neural networks often follow a similar principle (that we will expand in chapters 3 and 5) where an input X is first projected onto an unobserved space H (i.e., an appropriately shaped manifold) more amenable to perform other types of downstream tasks (e.g., regression and classification) [28].

Chapter 2

Motivation and Engagement

2.1 Introduction

In this chapter we will discuss the concept of motivation and how it can be used to describe the interactions that individuals have with particular objects or activities.

We will first provide a general introduction to the construct of motivation followed by a short historical review of theories of reward-driven motivation. We will then focus on how these led to the incentive salience hypothesis formulated by Berridge and Robinson [31]. While acknowledging the contribution of other theories in the definition of motivation (and reward-driven motivation in particular) this thesis will mostly focus on the areas of behavioural and affective neuroscience and make use of the framework provided by Berridge and Robinson [31]. The reason behind this choice lies in the fact that we found incentive salience to be more specific to motivation and most importantly to have a more robust and clear connection to behaviour. The chapter will then provide a description of the concept of "engagement" and how this can be seen as a behavioural derivative of the latent states generated by motivational processes. We will then contextualise engagement in the domain of videogames presenting a quick overview of the various taxonomies describing how different videogames' structural characteristics fuel the motivational processes underlying the engagement behaviour. The chapter closes by presenting approaches for measuring, describing and predicting both engagement and motivation, with a particular focus on the challenges created by their application in situations where large-scale behavioral data are available.

2.2 Motivation as a Reward-Driven Process

The construct of motivation is a key concept for understanding why individuals seek out specific objects or experiences at particular times and why they react in particular ways when encountering objects considered of particular relevance [32]. In this view motivation can be defined as the process

leading to the modulation and reiteration of goal directed behaviours that, once reached, exert positive effects on the individual [33]. The term "positive effects" is purposely left generic here. Indeed, what constitutes a positive outcome often vary depending on individual, environmental and societal factors. What doesn't vary (or it does so to a lesser extent) are the processes underlying the generating and processing of such effects. In order to be more easily generalizable across individuals, a formal definition of motivation should encompass both the behavioural and the underlying psychobiological level. We will focus first on behavioural aspects of the process and outline in the following sections a theoretical account of motivation which also covers the psychobiological level.

As we mentioned before, motivation describes why individuals react in particular ways when encountering objects regarded of high relevance and why they approach those objects at particular times [32]. These type of objects are said to possess "rewarding properties", which can be defined as the positive value ascribed to an object, a behavioral act or an internal physical state as the result of an active process of the mind and the brain [34, 35]. Importantly, motivation is not just driven by the fulfilment of fundamental needs like nutrition or reproduction (so-called "primary reward objects" [36]) or the avoidance of negative consequences like physical pain. It also extends to those volitional objects and activities which do not appear to be necessary for the survival of the individual (i.e., "secondary reward objects" [35, 37]). For those activities, the expectation of the amount of reward received is learned over time and may vary significantly between individuals [33, 35].

In spite of this, it would be inefficient to have dedicated and specialized motivational systems for every combination of individuals and objects (e.g., an individual's motivation for playing sport or eating food). Instead, we can think of motivation as a single overarching entity that controls the interaction between individuals and objects [33]. An analogy may be drawn with the geometric concept of a vector. Looking at Figure 2.1 we can imagine the focus on a specific object being represented by the angle of the vector while its length is the intensity of the motivational process (or the amount of motivated behaviour) [33]. This, can be thought as a dynamic quantity defined by the state of the individual and the rewarding properties of the object [14, 32, 38].

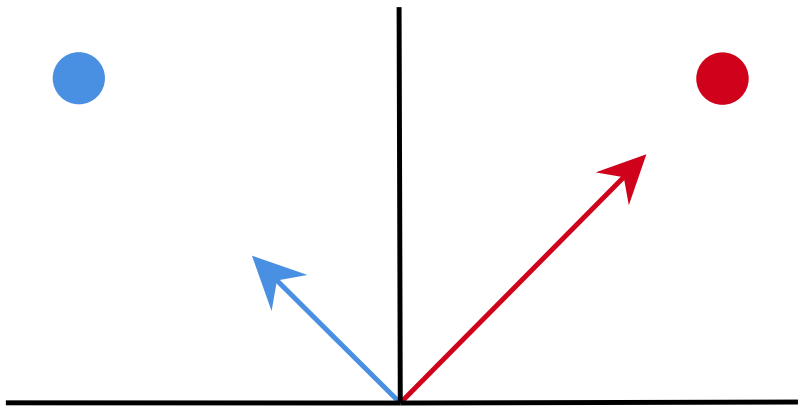


Figure 2.1: Blue and red dots represent two objects with different characteristics while the two arrows illustrate the hypothetical motivational propensity of an individual (or two individuals) towards them. The black segments delineate the space created by the combination of the objects' characteristics and the motivational propensity of the individuals. Here the red object has the potential to generate more behaviour than the blue object possibly as a result of its characteristics and those of the individual interacting with it.

In summary, we can say that from a motivational point of view, the behaviour of an individual is driven by the expectancy of pleasurable outcomes from the goal the behaviour is aiming to reach [32]. Therefore, if motivation acts as a single overarching process, we expect it to hold predictive and explanatory power over goal directed behaviour seamlessly across a heterogeneous range of situations and individuals. Motivational theories based on the concepts of reward and incentive are promising candidates for this because, relying on consistent and plausible psychobiological bases, they tend to operate at a level that is abstracted from the precise nature of the individuals and the objects. [31, 32, 39–42].

2.2.1 An Historical View of Reward-driven Motivation

Introducing the processes of classical and operant conditioning is an essential step for describing theories of reward-driven motivation, especially if we are interested in their behavioural correlates. Both constructs heavily rely on the general concepts of a "reinforcer" and "reinforcement process".

Reinforcers are objects or actions that have the ability to alter the likelihood of appearance of specific behaviours [43–45]. A reinforcement *process* instead defines the learning mechanisms by which a specific behaviour becomes, over time, more or less probable conditional on the presence of particular reinforcers [43].

In this view we can think of classical and operant conditioning as two complementary operationalizations of the reinforcement process. Classical conditioning describes the learning process in which, independently from the activity of an individual, the repeated pairing of two objects will cause one to acquire the eliciting properties of the other [45]. In other words, the repeated pairing of a neutral object with reinforcing consequences will imbue the first with reinforcement properties making it a reinforcer. Operant conditioning on the other hand, extends the concept of classical conditioning introducing the agency of the individual [44]. The frequency of behaviour produced by an individual tends to increase when precise consequences are associated to it [44]. Hence, an operant can be formalized as a goal directed behaviour while all the elements reinforcing the re-iteration of this behaviour are called reinforcers [44]. The learning process here results from the relationship between a behaviour and its consequences, therefore the probability that a behaviour will take place is related to its capability to generate reinforcement. [43].

Both classical and operant conditioning are somewhat simplistic accounts of human behaviour, but they are, nevertheless able to succinctly illustrate a fundamental process by which most (if not all) individuals are able to learn and leverage the association between actions and the positive (i.e. rewarding) consequences associated to them [33, 34]. In this regard, it is not surprising that many theories of reward-driven motivation stem directly from these two constructs.

For example, Bolles [46] suggested that individuals were motivated by the "expectations of incentive outcomes". These expectations are formed through a learning process where an association between actions and potential pleasurable outcomes is created [32, 46]. Expanding on this idea, Bindra [47]

suggested that the learning process does not just generate pleasure expectations in response to specific behaviours but it also allows individuals to perceive the behaviours themselves as a source of hedonic reward [32, 47]. This introduces the concept of learning through reinforcement: an object and the behaviours associated to it become relevant and salient for an individual as a consequence of learning its incentive properties [32]. A third theoretical formulation by Toates [38], asserted that the magnitude of the perceived incentives introduced by both Bolles and Bindra is modulated by the internal states of the individual [32, 38]. In other words, the incentive expectations (and consequently the associated motivated behaviours) learned by an individual can change over time depending on the individual's internal state.

Until now we have mainly used the terms "reinforcer" and "incentive" to identify objects able to drive and shape behaviour, but when it comes to define effective reinforcers, it is not sufficient to merely pair a behaviour with a stimulus. The stimulus itself must have particular properties. In this view, stimuli that can generate pleasurable feelings in the individual are the best candidates for being effective reinforcers, and they are said having 'rewarding properties' [32]. So what is, and how can we define a reward? We can think of a reward as a process of pleasureable responses generated in response to a stimulus. In this view, to generate rewarding response, a stimulus needs two fundamental properties: it has to be *wanted* (i.e., it acquires the capacity to become desirable) and *liked* (i.e., it has to be able to generate pleasure in the individual) [48]. But how does a particular object acquires these properties? This is mostly a function of the same learning processes mentioned in section 2.2.1. The repeated pairing of a stimulus with the (positive) consequences it exerts on the individual will imbue the first with so called rewarding properties. Moreover, through operant conditioning not just the stimulus itself but also the connected instrumental behaviour will likely acquire the same rewarding properties [48].

As anticipated in section 2.2, a useful distinction that can be made is between objects having primary and secondary reward properties. Objects linked with essential evolutionary needs (i.e., satisfaction of homeostatic needs) are on a fast track for becoming reinforcers, their rewarding properties do not have to be learned but are, up to a certain extent, intrinsic to them [37]. Classical examples of primary rewards are food, mating-related activities and drug of abuse [32, 33]. On the other hand, objects with so called secondary rewarding properties do not hold an innate capacity to generate pleasurable experiences, but they acquire it by means of the same learning mechanisms we have just presented [37].

2.2.2 The Incentive Salience Theory of Motivation

The approaches proposed by Bolles, Bindra and Toates, provide an account of reward-based motivation butthey do not draw a distinction between the affective dimension of an incentive (i.e., how pleasurable it is) and the purely motivational aspect of it (i.e., how much goal directed behaviour it can produce) [38, 47].

However, Berridge and Robinson proposed that the motivational process controlling the interaction

between individuals and objects might not be a unitary mechanism but rather a composite process having specific and dissociable components which rely on specialized neurobiological mechanisms, namely: *liking*, *wanting* and *learning* [31, 48, 49].

Liking The *liking* component describes the pleasure expected by an individual when interacting with an object [48]. It is responsible for the hedonic quality of an experience and acts as a signal indicating that interacting again with that object might be beneficial. Despite the fact that *liking* plays an important role in the incentive salience hypothesis of motivation it is difficult to measure it outside controlled laboratory environments [31] and it will not form a central theme of this thesis. Instead, we will focus on the "wanting" and "learning" components.

Wanting The *wanting* component, or "incentive salience", has the function of generating and holding latent representation of objects and behavioural acts and of attributing value to them through learning mechanisms. These "valued representations" can then be used by action selection systems in order to make certain behaviours more likely [13, 31, 32, 50]. As a consequence of this, when an object is attributed with incentive salience it is more likely to draw the subject's attention and become the focus of goal directed behaviours [32]. Interestingly, *wanting* seems to be more than a simple form of value-caching but rather a dynamic process in constant change [14, 51–53]. This is because the saliency of an object depends both on its attributed value but also on the state of the individual interacting with it. A change in the individual's internal state can dampen, magnify or even revert the amount of attributed salience. [14, 51–53]. It is important to note that *wanting* is not the hedonic expectation associated to an object, (which is designated by *liking*), but rather the process promoting the approach towards an object and the interaction with it [48, 54]. Despite the fact that *liking* and *wanting* are often correlated (i.e. I want what I like and vice versa) they can be dissociated. Addictive behaviours for instance are a notable example of *wanting* without *liking* [51]. The functional dissociation between these two components is linked to differences in the underlying neurobiological substrate [48, 49]. Neurotransmitters and brain areas responsible for *wanting* appear to be more numerous, diverse and easily activated than those for *liking* [48, 54]. As a consequence, increased incentive salience can be obtained by raising dopamine levels in many portion of the striatum without the need for the synchronized activity in other areas [48, 49, 55]. This implies that the *wanting* component tends to produce more robust behavioural indicators in the form of increased amount and frequency of interactions between an individual and an object [31], which makes it a promising candidate for behavioural studies in conditions where strict experimental control is not possible.

Learning The last component in the formulation proposed by Berridge and Robinson [31, 32] consists of mechanisms that provide an individual with the capability to predict, based on past experiences, the occurrence of future pleasurable outcomes (i.e., *liking* reactions) when interacting with specific objects. These are similar to the learning processes illustrated in section 2.2.1 and have a twofold function. These mechanisms allow the attribution and change of incentive salience properties to previously *liked* objects (e.g., primary reward objects) but they also enable subjects to

learn the hedonic value of initially neutral stimuli (e.g., secondary reward objects). The *learning* mechanism is based on classical conditioning: through repeated interactions with an object an individual will learn its hedonic properties and consequently attribute incentive salience to it [32, 48]. This process is driven by mechanisms similar to those of reward-prediction error: learning is driven by increases in dopaminergic activity generated by a mismatch between expected and experienced rewards. [34, 56, 57].

In this view, motivation can be understood as arising by the combination of these three components [31, 32, 35] which provides the affective, behavioural and cognitive substrates for the process. Nevertheless, it is worth noticing that given their dissociable nature (see Berridge et al. [48]) it is possible to investigate them in separation (see the computational work carried out by McClure at al. [13] and Zhang et al. [14]) by making sure to respect the appropriate theoretical constrains (we will expand on this in chapter 3).

2.3 Engagement as a Derivative of Motivation

We will now momentarily diverge from our discussion on motivation in order to introduce the construct of engagement. The reason for this brief detour lies in the fact that presenting the construct of engagement allows us to better understand the practical implication that motivational processes have in our everyday life - and in particular in the context of videogames. Despite the fact that engagement has applications in a wide range of contexts, we think it is better understood when framed within a specific class of activities (the reason for this will emerge in the following sections). In this view, given the background from which this work has arisen, we will focus on the area of videogames but, by framing engagement as a byproduct of motivational processes, we will show that we can generalize it to other type of activities.

2.3.1 Theories of Engagement

Humans have played games throughout history to entertain themselves and relax [58]. Gameplay can be defined as a free-time activity with spatial and temporal boundaries able to intensely absorb someone who is involved in it [58].

A special case of the broader group of games are those delivered and experienced in a digital format (i.e., videogames) which have been substituting for more traditional playful activities over the past few decades [58, 59]. This has been reflected both in terms of number of people involved in playing videogames as well as in the amount of time spent engaging in this activity [59, 60]. One of the main reasons for this explosive phenomenon is the fact that videogames seems to be perfect medium for delivering pleasurable experiences [59], consequently holding a strong potential to engage and retain users.

Various attempts have been made to understand engagement in videogames, both as a unitary process and at the level of factors driving and influencing it, across individuals, communities and societies [59].

The literature on the subject while, abundant is extremely heterogeneous [59], with no clear consensus even around key terms and definitions. A clear example of this heterogeneity is the definition of engagement provided by O'Brien et al. [61]:

"...a quality of user experiences with technology that is characterized by challenge, aesthetic and sensory appeal, feedback, novelty, interactivity, perceived control and time, awareness, motivation, interest and affect..."

this definition, although providing a good holistic description, makes it exceptionally hard to define a clear, unitary framework for defining and measuring engagement let alone specify its mechanistic aspects. This lack of theoretical formalism is reflected in most (if not all) accounts of engagement and, to a certain extent, is inevitable given the breadth of the behavioural, cognitive and affective aspects that the construct tries to cover. This can, in part, be understood and justified by the strong inter-disciplinarity characterising the research on videogames engagement. Indeed it is common to find contributions to the subject within the humanities, natural and technical science and management and business domains. Each field relies on their own terminology and weltanschauungs. Nevertheless, inspecting the most prominent theories associated with the concept of engagement, we can identify some common themes that are useful for composing a unified framework.

Flow This is a classical construct often occurring in the videogame literature for explaining the phenomenon of engagement. Developed by Csikszentmihalyi [62], the construct of flow prescribes that when an individual is absorbed in an activity perceived as valuable they will experience a rewarding state of optimal pleasure constituting the "fuel" for the engagement process [59]. In this view, the *condicio sine qua non* for the flow state to arise is a balanced combination of the individual's state level and the characteristics of the interaction in which they are involved [59, 62]. Despite offering an interesting point of view, the concept of flow as a framework for explaining engagement might be prone to the fallacy of circular reasoning: is an individual engaged in a specific activity because this provides the optimal flow experience or is "flow" a byproduct of being engaged in the activity itself?

Immersion This construct is closely connected to that of flow but specifically concerned with the psychological experience of engaging with a computer game [63]. Immersion tries to describe the experience of engaging in a specific moment in time with a videogame rather than positing itself as a factor influencing or driving engagement [63]. According to Jennet *et al*. [63], as a result of a good gaming experience, an individual might lose track of time and space and will experience a sense of being completely "immersed" in the game.

Uses and gratification This theory states that the consumption of media is a way for individuals to satisfy the need for gratification (i.e. reward). The gratification-seeking behaviour is then driven and characterized by the nature of the underlying motive generating it (e.g., my need to connect with other people drives my social-media consumption) [64]. Again, according to this theory, individuals are not

passive bystanders but will actively interact with a specific media object based on its ability to meet the individual's underlying motivational drive. Uses and gratification theory introduce two important concepts: first that individuals engage in a spontaneous activity (i.e. media object interaction) in search of some form of gratification and second that this interaction is not passive but rather an active process.

2.3.1.1 The Engagement Process Model

We will focus now on a relatively atypical formalization of the construct of engagement, the engagement process model formulated by O'Brien and colleagues [61].

This framework, instead of presenting engagement as a static entity, proposes the idea that it is better understood as a dynamic process [61]. O'Brien *et al* avoid giving an exact definition of engagement but rather describe it as a process with distinct phases each one possessing "peculiar attributes" [61]:

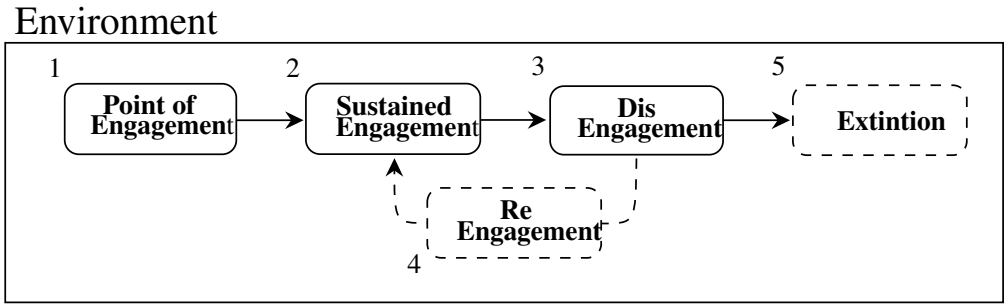


Figure 2.2: Solid and dashed lines represent compulsory and optional paths. Moving from the left to the right we can imagine 1 as being the first interaction that an individual has with an object. This might happen as the result of a prior belief that the object is able to provide a pleasurable experience. The individual will interact with the object for as long as this last one is able to provide a gratifying experience(i.e. stage 2). However, if this is not the case, or constraints from the surrounding environment emerge, the individual will gradually reduce their interaction with the object (i.e. stage 3). At this point we can either observe an alternation between re-engagement and disengagement (i.e. stage 4) or reach an inevitable state of complete withdrawal from the object (i.e. stage 5)

Point of engagement This is the starting point of the engagement process. It is the moment in which the individual's attention is directed towards a specific object or activity due its properties and capacity to fulfil specific motivational drives.

Period of engagement This is the period during which the individual has a sustained interaction with the object of interest. In this case, a situation of sustained and prolonged interaction is conditional on the ability of the object to provide a positive and stimulating experience.

Disengagement This stage defines the moment in which the individual reduces the interaction with the object due to internal or external factors. The internal factors are usually connected to loss of interest or a lack of time. External factors instead relate more to the inability of the activity to provide a positive experience or to the occurrence of external events in the environment surrounding the individual.

Re-engagement It identifies the moment in which the user returns to a sustained level of activity after disengagement occurred. This can happen both in the short and long term and it is the result of positive experiences with the activity, which are usually linked to rewarding incentives or novel content within the activity.

Extinction In the case of prolonged disengagement, marked unsatisfying experiences or impactful external events, the individual might terminate their interactions with the object leaving no further possibility to re-engage with it.

2.3.2 From Motivation to Engagement

From this brief overview of the literature on engagement theories applied to videogames, we show that engagement is best described as a process controlled by 1) the characteristics of an object, 2) the internal state of the individual interacting with it and 3) the environmental factors external to both. In this view engagement is a second order factor generated and derived from the internal state of the individual and concerned with the description and quantification of their interactions with an object [58, 59, 61–64]. The quality and quantity of these interactions seem to depend on the ability of the object to provide feelings of enjoyment and pleasure [58, 59, 61–64]. For this reason engagement is better understood when framed within a specific context: despite the fact that it is generated by processes that are common across any human being (e.g., motivation) its specific connotation is tightly bound to the characteristics of the object of engagement. In other words, engagement can be better understood when framed as engagement *with* something rather than as an abstract entity.

We can already see a connection between the construct of engagement and the reward-driven motivational processes presented in section 2.2, especially if we look at the behavioural level. In the context of videogames, motivational processes seem to pertain to the formation and modulation of unobservable (i.e. latent) states characterizing the individual before, during and after the interaction with a videogame. On the other hand, engagement appears to describe the observable aspects of this interaction both at the behavioural and experiential level [58, 59, 61–64]. A clearer illustration of this idea is presented in Figure 2.3

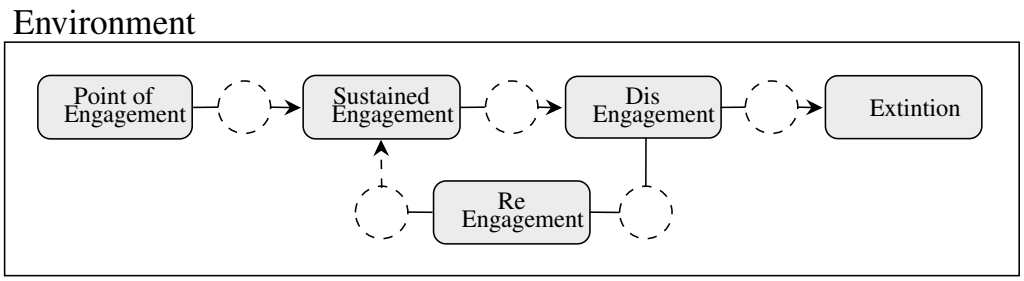


Figure 2.3: Solid and dashed lines represent observable and unobservable element in the process. The description of the process is the same as described in Figure 2.2 but the transition between observable engagement stages is controlled by a latent variable defining the motivational propensity of the individual towards the object they are engaging with.

Here we adapted the engagement process model of O'Brien and colleagues [61] presented in Figure

2.2 incorporating the state of the individual. In this view, the motivational states of an individual determine which phase of the engagement process an individual will be located in when interacting with a specific object (i.e. a videogame). This implies that, at any point in time, the history of observed engagement indicators (e.g. frequency and amount of interactions) can provide information about the unobservable motivational state of the individual.

In section 2.2.1 we specified that the motivational propensity that an individual might have towards a certain object is in part determined by the rewarding properties of the object itself [32]. But how would these properties look in the context of a videogame? Surely playing videogames is not relevant for satisfying fundamental physiological needs nor it is directly linked to other type of powerful reinforcers (e.g. money). In this case, the distinction between primary and secondary rewards made in section 2.2.1 becomes particularly useful for understating the framework in which videogames lie: the structural characteristics defining the videogame itself act as reinforcers during the playing behaviour [65–67]. We will better define the concept of videogame structural characteristics later on but for now we can think of them as any type of in-game element that an individual might interact with during play [65, 66]. None of these structural characteristics are intrinsically rewarding, but they can become so over time through conventional learning processes if they are pleasurable [32, 44, 68]. In a dynamic fashion, interacting with specific features in the videogame might produce positive reactions in the individual and make new interactions more probable.

In this view engagement can be seen as the observable manifestation of the unobservable motivational propensity that an individual has towards a specific videogame. We can think of engagement as the behavioural realisation of a motivational process aimed at maximizing the positive experiences provided by the in-game elements.

2.3.3 Videogames Structural Characteristics

It should be evident by now that the ability to construct videogames with effective rewarding characteristics is of pivotal importance for generating and sustaining engagement [65–67].

Various attempt has been made to build a taxonomy of these characteristics, King et al. [66] for instance outlined a series of features that effective videogames structural characteristics should have. These are: social features, manipulation (e.g. crafting) features, narrative features, achievement and punishing features and aesthetic features. Westwood *et al* [69] were able to use this taxonomy to identify the specific structural characteristics driving playing behaviour in a group of videogame players. The taxonomy provided by King et al. [66], although exhaustive, provides a descriptive rather than a mechanistic account of the structural characteristic driving playing behaviour [65]. Adopting a different approach, Wang and Sun framed the work of King et al. in terms of reinforcers [66, 70]. In this view, playing behaviour in different games is sustained by different reinforcing mechanics covering the areas described by King et al. [66,70]. Among these mechanics there are: scoring systems, in-game items and resources, achievements systems, feedback messages, animation events and the unlocking of new game contents [70]. The authors also described a series of attributes that in-game

rewards require to be effective: they need to possess social value within a shared environment, they need to have visible effects within the game world and they need time and effort to be obtained [70]. Extending the work of Wang and Sun, Philips and colleagues focused on a description of the temporal characteristics that in-game reinforcers may possess, namely their duration, consumability and context dependency [71].

A connected although separate stream of research tried to categorize which elements inside a videogame might produce pleasurable experiences for the players however focusing more on the characteristics of the individuals rather than those of the game itself. Similarly to trait theories in psychology, this approach assumed that different individuals have static, consistent and well defined preferences for some aspects (i.e. structural characteristics). of the playing experience. In his seminal work Bartle tried to identify different approaches that players might have had in playing Multi User Dungeons (MUDs), an early version of the modern Massively Multiplayer Online Role Playing Games (MMORPGs) [72]. Projecting the players' attitudes towards the game on two axis: oriented towards action or interaction and oriented towards the game world or the players, Bartle proposed four mutually exclusive categories [72].

Table 2.1: **Bartle Taxonomy**

Category	Description
Achievers	Action oriented towards the game world. Players interested in mastering the game, aiming to build a status within the game towards their interaction with the environment.
Socializers	Interaction with other players. Players driven by the perspective of interacting with other players, deriving satisfaction from their friendship, contacts and social influence within the game
Explorers	Interaction with the game world. Players aiming to be surprised by the game world, seeking the stimulation derived by the discovery of new areas and the acquisition of knowledge.
Killers	Action oriented towards other players. Players interested in demonstrating their superiority over other players posing great value on the reputation obtained through in-game fighting skills.

Despite the fact that this early formulation only considers a specific subset of games and lacks any kind of empirical validation, Bartle's work was the starting point for most of the later efforts on the subject [72]. Bartle's taxonomy was built on assumptions that were never tested and the proposed categories showed a certain degree of inter-correlation. For this reason Yee tried to develop a methodology for assessing the players' preferences for specific game characteristics [73]. After gathering information from a large sample of players and performing dimensionality reduction, 10 major factors emerged, these were then condensed into 3 facets in a second round of dimensionality reduction [73]. The three main factors that emerged were:

The model proposed by Yee introduced two interesting variations on Bartle's work: the components are not necessarily mutually exclusive and the focus is shifted from a characterization of the players

Table 2.2: **Yee Taxonomy**

Facet	Description
Achievement	Factor indicating a tendency towards advancing in the game, exploiting its mechanics or competing with it.
Social	Factor indicating a preference for those game characteristics centred on socializing, creating relationship and developing teamwork.
Immersion	Factor represents the drive towards the discovery, role-playing and customization mechanics of the game.

to a characterization of the in-game elements with which the individuals interact [72, 73]. Despite these improvements, the focus was still on a specific game genre and heavily relied on static and potentially biased measures (i.e. self-report). In order to overcome the limitations of a taxonomy heavily influenced by a specific game genre, Nacke et al. developed a more comprehensive system with the intent of capturing the players' preferences for particular game mechanics regardless of the genre [74]. The categories proposed by this model were:

Table 2.3: **BrainHex Taxonomy**

Category	Description
Seekers	Players driven by in game mechanics which produces interest and satisfy curiosity..
Daredevils	Players motivated by the thrill derived from taking risks in the game.
Masterminds	Players who enjoys solving puzzles and finding the best strategies to adopt within the game.
Conquerors	Players who derive satisfaction from overcoming the challenges provided by the game and from the struggles characterizing the process.
Socializers	Players driven by the interaction with other players.
Achievers	Players driven by reaching long term goals, often aiming to fully complete the game.
Survivors	Players driven by thrilling experiences provided by the game and by the ability of the game environment to generate arousal.

This formulation appear to have a higher level of details than the approaches of Yee and Bartle, however the number of considered factors grows proportionally to the number of game characteristics taken into consideration [74]. We will not expand on the supposed link between facets, "neurobiology" and personality components proposed by the authors. The first is purely anecdotal, not supported by any evidence and adopt a conceptualization of the human brain that is, at the very least, primitive (reminiscent of the phrenologist Franz Joseph Gall [75]) [74]. Evaluating the relationship between the facets and personality traits could have been an interesting angle to explore, but the use of the infamous and largely discredited Myers-Briggs model of personality [76] makes the results hard to interpret.

Differently from most taxonomies presented so far, which are specific to the videogame literature, the work of Przybylski and colleagues leverage the self determination theory of Rayan and Deci [77, 78], a socio-psychological theory of motivation, to explain the process through which videogames drive

sustained engagement [68] in an empirical way. The original work by Ryan and Deci states that humans are driven in their every-day life by the satisfaction of basic needs for competence, autonomy, relatedness and control which are more effective motivational factors than any kind of external incentive [77, 78]. These basic needs are radically different from the physiological one presented in section 2.2. On the other hand, the external incentives mentioned by the theory are closely related to the concept of reinforcers presented in section 2.2.1. Ryan and Deci make a distinction between intrinsically motivated behaviours (behaviours motivated by the satisfaction of one of the aforementioned fundamental needs) and extrinsically motivated ones (behaviours motivated by reinforcers coming from the environment like money). In this view, Przybylski and colleagues proposed that videogames, lacking the presence of external reinforcers, provide appeal via the inherent properties of the playing experience [68]. If a videogame is able, through its structural characteristics, to satisfy the individual in one or more of the four fundamental dimension mentioned before it will also be able to promote a state of sustained engagement [68].

Table 2.4: **Self-Determination Taxonomy**

Facet	Description
Competence	The satisfaction of this need can be provided by the optimal balance between game difficulty and player skill. The player should never be bored or overwhelmed by the game instead should feel competent while playing.
Autonomy	The satisfaction of this need can be provided allowing the player to advance through different challenges and shape the game world in accordance to their will, allowing them to act with freedom within the game.
Relatedness	The satisfaction of this need can be achieved though social interactions within the game.
Control	The satisfaction of this need can be achieved allowing the player to master the controls of the game putting them in the position of experiencing a sense of control and receiving appropriated feedbacks from the game.

One of the core idea behind the work of Przybylski et al. is that when the playing activity is focused on obtaining reinforcers and avoiding punishments it fosters extrinsic motivation, with potential negative effect on engagement [68].

Despite the effort made for creating a connection between psychology and the literature on videogames, we belive that the work of Przybylski et al. cannot be reconciled with a long experimental tradition showing the pivotal role of reinforcers in shaping and driving motivation at the behavioural, cognitive and affective level [32, 34, 44]. Moreover, one clear limitation of the Ryan and Deci theory for understanding the motivational drive of videogames is the staggering success of titles heavily and explicitly relying on reinforcement and punishment mechanisms [79, 80]. We will not go into the merit of arguing whether self determination theory is an appropriate approximation of motivational processes in humans, but we believe that its application to the videogame context seems to be more appropriate for describing issues related to usability and only tangentially relevant for describing the motivational drive of videogames.

In summary we could say that each videogame can be considered as an object with particular structural characteristics (i.e., in-game features) and rules (i.e., games mechanics). Individuals purposely decide to interact with these objects without any (in most cases) requirement from the external world (again, playing games in most cases is not necessary for our survival nor is it mandated by any societal norm). What drives these interactions are solely the characteristics of the game. If they provide a pleasurable experience in a particular point in time, these characteristics will promote more interactions between the individual and the game object.

We have seen how there is no real consensus in the literature over a specific taxonomy describing the structural characteristics underlying the motivational drive offered by videogames. However some common themes seem to emerge. A structural characteristic acts as a reinforcer if it is able to generate positive reactions in the player. At any given point in time, the game feature with which an individual interacts the most can be seen as a promising candidate for the role of motivational driver (i.e., reinforcer). A common set of characteristics and facets that could act as reinforcers seems to emerge across the works presented so far, namely: achievement, socialization, exploration and fighting. However, this might be an artefact created by the fact that most taxonomies appear to have the work of Bartle as a common ancestor [72–74].

In conclusion, we can say that the lack of a unified theory defining how videogames produce motivated behaviour (engagement) might be the result of various factors. One, is the historical preference for holistic descriptions of the experiential aspects of engagement rather than its mechanistic functioning. The second is a strong focus on producing (relatively game specific) taxonomies of videogames structural characteristics rather than a general theory deriving how these contribute to control engagement. Finally, there has been a disconnect between theories of engagement in a videogame setting and robust psychological and neuroscientific constructs (e.g. motivation). We believe that trying to understand engagement through the lens of well established theories of motivation will provide us with a unifying framework for making sense of the heterogeneous evidence that we presented so far.

2.4 Measuring Engagement and Motivation

In the previous section we proposed that engagement is the observable realization of latent states generated by the psychological processes that control the interactions between individuals and videogames. This implies that unlike constructs like motivational states it should be easier to derive quantitative and qualitative measures of the amount and direction of engagement. We describe here three different approaches, with their relative strengths and weaknesses, for the measurement of engagement (in particular withing a videogame setting).

2.4.1 Self-report Measures

These measures include all the techniques that require individuals to report their experiences and personal, psychological or demographic characteristics usually with the aim of measuring static/stable

attributes.

Most of the time the measurement is carried out through questionnaires constructed to reliably measure a common construct. Examples used for gathering measures of motivational propensity are the BIS-BAS scale [81], focusing on assessing the responsiveness to incentives, or the many different questionnaires developed for measuring the constructs of intrinsic and extrinsic motivation proposed by Ryan and Deci [77].

The advantages provided by these measures are their relative simplicity, ease of use, scalability and possibility of investigating large sets of constructs simultaneously.

However, questionnaires often require the cognitive introspection of actions, emotions or attitudes. This is an important component for a questionnaire to be effective, nevertheless individuals are not always aware of (or able to retrieve and precisely describe) the motives behind a set of actions or emotional responses [82]. Indeed we must stress that conscious appraisal and actual experience may interact and overlap but often do not coincide [83].

This is supported by the fact that despite many attempts to examine associations between in-game behaviour and self-report measures, the findings have often been fragmented, inconsistent or inconclusive [84–86]. For example, in a work by Van Lankveld and colleagues [87] the authors investigated the relationship between a large number of behavioural metrics (derived from the interaction of 44 participants with a game) and the score to the NEO-PI-R (a questionnaire for the evaluation of the five-factor model of personality) [88]. The results highlighted multiple, but nonspecific, correlational patterns. By nonspecific we mean that they were not justified by a-priori hypotheses. This can be problematic when a large number of correlation indices are computed as it becomes challenging to distinguish between legitimate and spurious relationships. Following this approach, Canossa and colleagues [84] also tried to investigate the relationship between self-report measures of psychological characteristics and gaming behaviour however in a larger sample. The results were similar to the work of Van Lankveld et al. in the sense that in-game behaviours appeared to relate with various psychological traits but the underlying meaning of this relationships was hard to derive. In another work by Lankveld et al. the authors narrowed the focus to a single personality trait (extraversion) and specifically designed a short game for retrieving behavioural metrics related to that construct. Although promising, the results can be regarded as borderline inconclusive (i.e., of the 26 in-games metrics considered only 5 were significantly related to the construct of extraversion) [89].

One reason for these unsatisfactory results might be the nature of the questionnaire: conventional psychometric measures have not been developed for describing an individual's behaviour withing a digital setting [67]. Ad-hoc questionnaires have been developed for measuring game specific constructs [73, 90], but unlike their psychological counterparts they often have little validation. A series of other limitations can be found in the adoption of self report measures, namely: the adherence of the respondents to social desirability norms, erroneous interpretation of the questions, untruthful answers, constraints in the possible answers, random or systematic measurement errors and in case of

mass administrations (i.e., mailing or online recruiting) poor sampling control [87].

2.4.2 Psychophysiological Measures

As we mentioned in section 2.4.1, self-report methods aim to measure a latent construct, be it a trait or a state, through a series of questions. This quantification is by definition static and, as we mentioned before, prone to bias or systematic measurement errors [87].

On the opposite side of the spectrum we can find measures acquired through the recording of the physiological responses of an individual. These approaches are based on the assumption that particular physiological responses from the body are linked to affective and cognitive processes and can therefore be used as proxy measures for the underlying psychological process (i.e., psychophysiological measures) [91]. These kind of indices can be derived through various techniques ranging from more basic and generic (e.g. skin conductance, the electrocardiogram (ECG)) to more sophisticated and specific ones (e.g. the electroencephalogram (EEG), dunctional Magnetic Resonance Imaging (fMRI)).

Gathering psychophysiological measures in a videogame context is motivated by the idea that events inside the game can trigger specific psychological processes, and these can be inferred by measuring the body's physiological response [67, 92]. Monitoring these alterations during a game session can help in reconstructing the player's experience [93] as well as producing richer player profiles and models [67].

Typical indices used for assessing the motivational propensity of individuals range from more specific (e.g., the late positive potential in EEG or the contingent negative variation of blood oxygenation in the dopaminergic pathways [91]) to more generic ones (e.g. skin conductance responses or variations in pupil dilation [91]). Unlike self-report measures, psychophysiological indices can be very precise, are by nature dynamic measures and robustly reflect the activity of various cognitive and affective processes [91]. Moreover, they tend to be less prone to bias generated by the individual as they do not require an overt cognitive appraisal. That said, their adoption might be hindered by a series of limitations: they are often perceived as invasive by the individual [67], depending on the hardware used, they might be expensive, they require particular care in the recording phase, they are prone to artefacts, the signal pre-processing is often a long and laborious activity, their interpretability may be difficult if *a priori* hypotheses are not formulated, they often require careful and sometimes sparse experimental designs for controlling confounds and investigating only variables of interest [94].

2.4.3 Behavioural Measures

In between the two extremes defined by self-report and psychophysiological measures, we can find indices derived from the behavioural responses of the individual. These measures can account for external manifestations of some of the individual's cognitive and affective processes in a more objective, continuous and naturalistic manner than questionnaires but at the same time lack the ability

of pyschophysiological measures to precisely capture the dynamics of these processes.

Typical example of behavioural indices used for quantifying the motivational state of an individual are measures of the frequency, amount and duration of approach to an object or situation of interest [32,33]. In experimental settings these measure are usually acquired by keeping track of the actions performed by the individual during a specific task [33,48]. In a videogames context this is carried out by telemetry systems [24]. These systems gather, at a high frequency, large and heterogeneous records of the player behaviour inside the game world [24]. Given the context in which these measures are acquired, they behaves similarly to psychophysiological measures in terms of bias reduction while retaining a greater degree of ecological validity and, most importantly, scalability [24]. The availability of such measures seems particularly appealing for deriving more faithful measures of the motivational drives underlying the engagement in videogame play. They also overcome some of the limitations we highlighted at the end of section 2.3.

Challenges from Large Scale Behavioural Measures Although appealing, the use of large volumes of behavioural data acquired through ecologically robust but uncontrolled methods (i.e., telemetry) has limitations. These include the complexity of the data, the lack of strict experimental control (which can worsen the noise to signal ratio and introduce bias) and, paradoxically, the very high statistical power associated with the large number of observations [95]. These issues in combination with the availability of a high number of "researcher degrees of freedom" [96] make the use of conventional statistical testing procedures problematic. Instead, predictive or inferential modelling approaches provide a more flexible framework where the lack of experimental control is counterbalanced by the ablility to model individual player in a more holistic manner and leverage the complexity of the data in a more principled way [67].

2.5 Estimating Motivation and Predicting Engagement from Large Scale Behavioural Measures

Before diving into potential approaches for modelling the state of an individual from large scale behavioural metrics (i.e., videogames telemetry) a distinction between profiling and modelling needs to be drawn.

Profiling mostly pertains to a static description of latent or observable characteristics of an individual that supposedly does not change during the gameplay [67] (although noticeable exceptions can be found in [5, 97, 98]). Classical examples of profiling include the work of Bartle, Yee and Nacke presented in section 2.3.1 [72–74]. A profiling approach aims to define a restricted number of categories into which players can be divided, the characteristics of each category should be able to describe the player behaviour in a wide range of situations [67, 87, 89]. One way to relax this requirement is to remove the constraints on the number or type of categories and derive these directly from the data (Yee used a similar approach for defining the factors in its questionnaires [73]), Drachen

and colleagues pioneered this approach in the context of videogames by applying unsupervised machine learning algorithms (i.e. partitioning and clustering) to telemetry data [99–101].

Modelling can also be seen as the realization of a computational description of the individual behaviour within the game environment [67]. A modelling approach aims to predict the player's experience through the evaluation of cognitive, affective and behavioural patterns arising from the dynamic interaction of player and environment during gameplay [67]. In the context of modelling a further distinction in approach must be made:

Model based approach (top down) Following this approach, a a theoretical model is first hypothesized to explain a phenomenon (usually derived from a specific theoretical framework). An empirical testing phase then follows to determine experimentally if the previously hypothesis fits the observations. Care must be taken in the selection of the framework in respect to its generalizability. For instance, theories of motivation developed for explaining real world phenomena may not extend to a gaming context [67].

Model free approach (bottom up) In this approach, observations are collected and analysed to generate models without a strong initial assumption about what the model captures. This approach assumes the presence of an unknown function between the data and the construct that is under investigation but does not assume anything about the structure of this function [67]. Despite being able to achieve satisfying results, usually this approach generates insights that are hard to interpret with respect to the causes behind a specific phenomenon.

Hybrid approach A more flexible strategy is to consider a blend of the two aforementioned approaches where insights derived from a specific theoretical framework (or from a sets of experiments) are employed for better informing the construction of a model from a bottom up perspective [67].

This hybrid approach is especially relevant to the present work as it will constitute the general framework for our methodology: we will use the scaffolding provided in section 2.2 to define and constrain a computationally powerful and expressive bottom-up approach in the attempt to approximate the motivational states of players and predict their associated behavioural manifestations (i.e. engagement). This is not a trivial task: states generated by motivational processes are not directly observable or measurable. As we illustrated in section 2.3.2 they are latent constructs influencing measurable outcomes [102, 103]. In this view the challenge then becomes individuating the appropriate behavioural indicators from which we can try to reconstruct the underlying latent state.

2.5.1 Selecting the Appropriate Behavioural Measures

In a work by Yannakakis and colleagues [102], the authors retrieved a series of behavioural features derived from the interaction between player and hardware in a physical game an attempted to predict

the players' level of involvement and enjoyment. The authors found that the most informative features for this type of task were those representing the frequency and strength of interaction between the player and the physical hardware [102].

Narrowing the aim of the predictive model, we often see research focused exclusively on behavioural indices of disengagement or extinction, in the attempt to identify when a player transitions to a state of low motivation [104]. Many researhers have leveraged meta-behavioural metrics related to the frequency and amount of gaming behaviour in order to predict future dis-engagement [105–107]. In these analyses, metrics related to frequency and amount of playing behaviour appear to be suitable for constructing predictive models of engagement. This is in accordance to common practices in behavioural science when it comes to quantifying the amount of motivational drive [91] but we will expand on this in section 2.5.2.

As mentioned in section 2.3.1, the structural characteristics of a videogame are hypothesised to have a pivotal role in determining the motivational drive that an individual might have towards the playing behaviour. In this view, if metrics about the amount and frequency of playing behaviour can be interpreted as a proxy for the motivational saliency attributed to the act of playing in general, knowing which aspect of the game produced the playing behaviour can give us a sense of the direction of the motivational drive. Cowley and colleagues for instance proposed a method based on in-game behaviour for evaluating which specific in-game elements were driving engagement [108]. The underlying idea was that the actions performed by a player within a videogame could be grouped in three different categories depending on their goal (a similar approach can be found in [72]): social (directed towards other players), achievement (directed towards in-game goals) and fantasy (directed towards escapism). In this method, analyzing the type of actions taken by the player in a completely natural setting could provide an indication of the underlying motivational state driving engagement [108]. The resulting categories resemble those identified by some of the models presented previously [72, 73], but the approach here is based on in-game behaviour rather than relying on self-report measures. Despite being an interesting perspective and introducing a novel approach in which the player cognition can be inferred from the game behaviour, again the use of pre-defined categories fails to generalize to a wider range of games (e.g., a user evaluated in a game which does not have social features will not be able to express the propensity towards social factors). In a similar fashion, Wang et al. attempted to infer, using telemetry data, which elements inside a videogame were acting as reinforcers and supporting the playing behaviour [109].

If we recall our illustration of the engagement process model in Figures 2.2 and 2.3 we can see how the entire process is better understood when embedded within an "environment". Environment here refers to all those external factors that influence engagement (e.g. cultural norms or temporal factors). Bialas and colleagues for instance, investigated the influence of country of origin (used as a proxy measure for the players' cultural characteristics) on playing behaviour [110], although some differences emerged, the estimated effect sizes were quite modest.

In a subsequent pre-print, Zendle and colleagues [60] analyzed trans-national mobile gameplay data

from a set of 215 countries evenly distributed around the world. They found that the amount and distribution of hours played can vary considerably from region to region and hypothesized that observed difference might be due to cultural and socio-economical factors. In work carried out by Vihanga et al. the authors individuated a series of temporal patterns in how individuals distribute their gaming behaviour, indicating the hour of the day or the period of the year might affect the amount of engagement [111]. It is worth stressing however that these factors do not necessarily influence the motivational propensity of the individual (which should mostly be driven by the reinforcing mechanics specified in section 2.3.2) but rather its behavioural manifestation.

2.5.2 Models for Engagement Prediction and Profiling

In this section we will provide an overview of the most prominent model-based approaches aiming to predict engagement from behavioural indices. In particular we will focus on methodologies that rely on telemetry data acquired within a videogame context. Our focus will be on highlighting common themes connecting the various approaches as well as creating connections with the construct of motivation.

The work on engagement modelling, when based on behavioral measures comes in two forms: "prediction" and "description" of in-game behaviour [24]. The prediction of in-game behaviour is usually formulated as the solution to a supervised learning problem [24]. In this context a set of metrics of interest X are used for predicting one or multiple targets y by estimating the parameters of a function $y = f(X;\theta)$ [112]. The focus is less on the parameters of the function (which nevertheless must be inferred in an unbiased manner) and more on the accuracy of the performed predictions. The literature on predictive modelling for videogame engagement focuses almost exclusively on two prediction targets: churn and survival time (i.e., duration of engagement with a game).

Churn can be defined as the decision of an individual to stop interacting with a game. It is usually formalized as a prolonged period of inactivity [105–107, 113, 114]. Churn can be mapped to specific stages in the engagement process model of O'Brien et al., namely entering either the disengagement or extinction stages [61]. From a psychological point of view we can imagine the decision of an individual to stop playing a specific videogame as partially influenced by their motivation state. If we use the incentive salience framework presented in section 2.2, this would correspond to a decrease in the *wanting* component potentially caused by the inability of the game to provide enough rewarding experiences [32].

Survival time on the other hand, despite being closely related to churn, does not map to a specific stage of the engagement process model but rather quantify the extension of the process itself (i.e., how much playing activity can we observe from the point of engagement to disengagement or extinction) [106, 115–118]. Despite this,we can think of churn as a discretized version of survival time [104]. From the perspective of the underlying motivational state, survival time is a much more interesting and complex problem to solve. Indeed when predicting survival time we are not just interested in estimating if an individual is in a state of low motivational drive but rather where they

are located on the full spectrum of motivational state.

Engagement prediction modelling is intriguing because when we task a machine learning algorithm to solve $p(y|X) = f(X;\theta)$, we are implicitly inferring the underlying data generating process that is an approximation of the latent motivational state of the individual, conditional on the nature of X, (we will expand on this in the next chapter).

The other goal of engagement modeling is "description". This usually involves tackling a specific type of unsupervised problem of the form $p(K) = f(X;\theta)$ where K is a set of groups, clusters or partitions in which X can be divided by a suitable procedure f (we keep the functional notation for convenience) once appropriated parameters are estimated [112] [1]. This approach closely related to the idea of profiling presented in section 2.5 and is concerned with the estimation of the *direction* of the underlying motivation drive rather than its *magnitude*: individuals are assigned to different clusters or partitions based on how frequently and intensely they interact with specific mechanics inside the game (see section 2.5).

When inspecting the literature on engagement modelling (see Table 2.5)

we noticed how the focus was very often on the applied side: identifying and selecting the right algorithm for solving the specific practical problem at hand. Most (if not all) research focuses on what we can call "local models": in which a specific machine learning algorithm is selected, fitted and tested on data coming from a single videogame title [105–107, 134]. These type of approaches are constrained to learn models that are "local" to that specific context under scrutiny rather than being "global" and able to describe the interaction of an individual with different game objects. As we mentioned in section 2.2, this last characteristic is essential for constructing a good approximation of the motivational state of an individual. A notable exception was the work done by Liu et al. [123], where churn was formalized as edge prediction in a dynamic graph and modelled through an ANN. This produced a single model able to generalize across multiple game titles however with the limitation of them being all mobile titles and only considering a relatively low number of users (i.e. less than 20 thousand). In regard to the literature on survival analysis, we found that most works employed Cox Regression [135], or some variation of it, for estimating the probability to survive (i.e. not have churned) after a specific period of time [115–117]. Despite being a similar formulation, this is not equivalent to estimating the survival time (i.e. the amount of future playing time), which becomes much more interesting when trying to assess not only measures of disengagement but also measures of future sustained engagement. A notable exception to this is the churn and survival analysis competition presented by Lee et al. [129], where the goal was to estimate both churn probability and survival time.

What emerges here is a strong tendency to tackle the problem from a bottom-up perspective (see section 2.5). Usually compelling solutions for practical problems are produced though a black box approach: a machine-learned model is generated for solving a specific task but no attempts are made

[1] Although our notation assumes that f is parametric, this is not always the case.

Table 2.5: **ANN**: Artificial Neural Network, **DT**: Decision Trees, **KNN**: K-Nearest Neighbour, **HMM**: Hidden Markov Model, **LR**: Logistic Regression, **RL**: Reinforcement Learning, **CR**: Cox Regression, **AA**: Archetypal Analysis, **MF**: Matrix Factorization, **KM**: K-Means, **SC**: Spectral Clustering, **GMM**: Gaussian Mixture Model, **HS**: HDBSCAN, **DC**: DEDICOM.

Algorithm	Task	Author	Year
ANN	Churn Prediction	Runge et al. [105]	2014
DT	Churn Prediction	Hadiji et al. [107]	2014
KNN	Churn Prediction	Castro et al. [119]	2015
HMM	Churn Prediction	Rothenbuehler et al. [120]	2015
HMM	Churn Prediction	Tamassia et al. [121]	2016
DT	Churn Prediction	Drachen et al. [113]	2016
DT	Churn Prediction	Milosevic et al. [114]	2017
DT, ANN, LR	Churn Prediction	Kim et al. [106]	2017
ANN	Churn Prediction	Guitart et al. [122]	2018
ANN	Churn Prediction	Liu et al. [123]	2018
ANN	Churn Prediction	Kristensen et al. [124]	2019
ANN	Churn Prediction	Liu et al. [125]	2020
RL	Churn Prediction	Roohi et al. [126]	2020
CR	Survival Time Prediction	Bertens et al. [117]	2017
CR	Survival Time Prediction	Perianez et al. [115]	2017
CR	Survival Time Prediction	Demediuk et al. [116]	2018
CR	Survival Time Prediction	Viljanen et al. [118]	2018
CR	Survival Time Prediction	Guitart et al. [127]	2019
CR	Survival Time Prediction	Fernandez del Rio et al. [128]	2019
DT	Survival Time Prediction	Liu et al. [129]	2019
AA, MF, KM	Profiling	Drachen et al. [130]	2014
KM, MF, SC	Profiling	Bauckhage et. al. [131]	2014
GMM, AA, KM	Profiling	Drachen et al. [131]	2016
KM, HS	Profiling	Makarovychet al. [132]	2018
DC	Profiling	Aung et al. [133]	2019

to inspect or interpret it [124, 125, 129, 136]. Moreover, when these attempts are made, the lack of a solid and predefined theoretical framework tends to lead to *post-hoc* interpretations which are sometimes difficult to verify or relate to actual human behaviour [113, 128]. One of the reason is that these solutions are usually produced by considering an unconstrained set of game-specific metrics. As a result, *a-posteriori* justifications are provided [100, 101, 132], which, without an overarching explanatory theoretical framework, appear to be be very context-specific and difficult to interpret.

What we see in the literature is that attempts are made to model a single behavioural manifestation of engagement rather than the construct in its entirety. A noticeable exception in this regard is the work by Reguera et al. [137], who adopted a complete data-driven approach to derive a general law for describing and quantifying the engagement process, similarly to Bauckhage *et al* [103]. However, neither group interpret their findings through the lens of existing theories of human behaviour.

We believe that a holistic model of engagement can be generated, constraining the great flexibility provided by data-driven approaches by employing well established theoretical priors [67]. To do so, an *a-priori* theoretical framework which will generalise to different situations should be defined. Such a framework should clearly identify the observable and measurable indicators of engagement and how they are expected to vary in relation to the construct's dynamics. By doing this, the findings emerging from data-driven approaches can be compared with predictions from the theoretical framework.

2.5.3 Models for Estimating Motivation-related Latent States

In the course of this chapter we have often referred to the concepts of "motivational state" or "latent state", but what do we mean by state? We know that individuals' body and mind are subject to continuous changes driven by physiological, cognitive and affective processes. These changes are the constituent parts of so called "internal states" which can be thought as dynamic, latent constructs able to modulate observable behaviour [15–18].

Coming back to what we presented in section 2.2.2, we can think of the concept of *wanting* (i.e. level of attributed incentive salience) as a latent internal state generated by reward and motivational processes which has the effect of biasing and directing behaviour and cognitive processes towards specific objects [35]. Despite their relevance, the study of these entities in naturalistic contexts is not straightforward: their inference is often the solutions to an "inverse problem" [112] where observable and easy to acquire measures (e.g. patterns of behaviour) are used to estimate the internal factors that generated them (e.g. latent states related to motivation and reward processing) [16, 138]. This idea is not new [139], but it has gained traction in recent years because of the increased availability of large volumes of data collected both inside and outside controlled experimental settings. Because they enable the study of phenomena in naturalistic settings, data collected with ecologically valid approaches are particularly interesting but come with their own set of challenges [140] which largely overlap with those identified in paragraph 2.4.3.

Recent approaches based on latent variable models have shown promise in taming some of these problems [18]. Calhoun et al. for instance employed a combination of Hidden-Markov-Model (HMM) and Generalized Linear Model (GLM) for deriving the latent states underlying motor behaviour [18]. Approaches based on HMM, despite being easily interpretable, might struggle to overcome the issues related to complexity [15, 141] and scalability (e.g. challenges in fitting large state spaces) [142]. In this regard, a promising line of research is the approximation of latent states through the representation generated by Artificial Neural Networks (ANNs) [15–17, 20–23]. ANNs are well-suited for applications with large amounts of data [143], provide noise resiliency and are able to capture complex interactions in the data [28]. The underlying assumption is that the latent states, despite being embedded in a high dimensional space (e.g. patterns of behaviour or brain activity), lie on a so called manifold (we will expand on this concept in the next chapters) that can be effectively described using far fewer degrees of freedom [10, 19, 20]. As we illustrated in Figure 2.1 for instance, the motivational drive of a particular individual could be reduced, at any given time, to a 2D plane

representing the intensity and the target of the the goal-directed behaviour [33]. The type of tasks that these models attempt to accomplish are very similar to the unsupervised problem presented in section 2.5.2, however given an input $X \in \mathbb{R}^D$ instead of trying to find a suitable way for clustering or partitioning it the objective is to derive a representation $h \in \mathbb{R}^K$ able to explain most of the variations in X with the constraint $K \ll D$ [27, 112]. At this point, the problems boils down to finding the right algorithm for capturing and representing (as faithfully as possible) the complexity of the latent representation [15, 141] while also being able to scale to large volumes of data [142].

Approaches based on ANN seems to make use for the most part of unsupervised (e.g. autoencoders) [20, 22] or generative methods (e.g. generative adversarial networks) [15, 22], with a particular focus on algorithms that can capture the dynamics underlying changes in latent states [15, 16]. These techniques usually work by discovering a lower-dimensonal manifold structure of the latent states [15] which however might still be embedded in a very high dimensional space (i.e. the size of the representation generated by the model). In order to inspect the generated representations, non-linear dimensionality reduction is usually applied [22] using algorithms such as the t-distributed stochastic neighbor embedding [144] (t-SNE) or Uniform Manifold Approximation and Projection (UMAP) [30].

The advantage of ANNs lies not just in their scalability and representational power but also in their architectural flexibility. Indeed it is possible to design an ANN in such a way that its computation is modified by specific constraints imposed by the experimenter [15].These desirable properties however come at the cost of interpretability and ANNs are often characterised as inaccessible black boxes only capable of efficient input-output mapping. In line with a growing tendency in the literature [20–23, 145, 146], we argue that this is only partially true and that given full access to the computations performed by an ANN, a certain degree of interpretability can be achieved. Through the use of prior theoretical knowledge, it is possible to constrain the input, the objective and the architecture of an ANN to generate so called "latent representations" (which can be thought as un-observed variables able to explain observable phenomena). Through reverse engineering, it is possible to extract the manifold structure embedded in these high dimensional representations and test it against theory driven hypotheses [145, 146].

2.6 Summary

In this chapter we provided an overview of the concept of reward driven motivation focusing in particular on the incentive salience hypothesis proposed by Berridge and Robinson [31]. This construct can be thought as a psycho-biological process that helps individuals constructing latent representations of objects which are imbued with saliency and used for directing behaviour [32].

We then presented an overview of the concept of engagement in digital games proposing the idea that it can be interpreted as a behavioural manifestation of the changes occurring in the latent motivational states of players [32, 61]. These latent states would be constructed and modified by

individuals in a dynamic fashion when interacting with a specific game objects [32, 61]. Various factors are involved in the underlying dynamics, namely: the ability of the game objects to provide pleasurable experiences to the individuals (i.e. through their structural characteristics), the internal state of the individual (e.g. cognitive, affective and physiological changes) and the surrounding environment [32, 58, 59, 61–64].

We then proceeded with a small introduction to the methods used for assessing and estimating the engagement and motivational state of individuals. From this brief overview it emerged that behavioural metrics appear to be a good compromise between scalable and easy to interpret measures (e.g. self-report questionnaires) and objective indices (e.g. psychophysiological measures). Moreover, when acquired through telemetry systems, these indices offer a reliable source of ecologically robust measurements that can span many different dimensions (i.e. virtually any type of behaviour and any desired frequency of logging, if tracked by the telemetry system) and can be acquired in large quantities [24]. However, the complexity of the the data and the lack of experimental control during their recording makes it challenging to run controlled experiments. Therefore, modelling approaches (usually carried out through machine-learning techniques) are usually preferred over conventional hypothesis testing or inferential statics ones.

We argue that a connection between models used for estimated latent states of individuals and those employed in the videogame literature for performing engagement prediction. Indeed, many of the approaches presented in section 2.5.2 rely explicitly on the estimation of a latent representation which is instead the focus of the solutions presented in section 2.5.3. For instance, we could re-write the supervised learning problem presented previously as a chain of two functions: one producing the latent representation $h = f_1(X;\theta_1)$, and another performing the prediction $y = f_2(h;\theta_2)$. In this way, engagement prediction model can be easily connected with the latent-states estimation techniques presented in section 2.5.3 and be considered under a unified modelling framework.

In line with this idea, the next chapter will outline theoretical and methodological foundations for approximating the manifold structure of motivation and reward related latent states using behavioural data and ANN.

We will show how the estimation of the motivational state of an individual is strongly connected with engagement prediction and can be cast to a supervised learning problem. Our approach will be a blend of bottom-up and top-down modelling [67]. We will subject the expressive power of ANNs to specific theoretical constraints to increase their interpretability and test specific *a-priori* hypotheses.

Despite the fact that our work aims to generalize to various areas of application, our experimental efforts will exclusively make use of telemetry data coming from videogames - although we will consider a heterogeneous set of game genres. Since video games rely heavily on reward and motivational processes for producing playing behaviour [70, 71, 82, 147–149] and allow to record large volumes of behavioural data in a naturalistic fashion [25], we consider them to be a suitable test bed for our work.

Chapter 3

From Theory to Modelling

3.1 Introduction

In this chapter, we will lay down the theoretical and computational foundations of our approach for approximating the manifold structure of attributed incentive salience in scenarios where only large volumes of behavioural data are available.

First, we will briefly illustrate how previous work has framed the modelling of this construct as a reinforcement learning problem and solved it using Temporal Difference Learning (TD Learning) [150]. This, will provide us with a psycho-biologically plausible computational model of attribute incentive salience and constitute the starting point for our approach. Then, we will highlight how video games are promising candidates for studying the behavioural aspects of incentive salience attribution in naturalistic settings. Finally, combining these two ideas, we will show how estimating the manifold structure of attributed incentive salience can be cast as the solution to a supervised learning problem and why Artificial Neural Networks (ANNs), thanks to their representation learning capabilities, are well suited for the task.

3.2 Manifold Representation of Attributed Incentive Salience

As anticipated in the previous chapter, we can think of the level of attributed incentive salience (i.e. the amount of *Wanting* attached to a particular object) as a latent state influencing motivated behaviour. The representation of these latent states must be translated to behaviour by the activity of multiple brain regions. As we mentioned before, these multidimensional patterns of representation are believed to reside on a manifold [10,19]: a connected low-dimensional region embedded within a high-dimensional space [28]. An intuitive example of this is the way in which the brain generates and stores mental maps of the environment which are then used for navigation tasks [8,9]. The dimensionality of the encoding signal is much larger than the intrinsic dimensionality of the spatial information encoded

within it, indeed the activity of large neuronal populations is involved in generating a mapping that needs to be only three-dimensional. When applied to incentive salience an intuitive representation sees the manifold as a two-dimensional space, similar to what presented in Figure 2.1, generated by the activity of all those brain areas involved in the attribution of value and subsequent modulation of future motivated behaviour. This two-dimensional space would represent, at any given time, the motivational saliency than an organism attributes to a potentially rewarding object and therefore also the intensity of the expected future behaviour [31, 33]. This idea of a neural manifold has found experimental support in different areas (e.g. motor control [7], mnemonic processes [8, 9], reward processing [151] visual [10, 11] and olfactory [12] perception) and relies on the fact that neural activity is highly redundant and reducible to just few correlation patterns [7]. In light of this, the use of dimensionality reduction techniques to represent the manifold structure of highly dimensional data has proven to be valuable in making un-observable entities like latent states more easily accessible and interpretable, while also facilitating their mapping onto brain [152, 153] and behavioural data [20–23].

3.3 Computational Framework

The theoretical approaches presented in chapter 2 might appear to provide an over-simplistic view on motivation. However, processes such as classical and operant condition (and by extension reinforcement learning) are some of the most important and elegant (in terms of complexity to explanatory power ratio) mechanism able to describe motivated behaviour [32, 34]. Due to the relative formalism with which they are described, it is easier to leverage them for deriving computational models of reward-driven motivational processes such as incentive salience attribution [13, 14, 32].

3.3.1 Temporal Difference Learning

The first attempt to model incentive salience attribution was carried out by McClure *et. al.* using TD Learning [13]. The use of TD Learning in simulation studies involving reward learning, is often motivated by its good approximation of the reward-prediction error signal generated by dopaminergic neurons [34, 57]. Algorithms in the family of TD Learning attempt to learn a value function V by iteratively refining an estimate $\widehat{V}$ over time [154]. In the most basic form, called TD(0), this is done by simply observing the reward r associated with a particular state s at time t_{+1} and using it to adjust the estimate of V produced at time t [154]. Here t is an arbitrary unit of time, it can be specific (i.e. seconds) or generic (i.e. a point in an ordered series of events) depending on the type of application. If we let $S = \{s_t : t \in T\}$ be a sequence of states, then the value V at s_t is given by the sum of all future discounted rewards expected when transitioning from s_t to s_{t+1}

$$\begin{aligned} V(s_t) &= \mathbb{E}[r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + ... + \gamma^T r_T] \\ &= \mathbb{E}[r_{t+1} + \gamma V(s_{t+1})] \end{aligned} \tag{3.1}$$

with $\gamma \in [0,1]$ being a discounting factor for r. The iterative refining of $\widehat{V}$ carried out by TD learning is achieved by first computing an error signal δ at time t

$$\delta_t = r_{t+1} + \gamma \widehat{V}(s_{t+1}) - \widehat{V}(s_t) \tag{3.2}$$

which quantifies the difference between the current $\widehat{V}$ and what is expected when transitioning to s_{t+1}. Once the error signal is computed, $\widehat{V}(s_t)$ is updated as:

$$\widehat{V}(s_t) \leftarrow \widehat{V}(s_t) + \alpha(\delta_t) \tag{3.3}$$

where $\alpha \in [0,1]$ is a constant controlling the amount of updating or the "learning rate". This process called TD update is illustrated by the diagram presented in Figure 3.1. Conventionally the transition from s to s_{t+1} is the result of an action selection process guided by $\widehat{V}$, because in optimal control settings the role of reinforcement learning is, in simple terms, to select the course of action that maximizes future rewards [13, 34, 154].

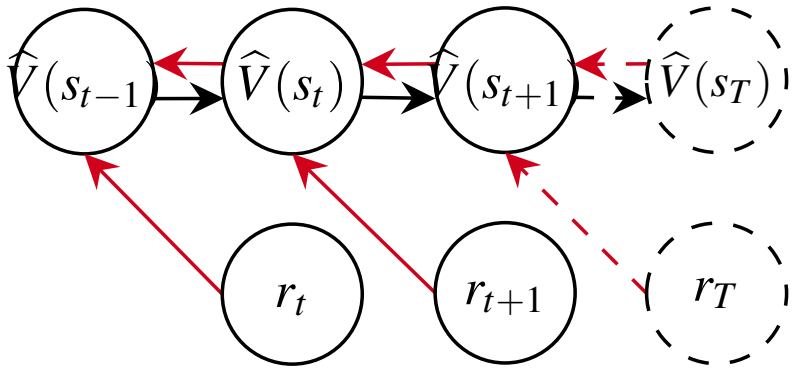


Figure 3.1: Red arrows indicate the flow of the computations for deriving δ and updating $\widehat{V}$ expressed by equations 3.2 and 3.3. Black arrows instead indicate the changes of $\widehat{V}$ moving from s to s_{t+1}. Solid circles indicate states which have already been observed while dashed ones represent future not-yet observed states.

McClure *et. al.* proposed that incentive salience is represented by V as defined in equation 3.1 while the error signal expressed by equation 3.2 represents the activity of dopaminergic neurons with the dual function of driving the attribution of incentive salience (through reward prediction error coding as specified in section 2.2.2) and guiding the previously mentioned action selection process [13, 34, 155]. However, in later work, Zhang *et. al.* highlighted the fact that the model proposed by McClure *et. al.* fails to take into account an important part of the original incentive salience hypothesis: the dynamic modulation produced by the individual's internal state (see section 2.2.2) [13, 14, 32, 38, 52, 53]. Zhang *et. al.* therefore proposed a modification of the original TD Learning model to include a modulatory factor $k \in [0, +\infty]$ which can enhance ($k > 1$), dampen or even revert ($k < 1$) previously learned incentive salience values

$$V(s_t) = \mathbb{E}[\tilde{r}(r_{t+1}, k_t) + \gamma V(s_{t+1})] \tag{3.4}$$

here $\tilde{r}(.,.)$ is a function of two variables and can assume either an additive or multiplicative form

[1]. The main difference between the approaches of McClure *et. al.* and Zhang *et. al.* lies in the interpretation of V. Both authors see it as a combination of cached value (i.e. what has been learned from past experiences) and expectation over future r but for McClure *et. al.* all the future r have the same weight while for Zhang *et. al.* the individual's state modulates the weighting of r dynamically. Using the notation from section 2.2.2, we can say that the interaction s between I and O at time t_{+1} arises from the V (i.e. incentive salience) generated after s_t. The mismatch between the predicted amount of reward and the actual reward received at time t_{+1} generates an error signal that allows I to learn about the "correct" magnitude of $V(s_t)$ [156] . As an example, an individual may anticipate that eating their favourite meal would be a rewarding experience but instead (for some reason) it was underwhelming. They therefore reduce the salience previously attributed to it. Importantly, $V(s_t)$ does not just encompass the previous history of interactions between an individual and an object but also the current state of the individual themselves: the individual has learned from long experience that eating is a pleasurable activity but currently, since they are satiated they do not expect much reward from doing it again in the near future.

In this view, motivation can be described as a mechanism that guides the interaction between individuals and objects. It controls and selects behaviours which are expected to lead to pleasurable outcomes for the individual (i.e. incentives or reward). These expectancies are the product of a learning process that can be modulated by the internal state of the individual. Therefore, from a behavioural point of view, an objects O can acquire salience for an individual I conditioned on its capacity to elicit rewarding experience r [13, 31]. The amount of attributed salience is a valued representation of O generated by I and controls how likely and intense future interactions between the two will be. [13, 31]. Let B represents the strength of an interaction between I and O, r a measure (e.g., a signal or a representation) of how rewarding the interaction with O is perceived to be and V the generated attributed incentive salience. Following Figure 3.2, at time $t+1$ an individual will produce an interaction with an object of strength B according to the previous V_t. If we recall from section 2.2.2, this process relies heavily on learning mechanisms making V by nature dynamic and mutable. It should be noted that B can be represented as a multidimensional variable defined by the instrumental behaviours conventionally used for assessing the *wanting* component in animal studies (e.g., frequency, amount and duration of feeding behaviours like bites, nibbles and sniffs) [31]. During and after the interaction I will experience a variable degree of reward r_{t+1} that, weighted by their internal state, will then be used for updating V_t. It is worth noting that the individual's internal state is not the only factor involved in the modulation of r. The context in which I and O interact (Env_{t+1} in Figure 3.2) also seems to contribute to this [157]. Following the idea presented in section 3.2, the latent state defined by V could be represented as a manifold defined by the activity of those regions responsible for the attribution of incentive salience. Moreover, given the strong coupling between attributed incentive salience and behaviour [31] we would also expect the structure of this manifold to be a suitable descriptor of the behavioural aspects of attributed incentive salience.

[1] See [14] for detailed description of the two forms and their functional differences.

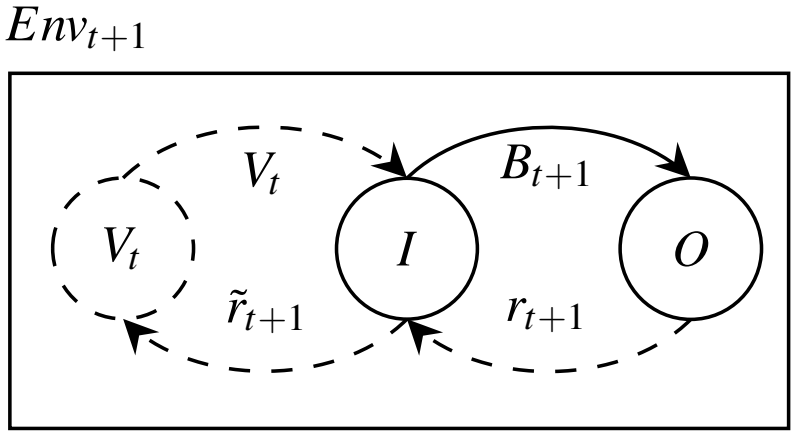


Figure 3.2: Solid and dashed elements represent respectively observable (i.e. behavioural) and latent aspects of the process. The individual and the object that they interact with are indicated by I and O. The strength of the interaction is represented by B. The salience that I attributes to O is expressed by V while r and $\tilde{r}$ are the experienced reward and its weighted version produced by the state of I. All the contextual factors influencing both the amount of r perceived and the magnitude of B produced are represented by Env. The dynamical natures of the process is expressed though the arbitrary temporal unit t.

From TD to Supervised Learning The approaches discussed above frame the estimation of attributed incentive salience as a reinforcement learning task. This requires the simulation of a sequence of interactions between I and O and the concomitant delivery of r [13, 14, 34]. However, it is not always straightforward to replicate these interactions in real world scenarios, especially when dealing with human participants. The control on the internal state of I and amount of r delivered that McClure and Zhang assume is usually based on strict assumptions and can be achieved only in controlled experimental settings [13, 14]. As an alternative solution for inferring V outside the laboratory we propose to learn its manifold structure through supervised learning. Differently to what reported in the literature [18, 20–23] we argue that in this case the use of supervised in place of un-supervised techniques is to be preferred. Indeed, since we are dealing exclusively with behavioural data and trying to solve an inverse problem we would like to learn a manifold structure which is not just a generic indicator of behavioural phenotype [20] but also obeys to specific functional constraints (e.g., the predictive properties of attributed incentive salience).

In this approach, an experimenter gathers data on a set of interactions between I and O and let a learning algorithm to estimate two functions:

$$\begin{aligned} V(s_t) &= f^1(O, \tilde{r}_t, V(s_{t-1}); \theta^1) \\ r_{t+1} &= f^2(V(s_t); \theta^2) \end{aligned} \tag{3.5}$$

here f^1 and f^2 are arbitrarily complex functions while θ^1 and θ^2 are parameters that the learning algorithm has to infer. The future reward that an individual expects after an interaction with an object is produced by the current level of attributed salience, which itself is a function of the current internal state of the individual (expressed through the amount of reward just experienced) and the incentive salience previously attributed to the object. It is important to note that the two functions above need to be recursive over all $s \in S$ (see equations 3.1 and 3.4) in order to provide $V(s_t)$ with the

dual purpose of caching all the past V and being a suitable predictor for all the r. This formulation however still requires a measure of the r experienced by I (or more precisely its weighted version $\tilde{r}$) after interacting with O, which is not easily accessible. However, Thorndike's law of effect [158] and Skinner's operant conditioning principles [159] suggest that r, which like V is a non observable latent variable, manifests itself through the intensity of interactions between I and O (i.e. B in Figure 3.2): the frequency and amount of object-directed behaviours increase or decrease as a function of the rewards an individual expects to receive [32, 156]. Since $V(s_t)$ predicts how much r an I expects to receive from interacting with O, we should also expect the strength of their future interactions to be a function of $V(s_t)$. This can be represented re-arranging the equations in 3.5 in a more compact form as a chain of functions

$$\mathbb{E}[B_{t+1}] = f^2(f^1(O, B_t, V(s_{t-1}); \theta^1); \theta^2) \tag{3.6}$$

To approximate the above expression, a learning algorithm would require records of behaviours generated by individuals while interacting with a diverse set of potentially rewarding objects. Here, we argue that video games are one way to obtain this type of data at scale while also achieving some level of ecological validity.

The Problem with Inverse Reinforcement Learning Instead of relying on supervised learning approaches for inferring the latent motivational state of an individual we could have leveraged a class of algorithms more closely connected with RL: those belonging to the Inverse Reinforcement Learning (IRL) family [29]. These algorithms are concerned with modelling the optimal choices of an agent without specifying a reward function but instead inferring it from observed behaviour [29]. In the videogame literature, we found one example of application of IRL to the problem of reward function estimation [109], suggesting the relevance but at the same time potential challenging nature of such algorithms. These challenges became particularly relevant once we attempted to formulate our problem within an IRL framework:

- The optimisation process associated to inferring the reward function is an ill-defined problem: many reward functions can explain a single trajectory of observed behaviour [29]. This same issue is common to our current work so we can consider it as irreducible.
- From an application point of view, it is challenging to convert the outcomes produced by an IRL algorithms to business-related applications. A supervised learning model directly expose predictions that are relatively easy to understand by technical and non-technical users. Differently, IRL provides a reward function that may or may not translate directly to actionable interventions.
- Despite the appeal of directly estimating a reward function underlying observed behaviour (this aspect is integrated out by our approach), this still requires to specify the agent's dynamics in charge of this transition c [29]. Although this is not infeasible it is a very challenging task to be performed within the context of human behaviour: it is not clear how the function governing

state transitions should look like (unless some strong simplifying assumptions are made).

- A more practical concern regards the complexity of casting the inference problem withing an IRL framework [29]. Algorithms used for IRL scales in time and space complexity with the size of the Markov Decision Process (MPD) they need to evaluate. Unfortunately in the context of naturalistic human human behaviour the MDP is not just extremely large (e.g., it is very hard to constrain the number of actions and states) but also very often continuous.
- Finally, from a theoretical point of view IRL applications are usually developed for applications in optimal control settings [29] or in situations where there is a clear definition of the states generated by the agent [29]. When we attempt to infer the reward function underlying motivated behaviour solely from observational data, the situation is rather different. Indeed, the concept of "optimality" in this setting is quite fuzzy: human agents "in the wild" do not necessarily behave rationally or optimize for a clear and consistent objective. Moreover, it is never possible to have a clear definition of the state generated by a given action. This, is not directly observable from data but rather a combination of generated behaviour and changes in the internal state of to the individual (e.g., the previous and current level saliency attributed to a given object).

Overall, despite the potential that IRL shows in the context of inferring the motivation-related latent states, the limitations we just outlined make it an impractical solution to our problem. Moreover, the connection between IRL and supervised learning [29], suggests that this last one might be considered an acceptable framework for this type of inference problem, especially in situation when only behavioural data are available.

3.3.2 Video Games and Telemetry

As we highlighted in chapter 2, in conventional circumstances, interacting with video games is a volitional activity driven largely by the capacity of the games to provide pleasurable experiences [59].

Behaviour within the game is best understood as the result of a value attribution process similar to that of secondary reward objects (see section 2.2.2). Indeed, it appears that the play behaviour is often produced and maintained by the structural characteristics of the game (e.g. game mechanics) [66] which, working like conventional reinforcement mechanisms [70, 71, 82, 147], produce effects similar to operant conditioning [159].

Although caution should be applied when complex activities are investigated using neuroimaging techniques, evidence suggest that the maintenance of video games playing behaviour engages the same cortico-striatal structures [160–165] and neurotransmitters [166] involved in reward processing. This, seems also supported at the behavioural level where the amount of experienced in-game reward appears to play a role in controlling how likely is an individual to keep engaging in playing behaviour [148,149]. This, in conjunction with a growing literature highlighting similarities between certain video game mechanics and activities driven by secondary rewards (e.g. gambling) [65, 167, 168], corroborates the

idea that video games can elicit behavioural responses through incentive mechanics.

In this view, video games with different structural characteristics could be seen as "objects" possessing rewarding properties that heavily depend on the individuals interacting with them (e.g. an individual's preference for a specific game mechanic). Hence, similarly to the process specified in chapter 2, we can expect that through repeated interactions, an individual will experience varying degrees of reward determined by their internal state and the characteristics of the game. These interactions will produce continuous adjustments in the level of saliency attributed to playing that specific game which in turn will influence the frequency and amount of future interactions with that same game.

Other than offering a context for observing the process of incentive salience attribution, video games allow us to obtain large volumes of behavioural data (similar to those mentioned in chapter 2) in a naturalistic and less obtrusive fashion. This is made possible by the widespread practice of obtaining high frequency records (i.e. telemetry[2]) of players' behaviour during the game [25]. This approach, despite offering less control and rigour than conventional experimental procedures, allows us to obtain a more faithful representation of natural behaviour (similarly to field studies) while avoiding some of the limitations connected with laboratory-based studies (e.g. sampling and observer biases).

In order to use this type of behavioural data to model attributed incentive salience, a learning algorithm should possess the following properties. First, it should be scalable and noise resilient, to leverage large volumes of naturalistic data in an efficient and effective manner. Second, it should be able to approximate arbitrarily complex functions, given that the shape of the functions specified in equation 3.3.1 is not known a-priori. And finally, it should be able to produce an approximation of $V(s_t)$ that can be inspected in order to evaluate if its functional properties can be compared with those of attributed incentive salience. Since Artificial Neural Networks (ANNs) appear to satisfy these requirements we will now proceed at contextualise their use within the current work.

3.3.3 Artificial Neural Networks

In their conventional form, ANNs can be seen as chains of nested functions (the layers of the network). These layers are vector valued (there are multiple units or neurons in each layer) and organized as directed acyclic computational graphs (information only flows forward). When the number of layers is greater than two, the prefix "deep" is usually applied [28].

The goal of this ensemble of functions is to create a mapping between an input X and a target y. Following the example illustrated in Figure 3.3, given the set of parameters $\Theta = \{\theta^1, \theta^2\}$ an ANN would first infer a function $H = f(X;\theta^1)$, mapping the input to a new representation H. The same representation H would then become the input of a second function $\widehat{y} = f^1(H;\theta^2)$ which produces an estimate of the target [28]. In the interest of clarity, it is worth mentioning that the parameters θ of each function are, in the simplest case, constituted by a weight matrix W and a bias vector b, similarly to what we have in conventional linear models.

[2] See [24] for a more technical description of telemetry in video games.

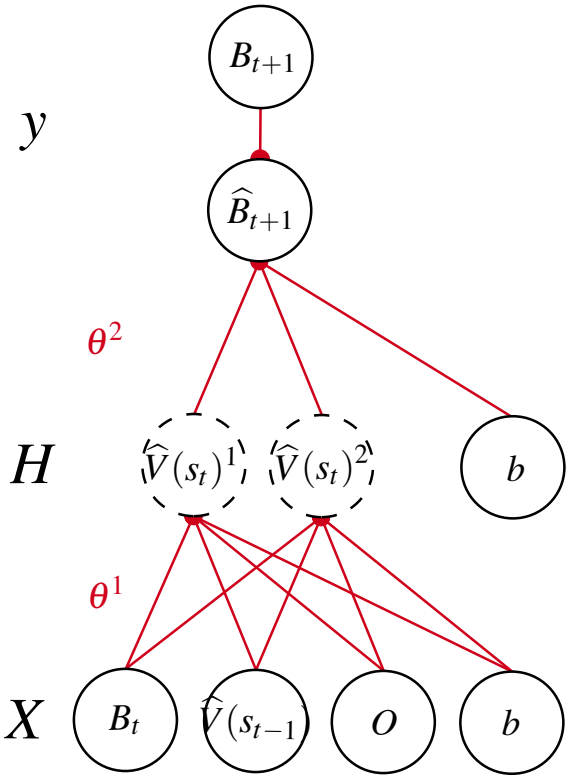


Figure 3.3: The figure represents how a feedforward ANN could be used for estimating $V(s_t)$ given a sequence of observed behaviors (B) produced while interacting with an object (O). Here X and H indicates the model's input and the inferred representation. y indicates both the target and the estimate produced by the model while b is a bias term. The collection of all the red lines indicates the Θ that the ANN has to estimate while each line represents a single parameter w or b. The circles are computational units (i.e. artificial neurons) whose outputs are given by $\phi(W_i^\top X + b)$. Here, ϕ is a non-linear function conventionally called activation, W_i the weight matrix for the i^{th} unit, b the bias vector and X the output from the previous layer.

In this sense, we can think of each layer as a collection of many non-linear vector to scalar functions taking the previous layer as input and generating the units for the layer that follows [28]. By increasing the number of layers and units, ANNs can approximate an extremely large class of functions [169].

An ANN finds the optimal values for Θ by taking forward and a backward passes through the computational graph. In the forward pass, information flows from the input X to the estimate $\widehat{y}$ according to the operations specified in Figure 3.3. During the backward pass, the error between $\widehat{y}$ and the target is first computed

$$\mathscr{L} = l(y, \widehat{y}) \tag{3.7}$$

Here l is a generic convex and differentiable function measuring the distance between y and $\widehat{y}$. Then, the gradient of the error with respect to all the parameters is found and an update is performed taking steps of size $\alpha \in [0,1]$ in the direction opposite to the gradient

$$\Delta w_i^j = -\alpha \frac{\partial \mathscr{L}}{\partial w_i^j} \tag{3.8}$$
$$w_i^j \leftarrow w_i^j + \Delta w_i^j$$

What we illustrated here is the application of the delta rule for updating the i^{th} weight of the j^{th} layer through gradient descent [170]. Deep feedforward ANNs rely on a generalization of this rule (i.e. backpropagation [169]) for efficiently computing the gradient for all the parameters in the network.

Returning to the supervised learning problem specified in section 3.3.1, a feedforward ANN approximates $V(s_t)$ by mapping the inputs of equation 3.6 to a candidate $\widehat{V}(s_t)$ which is then used to generate an estimate $\widehat{B}_{t+1}$.

Then, during the backward pass $\widehat{V}(s_t)$ is adjusted based on the degree of mismatch between the estimation that it produced and the real value of B_{t+1}. It is of interest to note that there is a certain degree of overlap between how ANNs adjust their weights and the TD update illustrated in section 3.3.1. Indeed, in single-step scenarios (i.e. predicting s_{t+1} based on s_t for each $s \in S$) the parameter changes produced by the two methods are the same [150]. The major difference lies in the delivery of the update: TD learning performs it at every step while backpropagation-based algorithms must wait until the end of the sequence in order to collate all the observed errors in a single term [150].

Recurrent Neural Networks Despite their universal function approximation properties [171], feedforward ANNs are not suitable for the type of recursive operations expressed in paragraph 3.3.1 [28]. As we can see from Figure 3.4A, given a sequence of inputs and targets, a conventional feedforward ANN would be limited to learning a temporally local function of the form

$$\mathbb{E}[B_{t+1}] = f^2(f^1(O, B_t; \theta^1); \theta^2) \tag{3.9}$$

Even when Θ are shared across time, the estimated $\widehat{V}(s_t)$ cannot incorporate information from past $\widehat{V}(s)$ nor guarantee predictive power for the future B.

A solution to this problem is offered by ANNs with feedback connections like Recurrent Neural Networks (RNNs). These are a class of ANNs that are able to efficiently process long sequences of data while also relaxing the requirements of conventional feedforward ANNs for fixed length inputs [28]. Looking at Figure 3.4B, we see that for each $t \in T$ a RNN would compute $\widehat{V}(s_t)$ using both the input OB_t and the previously estimated representation $\widehat{V}(s_{t-1})$. This, in combination with the recursive application of Θ, allows the network to learn a function over the entire temporal sequence and to provide $\widehat{V}(s_t)$ with the desirable properties mentioned in section 3.3.1.

The structure of Θ is more complex in RNNs than in feedforward ANNs [3] and a detailed derivation of the underlying optimization process is outside the scope of the present work. Nevertheless, it is worth singling out how the recurrent nature of the computations underlying the generation of $\widehat{V}(s_t)$ makes RNNs suitable for approximating the function specified in section 3.3.1.

Following Figure 3.4B, let $\widehat{V}(s_t)$ be the representation inferred by the model at time t and its associated parameters. Optimal parameter values are found through the same update rule used in feedforward ANNs

$$\widehat{V}(s_t) \leftarrow \widehat{V}(s_t) + -\alpha \frac{\partial \mathscr{L}}{\partial \widehat{V}(s_t)} \tag{3.10}$$

however, since $\mathscr{L}$ can now only be observed at the end of a temporal sequence, computing $\frac{\partial \mathscr{L}}{\partial \widehat{V}(s_t)}$ requires us to take into account all the intermediate steps from t to T. This is achieved applying the chain rule and propagating the error gradient backward in time [28, 172]

$$\frac{\partial \mathscr{L}}{\partial \widehat{V}(s_t)} = \frac{\partial \mathscr{L}}{\partial \widehat{V}(s_T)} \frac{\partial \widehat{V}(s_T)}{\partial \widehat{V}(s_{T-1})} \cdots \frac{\partial \widehat{V}(s_{T-n})}{\partial \widehat{V}(s_t)} \tag{3.11}$$

This implies that, similarly to TD update, the error flow forces $\widehat{V}(s_t)$ to retain information from OB_t and $\widehat{V}(s_{t-1})$ in order to perform estimation of B_{t+1} while still being useful for generating $\widehat{V}(s_{t+1})$ as we can see from Figure 3.4B. This process is made more efficient by an RNN variant called Long Short-Term Memory (LSTM) [173], which, as well as improving the propagation of the error gradient, has specialized mechanisms for inferring, at each point in time, which portion of information should be kept or discarded in order to minimize $\mathscr{L}$ [28, 173].

3.3.4 Artificial Neural Networks for Manifold Learning

As mentioned in the previous sections, ANNs are tasked to create latent representations (e.g. $V(s_t)$) which are not explicitly defined by their input or target but are nevertheless functional for connecting the two [28, 169, 174]. This is based on the hypothesis that the relationship between the input and the target can be expressed in terms of variations in coordinates on a manifold [28]. In the lower dimensional space of this manifold, the input is re-organized to improve estimation and elements which are similar to each other tend to appear close together [28].

In this view, during optimization, each layer of an ANN attempts to place its input on a manifold that is useful for the layer that follows. This process continues until the last layer. Here the inputs are organized in such way that it makes easier for the network to produce good predictions of the

[3] See [28] for a description of the parameters' structure in RNNs.

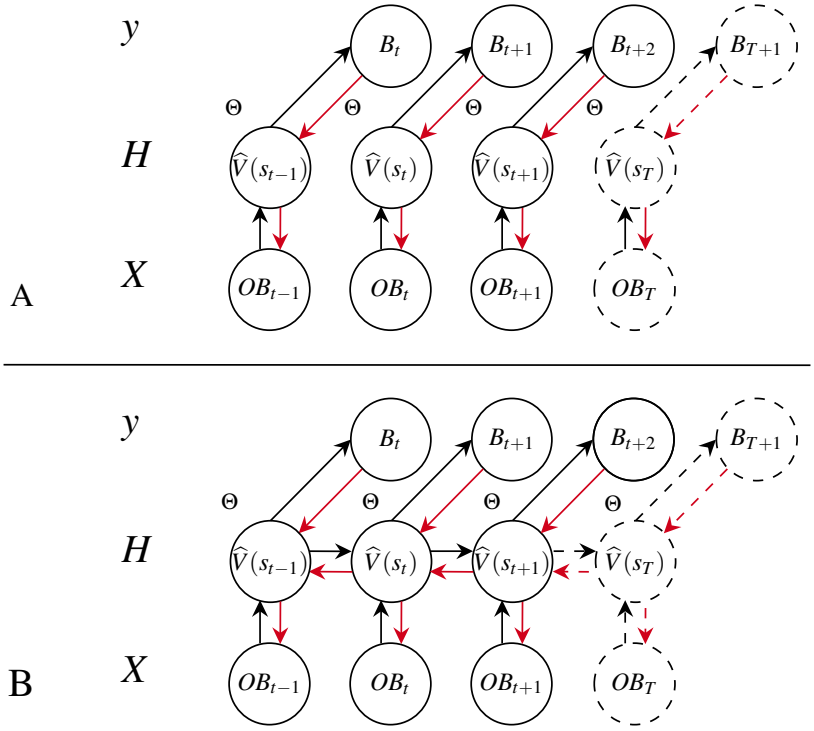


Figure 3.4: Adapted from [28, 172], the figure represents how feedforward and recurrent ANNs could be used for estimating $V(s_t)$. Here $OB = \{OB_t : t \in T\}$ indicates the series of inputs of length T that the network receives while the target is the lead 1 version of the B portion of the same series. The series $\widehat{V} = \{\widehat{V}(s_t) : t \in T\}$ correspond to the representations generated combining the input with the parameters Θ learned by the network in order to approximate the target. Circles indicate computational blocks similar to those present in figure 3.3. Black and red arrows are respectively the direction of the computations and the flow of the error gradient.

target [28]. Moving along this final manifold allows one to reach inputs with different characteristics leading to variations in the predictions produced by the model.

We hypothesize that the amount of attributed incentive salience (i.e. $V(s_t)$) can be modeled as a manifold on which the history of individual-object interactions is placed in order to best predict the intensity of all future interactions. This relates to the concept of motivation as a vector presented in sections 2.2: the representation $V(s_t)$ estimated by an ANN can be thought of as a vector in an h dimensional space, where h is the number of units of the layer producing the representation, indicating the amount of attributed incentive salience after observing t interactions. As we can see, differently from completely un-supervised approaches this approach forces the learned manifold to obey to specific representational and predictive functionalities that are shared with the construct of attributed incentive salience.

Given the potentially large number of layers in an ANN, locating this representation and most importantly ensuring that it is a suitable approximation of $V(s_t)$ are potential issues. A possible solution is to impose a form of architectural constraint on the optimization process through multi-task learning. Multi-task learning closely resemble multivariate analysis, it works on the assumption that a common latent factor underlying a set of targets exists and it can be constrained in a single representation used by the ANN for producing multiple predictions [28]. An example of this process

is shown in figure 3.5. As mentioned in section 2.2.2, the amount of attributed incentive salience $V(s_t)$ that an individual I assigns to an object O should be a latent factor that indicates how intense future interactions with that object will be. Therefore, if a layer in an ANN is forced to produce a single representation which is then used to estimate multiple behavioural indicators of the intensity of these interactions, this should provide a sensible approximation of the amount of attributed incentive salience.

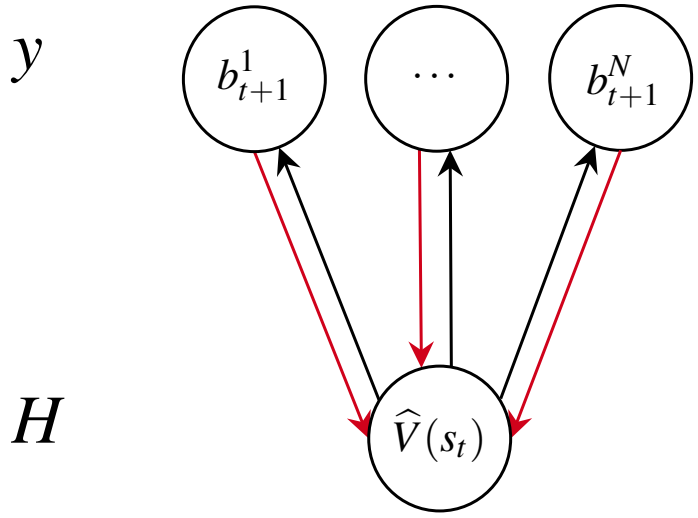


Figure 3.5: Adapted from [28]. The figure represents how multi-task learning could be used in an ANN to force the the latent representation h to be a sensible approximation of $V(s_t)$. Here $\widehat{V}(s_t)$ indicates the representation generated by a recurrent layer at time t while $B_{t+1} = \{b^n_{t+1} : n \in N\}$ are N targets quantifying the strength of the next interaction (in terms of frequency and amount of behaviour) between I and O. Black and red arrows are respectively the direction of the computations and the flow of the error gradient. Circles indicate computational blocks similar to those in figures 3.3 and 3.4.

As we mentioned before, in order to generate a latent representation that faithfully approximate the functionality of attributed incentive salience, an ANN should be fitted simultaneously across multiple O. This can be achieved through what is known as a global model [175] as represented in Figure 3.6.

These models can be thought as an extension of mixed effects models [176] to ANNs and assume that the data on which the model is fitted are drawn from a hierarchy of different populations, each one with its own heterogeneity which can however be brought back to a common single latent factor. The concept is similar to that of Bayesian hierarchical models [177] and has the effect of allowing information sharing across the different levels of the hierarchy while also promoting regularization (and therefore generalization) [177]. The underlying mechanisms is known as partial pooling [177] and can be seen as a middle ground between the over-generalizability of complete pooling (where a single model is estimated across all the levels of the hierarchy) and the over-specificity of un-pooling (where multiple models are estimated, one for each level of the hierarchy). As we can see from Figure 3.6 all the different O contribute to the estimation of a single latent representation $V(s_t)$ which also provide the error gradient for updating their associated parameters.

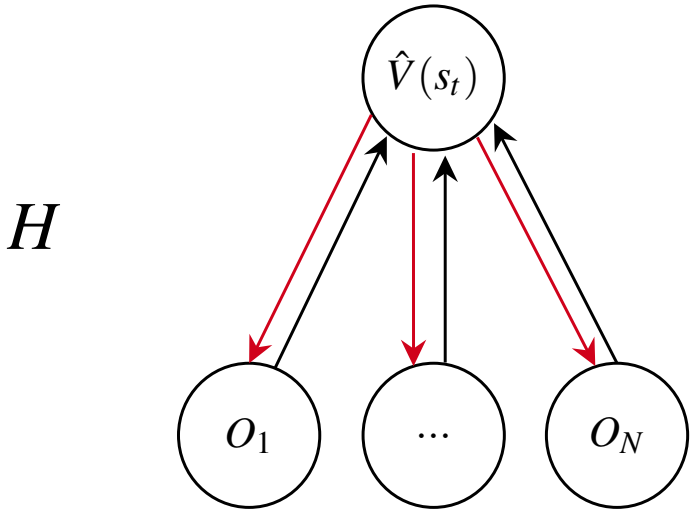


Figure 3.6: The figure represents how multiple O can simultaneously contribute to the estimation of $\widehat{V}(s_t)$. Here $\{O_1,\ldots,O_N\}$ indicate a set of object on which the ANN is simultaneously fit. Black and red arrows are respectively the direction of the computations and the flow of the error gradient. Circles indicate computational blocks similar to those in figures 3.3 and 3.4.

3.3.5 Modelling the contribution of Game Events and Environment Indicators

As we mentioned in chapter 2, the current amount of attributed incentive salience is in part conditioned by the characteristics of the game (e.g. the specific mechanics defining the core playing experience) while its behavioural manifestation can be affected by the environment surrounding the individual (e.g. whether a play session happened during a working day or a weekend). These two factors are not taken into consideration in the modelling framework we just proposed, indeed we assume them to be absorbed, to a certain extent, by the behavioural indicator B. If the game characteristics are not able to provide sufficient rewarding experiences or some environmental factors act as an impediment, we can expect to observe less intense and frequent interactions between I and O. However in this setting it would be hard to disentangle which factors contributed to the reduction in observed behaviour: was it due to a long-running decline in the level of attributed incentive salience or to an unfavourable conjunction of in-game and out-game events?

In this view introducing historical information about the interaction that an individual had with particular characteristics of the game object and the environmental context in which they occurred should not just improve the prediction of $\widehat{B}_{t+1}$ but also the estimation of $\widehat{V}(s_t)$. Of course introducing and modelling an extra set of predictors in an ANN is relatively straightforward and increasing the number of available parameters should help incorporating the additional information, however this would make the interpretation of the derived latent representation even more challenging. A possible solution to this problem is to modify the architecture of the ANN in order to integrate the additional information in a similar way to what Generalized Additive Model (GAM) would do [178].

Solving the problem of predicting B_{t+1} within a GAM framework for a single game object could have the following formulation:

$$\mathbb{E}[B_{t+1}] = \beta + f^B(B_{t:t-n}; \theta^B) + f^G(G_{t:t-n}; \theta^G) + f^{Env}(Env_{t:t-n}; \theta^{Env}) \tag{3.12}$$

here β is a generic intercept while $B_{t:t-n}$, $G_{t:t-n}$ and $Env_{t:t-n}$ are temporal series of behavioural metrics, game events and environment indicators up to time t. The corresponding functions f^B, f^G and f^{Env} are usually linear or non linear (e.g., splines) additive models. The framework offered by GAM is a good compromise between complexity and explainability as it allows to consider many predictors while enforcing a structural form that makes the function associated to each of them directly interpretable [178]. In a work by Agarwal et al. [179] the authors extended the GAM framework to ANN (Neural Additive Models, NAM), the general idea behind this is to substitute each function expressed in equation 3.12 with an appropriated ANN and to derive the prediction of the model as a linear combination of them. The additive nature of NAM, despite posing a constrain on how each function gets integrated with the others, makes it possible to know exactly how each function contribute to the final prediction [148, 178].

In our setting we are less interested in retrieving the exact functional form associated to any of the three sources of information (i.e., behaviour, game characteristics and environment) and more in being able to generate separable representations for each one of them. In this view, combining the functional form of equation 3.12 with the ideas presented in this chapter we could formulate the estimation of $V(s_t)$ as a non-linear combination of separate functions (all parametrized by ANNs) and subsequently use this for predicting B_{t+1}

$$\begin{aligned}
\widehat{B_t} &= f^B(B_{t:t-n}, O; \theta^B) \\
\widehat{G_t} &= f^G(G_{t:t-n}, O; \theta^G) \\
\widehat{Env_t} &= f^{Env}(Env_{t:t-n}, O; \theta^{Env}) \\
\widehat{V}(s_t) &= f^V(\widehat{B_t}, \widehat{G_t}, \widehat{Env_t}, \widehat{V}(s_{t-1}); \theta^V) \\
\mathbb{E}[B_{t+1}] &= f^{out}(\widehat{V}(s_t); \theta^{out})
\end{aligned} \tag{3.13}$$

here $\widehat{B_t}$, $\widehat{G_t}$, $\widehat{Env_t}$ and $\widehat{V}(s_t)$ are representation generated by the respective functions f. Each function can be though as being parametrized by a recurrent ANN as specified in section 3.3.3. The final representation $\widehat{V}(s_t)$ would then be used for solving the same type of multi-task learning problem presented in section 3.3.4. A graphic representation for this can be found in figure 3.7.

3.4 Summary

In this chapter we introduced the idea that latent states, like attributed incentive salience, despite being encoded by high-dimensional signals (e.g. patterns of brain or behavioural activities), can be

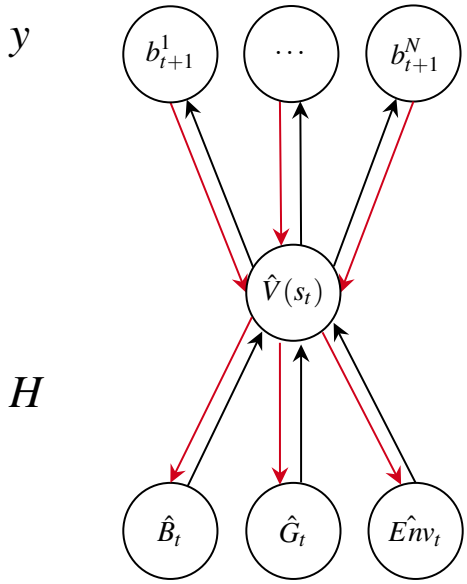


Figure 3.7: Adapted from [28]. The figure represents how multi-task learning could be extended for distinctly modelling the contribution of different factors in the estimation of $\widehat{V}(s_t)$. Here $\widehat{B}_t$, $\widehat{G_t}$ and $\widehat{Env_t}$ indicate the representations generated by 3 distinct recurrent layers taking as input: behavioural, game-events and environmental indicators. The three representations are then parsed by a subsequent recurrent layer producing $\widehat{V}(s_t)$. Similarly to figure 3.5, $B_{t+1} = \{b^n_{t+1} : n \in N\}$ are N targets quantifying the strength of the next interaction (in terms of frequency and amount of behaviour) between I and O. Black and red arrows are respectively the direction of the computations and the flow of the error gradient. Circles indicate computational blocks similar to those in figures 3.3 and 3.4.

effectively approximated by a lower dimensional manifold [7–12, 151].

We then specified how in the literature the modelling and estimation of attributed incentive salience was carried out through reinforcement learning (i.e., TD learning) [13, 14]. This approach allows to specify the dynamics underlying the process of saliency attribution and offers a direct interpretation of the estimated representations. However, its application in complex naturalistic scenarios can be challenging. Leveraging the knowledge presented in chapter 2 in combination with insights derived by previous computational model of attributed incentive salience [13, 14] we proposed to approximate the amount of attributed incentive salience through supervised learning. Due to their reliance on reward mechanics and the ability to provide large volumes of ecologically sound data we thought videogames to be the optimal test bed for our methodology.

For this purpose we designed an ANN able to incorporate information about the state of the individual, the environment surrounding them and their interaction with the game. We argued that ANNs with recurrent operations would be well suited for the task as they generate latent representations through dynamical mechanisms that are similar to those of TD learning [26]. We also stressed the necessity to learn a single model able to simultaneously incorporate information across multiple videogames in order to obtain representations that obey to the same functional constrains of motivation that we specified in chapter 2: namely the ability to simultaneously describe the propensity towards multiple rewarding objects.

In the next chapter we will proceeded at illustrating the implementation of the computational model presented so far and the experimental validation of its underlying assumptions.

Chapter 4

Model Implementation and Performance Analysis

4.1 Introduction

[1] In this chapter we will present the results of the implementation and validation process used for developing the predictive model described in chapter 3. Indeed, relevant to our approach for approximating the latent motivational state of an individual is to have a model able to reliably predict the intensity of future behaviour (i.e., future engagement in a videogame context) given the history of interactions between an individual and a potential rewarding object (i.e., a videogame).

To achieve this, we adopted a variation of bottom-up iterative model building [177] in which first the simplest version of a model is designed, built and evaluated and then, based on performance and addition theoretical assumptions, a new improved version of the same model is proposed. At each stage of the process we evaluated the new version of the model against alternative approaches in order to test for hypotheses stated in the model design stage.

Each section of this chapter corresponds to a different version of the model and will have the following structure.: In **Model Design** we will state the task the model is trying to solve, the design of such model and the theoretical assumptions that informed the design. In **Data** we will describe the dataset used for evaluating the performance of our model along with any data-related processing. In **Model Comparison** we will outline the alternative approaches against which the current model is compared and the various procedures and statistical analyses employed. In **Results** we will report the outcomes

[1] All the code, excluding the SQL queries used for gathering the data, associated with the work presented in this chapter can be found in the following repositories

- ChurnSurvivalJointEstimation
- ApproxIncentiveSalience
- ImprovedApproxIncentiveSalience

of the model comparison procedure with particular focus on the assessment of the predictive accuracy. In **Model Criticism** we will discuss what presented in the results section,in light of the theoretical assumptions used when designing the model, and highlight potential improvements to be carried out in the subsequent stage of the model building process.

Despite the **Model Comparison** stage differed slightly between the various stages of the model building process, a common experimental pipeline was adopted. We can see a graphical representation of the latter in Figure

4.1

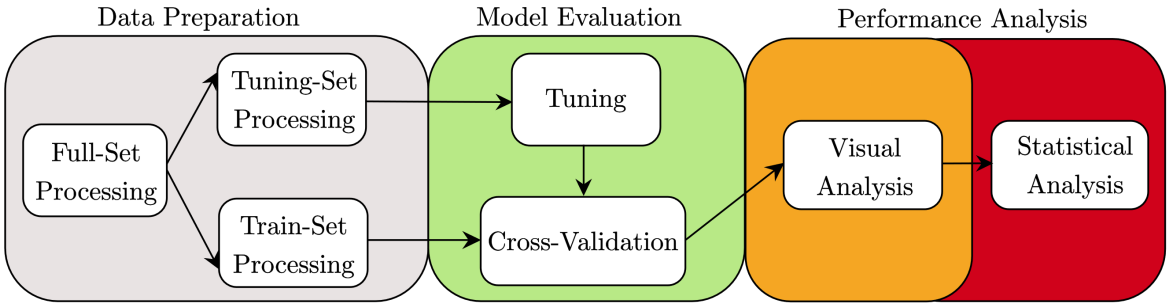


Figure 4.1: Arrows indicate the flow of the pipeline. Big coloured blocks are major pipeline steps, white rectangles indicate sub-tasks within each step.

4.2 Joint Prediction of Long Term Behavioural Intensity

In the first iteration of our model building process we focused on implementing and testing some of the core theoretical assumptions presented in chapter 3, namely the importance of modelling temporal and non-linear dynamics in the interactions between an individual and a videogame.

The experimental task used for this purpose aimed at predicting the long term intensity of interactions between an individual and a videogame given a set of behavioural metrics recorded during an initial observation period. Here, the observation period is defined as a small sequence of initial interactions between an individual and a videogame. In this specific context, we used churn and survival time as behavioural proxies for the expected intensity of future interactions. We briefly introduced these two concepts in section 2.5.2. More formally we can say that given a sequence of interactions with associated behavioural intensity $B_{t_1:T}$, survival time is the amount of expected future playing activity at a given point t_n [106, 115–118]

$$survival = \sum_{t_n}^{T} B_t \tag{4.1}$$

while churn is the probability of observing a terminal event C (usually formulated as a prolonged

period of inactivity from the individual) after a given point t_n [105–107, 113, 114].

$$churn = \sum_{t_n}^{T} C_n = 1 \tag{4.2}$$

here the interactions going from $t_1 : t_n$ define our observation period. It has to be noted that this task can be viewed as a special case of the more general formulation presented in section 3.3.1, precisely

$$\mathbb{E}[\sum_{t_n}^{T} B_t] = f(B_{t_1:t_n}; \theta) \tag{4.3}$$

and correspond to predicting extinction and future amount of sustained engagement after observing a handful of interactions following the point of engagement (according to O'Brien et al. [61] framework presented in section 2.3.2).

Given this experimental context, the hypotheses that we wanted to evaluate in order to support our modelling intuitions were the following:

1. Leveraging all the information provided by the behavioural data defining the observation period is required for achieving higher predictive power.
2. Approaches able to account for non-linear interactions in the considered behavioural metrics can outperform simpler additive models.
3. The ability to model the type of sequential dynamics presented in section 3.3.1 can have a positive effect on the predictive performance.
4. It is possible to have a single model jointly predict the considered metrics of long term behavioural intensity without any loss of predictive accuracy.
5. It is possible to have a single model performing the predictive task simultaneously across a wide range of games.

4.2.1 Model Design

The first iteration of our ANN architecture, loosely inspired by the winning entry of the 2018 CIG churn and survival time competition [129], aimed to jointly predict survival time and churn probability across a set of 6 different game contexts. This architecture, the 'Bifurcating Model' (BM), is illustrated in Fig. 4.2.

The model receives as input a set of variable length multivariate time series composed of 5 behavioural metrics described in section 4.2.2, resulting in an $B \in \mathbb{R}^{N \times T \times 5}$ tensor where N is the number of time series and T is the length of the longest series. All the series shorter then T are made of the

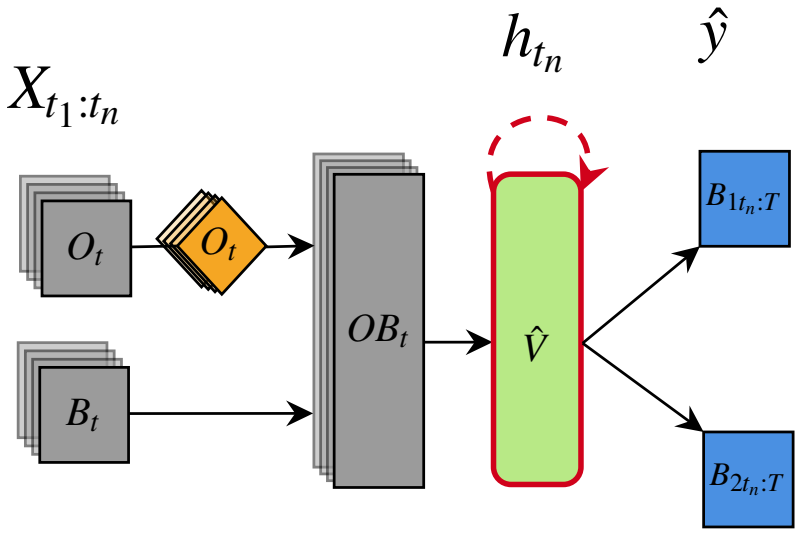


Figure 4.2: Blue, orange and green shapes represent respectively feedforward, embedding and LSTM operations. Gray shapes indicate operations with no learnable parameters, such as tensor instantiation and concatenation. Stacked, transparent colouring indicates tensors with a sequential structure. Straight and curved arrows refer to the presence of feed-forward or recurrent information flow. The red highlight shows the portion of the model we hypothesize is inferring an approximation of attributed incentive salience.

same length by adding a padding value $pad = 9999$ at the end (this value will never appear in the dataset).

An additional set of univariate time series of shape $O \in \mathbb{N}^{N\times T\times 1}$ is used for indicating to which game context the behavioural metrics belongs to. It has to be noted that these series contain numerical encoding of the game context (e.g. $jc3 \mapsto 1$, $lis \mapsto 2$ etc.) in order to be able to represent the associated information through a so called embedding operation (see Appendix A.21). Given the input series, the operation will generate an $O^* \in \mathbb{R}^{N\times T\times h}$ tensor with h being the number of hidden units chosen for the embedding. This can be thought as a form of over-parametrization that allows each single context to have a non-sparse representation and to be projected into a multi-dimensional space where the relationships between elements become meaningful (e.g. game contexts which are similar to each other in respect to the objectives will be located closer to each other in the embedding space). This, other than allowing each context to carry more information, provides a general representation that becomes more effective the more categories are included into it. Obviously this would require to re-fit the model whenever a new unseen context is added as this approach does not support out-of-sample generalizability. It is important to highlight that this embedding operation is fundamental to the construction of a global model as outlined in section 3.3.4. Indeed, it allows to appropriately associate parameters to all the game contexts taken in consideration and include and optimize them as part of a single model.

Next, the raw behavioural input and the embedded game context vector are concatenated along the temporal dimension producing a single $OB \in \mathbb{R}^{N\times T\times h+5}$ tensor and a masking operation is applied. This operation allows the model to skip computations whenever the *pad* value is encountered [180].

These newly obtained features are then modelled across time using a recurrent neural network (RNN) with Long Short-Term Memory (LSTM) operations (see Appendix A.22). In this specific setting the RNN is of type many-to-one [28] therefore producing an $H \in \mathbb{R}^{N \times h}$ matrix (with h being the number of hidden units chosen for the recurrent operation). As we already highlighted in section 3.3.1 the use of an RNN in this context is of pivotal importance for modelling the sequential dynamics underlying the process of incentive salience attribution. Indeed, if we think of the latent representation generated by the RNN in terms of expectation

$$\mathbb{E}[B_{t_n:T}] = f(h_{t_n}; \theta) \tag{4.4}$$

what the LSTM operation allows us to do is to estimate h_{t_n} as the result of a process with memory where the intensity of previous interactions with a specific game determines the expected intensity of all future interactions.

The final step of this architecture uses the matrix generated by the RNN for producing predictions for survival time and churn. This is accomplished by means of multi-task learning with two ANNs with fully connected operation (see Appendix A.19) receiving the matrix generated by the RNN as input and producing the predictions for the relative behavioural targets in the form of a vector of shape $\hat{y} \in \mathbb{R}^N$.

The model used *ReLU* as activation function (see Appendix A.4) for the hidden units whereas the link functions used for generating churn and survival time predictions were the *identity* (see Appendix A.1) and *sigmoid* (see Appendix A.2) functions. We applied two regularization techniques after each fully connected operation, namely: batch normalization [181] and dropout [182] (see Appendix A.13 and A.15).

Competing Models In order to test the hypotheses mentioned in section 4.2 we implemented a series of competing models for disjoint estimation of survival time and churn probability.

Furthermore, we decided to include a baseline model (MM) which generates predictions based on the average of the targets in the training set. The presence of a baseline which does not require any form of parameters estimation provides a sanity check on the overall quality of the models and complexity of the problem at hands (e.g., if a predictive task is trivial to solve we expect a relatively satisfactory level of accuracy even by a naive baseline model).

The competing models were chosen according to two main criteria: the ability to capture linear and non-linear interactions between features and most importantly the capability to fit large data-sets (e.g. matrix of dimension $x \in \mathbb{R}^{\approx 10^6 \times 10^2}$). According to this criteria we opted for linear regression with ElasticNet regularization (see appendix A.18) (EN) [183], logistic regression (LR) and two Multi-Layer Perceptrons using *id* (MLPr) and *sigma* (MLPc) as link functions. Multi-Layer Perceptrons are ANNs using the type of feedforward operation described in Appendix A.19. Given the similarities between linear models and ANNs, which can be seen as a stacked version of the former but with

more "expressive power", the chosen algorithms constituted a natural progression in the modelling approach.

4.2.2 Data

In order to evaluate the performance of the different models in the experimental predictive task, we needed to acquire records of interactions between individuals and potentially rewarding objects in naturalistic contexts.

As mentioned in section 3.3.2, video games are particularly suited for this purpose given their learning-dependent reinforcing properties and the large amount of longitudinal data streams that they can generate. For this reason we gathered data from six different games published by our partner company, *Square Enix Ltd*. Focusing on maintaining heterogeneity in genre and platform, we considered the following titles: *Hitman Go* (hmg), *Hitman Sniper* (hms), *Just Cause 3* (jc3), *Just Cause 4* (jc4), *Life is Strange* (lis), and *Life is Strange: Before the Storm* (lisbf). A general description of each of these titles can be found in Table 4.1. Due to the diversity in their in-game mechanics, each of these games was considered as an "object" with different reinforcing properties (see section 3.3.2). This allowed us to mimic a situation where a single model had access to data coming from a heterogeneous set of potentially rewarding entities (similarly to what we described in section 2.2.1).

Data were gathered from any user playing between the game's release and February 2019, allowing us to adopt more robust sampling strategies which for each game utilizes the breadth of virtually the entire user-base. To rule out possible "faulty" but not "naturally abnormal" data, we restricted the data cleaning process to a single filter applied at query time to ignore users having at least one of the considered metric over the game population's 99^{th} percentile. This allowed us to make little assumptions on the distribution of the data as well as providing a convenient stress test for eventual future real-world applications.

Defining the Observation Period Because we were interested in predicting survival time and churn probability based only on a restricted number of early individual-game interactions it was important to define a cut-off for what would be considered an "early sequence of interactions" (the so called observation period (OP)).

Choosing the length for the OP was not a trivial task as there is little indication in the literature about optimal cut-off values. Hence, we decided to visually inspect the data a-priori and extend rules proposed in [113, 114] to take into account natural inter-individual differences. Therefore, we defined the cut-off as:

$$\text{cutoff} = \left\lceil \frac{min(T, completion_t)}{3} \right\rceil \tag{4.5}$$

Where T is the total number of game play sessions and $completion_t$ is the number of game play

Table 4.1: For each game we retrieved 80,000 Churners and 80,000 Non-Churners randomly sampled from all the available users.

Game	Survival Time (Mins)		Churners	Non Churners	Observation Period		Type
	Min	Max			Min	Max	
hmg	11	260	80,000	80,000	1	7	Mobile Strategy
hms	2	454	80,000	80,000	1	15	Mobile Shooting Gallery
jc3	32	12,695	80,000	80,000	1	20	Console Action Open World
jc4	7	1,135	80,000	80,000	1	9	Console Action Open World
lis	5	704	80,000	80,000	1	6	Console Graphic Adventure
libf	14	1,214	80,000	80,000	1	10	Console Graphic Adventure

sessions before the user completed the game for the first time. In this way we take the first $\frac{1}{3}$ of all played sessions for players who churned and the first $\frac{1}{3}$ of played sessions before a non-churning player completed the game for the first time. We apply this cut off to the ordered list of all recorded play sessions for a specific user. We decided to use game sessions as the temporal dimension, rather than total minutes played, since we believed it better adjusted for each user's "pace" (i.e. not all the users have the possibility to play at the same frequency).

Since the length of the OP has a naturally different distribution between the churning and non-churning population, we stratified our sampling technique to maintain a similar ratio of OP lengths among churners and non churners. This becomes particularly relevant when employing models that leverage the entire sequence of interactions where the length of the OP could leak information about the considered targets (e.g. churn is on average associated with a small number of interactions).

Summarizing, if an individual had 9 total sessions recorded, we considered the first 3 for making predictions on what happened from the 4^{th} to the 9^{th} session.

Defining the Behavioural Metrics and Targets We considered a set of 5 metrics, easily generalizable across games and comparable with metrics employed in other behavioural studies of incentive salience attribution [13, 14, 31], and retrieved them temporally (i.e. over each game session during the

OP), see Table 4.2 for a description. Additionally, we acquired a single context feature specifying the game context from where the metrics were originated.

For determining the targets for our survival and churn estimation tasks, we leveraged existing literature on churn prediction [105–107, 113–115, 129, 134] and survival analysis [116–118, 129], extending existing rules to accommodate the need to define churn and survival time in single player games with a defined life cycle (i.e. non Games-as-a-Service GaaS). Indeed, while GaaS can only rely on inactivity periods for determining churn, titles with a defined life cycle (e.g. single player games) can utilize a defined end-game period as a hard cut-off for distinguishing between churners and non-churners: users finishing a game are not churners even if they stop playing afterwards.

In this view, we took advantage of having access to the complete session history for all users to create a churn definition which was robust to the variance in play patterns across games by taking into account all the recorded inter-session distances. The criteria we adopted for defining a user as churner were both:

1. Not completing the game
2. Being inactive for a period equal or greater to:

$$\text{inactivity} = mean(\mathbf{x}) + 2.5 \cdot std(\mathbf{x}) \tag{4.6}$$

For better adjusting for inter-individual differences, we could have applied formula 4.6 to each user individually but this could have created accuracy issues for individuals with very few recorded sessions. Therefore, we opted for a conservative but more robust approach applying inactivity ($\mathbf{x}$) $\forall \mathbf{x} \in X$ where X is the collection of all the considered games and $\mathbf{x}$ is the vector of inter-sessions distances in minutes for a specific game. The use of formula 4.6 allowed us to estimate an inactivity period which was not arbitrarily chosen but statistically defined as “extraordinary long” in accordance with characteristics of play patterns in a particular game. For defining the survival time, we simply computed the total amount of play time in minutes for a user minus the amount of play time during the OP.

Table 4.2: Description of selected telemeteries

Metric	**Description**
Absence	Temporal distance between sessions (hours)
Session Time	Overall session duration (minutes)
Session Activity	Count of user initiated gameplay-related actions. E.g. ”Attack an enemy” is considered a valid action while ”Being attacked by an enemy” is not.
Session Diversity	Number of distinct gameplay-related actions
N°Sessions	Number of played sessions.
Context	Game context identifier.

Data Preparation We adopted specific data preparation for testing the various hypotheses specified in section 4.2.

We first generated a dataset collapsing the original data over the temporal dimension retrieving mean and standard deviation of each considered metric and adding a one-hot encoded (see Appendix A.20 version of the game context indicator. Then we generated a second data-frame maintaining the original temporal structure and numerically encoding the game context indicators. As we mentioned before, different users had OP of different lengths so we applied the *pad* value to each metric column in the data-set so to have each sequence of considered sessions to the length of the longest sequence in the data-set.

For each experiment we created a tuning and test subsets (i.e., 20 and 80 % of the original data-set) via stratified shuffle split [184], employing the first for hyper-parameters tuning and the second for model evaluation. Each time a model was fit on the considered dataset, we re-scaled each metric separately for each game in an outliers-robust way:

$$\text{RobustRescale} = \frac{\mathbf{x} - Q_2(\mathbf{x})}{Q_3(\mathbf{x}) - Q_1(\mathbf{x})} \tag{4.7}$$

where $\mathbf{x}$ is the feature vector to be re-scaled and Q_n is the n^{th} quartile.

4.2.3 Model Tuning and Comparison

For all the considered models we first, determined the best hyper-parameters via simple grid search 10-fold stratified cross validation [184] on the tuning set. At each iteration of the 10-fold stratified cross validation a small sub-set of the fold used for fitting the model (i.e., 10%) was set aside and used to evaluate model convergence and eventually trigger an early-stopping policy (so to avoid over-fitting).

For all the models convergence was determined by the loss not improving for at least 3 consecutive epochs. The models were trained though mini-batch stochastic gradient descent (using a batch size of 256) using the Adaptive Moment Estimation (ADAM) optimizer [185] with learning rate adjusted through a cyclical policy [180, 186]. We used Mean Squared Error (MSE) and Binary Cross Entropy (BCE) respectively (see Appendix A.7 and A.9) as objective functions for the disjoint models (i.e., linear models and MLPs). Differently from the other models the objective function for the the BM model was given by the unweighted sum of the losses associated with the two branching ANNs, namely BCE for the churn prediction task and the Symmetric Mean Absolute Percentage Error (SMAPE) (see Appendix A.8) for the survival prediction task. The SMAPE is bounded between 0 and 100 and can be interpreted as percentage deviation from the target with lower values indicating better model fit. The choice of SMAPE was motivated by the fact that its scale invariance allowed better comparisons of results across game contexts.

For EN the best hyper-parameters were $\lambda = 0.1$ and $alpha = 0.5$ for the regularization term regu-

larization. For LR a $\lambda = 0.01$ for the lasso regularization was found to yield the best results. Both MLPr and MLPc employed an Ridge regularization with $\lambda = 0.01$ and utilized a 3 layers architecture with 200, 100 and 50 hidden units. The BM architecture used an embedding with 40 hidden units, an LSTM RNN with 100 hidden units and two fully connected ANNs with 300 hidden units each. The best dropout rate was found to be $p = 0.1$ meaning that at each forward pass 1 in ten units for the fully connected ANNs were set to 0.

Once the best hyper-parameters were found, each model was evaluated on the test set by fitting it on 90% of the data and performing out-of-sample prediction for the remaining 10%. All the disjoint models were tested on both the collapsed and unfolded datasets while the BM model only on the last one. This is because the BM model was specifically designed for performing predictions using data in a time series format.

Moreover, following the intuition from [187], the BM model supported Montecarlo dropout: a method for approximating Bayesian inference and producing posterior distribution of the model's predictions. This was achieved by querying the model 50 times at prediction time and retaining all the produced values. When computing the performance metrics we then used the mean of the estimated values, since they were expected to roughly follow a normal distribution.

For the churn prediction task the chosen metric was the macro-averaged F1 score (see Appendix A.10) (i.e., employing the unweighted mean of precision and recall for both classes) while for the survival time prediction task the chosen metric was the SMAPE.

The code for tuning and evaluating the models was written in Python 3.6, with the algorithms provided by the library scikit-learn [184] and Keras (with Tensorflow as a back-end) [180].

4.2.3.1 Computational Resources and Run Times

The experiment was conducted on a single machine with 8 GB of RAM and a CPU with two 2.5 GHz cores. Training was carried out with no GPU acceleration. The run time for the experiment (hyper-parameters tuning and model comparison) was of roughly 8 hours.

4.2.4 Results

We will first present results for each disjoint model as well as for the baseline model. Next we will illustrate in detail the performance of the BM model both in terms of its raw accuracy and its capability to include uncertainty in its predictions. Note that for all reported SMAPE results the smaller the better as it represents the error between the prediction and ground truth. Conversely, for F1 the larger the better since it measures the discriminate power of the models on a classification task. The probability threshold employed for discriminating between classes was set to $p = 0.5$.

We want to highlight that, given our chosen model evaluation strategy, all the results presented here report the expected value (i.e., the mean) of a scoring metric over a single hold-out test hence conventional sample-based statistical analysis are not feasible. However, in order to quantify the

uncertainty in the computed evaluation metrics we reported the standard error of the mean along with the expected value. For example in the case of SMAPE we can see from the formula in Appendix A.8 that the metric computes the empirical mean over the Symmetric Percentage Error (SAPE), in order to obtain a measure of uncertainty of this estimate is sufficient to compute $SE_{SAPE} = \frac{\sigma_{SAPE}}{N}$ with σ_{SAPE} being the standard deviation of the SAPE computed over the entire test set N. Moreover in order to gather a general sense of global performance, in Figure 4.3 for each model we visualized the expected performance collapsing on game context.

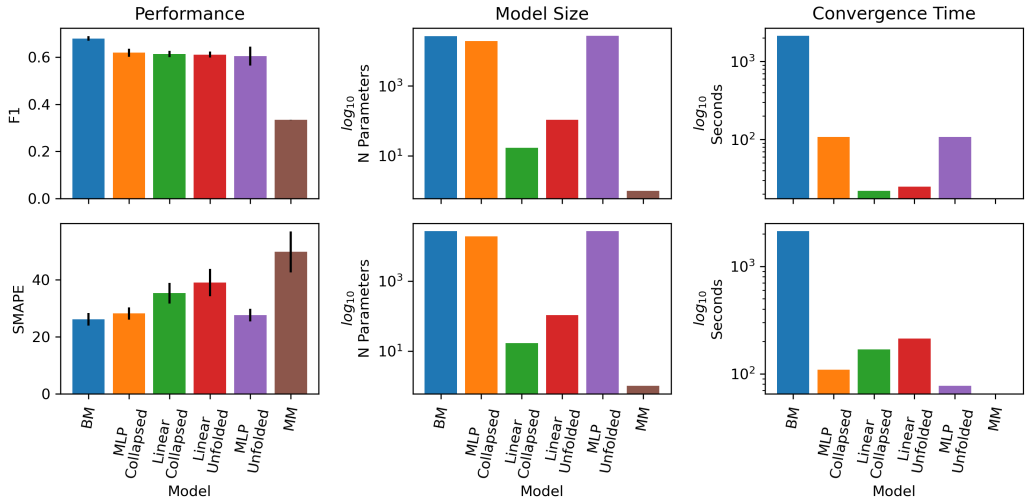


Figure 4.3: The BM architecture outperformed all competing ones on both target using however more parameters and computation time. Bar-plots in the first column show the average performance across the 6 game contexts along with estimated standard errors of the mean. The bar-plots on the second and third columns express the number of parameters and the convergence time for each model.

Table 4.3: **Performance Baseline Mean Model**

Game	Model	SMAPE	F1
hmg	**MM**	76.7 ± 0.1	0.500 ± 0.003
hms		58.1 ± 0.1	0.507 ± 0.003
jc3		63.2 ± 0.3	0.499 ± 0.004
jc4		36.6 ± 0.2	0.499 ± 0.001
lis		40.4 ± 0.1	0.500 ± 0.003
lisbf		24.4 ± 0.2	0.500 ± 0.005

The results from the predictive task carried out using the collapsed dataset, Table 4.4, showed how all the 4 models strongly outperformed the MM baseline, Table 4.3, in all games, while also achieving an overall satisfying performance. Moreover we noticed how MLPr and MLPc markedly outperformed EN and LR in both churn probability and survival time prediction across all games.

Looking at the same modelling approaches on the unfolded version of the same dataset we can observed a similar pattern of results, see Table 4.5, regarding baseline and inter-models comparisons. However, it was clear that using unfolded, temporal data lead to only small improvements over the aggregated data. This might be explained by the fact that the chosen modelling approaches are not

Table 4.4: **Performance Collapsed Format**

Game	Model	SMAPE	Model	F1
hmg	**EN**	51.3 ± 4.3	**LR**	0.591 ± 0.004
hms		33.1 ± 2.0		0.624 ± 0.004
jc3		42.3 ± 0.8		0.601 ± 0.004
jc4		35.1 ± 0.6		0.663 ± 0.002
lis		28.7 ± 0.4		0.626 ± 0.003
lisbf		23.9 ± 0.3		0.591 ± 0.003
hmg	**MLPr**	30.4 ± 0.8	**MLPc**	0.660 ± 0.006
hms		24.1 ± 0.7		0.670 ± 0.006
jc3		36.0 ± 0.3		0.654 ± 0.004
jc4		33.4 ± 0.2		0.678 ± 0.004
lis		25.6 ± 0.3		0.664 ± 0.003
lisbf		21.9 ± 0.2		0.622 ± 0.003

explicitly designed for taking temporal structure into account, for example they have no explicit mechanics for temporal modelling such as those provided by a LSTM.

Table 4.5: **Performance Unfolded Format**

Game	Model	SMAPE	Model	F1
hmg	**EN**	54.5 ± 2.4	**LR**	0.612 ± 0.004
hms		55.0 ± 2.0		0.626 ± 0.004
jc3		38.4 ± 0.3		0.607 ± 0.003
jc4		34.9 ± 0.2		0.660 ± 0.003
lis		30.2 ± 0.1		0.641 ± 0.004
lisbf		23.5 ± 0.2		0.578 ± 0.003
hmg	**MLPr**	29.3 ± 0.4	**MLPc**	0.683 ± 0.005
hms		22.6 ± 0.4		0.682 ± 0.004
jc3		36.0 ± 0.3		0.643 ± 0.004
jc4		33.1 ± 0.2		0.681 ± 0.003
lis		25.6 ± 0.2		0.673 ± 0.005
lisbf		21.8 ± 0.1		0.627 ± 0.003

Indeed, when evaluating the performance of the BM, Table 4.7, on the unfolded data we can see how the model achieves consistent improvements in both churn probability and survival time prediction in all game contexts compared to the previous best model (MLPr and MLPc). From a visual inspection of Figure 4.4

we can see the presence of a positive linear relationship between estimated and ground truth survival time (indicative of accordance between the two), with a roughly even distribution of error along the entire range of values. In Table 4.6 we can observe how the model performance is evenly split across the two classes highlighting similar levels of precision and recall.

Finally, observing the density plots in Figure 4.5a and 4.5b we can see how the model was able to encode different levels of uncertainty in the predicted values through different levels of posterior variance.

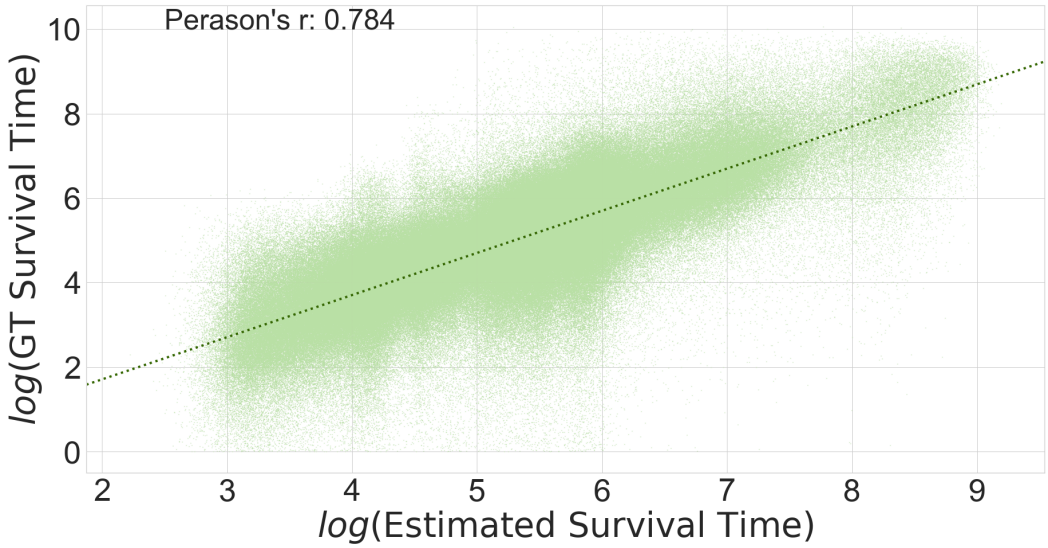


Figure 4.4: The scatter plot shows the relationship between the survival time prediction provided by the BM and the ground truth values. Since the relationship is evaluated on the *log* of both variables, due to the presence extreme outliers in the ground truth, this acts as mostly as a qualitative complement to the more reliable SMAPE measure.

Table 4.6: Here the diagonal shows the % of correct predictions for each label across all games.

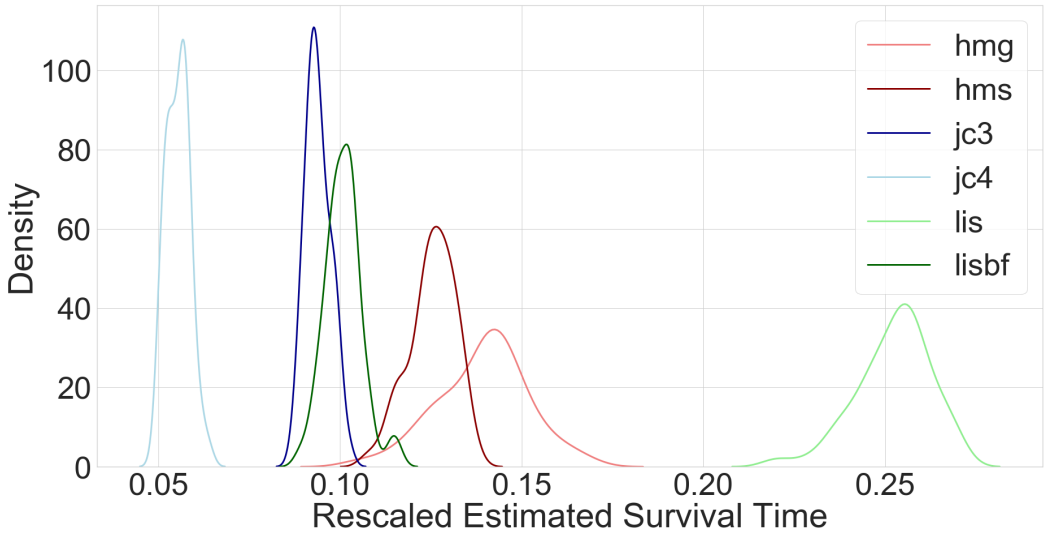
		Prediction	
		Churner	Non-Churner
Ground Truth	Churner	0.69	0.31
	Non-Churner	0.33	0.66

4.2.5 Model Criticism

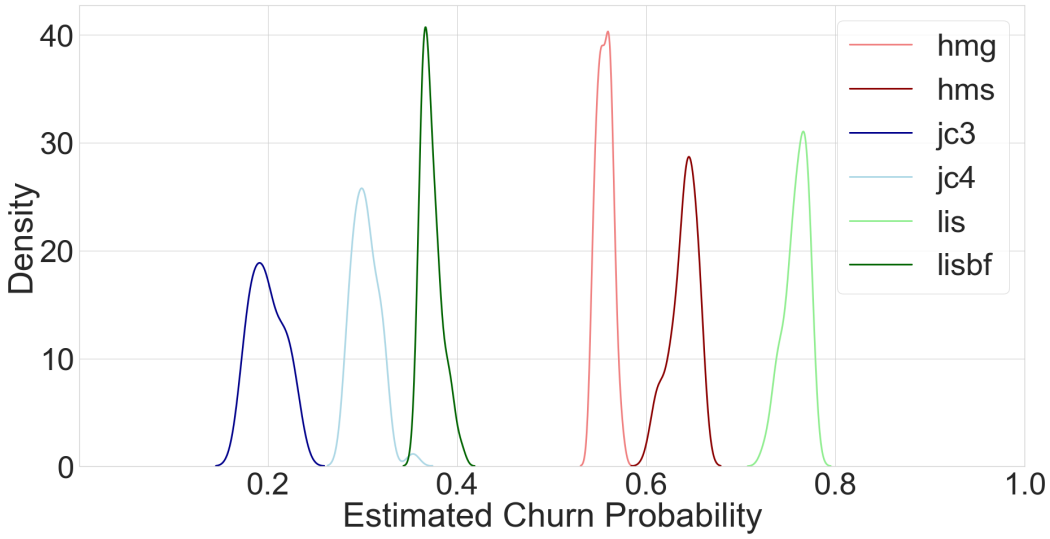
The different level of performance achieved by the baseline model highlighted how different game context proved to be a more challenging ground (e.g. $jc4$ and $lisbf$) than others with respect to the predictive task.

Although we did not perform an exhaustive test for evaluating if disjoint models (i.e. models fitted separately to each game context) would outperform our joint modelling approach, we noticed how each parametric model (which all adopt a joint modelling approach) outperformed the baseline(which adopts a disjoint modelling approach) while maintaining a similar performance distribution across game contexts.

By looking at the performance of the various parametric models we saw how the use of models able to capture non-linear interactions between features provided substantial improvements in predicting measures of future behavioural intensity. We also showed that considering the entire history of individual-game interactions provided a consistent but small edge over metrics representations which are collapsed over time. However, this improvement appeared to be markedly more pronounced and



(a) Survival Predictions



(b) Churn Probability Predictions

Figure 4.5: For better comparison the survival estimates are re-scaled game-wise in the range 0 to 1. The highest density point in the distribution represents the most probable predicted value (i.e. the actual prediction), while the area under the curve instead can be seen as measure of uncertainty (i.e. how confident is the model in its prediction).

consistent when the predictive task was carried out by non-linear models able to take the dynamical nature of the data into account (e.g., the proposed BM architecture). Finally, by looking more in details at the performance of the the BM highlighted how the proposed methodology generalizes well when trying to predict survival time and churn probability as well as successfully incorporating measures of uncertainty in its estimations.

In summary, the BM architecture designed following the theoretical principles outlined in chapter 3 showed improved predictive accuracy when compared to alternatives with comparable computational expressiveness (i.e., the MLP architectures).

That said, it is evident that the architecture does not come fully equipped with all the constrains

Table 4.7: **Performance Bifurcating Model**

Game	Models	SMAPE	F1
hmg	**BM**	27.5 ± 0.1	0.693 ± 0.002
hms		20.0 ± 0.1	0.701 ± 0.003
jc3		34.4 ± 0.3	0.671 ± 0.005
jc4		32.5 ± 0.2	0.685 ± 0.002
lis		24.6 ± 0.2	0.688 ± 0.003
lisbf		20.8 ± 0.1	0.645 ± 0.003

outlined in chapter 3 which are necessary for obtaining a good approximation of the type of construct (i.e., attributed incentive salience) presented in the works of McClure et al. [13] and Zhang et al. [14]. First of all the BM architecture aims at predicting the cumulative expected amount of future behaviour at an arbitrary early point in time $t \ll T$ (the OP defined in section 4.2.1). This implies that, despite the latent representation generated at time t possesses similar predictive powers to attributed incentive salience, these are not on the same time scale and most importantly are not defined for all the $t \in T$. Secondly, despite churn probability and survival time can be considered good approximations of future behavioural intensity, they surely do not fully capture the complexity of the phenomenon. Moreover, these two metrics can be considered of second order (i.e., they are derived from raw quantifiers of behaviour intensity) and artificially created for serving specific concrete applications in the context of engagement quantification.

Since the aim of this thesis is to estimate motivation related latent states (from which, as we said in chapter 2, the behavioural manifestation of engagement can be directly derived) the next iteration in our model building process focused on modifying the BM architecture in order to make its functional form closer to what highlighted in chapter 3.

4.3 Dynamic Prediction of Future Behavioural Intensity

In the second iteration of our model building process we focused on improving and expanding the BM architecture in order to increase its flexibility and ability to incorporate the functional constrains specified in chapter 3.

Two major constrains were introduced:

1. The new architecture had to jointly predict 5 behavioural metrics. These, differently from churn and survival time, were first order indicator of future behavioural intensity similar to those find in the behavioural neuroscience literature [13, 14, 32, 34]
2. We moved the predictive task from a "static" (i.e., predicting a long-term target after observing a fixed sequence of individual-game interaction) to a "dynamic" one (i.e., performing prediction after each new interaction is observed).

In line with these constrains the experimental task used in this second stage of the model building process aimed at predicting the behavioural intensity of the next interaction between an individual and a videogame after observing an arbitrary number of interactions. More formally, given a sequence of interactions with associated behavioural intensity $B_{t_1:T}$, the target of the model at any given point in time t_n became

$$B_{t_{n+1}} = \{b^1_{t_{n+1}}, \ldots, b^5_{t_{n+1}}\} \tag{4.8}$$

with $\{b^1_{t_{n+1}}, \ldots, b^5_{t_{n+1}}\}$ being the lead-1 version of the behavioural inputs provided to the model (more details are provided in sections 4.3.1 and 4.3.2. This new task is now in line with the formulation presented in section 3.3.1 and correspond to predicting the behavioural manifestations of all the phases of the engagement process model [61], however in a dynamic fashion.

The major aim of this second experimental task was to validate the results obtained during the evaluation of the BM architecture while submitting to model constrains that were consistent with the theoretical framework presented in chapter 2 and 3. In this view, the hypotheses that we wanted to validate were tightly connected with those presented in section 4.2, and had the following form:

1. Approaches able to account for non-linear interactions in the considered behavioural metrics can outperform simpler additive models.
2. The ability to model the type of sequential dynamics presented in section 3.3.1 can have a positive effect on the predictive performance.
3. It is possible to have a single model jointly multiple first-order metrics of behavioural intensity without any substantial loss in accuracy
4. It is possible to have a single model performing the predictive task simultaneously across a wide range of games.

4.3.1 Model Design

The second iteration of our ANN architecture was a direct descendant of the BM architecture and aimed to jointly predict a set of 5 behavioural metrics all indicative of the intensity of interactions between an individual and a videogame.

Despite being markedly different, our architecture shares similarities with technique used in the neuroscience literature for inferring latent states able to predict observable behaviours [18]. For example Calhoun et al. used a combination of Hidden Markov Model (HMM) and Generalized Linear Model (GLM) for analyzing the natural behaviour of flies [18]. Similarly to our proposed architecture, the HMM-GLM approach first infer a latent representation of the observed behaviour though a dynamical model (i.e., HMM) and then uses the same representation for performing predictions using a set of non-linear additive model (i.e., GLM). Our architecture, that we call recurrent (RNN) for simplicity, illustrated in Figure 4.6 extends the work of Calhoun et al. by removing the additive and markovian assumptions and by dropping the requirement for a fixed number of hidden states.

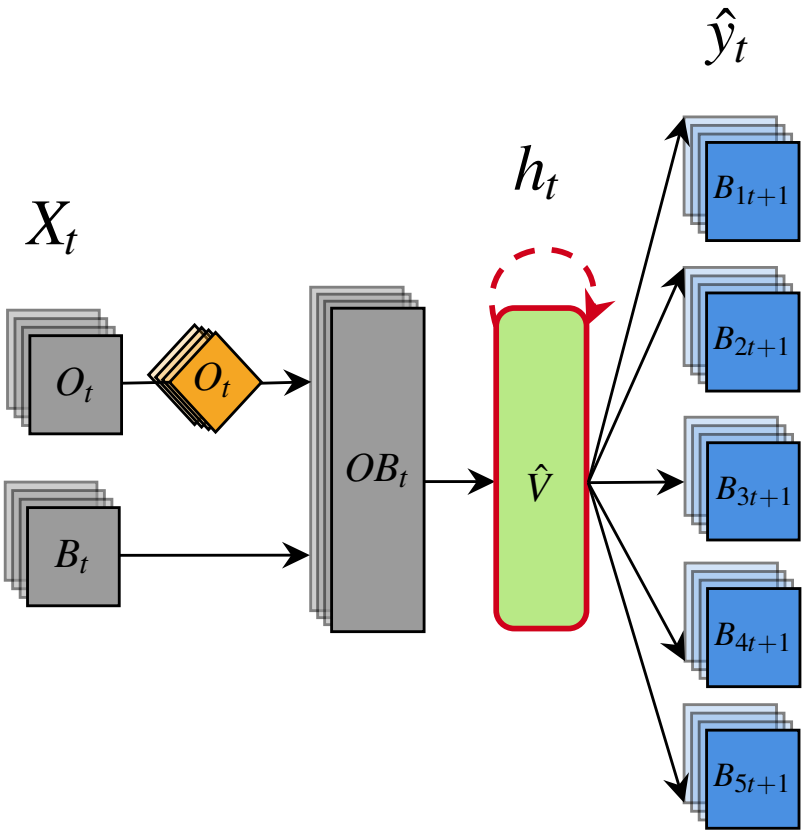


Figure 4.6: Blue, orange and green shapes represent respectively feedforward, embedding and LSTM operations. Gray shapes indicate operations with no learnable parameters, such as tensor instantiation and concatenation. Stacked, transparent colouring indicates tensors with a sequential structure. Straight and curved arrows refer to the presence of feed-forward or recurrent information flow. The red highlight shows the portion of the model we hypothesize is inferring an approximation of attributed incentive salience.

The model receives the same input of the architecture described in section 4.3.1 with the major difference that no padding was required. Indeed we decided to exploit the ability of RNN to flexibly fit variable-length time series and opted for a data generator approach [2], feeding data to the models in random batches with constant length within a batch.

The type of operations performed by the architecture were identical to those presented in section 4.2.1 although with 2 major differences. First, the recurrent component of the architecture was now of type many-to-many therefore producing an $H \in \mathbb{R}^{N \times T \times h}$ tensor constituting the sequence of hidden states produced by the LSTM operation after observing each element of the input time series [28]. Second, the final step of the architecture now exploits the temporal nature of the latent representation generated by the RNN for predicting not just a single value but a sequence of values, namely the lead-1 version of the input time series. This was again achieved by means of multi-task learning with five ANNs with fully connected operations.

In order to apply the fully connected operations to a sequence of latent states we distributed their computations across time [3] (see Appendix A.23) so that each ANN could produce a tensor of shape

[2] See [180, 188] for implementation details.
[3] See [180] for implementation details

$\hat{y} \in \mathbb{R}^{N \times T \times 1}$.

We want to highlight that the model now fully reflect the formulation presented in section 3.3.1, indeed if again we look at the representations generated by the LSTM in terms of expectation

$$\begin{aligned} h_t &= f^1(O, B_t, h_{t-1}; \theta^1) \\ \mathbb{E}[B_{t+1}] &= f^2(h_t; \theta^2) \end{aligned} \tag{4.9}$$

we can can see how at each point in time the generated representation h_t is mostly indicative of the behavioural intensity a time $t+1$ but it also needs to retain some predictive power for all the subsequent time steps.

The model used a combination of *ReLU*, *ELU* and *LReLU* (see Appendix A.6 and A.5) as activations functions for the hidden units (more details on their selection are specified in 4.3.3 whereas the link function used for generating the prediction was always the *identity* function. As a regularization strategy we decided to drop the use of batch normalization, as it proved challenging to adopt with a multivariate time-series target, and introduces a variant of dropout (i.e., One Dimensional Spatial Dropout see Appendix A.14) more suitable for time series data and applied it after each layer in the architecture.

4.3.1.0.1 Competing Models As in the previous stage of the model building process a series of competing models were implemented for testing our hypotheses.

First, we decided to improve our baseline by establishing two reasonable single-parameter models. The first is a Lag 1 model producing predictions according to the following rule:

$$B_{t+1} = B_t \tag{4.10}$$

here t represent a single game session in a sequence of T observed interactions while B are the same behavioural metrics the RNN model receives as input (they are thoroughly described in section 4.3.2). The second is a Median model computing the expectation of each of the 5 targets according to the formula:

$$\begin{aligned} \overline{B_{t+1}} &= \frac{\sum_{i=1}^{t+1} wB_i}{\sum_{i=1}^{t+1} w} \\ B_{t+1} &= median(\overline{B_{t+1}}) \end{aligned} \tag{4.11}$$

here $\overline{B_{t+1}}$ is an exponentially weighted average of all the B_t up to $t+1$ observed when fitting the

model. This is computed separately for every individual in the dataset and the median value of each of the 5 targets is used a a constant prediction. These apparently naive models provide a surprisingly robust prediction baseline for time series that are not white noise [189] other than having a nice interpretation in terms of behavioural momentum [190]: in conditions of high experienced reward the behaviour of an individual tends to be consistent over time (i.e. resistant to change).

Similarly to the model used in the assessment of the BM architecture, a multi-task linear model with Elastic-Net regularization (ENet) [183] was used to evaluate the performance of simple additive models. While for testing the effect of non linearity, a multi-task Multilayer Perceptron (MLP) was employed. Both model used an embedding operation for encoding information from the game context but the ENet model had the number of hidden units fixed to 1 an relied on an *identity* activation function. This was done for avoiding to introduce non-linearity in the ENet model. Figure 4.7 show the architecture used for both the ENet and MLP models. We can see that differently from the models used in section 4.2 the only difference with RNN architecture lies in the use of time distributed fully connected operations instead of those provided by the LSTM.

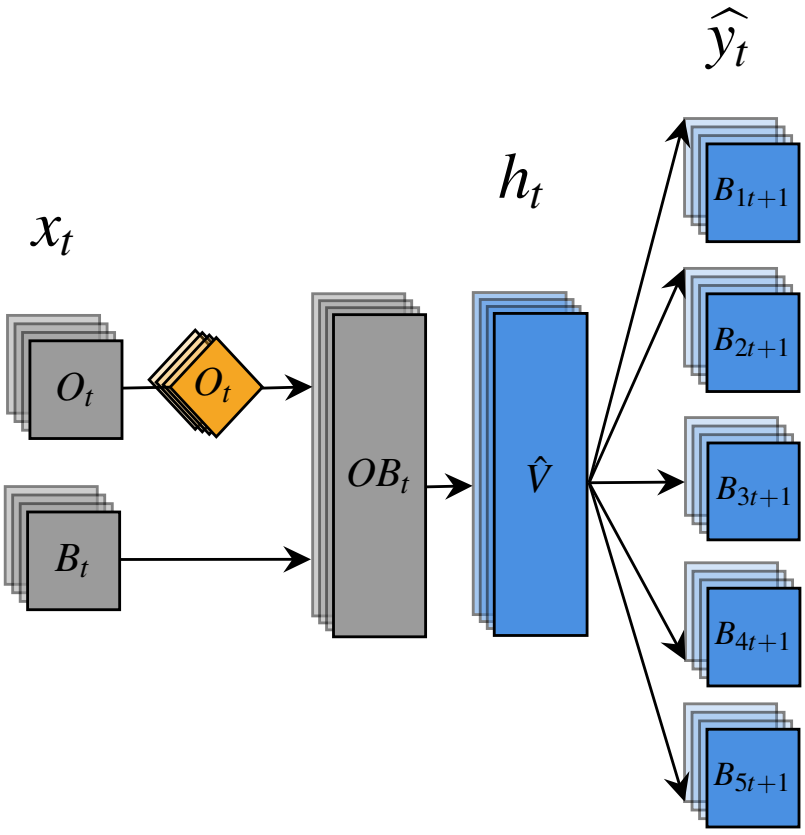


Figure 4.7: Blue and orange shapes represent respectively feedforward and embedding operations. Gray shapes indicate operations with no learnable parameters, such as tensor instantiation and concatenation. Stacked, transparent colouring indicates tensors with a sequential structure. Straight arrows refer to the presence of feed-forward information flow. All the feedforward operations are time distributed.

4.3.2 Data

In this iteration of the model building processes we decided to expand and improve the dataset used for evaluating the performance in the predictive task.

We again used gameplay data from the same six video games published by our partner company, *Square Enix Ltd.* however this time we increased the number of considered individuals by almost 3-fold. The resulting dataset contained entries from 3,209,336 individuals, evenly distributed across the six games, and randomly sampled from all users who played the games between their respective release date and January 2020. All data were obtained and processed in compliance with the European Union's General Data Protection Regulation [1].

Defining the Behavioural Metrics and Targets In order to represent the behavioural manifestation of state transition dynamics (i.e., changes in the level of attributed salience during sequences of interactions between a user and a videogame), for each individual we retrieved a set of six different types telemetry over variable-length sequences of game sessions. A game session was defined from the moment an individual started the game software until it was closed. We retrieved all sessions produced by an individual from the moment the data they generated first appeared in the game's servers. Since our modelling approach required to predict, in a supervised manner, the intensity of future playing behaviour given the history of previous interactions, we only considered users with two or more observed game sessions. The reason for this is two fold: sequences of length one do not entail any temporal structure and do not allow to generate a supervised target.

The telemetry data (see Table 4.8) were almost the same of those used in the validation of the BM architecture with the only exemption of the metric "Activity Diversity" which was replace by "Active Time". This was motivated by the fact that we could not find any reference in the literature indicating that "Activity Diversity" was a suitable descriptor of behavioural intensity.

Table 4.8: **Description of Selected Telemetries**

Metric	**Description**
Absence	Temporal distance between sessions (hours)
Session Time	Overall session duration (minutes)
Active Time	Percentage of Session Time actively playing
Session Activity	Count of user initiated gameplay-related actions. E.g. ”Attack an enemy” is considered a valid action while ”Being attacked by an enemy” is not.
N°Sessions	Number of played sessions.
Object	Game object identifier.

We want to highlight that the high dispersion values (Inter Quartile Range or IQR), reported for some

of the telemetry are due to the extreme skewness in the distribution of the data. This is caused both by the nature of the phenomenon they describe (e.g. Absence is a classic case of time-to-event measure) and by their typical behaviour in the context of videogames [103]. The final dataset was composed of 6 columns and 28,155,199 rows. A table of descriptive statistics can be found in 4.9.

Table 4.9: **Descriptive Statistics of Considered Metrics and Games**

Game	Sample Size	Number of Sessions	Absence (minutes)	Session Time (minutes)	Active Time (% Session Time)	Session Activity	Type
				(Median $\pm$ IQR)			
hmg	501,649	3 ± 3	$84 \pm 2{,}169$	22 ± 22	64 ± 42	25 ± 31	Mobile Strategy
hms	504,504	8 ± 9	24 ± 198	28 ± 8	42 ± 35	6 ± 8	Mobile Shooting Gallery
jc3	540,000	7 ± 8	64 ± 488	162 ± 23	60 ± 55	19 ± 23	Console Action Open World
jc4	571,501	5 ± 6	64 ± 406	133 ± 64	43 ± 46	46 ± 64	Console Action Open World
lis	533,364	4 ± 4	$143 \pm 3{,}004$	96 ± 50	48 ± 44	40 ± 50	Console Graphic Adventure
lisbf	517,782	4 ± 5	$71 \pm 1{,}162$	102 ± 32	79 ± 20	23 ± 32	Console Graphic Adventure

Data Preparation When querying the data from the game servers, we excluded from the random sampling procedure individuals having at least one of the considered behavioural metrics over the game population's 99$^{\text{th}}$ percentile. This allowed us to eliminate potentially faulty data which are often present when dealing with telemetry.

At this point data were re-arranged in a format suitable for time series modelling and randomly split into a tuning (i.e. 10 %) and validation set (i.e., 90 %). As we mentioned in section 4.3, when fitting each model we adopted a data generator approach, this was done by batching both datasets in a series of multidimensional arrays of shape $(N \times T \times 5)$ (with T being the number of available game sessions within a batch and N the number of individuals inside said batch) and saving them on a local machine. When fitting or performing predictions a generator would then pass the batches in random order to the model allowing it to parse time series with different lengths between batches. For the sake of clarity we report an example of how the data from a single game session are generated and how they are

parsed by the models.

"A user decides, 36 hours after the release of game X, to enter the game world for the first time. This is when a session starts and actual playing behaviour can be observed. During this session they engage in various activities leading to 20 non-unique and user-initiated actions (e.g. being attacked by a non-playable character is not counted as a valid action). After roughly 60 minutes spent playing, the user exits the game world and the session ends. Of this session, 80% of time has been spent actively playing, the remaining 20% has seen the game on pause or the user away from the console (i.e., idle time). After 48 hours the user logs into the game world again and a new session starts"

What we described here would correspond to a single time step t_1 in a sequence of T total interactions (i.e., sessions) between a user and the specific game context X. The models will parse this session as a vector of length 4 with values 36, 20, 60, 80 and 20 along with another vector of length 1 containing the numerical index for the game. When all the sessions are observed the models will receive as inputs sequences of length T of the same vectors.

The behavioural metrics were min-max scaled according to the formula

$$MinMax(x) = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{4.12}$$

where x is the input vector to be scaled, while the categorical input (i.e. game object) was encoded ordinally.

4.3.3 Model Tuning and Comparison

Learning from the shortcoming of our previous model testing and in line with the increased complexity of considered architectures, we decided to improve our approach to hyper-parameter searching and model comparison.

The reason behind a more exhaustive, effective and efficient selection of hyper-parameters was motivated not just by performance concerns (i.e., the accuracy of ANNs can be substantially influenced by the choice of hyperparameters) but also by methodological ones. Indeed, architectural choices in ANNs can be characterized by an elevated number of degrees of freedom some of which would need to be factored out in order to perform a fair comparison between models. For example, when comparing our RNN architecture against linear or MLP one, we wanted to be able to attribute differences in performance to the introduction of non-linear and sequential operations rather than to the number of layers, hidden units or choice of activation functions. Indeed, these factors can influence the number of free parameters and expressive power of an ANN. Manually picking their optimal value is often a challenging combinatorial problem that can lead to unexpected outcomes if left to the subjective choice of the experimenter.

In this view, the first step in our model comparison phase aimed to control for the contribution of hyper-parameters in the performance of the parametric models (especially for MLP and RNN)

using an algorithmic approach. This was done using the Keras Tuner implementation [191] of the Hyperband algorithm [192]. Hyperband is an optimized version of random search that achieves faster convergence through adaptive resources allocation and early termination of training. It can lead to better or equivalent results to other optimization algorithms but in a fraction of the time [192]. When initializing the tuning step we allowed each model to grow as much as the others (except for E-Net, which, due to the fact that it is a linear model, is naturally constrained to a fixed number of parameters) so that any observed difference in number of parameters was related to characteristics of the model architecture. The tuning step was conducted running one full iteration of Hyperband with a budget of 40 epochs [4] on the tuning set. To trigger early stopping for a specific configuration of hyperparameters, we monitored the decrease in loss over a 20% random sample of the tuning set (we call this the validation tuning set) and we terminated training when the loss reduced by less than $\delta = 1e{-}4$ for 10 consecutive epochs.

Once the best set of hyperparameters was found we proceeded to fit all the models specified in section 4.3.1.0.1 on the validation set using a 10-fold Cross Validation Strategy. This divided the validation set in 10 equally sized folds and iteratively used 9 of them for training and 1 for testing. In order to take into account the contributions of time, game and target, the performance of each model was given by computing the Symmetric Mean Percentage Error (SMAPE) [194] for each combination of the aforementioned dimensions (e.g. SMAPE of Session Time at $t1$ for the game object hmg). Each model was trained for a maximum of 200 epochs and interrupted using the same early stopping strategy mentioned above (i.e. absence of δ reduction in loss on a 20% hold-out for 10 consecutive epochs). The models were trained with stochastic gradient descent using the Adaptive Moment Estimation (Adam) [185] algorithm as optimizer. We decided to drop the use of a scheduling approach for specifying the optimizer learning rate as it empirically showed to not provide any substantial improvement. Instead we kept default values provided by the implementation provided by the Keras library [180]. The optimizer was minimizing the SMAPE between the targets and the predictions generated by the model. The choice of SMAPE was dictated by the fact that the targets were expressed on largely different scales (i.e. coming from different games and expressed on different units of measure see Table 4.8) and therefore required a loss function measuring relative distance from the target.

To evaluate model performance we used a combination of visual inspection and inferential statistics. First, we visualized the variations in SMAPE for each combination of of target and model by collapsing the scoring metric over different dimensions (e.g. time and game context). Second, to evaluate the overall performance, we first summed the SMAPE relative to each target in a single global performance indicator: this is the loss function that each model attempts to minimize during training. We then divided the total by 5 (i.e. the total number of targets) in order to express the metric in its original scale (i.e. 0 to 100). This was then regressed using a Linear Mixed-effects Model (LMM) with game object and time as random effects and model type as fixed effect (treatment coded with the RNN architecture as reference). Subsequently, for a more thorough investigation of model performance

[4] See [192, 193] for a detailed description of the Hyperband technique.

we conducted the same regression analysis separately on each target. Both regression analyses were followed by post-hoc comparisons (i.e. t-tests with Bonferroni correction) for testing the following pairwise hypotheses on the estimated coefficients: Lag 1 $<$ Median $<$ ENet $<$ MLP.

A similar type of analysis was conducted within a Bayesian framework in order to more reliably assess the performance of our models. No substantial discrepancies with the frequentist analyses could be found, hence details and additional results are provided in Appendix C.2.

All statistical analyses were conducted using the python library statsmodels [195]. All the models, except for Median, were implemented using Tensorflow's high level API "Keras" [180, 188]. The Median model was implemented using the libraries for scientific computing Pandas and Numpy [196, 197].

4.3.3.1 Computational Resources and Run Times

The experiment was conducted on a single machine with 16 GB of RAM, a CPU with four 2.8 GHz cores and a dedicated GPU with 8 GB of VRAM. Training was carried out with GPU acceleration. The run time for the experiment (hyper-parameters tuning and model comparison) was of roughly 48 hours.

4.3.4 Results

Inspecting the performance of the RNN architecture over the time dimension, Figure 4.8, we can see how the model not only outperformed competing approaches for most of the considered targets (only for the metric Future Absence the performance is comparable to the MLP architecture) but it did so over all the considered time horizons.

Additionally we can see how in general the performance improved the more historical information were available to the model. A similar effect can also be observed for the MLP architecture but to a much lesser extent.

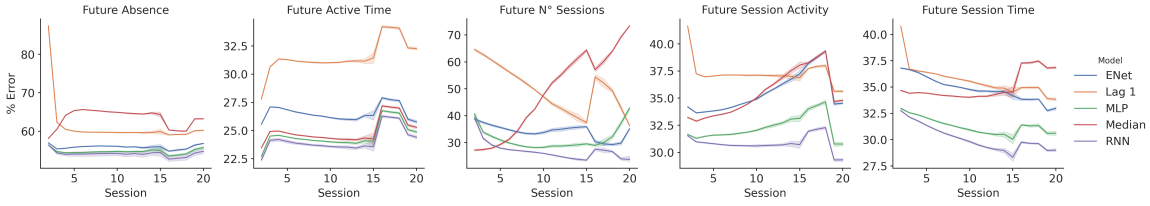


Figure 4.8: Overall, our approach (RNN) outperforms all the competing approaches at every time horizon. Each column represent the performance of the considered models on a specific target. Solid lines indicate the expected % error over time for a specific combination of target and model. Dashed areas indicate the standard error of the mean.

Similar conclusions can be drawn from inspecting the performance over the game-context dimension, Figure 4.9, where the RNN achieved the lowest error rate in almost every game context-target combination. In the few occasions in which this was not the case the performance was at least

comparable with that of the MLP architecture (although always better when looking at the expected performance).

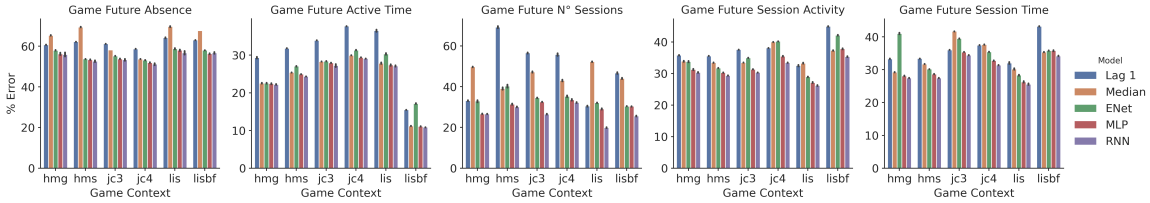


Figure 4.9: Overall, our approach (RNN) outperforms all the competing approaches in most of the target-game context combinations. Each column represent the performance of the considered models on a specific target. Bars indicate the expected % error for a specific combination of game context, target and model. Black vertical lines indicate the standard error of the mean.

These results remain almost unchanged even wen considering the interaction between game context and time (see Figure 4.10).

At the level of global performance the RNN model markedly outperformed all competing approaches as clearly shown in Figure 4.11. This can be also seen in the results of the regression (see Table 4.10) and *post hoc* analysis (see Table 4.11). From the *post hoc* analysis we can also observe how all the pairwise hypotheses presented in paragraph 4.3.3 are confirmed. Here model performance is given by the sum of all the losses produced by the five targets and therefore provides a general indicator of model fit where lower values indicate a better performance overall.

Table 4.10: **Results of LMM on Collapsed Targets (Sum)**

Model	β	**Z**	**p**	**95% C.I.**
Collapsed Targets (Sum)				
Intercept (RNN)	33.129	41.799	$<.01$	31.575 - 34.682
Lag 1	10.010	80.023	$<.01$	9.765 - 10.255
Median	7.482	59.813	$<.01$	7.237 - 7.727
ENet	4.189	33.491	$<.01$	3.944 - 4.435
MLP	1.411	11.280	$<.01$	1.166 - 1.656

Table 4.11: **LMM Post-Hoc on Collapsed Targets (Sum)**

Contrast	β_1 - β_2	**Z**	**p**	**95% C.I.**
Collapsed Targets (Sum)				
Lag 1 - Median	2.528	20.210	$<.01$	2.283 - 2.773
Median - ENet	3.292	26.322	$<.01$	3.048 - 3.538
ENet - MLP	2.778	22.211	$<.01$	2.533 - 3.024

The superiority of the RNN model can still be observed when comparing the models on each target separately. However, the size of the effect varies depending on the target (see Table 4.12). The same trend is also present in the *post hoc* analysis (see Table 4.13) where we observe only a partial confirmation of the pairwise hypotheses. The ENet model is outperformed by the Median baseline for three targets, namely Future Active Time, Session Time and Session Activity. All the coefficients in

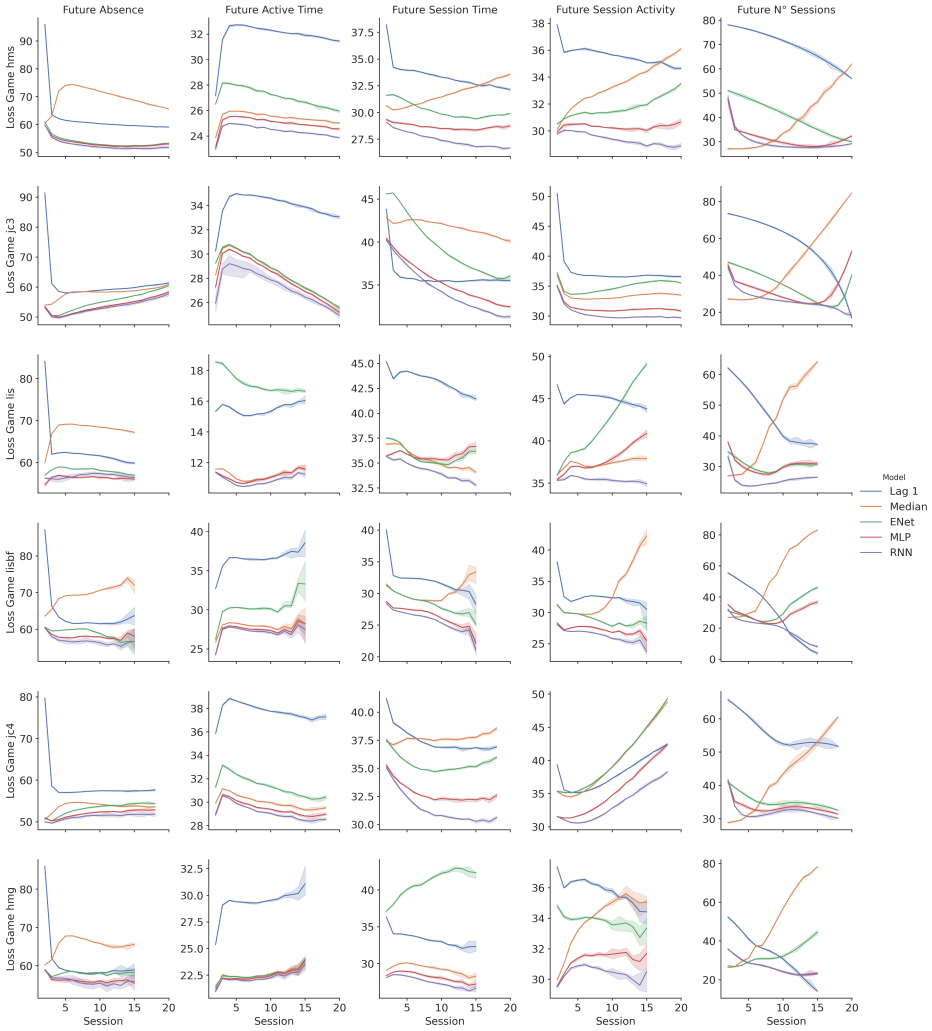


Figure 4.10: Overall, our approach (RNN) outperforms all the competing approaches in most of the target-game context combinations and temporal horizons. Each column represents the performance of the considered models on a specific target while each row reports the performance on a specific game context. Solid lines indicate the expected % error over time for a specific combination of target and model. Dashed areas indicate the standard error of the mean.

the regression analyses and the differences in the *post hoc* analyses are non-standardized and can be interpreted as absolute changes in percentage error (i.e. SMAPE).

In order to make these values more easily interpretable, we can use the information Table 4.9. For example, knowing that the median Session Time for the jc3 object is 162 minutes we can derive that when the RNN model achieves a SMAPE of 30% in predicting Future Session Time, this equates on average to an absolute error of $1.62 \times 30 \sim 48$ minutes. All the p-values in the *post hoc* analyses are Bonferroni corrected for multiple comparisons.

The results of the statistical analyses suggest positive additive effects of non-linearity and recurrency on model performance both at the level of global and target-specific performance. This effect is more pronounced for certain targets (e.g. Future Session Time, Future N° Sessions) than for others (e.g. Future Absence, Future Active Time). Moreover, looking at Figure 4.12 it appears that RNN improved

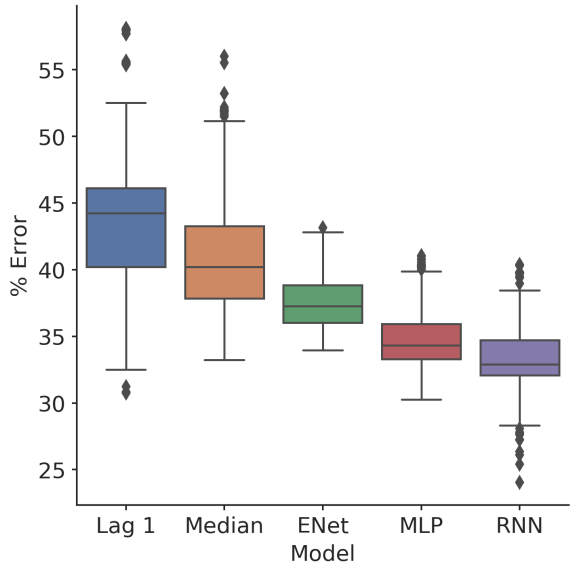


Figure 4.11: Overall, our approach (RNN) outperforms all the competing approaches. Box-plots show the 10-fold cross-validation performance expressed as the total percentage of error (i.e. SMAPE) of each model over the five targets.

on MLP (i.e., the second best model) using roughly half the parameters and per-epoch computation time.

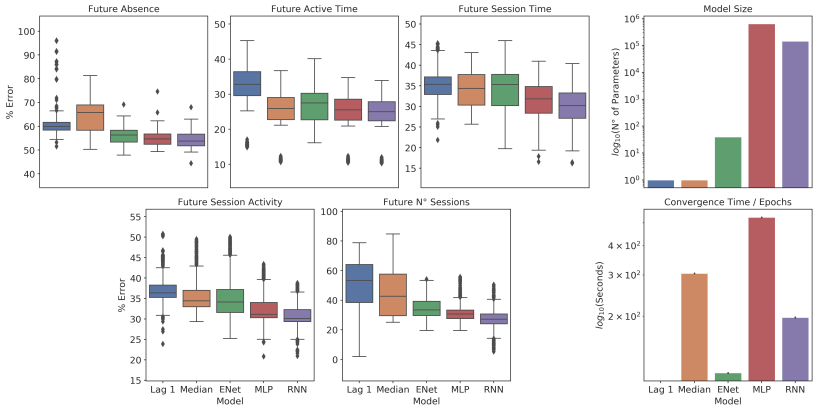


Figure 4.12: Our approach (RNN) outperformed all competing ones on each target. It consistently used fewer parameters and had shorter computation time than the second best performing model. Box-plots show the 10-fold cross-validation performance expressed as percentage of error (i.e. SMAPE) of each model for the five targets. The bar-plot on the top row indicates the number of free parameters for each model while the bar plot on the bottom row shows the average time for each training epoch. Both bar-plots are log_{10} scaled.

4.3.5 Model Criticism

Similarly to what presented in section 4.2.4 we were able to observe different level of performance for the baseline models in different game contexts, suggesting once again heterogeneity in the level of challenge for the predictive task. A similar level of heterogeneity was also found among the considered behavioural targets (e.g. Future Absence appeared to be a much harder target to predict than Future N° of Sessions) and temporal horizons (e.g. predictions made at an early or late stage showed to be more or less challenging depending on the considered target, game context and model architecture).

Table 4.12: **Results of LMM on Non-Collapsed Targets**

Model	β	Z	p	95% C.I.
Future Absence				
Intercept (RNN)	54.46	144.316	<.01	53.72 - 55.20
Lag 1	7.40	40.509	<.01	7.04 - 7.76
Median	9.47	51.814	<.01	9.11 - 9.83
ENet	1.71	9.353	<.01	1.35 - 2.06
MLP	.53	2.915	<.01	.175 - .891
Future Active Time				
Intercept (RNN)	23.36	20.019	<.01	21.78 - 24.93
Lag 1	7.32	119.028	<.01	7.2 - 7.44
Median	.77	12.515	<.01	.649 - .891
ENet	2.55	41.551	<.01	2.43 - 2.67
MLP	.41	6.739	<.01	.294 - .535
Future Session Time				
Intercept (RNN)	30.02	63.663	<.01	29.1 - 30.95
Lag 1	5.62	49.158	<.01	5.39 - 5.83
Median	4.45	38.957	<.01	4.23 - 4.67
ENet	4.81	42.098	<.01	4.59 - 5.03
MLP	1.08	9.529	<.01	.86 - 1.31
Future Session Activity				
Intercept (RNN)	31.04	70.241	<.01	30.17 - 31.9
Lag 1	6.52	61.025	<.01	6.31 - 6.73
Median	4.37	40.879	<.01	4.16 - 4.58
ENet	4.42	41.319	<.01	4.21 - 4.63
MLP	1.36	12.762	<.01	1.15 - 1.57
Future N° Sessions				
Intercept (RNN)	26.77	7.445	<.01	19.72 - 33.81
Lag 1	23.17	44.430	<.01	22.15 - 24.19
Median	18.34	35.166	<.01	17.31 - 19.36
ENet	7.44	14.277	<.01	6.42 - 8.46
MLP	3.65	7.005	<.01	2.63 - 4.67

By looking at the performance of the various parametric models we were able to replicate our initial findings, suggesting the importance of modelling non-linear dynamics when predicting measures of future behavioural intensity. When a model provided better performance than a competing one it did so among all the considered game contexts and behavioural targets, again confirming that the adopted global multi-task learning approach did not cause any degradation in predictive accuracy.

In summary, improving the BM model and modifying its architecture in order to better incorporate the theoretical constrains outlined in chapter 3 did not substantially change any of the findings reported in the first iteration of the model building process.

That said, despite the RNN architecture was now able to generate representations compatible with the computational frameworks presented by McClure et al. [13] and Zhang et al. [14] it was still

Table 4.13: **LMM Post-Hoc on Non-Collapsed Targets**

Contrast	β_1-β_2	Z	p	95% C.I.
Future Absence				
Lag 1 - Median	-2.06	-11.305	<.01	-2.42 - -1.70
Median - ENet	7.76	42.461	<.01	7.40 - 8.12
ENet - MLP	1.17	6.438	<.01	.81 - 1.535
Future Active Time				
Lag 1 - Median	6.55	106.513	<.01	6.433 - 6.67
Median - ENet	-1.78	-29.037	<.01	-1.9 - -1.66
ENet - MLP	2.14	34.812	<.01	2.02 - 2.26
Future Session Time				
Lag 1 - Median	1.16	10.201	<.01	.94 - 1.39
Median - ENet	-.35	-3.141	<.01	-.58 - -.13
ENet - MLP	3.72	32.579	<.01	3.5 - 3.95
Future Session Activity				
Lag 1 - Median	2.15	20.146	<.01	1.94 - 2.36
Median - ENet	-.04	-.441	1.	-.257 - .163
ENet - MLP	3.05	28.558	<.01	2.84 - 3.26
Future N° Sessions				
Lag 1 - Median	4.83	9.264	<.01	3.8 - 5.85
Median - ENet	10.89	20.889	<.01	9.87 - 11.91
ENet - MLP	3.79	7.272	<.01	2.77 - 4.81

not taking into account factors that might be relevant for reliably estimating the level of attributed incentive salience. Indeed, as we mentioned in chapter 2, we know that this type of latent state, and its associated dynamics, are modulated by the internal condition of the individual [14], by the characteristics of the rewarding object (a specific videogame in this case) with which the individual is interacting and by the environment in which the interaction is occurring [157]. These factors, were only partially and indirectly captured by the RNN architecture as they require more granular information (i.e., in-game and environmental information) and with a higher temporal resolution (i.e., within rather than between game sessions). As a consequence we noticed how the RNN architecture, despite outperforming competing ones, still achieved a relatively high error rate in predicting some of the considered behavioural targets (e.g., future Absence).

A possible solution in this case would have been to incorporate information about the environment in which the observed behaviour occurred (e.g., time and location of a specific game session) and about the characteristics of the rewarding objects (i.e., the so called videogame structural characteristics that we mentioned in chapter 2). As well as improving the predictive performance of the model, these new types of information should also increase the quality of the generated representation and consequently allow for a better approximation of the level of attributed incentive salience. In line with the ideas presented in section 3.3.5, the next iteration of the model building process will aim at modifying the RNN architecture in order to incorporate the type of environmental and game-specific information that we just mentioned.

4.4 Dynamic Prediction of Future Behavioural Intensity with Environmental and Game Covariates

In the third and last iteration of the model building process the focus was mostly on modifying the RNN architecture in order to incorporate information from the environment surrounding the individual and the in-game events characterizing the playing sessions. However, given the computational flexibility of ANN, this needed to be done through a strategy that maintained compliance with the theoretical constrains mentioned in chapter 3.

Our approach was then to assume that the amount of incentive salience that an individual attributes to a potentially rewarding object (i.e., a videogame) during a specific interaction could be decomposed in 3 major components:

1. The current level of attributed incentive salience. This can be inferred by the history of interactions an their associated behavioural intensity. In our setting this was represented by the latent state inferred by the model after observing all the previous interactions.

2. The impact of the environmental conditions in which each interaction occurred. This information does not directly influence the amount of attributed incentive salience but help accounting for discrepancies in the observed behavioural manifestations. For example, an individual might ascribe high saliency to a specific object but show reduced goal directed behaviour due to impediments in the surrounding environment. In our setting these were represented by a series of indicators (e.g., hour of day, day of the week) specifying the environmental conditions in which a playing session was observed.

3. The impact of the, potentially rewarding, characteristics of the object. This factor is directly involved in determining the level of attributed saliency as it informs about which aspects of the game are likely to produce rewarding experiences. In our setting these were represented by metrics describing the amount and type of game events triggered by an individual during a playing session. It is important to note that given the nature of video games (i.e., secondary rewarding objects) the impact of these characteristics could vary from individual to individual and from interaction to interaction. In this view they are better understood in conjunction with the behavioural intensity of the game session in which they occur. For example, if an individual produces a prolonged playing session characterized by a high number of exploration events, it is likely that in this instance these specific characteristics contributed to the rewarding experience provided by the game.

By adopting a similar approach to Neural GAMs [179] each component was estimated independently and subsequently combined in a single, more informative, representation. This was done with the aim to achieve a higher level of predictive accuracy while maintaining control on the mechanics generating the latent representation (more details will be provided in section 4.4.1).

In order to investigate the contribution of the included covariates in improving the overall predictive

accuracy we proceeded at validating the following hypotheses:

1. Including information about the environmental context in which an interaction occurred can have a positive impact on the predictive accuracy of the model.
2. Including information about the amount and type of game events occurring during an interaction can increase predictive accuracy more than when considering environmental factors alone.
3. Considering the joint contribution of environmental factors and game characteristics can have a greater impact on predictive accuracy than when considering the two factors alone.

We want to stress how in this context we considered the ability to account for temporal dynamics as a factor of pivotal importance. The model had now to track changes not just in the behavioural intensity of the observed interactions, but also in the surrounding environment and in the preference for different game characteristics. For this reason we also tried to replicate some of the findings observed in the previous stages of the model building process by evaluating the following additional hypotheses

1. Including sequential dynamics in the modelling approach can have a positive effect on the predictive performance.
2. It is possible to have a single model jointly multiple first-order metrics of behavioural intensity without any substantial loss in accuracy
3. It is possible to have a single model performing the predictive task simultaneously across a wide range of games.

4.4.1 Model Design

The final iteration of our ANN architecture was an extension of the previous RNN model and aimed to jointly predict the same set of 5 behavioural metrics. This new architecture however, leveraging the same principles of Neural GAMs (as outlined in section 3.3.5), tried to obtain a better estimate of the latent state used for predicting future behaviour by decomposing it in 3 functional components. The architecture is illustrated in Figure 4.13

The model received the same behavioural inputs of the architecture described in section 4.3.1 along with two new types of inputs. A set of 5 univariate time series of shape $E \in \mathbb{N}^{N\times T\times 1}$ was used for introducing information about the environment in which an interaction occurred (a full description of the metrics employed is provided in table 4.15). Two multivariate time series, one of shape $G_e \in \mathbb{N}^{N\times T\times 33}$ and one of shape $G_f \in \mathbb{R}^{N\times T\times 33}$, were used for introducing information about the type and frequency of game events triggered during a playing session (a complete list of events and their description can be found in table 4.16). It has to be noted that all the series in E and G_e contained metrics that were numerically encoded (e.g. $Monday \mapsto 1$, $Tuesday \mapsto 2$, $Shooting \mapsto 1$ etc.) in order to leverage the same embedding operation used for representing the game context.

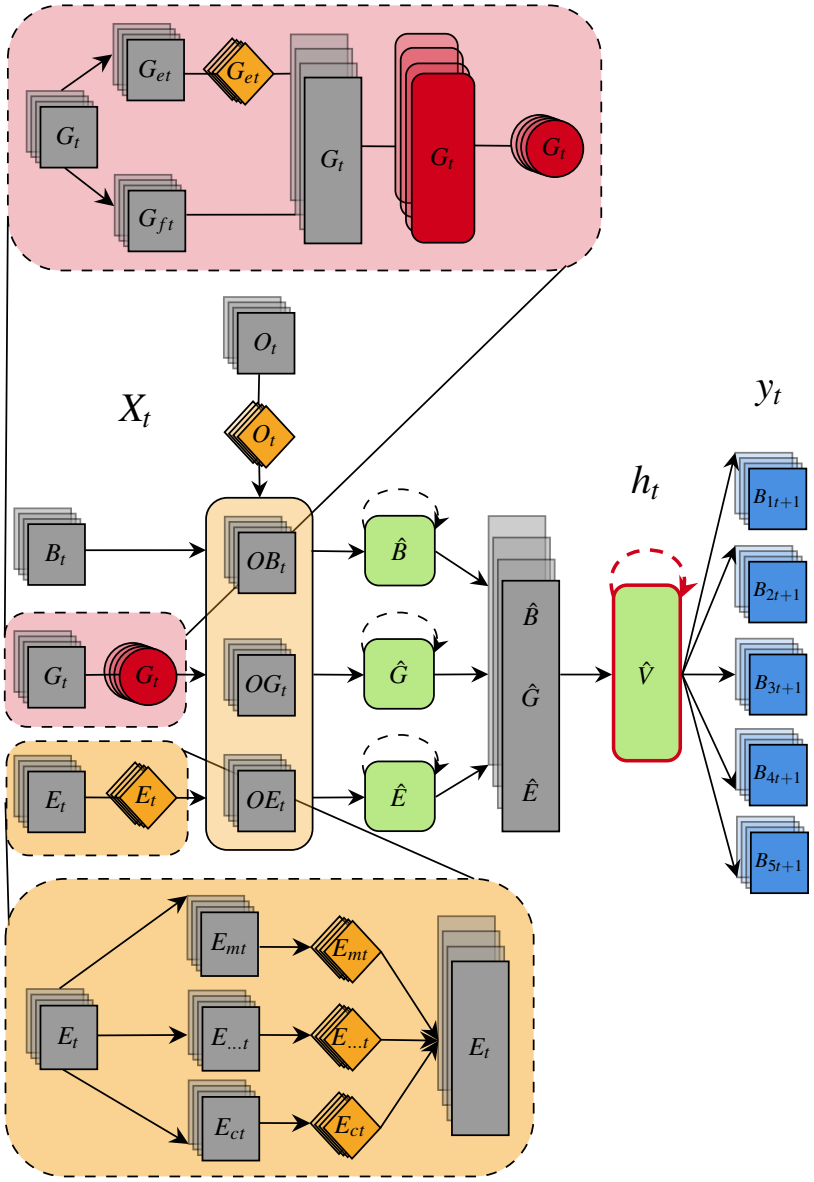


Figure 4.13: Blue, orange and green shapes represent respectively feedforward, embedding and LSTM layers. Embedding layers are a type of feedforward layers specifically designed for dealing with categorical inputs [180]. Red rectangles and circles represent respectively 1-dimensional convolution and 1-dimensional global average pooling operations. Gray shapes indicate operations with no learnable parameters, such as tensor instantiation and concatenation. The orange transparent shape indicate the concatenation of a single embedding with multiple tensors. Stacked, transparent colouring indicates arrays with a sequential structure. Straight and curved arrows refer to the presence of feed-forward or recurrent information flow. The red highlight shows the portion of the model we hypothesize is inferring an approximation of attributed incentive salience.

For incorporating information related to game events, which required to jointly embed sets of categorical (i.e., the numerical identifiers of specific game events) and continuous indicators (i.e., the frequency at which each event occurred in a session), we modified the approach used by Zhao et al. in [198]. First the series G_e were transformed using an embedding operation, resulting in a $G_e \in \mathbb{R}^{N\times T\times 33\times h}$ tensor (with h being the number of hidden units chosen for the embedding operation operation). This tensor was then concatenated with the series in G_f and 1 dimensional convolution operation (see A.24) with global average pooling (see A.25) was applied, resulting in a $G \in \mathbb{R}^{N\times T\times h}$ (with h being

the number of hidden units chosen for the convolution operation). We want to highlight that in this case, the use of a common embedding operation allowed the architecture to share information across different game objects and events.

Subsequently all the three inputs (i.e., behavior, environment and events) were concatenated with the embedding generated for the game object context (using the same approach of the RNN architecture) producing 3 $O^* \in \mathbb{R}^{N\times T\times h+h_O}$, with h being the original dimensionality of the input and h_O the number of hidden units chosen for the embedding operation. The resulting tensors where then passed through 3 separate LSTM operations and subsequently concatenated in a single one of shape $\widehat{B}\widehat{G}\widehat{E} \in \mathbb{R}^{N\times T\times h_b+h_g+h_e}$. Following the diagram in Figure 4.13 we can see that differently from the RNN architecture the expected intensity of future interactions is now a function of three separate factors (i.e., the history of behavioural intensity, environmental changes and game events) and, thanks to the use of recurrent operations, their variation over the past. This implies that representations generated at this stage of the model architecture embed information on how intense future interactions with a specific game object will be after observing sequences of of any of the 3 aforementioned factors

$$\begin{aligned}\mathbb{E}[B_{t+1}] &\approx f(\widehat{B}_t;\theta)\\ \mathbb{E}[B_{t+1}] &\approx f(\widehat{G_t};\theta)\\ \mathbb{E}[B_{t+1}] &\approx f(\widehat{E}_t;\theta)\end{aligned} \tag{4.13}$$

For example, we might infer that for a specific game object, interactions characterized by a large number of fighting events are associated, on average, with high future behavioural intensity. This is in line with our contextualisation of attributed incentive salience within a video game setting (see chapter 2): after accounting for the contribution of the environment, the saliency that an individual attributes to a videogame object can change as a function of how rewarding the game characteristics are perceived to be.

That said, considering environmental factors and game events alone would provide information that hold on average (e.g., the sequence of events A is more likely to produce playing behaviour than the sequence B) but fail to capture inter-individuals and inter-interactions differences (e.g. changes in the level of reward that an individual experience when interacting with a specific game characteristic). In order to account for this limitation, the architecture attempted to model interactions between the three considered factors in a dynamic fashion. This was done in the same way as in the RNN architecture: the representations associated to each of the three factors were jointly parsed by a single LSTM operation and the generated embedding was used for predicting all the behavioural targets by means of time-distributed fully-connected operations.

As a regularization strategy we relied on the same one dimensional spatial dropout used in the RNN architecture.

Competing Models The type of competing models used for validating the assumption of our new model architecture changed markedly from the previous stage of the model building process.

Since the superiority of non-linearity was at this point widely confirmed, we decided to not include in the comparison the Elastic Net model. The same thing was done for the lag-1 and median models as they showed to markedly under-perform any parametric approach.

In order to evaluate the contribution of modelling environmental factors and game events in the predictive performance of the architecture, we adopted a variation of ablation study [199]. Ablation studies aim at assessing the contribution of the components of a machine learning system by evaluating variations in the system's performance after selectively removing (i.e., ablating) them. In our case we opted for a more expensive although more robust approach, we generated three versions of the architecture presented in section 4.4 and fitted all of them separately to the data. One architecture included only the behavioural input, one included only the game events input and one included only the environmental inputs. In the context of very expressive algorithms, like ANNs, allowing each architecture to benefit from a separate hyper-parameter tuning and optimization step could allow the emergence of compensation strategies not contemplated in conventional ablation studies.

Finally in order to replicate our finding on the importance of recurrency, we decided to consider a variation of the architecture presented in section 4.4 substituting the LSTM operations with time-distributed feed-forward ones (see Figure 4.14).

4.4.2 Data

In order to validate the architecture presented in section 4.4, we leveraged a dataset with similar characteristics to the one used in section 4.3.2. Despite still relying on gameplay data provided by our partner company, *Square Enix Ltd.*, we had to drop one of the title (Hitman Sniper, hms) used in the previous stages of the model building process, as it was discontinued. As a replacement we decided to include data from a new open world action role-play game with cooperative multiplayer characteristics (Outriders, outr).

The dataset contained entries from 2,805,952 individuals, evenly distributed across the six games, and randomly sampled from all users who played the games between their respective release date and January 2021. The reason for considering a reduced number of individuals in this stage of the model building process was mostly technical. The model outlined in section 4.4, along with its variation presented in section 4.4.1, required a consistently larger amount of information per individual, making the model tuning and comparison steps rather demanding given the the available hardware. All data were obtained and processed in compliance with the European Union's General Data Protection Regulation [1].

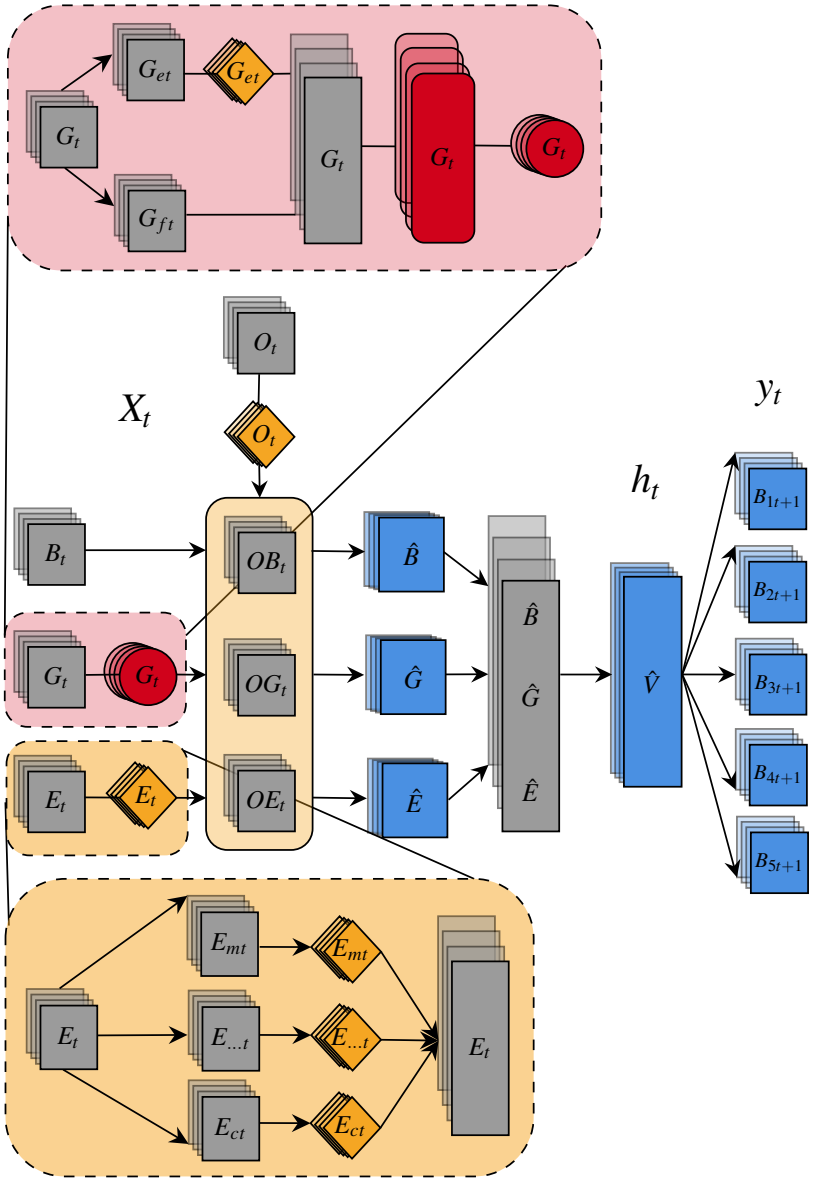


Figure 4.14: Blue and orange shapes represent respectively feedforward and embedding operations. Red rectangles and circles represent respectively 1-dimensional convolution and 1-dimensional global average pooling operations. Gray shapes indicate operations with no learnable parameters, such as tensor instantiation and concatenation. Stacked, transparent colouring indicates tensors with a sequential structure. Straight arrows refer to the presence of feed-forward information flow.
All the feedforward operations are time distributed.

Defining the Behavioural Metrics and Targets The metric used for representing the behavioural manifestations of state transition dynamics, were the same of those used for validating the RNN architecture with only a small variation, we modified the way the metric Session Activity was computed. We moved from the count of user initiated gameplay actions to the ratio of actions per minute. The reason was that from a correlation a analysis we realized that this metric was highly correlated with the metric Session Time.

Table 4.14: **Description of Selected Telemetries**

Metric	Description
Absence	Temporal distance between sessions (hours)
Session Time	Overall session duration (minutes)
Active Time	Percentage of Session Time actively playing
Session Activity	Ratio of user initiated gameplay-related actions per minute in Session Time. E.g. ”Attack an enemy” is considered a valid action while ”Being attacked by an enemy” is not.
N°Sessions	Number of played sessions.
Object	Game object identifier.

Similarly to what was done for the RNN architecture, the target consisted of the lead-1 version of the behavioural metrics described in table 4.14.

Defining the Environmental and Game Events Covariates In order to represent the environmental covariates we simply decomposed the trigger identifier associated with the start of a game session (we point to section 4.3.2 for a more detailed definition of what defines a game session). These identifier usually holds information on where and when an individual initiated the game activity and are used for mapping multiple game events triggered during the playing activity to a single session. It is worth noticing that all the trigger identifiers were logged in Central European Time (CEST), we decided to not perform any time zone conversion and simply include indicators of the geographical area from where the trigger was recorded. A description of the information extracted can be found in table 4.15

Table 4.15: **Description of Selected Environmental Telemetries**

Metric	Description
Hour of the Day	Hour of the day in which the session started.
Day of the Week	Day of the week in which the session started.
Day of the Month	Day of the month in which the session started.
Month of the Year	Month of the year in which the session started.
Country Area	Broad indicator of the country area in which the session started.

For incorporating information on the game events we first individuated which were most representative of the core gameplay of each title (e.g., interacting with the menu was not considered a relevant event). Subsequently for each session we obtained a list of the unique events triggered within that session and their associated frequencies. It is important to note that this list could be empty in case an individual played a session without interacting with any of the selected core gameplay events. A description of

the considered game events can be found in table 4.16

Table 4.16: **Description of Selected Game Events Telemetries**

Event Name	Game Context	Event Description
Operation	jc4	Completing the operation events
Chaos Created	jc4	Completing chaos mini events
Customization	lisbf	Customize the outfit of the protagonistt
Skill	outr	Increasing the character skill
Back Talk	lisbf	Completing the backtalk mini events
Episode Finished	lis	Finishing a major section of the game
Mission Attempt	jc3	Starting mision events
Province Liberated	jc4	Completing province liberated mini events
Skip Cinematic	outr	Skipping non-playable cinematics
Hint Used	lis	Using hints for solving puzzles
Major Choice	lisbf	Selecting option in major choice events
Rewind Used	lis	Triggering the rewind power
Challenge	outr	Completing challenge mini events
Vendor	outr	Interacting with the in-game vendors
Achievemnt	hmg	Unlocking in-game achievements
Item Crafted	outr	Crafting in-game items
Graffiti	lisbf	Completing the graffiti mini-events
Node Started	jc4	Starting node mini events
Objective Start	jc4	Starting objective mini events
Attempt Start	jc4	Attempting to complete mission events
Class Skill	outr	Increasing the class skill
Collectible	jc3	Obtaining collectible in-game items
To Shop	hmg	Going to the in-game shop
Puzzle Start	lis	Srtaing puzzle mini events
Supply Drop	jc3	Obtaining supplies
Emote	outr	Triggering emote actions
Vehicle	jc3	Entering \Exiting \Driving a vehicle
Minor Choice	lisbf	Selecting option in minor choice events
Coop	outr	Entering in coop mode
Upgrade	hmg	Upgrading in-game items
Collectible	hmg	Obtaining in-game collectibles
Sub Context Start	hmg	Starting sub-context events
Achivement	outr	Unlocking in-game achievements

The final dataset was composed of 13 columns and 28,804,987 rows. A table of descriptive statistics

for the behavioural metrics can be found in 4.17.

Table 4.17: **Descriptive Statistics of Considered Metrics and Games**

Game	Sample Size	Number of Sessions	Absence (minutes)	Session Time (minutes)	Active Time (% Session Time)	Session Activity	Type
				(Median $\pm$ IQR)			
hmg	452,386	5 ± 6	141 ± 219	12 ± 15	87 ± 24	1.3 ± 0.9	Mobile Strategy
jc3	491,329	7 ± 7	27 ± 365	64 ± 143	99 ± 1	0.18 ± 0.22	Console Action Open World
jc4	485,276	6 ± 7	31 ± 356	59 ± 87	82 ± 25	0.28 ± 0.45	Console Action Open World
lis	483,783	5 ± 6	70 ± 844	64 ± 104	78 ± 25	0.5 ± 0.4	Console Graphic Adventure
lisbf	431,355	6 ± 6	173 ± 574	67 ± 87	81 ± 26	0.08 ± 0.15	Console Graphic Adventure
outr	461,823	7 ± 7	47 ± 33	64 ± 121	88 ± 65	0.02 ± 0.04	Console Action Open World Co-op

Data Preparation When querying the data from the game servers, we relied on the usual permissive filtering strategy by excluding from the random sampling procedure individuals having at least one of the considered behavioural metrics over the game population's 99^{th} percentile. Data were re-arranged in the conventional format suitable for time series modelling and randomly split into a tuning (i.e., 10 %) and validation set (i.e., 90 %).

Model fitting was again done using the same batching strategy presented in section 4.3 with the only difference that the data generator had now to provide the model with multiple input: a single multidimensional array of shape $(N \times T \times 5)$ for the behavioural metrics, 5 separate of arrays of shape $(N \times T \times 1)$ for the environmental covariates and 2 separate arrays of shape $(N \times T \times 7)$ for the game events covariates. Since the number of game events could vary from session to session, we first individuated the maximum number of unique events triggered in a session for the the entire dataset (i.e. 7), and subsequently zero padded the multidimensional arrays accordingly.

The behavioural metrics were min-max scaled according to the same formula defined in equation

4.12 while the categorical inputs (i.e. game object, environmental and game events covariates) were encoded ordinally.

4.4.3 Model Tuning and Comparison

Model tuning and comparison followed almost the exact same strategy employed for validating the RNN architecture with only few minor differences. Due to time constrains and hardware limitations we reduced the budget assigned to Hyperband from 40 to 20 epochs and limited the number of cross-validation folds to 8.

The cross validation results, with collapsed and non-collapsed targets, were again analysed using a Linear Mixed-effects Model (LMM) with game object and time as random effects and model type as fixed effect (treatment coded with the RNN architecture as reference). In order to investigate the hypotheses outlined in 4.4 we carried out post-hoc comparisons (i.e. t-tests with Bonferroni correction) on the coefficient estimated by the LMM model expecting the following inequalities: RNN Env $<$ RNN Even $<$ MLP $<$ RNN Env Even. Also in this case the ancillary analyses conducted within a Bayesian framework did not diverge substantially from the frequentist analyses, details and additional results are provided in Appendix C.2.

All statistical analyses were conducted using the python library statsmodels [195]. All the models were implemented using Tensorflow's high level API "Keras" [180, 188].

4.4.3.1 Computational Resources and Run Times

The experiment was conducted on a single machine with 16 GB of RAM, a CPU with four 2.8 GHz cores and a dedicated GPU with 8 GB of VRAM. Training was carried out with GPU acceleration. The run time for the experiment (hyper-parameters tuning and model comparison) was of roughly 168 hours.

4.4.4 Results

Inspecting the results of the regression analysis on the collapsed targets (see Table 4.16 and Figure 4.15) we can see how introducing environmental and game events covariates led to a modest but robust improvement in performance when compared with the RNN architecture.

An interesting although surprising result was that the MLP architecture performed worse than the simple RNN architecture, despite having access to substantially more information (see Figure 4.19). Since this result was not expected a-priori and the estimated effect size is quite modest, we invite to interpret it with caution. Especially given the relatively small number of parameters assigned to the MLP architecture by the tuning procedure (see Figure 4.19).

From the post-hoc analysis 4.19, we can see how the hypotheses presented in section 4.4.3 were confirmed for all architectures except MLP, which resulted to be the worst performing one (in some cases even by a substantial margin).

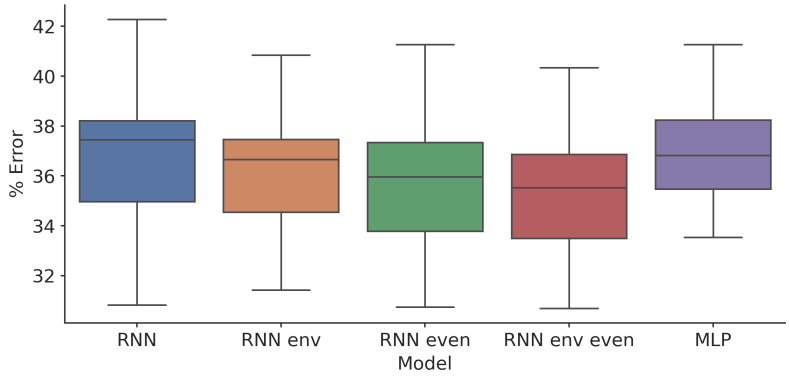


Figure 4.15: Overall, the joint contribution of game events and environmental covariates appear to have a positive effect on performance. Box-plots show the 8-fold cross-validation performance expressed as the total percentage of error (i.e. SMAPE) of each model over the five targets.

Table 4.18: **Results of LMM on Collapsed Targets (Sum)**

Model	β	Z	p	95% C.I.
Collapsed Targets (Sum)				
Intercept (RNN)	36.518	41.374	<.01	34.788 - 38.248
MLP	.366	7.569	<.01	.271 - .460
RNN env	-.494	-1.224	<.01	-.589 - -.399
RNN even	-.938	-27.424	<.01	-1.419 - -1.230
RNN env even	-1.325	-19.414	<.01	-1.032 - -.843

Figures 4.16 and 4.17 appear to show how including environmental and game events covariates seem to improve model performance in an additive fashion, with results that are consistent across temporal horizons

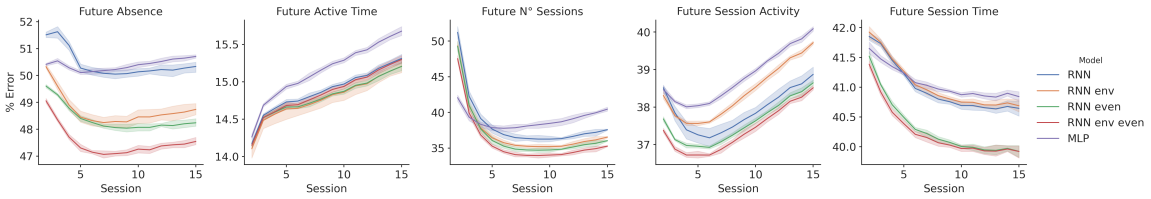


Figure 4.16: Overall, including environmental and game events covariates appear to improve model performance in an additive fashion at every time horizon. The size of the improvement however seems to be heterogeneous across targets. Each column represent the performance of the considered models on a specific target. Solid lines indicate the expected % error over time for a specific combination of target and model. Dashed areas indicate the standard error of the mean.

and game contexts. From the two figures we can also observe a number of relevant phenomena.

First, the largest performance improvement was observed for the metric Future Absence. Second, as in the previous two model comparison iterations we can observe an heterogeneity in absolute performance driven by the game context and target taken into consideration. Third, the contribution of the environmental covariates on performance appeared to be rather erratic (occasionally even hurting performance) but always led to a consistent positive impact when coupled with game events

Table 4.19: **LMM Post-Hoc on Collapsed Targets (Sum)**

Contrast	β_1 - β_2	Z	p	95% C.I.
Collapsed Targets (Sum)				
MLP - RNN env	.859	17.793	<.01	.765 - .954
MLP - RNN env even	1.690	34.993	<.01	1.596 - 1.785
MLP - RNN even	1.303	26.983	<.01	1.209 - 1.398
RNN env - RNN env even	.830	17.200	<.01	.736 - .925
RNN env - RNN even	.443	9.190	<.01	.349 - .539
RNN env even - RNN even	-.386	-8.010	<.01	-.482 - -.292

information. Finally, despite the proposed modification to the RNN architecture had a consistent positive impact across the board, we can see from Figure 4.18 that the effect of environmental and game events covariates alone appeared to vary across game contexts and targets.

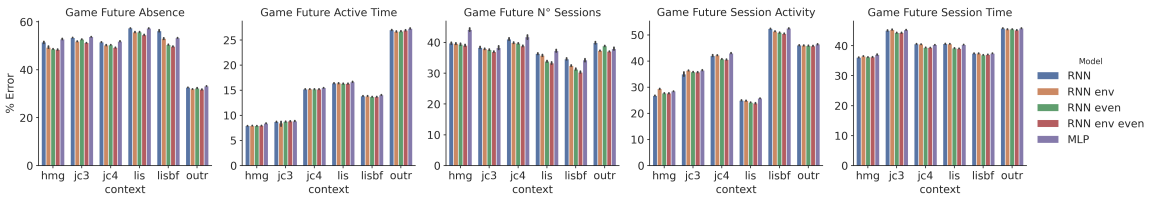


Figure 4.17: Overall, including environmental and game events covariates appear to improve model performance in an additive fashion at every time horizon. The size of the improvement however seems to be heterogeneous across targets. Each column represent the performance of the considered models on a specific target. Bars indicate the expected % error for a specific combination of game context, target and model. Black vertical lines indicate the standard error of the mean.

This was confirmed by the analyses carried out at the target level, where a more heterogeneous performance landscape emerged. From table 4.20 we can see that despite the inclusion of environmental and game events covariate improved the performance of the RNN architecture for most targets, there were some noticeable exceptions.

For the target Future Active Time there seemed to be virtually no effect associated to the inclusion of the two covariates, this was likely due to marked performance differences between game contexts (see Figure 4.18). Consistently with previous findings, the MLP architecture appeared to under perform the simple RNN for all the considered targets.

The results from the post-hoc analysis (see Table 4.21) partially validated our pair-wise hypotheses. No meaningful difference between architectures was found for the target Future Active Time. For all the other targets we observed that:

1. The use of environmental covariates alone performed equally or worse than the use of game events alone.
2. Combining the two type of covariates markedly outperformed the use of game events alone only for two targets: Future Absence and Future Number of Sessions
3. The MLP architecture appeared to be the worst performing one.

Table 4.20: **Results of LMM on Non-Collapsed Targets**

Model	β	Z	p	95% C.I.
Future Absence				
Intercept (RNN)	50.44	42.24	<.01	48.10 - 52.78
MLP	-.050	-.845	.398	-.16 - .06
RNN env	-1.74	-29.30	<.01	-1.85 - -1.62
RNN even	-2.93	-49.34	<.01	-3.05 - -2.81
RNN env even	-2.05	-34.60	<.01	-2.17 - -1.94
Future Active Time				
Intercept (RNN)	14.877	14.99	<.01	12.93 - 16.82
MLP	0.26	6.65	<.01	.19 - .34
RNN env	-.09	-2.41	.016	-.17 - -.01
RNN even	-.08	-2.02	.043	-.16 - -.002
RNN env even	-.03	-0.776	.438	-.111 - .04
Future Session Time				
Intercept (RNN)	40.978	77.831	<.01	39.94 - 42.01
MLP	.08	2.19	.02	.009 - .16
RNN env	.05	1.28	.2	-.02 - .12
RNN even	-.67	-16.98	<.01	-.75 - .59
RNN env even	-.73	-18.48	<.01	-.81 - -.65
Future Session Activity				
Intercept (RNN)	37.91	27.24	<.01	35.18 - 40.64
MLP	.910	5.93	<.01	.61 - 1.21
RNN env	.49	3.25	<.01	.19 - .79
RNN even	-.32	-2.091	.03	.62 - -.02
RNN env even	-.51	-3.35	<.01	-.81 - -.21
Future N° Sessions				
Intercept (RNN)	38.37	73.91	<.01	37.35 - 39.39
MLP	.61	5.13	<.01	.37 - .84
RNN env	-1.17	-9.89	<.01	-1.41 - -.94
RNN even	-1.55	-13.05	<.01	-1.78 - -1.32
RNN env even	-2.41	-20.24	<.01	-2.64 - -2.17

From Figure 4.19 we can see that, differently from the RNN architecture, the improvements in performance produced by the new covariates mechanisms required a substantial increase in the number of parameters and computation time. The MLP architecture again produced surprising results, despite having the least number of parameters it showed one of the highest computation time per epoch suggesting potential inefficiency during the fitting procedure.

4.5 Model Criticism

Mirroring the results presented in sections 4.2.4 and 4.3.4 we were able to observe different level of performance in different game contexts and for different targets. This time however, the improvements provided by the specific variations of the RNN architecture were not homogeneous across contexts

Table 4.21: **LMM Post-Hoc on Non-Collapsed Targets**

Contrast	β_1-β_2	Z	p	95% C.I.
Future Absence				
MLP - RNN env	1.69	28.458	<.01	1.57 - 1.80
MLP - RNN env even	2.88	48.503	<.01	2.76 - 3.00
MLP - RNN even	2.00	33.759	<.01	1.89 - 2.12
RNN env - RNN env even	1.19	2.045	<.01	1.07 - 1.30
RNN env - RNN even	.31	5.301	<.01	.19 - .43
RNN env even - RNN even	-.87	-14.744	<.01	-.99 - -.76
Future Active Time				
MLP - RNN env	.36	-11.305	<.01	.28 - .44
MLP - RNN env even	.3	42.461	<.01	.22 - .38
MLP - RNN even	.35	42.461	<.01	.27 - .43
RNN env - RNN env even	-.06	42.461	.6	-.14 - .01
RNN env - RNN even	-.01	42.461	1	-.095 - .06
RNN env even - RNN even	.05	6.438	1	-.02 - .13
Future Session Time				
MLP - RNN env	.03	.918	1	-.04 - .11
MLP - RNN env even	.82	2.686	<.01	.74 - .89
MLP - RNN even	.76	19.179	<.01	.68 - .83
RNN env - RNN env even	.78	19.768	<.01	.70 - .86
RNN env - RNN even	.72	18.261	<.01	.64 - .80
RNN env even - RNN even	-.05	-1.507	.79	-.13 - .01
Future Session Activity				
MLP - RNN env	.41	2.688	.042	.11 - .71
MLP - RNN env even	1.42	9.289	<.01	1.12 - 1.72
MLP - RNN even	1.23	8.029	<.01	.93 - 1.53
RNN env - RNN env even	1.01	6.601	<.01	.71 - 1.31
RNN env - RNN even	.81	5.341	<.01	.51 - 1.11
RNN env even - RNN even	-.19	-1.260	1	-.49 - .10
Future N° Sessions				
MLP - RNN env	1.78	15.033	<.01	1.55 - 2.02
MLP - RNN env even	3.02	25.382	<.01	2.78 - 3.25
MLP - RNN even	2.16	18.197	<.01	1.93 - 2.39
RNN env - RNN env even	1.23	1.349	<.01	.99 - 1.46
RNN env - RNN even	.37	3.164	.012	.14 - .61
RNN env even - RNN even	.85	-7.185	<.01	-1.08 - -.62

and targets, suggesting a slight inefficiency of the global multi-task approach. Overall most of the hypotheses introduced in section 4.4 were confirmed by the model comparison results (at least at the aggregate level). From this we can derive a series of conclusions on the effectiveness of the different architectures in generating a latent representation with adequate predictive powers.

1. Considering changes in the surrounding environment might only marginally contribute to the predictive perfromance.

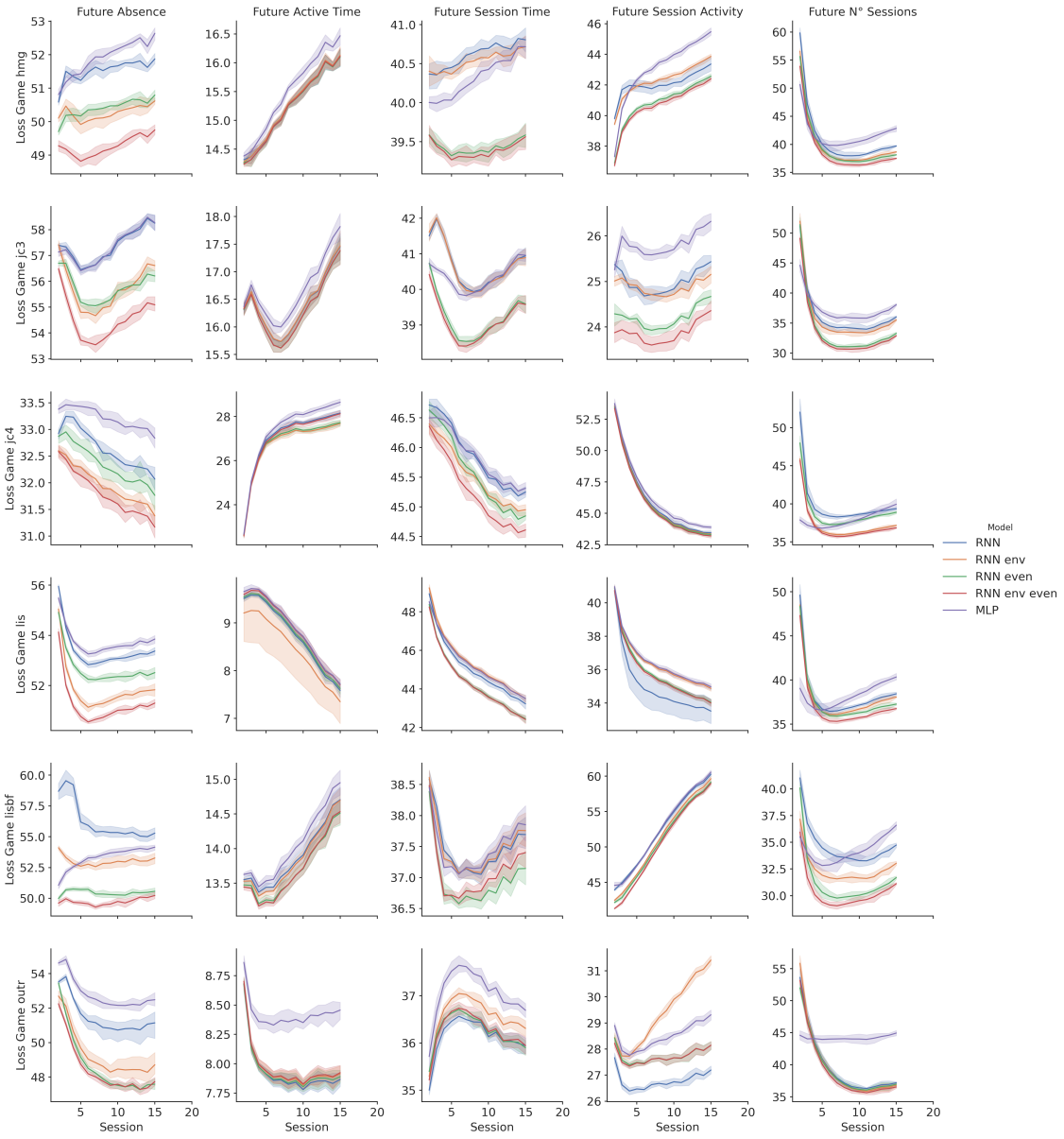


Figure 4.18: Overall, the joint contribution of game events and environmental covariates appear to have a positive effect of performance in most of the target-game context combinations and temporal horizons. Each column represents the performance of the considered models on a specific target while each row reports the performance on a specific game context. Solid lines indicate the expected % error over time for a specific combination of target and model. Dashed areas indicate the standard error of the mean.

2. Considering the characteristics of the object can have a greater impact on the predictive accuracy.

3. In line with the what presented in chapters 2 and 3, the combination of the two covariates provides the most effective strategy for producing accurate predictions.

4. Despite the poor performance of the MLP architecture should be interpreted with caution, it strengthens the idea that dynamic operations are important for computing the latent representation.

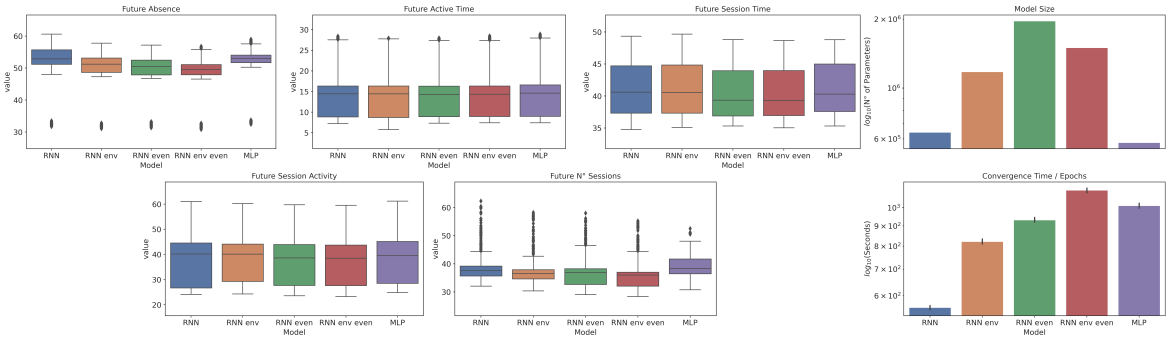


Figure 4.19: The combination of environmental and game events covariates appeared to improve performance in all the targets except Future Active Time. Box-plots show the 10-fold cross-validation performance expressed as percentage of error (i.e., SMAPE) of each model for the five targets. The bar-plot on the top row indicates the number of free parameters for each model while the bar plot on the bottom row shows the average time for each training epoch. Both bar-plots are log_{10} scaled.

We want to stress that despite the proposed architectures improved upon the limitations of their predecessor, the largest effects have been observed mostly at the aggregated level or for some particular combinations of game contexts and targets. This might have been caused by either a true inefficiency of the proposed architectures or by a less thorough an ineffective hyper parameter tuning. Indeed, the inclusion of environmental and game events covariates substantially increased the number of degrees of freedom available for defining the architectures. In this view, the budged provided to the Hyperband algorithm might have not been sufficient for individuating the best hyper-parameters configuration ultimately leading to a sub-optimal performance.

In addition to the technical criticalities that we just described we can also individuate a number of methodological ones. First and foremost, we have to stress how the goal of this model building exercise has always been to create a methodology for approximating the latent state described by the work of McClure *et. al.* [13] and Zhang *et. al.* [14]. In this regard, our approach was formally different from that of TD Learning [5] and did not model the process of incentive salience attribution but rather attempted to approximate the product of this process (i.e., changes in attributed incentive salience). For this reason a direct comparison with the work of McClure *et. al.* [13] and Zhang *et. al.* [14] is difficult.

Moreover, unlike TD learning [34] our model is not guaranteed to converge on a quantification of V that is directly comparable to its biological counterpart or that has arisen from the same type of computations. This is also reinforced by the differences in mechanistic functioning between biological and artificial neural networks [172, 174]. These issues are partially attenuated by the constraints provided by our theoretical framework but in line with similar reports in the literature [18, 138] a verification based on controlled experiments is desirable.

Differently from the works of Calhoun *et. al.* [18], McClure *et. al.* [13] and Zhang *et. al.* [14], our methodology relied on a supervised learning approach to perform both prediction of future behaviour and latent state estimation, making this two tasks infeasible before any data was observed.

[5] See [26] for a review of the differences between supervised and reinforcement learning.

This limitation could be attenuated by initializing our model using a representation generated in an unsupervised manner.

The model comparison step in our experimental pipeline lacked of an important control condition: an architecture with linear recurrent operations. Indeed, despite we were able to assert the superiority of the combination of non-linearity and recurrency, we could not evaluate the impact of recurrency alone. Since the LSTM operation relies on non-linearity for both state generation and information filtering, we could not foresee which impact its removal could have had, hence we preferred to not burden with an additional architecture an already demanding part of the experimental pipeline. Future works should aim to clarify the contribution of recurrency alone on predictive accuracy and representation generation.

Lastly, despite our approach appeared to deal gracefully with objects having different structural characteristics (i.e., different in-game characteristics), these were limited to the domain of video games. In order to verify the generalizability of our approach, future work should include data generated from a variety of contexts (e.g. web services, online and laboratory-based experiments).

4.6 Summary

In this chapter we focused on implementing and validating the predictive model designed in chapter 3. This was done through an iterative approach that allowed us to construct, in a principled way, solutions of increasing complexity while also empirically test some of the theoretical assumptions introduced in chapters 2 and 3.

We showed that metric quantifying the strength of interactions between a group of individuals and a set of potentially rewarding objects can be effectively used by a computational model for predicting the intensity of future ones. A finding that is in line with a long tradition of experimental results in behavioural psychology [44, 158] and neuroscience [32, 34].

We also showed how a single global model can be used for generating these predictions across a wide range of objects. This idea resonates with the concepts of motivation and engagement as presented in chapter 2: single overarching processes able to describe and predict how strongly an individual will interact with an object.

It appeared evident that in order to produce more accurate predictions it was important to consider the full history of previous interactions rather summarizing them using a small number of descriptive metrics. From this, we can derive that for estimating constructs that are dynamic in nature (e.g., latent states like the level of attributed saliency), we must rely on data that reflect the characteristics of the process that generated them. In particular, this appeared to extend also to the type of computations performed by the model used for estimation. For instance, in our case we showed that techniques able to represent complex non-linear temporal dynamics in the data outperformed competing approaches, even when these had comparable computational power and expressiveness (i.e., architectures relying

on feedforward operations). As we highlighted in chapter 3, this is in line with the dynamical nature of motivation, incentive salience attribution and, by extension, engagement which can often be considered a temporal processes with memory [14, 38, 51–53]. Indeed, results from computational studies suggest that the attribution of value and saliency to potentially rewarding objects or actions is often carried out by non-linear recurrent operations [16, 138] and that ANN with recurrent connections are well suited for approximating these operations [146].

Finally we were able to partially support the idea that considering only information on the intensity of interactions was a limiting factor preventing the generation of latent representations with higher predictive powers. We showed how incorporating information on the characteristics of the objects and and the environment in which the interactions occurred improved, even if marginally, the accuracy of the model during the predictive task. These results are in line with findings in the literature showing how in-game mechanics are the source of the rewarding experience generated during the playing activity [65–67, 69, 71] and hence directly involved in its maintenance.

It is important to highlight that in this chapter we used model selection and predictive accuracy mostly as a device for testing hypotheses and guiding us towards the most promising approach for generating good approximations of the constructs we were trying to model. In this view, the next chapter will focus on extracting, visualizing and analyzing the representations produced by the RNN architecture and by its derivatives. This will be done in order to evaluate how these representations vary within and between models and to verify if they show a set of functional characteristics proper of attributed incentive salience. For doing so we will try to map the information embedded in the high dimensional representations generated by our ANN architectures back to the lower dimensional space of observed behaviour.

Chapter 5

Representation Analysis

5.1 Introduction

[1] In this chapter we will analyze the representations inferred by the architectures developed in chapter 4. Despite the fact that the predictive properties assessed in chapter 4 are a necessary condition for a good approximation of the motivational state of an individual (see section 3.3), they are not sufficient. In this chapter, our aim will be to evaluate if they possess some of the functional characteristics of attributed incentive salience: a true model of physiological motivational state.

In contrast to chapter 4, we will conduct our investigation using a combination of dimensionality reduction, unsupervised learning and visual analyses. We will focus on evaluating differences and similarities between the representations derived from the RNN architecture and its improved version (i.e., RNN with environmental and game events covariates) as they represent the final versions of our methodology.

We will first briefly introduce the idea that ANNs are able to learn a manifold structure of the data (which reflects the objective function that they are trying to optimize) by embedding it in a high dimensional latent representation. We will also clarify the procedure we followed for extracting these representations using our ANN architectures. Subsequently we will illustrate how it is possible to visualize their manifold structures by means of appropriate dimensionality reduction technique. After that we will define which type of functional characteristics we expect these structures to show and which differences we expect to see between the two architectures. Finally by means of unsupervised learning we will conduct a series of exploratory partition analysis on the representations generated by the ANN. This will be done in the attempt to individuate profiles able to map what inferred by

[1] All the code, associated with the work presented in this chapter can be found in the following repositories

- ApproxIncentiveSalience
- ImprovedApproxIncentiveSalience

the models back to the observable behavioural space. The steps of the analysis pipeline used in this chapter are illustrated in Figure 5.1.

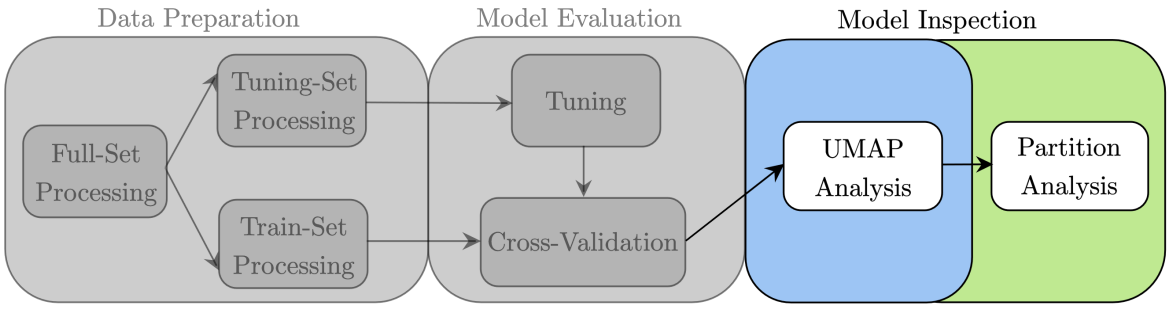


Figure 5.1: Arrows indicate the flow of the pipeline. Big coloured blocks are major pipeline steps, white rectangles indicate sub-tasks within each step. This experimental pipeline stems directly from the "Model Evaluation" stage outline in figure 4.1.

5.2 Extracting and Visualizing the Latent Representation

5.2.1 Neural Networks, manifolds and embeddings

As we mentioned in section 3, ANNs approximate a function by mapping the input they receive to a lower dimensional manifold. By moving along this manifold it is possible to reach inputs with different characteristics and observe how they relate with the output produced by the model.

Despite the fact that the manifold structure learned by the model might be intrinsically low dimensional, it is usually stored (or better, it is "embedded"), in a (potentially sparse) high dimensional representation [28]. This representation usually has fewer degrees of freedom than the original input but it is still challenging to parse from a human perspective. Simplifying, this can be compared to storing the "instructions" on how to extract the low dimensional manifold from the input in a distributed fashion across all the parameters of the ANNs. In our case if we look at Figures 4.6 and 4.13, the portion of the architecture marked in red should, once fitted to the input data, provide us with the relevant "instructions" on how to obtain an approximation of the manifold structure describing the level of attributed incentive salience (see chapter 3 for the theoretical reasons behind this assertion).

As illustrated in sections 3.3.3 and 3.3.4, since an ANNs can be thought as directed acyclic computational graphs (DAGs), to obtain the representation produced at any point of a specific architecture it is sufficient to pass a given input through all the operations performed before that point. Figure 5.2 illustrate the process for the RNN architecture presented in section 4.3.

Borrowing the terminology from the self-supervised deep learning literature [28] we call this truncated version of the original architecture encoder. Encoders can be thought as functions (whose parameters have been learned during the fitting procedure) mapping input data onto the manifold space learned

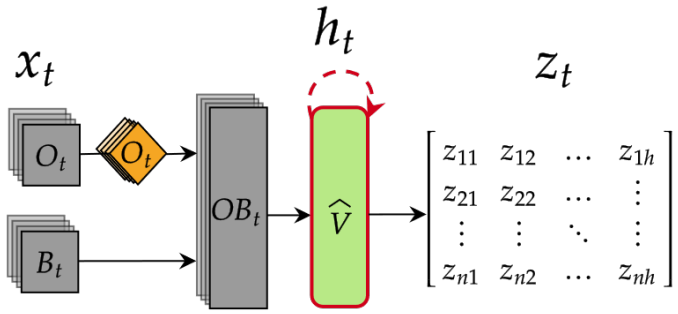


Figure 5.2: Orange and green shapes represent respectively embedding and LSTM layers. Embedding layers are a type of feedforward layers specifically designed for dealing with categorical inputs [180]. Gray shapes indicate operations with no learnable parameters, such as tensor instantiation and concatenation. The orange transparent shape indicate the concatenation of a single embedding with multiple tensors. Stacked, transparent colouring indicates arrays with a sequential structure. Straight and curved arrows refer to the presence of feed-forward or recurrent information flow. The red highlight shows the portion of the model that we hypothesize to be inferring an approximation of attributed incentive salience. Given inputs $O \in \mathbb{Z}^{N\times t}$ and $B \in \mathbb{R}^{N\times t}$, the matrix $Z_t \in \mathbb{R}^{N\times h}$ represents the h dimensional (where h is the number of hidden units in the recurrent layer) representation generated by the ANN at time t after all operations in its underlying computational graph have been performed.

by the original architecture. From now on, we will use the term encoder for referring to the two considered architectures truncated at the point of the last shared recurrent operation.

5.2.2 Dimensionality Reduction and Manifold Approximation

If we look at figure 5.2 we can see that with as the size of h increases, it becomes more and more challenging to inspect the representation generated by the encoder. However we recall from section 5.2.1 that the intrinsic dimensionality for this representation should be much smaller. In the case of a latent state like attributed incentive salience this could be as small as one dimensional, or two if we consider the nature of the rewarding object (see section 2.2.1 and Figure 2.1 in particular).

A convenient approach for inspecting the shape of this learned manifold would be to peform some form of dimensionality reduction. Principal Component Analysis (PCA) [200] would be a reasonable approach given the straightforward interpretation of the derived components. However the choice of which algorithm to chose is not necessarily that straightforward.

Looking at Figure 5.3, we can see the example of a dataset constituted by three separate (the separation aspect is important and will be further in section 5.3.1) point clouds (denoted by different colours) in a three dimensional ambient space (the space in which a low dimensional manifold might be embedded). Red and green clouds are examples of the synthetic Swiss Roll dataset [184], while the blue cloud is a simple random projection of a square. Both datasets are basically equivalent (i.e. intrinsically two dimensional with the main dimension highlighted by the colour gradient) but differ in their layout in ambient space: the square is linear while the Swiss roll warps in a non linear fashion.

Figure 5.4 shows a dimensionality reduction performed by PCA along with an alternative non-linear dimensionality reduction approach: the Uniform Manifold Approximation and Projection (UMAP) [201].

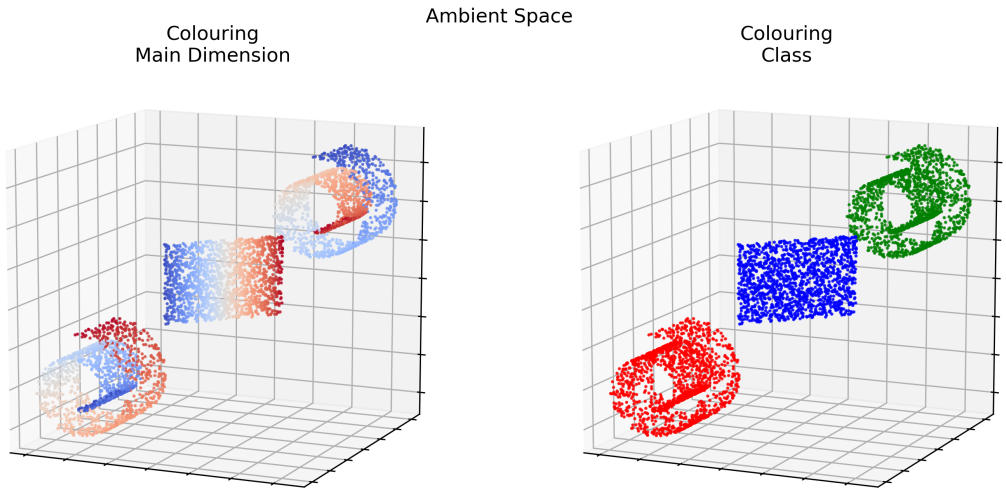


Figure 5.3: The figure shows three point clouds which are intrinsically two-dimensional (a Swiss roll and a square) embedded in a three-dimensional ambient space. In both panels the X, y and z axes represent the coordinates of the ambient space. The colours in the first panel indicates the main dimension of variation while those in the second panel simply identify membership of points to a specific cloud.

The UMAP algorithm is a dimensionality reduction technique based on manifold learning. Given a high dimensional dataset, UMAP first infers its topological structure by means of a k-nearest neighbours graph and then, using stochastic gradient descent, attempts to structurally reproduce it in a lower dimensional space (two or three for visualization purposes) [201]. Despite being a local algorithm, compared to other similar approaches (for example, the t-distributed Stochastic Neighbor Embedding [144]), UMAP has the ability to better preserve the global structure of the original data. Moreover, when given datasets that are sequential in natures (like those produce by the recurrent part of our architecture) UMAP is able to include this characteristics during the optimization process [2] generating lower dimensional representations that are aligned over time.

What we observe in Figure 5.4 is that both techniques manage to keep the separation between point clouds, but that PCA, unlike UMAP, struggles to faithfully represent the the intrinsic structure of the data. This toy example is particularly relevant in our case as the representations generated by ANNs are, by design, highly non-linear. This will be made evident in the next section where we will proceed at visualizing the representation learned by our ANNs architecture.

5.3 Representation Analysis

In order to support our idea that the representation learned by the different architectures could be interpreted as an approximation of the latent states generated by incentive salience, we carried out a series of qualitative analyses. If our intuitions were in the right direction, we expected the representations inferred by the architectures to exhibit a series of characteristics and functional

[2] See [202] for implementation details.

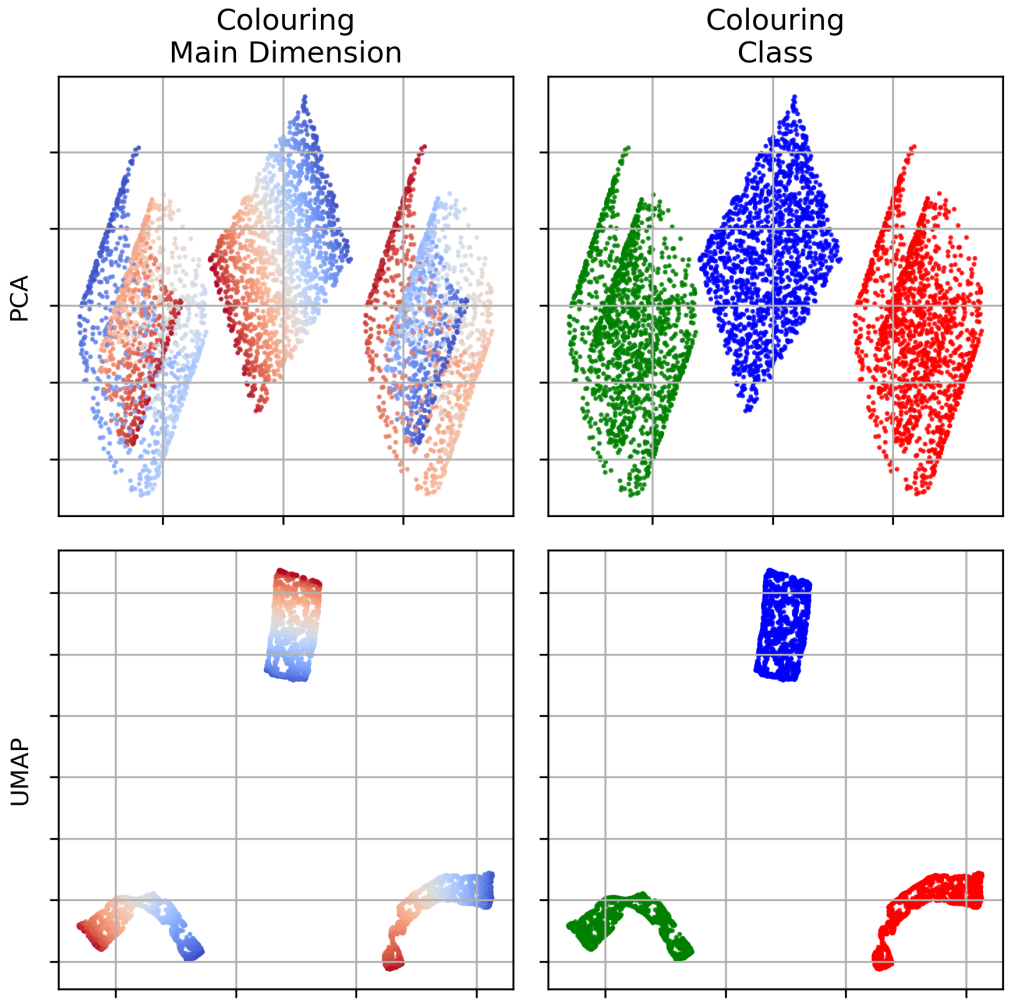


Figure 5.4: The figure show the reduction of the three-dimensional point clouds to a two-dimensional plane. The panels in the first row show the reduction produced by PCA while those in the second row show the reduction produced by UMAP. In all panels the x, y axes represent the components extracted by PCA and UMAP. The colours in the two columns again indicate the main dimension of variation and the membership of points to a specific cloud.

properties:

1. To possess an intrinsic dimensionality that is much smaller than the observed one.
2. To be able to effectively distinguish between different game objects.
3. To be able to effectively distinguish between individuals based on the expected intensity of their future interactions with the considered game objects.
4. To be able to show the two aforementioned characteristics consistently over time.

The characteristics mentioned above are concerned with a general validation of the properties of the latent representation for all the considered architectures, however we also wanted to understand if the included covariates altered and improved the generated representation. We therefore

inspected the representation extracted by the encoder derived from the improved version of the RNN architecture.

The procedure followed for extracting the latent representations was the same for all architectures. First, we re-fitted all models on a random sample (i.e. 90%) of the validation-set following the same procedures specified in chapter 4. Then, we created six encoders using the approach illustrated in paragraph 5.2.1 and Figure 5.2. Two were used for extracting the representations expected to approximate the level of attributed incentive salience (red highlights in Figures 4.6 and 4.13. One for extracting the same type of representation inferred however by the MLP architecture (this was done for comparative purposes). And three were used for extracting the intermediate representations generated by the improved version of the RNN architecture (i.e. those relative to the behavioural, environmental and game events input). Subsequently, we passed the remaining portion of the validation-set (i.e. 10%) as an input to the encoders, producing arrays of shape $(N \times T \times h)$ with N being the number of considered individuals, h the number of hidden units in the last layer of the encoder and T the number of observed interactions for the considered individuals. This means that all representations have been generated with out-of-sample data. In our case, since all architectures were of type sequence-to-sequence we were able to investigate not just the properties of the generated representation at specific point in time, but also the dynamics underlying their evolution.

To summarize, each considered encoder was tasked with generating a high dimensional representation where distance could be interpreted as similarity between individuals with respect to the expected intensity of their future interactions with a specific game object (see the manifold hypothesis of attributed incentive salience presented in paragraph 3.2). Dimensionality reduction was then used for approximating the intrinsic manifold structure of these representations on a two dimensional plane in order to allow for qualitative visual analysis.

The reduction to a 2D plane used the UMAP algorithm. The algorithm first inferred the topological structure of the produced representations by computing the cosine distance in a local neighborhood of 1000 points with a minimum distance of 0.8. The projection on a two dimensional plane was then achieved by running the optimization part of the algorithm for 2000 iterations. The choice of a large neighborhood and minimum distance was made to better capture the global structure of the manifold [3].

In order to gather an understanding on the characteristics of the function used for generating the latent representations, we also conducted a set of purely exploratory investigations of the relationship between hidden units' activation in the recurrent layers and the predictions produced by the model. To quantify the strength of the observed relationship we employed the Maximal Information Coefficient (MIC) [204], a measure of mutual information that can quantify both linear and non-linear association between variables. The MIC can assume values between 0 to 1 with 1 corresponding to a perfect association.

[3] See [203] for a visualization of the effects of these hyperparameters in UMAP.

We adopted the implementation of UMAP provided McInnes *et. al.* [30] while the MIC was computed using the python library minepy [205]. Visualizations were produced using the python libraries matplotlib [206] and seaborn [207].

5.3.1 Validating the Functional Properties of the Inferred Latent Representation

We first asked if the assumption about the intrinsic dimensionality of the latent representation was reasonable. Looking at Figure 5.5

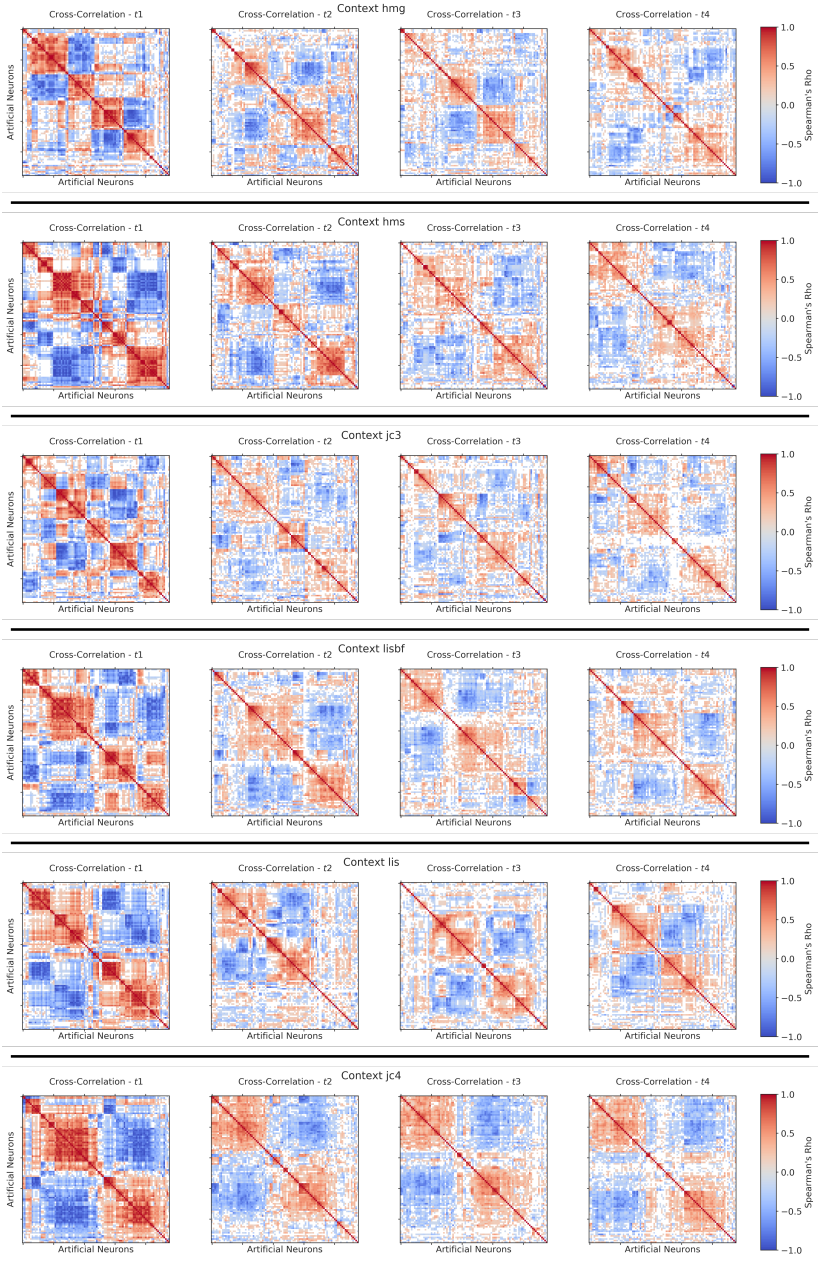


Figure 5.5: Each panel shows the cross correlation between the activity of the RNN's artificial neurons in all the game objects going from $t1$ to $t4$. The y and x axes are symmetrical and identify the RNN artificial neurons while the coloured cells indicate the Spearman's Rho correlation coefficient for the activation of each pair of neurons. White cells represent combinations for which the correlation coefficient was lower than 0.05.

we can observe consistent patterns of cross-correlation for the activity of the hidden units constituting the latent representation, suggesting the presence of redundancy. In order to support this finding and to gain a general sense of the intrinsic dimensionality of the embedded manifold, we conducted a Principal Component Analysis (PCA). Despite PCA and UMAP working under radically different assumptions and mechanisms, we thought this could provide us with a general idea of how much variance we would be able to capture by reducing the representation to a lower dimensional space.

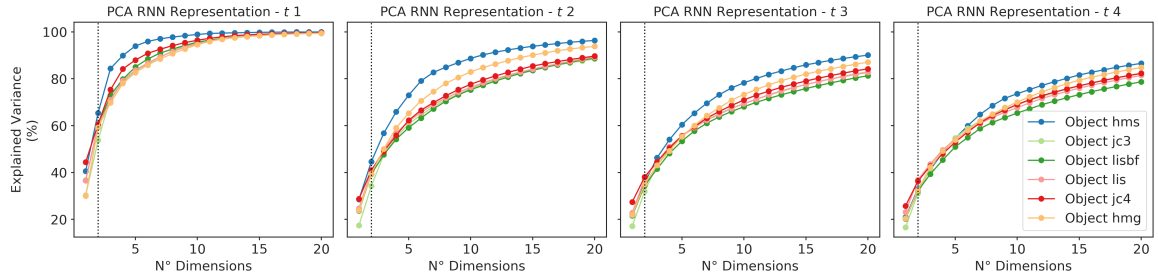


Figure 5.6: The panel shows the percentage of explained variance by considering 2 to 20 principal components for each game object going from $t1$ to $t4$. The y axis indicates the percentage of explained variance while the x axis the number of principal components considered.

Looking at Figure 5.6 we can see that two principal components can explain a large portion of variance in the representation generated by the RNN. To be precise, across game contexts the first two principal components were able to explain from 30 to 60% of the variance, with maximum explanatory power around 6 and 8 principal components. However, as we illustrated in section 5.2.2, relying on PCA alone could give us an incomplete and potentially distorted picture of the manifold structure inferred by the architecture. Looking at Figure 5.7

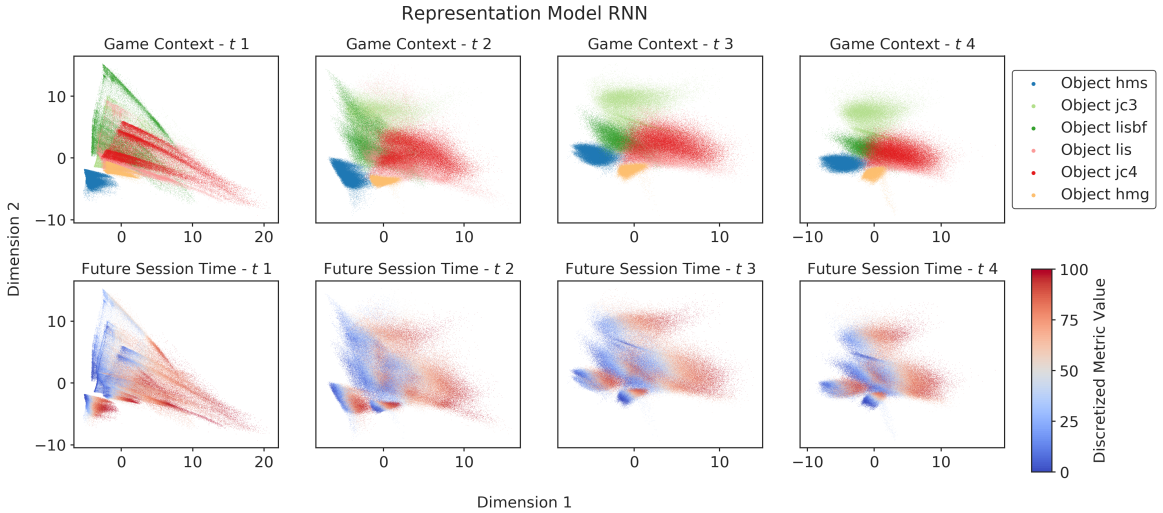


Figure 5.7: The figure shows the two-dimensional projection, produced by PCA, of the multi-dimensional representation inferred by the RNN for interactions going from $t1$ to $t4$. We can read the values of the x and y axes as the first two directions of maximum variation in the latent representation. Each point indicates the representation inferred by the RNN model after observing one game session from a single user. The colours in the first row indicate the game object from which the representation is coming. Colours in the second row represent the discounted sum of all future predictions for a particular target (for example, estimated Future Session Time) $\widehat{B}_{t2:T}$ which is given by $\sum_{i=0}^{t2:T} \gamma^i \widehat{B}_i$ with $\gamma = 0.1$ as illustrated in the TD Learning equation 3.1

we can see that the representation seems to embed a single gradient line that distinguishes individuals

based on the expected length of their future interactions (here future session time) with the considered game objects. However, the low dimensional projection appears to be organized in a disorderly manner, with game contexts blending into each other causing the gradient line to look discontinuous.

The projection produced by the UMAP algorithm, on the other side, appears to provide a better picture of the inherent structure of the inferred representation. Looking at Figure 5.8A

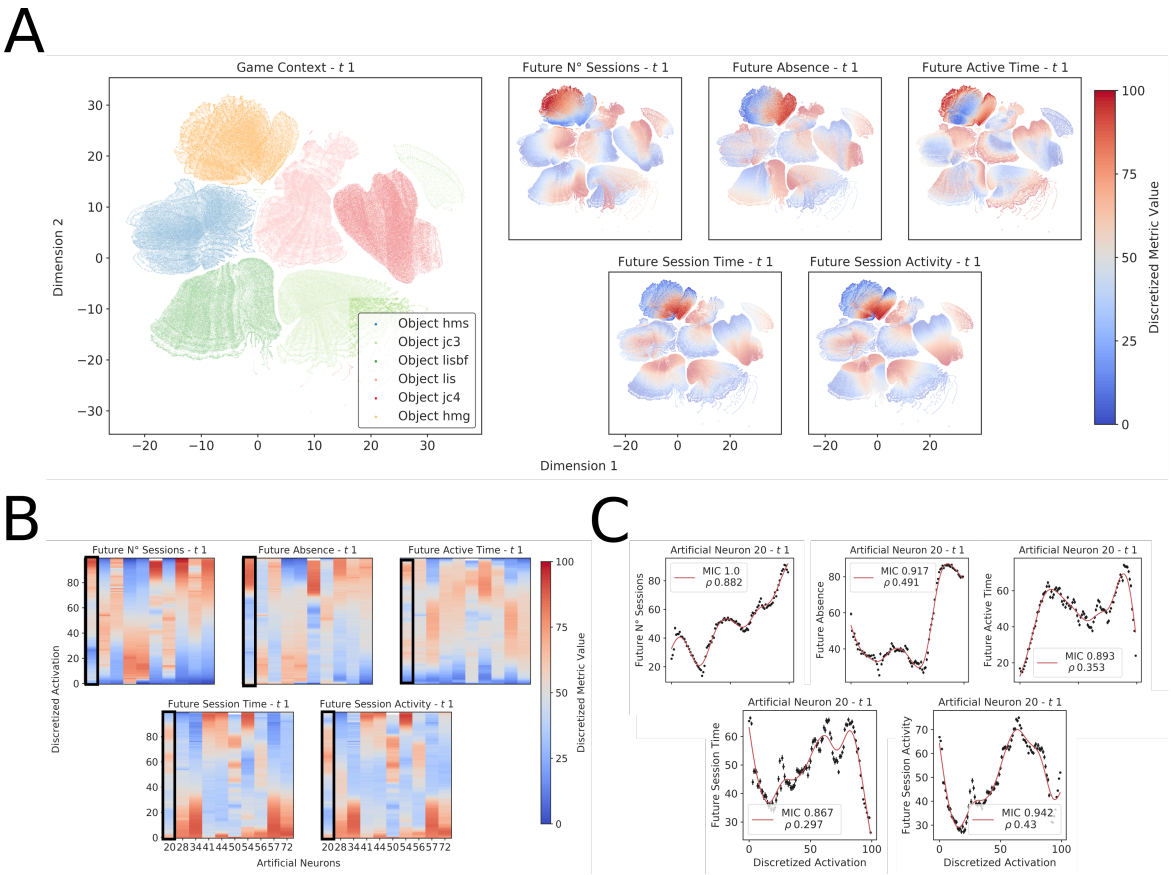


Figure 5.8: The representation generated by the RNN model distinguishes between different game objects while maintaining an overarching organization able to capture variations in the expected intensity of future interactions that individuals will have with a specific game object. Panel A shows the two-dimensional projection, produced by UMAP, of the multi-dimensional representation inferred by the RNN at **t1** as produced by UMAP. We can read the values of the x and y axes as a coordinate system where proximity represents similarity between points in the original high-dimensional space. Each point indicates the representation inferred by the RNN model after observing one game session from a single user. The colours in the Game Context panel indicate the game object from which the representation is coming. Colours in the small panels represent the discounted sum of all future predictions for a particular target (for example, estimated Future Session Time) $\widehat{B}_{t2:T}$ which is given by $\sum_{i=0}^{t2:T} \gamma^i \widehat{B}_i$ with $\gamma = 0.1$ as illustrated in equation 3.1. Each unit encodes the intensity of future interactions through multiple non-monotonic functions. Panels B and C show the relationship between the activation of randomly-selected hidden units in the LSTM layer of the RNN and the model's predictions at **t1**. Panel B shows the relationship between the discretized activation of 10 randomly selected units (artificial neurons) plotted along the y axis and the predictions made by the model at $t1$ (colour coded from blue to red as in the small panels in A) for the game object *hmg*. Panel C shows in more detail the relationship between discretized activation and RNN predictions for a single unit highlighted by a black box in Panel B. Here the x axis indicates the discretized activation while the y axis the mean discretized discounted sum of all future predictions produced by the model. Vertical lines are standard errors of the mean. The red curve is the line of best fit provided by a generalized additive model [208] while the box reports the MIC and the correlation coefficient (Spearman's ρ) between the artificial neuron activation and the model's predictions.

we observe how the model was able to effectively distinguish between different game objects while simultaneously encoding for variations in the expected intensity of future interactions. This is illustrated by the fact that each game object occupies different and distinct regions in the representation

space while showing a within-object gradient-like organization that places individuals (i.e. single dots) on a continuum based on the estimated magnitude of their future behaviour.

This organization is preserved for each of the six targets showing how the representation inferred by the model is a suitable meta-descriptor for different behavioural indicators. As expected, some targets show a very similar but not identical organization (e.g. Future Session Time and Future Session Activity) while others appear to be independent (e.g. Future Session Time and Future Absence). We note that the absolute location of each game aggregate (i.e. all the points belonging to a specific game object) on the 2D plane is arbitrary.

Panels 5.8B and 5.8C provide more insight into the activation profiles of individual hidden units constituting the generated representation. Panel 5.8B shows the relationship between the activity of 10 randomly-chosen units and the predictions generated for the five targets. These are essentially transducer functions illustrating how the estimate for a particular target varies (on average) as the output of a units increases or decreases. Each unit seems to encode for multiple non-monotonic functions, one for each of the considered targets. Differences in the shape of these functions reflect similarities between their associated targets. For example, the functions associated to two highly related targets like Future Session Time and Future Session Activity (see panel 5.8A) appear to be very similar in shape (see panels 5.8B and 5.8C). Interestingly, although most units appear to encode for unique functions some of them (e.g. 41 and 44) show an almost identical behaviour. This suggests the presence of redundancy in the functions underlying the representation generated by the RNN model. These observations are clarified in panel 5.8C, where the functions associated with a single unit (20, indicated by a dark box in 5.8B) are presented. Here we observe a strong, non-linear relationship between the unit's activity and the estimated targets (see the high MIC values and the line of best fit). In addition, the between-targets variation in MIC values suggest how the chosen unit is not equally informative for all targets but rather shows specialized behaviour.

The analyses in Figure 5.8 were performed at a single time point $t1$. However, as we mentioned before, the characteristics of our architectures allowed us to also evaluate how the latent representation evolved over time.

Looking at Figure 5.9, we can see that the characteristics detected at a single point in time remain qualitatively consistent over time. For example, focusing on Future Session Time (see Appendix D.1 for results connected to other targets), we see in Figure 5.9A that the model's ability to segregate different game objects while providing an overarching representation of the intensity of future interactions is preserved over time. This supports the hypothesis that the representation inferred by our model is dynamic in nature which is further corroborated by panel 5.9D. There we can see how the RNN model was able to individuate a "space" with temporally consistent "hot" and "cold" regions between which individuals gradually moved over time depending on the expected intensity of their future interactions. This means that given the history of interaction of a particular individual with a specific game object, our model would determine their "position" (i.e. their "internal state") in a latent space indicative of the expected intensity of future interactions with that object. If we recall our

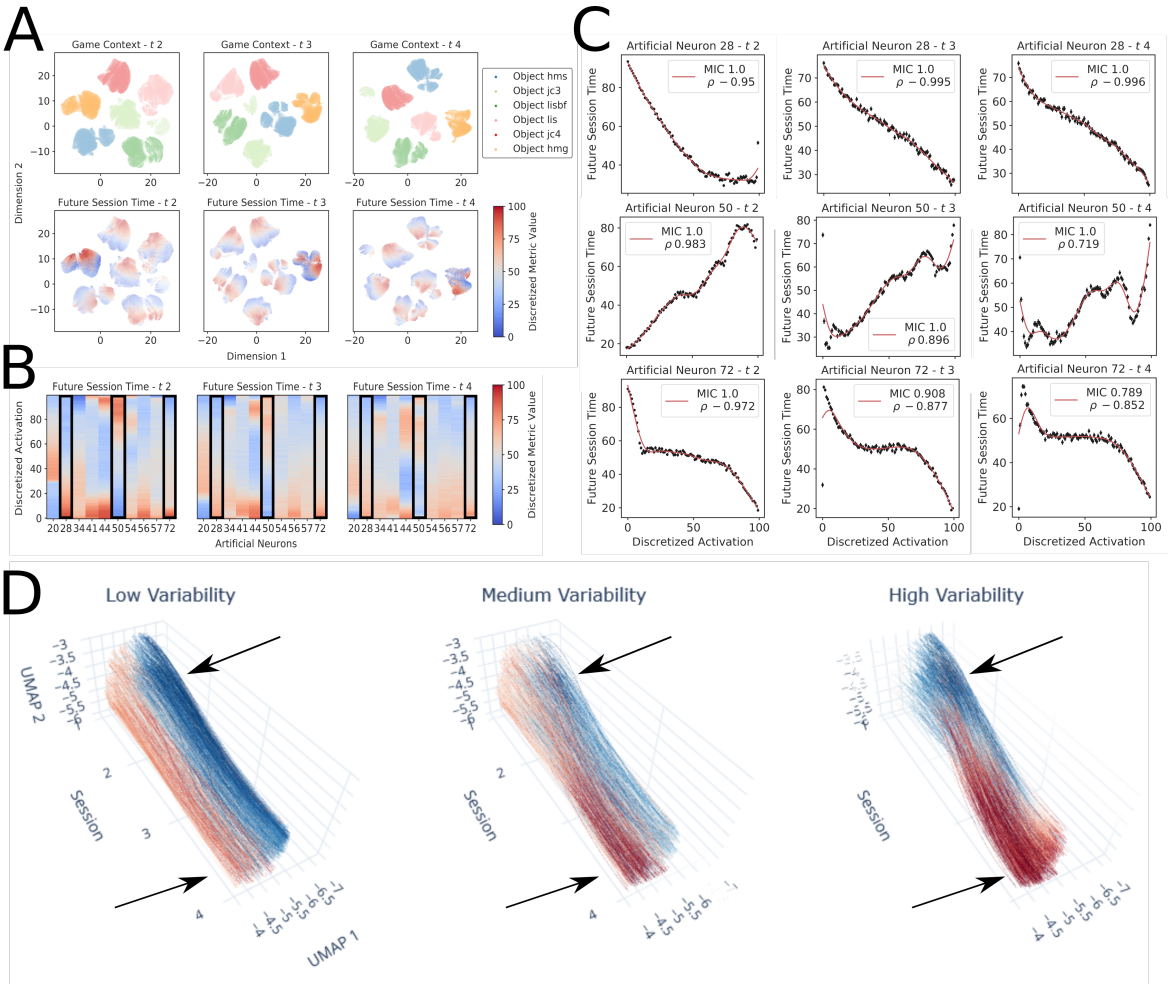


Figure 5.9: The representation generated by the RNN model appears to maintain its discriminant properties over time. Panel A shows a two-dimensional projection of the multi-dimensional representation inferred by the RNN at $t2$, $t3$ and $t4$. The inferred representation maintains its gradient-like organization over time with an increased ability to differentiate between game objects. As in Figure 5.8, x and y axes are dimensions individuated by the UMAP algorithm and can be interpreted as a coordinate system where proximity represents similarity between points. Colours in the first row indicate which game object the representation is coming from while those in the second row indicate the discounted sum of future predictions for a single target (i.e. "Future Session Time"). **The units constituting the generated representation encode for functions that are consistent over time.** Panels B and C show the relationship between units' activation and the model's predictions over time for the game object *hmg*. Different units appear to encode the same target with different non non-monotonic functions which are relatively consistent over time. Panel B illustrates the relationship between the same 10 randomly selected units specified in figure 5.8 and the predictions made by the model for Future Session Time at $t2$, $t3$ and $t4$. Panel C shows in more detail the relationship of the three artificial neurons, highlighted by black boxes in B, across time. Each row is a different unit while each column corresponds to a different t. The x axis indicates the discretized activation while the y axis the mean discretized discounted sum of all future predictions. Vertical lines are standard errors of the mean. The red curve is the line of best fit provided by a generalized additive model [208] while the box report the MIC and the correlation coefficient (Spearman's ρ) between the artificial neuron activation and the model's predictions. **The generated representation produces areas of low and high expected intensity among which individuals move over time.** Panel D shows trajectories through time produced by a version of UMAP that incorporates temporal information. Data are drawn from random subsets of individuals having low, medium and high variability in their expected amount of future behaviour. The representation inferred by the RNN model produces "hot" (i.e. the left side) and "cold" (i.e. the right side) regions, representing high and low expected Future Session Time, that are spatially consistent over time. Individuals appear to either stay in the same region or to move between regions over time. Here each line represents variations in the representation generated by the RNN model for a single user over four temporal steps. Continuity is generated by means of cubic spline interpolation for the lines and by linear interpolation for the colours. The x and y axes are the dimensions individuated by the UMAP algorithm while the z axis indicates the associated point in time. Colours indicate the discounted sum of future predictions produced by the model at a specific point in time.

definition of attributed incentive salience from chapters 2 and 3 we can suggest that the model has inferred the location of individuals in what is an approximation (from the functional point of view)

of the "attributed incentive salience space". This aligns with the manifold hypothesis mentioned in sections 3.2 and 3.3.4: changes in the propensity to interact with a specific game object (i.e. variations in the amount of attributed incentive salience) can be expressed moving on a manifold embedded within an h dimensional space, with h being the dimensionality of the representation generated by our RNN model.

It appears that the hidden units constituting this representation tend to be consistent over time in the type of functions they encode (see Figure 5.9B and C). As expected, we can again observe a strong non linear association between units' activation and targets' predictions, see MIC values and lines of best fit. The decrease in MIC value observed in Figure 5.9C for the artificial neuron 72 might indicate how certain units lose their informative power over time.

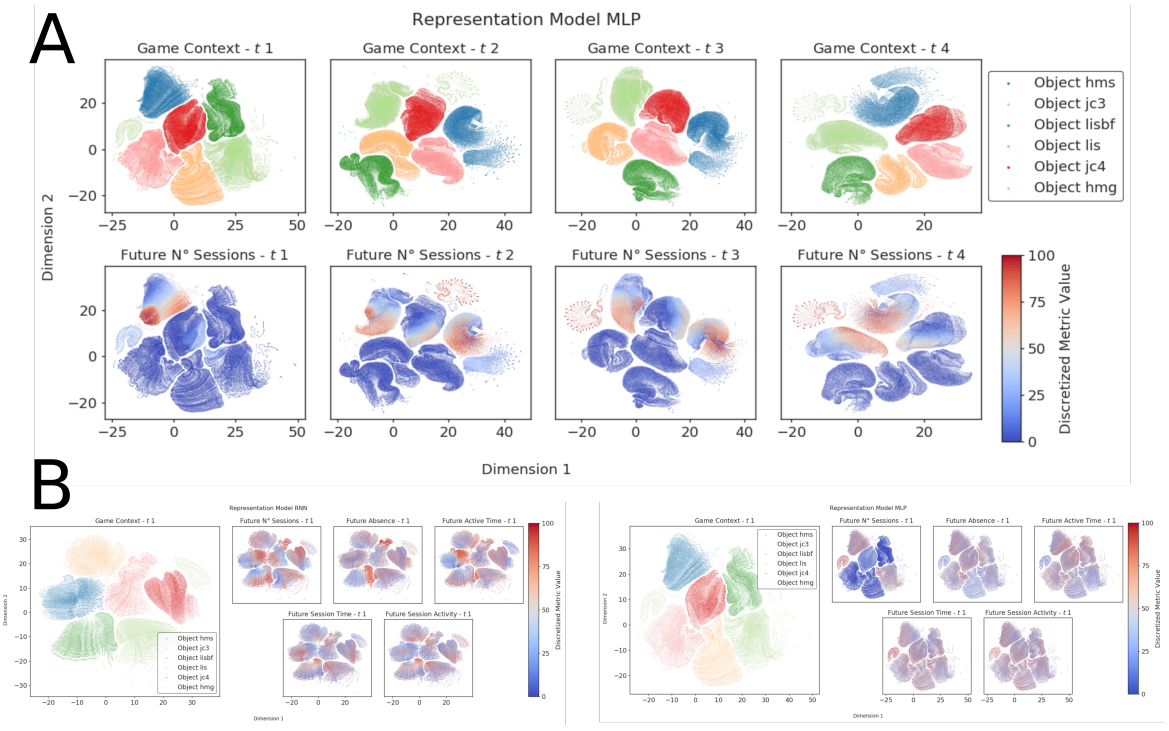


Figure 5.10: The representation generated by the MLP model is less effective at distinguishing between different game objects and different levels of expected future behaviour intensity. Panel A shows a two-dimensional projection of the multi-dimensional representation inferred by the MLP at $t1$, $t2$, $t3$ and $t4$. Differently from the RNN, the representation shows a disruption in the gradient-like organization and a reduced ability to differentiate between game objects which remain constant over time. The x and y axes are dimensions individuated by the UMAP algorithm and can be interpreted as a coordinate system where proximity represents similarity between points. Colours in the first row indicate which game object the representation is coming from while those in the second row indicate the discounted sum of future predictions for a single target (i.e. "Future N° of Sessions") **The representation generated by the MLP model is less effective at at distinguishing different levels of expected behaviour intensity for states that are further away in the future.** Panel B shows a two-dimensional projection of the multi-dimensional representation inferred by the RNN (left) and MLP(right) at $t1$ but colour coded with the discounted sum of future predictions from $t4$ onward. The representation generated by the RNN is able to maintain a gradient-like organization even from states that are further away in the future while this capacity is almost entirely lost for the MLP. The colours in the Game Context panel indicate the game object from which the representation is coming. Colours in the small panels represent the discounted sum of all future predictions for a particular target computed from $t4$ onward instead that from $t1$. The x and y axes are the dimensions individuated by the UMAP algorithm.

If we look at the differences between MLP and RNN-like architectures from section 4.3, we can see that they all aim to solve the same task: predict the intensity of future behaviour given the history of interactions. They do so relying on the same type of metrics, leveraging similar computational

mechanisms (i.e. multitask learning and non-linearity) and producing representation according to the same underlying principle (i.e. the manifold hypothesis). Nevertheless, the fact that MLP architectures consistently provided poorer fit to data already suggests that whatever representation it had inferred it was likely a sub-optimal approximation of the manifold structure of attributed incentive salience.

Looking at Figure 5.10A, and knowing that UMAP represents differences and similarities between points through distance, we can see how the representation generated by the MLP less clearly differentiate between game objects. On the same figure, we can notice how the gradient representation for the metric Future N° Sessions is largely disrupted. This effect is however consistently less pronounced for other metrics (see Appendix D.2 for additional visualizations) and in accordance to the differences in predictive perfromance that we observed in chapter 4.

Recalling what mentioned in section 3.3, the latent state produced by the level of attributed incentive salience should retain at any point in time some predictive power over the intensity of all the future interactions (i.e. not just the one that follows). Figure 5.10B shows the representation generated by RNN and MLP at $t1$ but color coded with the discounted sum of the predictions made from $t4$ onward. We can see that, even if degraded, RNN still preserves some of the desired gradient-like organization which is instead much more disrupted for MLP. This is in accordance to what is shown by Figure 5.9D: the RNN appears to define regions of high and low expected behavioural intensity which are consistent over time rather than generating representations that are mostly representative of what is expected at $t + 1$.

5.3.2 Evaluating the Contribution of Environmental and Game Events Covariates

Shifting our attention to the representations generated by the improved version of the RNN architecture we can see some noticeable differences. Looking at Figure 5.11

we can see how the representation generated from the behavioural metrics changes considerably from the simple RNN architecture. Despite the ability to differentiate between game objects is, up to a certain degree, preserved, the quality of the gradient organization is markedly diminished. This is doesn't come as a surprising result: similarly to what happens when covariates are added in a linear model, the behavioural metrics are now only one of the components making up the final representation in charge of producing the model's predictions. It is however worth noticing that the ability to differentiate between individuals with respect to the intensity of their future interactions is not completely removed suggesting the intensity of past interactions still plays a role in determining the intensity of future ones.

The same cannot be said for the representation generated by the environmental metrics. If we look at Figure 5.12 we can see that not just the ability to differentiate between game objects is almost completely disrupted, but also the capacity to distinguish between individuals with high and low expected intensity of future interactions.

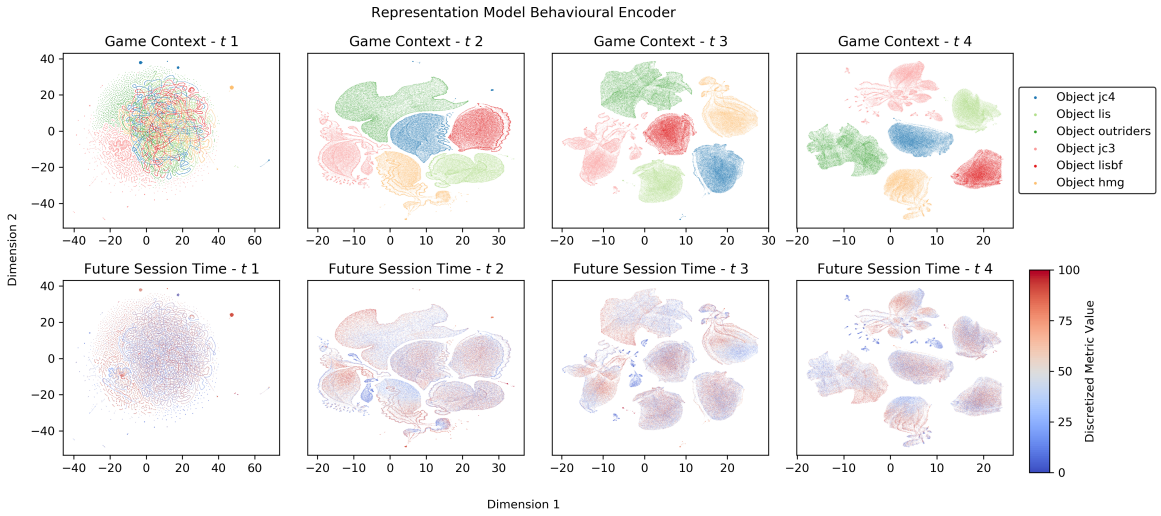


Figure 5.11: Each panel shows a two-dimensional projection of the multi-dimensional representation inferred by the improved RNN architecture at $t1$, $t2$, $t3$ and $t4$. The representations presented in this figure have been generated by the portion of the architecture receiving the behavioural metrics as input. As in Figure 5.9, x and y axes are dimensions individuated by the UMAP algorithm and can be interpreted as a coordinate system where proximity represents similarity between points. Colours in the first row indicate which game object the representation is coming from while those in the second row indicate the discounted sum of future predictions for a single target (i.e. "Future Session Time").

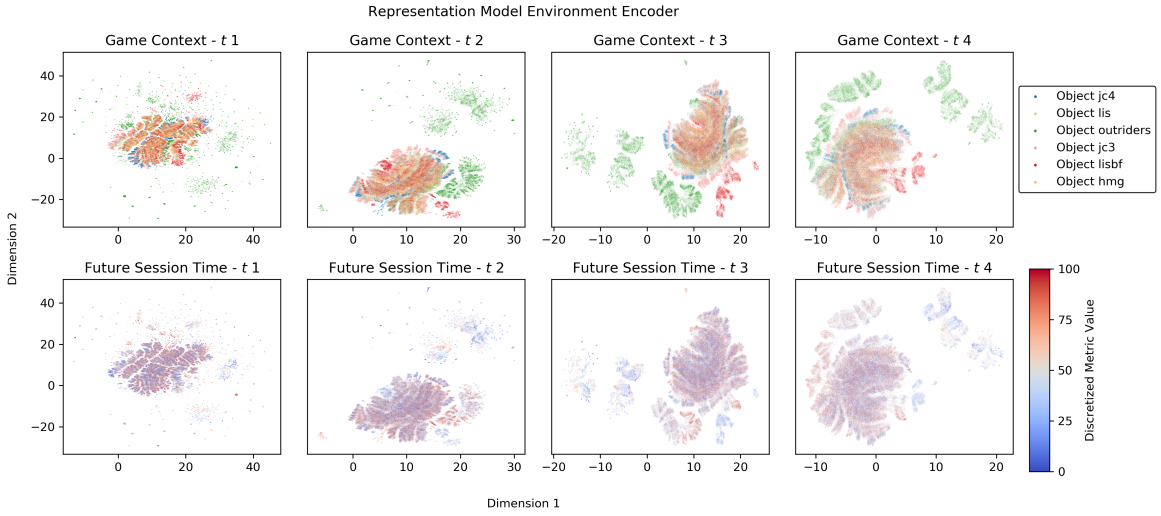


Figure 5.12: Each panel show a two-dimensional projection of the multi-dimensional representation inferred by the improved RNN architecture at $t1$, $t2$, $t3$ and $t4$. The representation presented in this figure has been generated by the portion of the architecture receiving the environmental metrics as input. As in Figure 5.9, x and y axes are dimensions individuated by the UMAP algorithm and can be interpreted as a coordinate system where proximity represents similarity between points. Colours in the first row indicate which game object the representation is coming from while those in the second row indicate the discounted sum of future predictions for a single target (i.e. "Future Session Time").

As we anticipated in sections 4.4 and 3.3.5, we did not expect the environmental variables to have strong predictive power on future ammount of gaming behaviour, but rather to act as an absorbing factor for possible noise observed in the behavioural metrics. Or better, we argued that environmental factors might affect the behavioural manifestations of a certain latent state (i.e. attributed incentive salience) both in the past and in the future, but do not play a central role in the shaping of the state itself (i.e. the level of attributed incentive salience). We will expand more on this section 5.4.2).

Another possibility is that the environmental information do not play a relevant role in generating a latent representation with good predictive power and are just treated as noise by the model. However, by looking at the occasional improvements in predictive performance that they provided in section 4.4.4, it seems more plausible that they are to be considered as nuisance metrics supporting the role of the other model inputs.

The representation generated from the game events information sits in stark contrast to the previous two. Looking at Figure 5.13 we can see how different it is from what we observed in Figure 5.11. The representation appears highly fragmented but with each fragment consistently belonging to distinct game contexts. This suggests the representation attempted to encode the heterogeneity in sequences of game events while maintaining them within the same context.

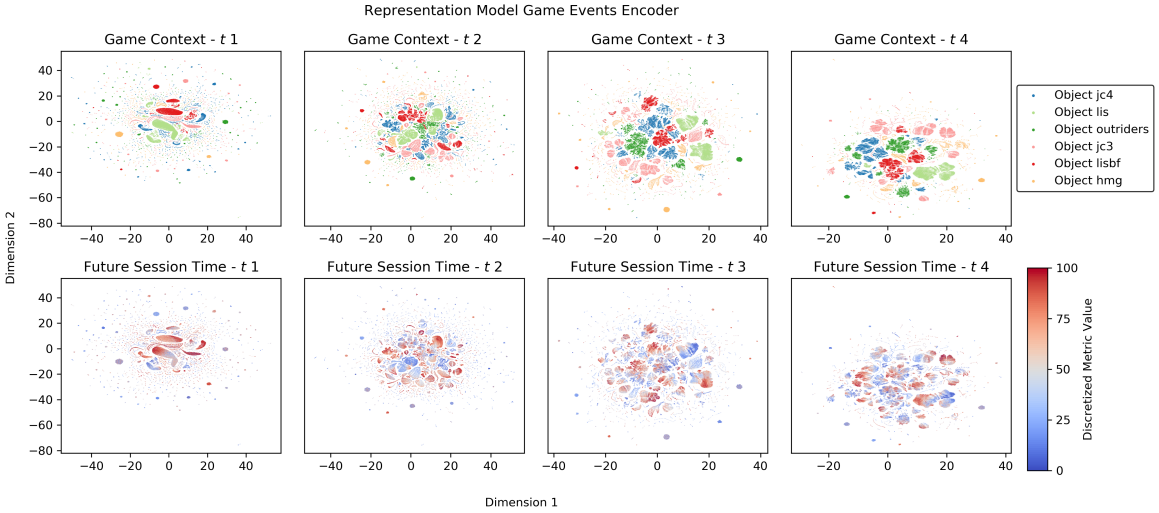


Figure 5.13: Each panel show a two-dimensional projection of the multi-dimensional representation inferred by the improved RNN architecture at $t1, t2, t3$ and $t4$. The representation in this figure has been generated by the portion of the architecture receiving the game events metrics as input. As in Figure 5.9, x and y axes are dimensions individuated by the UMAP algorithm and can be interpreted as a coordinate system where proximity represents similarity between points. Colours in the first row indicate which game object the representation is coming from while those in the second row indicate the discounted sum of future predictions for a single target (i.e. "Future Session Time").

The observed level of fragmentation is in line with all the possible sets arising by the combination of the considered game events and their relative frequency of interactions. For example considering the tuple $\{event, frequency\}$ we could have:

$$\begin{gathered}\{\{combat,10\},\{explore,3\},\{achievement,2\}\} \\ \neq \\ \{\{combat,2\},\{explore,10\},\{achievement,5\}\} \\ \neq \\ \{\{dialogue,2\},\{puzzle,10\},\{achievement,5\}\} \\ \neq \\ \ldots\end{gathered} \tag{5.1}$$

This type of behaviour is comparable to what can be observed when analyzing the word embeddings of different text corpora (e.g. see the Open Syllabus project [209]) and suggest that the architecture was able to discern differences in the sequential choices made by the individuals when interacting with different in-game elements.

What is most interesting however is that among the representations inspected so far, the one generated from the game events metric is the one that best preserves the gradient organization observed for the simple RNN architecture. This suggests that the sequences of observed interactions with specific in-game mechanics plays a role in differentiating between individuals with respect to the expected intensity of their future interactions with a specific game object. This, in turn, is something that we anticipated in sections 2.3.1 and 3.3.5 and that is in line with previous findings in the videogame literature [132].

Finally, when looking at the shared representation in Figure 5.14, we can see how it resembles the one generated by the simplified version of the RNN architecture with the only difference being a more clearly defined gradient organization. This representation is functionally equivalent to the one extracted by the simplified RNN architecture (i.e. it is used for performing multi-task learning) and is hypothesized to approximate the manifold structure of attributed incentive salience.

This qualitative improvement can be better appreciated when comparing the representation generated by the two version of the RNN architecture, color coded using the ground truth values rather than the predictions provided by the models. Looking at Figure 5.15

we can see how the simplified RNN architecture shows a greater level of disruption in the gradient like organization than its improved version, suggesting that this last one is likely to be a better approximation of the functional properties of the latent state (i.e. the level of attributed incentive salience) that generated the observed behaviour. By looking at figure 5.14, we can notice how the topological characteristics of the previous three representations are replaced by the global-local organization that we described in 5.3.1. This consistency corroborates the idea that this type of organization is the most suitable one for the type of predictive task that the two architectures are aiming to solve.

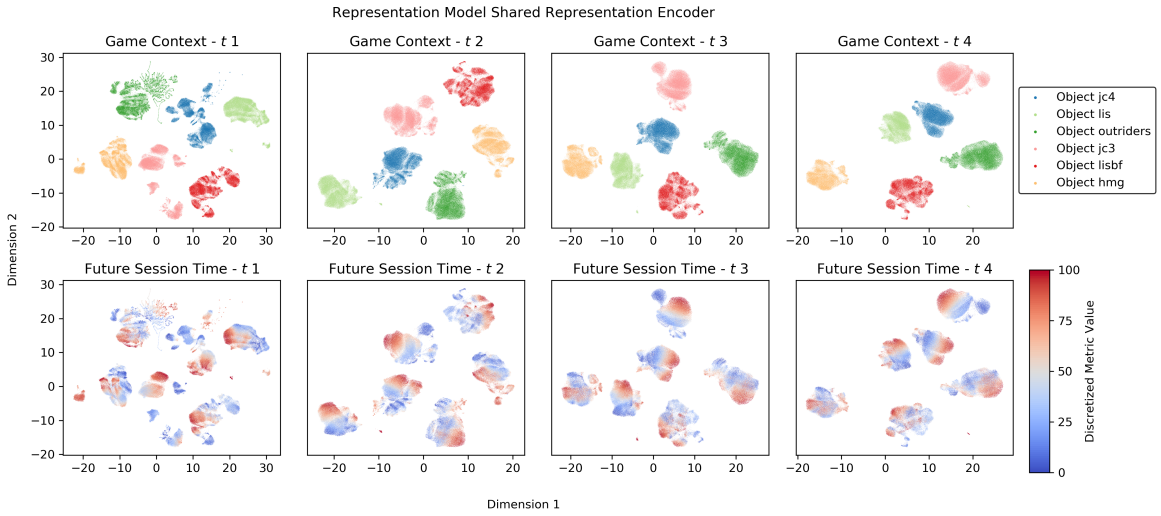


Figure 5.14: Each panel show a two-dimensional projection of the multi-dimensional representation inferred by the improved RNN architecture at $t1$, $t2$, $t3$ and $t4$. The representation presented in this figure has been generated by the portion of the architecture receiving as inputs the representations associated with the behavioural, environmental and game events input and is the one hypothesized to approximate the manifold structure of attributed incentive salience. As in Figure 5.9, x and y axes are dimensions individuated by the UMAP algorithm and can be interpreted as a coordinate system where proximity represents similarity between points. Colours in the first row indicate which game object the representation is coming from while those in the second row indicate the discounted sum of future predictions for a single target (i.e. "Future Session Time").

5.4 Partition Analysis

In order to gather additional insights on the functional properties of the representations generated by our architectures, we attempted to map what inferred by the models back to the observable behavioural space. As specified in section 3.3.4 these representations are derived from the input metrics and can be interpreted as coordinates on the manifold structure inferred by the architectures. In this view, partitioning them allows identify areas of the manifold holding information about the history of interactions between an individual and a given video game object. Moreover, since the manifold is constructed to be informative of the intensity of future interactions, different partitions might represent not just variations in the input metrics (e.g. differences in the sequences of events triggered in the game) but also in the level of attributed incentive salience. By individuating the set of metrics that contributed to defining the topology of different regions of the inferred manifold we may hope to gather insights on their role in determining the intensity of future behaviour.

To perform the mapping, we opted for an unsupervised approach and conducted a partition analysis on the representations extracted by the different encoders. To partition the data, we decided to apply Mini-Batch K-Means [210], a variation of K-Means, to the representation extracted by the three encoders mentioned in section 5.3. Given a dataset, the algorithm attempts to divide it by iteratively moving k centroids so as to reduce variance within each partition. The choice of Mini-Batch K-Means was dictated by the fact that it is one of the few distance-based algorithms that scales to very large datasets. The reason for choosing a distance-based algorithm can be found in paragraph 3.3.4, there we specified how distance in the manifold structure inferred by an ANN can be interpreted as a measure of similarity between its input with respect to the objective function that the model is trying

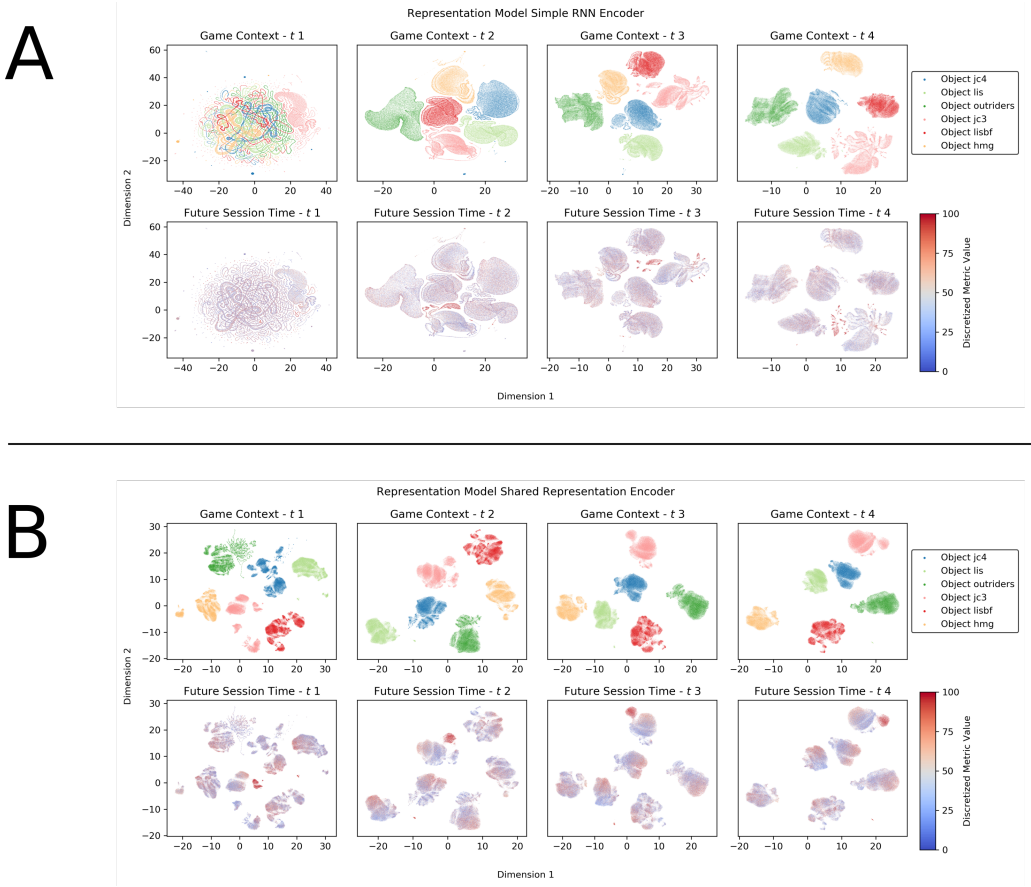


Figure 5.15: The figure show the two-dimensional projection, produced by UMAP, of the multi-dimensional representation generated by the RNN architecture and its improved version at $t1$, $t2$, $t3$ and $t4$. Both representations have been extracted by the portion of the architecture hypothesized to approximate the manifold structure of attributed incentive salience. Panel A refers to the RNN architecture while panel B to its improved version (including environmental and game event covariates). As in Figure 5.9, x and y axes are dimensions identified by the UMAP algorithm and can be interpreted as a coordinate system where proximity represents similarity between points. Colours in the first row indicate which game object the representation is coming from while those in the second row indicate the discounted sum of future ground truth values for a single target (i.e. "Future Session Time").

to minimize.

To select the optimal k value, we first fitted the algorithm with a varying number of centroids (i.e. 2 to 10) and computed the associated inertia, a measure of within cluster variance (see A.11. Since inertia tends to zero as k approaches the number of points in the dataset, we defined the optimal number of partitions as the value of k at which the inertia reached its "elbow" or maximum curvature [211]. This allows us to identify the point at which increasing the number of partitions provides diminishing returns in terms of within cluster variance reduction. Despite the fact that this procedure might be prone to errors or imprecision (e.g. the elbow might be non-unique or change depending on the maximum number of considered centroids) we thought it would be preferable to an arbitrary choice.

Every instance of Mini-Batch K-Means was initialized 3000 times at random and ran for a maximum of 3000 epochs. The input data were re-scaled to have zero mean and unit-variance and passed to the algorithm in random batches of size $(512 \times h)$. The associated behavioural profiles were

found by applying this methodology separately to each game object and retrieving, for each partition, the expected deviation of all the behavioural metrics from their relative mean (computed along the temporal dimension) in each game context. In order to have an indication of the quality of the individuated partitions, we decided to compute the average silhouette score (see appendix A.12). The silhouette score can be interpreted as an index of cluster cohesion. It is bounded between -1 and 1 and a high value indicates that, on average, all the considered points are well matched to their own partition and poorly matched to neighboring partitions, while the reverse is true for low or negative values.

We will first focus on inspecting the partitions derived from the representations generated by encoding the behavioural metrics using the RNN architecture and its improved version. We will subsequently move onto the partitions associated with those representation related to environmental and game events covariates. When analysing the obtained partitions we will focus only on those related to the game object *outriders*, results related to other game objects will be reported in appendices D.4, D.5 and D.6 and mostly used for drawing general remarks.

The Mini-Batch K-Means implementation used for this analysis was provided by the python library scikit-learn [184]. All the analyses were conducted using Python programming language version 3.6.2 [212].

5.4.1 Partitioning the representation associated to the behavioural inputs

As we can see from Figure 5.16, following the methodology outlined in section 5.4, among all the Mini-Batch K-Means runs, the one with $k = 4$ was identified as the optimal one for both representations. All the partitions were associated with a distinct behavioural profile, each one with its own offset and temporal evolution. It is relevant to note that the profiles identified for the two representations are virtually identical (apart from small variations in some of the metrics). However the representation extracted by the improved version of the RNN architecture appears to allow for qualitatively superior (i.e. more compact) partitions as highlighted by the higher average silhouette score.

At a global level, the four partitions seem to belong to two general groups: a group with a high propensity to produce future interactions (i.e. partitions 1 and 2) and a group with low propensity (partitions 3 and 4). Noticeably, when looking in detail at each specific partition they appear as variations on the macro group they belong to. Interestingly the percentage of Session Time spent actively interacting with the game object (i.e. Active Time) and the expected N°Sessions seem to be a relevant components in this more granular characterization.

We will now report examples of the type of behavioural "phenotype" that we were able to derive from the partition analysis. However, Given the unsupervised nature of the adopted methodology and the lack of any strong a-priori expectations on the profiles' outlook, we suggest to interpret them with cautions and consider them as the result of a purely exploratory analysis.

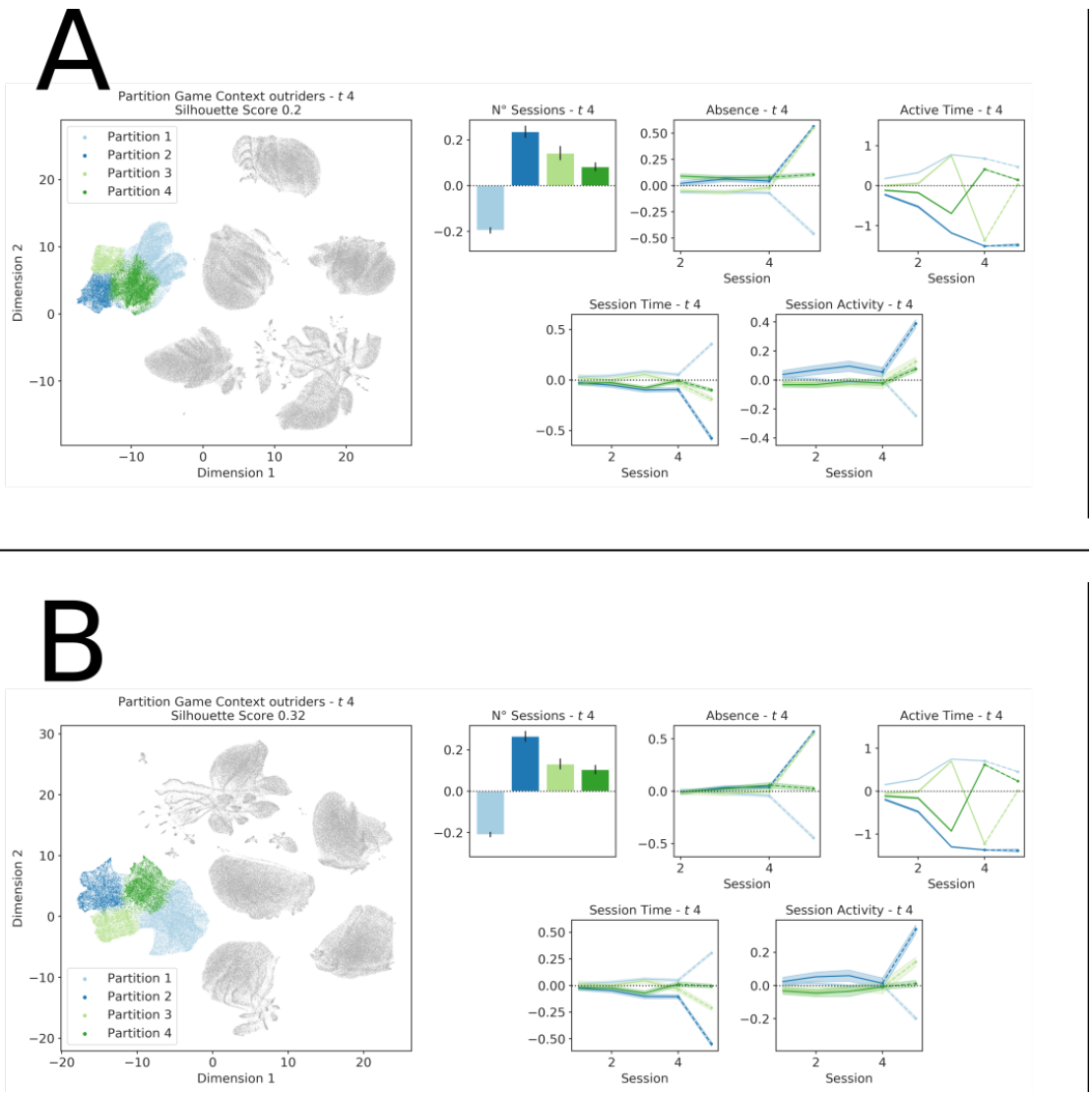


Figure 5.16: The two panels shows the individuated partitions and associated behavioural profiles at $t4$. The big panels report the same UMAP reduction presented in the last column of Figure 5.9. Each dot is the representation associated with a particular individual and is colour coded based on the partition to which it belongs. Small panels represent the temporal evolution of the considered behavioural metrics for each individuated partition. The panel showing N°Sessions only reports the prediction produced by the model as the number of preceding session is constant for all the partitions. The x axis reports the game sessions while the y axis the value assumed by the considered metric at a specific point in time. The y axis is expressed in terms of number of standard deviations from the game population mean (i.e. z-scores). Each line indicates the mean z-score while the shaded area around the line its 95% confidence interval. The solid part of each line indicates the portion of the temporal series observed by the model (i.e. the input) while the dotted part the predictions produced at that point in time. The first columns shows the partitions associated with the representation extracted by the RNN architecture while the second those associated with the representation extracted by the version of the RNN architecture that included environmental and game events covariates. Each row indicates the partitions identified for the six considered game contexts.

Partition 1 represents individuals producing high intensity interactions (see Session Time) at a high frequency (see Absence). The high amount of Active Time highlights how the individuals were actively interacting with the game object. The individuals in this partition are projected to produce a number of future interactions that is below average while maintaining a high intensity profile. It can be speculated that the history of high intensity interactions reflected a positive propensity towards the game. This might have prompted individuals in this partition to consume most of the available contents

in the game leading to a reduced amount of expected future interactions (i.e. N° Sessions).

Partition 2 describes individuals that have a history of very infrequent (see Absence) and brief interactions with burst of activities and long idle times (they have the lowest Active time among the individuate partitions). These individuals are expected to maintain this trend in the future although producing a number of interactions that is largely above the average. An hypothetical explanation might see individuals in this partitions constituting a variant of those in Partition 1. The high frequency and intensity of interactions could suggest an eagerness to interact with the game object. This, combined with the low amount of consumed content (see Session Time and Active Time) could explain the projected high amount of future interactions.

Partition 3 includes individuals whose interactions have been very frequent and average both in terms of length and amount of activity until session 3. From there, a burst in both the length and active time can be observed concomitant with a reduction in latency before the following interaction. This is followed by a re-bounce effect with a marked reduction in the length and intensity of the next interaction. These individuals are predicted to produce a number of future interactions slightly above average while also maintaining a low intensity profile. These individuals might have started with a normal propensity towards the game which suddenly increased around session 3 and had a "physiological" downturn around session 4.

Partition 4 contains individuals producing the least intense and frequent interactions. With the only exception of a brief increase in active time around session 4. These individuals are estimated to produce a number of future interactions just above average while maintaining the original low intensity profile. These individuals started and maintained a low intensity profile, suggesting a negative propensity toward the game.

It is interesting to note that despite the fact that the partitions identified for the various game objects always show the two "macro groups" mentioned above, their finer grain characterizations (i.e. the behavioural profiles extracted from the partitions) vary between game contexts suggesting the presence of a possible interaction (see Appendix D.4).

Although the individuated profiles can provide valuable information on the behavioural "fingerprint" of group of individuals, the most relevant and reliable information can be found in the relationship between the considered metrics. We observe that Session Time and Session Activity are usually highly correlated. Low Absence seems to be a good indicator of the propensity to produce more interactions in the future. Similarly, high Absence seems to be associated with a general history of low intensity interactions. It is also worth noting that variations in this metric seem to follow and be proportional to increases and decreases in interactions' intensity.

We can also notice how there is often (but not always) a very high correspondence between the profiles associated with the representations generated by the two versions of the RNN architecture.

However, looking at the average silhouette score values, it seems that including environmental and game event covariates helps to generate tighter and more consistent partitions. This might have been driven by mechanism similar to one that we find in conventional linear models: by including appropriate covariates it is possible to obtain more reliable and robust estimates of a specific coefficient of interest [213].

5.4.2 Partitioning the representation associated to the environmental covariates

Looking at Figure 5.17 we can see the partitions individuated by the Mini-Batch K-Means for the representation generated from the environmental metrics. Given that all the entries included in our dataset were logged using the same time zone (i.e. CEST) we decided to focus in this analysis only on individuals playing from Central European countries. By looking at Figure 5.17A we can observe how individuals tend to distribute their interactions with a game object differently depending on the day of the week or the hour of the day. This finding is not surprising but is in line with the idea mentioned in sections 2.5 and 3.3.5: the environment in which an interaction (here, between an individual and a game object) occurs help to determine its observed behavioural intensity.

Looking at Figure 5.17A we can observe that the amount of playing activity in the considered sample tends to grow monotonically (but not linearly) from early in the morning (roughly around 6 a.m.) until early in the evening, peaking at around 7 p.m. A finding that is compatible with the expected daily schedule and circadian rhythm of individuals in the considered regions [214] and with the results obtained by Vihanga and colleagues [111]. A similar pattern can be observed for the distribution of playing activity during the days of the week: starting from Monday, individuals seem to progressively initiate more game sessions over time with peak activity recorded between Thursday and Friday.

Interestingly, it appears that the nature of the object (i.e. the game context) plays a role in determining how different environmental factors influence the intensity of the interactions. A finding that is again in line with the work of Vihanga and colleagues [111, 215]. Looking at the Figures in Appendix D.5, we can see for example that the game object *hmg* shows a different distribution of play sessions during the hours of the day with respect to $outriders$. For *hmg* sessions seem to be roughly equally distributed during the day while for $outriders$ they peak towards the end of the day. This might be explained by the fact that being *hmg* a mobile game, it allows individuals to more easily initiate playing sessions at any moment during the day. In other words, the considered environmental factors would be posing less constraints on the intention to initiate the gaming activity. This type of differences appear even more pronounced and pervasive when looking at the distribution of playing sessions over the days of the week.

More nuanced differences appear to emerge when inspecting the profiles derived by the partition analysis. Looking at Figure 5.17B we can see how different groups of individuals distribute their playing activity differently in a way that is not necessarily compliant with what emerged from Figure 5.17A. As in the case of the behavioural profiles we suggest that these findings should be interpreted with caution and consider them as descriptive rather than prescriptive.

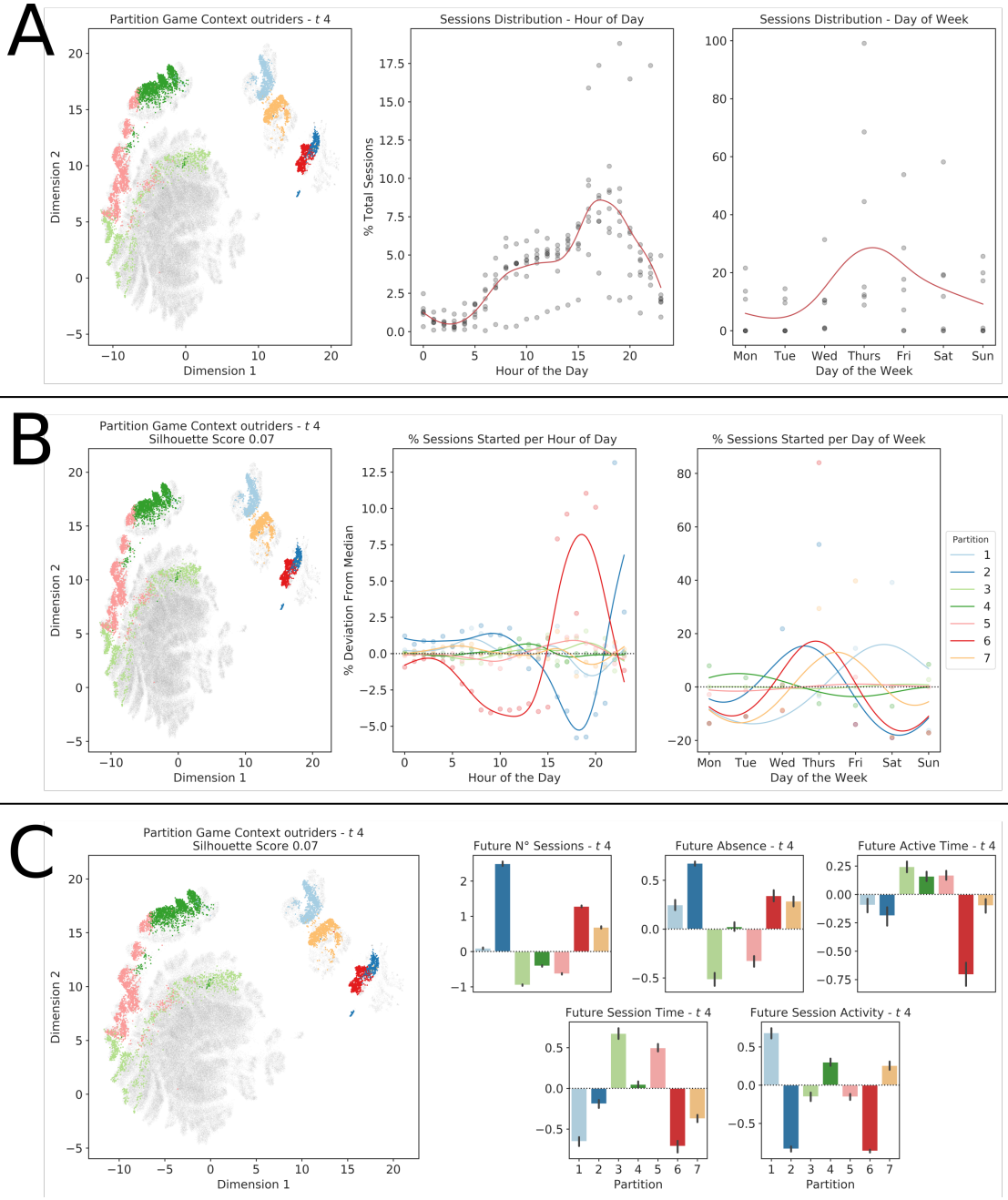


Figure 5.17: The panels show the individuated partitions and associated profiles for the representation encapsulating all the environmental information up to $t4$. Each panel shows the conventional UMAP reduction of the inferred latent representation with the colours representing membership to a specific partition individuated by the Mini-batch KMeans. The other two plots in panels A and B represent the percentage of total sessions that each partition initiated during a specific hour of the day or day of the week. The x axis reports either the hour of the day or the day of the week while the y axis the share of initiated sessions. For each panel we reported the line of best fit provided by a generalized additive model [208]. Panel A shows how each partition distributes its game sessions between different parts of the days or days of the week while panel B provides the same information at the sample level integrating over all partitions. The y axis in panel A is expressed in number of standard deviations from the sample mean. Panel C shows, similarly to Figure 5.16 , differences in the predictions produced by the model for each partitions. The x axis indicates the partition number while the y axis the expected prediction made for a specific partition. The y axis is expressed in terms of number of standard deviations from the sample mean (i.e. z-scores) with black line indicating the 95% confidence interval

Partitions 3, 4 and 5 represent individuals with a distribution pattern which is not radically different from what observed in Figure 5.17A. Partitions 3 and 5 show a slight preference for evening rather

than morning interactions while the opposite is true for partition 4 which seems to also have initiated more play sessions at the beginning of the week. They are expected to have a number of future playing sessions below average, but their next interaction is expected to be of moderately high intensity and to occur in a short amount of time. These individuals might have had a regular schedule of relatively long playing sessions that allowed them to consume a considerable amount of in-game content.

Partitions 1 and 7 include individuals showing a slight preference for early morning rather than evening interactions. Partition 7 appears to have initiated more sessions late in the week (e.g. Thursday and Friday) while partition 1 seems to have favoured the weekend. The profile emerging from the expected intensity of their next interaction appear to be similar for both partitions: a brief but intense session occurring after a relatively long hiatus. These individuals might have had only a relatively narrow window of time for accumulating their playing activity. Nevertheless they are expected to keep interacting with the game object (see the expected Future N° Sessions) suggesting that, despite the environmental constraints, they might be attributing high value to the playing activity.

Partitions 2 and 6 are the ones showing the most unusual and extreme patterns. Individuals belonging to partition 2 appear to have initiated most of their session very late at night or very early in the morning avoiding the afternoon and evening. Partition 6 shows the opposite patterns, with playing sessions happening exclusively between late afternoon and early in the evening. Both profiles appears to have logged most of their activity in the middle of the week at the expenses of the weekend. The next session for these individuals is expected to be brief, of low intensity and occurring after a relatively prolonged period of time. However, these partitions encompass individuals that are expected to have the highest number of future playing sessions. This suggest that, similarly to what we observed for partitions 1 and 7, these individuals might have enjoyed the playing activity but the manifestation of their playing behaviour has been hampered by environmental constraints.

The results of this partition analysis, in conjunction with what emerged in sections 4.4.4 and 5.3.2 seems to support our intuition regarding the role of environmental covariates in the representation generated by the RNN architecture.

These type of factors might influence the intensity of future interactions that an individual has with a particular game object. However they act mostly as facilitators or impediments to the observed behaviour (see partitions 2 and 6) rather than inherently influencing the internal state of the individual.

Indeed, in contrast to what we observe in Figures 5.13 and 5.11, the representation extracted by the environmental encoder appears unable to clearly distinguish between individuals based on the expected intensity of their future interactions. Interestingly, the results of the partition analysis show that the representation is not just noise but can be used for discerning different patterns of interactions. This suggest that despite the fact that the model was able to organize information to capture similarities between interaction patterns, this might not have been crucial for the minimization of the model

objective, or at least not in the same way as behavioural and game events covariates showed to be.

The only notable exception to this seems to be when environmental covariates describe unusual and exceptional situations (e.g., partitions 2 and 6). The reason might be that in these cases there is a correspondence between the environmental conditions in which an interaction occurred and the expected intensity of the next one. For example, speculating that individuals in partition 2 might be doing some form of shift work that allows them to play only late at night or early in the morning, it is reasonable to expect their future interactions to be of low intensity. This is because there is a consistent constrain on how much time they can dedicate to the playing activity, regardless of how enjoyable they might find it.

5.4.3 Partitioning the representation associated to the game events covariates

Figure 5.18 reports the profiles individuated by partitioning the representation derived from the game events metrics for the game context "Outriders". By looking at Figure 5.18A we can see that partitioning the representation we were able to distinguish between individuals based on the dynamics of their interactions with different in-game mechanics. Apart from partition two, which appeared to have a consistent pattern of engagement (i.e., all the available mechanics were triggered in a similar way over time) the other profiles were able to capture temporal variations in the preference of individuals for different in-game mechanics.

Interestingly, the various partitions appear to encode differences in the amount and intensity of expected future behaviour. This suggests that the inferred representation might have grouped individuals based not just on the similarity between the sequences of triggered events but also based on the impact that these might have had on the generation of future behaviour.

5.5 Summary

As mentioned in section 2.2.2, incentive salience attribution produces latent representations of objects that, when imbued with value, make future interactions with those objects more likely and intense [31, 32].

The fact that the representation generated by our model could be effectively described by a relatively small number of dimensions (potentially two) appeared to be in line with this and compatible with the idea presented in section 2.2 that motivation-related internal states can be compared to the magnitude of a two-dimensional vector. This metaphor of the "motivational vector" seems to be compatible with the global-local organization of the representation generated by the RNN architecture and its improved version.

At the global level, different game "objects" were organized in distinct and coherent regions (see Figure 5.8A) showing how the model attempted to operate on a meta-level by partitioning a global

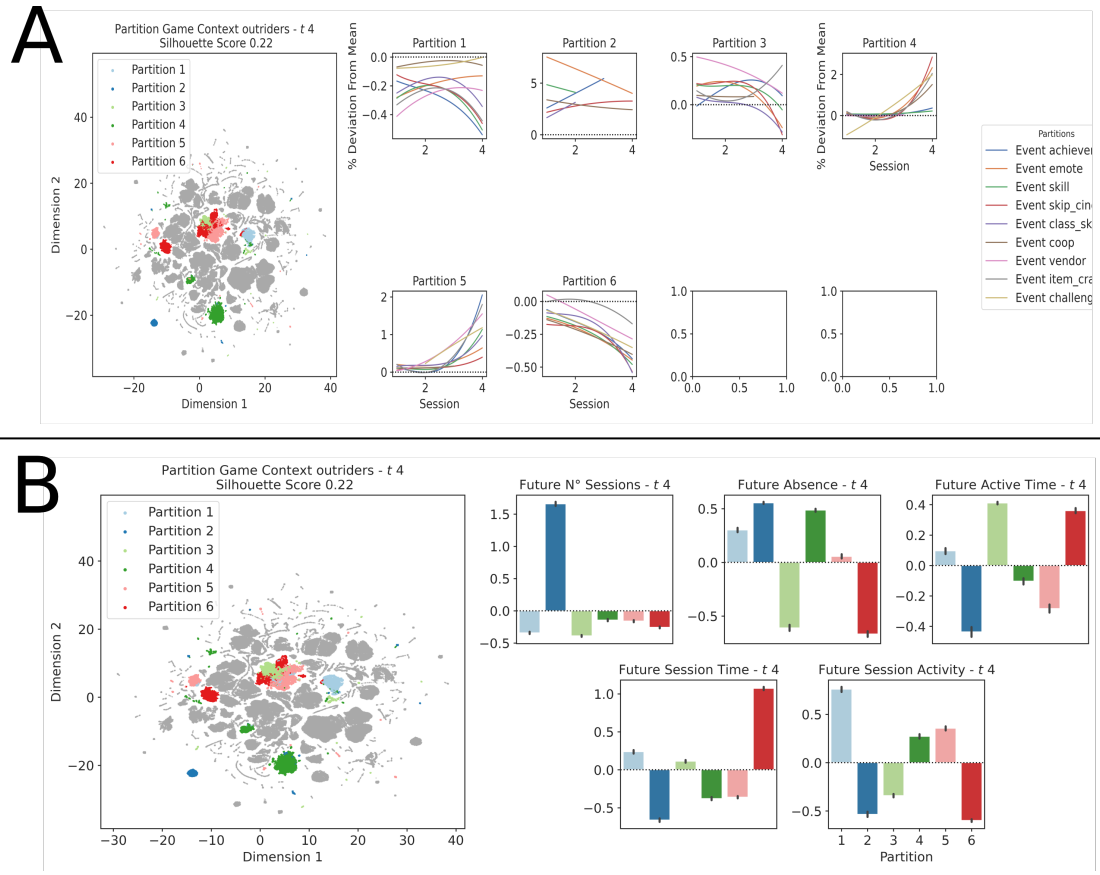


Figure 5.18: Panels show the individuated partitions and associated profiles for the representation encapsulating all the game events information up to *t*4. Each panel reports the conventional UMAP reduction of the inferred latent representation with the colours representing membership to a specific partition individuated by the Mini-batch KMeans. Panel A shows for each partition the share of events triggered during each considered game session expressed as number of standard deviations from the sample mean (i.e. z-scores). This information is conveyed through the line of best fit provided by a generalized additive model [208]. Panel B shows, similarly to Figure 5.16 , differences in the predictions produced by the model for each partitions. The x axis indicates the partition number while the y axis the expected prediction made for a specific partition. The y axis is expressed in terms of number of standard deviations from the sample mean (i.e. z-scores) with black line indicating the 95% confidence interval.

representation in several object-specific ones. This finding aligns with what highlighted in various work on neural manifold where the responses related to qualitatively different stimuli tends to show a cluster-like organization when reduced to a lower dimensional space [7, 11, 12].

At the local level, each object-specific representation showed an internal gradient-like organization distinguishing individuals based on the estimated intensity of their future interactions with that specific object. This was true for each of the considered behavioural targets (see Figure 5.8A) showing how the model attempted to provide an holistic description of the intensity of future interactions. The presence of this type of gradient-like organization emerged in work by Nieh et al. [9] when analyzing neural responses during an evidence accumulation task in virtual reality. When reducing the neural activity to a three-dimensional space, the resulting manifold presented a clear gradient able to code simultaneously for position and levels of accumulated evidence [9]. A similar finding was present in the work by Stopfner et al. [12] where the manifold structure extracted from the activity of olfactory neurons was able to represent qualitative and quantitative differences between odours through a global-local organization similar to that showed in section 5.3.1.

The dynamic nature of the representation generated by our approach also fits nicely with that of attributed incentive salience [14, 38, 51–53]. In particular, the fact that the aforementioned global-local organization is maintained over time (see Figure 5.9A) corroborate the hypothesis that our model approximated state changes originated from a dynamic process. In support of this, we also observed that the representation generated by our model was spatially coherent over time: it produced distinct regions of low and high expected intensity between which individuals moved over time (see Figure5.9D). These results appear to match the definition of motivation and incentive salience attribution specified in section 2.2: a single overarching process able to dynamically predict the likelyhood and intensity by which individuals will interact with a varied set of objects [14, 32, 33, 38].

Many other cognitive and affective functions might rely on a latent representation that is functionally similar to the one described in our work (e.g. credit assignment and optimal control [26, 138, 216], cognitive control, learning [159],various forms of reward processing [34, 36]). Similar to attributed incentive salience, these functions are all involved in generating motivated behaviour and heavily rely on reward signals, however none of them is concerned with attributing and describing the motivational saliency that an object possess. This is made evident in the works by McClure et al. [13] and Zhang et al. [14] where the system involved in salience attribution is functionally separate from the one assigning credit and executing actions: the former provide a representation that informs and biases the decisions taken by the latter serving an almost exclusively qualifying role (see the role of attributed incentive salience in addiction-like conditions [51]). Similarly, the representation generated by our model does not provide any insight on the decision making process underlying the observed playing behaviour but simply provide an approximate description of the “motivational pull” that a particular game object has on a particular individual at a certain point in time.

The functions encoded by the hidden units constituting the representation appeared to have a series of

distinctive properties, namely: redundancy, non linearity, multiplicity (single units code for multiple functions) and consistency over time. These may have played a role in providing the representation generated by our model with its distinctive characteristics. For example, as we mentioned in section 3.2 redundancy and inter-correlation are characteristics of the signals from which the manifold representation of internal states arises [7, 10]. Multiplicity on the other hand, might be the factor underlying the ability of our model to produce a single unitary representation which holds predictive power over different behavioural targets. Finally, consistency over time could be the mechanisms supporting the type of temporal coherence observed in panel 5.9D. We want to stress that these findings are to be considered exploratory in nature since they do not rely on *a-priori* hypotheses. A comparison between these computational properties and those underlying the attribution of incentive salience is required and would constitute a potential venue for future investigations.

The introduction of environmental and game event covariates appeared to have produced a more consistent representation (see Figures 5.14 5.15) with better discriminatory powers, a finding in line with the, although marginal, improvements in predictive performance observed in section 4.4.4. It might be speculated that this improvement was associated both with the introduction of new informative covariates and with the ability of the architecture to separately model their contribution. Indeed, similarly to the idea underlying Neural GAMs [179], different portion of the architecture might have focused on inferring specialized functions then combined in a more effective latent representation.

However, the major advantage provided by separately modelling the contribution of behavioural, environmental and game-events metrics lied in the ability to obtain distinct representations that we were able to leverage for exploratory analyses. This allowed us to perform time series partitioning at a much larger scale that has been previously done in the videogames literature (potentially up to millions of time series) [111, 131–133]. Indeed, previous approaches in the literature attempted to cluster or partition directly the observable data space, a particularly cumbersome process for univariate and multivariate time series, especially if suitable techniques like Dynamic Time Warping [217] are to be used. In our case by leveraging the representational power of ANNs we were able to condense the information present in the the time series data to a more compact representation, this allowed us to then rely on relatively inexpensive algorithm (i.e. Mini-Batch KMeans) for performing the partitioning.

The partition analysis conducted on the behavioural representation allowed to individuate a set of profiles that are, to a certain extent, consistent with behavioural correlates of different levels of attributed incentive salience [32] and general operant conditioning principles [44, 158, 190]: the strength of past interactions positively correlate with the frequency and intensity of future ones. The various offsets that each partition showed might suggest different levels of predisposition towards the various game-objects. The dynamic nature of these profiles provided a more granular characterization allowing to observe variations in the entire history of interactions and not just in the expected intensity of future ones. For example, it was possible to see how a higher likelihood of future interactions was

supported both by a history of low intensity but high frequency interactions as well as by a series of high frequency and high intensity interactions. In this sense, these behavioural profiles can be seen as useful devices for investigating the existence of inter-individual differences in schedules of interactions with potentially rewarding objects.

On the other hand, partitioning the environmental representation provided us with insights on how different individuals distributed their interactions with a videogame over days of the week and hours of the day. We were able to replicate some of the findings of Vihanga and colleagues [111, 215], identifying similar profiles of interactions. However, our approach allowed us to gather insights not just on the characteristics of the different profiles but also on their potential impact on interaction intensity. A similar conclusion can be drawn from the partitioning of the game events representation. We were able to identify not just how different groups of individual distributed their playing activity between different in-game events but also how these evolved over time. Moreover in contrast to the work of Makarovych et al. [132], our approach did not just identify frequent patterns of interactions with in-games elements but also the associated variations in the intensity of future behaviour.

Chapter 6

Model Application and System Design

6.1 Introduction

As we mentioned in chapter 1, despite the main aim of this thesis has been to derive a methodology for approximating the motivational state of individuals while interacting with a potentially rewarding object (a videogame in this case), a secondary objective was to illustrate the potential application of this methodology within an industrial setting.

In this view, this chapter will focus on sketching the design of a system relying on our methodology for automated engagement prediction and quantification. First, we will provide an overview on why a company might need a process for quantifying and predicting engagement and which characteristics this process should have. Then, we will proceed at illustrating a system designed for serving this need, placing particular emphasis on how its components connect with the work presented so far. Finally, we will introduce a set of ethical considerations that should be taken into account when designing such system.

6.2 Automated Engagement Quantification and Prediction

In an industry setting the development of research projects often aims at the the resolution of specific problems or at the improvement processes central to success (be it measure in terms of revenues or perceived quality of goods and services).

So where does engagement quantification and prediction sits within the needs of the videogame industry? Very often (if not always) the success of a videogame title is strictly connected with either its ability to retain users or with the experience that users had with the product (i.e., a videogame title) [218,219]. The first is pivotal in scenarios where games are treated as a service sold to an audience (similarly to the function of streaming services) while the second is more relevant in situations where games are to be considered digital goods [218, 219].

In this context, engagement can be viewed as a measure of how a particular game was, is or will be able to retain users. For example, if an individual is engaged with a particular service (e.g., a videogame) it is likely that will keep paying a subscription (or any other form of pay-to-consume) for said service. Similarly, if an individual had a good experience with a particular digital good, it is more likely that they will promote it to other potential users, acquire similar products or buy products from the same seller.

In this view being able to predict the propensity that a user (or a group of users) has towards a particular game translates (in a more or less direct way) to the capacity of assessing if a game is likely to be a success of public and revenue. For this reason it is often the case that videogame publishers and studios try to leverage the information they have available through telemetry systems for taking the stock of how a particular game is performing [24].

This is a classical example of analytical reports summarizing various type of Key Performance Indicators (KPI) [24] or profiling techniques describing how users interacted with a particular game [24]. Despite this approaches are very relevant for gathering insights on the performance of a particular game title, they only allow to execute what we call "reactive" interventions. By reactive interventions we mean that mitigating actions for improving a sub-optimal situation (e.g., when a videogame is unable to foster engagement) can only be taken *ex-post*.

On the other hand, interventions based on the outcomes of a predictive model are by definition "pro-active". This is because knowing in advance if a particular situation is going to be problematic or not allows one to plan and deliver mitigating actions *ex-ante* [24, 104]. It is worth noticing that approaches based on the outcome of predictive models are not incompatible with techniques used within a "reactive" framework (e.g., reports and profiles) but rather complementary. For example, the same KPI calculated on observed data can and should be computed also on predictions and forecasts generated by a machine-learned model. Similarly, it is possible to create profiles that not just describe the historical interactions of users with a particular game but that are also informative of the expected future engagement of such users. Within this last "pro-active" framework lies the work that we presented in this thesis.

6.3 System prototype Design

In this section we will proceed at illustrating the design of a system prototype aimed at delivering predictions and insights that can foster *ex-ante* interventions for the mitigation and improvement of videogames engagement.

We will pose particular attention on the fact that our proposed system is designed to rely on a "global model" [175] suitable for application in multi-contexts scenarios. Examples of such scenarios includes situations in which a single large entity (e.g., a publisher) manages multiple videogame titles or when multiple small players (e.g., independent studios) decide to form a consortium. We will also illustrate how our system encompasses not only the three tasks mentioned so far, namely: prediction,

reporting and profiling, but also allow to generate representations that can be used as inputs to other machine-learning systems.

A diagram illustrating the proposed design for our system prototype is presented in Figure 6.1:

It is worth noticing that we will focus on illustrating the various components presented in Figure 6.1 but we will not describe the type of interventions that can be taken on the basis of the system's outputs. Indeed this, other than being an articulated and complex subject that in itself would require an additional research project, is outside the scope of the current work. The responsibility and knowledge required for designing, executing and evaluating interventions ultimately sits within those entities consuming the output the model.

6.3.1 Data Generation

This is the first component of the system and describes the entities generating the data that will then be used by the system. It is composed by two major elements: the users and the game contexts. By users we mean the pool of individuals that already had an interaction with one or more of the considered game contexts. With game contexts we extend the definition of game object presented in chapter 3 in order to include not just the game world but also the elements necessary for making the interface with the users possible (i.e., software and the hardware components). In the interest of simplicity and for better connecting this section to the work described in chapters 2 and 3, we invite to consider user and game context as aliases for individuals and game objects.

In line with the framework that we adopted through out this thesis we assume that each game context possesses properties (defined by their structural characteristics) that might result to be more or less rewarding to different users. By means of repeated interactions with the game contexts the users learn about these properties and progressively updates latent representations of the various contexts. These representations, which as we said in chapters 2 and 3 are imbued with value, act by either promoting or demoting future engagement with the game contexts (net of the contribution of the surrounding environment). The amount of engagement can be measured, form a behavioural point of view, by means of metrics indicative of the frequency and intensity of the playing behaviour.

The various game contexts can be managed by a single or multiple entities (e.g., publishers, studios or public organisations) and can be provided through different type of hardware. For example, the game contexts utilized for this thesis were managed by a single publisher (i.e., our partner company Square Enix Ltd.) and provided through an array of different hardware systems (e.g., smartphones, tablets, personal computers and gaming consoles). Ultimately, the software and console hardware not only generate the game object with which the users interact but also support the telemetry system in charge of recording, processing and transmitting metrics (behavioural and not) to the relevant data storage systems.

6.3.2 Model Owner

This component of the system represents the entity responsible for the acquisition and storage of telemetry data generated by the interactions between the users and the game contexts. It is also in charge of managing all the operations necessary for fitting a learning algorithm to the data and validating the derived model.

It is relevant to highlight that the model owner usually corresponds one-to-one with the entity managing the game contexts (or components of the game contexts) but the two do not necessarily have to coincide. In the context of federated learning [220] for instance, a model owner might distribute copies of the same learning algorithm across separate entities and only act as a pooling mechanism once they have been fitted to the data [221]. In this case, the entities do not necessarily need to be known to the model owner or to each other, in fact this information must be kept hidden. In this view, federated learning allows to generate robust global models (like the architecture presented in this thesis) by fitting a learning algorithm on heterogeneous data distributed across a large number of independent actors [220, 221]. Moreover, since the aim of federated learning is to generate such models while being compliant with strict privacy constrains [220, 221], this opens to the possibility of collaborations between competitors (e.g., consortium of small private companies) or neutral entities subjected to strict data sharing limitations (e.g., research institutions).

That said, for simplicity in this section we will focus on a situation where the model owner coincides with the entity managing the game contexts.

6.3.2.1 Data Storage

Once the in-game behaviour resulting from the interaction between the users and the game contexts have been recorded by the telemetry system the information need to be transferred to a suitable storage system. Depending on the needs and technical capacity of the storage owner this can happen either via data streams or batch processing [24]. The component in charge of storing the data should also take care of conducting suitable Extract, Transform and Load (ETL) operations in order to move from the raw format in which the telemetries are collected to a more suitable one employable by the learning algorithm [24]. The ETL operations should also take care of all those data validation, sanity checks and pre-processing steps that are not directly relevant to the generation of the machine-learned model. This would correspond to the various filters and data munging operations that we mentioned in sections 4.2.2, 4.3.2 and 4.4.2.

6.3.2.2 Model Generation

Once the raw information provided by the telemetry system has been processed, cleaned, organised and stored in a suitable format a learning algorithm can be fitted to the data. In this case we assume the learning algorithm to be one of the three ANN architectures presented in chapter 4 and hence relying on the same operations presented in sections 4.2.3, 4.3.3 and 4.4.3 for tuning, fitting and validating.

Data Generators the first step in the model generation require the construction of appropriate mechanisms for feeding the data to the learning algorithm. Different generators will need to be created for the various steps of the process: one for tuning, one for training (i.e., fitting) and one for validating the generated model. It is worth noticing that the data generators do not only need to provide batches of the data to learning algorithm but are also in charge of:

- Applying the appropriated transformations (e.g., input scaling) in such a way that no information leaking is occurring.
- Individuate and satisfy relevant stratification needs required for having a representative sample of the target population.
- Adopt adequate randomisation and splitting strategies to avoid introducing biases during the model fitting.

Model Configurations simultaneous to the construction of the data generators is the creation of multiple architectural configurations for the learning algorithm. This step assume the use of a tuning algorithm similar to HyperBand [192] for the search of the optimal hyperparameters. Indeed, as we specified in chapter 4 HyperBand works by iteratively training a large pool of models instantiated with a random selection of hyperparameters and progressively pruning those showing poor out-of-sample perfromance. At this stage the only thing that needs to be specified are the hyperparameters to optimize and the boundaries of the search space. Routinely validation checks should also be included in this stage for removing from the pool of potential configurations those that resulted to be corrupted. Indeed certain hyperparameters configurations might make training or even model instantiation infeasible.

Model Tuning once the model configurations have been generated it is possible to search for the best performing hyperparameters. Despite in this work we leveraged the HyperBand algorithm [192], other type of tuning algorithm can be taken into consideration. The choice of which tuning algorithm to chose should be informed by the problems at hand along with eventual time and resources constrains. For example, methods based on surrogate models like Gaussian Processes [222] are effective but expensive and complex approaches. On the other hand methods relying on naive random search despite their lack of efficiency are cheap and straightforward to implement. It is worth noticing that at this stage not just the hyperparameters relative to the architecture can be optimized but also those associated with other components used during the training stage (e.g., the optimizer learning rate). The amount of data assigned to the data generator used at this stage should strike a balance between efficacy and waste. If too little data are used, the hyperparameters search might produce sub-optimal results. However, going to the other extreme might hurt the training stage as the data used for searching the best model configuration cannot be re-used. Another piece of information that needs to be specified is the amount of computational budget to be given to the tuning algorithm a choice that again be informed by the available resources and time constrains.

Model Training at this stage the best model configuration should be fitted to the bulk of the available data. It is worth noticing that depending on the size of the training data this stage might be very resource intensive. However, model fitting should not necessarily always be carried out on the entire amount of data. For example, if the algorithm has recently been through a full training stage (e.g., leveraging all the available historical data) an update based on the latest available data might be sufficient. Moreover, if significant resource constraints are present, it might be possible to fit the learning algorithm using a suitably defined random sample of the original data. This of course depends on the number of parameters in the model, the complexity of the learning task and the heterogeneity and size of the available data. Fitting the model to the data should be carried out until convergence or a suitably defined level of accuracy is achieved, again the choice of such level might be dictated by time or resource constraints. In this regard, convergence should always be assessed and monitored through out-of-sample metrics with the option of triggering an early stopping policy if improvements are not observed for a long period of time (see details specified in sections 4.2.3, 4.3.3, 4.4.3). What is important at this stage is to create checkpoints of the model state whenever out-of-sample accuracy reliably improves and to keep detailed logs of the fitting procedure. The first avoids wasting efforts if the optimisation process suddenly diverges or come to an abrupt end (e.g., due to technical issues), the second allows keeping track of relevant metrics useful for assessing if and how convergence is reached.

Model Validation once the learning algorithm has been fitted on the data it is necessary to validate it in order to verify its usability in a production environment. The validation process can be extremely articulated and in large part depends on the needs of the model owner, however, two main aspects should certainly be validated. First, a series of tests should be conducted in order to verify that basic functionalities are not disrupted, in our case for example we might want to make sure that the model is able to produce predictions and representations, that these are within critical boundaries (e.g., no extremely large or small values are present) and show particular properties (e.g., statistical or distributional ones). Second, the convergence and accuracy of the model should be assessed and compared against a critical lower bound or the performance of a naive approach (we provided some examples of these in sections 4.2.1, 4.3.1.0.1 and 4.4.1). Convergence can be assessed by running automated tests on the logs created during Model Training, checking for example the difference between in-sample and out-of-sample performance, variations in the $l2$ norm of the error gradient or in the values of the parameters [1]. Accuracy can be evaluated by assessing model performance on a test set and then comparing it against the aforementioned benchmarks. In this case, statistical approaches based on inference can be used for evaluating if the expected gains provided by the model lie outside of a Region Of Practical Equivalence (ROPE) and can justify the cost associated to deploy the model in production (an example of this can be found in Appendix C).

Model Serving once the model has been validated it is possible to deploy it within a production environment. The frequency at which the model is queried for obtaining predictions or representation

[1] We suggest to consult [28] for a more exhaustive list of checks

depends on the needs of the model consumer and again the available resources. For example, a demanding streaming approach might be needed for supporting live predictions or a system aiming to react to a rapid turnover of large volumes of users. On the other hand, a conventional scheduled job executing a prediction pipeline on a much more relaxed schedule (e.g., once a day or even once a week) might be more than sufficient for generating static KPI reports and updating user profiles. The model generation process should be repeated end-to-end on a regular basis depending on how much distributional drift is expected in the data. However, parts of it like Model Training, Validation and Serving might be executed independently on a much more frequent schedule.

6.3.3 Model Consumer

The model consumer identifies the entity interacting with the model and leveraging its outputs. As specified in section 6.3.2 this doesn't necessarily have to correspond to the model owner but it is supposed to be identified in at least one of the entities managing the game contexts. Again, for simplicity we will assume that in this case they all corresponds to the same entity. The model consumer shouldn't have direct access to the algorithm generated by the model owner nor should be able to alter its inner working (e.g., it should be able to re-fit the algorithm on a new set of the data), its only focus should be to consume its output for different type of applications. We will now proceed at briefly illustrating some examples of these applications.

6.3.3.1 Representation Sharing

As we described in chapters 3 and 5, the latent representation inferred by the different types of RNN architectures can be thought as an approximation of the saliency that an individual has attributed to the act of interacting with a specific a game context. We can think of it as compressed trace of the state of a user (or a group of users) indicative of the amount of their future engagement.

As we have seen in section 5.3.2, this representation can be constructed from different types of inputs, being them the intensity of historical interactions with the game context or the sequences of game mechanics with which a user interacted. In this view, it can be seen as a set of features extracted by the ANN architectures (see sections 3.3.3 and 3.3.4) which can then be made available to other learning algorithms in order to improve their perfromance.

For example, a representation generated from game events metric (see section 5.3.2) can be interpreted as the set of future answering the question: "which sequences of in-game mechanics are expected to produce higher or lower level of future engagement?". In this view, a recommender system might leverage such representation in order to produced personalized suggestions for contents that are more likely to foster engagement [223]. A prime example of this type of applications is the recent work by Yang et al. [224], were the authors proposed a recommender system base on Graph Neural Networks which included a sub-module explicitly designed for engagement modelling.

Similarly, systems aiming to compute lifetime value (LTV) indexes (which are indicators of the expected profitability of user within a pay-per-play business model) might benefit from a representation

informative of the amount of future gaming behaviour expected by a group of users. Indeed such systems rely on two major components for deriving the LTV index: the average expenditure from a group of users along with their expected amount of future engagement [225].

More generally, any automated system indirectly relying on information about the expected level of engagement could benefit from such representation. The reason for this is that by providing "pre-computed" features informative of the level of expected engagement it is possible to alleviate the computational burden of other learning algorithms (or components of a learning algorithm) which focus should be on solving a potentially different task. This becomes particularly relevant in contexts where the amount and variety of available data make it challenging to perform manual or algorithmic feature selection.

6.3.3.2 Profile Generation

As we have seen in sections 5.4.1, 5.4.2 and 5.4.3 it is possible to partition the various representations generated by the ANN architecture in order to obtain corresponding behavioural profiles. These profiles distinguish themselves from those generated in previous work in the literature [5,103,130,132] in two fundamental ways:

1. They are, by design, able to represent behavioural dynamics (due the recurrent operations used for generating the representation space) within the full history of interactions. It has been common practice in the literature to apply clustering algorithm, sequentially, to time series data [5,97,226] with no guarantee that the temporal structure was preserved (a notable exception to this is the work by Makarovych et al. [132]).

2. They are engagement-specific (due to the objective function used for fitting the ANN architecture, more details can be found in section 3.3.4) and therefore offer a convenient framework for their interpretation. Other approaches in the literature preferred to adopt completely unsupervised approaches [101, 130] in order to derive more holistic but less specific profiles.

Interrogating the profiles generated from the representation extracted by our approach can provide insights not just on the most frequent sequences of events triggered within a game context, but also on their dynamics and associated level of expected engagement. This can be of use if the model consumer is interested in evaluating how users interacted with the various mechanics as the game progresses [97, 132] and if specific sequences of events are associated with variations in the expected amount of future engagement.

6.3.3.3 Live Predictions

The most straightforward application of our approach makes use of the predictions generated by the model for solving the conventional tasks of churn and survival time prediction (see chapters 2 and 4 for a more detailed description of these two tasks). These predictions are conventionally used as the building block for various type of automated decision-making systems such as in-app

notifications [114], delivery of in-game incentives or targeted marketing campaigns [24, 104].

The general principle behind these systems is relatively straightforward: the predictions produced by a model are used as an indication of the expected level of future engagement, if this level falls below or above a certain threshold a set of automated mitigating actions are triggered. For example, if a user is predicted to enter in a phase of dis-engagement (see the framework of O'Brien et al. presented in section 2.3.1.1) a set of small in-game incentives might be issued in the attempt to retain them. On the other hand, if a user is projected to produce a high amount of playing behaviour within a specific game context, a series of messages promoting novel in-game contents could be provided.

In this regard, the methodology presented in this thesis offer a series of technical advantages over previous approaches (see table 2.5 for an overview of the works on the subject). First, the prediction provided by our model are on a continuous scale allowing the model consumer to chose suitable thresholds for calibrating the triggering of the relevant mitigating actions. Second, instead of providing predictions for a single indicator (e.g., churn probability or survival time) our approach does it so for multiple behavioural metrics (i.e., the five targets presented in section 4.3.2 and 4.4.2) allowing for a more holistic characterisation of the level of expected engagement. Finally, since our model is able to provide credible intervals for its predictions (see section 4.2.4 for an example of that) it is possible for the model consumer to adopt a more cautious approach, or flat out reject predictions, in situations of high uncertainty.

6.3.3.4 Automated Reporting

As we mentioned in section 6.2, analytical reports covers an important role in the decision making processes taking place within the videogame industry [24]. The inspection of KPI reporting how many users are actively playing a game and how long and frequent are their interactions can provide a very effective description of how good or bad a game is performing. Moreover, the ability of stratify this indicators according to demographics of interest or ad-hoc business rules makes them extremely versatile tools for individuating potential areas of intervention [24].

That said, any action taken on the basis of such indicators is bounded to be "reactive" rather than "proactive". In order to circumvent this issue the same type of reports derived from historical data could be generated using the output of a predictive model. For example, it would be possible to monitor the number of users expected to have a marked decrease or increase in future engagement after a major change in the game has been released. Similarly, it would be possible to quantify the number of users projected to have a high or low engagement profile within specific demographic groups, sections of the game or regions of the world.

What is worth noticing is that all the applications examples illustrated so far can be derived from a single unified framework. This framework can simultaneously leverage a large amount data sources and serve multiple game contexts. This can in principle alleviate the costs that would be associated to the implementation, maintenance and deployment of multiple systems relying on different learning

algorithms (e.g., developing and maintaining a single cross-games pipeline instead generating single ones for each of the available game context). Moreover, fitting a single "global model" on a wide range of game contexts allow to produce representations and predictions that are simultaneously informed by all the available data therefore less prone to "silos effects" (e.g., lack of robustness and generalizability caused by the inability of different models to share information).

6.4 Some Ethical Consideration

Automated system leveraging behavioural data are now-days used extensively in both low (e.g., marketing) and high stakes (e.g., credit assessment) scenarios [227, 228], with the potential to have a direct and concrete impact on individuals [228]. For this reason, when designing automated data-driven applications, issues related to fairness should be taken into account.

By fairness we entail the set of principles and considerations that have been adopted recently to avoid that decisions based on a machine-learned model do not inadvertently bring harm to specific groups of people [227]. A complete overview of the issue of fairness in machine learning would be beyond the scope of not just this section but the entire thesis, as it is a vast landscape [227] hard to navigate due to its many levels of complexity [229]. We will therefore focus on three major aspects related to the work presented, in this chapter.

The first aspect concerns biases present in the data on which a machine learning algorithm is fitted. These might be induced by an over or under representation of certain strata of the population that an automated system will ultimately need to serve [227]. Given how machine learning algorithms are often fitted to the data (e.g., maximum likelihood) the risk is that their predictions will revert to the mean or in the worst case, will result to be biased with respect to the true characteristics of the population [227, 229]. Despite the harm that these biases might cause in the context of engagement prediction is not as pronounced as in other areas (e.g., criminal or medical risk assessment), they can still have unexpected repercussion on an individual if they assume that people engage in similar ways regardless of situational, personal and cultural differences. For example an individual might receive notifications in inappropriate contexts (e.g., during an emotionally challenging period) or be unfairly penalized within the game world (e.g., due to irregular playing patterns caused by personal or situational reasons) because they deviate from what is the expected behaviour in the data on which the algorithm was fitted.

To this connects the second aspect of fairness that we want to highlight, namely the risk of inadvertently cause harm to individuals which are either temporarily or structurally subject to some form of vulnerability. This might happen as a consequence of automated decision making based on what we call "unconstrained model predictions", what we mean by this is when the predictions from a model are used verbatim without knowing the context in which they will be applied. For example, if we imagine a system aimed at individuating high spending users within a game relying on gacha or loot-box mechanics, relying on unconstrained predictions might in this case inadvertently target

individuals with a predisposition to or an history of problematic gambling behaviour [230]. The most problematic aspect related to this issue is that often the information required for "constraining" an "unconstrained prediction" are not available to the system, either because they are not easy to derive (e.g., they are not directly observable) or should not be accessible (e.g., they are Personal Identifiable Information - PII [1]).

Related to the first two points is the third and final aspect of fairness that we would like to address which is connected with the right to object specified by the General Data Protection and Regulation act (GDPR) [1] which allows an individual to object to the processing of their data in any form and at any moment. Despite this aspect it is partially attenuated by rights granted by the GDPR (although this interests exclusively the European Union), it only covers the processing of data rather than the effect connected with automated decision making. An individual might object to the inclusion of their data during the decision making process but still be subject to the effect of this last one, for example, if the system entails a policy of blanket intervention based on the average user behaviour for all those individuals for which no data are available.

As we mentioned at the beginning of the paragraph, an exhaustive treatment of these issues and the relative mitigating actions that could be taken is beyond the scope of the current work. Nevertheless, we want to stress that addressing them in an effective manner is of pivotal importance when automated systems based on machine-learning (or other forms of statistical decision making) becomes part of the operations of a company.

6.5 Summary

In this chapter we proposed the design of a system aimed at leveraging the modelling approach described chapters 3 and 4 for automated engagement prediction and quantification. We first described why a company (or a consortium of companies) might be interested in such system and why it is important to rely predictive approaches for taking "proactive" actions towards the improvement of user engagement and retention. Subsequently we proceeded at describing the various components of the system trying to connect them with the work presented in the manuscript putting particular attention on providing examples of potential applications. Finally, we highlighted some ethical considerations that should be taken into account when developing and deploying such system.

It is important to note that this chapter is far from being an exhaustive treatment of the topic, we aimed mostly at illustrating how our approach could be translated in a system designed for satisfying specific needs of the videogame industry. Future work in this area should investigate and expand on a series of topic introduced in this chapter. For example, a more detailed description of the system and its sub-components is needed. This should pose particular attention on the resilience, reliability and efficiency of the system and on the challenges derived by deploying and maintaining it within a suitable production environment. The application examples reported in section 6.3.3 are presented in the form of Proof of Concept (POC) and therefore must be tested within concrete business cases. This

is an essential step required for gathering a better understanding of the type and magnitude of impacts that a system like the one we proposed might have. Finally, it would be ideal to illustrate which type of interventions could be carried out on the basis of the outputs of the system. The elaboration, execution and validation of such interventions however can only be the result of a collective endeavour where the designer of the system liaises and leverage the knowledge of various relevant figures within the business: engineers, designers, analysts as well as product and growth managers.

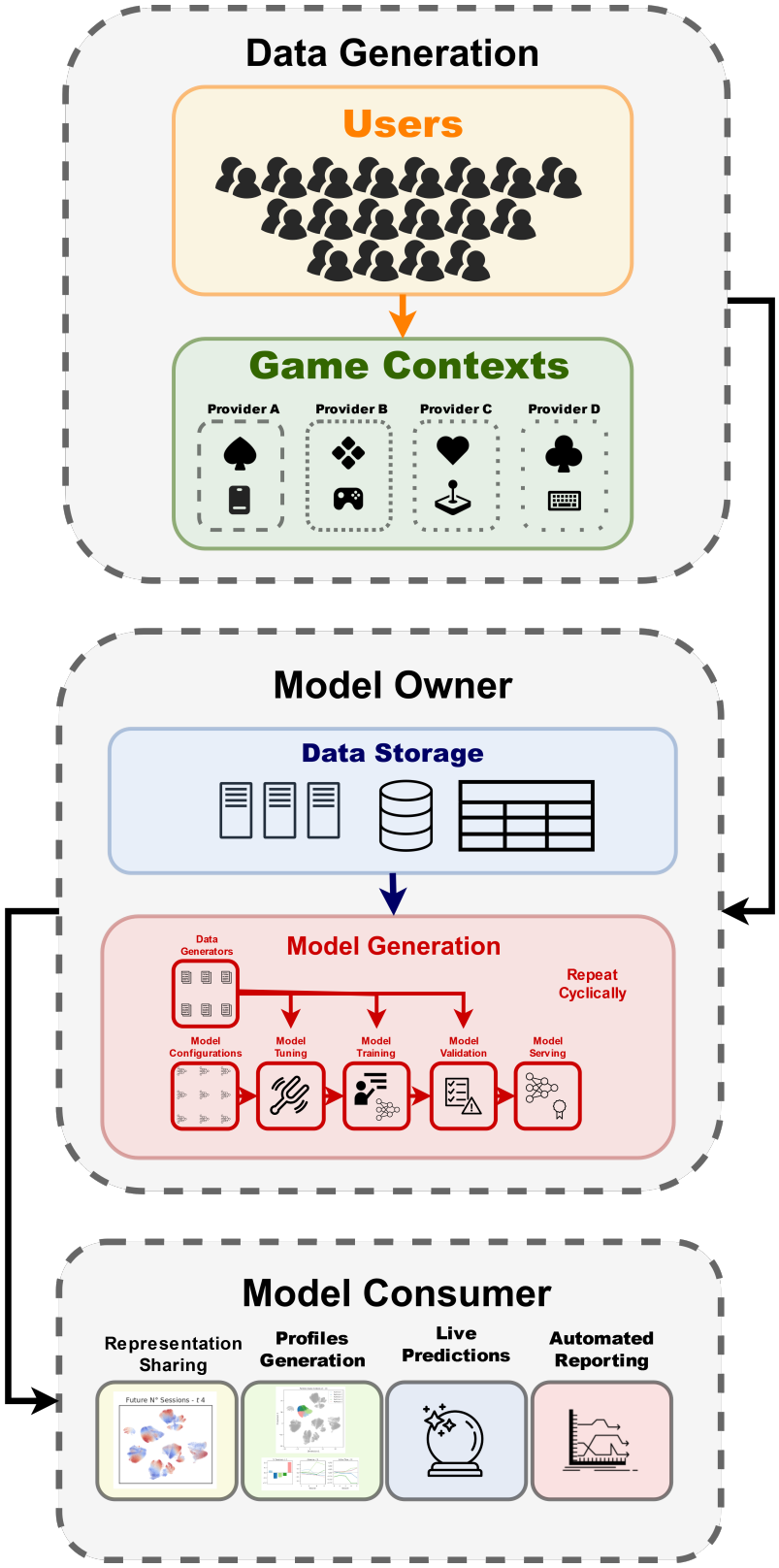


Figure 6.1: The figure represents a simplified system diagram for a potential application of the improved RNN architecture presented in section 4.4. Solid lines indicate low-level components in the system while dashed lines indicate high-level entities. Directional arrows represent the flow of operations inside the system.

Chapter 7

Conclusions

7.1 Contributions

We will now summarise the main contributions of the work presented so far, proceeding chapter by chapter.

7.1.1 Chapter 1 - Connecting Motivation and Engagement

In chapter 2 we focused on providing an overview of the psychological process of motivation highlighting its connection with the construct of engagement.

Relying on the framework proposed by Berridge et al. [31] we illustrated how motivation can be seen as a process generating latent representations of objects informative of their capacity to produce rewarding experiences. We also illustrated how these representations act as modulatory mechanisms promoting or discouraging future interactions with said objects [32].

Through a brief overview of the literature we showed how engagement with digital games can often be described in terms of the cognitive, emotional and behavioural manifestations resulting from the interactions that individuals have with particular game-object [59, 63, 68]. These manifestations arise from the interaction between the internal state of the individuals and the structural characteristics of the game [58, 59, 61–64]. Moreover, the surrounding environment also appears to have a role in this, acting as a modulatory factor [60, 61, 110, 111].

Despite attempts to connect these two processes [68, 74, 231], a clear mechanistic description of this relationship has been elusive. One major contribution of this chapter has been to outline a unifying theoretical framework for understanding the connection between neuroscientific theories of motivation and engagement. By modifying the Engagement Process Model of O'Brein et al. [61], we proposed to see motivation as a latent process arising from the interaction between individuals and (potentially rewarding) objects (i.e., videogames) and to consider engagement as its (noisy) manifestation in the

observed behavioural space.

Relying on this idea, we also highlighted the similarity between data-driven methodologies for engagement prediction in a videogame context and latent variable models used in the behavioural neuroscience literature. Despite the fact that these two approaches rely on radically different assumptions and mechanisms [27], they both assume the existence of an unobserved component in charge of input-output mapping.

We therefore proposed that if we consider the functional goals of motivation-related processes (e.g., the predictive capacity of incentive salience attribution [32]) it is possible to re-frame engagement prediction models as supervised variations of the unsupervised latent variables models often used in the neuroscientific literature [20, 22].

7.1.2 Chapter 2 - From Computational To Supervised Learning Models

In chapter 3 we attempted to translate computational models used for describing a specific type motivation-related latent process (i.e. attributed incentive salience) into a methodology for inferring the product of such such process from large volume of behavioural data. This was done by adapting a simulation study developed within a reinforcement learning framework into a supervised learning task.

The major contribution of this chapter has been to lay out a theoretical framework justifying the use of ANN for generating latent representations that approximate the functionalities of attributed incentive salience. In particular, based on existing computational models [13, 14] we illustrated why ANNs, and their recurrent variant in particular, are well suited for this task. We also proposed a series of architectural constrains that encourage an ANN to generate latent representations compliant with the functional characteristics of attributed incentive salience. These were:

1. The use of a global model architecture for obtaining a unified representation able to encompass multiple objects.
2. The use of multi-task learning in order to force such representation to be a good holistic descriptor of observed behaviour.
3. The use of GAM-like mechanisms in order to model the contribution of game-object characteristics (e.g., game mechanics) and environmental factors while maintaining separability and interpretability of the inferred representations.

7.1.3 Chapter 3 - Designing, Implementing and Testing the Models

In chapter 4 we designed, implemented and tested the ANN architecture presented in chapter 3.

From a theoretical point of view, the contribution of this chapter has been to validate some of the assumptions that were proposed in chapter 3. We showed the feasibility of using a single global model for generating latent representations able to predict the intensity of future interactions between

individuals and a variety of videogame "objects". We were able to confirm the important role of recurrency and non linearity for generating such representations, a finding that we replicated for all the three considered models. Finally, we showed that modelling the contribution of in-game structural characteristics and environmental factors increased the predictive power of the inferred representation, albeit only marginally.

From an application point of view, the ANN architectures presented in this chapter (i.e., the BM, RNN and improved RNN architectures) are the first attempts (to our knowledge) to test a "global model" on large amount of behavioural data generated from multiple videogame objects [115, 123–126]. Other works investigated the potential of such "global models", but they tended to focus on a small amount of game contexts [198] or leveraged data sources that were orders of magnitude smaller than ours [125, 198]. Despite this, we think that the amount of data on which a learning algorithm is fitted does not tell much about its effectiveness although it certainly plays an role in the assessment of its feasibility and robustness. The methodologies illustrated in this chapter also appeared to be the first to leverage multi-task learning for more than two targets [125, 198]in order to obtain a more holistic description of engagement from a behavioural point of view.

Finally, by using an iterative model-building approach we illustrated a methodology for dealing with complexity in a gradual but principled way. In particular, we showed how it is possible to use a pre-existing theoretical framework to inform the design and implementation of an ANN and understand how its components contribute to the improvement (or detriment) of predictive performance.

7.1.4 Chapter 4 - Validating the Properties of the Representations

In chapter 5 we inspected and validated, from a qualitative point of view, the representations generated by our methodology.

The contribution of this chapter was two-fold. First it illustrated how the representations generated by our approach showed some of the functional characteristics associate to motivational states (e.g., the level of attributed incentive salience in particular) presented in both chapter 2 and chapter 3. These were:

1. The ability to distinguish between different potentially rewarding objects (i.e., videogames).
2. The ability to differentiate between individuals based on the expected intensity of their future interactions with such objects.
3. The ability to show the two characteristics above consistently over time.
4. The ability to capture the dynamics underlying changes in the expected intensity of future interactions.

Second, it provided an overview of the potential behavioural correlates associated with the representation generated by our models. This allowed us to individuate various type of behavioural

"phenotypes" and to shed some light on their potential role in promoting or discouraging interactions with a rewarding object (i.e., a videogame).

From an application point of view, similarly to what we specified for chapter 4, we showed how it is possible to leverage the representation generated by a single "global model" to derive engagement profiles from multiple game contexts. We also showed how these profiles can be derived from a diverse number of input metrics in order to take into account the behavioural characteristics of the individuals, the environment that surround them and the characteristics of the game objects with which they are interacting. To our knowledge, this is the first attempt to condense in a single coherent modelling framework both the predictive and the profiling (i.e., quantification) aspects of engagement modelling (see the overview of the literature on the topic presented in Chapter 2)

7.1.5 Chapter 5 - Illustrating Potential Industrial Applications

In chapter 6 we reconciled the work presented in the thesis with the industrial setting in which it was developed.

The major contribution of this chapter has been to sketch the prototype of a system leveraging the modelling approach presented in this thesis for large-scale automated engagement prediction. In doing so we put particular focus on describing how the various characteristics of our methodology would fit within the different components of such system.

From an application point view, we showed how relying on a single "global model" allows the design of a coherent framework unifying all the components of the system, from data parsing and pre-processing to model generation and serving. Such "global model" can serve multiple game contexts (or multiple game providers) at the same time allowing to pool information across them. This gives access to a higher degree of effectiveness and generalizability while potentially decreasing maintenance and deployment costs. Finally we showed that relying on a methodology that focuses not only on prediction but also on representation learning allows the design of systems able to cover multiple functions: going from predictive analytics, to feature extraction and sharing.

7.2 Conclusion

We have outlined a methodology for approximating motivation-related latent states in situations where large volumes of behavioural data are available but no direct contact with the individuals that generated them is possible. Importantly, our approach attempts to respect both computational and theoretical constraints provided by previous works in the field of behavioural and affective neuroscience.

In doing this, we showed that it is possible to embed theory-driven knowledge in data-driven approaches, allowing us to more easily interpret and test hypotheses regarding individuals' latent states. In comparison to other works focusing on the identification of latent states (or their manifold representation) from behavioural data [18, 20–23], our methodology offered a series of advantages: It did not

require the Markov assumption, it generated a continuous rather than discrete state space and it relied on a more easily scalable class of algorithms.

In contrast with a general tendency of using completely unsupervised techniques for capturing the manifold structure underlying behavioural data [18, 20–23], we proposed a methodology based on supervised learning in order to obtain representations compliant with specific functional constraints. This allowed us to better frame the current work within existing neuroscientific theories of human motivation.

Despite the fact that our work never attempted to model the mechanics of psychological processes directly, we showed how it is possible to leverage computation model of cognitive and affective functions to design data-driven methodologies for inferring the product of such processes. This is important in two fundamental ways. First, it allowed us to construct data-driven solutions in a more principled way, rooting their functional form in the nature of the problem they were trying to solve. Second, it made it easier to specify theory-informed hypotheses *a-priori* and compare them against the behaviour of an otherwise black-box approach.

By connecting this methodology to the construct of engagement we showed how it could be leveraged within an industrial setting as the machine-learned component of a system designed to perfrom cross-games automated engagement prediction and quantification.

The present work leveraged data coming from video games but the approach could easily be applied to other contexts. They only key requirement would be the access to behavioural quantifiers describing the amount and intensity of interactions that an individual has with a particular object, service or task. This means that natural areas of application for our approach are those relying on the remote acquisition of behavioural data (e.g. web services or online experiments) but also situations in which large volumes of experimental data are available (e.g. large multi-center studies).

7.3 Limitations

Of course, this work is not exempt from limitations:

Because we are solving an under-determined inverse problem, the issue of uniqueness arises. Many different latent states might have produced the behavioural patterns that our model observed and there is no guarantee of a strict one-to-one mapping between the representation generated by our model and attributed incentive salience.

Although the findings illustrated in chapter 5 suggest a similarity between the functional characteristics of the representation inferred by our approach and the construct of attributed incentive salience these are the result of mostly qualitative, descriptive or exploratory analyses.

Despite we proposed a way to constrain an ANN architecture to obtain more interpretable latent representations, our approach is still far from being compliant with Occam's razor as it heavily rely on an over-parametrized model.

The mapping from the inferred latent representation to the observed behavioural space presented in section 5.4, despite producing results broadly in line with theories of reward-driven motivation [32, 158, 159] appeared to show some unexpected and potentially contradictory results. Given the observational setting of our work and the unsupervised approach we adopted, the explanations provided in section 5.4 should be taken with caution and be seen as a method for generating testable hypotheses used as the starting point for future investigations.

The limited computing resources available during the development of the thesis didn't allow a more extensive evaluation of our methodology. For example, despite the availability of a large pool of videogame titles we had to select a relatively small number of them in order to be able to effectively run our experiments. A similar issue arose in connection with the number of considered individuals, despite we were able to conduct experiments on sample sizes that were (at the time of conducting the experiments) unprecedented in the videogame literature these were only a relatively small fraction of the user base we had access to.

Despite we proposed the design of a system leveraging our methodology for automated engagement prediction, this was never tested within a production environment.

Lastly, despite our approach having appeared to deal gracefully with objects with different structural characteristics, these were limited to the domain of video games.

7.4 Future Directions

More effort in future research must focus on obtaining a clear formulation of the computations carried out by our architectures. Alternatively, validation could be carried out by comparing the representations generated by our architectures with those inferred from data gathered through laboratory or simulation experiments. An interesting venue for future research would be to evaluate similarities and differences between representations generated by learning algorithms designed according to different models of human motivation (e.g., error-based [34] and error-free [216] models).

Future work should aim to derive quantifiable indices from the representation generated by complex, non linear models such as ANN (in a similar way to the work carried out by Zhang and colleagues [14]). Alternatively, it would be ideal to assess the degree of overlap between model-generated and human-generated representations (as in the work conducted by Roads and Love [232]).

Given a growing literature on the computational substrates of motivation-related cognitive processes [13, 14, 34, 138] an interesting venue for future research would be to move from an approach based on ANNs to one based on latent variable models (e.g., state-space models or methodologies similar to those proposed by Calhoun et. al [18]). This type of approaches, although less scalable, offer a higher degree of control on the computations underlying the dynamics of the latent representations. They allow to specify complex but parsimonious (in terms of number of free parameters) models with very well defined structural forms (e.g., they require the explicit specification of state equations).

Utilising an iterative model building approach similar to the one presented in this thesis it would then be possible to more accurately assess the role of various components of the model while also obtaining more precise functional forms with parameters that are directly interpretable.

Clarifying the the nature of some of the observed discrepancies between what specified by behavioural theories of motivation the results of our partition analyses may require experimental work in more controlled settings.

Finally, more generous computing resources would have allowed for a higher number of experimental replications, better hyper-parameter tuning and more detailed exploration of architectural variations. In addition to more through generalizability stress tests (i.e, considering more videogames contexts and individuals). In this view, future research should focus on scaling and testing our approach against considerably higher volumes of data.

Future effort should focus on evaluating the feasibility of deploying our solution within an industrial setting and most importantly thoroughly documenting eventual advantages and pitfalls that might emerge.

In order to verify the generalizability of our approach, future work should include data generated from a variety of contexts (e.g. web services, online and laboratory-based experiments).

7.5 Closing Remarks

Given the inter-disciplinary nature of the current work, there are multiple directions I would like to see explored in the future.

7.5.1 Engagement Predictions

In the field of engagement predictions the directions I would like to see explored in the future are the following:

- I would like to see more rigorous theoretical works trying to bridge well-established psychological and neuroscientific theories of reward and motivation with the concept of engagement and its behavioural manifestation.

  I foresee that focusing on a blend of computational, experimental and phenomenological approaches would be the most fruitful strategy for generating theories that are not just formally more rigorous but also easier to interpret and validate.

- I would like to see more focus on generic and generalizable models of engagement that do not focus exclusively on a single domain of application (e.g., videogames) but on multiple ones. Treating engagement as an holistic constructs that should be modelled jointly from heterogeneous data generated by different sources.

- I would like to see more works focusing on iterative model building with thorough performance analyses carried out at every stage of the process.

  I believe that partially moving away from the current "rat race" approach to model comparison is an important step for understanding why a given approach performs better or worse than a competing one.

  Very often it is more informative to know which specific aspects of a model drive its behaviour rather than acknowledging its higher or lower perfromance with respect to the state of the art (which is nevertheless still a very important goal to aim for).

7.5.2 Motivation-Related Latent States Estimation

In the field of latent state estimation the directions I would like to see explored in the future are the following:

- I would like to see more hybridisation between data driven works focusing on large volume of observation data and laboratory based experiments.

  In particular, I think the first are useful devices for suggesting research directions and evaluating the generalizability of experimental findings to naturalistic contexts. The second are essential validation procedures required for providing substance to patterns emerged from observational data.

- I would like to see more effort spent on attenuating the inverse problem issue that has affected our work. A focus on unraveling causal rather than correlational links between observed behaviour and latent states would allow us to better understand potential connections with existing findings in the biological neuroscience literature.

- I would like to see more research comparing the outcomes of machine learning, computational and experimental methodologies when attempting to infer latent states generated by psychobiological constructs. Evaluating how and if there is any type of convergence would be extremely important for understanding strengths and limitations of the various approaches.

List of References

[1] European Commission. 2018 reform of eu data protection rules. `https://ec.europa.eu/commission/sites/beta-political/files/data-protection-factsheet-changes_en.pdf`.

[2] Valerio Bonometti, Charles Ringer, Mathieu Ruiz, Alex Wade, and Anders Drachen. From theory to behaviour: Towards a general model of engagement. *arXiv preprint arXiv:2004.12644*, 2020.

[3] Valerio Bonometti, Mathieu J Ruiz, Anders Drachen, and Alex Wade. Approximating attributed incentive salience in large scale scenarios. a representation learning approach based on artificial neural networks. *arXiv preprint arXiv:2108.01724*, 2021.

[4] Valerio Bonometti, Charles Ringer, Mark Hall, Alex R Wade, and Anders Drachen. Modelling early user-game interactions for joint estimation of survival time and churn probability. In *2019 IEEE Conference on Games (CoG)*, pages 1–8. IEEE, 2019.

[5] Myat Aung, Valerio Bonometti, Anders Drachen, Peter Cowling, Athanasios Vasileios Kokkinakis, Christian Yoder, and A Wade. Predicting skill learning in a large, longitudinal moba dataset. In *2018 IEEE conference on computational intelligence and games (CIG)*, pages 1–7. IEEE, 2018.

[6] Ozan Vardal, Valerio Bonometti, Anders Drachen, Alex Wade, and Tom Stafford. Mind the gap: Distributed practice enhances performance in a moba game. *PloS one*, 17(10):e0275843, 2022.

[7] Juan A Gallego, Matthew G Perich, Lee E Miller, and Sara A Solla. Neural manifolds for the control of movement. *Neuron*, 94(5):978–984, 2017.

[8] Dori Derdikman and Edvard I Moser. A manifold of spatial maps in the brain. *Space, Time and Number in the Brain*, pages 41–57, 2011.

[9] Edward H Nieh, Manuel Schottdorf, Nicolas W Freeman, Ryan J Low, Sam Lewallen, Sue Ann Koay, Lucas Pinto, Jeffrey L Gauthier, Carlos D Brody, and David W Tank. Geometry of abstract learned knowledge in the hippocampus. *Nature*, 595(7865):80–84, 2021.

[10] H Sebastian Seung and Daniel D Lee. The manifold ways of perception. *science*, 290(5500):2268–2269, 2000.

[11] Elad Ganmor, Ronen Segev, and Elad Schneidman. A thesaurus for a neural population code. *Elife*, 4:e06134, 2015.

[12] Mark Stopfer, Vivek Jayaraman, and Gilles Laurent. Intensity versus identity coding in an olfactory system. *Neuron*, 39(6):991–1004, 2003.

[13] Samuel M McClure, Nathaniel D Daw, and P Read Montague. A computational substrate for incentive salience. *Trends in neurosciences*, 26(8):423–428, 2003.

[14] Jun Zhang, Kent C Berridge, Amy J Tindell, Kyle S Smith, and J Wayne Aldridge. A neural computational model of incentive salience. *PLoS Comput Biol*, 5(7):e1000437, 2009.

[15] Eyrun Eyjolfsdottir, Kristin Branson, Yisong Yue, and Pietro Perona. Learning recurrent representations for hierarchical behavior modeling. *arXiv preprint arXiv:1611.00094*, 2016.

[16] H Francis Song, Guangyu R Yang, and Xiao-Jing Wang. Reward-based training of recurrent neural networks for cognitive and value-based tasks. *Elife*, 6:e21492, 2017.

[17] Josh Merel, Diego Aldarondo, Jesse Marshall, Yuval Tassa, Greg Wayne, and Bence Ölveczky. Deep neuroethology of a virtual rodent. *arXiv preprint arXiv:1911.09451*, 2019.

[18] Adam J Calhoun, Jonathan W Pillow, and Mala Murthy. Unsupervised identification of the internal states that shape natural behavior. *Nature neuroscience*, 22(12):2040–2049, 2019.

[19] Rich Pang, Benjamin J Lansdell, and Adrienne L Fairhall. Dimensionality reduction in neuroscience. *Current Biology*, 26(14):R656–R660, 2016.

[20] Kevin Luxem, Falko Fuhrmann, Johannes Kürsch, Stefan Remy, and Pavol Bauer. Identifying behavioral structure from deep variational embeddings of animal motion. *BioRxiv*, 2020.

[21] Talmo D Pereira, Joshua W Shaevitz, and Mala Murthy. Quantifying behavior to understand the brain. *Nature neuroscience*, 23(12):1537–1549, 2020.

[22] Michael H McCullough and Geoffrey J Goodhill. Unsupervised quantification of naturalistic animal behaviors for gaining insight into the brain. *Current opinion in neurobiology*, 70:89–100, 2021.

[23] Changhao Shi, Sivan Schwartz, Shahar Levy, Shay Achvat, Maisan Abboud, Amir Ghanayim, Jackie Schiller, and Gal Mishne. Learning disentangled behavior embeddings. *Advances in Neural Information Processing Systems*, 34, 2021.

[24] Magy Seif El-Nasr, Anders Drachen, and Alessandro Canossa. *Game analytics*. Springer, 2016.

[25] Anders Drachen. Behavioral telemetry in games user research. In *Game User Experience Evaluation*, pages 135–165. Springer, 2015.

[26] Andrew G Barto and Thomas G Dietterich. Reinforcement learning and its relationship to supervised learning. *Handbook of learning and approximate dynamic programming*, 10:9780470544785, 2004.

[27] Kevin P Murphy. *Probabilistic machine learning: an introduction*. MIT press, 2022.

[28] Yoshua Bengio, Ian Goodfellow, and Aaron Courville. *Deep learning*, volume 1. MIT press Massachusetts, USA:, 2017.

[29] Saurabh Arora and Prashant Doshi. A survey of inverse reinforcement learning: Challenges, methods and progress. *Artificial Intelligence*, 297:103500, 2021.

[30] Leland McInnes, John Healy, Nathaniel Saul, and Lukas Grossberger. Umap: Uniform manifold approximation and projection. *The Journal of Open Source Software*, 3(29):861, 2018.

[31] Kent C Berridge and Terry E Robinson. What is the role of dopamine in reward: hedonic impact, reward learning, or incentive salience? *Brain research reviews*, 28(3):309–369, 1998.

[32] Kent C Berridge. Motivation concepts in behavioral neuroscience. *Physiology & behavior*, 81(2):179–209, 2004.

[33] Eleanor H Simpson and Peter D Balsam. *Behavioral neuroscience of motivation*. Springer, 2016.

[34] Wolfram Schultz, Peter Dayan, and P Read Montague. A neural substrate of prediction and reward. *Science*, 275(5306):1593–1599, 1997.

[35] Kent C Berridge and Morten L Kringelbach. Affective neuroscience of pleasure: reward in humans and animals. *Psychopharmacology*, 199(3):457–480, 2008.

[36] Wolfram Schultz, Léon Tremblay, and Jeffrey R Hollerman. Reward processing in primate orbitofrontal cortex and basal ganglia. *Cerebral cortex*, 10(3):272–283, 2000.

[37] Guillaume Sescousse, Xavier Caldú, Bàrbara Segura, and Jean-Claude Dreher. Processing of primary and secondary rewards: a quantitative meta-analysis and review of human functional neuroimaging studies. *Neuroscience & Biobehavioral Reviews*, 37(4):681–696, 2013.

[38] Frederick Toates. Comparing motivational systems—an incentive motivation perspective. 1994.

[39] Satoshi Ikemoto and Jaak Panksepp. The role of nucleus accumbens dopamine in motivated behavior: a unifying interpretation with special reference to reward-seeking. *Brain Research Reviews*, 31(1):6–41, 1999.

[40] John D Salamone and Mercè Correa. Motivational views of reinforcement: implications for understanding the behavioral functions of nucleus accumbens dopamine. *Behavioural brain research*, 137(1-2):3–25, 2002.

[41] Jorge Armony and Patrik Vuilleumier. *The Cambridge handbook of human affective neuroscience*. Cambridge university press, 2013.

[42] Laura H Corbit and Bernard W Balleine. Learning and motivational processes contributing to pavlovian–instrumental transfer and their neural bases: dopamine and beyond. In *Behavioral neuroscience of motivation*, pages 259–289. Springer, 2015.

[43] Julius William Kling. *Woodworth and Schlosberg's experimental psychology*. 1971.

[44] Burrhus Frederic Skinner. *Science and human behavior*. Number 92904. Simon and Schuster, 1953.

[45] Larry Squire, Darwin Berg, Floyd E Bloom, Sascha Du Lac, Anirvan Ghosh, and Nicholas C Spitzer. *Fundamental neuroscience*. Academic press, 2012.

[46] Robert C Bolles. Reinforcement, expectancy, and learning. *Psychological review*, 79(5):394, 1972.

[47] Dalbir Bindra. How adaptive behavior is produced: A perceptual-motivational alternative to response-reinforcement. *Behavioral and brain sciences*, 1(1):41–91, 1978.

[48] Kent C Berridge, Terry E Robinson, and J Wayne Aldridge. Dissecting components of reward:'liking','wanting', and learning. *Current opinion in pharmacology*, 9(1):65–73, 2009.

[49] Kyle S Smith, Kent C Berridge, and J Wayne Aldridge. Disentangling pleasure from incentive salience and learning signals in brain reward circuitry. *Proceedings of the National Academy of Sciences*, 108(27):E255–E264, 2011.

[50] Satoshi Ikemoto and Jaak Panksepp. Dissociations between appetitive and consummatory responses by pharmacological manipulations of reward-relevant brain regions. *Behavioral neuroscience*, 110(2):331, 1996.

[51] Terry E Robinson and Kent C Berridge. The neural basis of drug craving: an incentive-sensitization theory of addiction. *Brain research reviews*, 18(3):247–291, 1993.

[52] Amy J Tindell, Kyle S Smith, Kent C Berridge, and J Wayne Aldridge. Dynamic computation of incentive salience:"wanting" what was never "liked". *Journal of Neuroscience*, 29(39):12220–12228, 2009.

[53] Kent C Berridge. From prediction error to incentive salience: mesolimbic computation of reward motivation. *European Journal of Neuroscience*, 35(7):1124–1143, 2012.

[54] Mike James F Robinson, AM Fischer, A Ahuja, EN Lesser, and H Maniates. Roles of "wanting" and "liking" in motivating behavior: gambling, food, and drug addictions. In *Behavioral neuroscience of motivation*, pages 105–136. Springer, 2015.

[55] Paul J Meyer, Christopher P King, and Carrie R Ferrario. Motivational processes underlying substance abuse disorder. In *Behavioral Neuroscience of Motivation*, pages 473–506. Springer, 2015.

[56] Wolfram Schultz. Multiple reward signals in the brain. *Nature reviews neuroscience*, 1(3):199–207, 2000.

[57] Shelly B Flagel, Jeremy J Clark, Terry E Robinson, Leah Mayo, Alayna Czuj, Ingo Willuhn, Christina A Akers, Sarah M Clinton, Paul EM Phillips, and Huda Akil. A selective role for dopamine in stimulus–reward learning. *Nature*, 469(7328):53–57, 2011.

[58] Thomas M Connolly, Elizabeth A Boyle, Ewan MacArthur, Thomas Hainey, and James M Boyle. A systematic literature review of empirical evidence on computer games and serious games. *Computers & education*, 59(2):661–686, 2012.

[59] Elizabeth A Boyle, Thomas M Connolly, Thomas Hainey, and James M Boyle. Engagement in digital entertainment games: A systematic review. *Computers in human behavior*, 28(3):771–780, 2012.

[60] David Zendle, Catherine Flick, Darel Halgarth, Nicholas Ballou, Simon Demediuk, and Anders Drachen. Transnational patterns in mobile playtime: An analysis of 118 billion hours of human data. *OSF Preprints*, 2022.

[61] Heather L O'Brien and Elaine G Toms. What is user engagement? a conceptual framework for defining user engagement with technology. *Journal of the American society for Information Science and Technology*, 59(6):938–955, 2008.

[62] Mihaly Csikszentmihalyi. Toward a psychology of optimal experience. In *Flow and the foundations of positive psychology*, pages 209–226. Springer, 2014.

[63] Charlene Jennett, Anna L Cox, Paul Cairns, Samira Dhoparee, Andrew Epps, Tim Tijs, and Alison Walton. Measuring and defining the experience of immersion. *International journal of human-computer studies*, 66(9):641–661, 2008.

[64] Kristen Lucas and John L Sherry. Sex differences in video game play: A communication-based explanation. *Communication research*, 31(5):499–523, 2004.

[65] DL King, PH Delfabbro, and MD Griffiths. The role of structural characteristics in problem video game playing: A review. *Cyberpsychology: Journal of Psychosocial Research on Cyberspace*, 4(1), 2010.

[66] Daniel King, Paul Delfabbro, and Mark Griffiths. Video game structural characteristics: A new psychological taxonomy. *International journal of mental health and addiction*, 8(1):90–106, 2010.

[67] Georgios N Yannakakis, Pieter Spronck, Daniele Loiacono, and Elisabeth André. Player modeling. In *Dagstuhl Follow-Ups*, volume 6. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2013.

[68] Andrew K Przybylski, C Scott Rigby, and Richard M Ryan. A motivational model of video game engagement. *Review of general psychology*, 14(2):154–166, 2010.

[69] Dave Westwood and Mark D Griffiths. The role of structural characteristics in video-game play motivation: A q-methodology study. *Cyberpsychology, Behavior, and Social Networking*, 13(5):581–585, 2010.

[70] Hao Wang and Chuen-Tsai Sun. Game reward systems: Gaming experiences and social meanings. In *DiGRA Conference*, pages 1–15, 2011.

[71] Cody Phillips, Daniel Johnson, and Peta Wyeth. Videogame reward types. In *Proceedings of the First International Conference on Gameful Design, Research, and Applications*, pages 103–106. ACM, 2013.

[72] Richard Bartle. Hearts, clubs, diamonds, spades: Players who suit muds. *Journal of MUD research*, 1(1):19, 1996.

[73] Nick Yee. Motivations for play in online games. *CyberPsychology & behavior*, 9(6):772–775, 2006.

[74] Lennart E Nacke, Chris Bateman, and Regan L Mandryk. Brainhex: preliminary results from a neurobiological gamer typology survey. In *International conference on entertainment computing*, pages 288–293. Springer, 2011.

[75] Franz Joseph Gall. *On the functions of the brain and of each of its parts: With observations on the possibility of determining the instincts, propensities, and talents, or the moral and intellectual dispositions of men and animals, by the configuration of the brain and head*, volume 1. Marsh, Capen & Lyon, 1835.

[76] Gregory J Boyle. Myers-briggs type indicator (mbti): some psychometric limitations. *Australian Psychologist*, 30(1):71–74, 1995.

[77] Richard M Ryan and Edward L Deci. Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American psychologist*, 55(1):68, 2000.

[78] Richard M Ryan, C Scott Rigby, and Andrew Przybylski. The motivational pull of video games: A self-determination theory approach. *Motivation and emotion*, 30(4):344–360, 2006.

[79] From Software. Darksouls. Steam, 2009.

[80] King. Candy crush. Android, 2012.

[81] Charles S Carver and Teri L White. Behavioral inhibition, behavioral activation, and affective responses to impending reward and punishment: the bis/bas scales. *Journal of personality and social psychology*, 67(2):319, 1994.

[82] Darius Ašeriškis and Robertas Damaševičius. Computational evaluation of effects of motivation reinforcement on player retention. *Journal of Universal Computer Science*, 23(5):432–453, 2017.

[83] Susanne Poeller, Max V Birk, Nicola Baumann, and Regan L Mandryk. Let me be implicit: Using motive disposition theory to predict and explain behaviour in digital games. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pages 1–15, 2018.

[84] Alessandro Canossa, Josep B Martinez, and Julian Togelius. Give me a reason to dig minecraft and psychology of motivation. In *2013 IEEE Conference on Computational Inteligence in Games (CIG)*, pages 1–8. IEEE, 2013.

[85] Deividas Stankevicius, Hawraa Amira Jady, Anders Drachen, and Henrik Schoenau-Fog. A factor-based exploration of player's continuation desire in free-to-play mobile games. In *Eleventh Artificial Intelligence and Interactive Digital Entertainment Conference*, 2015.

[86] Mike Schaekermann, Giovanni Ribeiro, Guenter Wallner, Simone Kriglstein, Daniel Johnson, Anders Drachen, Rafet Sifa, and Lennart E Nacke. Curiously motivated: Profiling curiosity with self-reports and behaviour metrics in the game" destiny". In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, pages 143–156, 2017.

[87] Giel Van Lankveld, Sonny Schreurs, and Pieter Spronck. Psychologically verified player modelling. In *GAMEON*, pages 12–19, 2009.

[88] Paul T Costa Jr and Robert R McCrae. *The Revised Neo Personality Inventory (neo-pi-r).* Sage Publications, Inc, 2008.

[89] Giel Van Lankveld, Pieter Spronck, Jaap Van den Herik, and Arnoud Arntz. Games as personality profiling tools. In *2011 IEEE Conference on Computational Intelligence and Games (CIG'11)*, pages 197–202. IEEE, 2011.

[90] Gustavo F Tondello, Rina R Wehbe, Lisa Diamond, Marc Busch, Andrzej Marczewski, and Lennart E Nacke. The gamification user types hexad scale. In *Proceedings of the 2016 annual symposium on computer-human interaction in play*, pages 229–243, 2016.

[91] John T Cacioppo, Louis G Tassinary, and Gary Berntson. *Handbook of psychophysiology*. Cambridge university press, 2007.

[92] Anders Drachen, Pejman Mirza-Babaei, and Lennart E Nacke. *Games user research*. Oxford University Press, 2018.

[93] Pejman Mirza-Babaei, Lennart E Nacke, John Gregory, Nick Collins, and Geraldine Fitzpatrick. How does it play better?: exploring user testing and biometric storyboards in games user research. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1499–1508. ACM, 2013.

[94] Xuejie Liu, Kathryn Merrick, and Hussein Abbass. Toward electroencephalographic profiling of player motivation: A survey. *IEEE Transactions on Cognitive and Developmental Systems*, 10(3):499–513, 2017.

[95] Amy Orben and Andrew K Przybylski. The association between adolescent well-being and digital technology use. *Nature human behaviour*, 3(2):173–182, 2019.

[96] Joseph P Simmons, Leif D Nelson, and Uri Simonsohn. False-positive psychology: undisclosed flexibility in data collection and analysis allows presenting anything as significant. 2016.

[97] Rafet Sifa, Anders Drachen, Christian Bauckhage, Christian Thurau, and Alessandro Canossa. Behavior evolution in tomb raider underworld. In *2013 IEEE Conference on Computational Inteligence in Games (CIG)*, pages 1–8. IEEE, 2013.

[98] Johanna Pirker, Simone Griesmayr, Anders Drachen, and Rafet Sifa. How playstyles evolve: Progression analysis and profiling in just cause 2. In *International Conference on Entertainment Computing*, pages 90–101. Springer, 2016.

[99] Anders Tychsen and Alessandro Canossa. Defining personas in games using metrics. In *Proceedings of the 2008 conference on future play: Research, play, share*, pages 73–80, 2008.

[100] Anders Drachen, Alessandro Canossa, and Georgios N Yannakakis. Player modeling using self-organization in tomb raider: Underworld. In *2009 IEEE symposium on computational intelligence and games*, pages 1–8. IEEE, 2009.

[101] Anders Drachen, Rafet Sifa, Christian Bauckhage, and Christian Thurau. Guns, swords and data: Clustering of player behavior in computer games in the wild. In *2012 IEEE conference on Computational Intelligence and Games (CIG)*, pages 163–170. IEEE, 2012.

[102] Georgios N Yannakakis and John Hallam. Game and player feature selection for entertainment capture. In *2007 IEEE Symposium on Computational Intelligence and Games*, pages 244–251. IEEE, 2007.

[103] Christian Bauckhage, Kristian Kersting, Rafet Sifa, Christian Thurau, Anders Drachen, and Alessandro Canossa. How players lose interest in playing a game: An empirical study based on distributions of total playing times. In *2012 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 139–146. IEEE, 2012.

[104] Magy Seif El-Nasr, Truong-Huy D Nguyen, Alessandro Canossa, and Anders Drachen. *Game Data Science*. Oxford University Press, 2021.

[105] Julian Runge, Peng Gao, Florent Garcin, and Boi Faltings. Churn prediction for high-value players in casual social games. In *2014 IEEE conference on Computational Intelligence and Games*, pages 1–8. IEEE, 2014.

[106] Seungwook Kim, Daeyoung Choi, Eunjung Lee, and Wonjong Rhee. Churn prediction of mobile and online casual games using play log data. *PloS one*, 12(7):e0180735, 2017.

[107] Fabian Hadiji, Rafet Sifa, Anders Drachen, Christian Thurau, Kristian Kersting, and Christian Bauckhage. Predicting player churn in the wild. In *2014 IEEE Conference on Computational Intelligence and Games*, pages 1–8. IEEE, 2014.

[108] Benjamin Cowley and Darryl Charles. Behavlets: a method for practical player modelling using psychology-based player traits and domain specific features. *User Modeling and User-Adapted Interaction*, 26(2):257–306, 2016.

[109] Baoxiang Wang, Tongfang Sun, and Xianjun Sam Zheng. Beyond winning and losing: modeling human motivations and behaviors using inverse reinforcement learning. *arXiv preprint arXiv:1807.00366*, 2018.

[110] Mateusz Bialas, Shoshannah Tekofsky, and Pieter Spronck. Cultural influences on play style. In *2014 IEEE Conference on Computational Intelligence and Games*, pages 1–7. IEEE, 2014.

[111] Dulakshi Vihanga, Michael Barlow, Erandi Lakshika, and Kathryn Kasmarik. Weekly seasonal player population patterns in online games: A time series clustering approach. In *2019 IEEE Conference on Games (CoG)*, pages 1–8. IEEE, 2019.

[112] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[113] Anders Drachen, Eric Thurston Lundquist, Yungjen Kung, Pranav Rao, Rafet Sifa, Julian Runge, and Diego Klabjan. Rapid prediction of player retention in free-to-play mobile games. In *Twelfth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2016.

[114] Miloš Milošević, Nenad Živić, and Igor Andjelković. Early churn prediction with personalized targeting in mobile social games. *Expert Systems with Applications*, 83:326–332, 2017.

[115] África Periáñez, Alain Saas, Anna Guitart, and Colin Magne. Churn prediction in mobile social games: Towards a complete assessment using survival ensembles. In *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 564–573. IEEE, 2016.

[116] Simon Demediuk, Alexandra Murrin, David Bulger, Michael Hitchens, Anders Drachen, William L Raffe, and Marco Tamassia. Player retention in league of legends: a study using survival analysis. In *Proceedings of the Australasian Computer Science Week Multiconference*, page 43. ACM, 2018.

[117] Paul Bertens, Anna Guitart, and África Periáñez. Games and big data: A scalable multi-dimensional churn prediction model. In *2017 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 33–36. IEEE, 2017.

[118] Markus Viljanen, Antti Airola, Jukka Heikkonen, and Tapio Pahikkala. Playtime measurement with survival analysis. *IEEE Transactions on Games*, 10(2):128–138, 2018.

[119] Emiliano G Castro and Marcos SG Tsuzuki. Churn prediction in online games using players' login records: A frequency analysis approach. *IEEE Transactions on Computational Intelligence and AI in Games*, 7(3):255–265, 2015.

[120] Pierangelo Rothenbuehler, Julian Runge, Florent Garcin, and Boi Faltings. Hidden markov models for churn prediction. In *2015 sai intelligent systems conference (intellisys)*, pages 723–730. IEEE, 2015.

[121] Marco Tamassia, William Raffe, Rafet Sifa, Anders Drachen, Fabio Zambetta, and Michael Hitchens. Predicting player churn in destiny: A hidden markov models approach to predicting player departure in a major online game. In *2016 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 1–8. IEEE, 2016.

[122] Anna Guitart, Pei Pei Chen, and África Periáñez. The winning solution to the ieee cig 2017 game data mining competition. *Machine Learning and Knowledge Extraction*, 1(1):252–264, 2018.

[123] Xi Liu, Muhe Xie, Xidao Wen, Rui Chen, Yong Ge, Nick Duffield, and Na Wang. A semi-supervised and inductive embedding model for churn prediction of large-scale mobile games. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 277–286. IEEE, 2018.

[124] Jeppe Theiss Kristensen and Paolo Burelli. Combining sequential and aggregated data for churn prediction in casual freemium games. In *2019 IEEE Conference on Games (CoG)*, pages 1–8. IEEE, 2019.

[125] Xi Liu, Muhe Xie, Xidao Wen, Rui Chen, Yong Ge, Nick Duffield, and Na Wang. Micro-and macro-level churn analysis of large-scale mobile games. *Knowledge and Information Systems*, pages 1–32, 2019.

[126] Shaghayegh Roohi, Asko Relas, Jari Takatalo, Henri Heiskanen, and Perttu Hämäläinen. Predicting game difficulty and churn without players. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, pages 585–593, 2020.

[127] Anna Guitart, Ana Fernández del Río, and África Periáñez. Understanding player engagement and in-game purchasing behavior with ensemble learning. *arXiv preprint arXiv:1907.03947*, 2019.

[128] Ana Fernández del Río, Pei Pei Chen, and Africa Periánez. Profiling players with engagement predictions. In *2019 IEEE Conference on Games (CoG)*, pages 1–4. IEEE, 2019.

[129] EunJo Lee, Yoonjae Jang, Du-Mim Yoon, JiHoon Jeon, Sung-il Yang, SangKwang Lee, Dae-Wook Kim, Pei Pei Chen, Anna Guitart, Paul Bertens, et al. Game data mining competition

on churn prediction and survival analysis using game log data. *IEEE Transactions on Games*, 2018.

[130] Anders Drachen, Christian Thurau, Rafet Sifa, and Christian Bauckhage. A comparison of methods for player clustering via behavioral telemetry. *arXiv preprint arXiv:1407.3950*, 2014.

[131] Christian Bauckhage, Anders Drachen, and Rafet Sifa. Clustering game behavior data. *IEEE Transactions on Computational Intelligence and AI in Games*, 7(3):266–278, 2014.

[132] Sasha Makarovych, Alessandro Canossa, Julian Togelius, and Anders Drachen. Like a dna string: Sequence-based player profiling in tom clancy's the division. In *Artificial Intelligence and Interactive Digital Entertainment Conference*. York, 2018.

[133] Myat Aung, Simon Demediuk, Yuan Sun, Ye Tu, Yu Ang, Siva Nekkanti, Shantanu Raghav, Diego Klabjan, Rafet Sifa, and Anders Drachen. The trails of just cause 2: spatio-temporal player profiling in open-world games. In *Proceedings of the 14th International Conference on the Foundations of Digital Games*, pages 1–11, 2019.

[134] Hanting Xie, Sam Devlin, Daniel Kudenko, and Peter Cowling. Predicting player disengagement and first purchase with event-frequency based data representation. In *2015 IEEE Computational Intelligence and Games*, pages 230–237. IEEE, 2015.

[135] David R Cox. Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological)*, 34(2):187–202, 1972.

[136] Ana Fernández del Río, Anna Guitart, and África Periáñez. A time series approach to player churn and conversion in videogames. *arXiv preprint arXiv:2003.10287*, 2020.

[137] David Reguera, Pol Colomer-de Simón, Iván Encinas, Manel Sort, Jan Wedekind, and Marián Boguñá. Quantifying human engagement into playful activities. *Scientific Reports*, 10(1):1–7, 2020.

[138] Jane X Wang, Zeb Kurth-Nelson, Dharshan Kumaran, Dhruva Tirumala, Hubert Soyer, Joel Z Leibo, Demis Hassabis, and Matthew Botvinick. Prefrontal cortex as a meta-reinforcement learning system. *Nature neuroscience*, 21(6):860–868, 2018.

[139] Charles Spearman. " general intelligence" objectively determined and measured. 1961.

[140] Ibrahim Abaker Targio Hashem, Ibrar Yaqoob, Nor Badrul Anuar, Salimah Mokhtar, Abdullah Gani, and Samee Ullah Khan. The rise of "big data" on cloud computing: Review and open research issues. *Information systems*, 47:98–115, 2015.

[141] Benjamin Schuster-Böckler and Alex Bateman. An introduction to hidden markov models. *Current protocols in bioinformatics*, 18(1):A–3A, 2007.

[142] Panayiota Touloupou, Bärbel Finkenstädt, and Simon EF Spencer. Scalable bayesian inference for coupled hidden markov and semi-markov models. *Journal of Computational and Graphical Statistics*, 29(2):238–249, 2020.

[143] Kyoung-Su Oh and Keechul Jung. Gpu implementation of neural networks. *Pattern Recognition*, 37(6):1311–1314, 2004.

[144] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[145] Omri Barak. Recurrent neural networks as versatile tools of neuroscience research. *Current opinion in neurobiology*, 46:1–6, 2017.

[146] Tim C Kietzmann, Patrick McClure, and Nikolaus Kriegeskorte. Deep neural networks in computational neuroscience. *BioRxiv*, page 133504, 2018.

[147] Justin Chumbley and Mark Griffiths. Affect and the computer game player: the effect of gender, personality, and game reinforcement structure on affective responses to computer game-play. *CyberPsychology & Behavior*, 9(3):308–316, 2006.

[148] Tushar Agarwal, Keith Burghardt, and Kristina Lerman. On quitting: performance and practice in online game play. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11, pages 452–455, 2017.

[149] Mark Steyvers and Aaron S Benjamin. The joint contribution of participation and performance to learning functions: Exploring the effects of age in large-scale data sets. *Behavior research methods*, 51(4):1531–1543, 2019.

[150] Richard S Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3(1):9–44, 1988.

[151] Ethan S Bromberg-Martin, Okihide Hikosaka, and Kae Nakamura. Coding of task reward value in the dorsal raphe nucleus. *Journal of Neuroscience*, 30(18):6262–6272, 2010.

[152] Siyuan Gao, Gal Mishne, and Dustin Scheinost. Nonlinear manifold learning in functional magnetic resonance imaging uncovers a low-dimensional space of brain dynamics. *Human brain mapping*, 42(14):4510–4524, 2021.

[153] Joan Rué-Queralt, Angus Stevner, Enzo Tagliazucchi, Helmut Laufs, Morten L Kringelbach, Gustavo Deco, and Selen Atasoy. Decoding brain states on the intrinsic manifold of human brain dynamics across wakefulness and sleep. *Communications Biology*, 4(1):1–11, 2021.

[154] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[155] John P O'Doherty, Peter Dayan, Karl Friston, Hugo Critchley, and Raymond J Dolan. Temporal difference models and reward-related learning in the human brain. *Neuron*, 38(2):329–337, 2003.

[156] Wolfram Schultz. Reward prediction error. *Current Biology*, 27(10):R369–R371, 2017.

[157] Stefano Palminteri, Mehdi Khamassi, Mateus Joffily, and Giorgio Coricelli. Contextual modulation of value signals in reward and punishment learning. *Nature communications*, 6(1):1–14, 2015.

[158] Edward L Thorndike. The law of effect. *The American journal of psychology*, 39(1/4):212–222, 1927.

[159] Burrhus Frederic Skinner. *Science and human behavior*. Number 92904. Simon and Schuster, 1965.

[160] Fumiko Hoeft, Christa L Watson, Shelli R Kesler, Keith E Bettinger, and Allan L Reiss. Gender differences in the mesocorticolimbic system during computer game-play. *Journal of psychiatric research*, 42(4):253–258, 2008.

[161] Krystyna A Mathiak, Martin Klasen, René Weber, Hermann Ackermann, Sukhwinder S Shergill, and Klaus Mathiak. Reward system and temporal pole contributions to affective evaluation during a first person shooter video game. *BMC neuroscience*, 12(1):1–11, 2011.

[162] Steven W Cole, Daniel J Yoo, and Brian Knutson. Interactivity and reward-related neural activation during a serious videogame. *PLoS one*, 7(3):e33909, 2012.

[163] Martin Klasen, René Weber, Tilo TJ Kircher, Krystyna A Mathiak, and Klaus Mathiak. Neural contributions to flow experience during video game playing. *Social cognitive and affective neuroscience*, 7(4):485–495, 2012.

[164] Robert C Lorenz, Tobias Gleich, Jürgen Gallinat, and Simone Kühn. Video game training and the reward system. *Frontiers in human neuroscience*, 9:40, 2015.

[165] Tobias Gleich, Robert C Lorenz, Jürgen Gallinat, and Simone Kühn. Functional changes in the reward circuit in response to gaming-related cues after training with a commercial video game. *Neuroimage*, 152:467–475, 2017.

[166] Matthias J Koepp, Roger N Gunn, Andrew D Lawrence, Vincent J Cunningham, Alain Dagher, Tasmin Jones, David J Brooks, Christopher J Bench, and PM Grasby. Evidence for striatal dopamine release during a video game. *Nature*, 393(6682):266–268, 1998.

[167] Aaron Drummond and James D Sauer. Video game loot boxes are psychologically akin to gambling. *Nature Human Behaviour*, 2(8):530–532, 2018.

[168] David Zendle and Paul Cairns. Video game loot boxes are linked to problem gambling: Results of a large-scale survey. *PloS one*, 13(11):e0206767, 2018.

[169] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

[170] Bernard Widrow and Marcian E Hoff. Adaptive switching circuits. Technical report, Stanford Univ Ca Stanford Electronics Labs, 1960.

[171] Kurt Hornik, Maxwell Stinchcombe, Halbert White, et al. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[172] Timothy P Lillicrap and Adam Santoro. Backpropagation through time and the brain. *Current opinion in neurobiology*, 55:82–89, 2019.

[173] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[174] Timothy P Lillicrap, Adam Santoro, Luke Marris, Colin J Akerman, and Geoffrey Hinton. Backpropagation and the brain. *Nature Reviews Neuroscience*, pages 1–12, 2020.

[175] Yuyang Wang, Alex Smola, Danielle Maddix, Jan Gasthaus, Dean Foster, and Tim Januschowski. Deep factors for forecasting. In *International conference on machine learning*, pages 6607–6617. PMLR, 2019.

[176] Michael J Crawley. Mixed-effects models. *The R book*, pages 681–714, 2007.

[177] Andrew Gelman, Aki Vehtari, Daniel Simpson, Charles C Margossian, Bob Carpenter, Yuling Yao, Lauren Kennedy, Jonah Gabry, Paul-Christian Bürkner, and Martin Modrák. Bayesian workflow. *arXiv preprint arXiv:2011.01808*, 2020.

[178] Trevor J Hastie. Generalized additive models. In *Statistical models in S*, pages 249–307. Routledge, 2017.

[179] Rishabh Agarwal, Levi Melnick, Nicholas Frosst, Xuezhou Zhang, Ben Lengerich, Rich Caruana, and Geoffrey E Hinton. Neural additive models: Interpretable machine learning with neural nets. *Advances in Neural Information Processing Systems*, 34:4699–4711, 2021.

[180] François Chollet et al. Keras. `https://keras.io`, 2015.

[181] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[182] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[183] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320, 2005.

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

[185] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[186] Leslie N Smith. Cyclical learning rates for training neural networks. In *2017 Winter Conference on Applications of Computer Vision*, pages 464–472. IEEE, 2017.

[187] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059, 2016.

[188] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[189] Rob J Hyndman and George Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018.

[190] John A Nevin and Randolph C Grace. Behavioral momentum and the law of effect. *Behavioral and Brain Sciences*, 23(1):73–90, 2000.

[191] Tom O'Malley, Elie Bursztein, James Long, François Chollet, Haifeng Jin, Luca Invernizzi, et al. Keras Tuner. `https://github.com/keras-team/keras-tuner`, 2019.

[192] Lisha Li, Kevin Jamieson, Giulia DeSalvo, Afshin Rostamizadeh, and Ameet Talwalkar. Hyperband: A novel bandit-based approach to hyperparameter optimization. *The Journal of Machine Learning Research*, 18(1):6765–6816, 2017.

[193] Kevin Jamieson. Hyperband: A novel bandit-based approach to hyperparameter optimization.

[194] Lingxue Zhu and Nikolay Laptev. Deep and confident prediction for time series at uber. In *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 103–110. IEEE, 2017.

[195] Skipper Seabold and Josef Perktold. statsmodels: Econometric and statistical modeling with python. In *9th Python in Science Conference*, 2010.

[196] The pandas development team. pandas-dev/pandas: Pandas, February 2020.

[197] Charles R. Harris, K. Jarrod Millman, St'efan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fern'andez del R'ıo, Mark Wiebe, Pearu Peterson, Pierre G'erard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020.

[198] Shiwei Zhao, Runze Wu, Jianrong Tao, Manhu Qu, Hao Li, and Changjie Fan. Multi-source data multi-task learning for profiling players in online games. In *2020 IEEE Conference on Games (CoG)*, pages 104–111. IEEE, 2020.

[199] Richard Meyes, Melanie Lu, Constantin Waubert de Puiseau, and Tobias Meisen. Ablation studies in artificial neural networks. *arXiv preprint arXiv:1901.08644*, 2019.

[200] Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, 2(11):559–572, 1901.

[201] L. McInnes, J. Healy, and J. Melville. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *ArXiv e-prints*, February 2018.

[202] Alignedumap. `https://umap-learn.readthedocs.io/en/latest/aligned_umap_basic_usage.html`. Accessed: 2021-04-30.

[203] Adam Pearce Andy Coenen. Understanding umap.

[204] David N Reshef, Yakir A Reshef, Hilary K Finucane, Sharon R Grossman, Gilean McVean, Peter J Turnbaugh, Eric S Lander, Michael Mitzenmacher, and Pardis C Sabeti. Detecting novel associations in large data sets. *science*, 334(6062):1518–1524, 2011.

[205] Davide Albanese, Michele Filosi, Roberto Visintainer, Samantha Riccadonna, Giuseppe Jurman, and Cesare Furlanello. Minerva and minepy: a c engine for the mine suite and its r, python and matlab wrappers. *Bioinformatics*, 29(3):407–408, 2013.

[206] John D Hunter. Matplotlib: A 2d graphics environment. *IEEE Annals of the History of Computing*, 9(03):90–95, 2007.

[207] Michael L Waskom. Seaborn: statistical data visualization. *Journal of Open Source Software*, 6(60):3021, 2021.

[208] Daniel Servén and Charlie Brummitt. pygam: Generalized additive models in python, March 2018.

[209] David McClure. Open syllabus. `https://galaxy.opensyllabus.org/`.

[210] David Sculley. Web-scale k-means clustering. In *Proceedings of the 19th international conference on World wide web*, pages 1177–1178, 2010.

[211] Ville Satopaa, Jeannie Albrecht, David Irwin, and Barath Raghavan. Finding a" dle" in a haystack: Detecting knee points in system behavior. In *2011 31st international conference on distributed computing systems workshops*, pages 166–171. IEEE, 2011.

[212] Guido Van Rossum and Fred L. Drake. *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA, 2009.

[213] Andrew Gelman, Jennifer Hill, and Aki Vehtari. *Regression and other stories*. Cambridge University Press, 2020.

[214] Martha Hotz Vitaterna, Joseph S Takahashi, and Fred W Turek. Overview of circadian rhythms. *Alcohol research & health*, 25(2):85, 2001.

[215] Dulakshi Wannigamage. *Player Population Patterns in Digital Games: A Data Analytics and Machine Learning Approach*. PhD thesis, UNSW Sydney, 2021.

[216] Karl Friston, Spyridon Samothrakis, and Read Montague. Active inference and agency: optimal control without cost functions. *Biological cybernetics*, 106(8):523–541, 2012.

[217] Meinard Müller. Dynamic time warping. *Information retrieval for music and motion*, pages 69–84, 2007.

[218] Raphael Amit and Christoph Zott. Value creation in e-business. *Strategic management journal*, 22(6-7):493–520, 2001.

[219] Khaled Mohammad Alomari, Tariq Rahim Soomro, and Khaled Shaalan. Mobile gaming trends and revenue models. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pages 671–683. Springer, 2016.

[220] Qiang Yang, Yang Liu, Yong Cheng, Yan Kang, Tianjian Chen, and Han Yu. Federated learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 13(3):1–207, 2019.

[221] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

[222] James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. *Advances in neural information processing systems*, 24, 2011.

[223] Paul Bertens, Anna Guitart, Pei Pei Chen, and África Periáñez. A machine-learning item recommendation system for video games. In *2018 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 1–4. IEEE, 2018.

[224] Liangwei Yang, Zhiwei Liu, Yu Wang, Chen Wang, Ziwei Fan, and Philip S Yu. Large-scale personalized video game recommendation via social-aware contextualized graph neural network. In *Proceedings of the ACM Web Conference 2022*, pages 3376–3386, 2022.

[225] Pei Pei Chen, Anna Guitart, Ana Fernández del Río, and Africa Periánez. Customer lifetime value in video games using deep learning and parametric models. In *2018 IEEE international conference on big data (big data)*, pages 2134–2140. IEEE, 2018.

[226] Christian Bauckhage, Rafet Sifa, Anders Drachen, Christian Thurau, and Fabian Hadiji. Beyond heatmaps: Spatio-temporal clustering using behavior-based partitioning of game levels. In *2014 IEEE conference on computational intelligence and games*, pages 1–8. IEEE, 2014.

[227] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35, 2021.

[228] Yogesh K Dwivedi, Laurie Hughes, Elvira Ismagilova, Gert Aarts, Crispin Coombs, Tom Crick, Yanqing Duan, Rohita Dwivedi, John Edwards, Aled Eirug, et al. Artificial intelligence (ai): Multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy. *International Journal of Information Management*, 57:101994, 2021.

[229] Sam Corbett-Davies and Sharad Goel. The measure and mismeasure of fairness: A critical review of fair machine learning. *arXiv preprint arXiv:1808.00023*, 2018.

[230] Elena Petrovskaya, Sebastian Deterding, and David I Zendle. Prevalence and salience of problematic microtransactions in top-grossing mobile and pc games: A content analysis of user reviews. In *CHI Conference on Human Factors in Computing Systems*, pages 1–12, 2022.

[231] Sebastian Deterding, Marc Malmdorf Andersen, Julian Kiverstein, and Mark Miller. Mastering uncertainty: A predictive processing account of enjoying uncertain success in video game play. *Frontiers in psychology*, page 4214, 2022.

[232] Brett D Roads and Bradley C Love. Enriching imagenet with human similarity judgments and psychological embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3547–3557, 2021.

[233] patsy - describing statistical models in python. `https://zenodo.org/record/1472929`. Accessed: 2022-10-09.

[234] Aki Vehtari and Andrew Gelman. "recommendations for priors for student t's degrees of freedom parameter". `https://statmodeling.stat.columbia.edu/2015/05/17/do-we-have-any-recommendations-for-priors-for-student_ts-degrees-of-freedom-parameter`. Accessed: 2022-10-09.

[235] Alp Kucukelbir, Dustin Tran, Rajesh Ranganath, Andrew Gelman, and David M Blei. Automatic differentiation variational inference. *Journal of machine learning research*, 2017.

[236] John Salvatier, Thomas V Wiecki, and Christopher Fonnesbeck. Probabilistic programming in python using pymc3. *PeerJ Computer Science*, 2:e55, 2016.

Appendix A

Frequent Notation

We will provide here the notation associated with operations frequently used throughout the thesis. Notation is referenced when it appears in the main body of the manuscript for the first time. Vectors are written in low-case (x) while matrices in upper-case (X). Parameters are indicated by the Greek letter θ except for weights (W) and biases (b) used when defining ANN operations.

A.1 Identity function

Given a vector x the identity function id is defined as

$$id(x) = x \tag{A.1}$$

A.2 Sigmoid function

Given a vector x the sigmoid function $sigmoid$ is defined as

$$sigmoid(x) = \frac{1}{1+e^{-x}} \tag{A.2}$$

A.3 Hyperbolic Function

Given a vector x the hyperbolic function tanh is defined as

$$\tanh(x) = \frac{e^{2x}-1}{e^{2x}+1} \tag{A.3}$$

A.4 ReLU Function

Given a vector x the Rectified Linear Unit function *ReLu* is defined as

$$ReLU(x) = \max(0, x) \tag{A.4}$$

A.5 Leaky ReLU Function

Given a vector x the Leaky Rectified Linear Unit function *ReLu* is defined as

$$LReLU(x) = \begin{cases} x, & \text{if } x > 0 \\ \alpha x, & \text{otherwise} \end{cases} \tag{A.5}$$

with $\alpha = 0.3$ being a small constant controlling the slope coefficient for the negative part of the function.

A.6 ELU Function

Given a vector x the Exponential Linear Unit function *ELu* is defined as

$$ELU(x) = \begin{cases} x, & \text{if } x > 0 \\ \alpha(e^x - 1), & \text{otherwise} \end{cases} \tag{A.6}$$

with $\alpha = 1$ being the scale for the negative part of the function.

A.7 Mean Squared Error

Given a vector of ground truth values $y \in \mathbb{R}^N$ and predictions $\widehat{y} \in \mathbb{R}^N$ the mean squared error (MSE) is defined as

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \tag{A.7}$$

A.8 Symmetric Mean Absolute Percentage Error

Given a vector of ground truth values $y \in \mathbb{R}^N$ and predictions $\widehat{y} \in \mathbb{R}^N$ the symmetric mean absolute percentage error (SMAPE) is defined as

$$SMAPE(y, \widehat{y}) = 100 * \frac{1}{N} \sum_{i=1}^{N} \frac{|y_i - \widehat{y}_i|}{|y_i| + |\widehat{y}_i|} \tag{A.8}$$

A.9 Binary Cross-Entropy

Given a vector of ground truth values $y \in \mathbb{Z}_2^N$ and predictions $\widehat{y} \in \mathbb{R}_{[0,1]}^N$ the Binary Cross-Entropy (BCE) is defined as

$$\text{BCE} = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot log(\hat{y}_i) + (1-y_i) \cdot log(1-\hat{y}_i) \tag{A.9}$$

A.10 F1 Score

Given a vector of ground truth values $y \in \mathbb{R}_{[0,1]}^N$ and predictions $\widehat{y} \in \mathbb{R}_{[0,1]}^N$ the F1 score is defined as

$$\text{F1} = 2 \cdot \frac{(precision \cdot recall)}{(precision + recall)} \tag{A.10}$$

with $precision = \frac{TP}{(TP+FP)}$ and $recall = \frac{TP}{(TP+FN)}$, where *TP, FP, TN, FN* stand for True Positives, False Positives, True Negatives and False Negatives.

A.11 Inertia

Given a matrix $X \in \mathbb{R}^{N \times B}$, a set of centroids $C \in \mathbb{R}^{K \times B}$ inferred by the KMeans algorithm and used for dividing the matrix in K partitions, the inertia of the provided solution is defined as

$$\text{inertia} = \sum_{j=1}^{K}\sum_{i=1}^{N} \mid X_{ki} - C_k \mid^2 \tag{A.11}$$

with X_{ki} being the i^{th} entry in X assigned to the k^{th} partition.

A.12 Average Silhouette Score

Given a matrix $X \in \mathbb{R}^{N \times B}$, divided in k partitions by the KMeans algorithm, the average silhouette score of the provided solution is defined as

$$\text{S} = \frac{1}{N}\sum_{i=1}^{N} \frac{b_{ki} - a_{ki}}{max(a_{ki}, b_{ki})} \tag{A.12}$$

with a_{ki} being mean distance between i^{th} entry in X assigned to the k^{th} partition and all other entries in k and b_{ki} the mean distance to the next nearest partition.

A.13 Dropout Regularization

Given parameters $\theta \in \mathbb{R}^z$ the dropout regularization is defined as

$$\begin{aligned} dropout(\theta) &= W \cdot diag(d) \\ d &= \{d_1, \ldots, d_z\} \\ d_i &\sim Bernoulli(p) \end{aligned} \tag{A.13}$$

with p being the probability of a parameter being dropped.

A.14 One Dimensional Spatial Dropout Regularization

Given a sequence of weight matrices $W_{t_1:T} \in \mathbb{R}^{T \times z}$ the one dimensional spatial dropout regularization is defined as

$$\begin{aligned} dropout(W) &= W_{t:} \cdot d \\ d &= I^{z \times z} \cdot d_t \\ d_t &\sim Bernoulli(p) \end{aligned} \tag{A.14}$$

with p being the probability of an entire step in the sequence of parameters being dropped.

A.15 Batch Normalization Regularization

Given an embedding $H \in \mathbb{R}^{N \times z}$ the batch normalization regularization is defined as

$$BatchNorm(H) = \frac{h - \mu_b}{\sqrt{(\sigma_b)^2 + \varepsilon}} \tag{A.15}$$

with μ_b and σ_b being respectively the column-wise mean and standard deviation of the embedding H and ε a small constant for avoiding division by zero.

A.16 Ridge Regularization

Given parameters θ the ridge regularization function $l2$ is defined as

$$l2(\theta) = \lambda \sum_{n=1}^{N} \theta_n^2 \tag{A.16}$$

with λ being a a constant controlling the strength of regularization.

A.17 Lasso Regularization

Given parameters θ the ridge regularization function $l1$ is defined as

$$l1(\theta) = \lambda \sum_{n=1}^{N} |\theta_n| \tag{A.17}$$

with λ being a a constant controlling the ammount of regularization.

A.18 ElasticNet Regularization

Given parameters θ the ridge regularization function $ElasticNet$ is defined as

$$ElasticNet(\theta) = \lambda(\frac{1-\alpha}{2} l2(\theta) + \alpha l1(\theta)) \tag{A.18}$$

with λ and α being constants controlling respectively the global strength of regularization and the contribution of the $l1$ and $l2$ terms to the total ammount of regularization.

A.19 Fully Connected Operation

Given an input matrix $X \in \mathbb{R}^{N \times h}$, the fully connected operation carried out by L layers feedforward neural network can be defined as

$$\begin{aligned} H_0 &= X \\ H_1 &= \phi(W_1^\top H_0 + b_1) \\ H_2 &= \phi(W_2^\top H_2 + b_2) \\ &\dots \\ H_L &= \phi(W_L^\top H_{L-1} + b_L) \end{aligned} \tag{A.19}$$

with ϕ being a non-linear function, $\{W_1, \dots, W_N\}$ a set of learnable weights matrices of shape $W_l \in \mathbb{R}^{H_{l-1} \times H_l}$ and $\{b_1, \dots, b_N\}$ a set of learnable biases vectors of shape $b_l \in \mathbb{R}^{h_l}$.

A.20 One-Hot Encode Operation

Given an input set of numerical indices $X = \{1, 2, \dots, N\}$, the one-hot encode operation is defined as

$$1_X(x_i) = \begin{cases} 1, & \text{if } x_i \in X \\ 0, & \text{otherwise} \end{cases} \tag{A.20}$$

A.21 Embedding Operation

Given an input set of numerical indices $X = \{1, 2, \ldots, N\}$, the embedding operation is defined as

$$h = \phi(W_{X,*}) \tag{A.21}$$

with ϕ being a non-linear function and W an $\mathbb{R}^{N \times z}$ learnable weights matrix.

A.22 LSTM Cell Operation

Given an input set of multivariate time series $X_{t_1:T} \in \mathbb{R}^{N \times T \times B}$ with N being the number of series, T the length of the series and B the dimensionality of the series. An LSTM cell operation, applied recursively along the temporal dimension for each $x_{*,t,*} \in X$, is defined as

$$\begin{aligned}
f_t &= \sigma(W_{xf}^\top x_{*,t,*} + W_{hf}^\top H_{t-1} + b_f) \\
i_t &= \sigma(W_{xi}^\top x_{*,t,*} + W_{hi}^\top H_{t-1} + b_i) \\
o_t &= \tanh(W_{xo}^\top x_{*,t,*} + W_{ho}^\top H_{t-1} + b_o) \\
\widehat{c_t} &= \sigma(W_{xc}^\top x_{*,t,*} + W_{hc}^\top H_{t-1} + b_c) \\
c_t &= f_t \times c_{t-1} + i_t \times \widehat{c_t} \\
H_t &= o_t \times \tanh(c_t)
\end{aligned} \tag{A.22}$$

with σ being the sigmoid function, tanh the hyperbolic function, W_{xf}, W_{xi}, W_{xo}, $W_{xc} \in \mathbb{R}^{B \times h}$ and W_{hf}, W_{hi}, W_{ho}, W_{hc}, $W_{hf} \in \mathbb{R}^{h \times h}$ a set of learnable weights matrices, b_f, b_i, b_o, $b_c \in \mathbb{R}^h$, a set of learnable biases, c_t the value at time t of the conveyor belt matrix and h_t the value at time t of the hidden state matrix.

A.23 Time Distributed Operation

Given an input tensor $X \in \mathbb{R}^{N \times T \times h}$, a function f with parameters θ can be applied along the temporal dimension for each $x_{*,t,*} \in X$ as

$$H_t = f(x_{*,t,*}; \theta) \tag{A.23}$$

it has to be noted that this has the effect of sharing θ across all $t \in T$ but without temporal conditioning (e.g. like the LSTM operation presented in Appendix A.22 does).

A.24 One Dimensional Convolution Operation

Given an input set of multivariate time series $X_{t_1:T} \in \mathbb{R}^{N \times T \times B}$ with N being the number of series, T the length of the series and B the dimensionality of the series. A One Dimensional Convolution operation, applied recursively along the temporal dimension for each $x_{*,t,*} \in X$, is defined as

$$H_t = \phi(W_K^\top x_{*,t:t+k,*}) \tag{A.24}$$

with ϕ being a non-linear function and K a $K \in \mathbb{R}^{k \times z}$ learnable weights matrix.

A.25 One Dimensional Global Average Pooling Operation

Given an input set of multivariate time series $X_{t_1:T} \in \mathbb{R}^{N \times T \times B}$ with N being the number of series, T the length of the series and B the dimensionality of the series. A Global Average Pooling Operation, is defined as

$$H_t = \frac{1}{T} \sum_{t=1}^{t} X_{*,t,*} \tag{A.25}$$

Appendix B

Architectures Directed Acyclic Graphs

We will report here the details of the Directed Acyclic Graphs (DAGs) describing the computational graphs of each of the ANNs architectures used in this thesis. The graphs are generated by the Keras library plotting functionality [180].

B.1 Dynamic Prediction of Future Behavioural Intensity

The DAGs presented in this section refer to the architectures used in the second iteration of the model building presented in section 4.3.

B.1.1 ENet Architecture

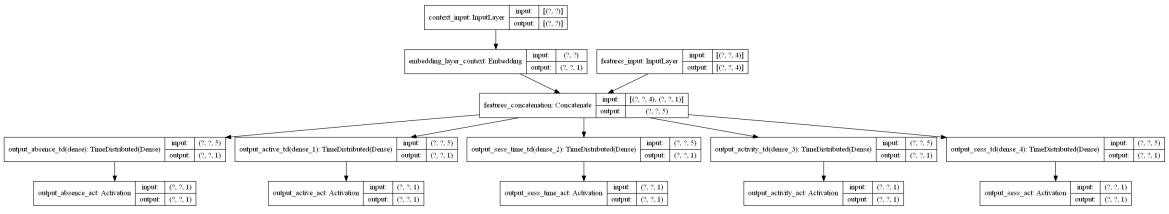


Figure B.1: Directed acyclic graph representation of the ElasticNet architecture used as a comparison in section 4.3

B.1.2 MLP Architecture

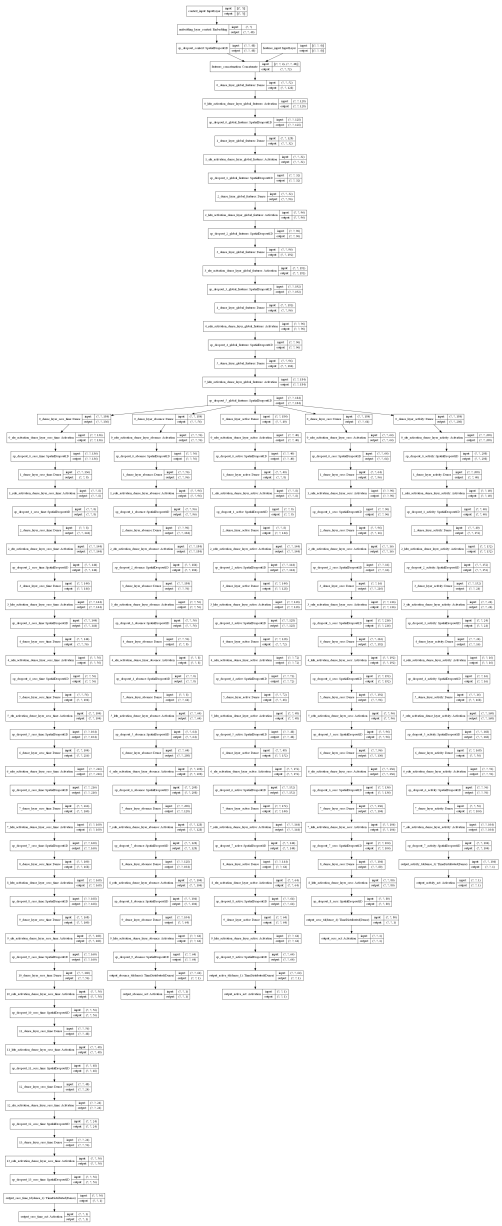


Figure B.2: Directed acyclic graph representation of the MLP architecture used as a comparison in section 4.3

B.1.3 RNN Architecture

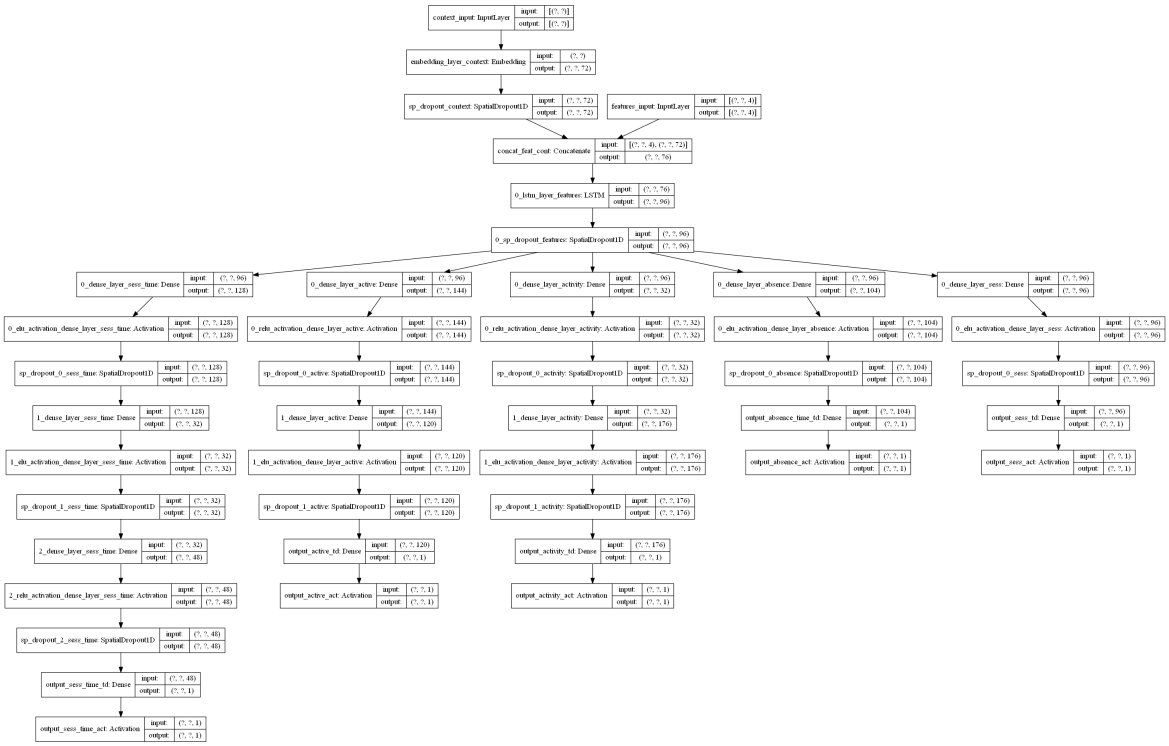


Figure B.3: Directed acyclic graph representation of the RNN architecture proposed in section 4.3

B.2 Dynamic Prediction of Future Behavioural Intensity with Environmental and Game Covariates

The DAGs presented in this section refers to the architectures used in the third and last iteration of the model building presented in section 4.4.

B.2.1 MLP Environment-Events Architecture

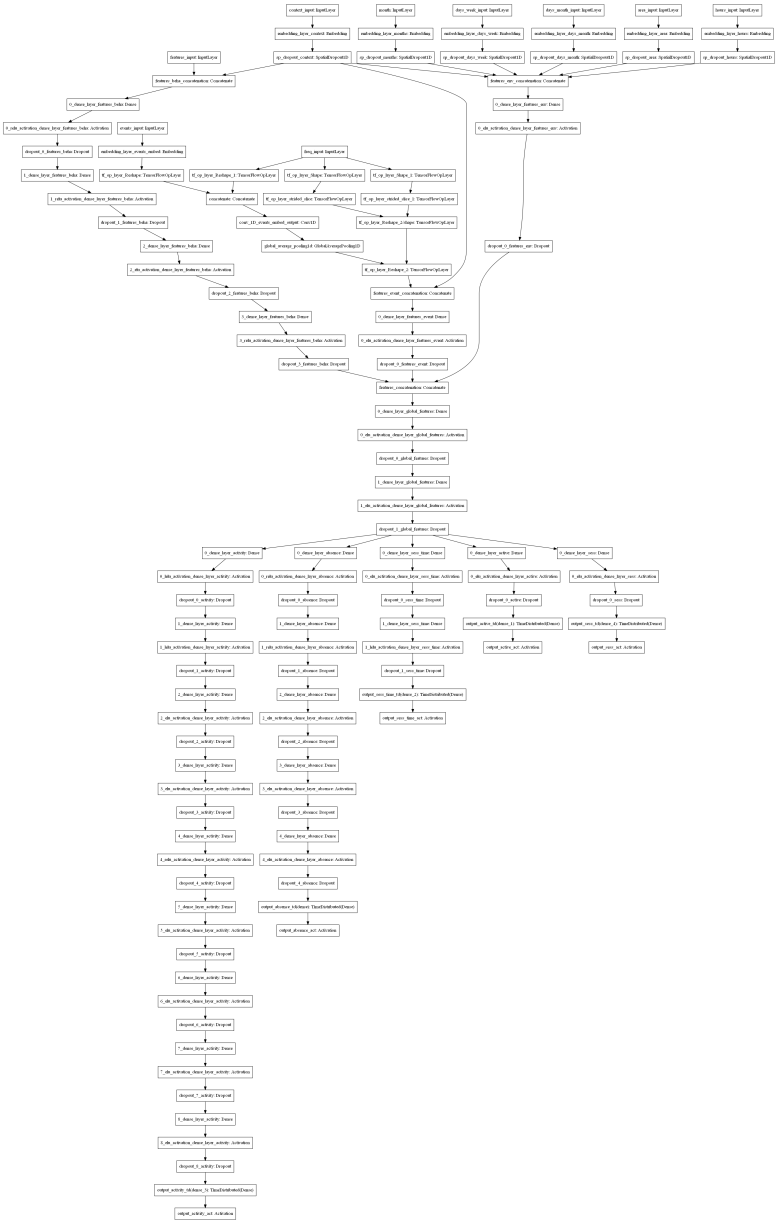


Figure B.4: Directed acyclic graph representation of the MLP architecture used as a comparison in section 4.4

B.2.2 RNN Architecture

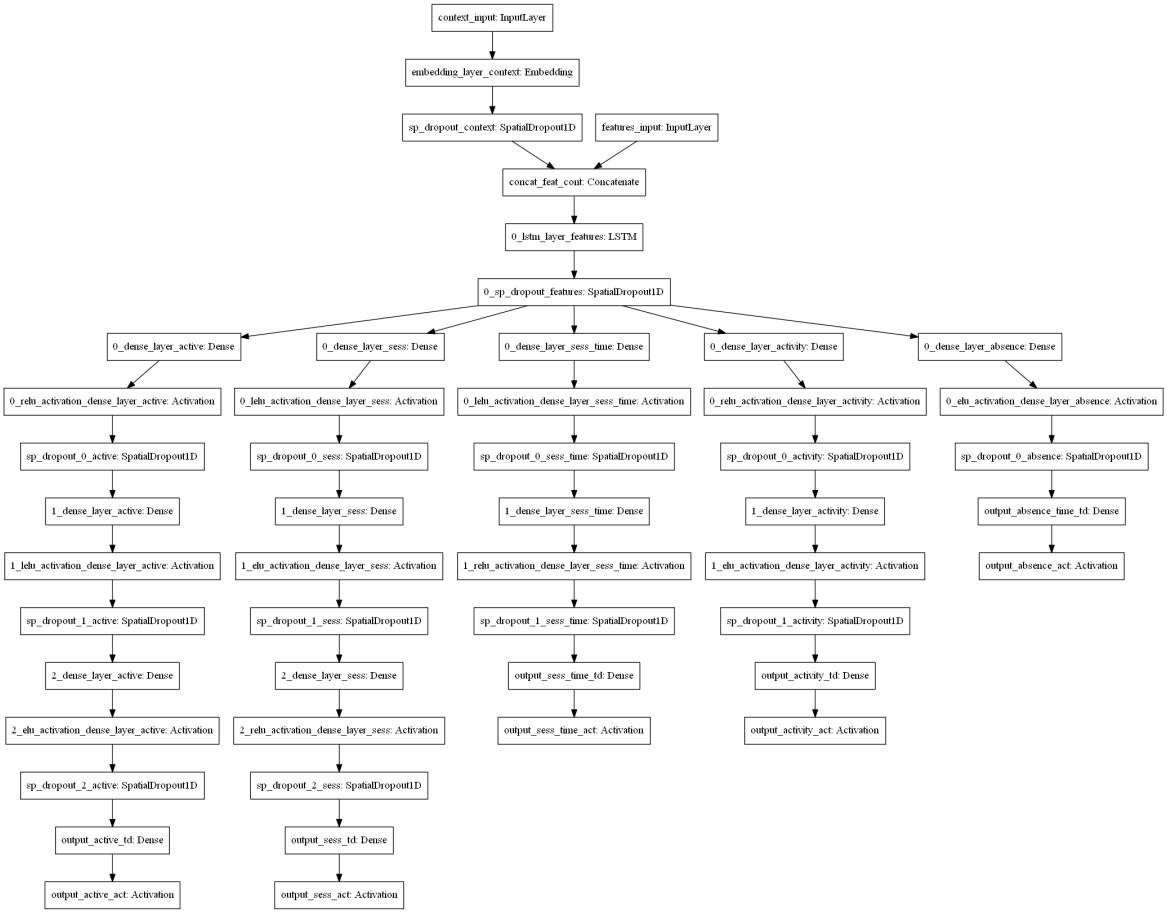


Figure B.5: Directed acyclic graph representation of the RNN architecture used as a comparison in section 4.4

B.2.3 RNN Environment Architecture

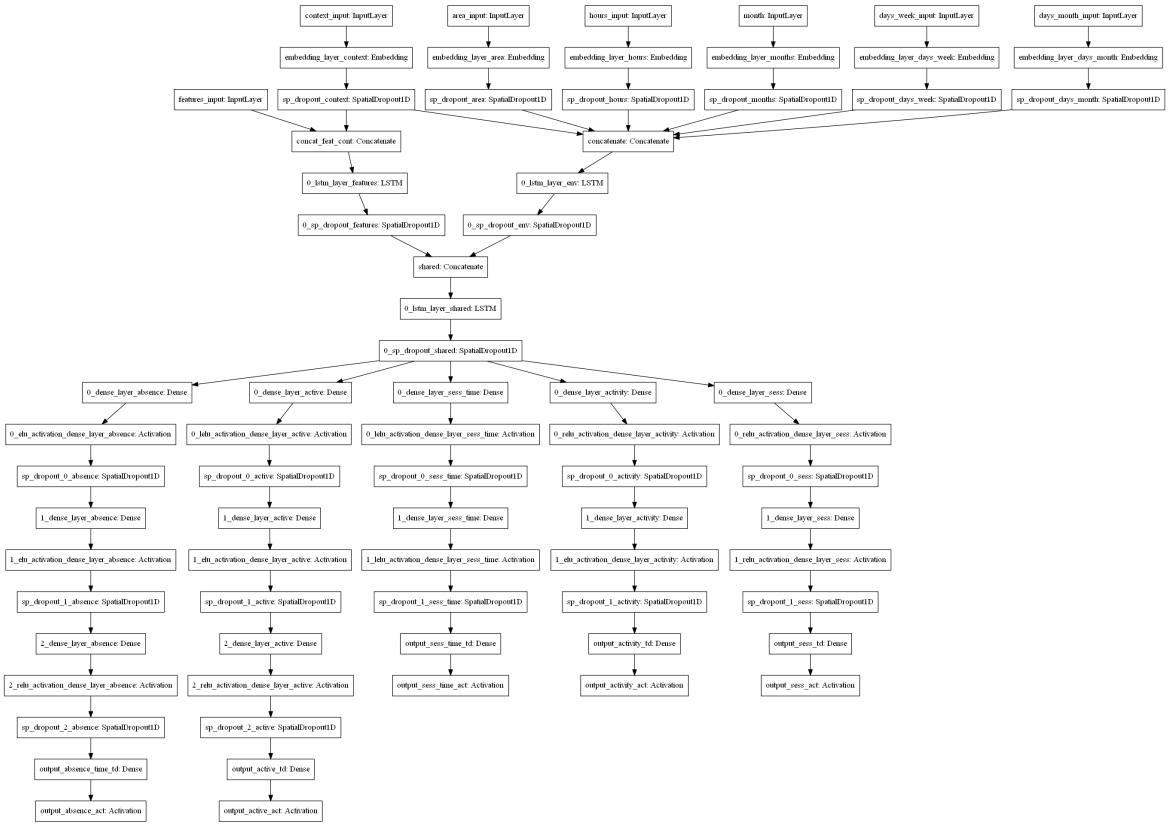


Figure B.6: Directed acyclic graph representation of the RNN Environment architecture used as a comparison in section 4.4

B.2.4 RNN Events Architecture

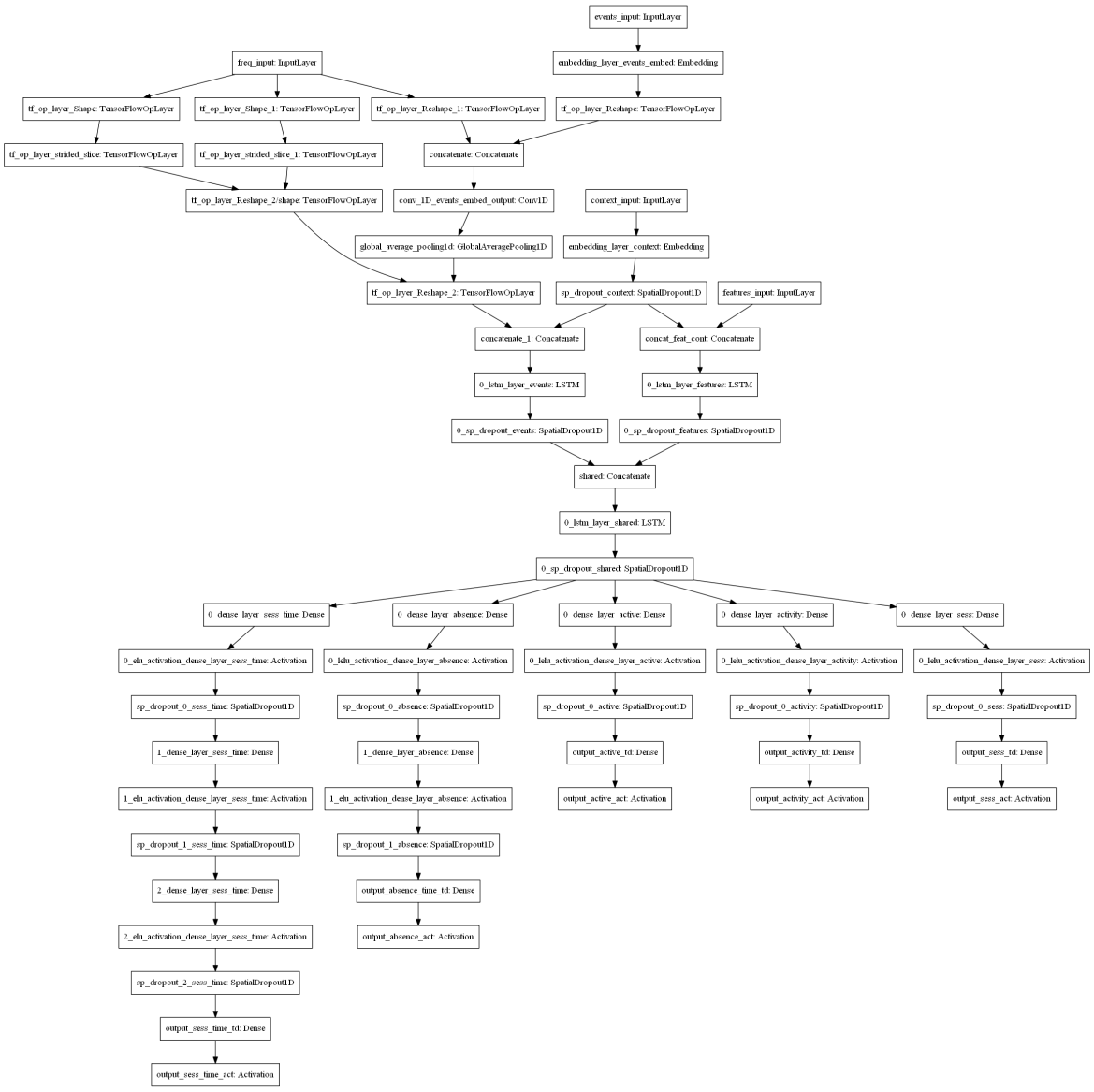


Figure B.7: Directed acyclic graph representation of the RNN Events architecture used as a comparison in section 4.4

B.2.5 RNN Environment-Events Architecture

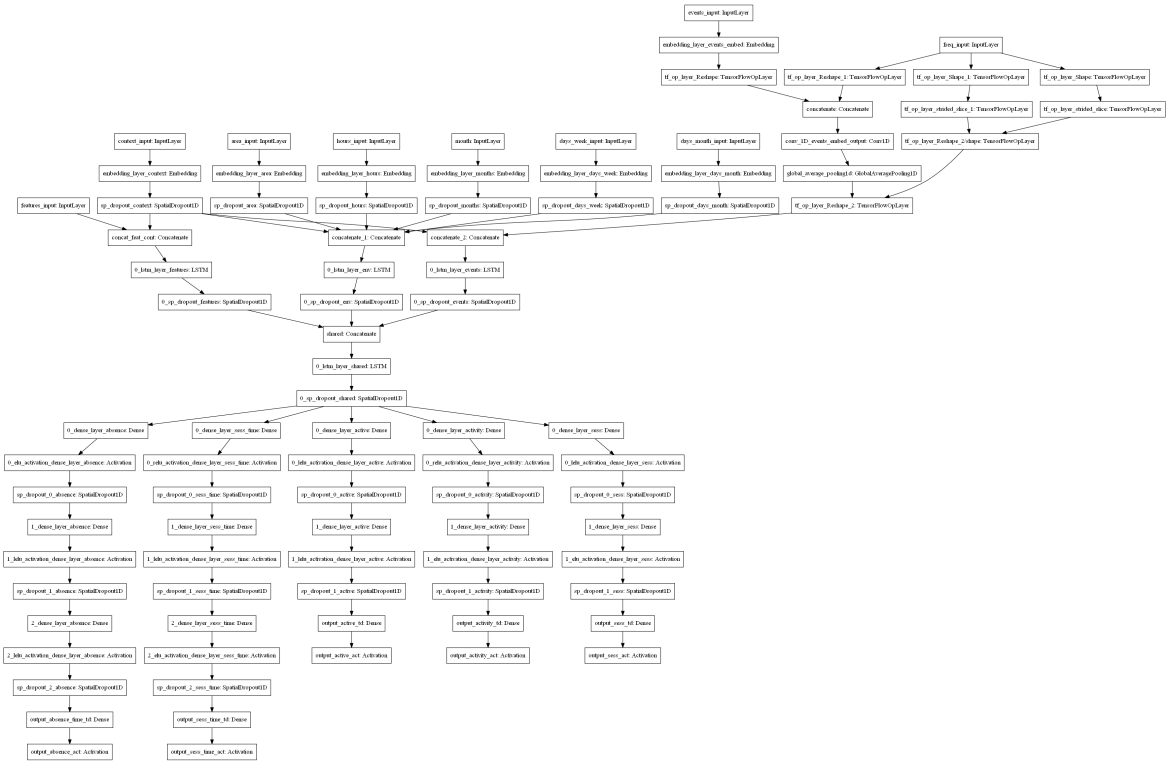


Figure B.8: Directed acyclic graph representation of the RNN Environment-Events architecture proposed in section 4.4

Appendix C

Ancillary Model Performance Analyses

C.1 Bayesian Multi-Level Model Definition

In order to assess in a more robust and reliable way the results of our models comparison experiments, along with the conventional frequentist analyses presented in chapter 4 we carried out an additional set of statistical analyses using a bespoke Bayesian multi-level model. We opted for a time-varying random intercept accounting for the impact of game context and time and a fixed slope for estimating the effect of model architecture. We adopted a partial pooling strategy for the time-varying random intercept as we could safely assume exchangeability of parameters for the various game contexts [177]. The model had the following formulation:

$$
\begin{aligned}
SMAPE_{i,j,t} &\sim StudentT(\nu, \mu, \sigma) \\
\nu &\sim Gamma(\alpha = 2, \beta = 0.1) \\
\sigma &\sim HalfCaucy(\beta = 1) \\
\mu &= \beta_{jt} + \alpha_i \\
\alpha_i &\sim Normal(\mu = 0, \sigma = 1) \\
\beta_{jt} &= \alpha_j^{\top} B_{t,*} \\
\alpha_j &\sim Normal(\mu_j, \sigma_j) \\
\mu_j &\sim Normal(\mu = 1, \sigma = 1) \\
\sigma_j &\sim HalfCauchy(\beta = 1)
\end{aligned}
\tag{C.1}
$$

with $B \in \mathbb{R}^{T \times dof}$ being a cubic B-spline matrix with $dof = 6$ as provided by the python library Patsy [233]. In this case j indicates the j^{th} game context, i the i^{th} model and t the t^{th} element in the considered sequence of SMAPE values. The $StudentT$ likelyhood allowed for an estimation of μ robust to outliers. The values for the the $Gamma$ distribution parameters were chosen according

to a sensible default as mentioned in [234]. The choice of priors for α_i were made so as to have a small regularizing effect: the priors assume each model to have an impact on the base SMAPE of $\sim \pm 2\%$ but with the expectancy of having no effect. The same model was fit first collapsing all the target in a single overall performance metric and then separately on each of the 5 targets. In order to estimate parameters for the model we used mean field approximation [235]. The model was implemented using the python library PyMC3 [236]. As mean field approximation does not allow conventional convergence checks for Bayesian model fitting, we relied exclusively on visual inspection of the marginal probability distributions. Comparison between different models was carried out by inspecting the marginal distribution of differences in the μ parameter, highlighting a region of practical significance (ROPE) equals to $\{-0.1, 0.1\}$. This means that we deemed worth of attention only differences in μ greater than 0.1% SMAPE.

C.2 Dynamic Prediction of Future Behavioural Intensity

We report here the results of fitting the Bayesian model to the performance data obtained from the experimental task described in section 4.3.

C.2.1 Targets Collapsed

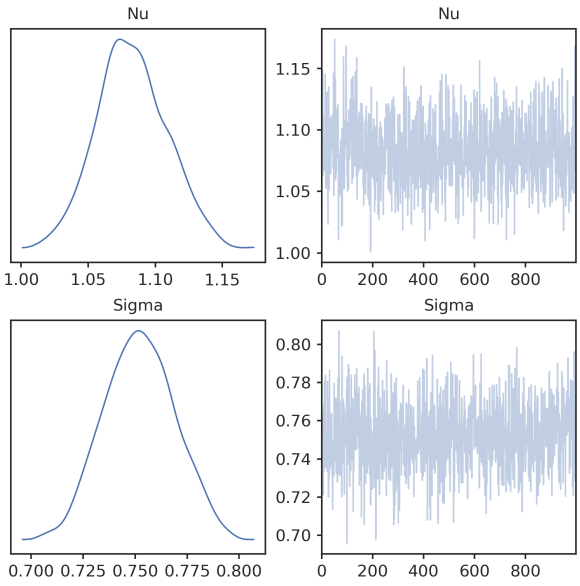


Figure C.1: Marginal distributions for the parameters ν, σ estimated by the model fitted for the Future Absence target.

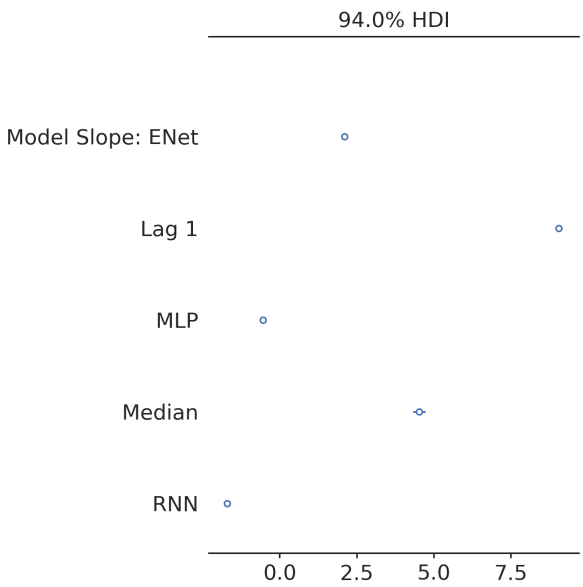


Figure C.2: Forest plot of the marginal distributions for the parameter α (i.e. model slope) estimated by the model fitted for the Future Absence target.

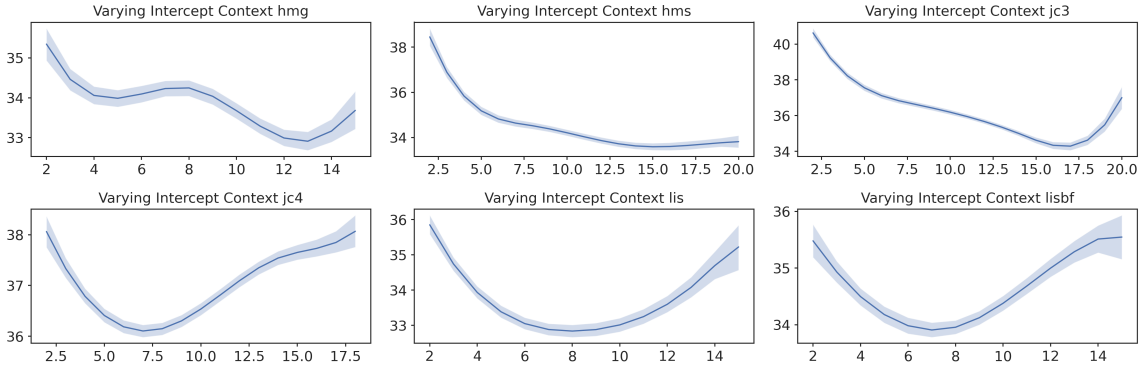


Figure C.3: Time varying random intercept estimated by the model fitted for the Future Absence target.

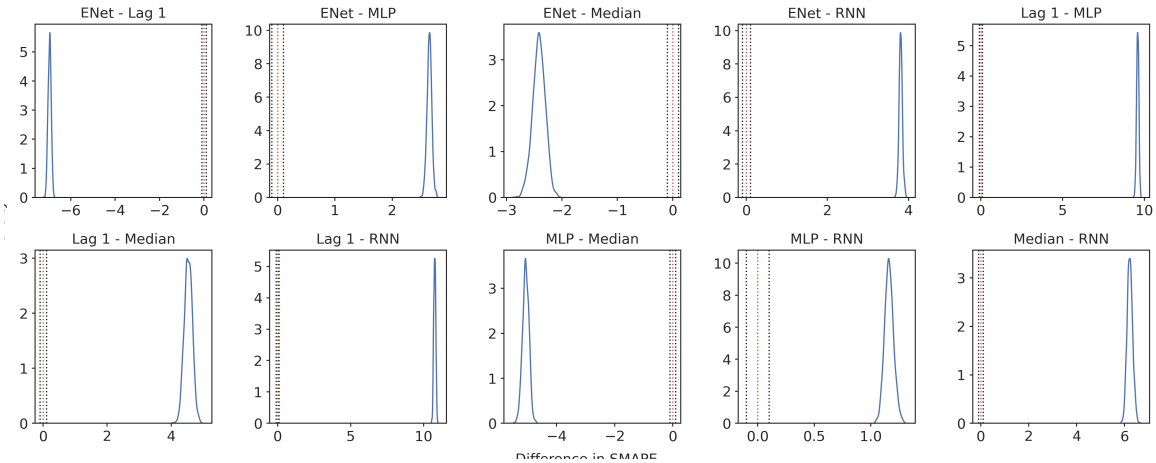


Figure C.4: Pairwise comparisons of the parameter α (i.e. model slope) estimated by the model fitted for the Future Absence target.

C.2.2 Future Absence

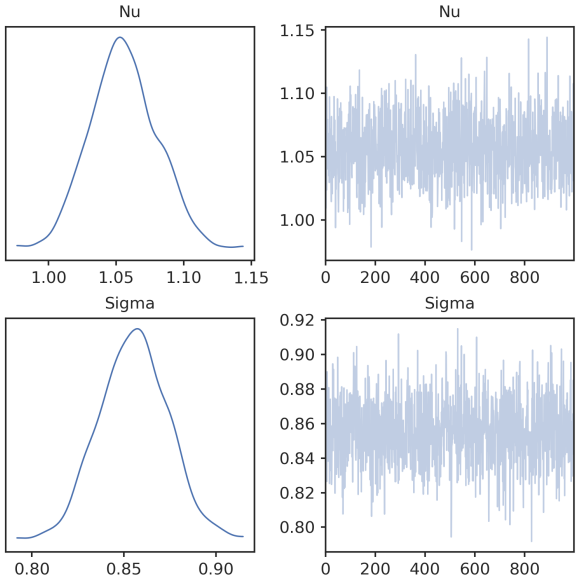


Figure C.5: Marginal distributions for the parameters ν, σ estimated by the model fitted for the Future Absence target.

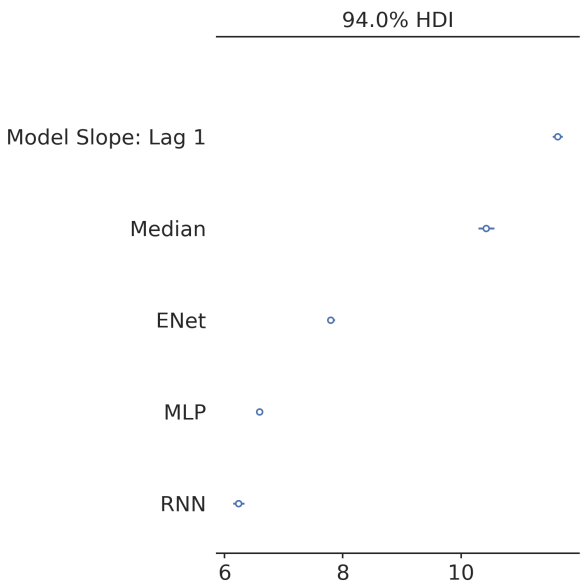


Figure C.6: Forest plot of the marginal distributions for the parameter α (i.e. model slope) estimated by the model fitted for the Future Absence target.

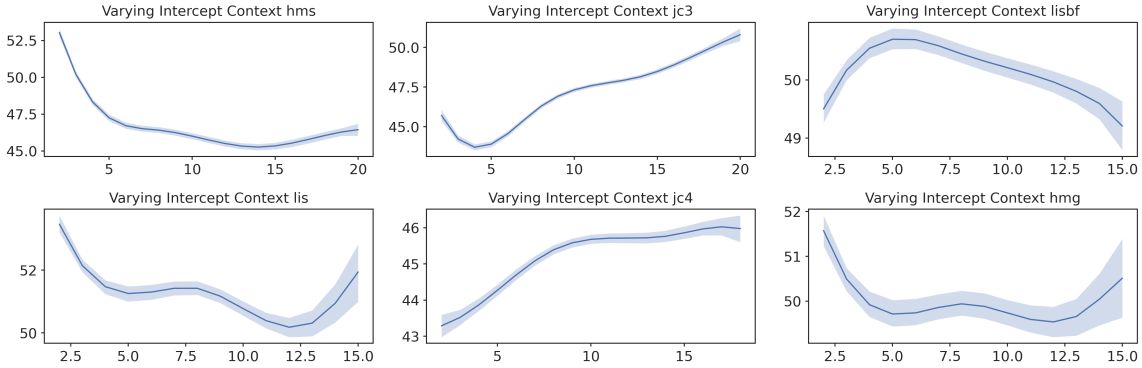


Figure C.7: Time varying random intercept estimated by the model fitted for the Future Absence target.

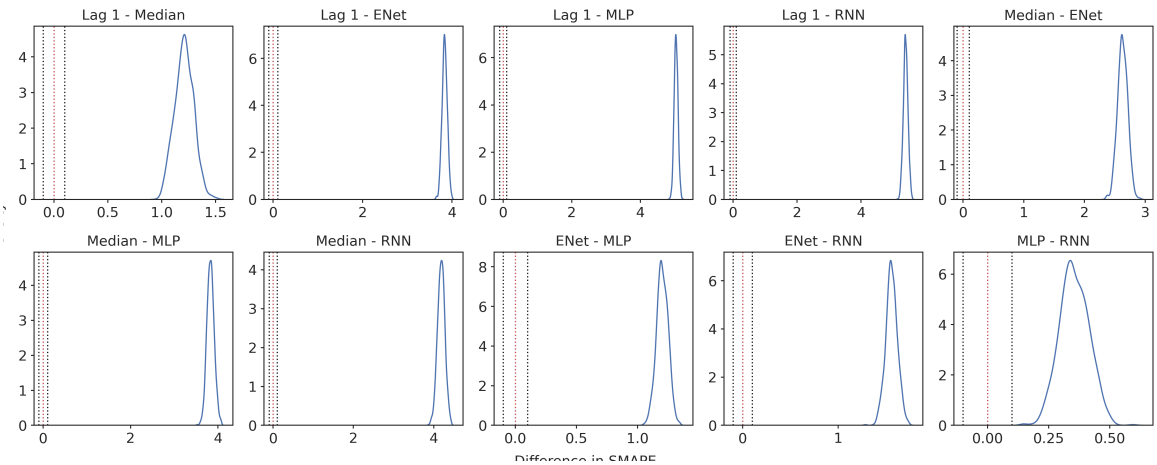


Figure C.8: Pairwise comparisons of the parameter α (i.e. model slope) estimated by the model fitted for the Future Absence target.

C.2.3 Future Active Time

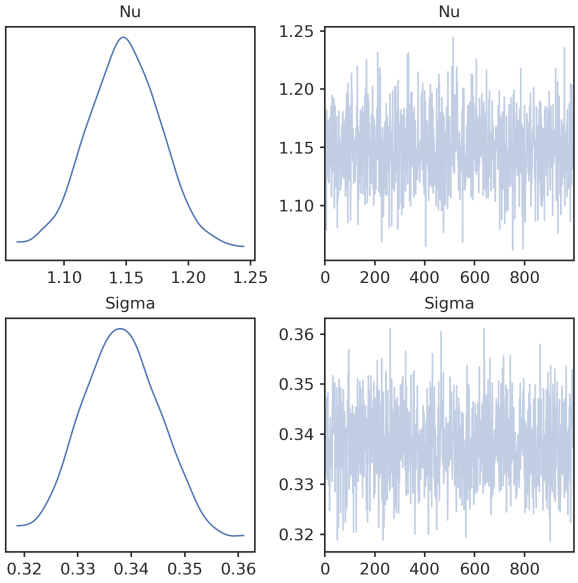


Figure C.9: Marginal distributions for the parameters ν, σ estimated by the model fitted for the Future Active Time target.

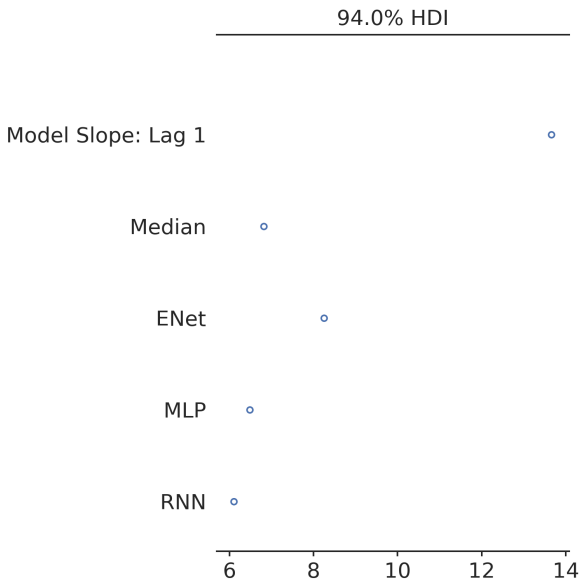


Figure C.10: Forest plot of the marginal distributions for the parameter α (i.e. model slope) estimated by the model fitted for the Future Active Time target.

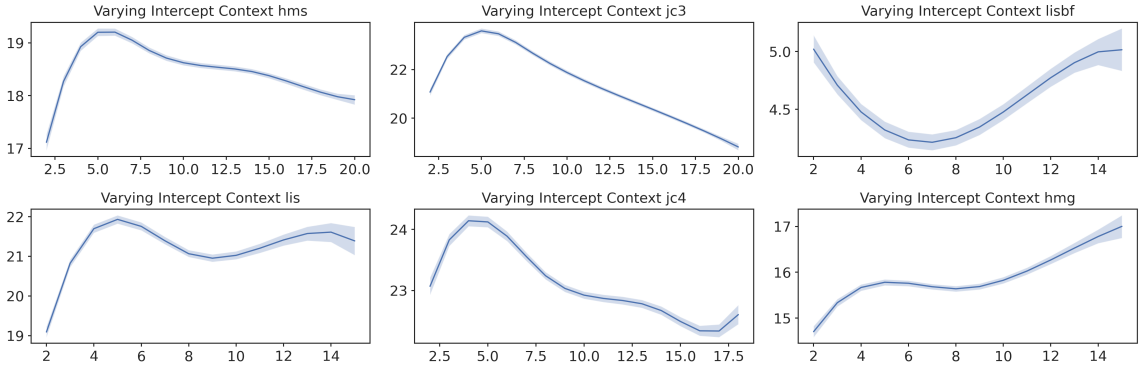


Figure C.11: Time varying random intercept estimated by the model fitted for the Future Active Time target.

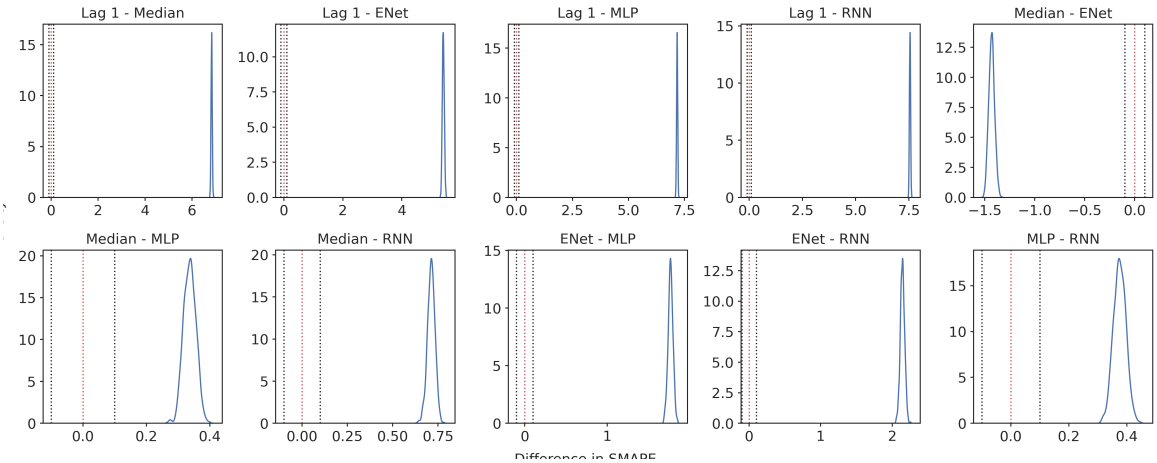


Figure C.12: Pairwise comparisons of the parameter α (i.e. model slope) estimated by the model fitted for the Future Active Time target.

C.2.4 Future Session Time

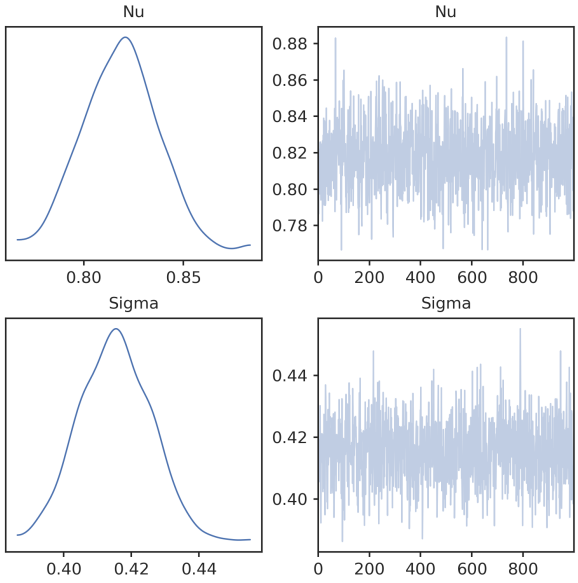


Figure C.13: Marginal distributions for the parameters ν, σ estimated by the model fitted for the Future Session Time target.

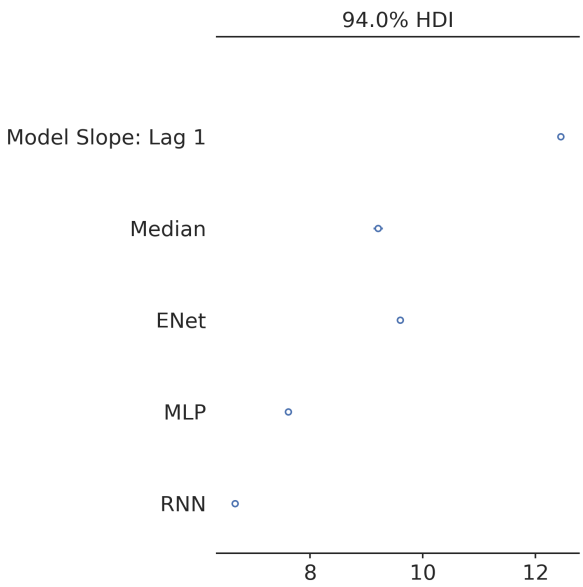


Figure C.14: Forest plot of the marginal distributions for the parameter α (i.e. model slope) estimated by the model fitted for the Future Session Time target.

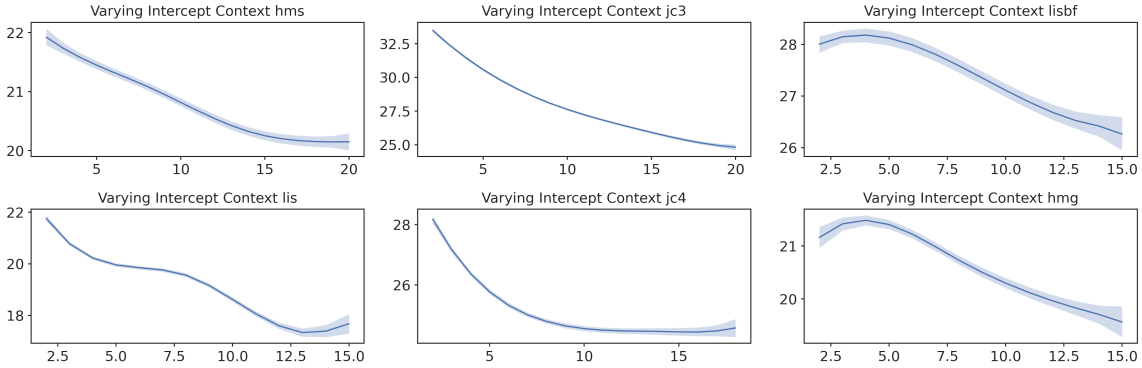


Figure C.15: Time varying random intercept estimated by the model fitted for the Future Session Time target.

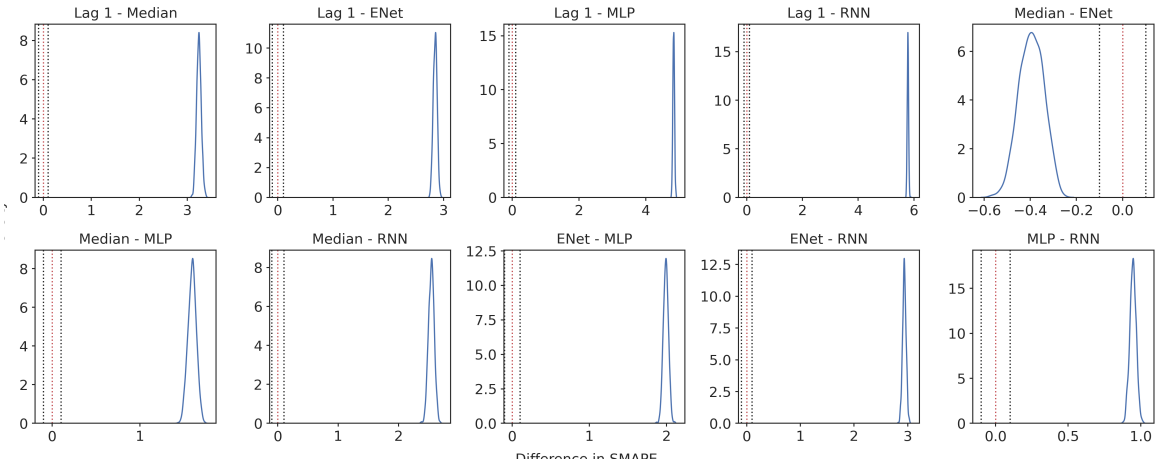


Figure C.16: Pairwise comparisons of the parameter α (i.e. model slope) estimated by the model fitted for the Future Session Time target.

C.2.5 Future Session Activity

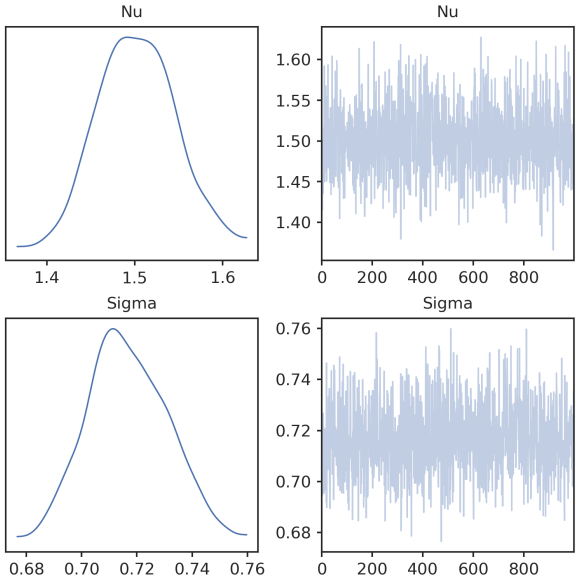


Figure C.17: Marginal distributions for the parameters ν, σ estimated by the model fitted for the Future Session Activity target.

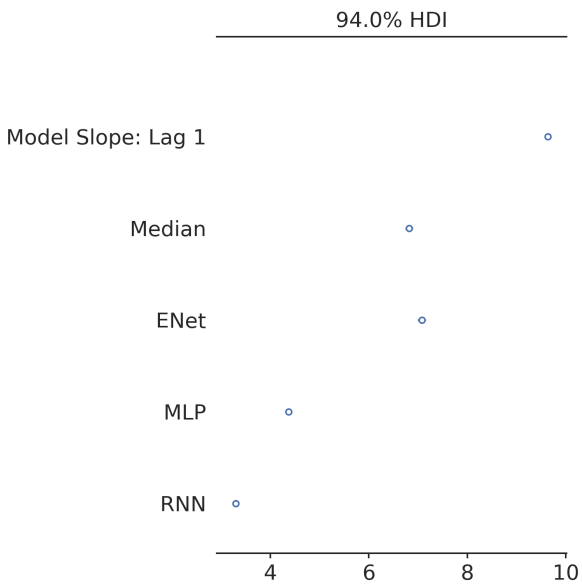


Figure C.18: Forest plot of the marginal distributions for the parameter α (i.e. model slope) estimated by the model fitted for the Future Session Activity target.

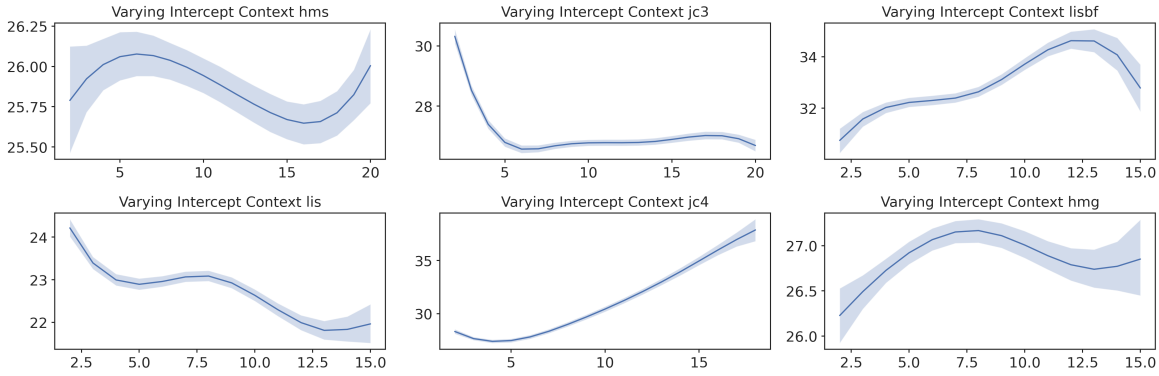


Figure C.19: Time varying random intercept estimated by the model fitted for the Future Session Activity target.

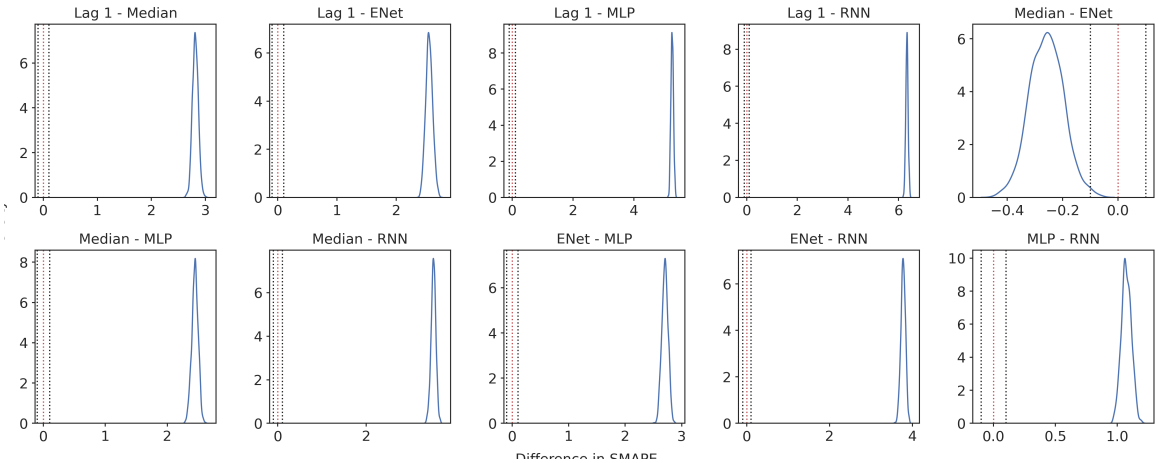


Figure C.20: Pairwise comparisons of the parameter α (i.e. model slope) estimated by the model fitted for the Future Session Activity target.

C.2.6 Future N° Sessions

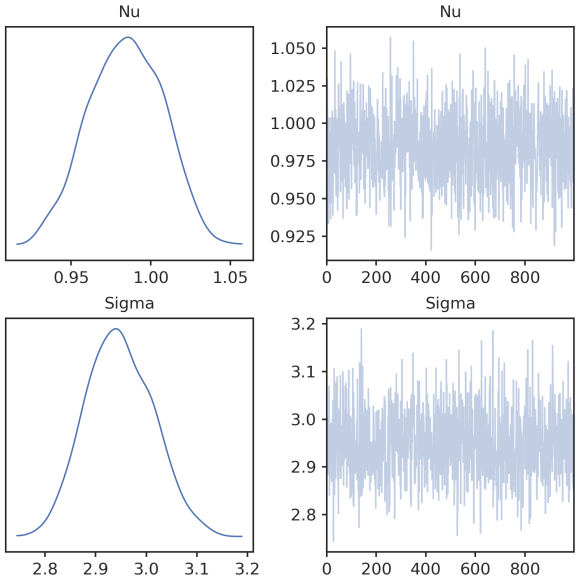


Figure C.21: Marginal distributions for the parameters ν, σ estimated by the model fitted for the Future N°sessions target.

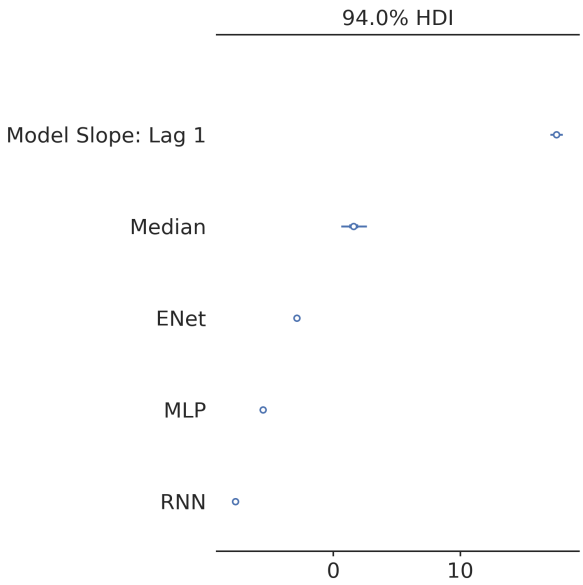


Figure C.22: Forest plot of the marginal distributions for the parameter α (i.e. model slope) estimated by the model fitted for the Future N°sessions target.

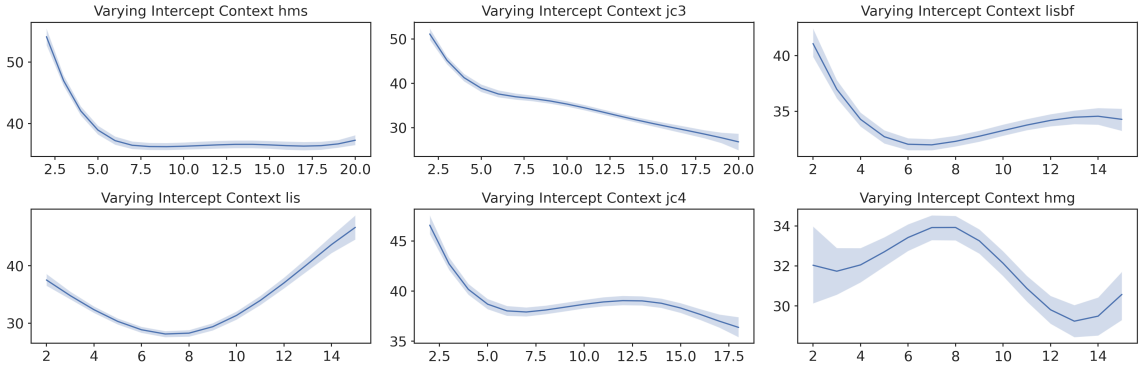


Figure C.23: Time varying random intercept estimated by the model fitted for the Future N°sessions target.

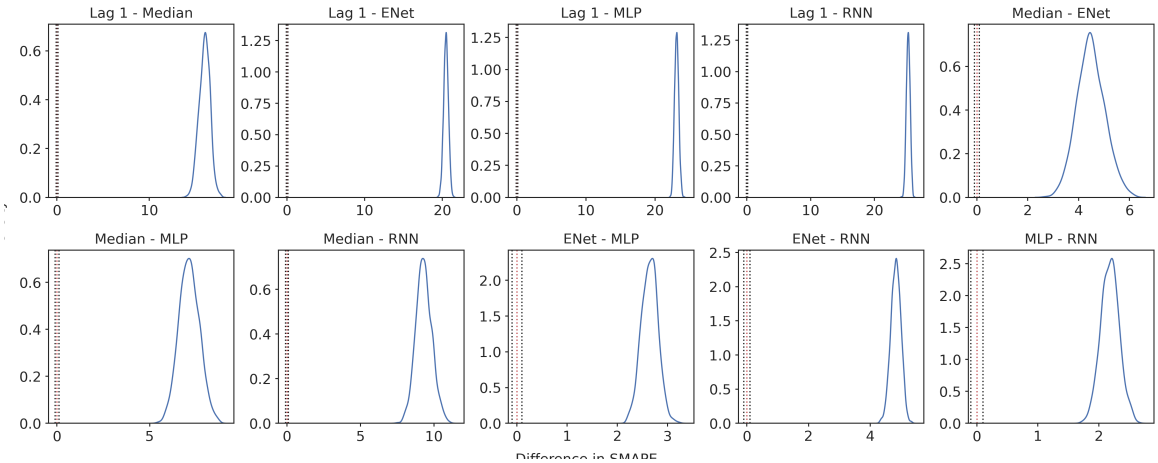


Figure C.24: Pairwise comparisons of the parameter α (i.e. model slope) estimated by the model fitted for the Future N°sessions target.

C.3 Dynamic Prediction of Future Behavioural Intensity with Environmental and Game Covariates

We report here the results of fitting the Bayesian model to the perfromance data obtained from the experimental task described in section 4.4.

C.3.1 Targets Collapsed

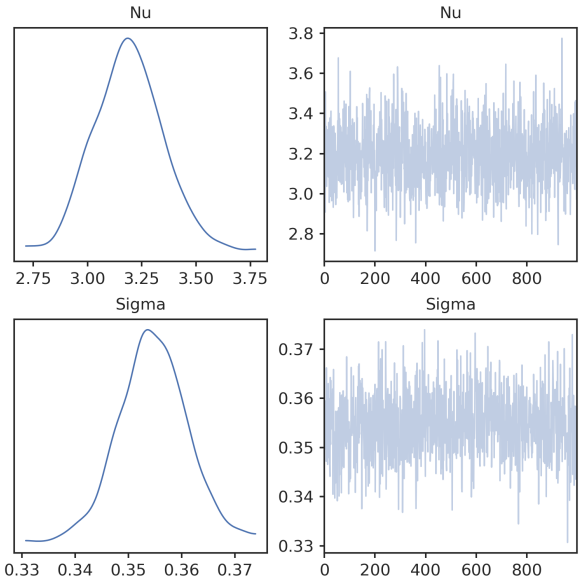


Figure C.25: Marginal distributions for the parameters ν, σ estimated by the model fitted for the Future Absence target.

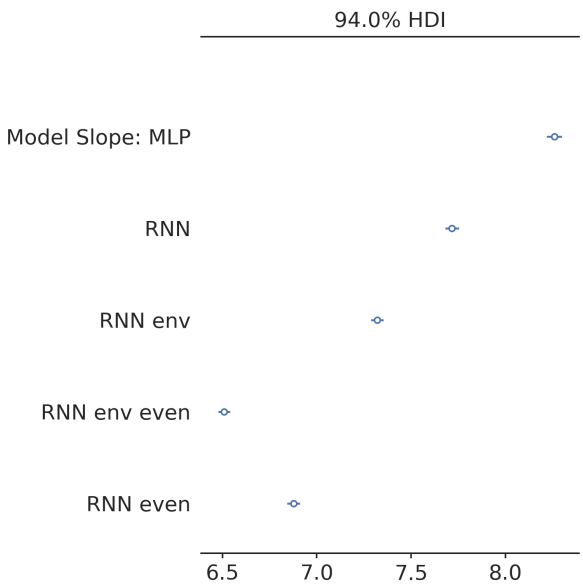


Figure C.26: Forest plot of the marginal distributions for the parameter α (i.e. model slope) estimated by the model fitted for the Future Absence target.

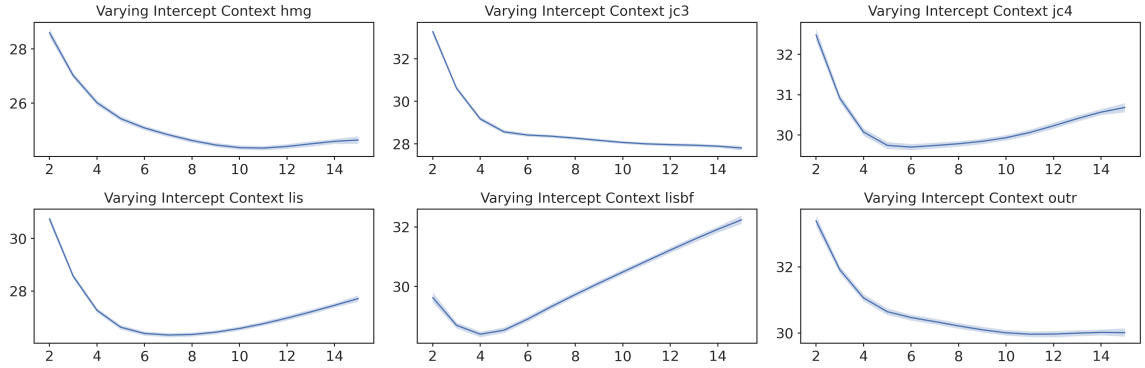


Figure C.27: Time varying random intercept estimated by the model fitted for the Future Absence target.

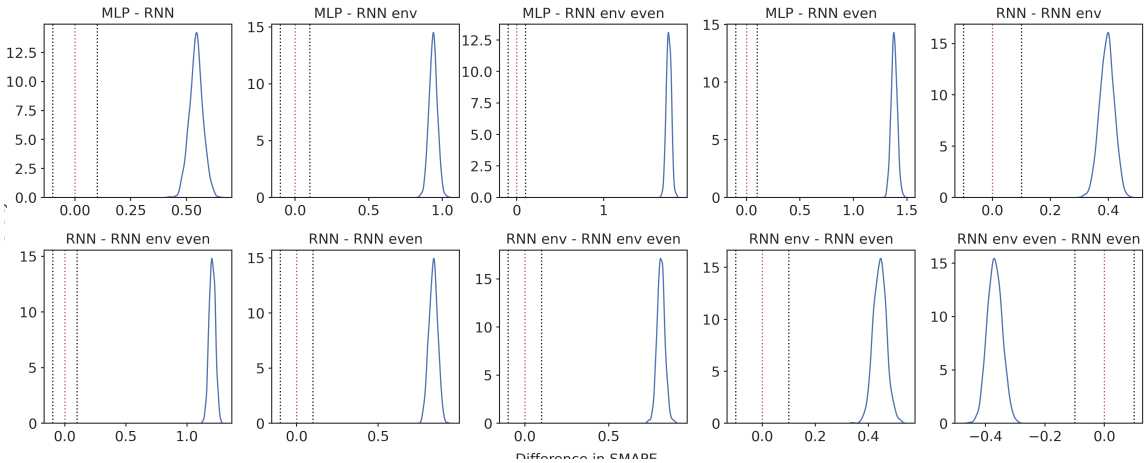


Figure C.28: Pairwise comparisons of the parameter α (i.e. model slope) estimated by the model fitted for the Future Absence target.

C.3.2 Future Absence

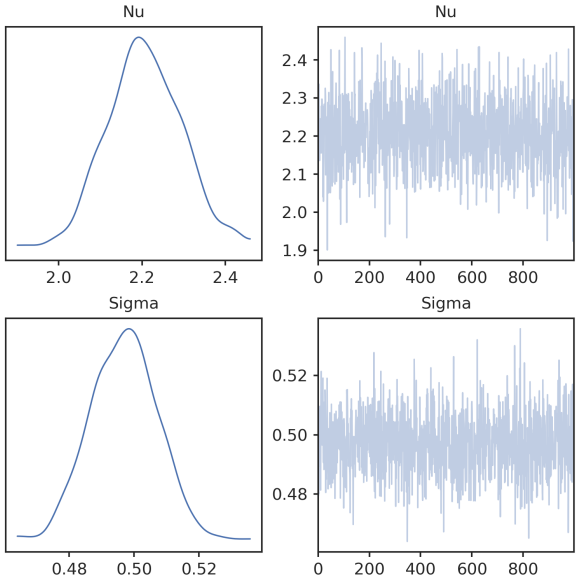


Figure C.29: Marginal distributions for the parameters ν, σ estimated by the model fitted for the Future Absence target.

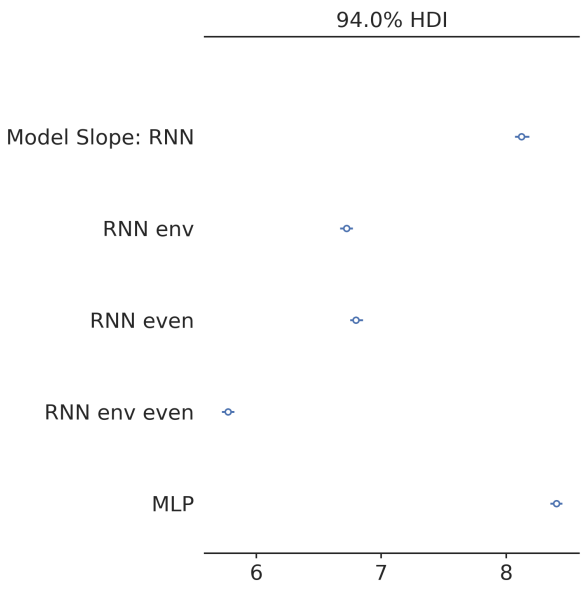


Figure C.30: Forest plot of the marginal distributions for the parameter α (i.e. model slope) estimated by the model fitted for the Future Absence target.

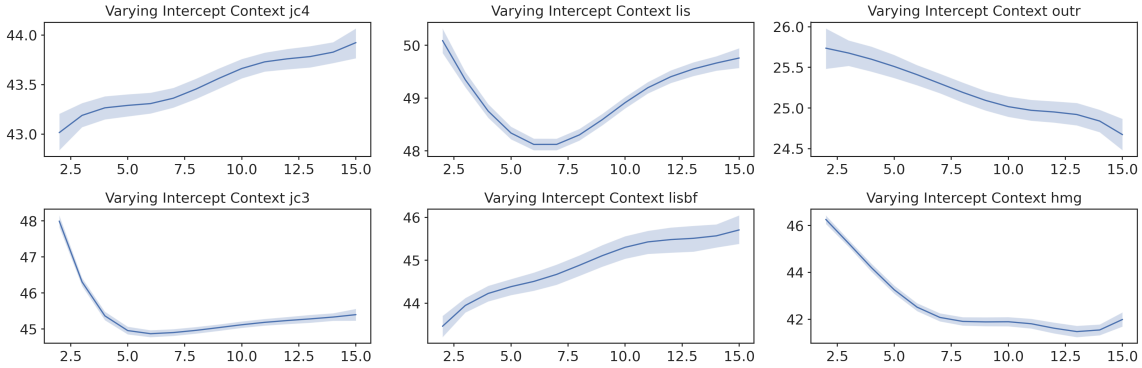


Figure C.31: Time varying random intercept estimated by the model fitted for the Future Absence target.

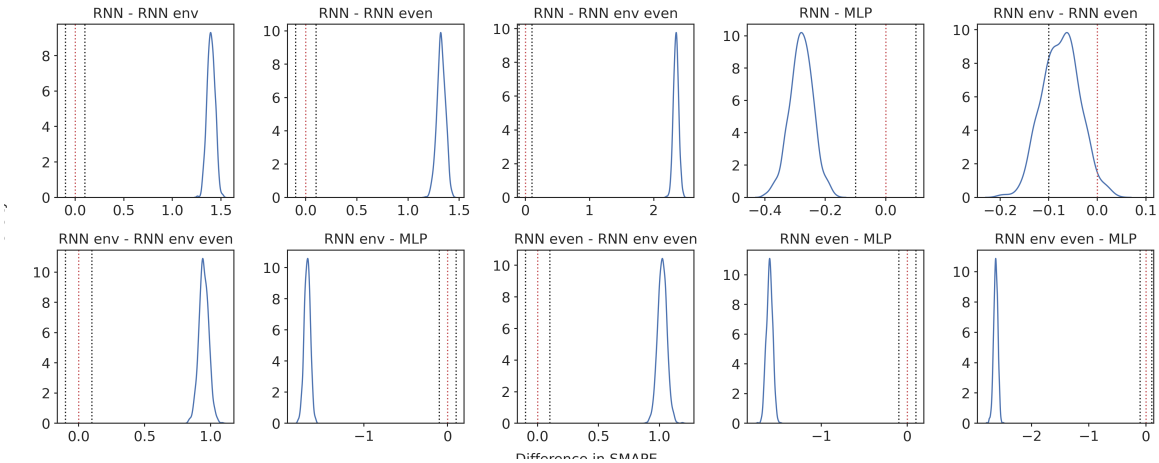


Figure C.32: Pairwise comparisons of the parameter α (i.e. model slope) estimated by the model fitted for the Future Absence target.

C.3.3 Future Active Time

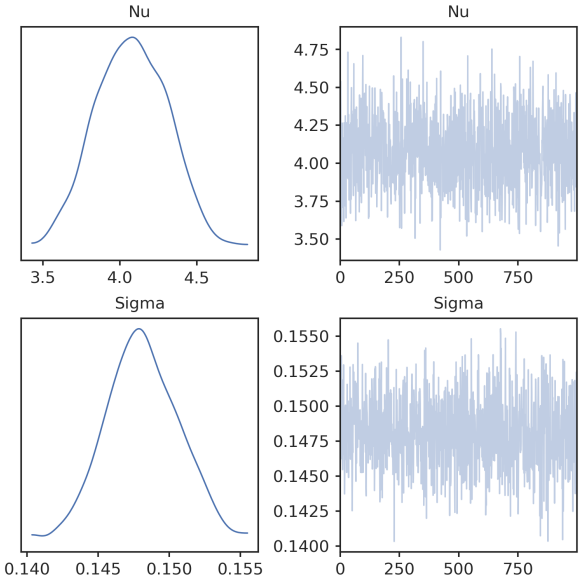


Figure C.33: Marginal distributions for the parameters ν, σ estimated by the model fitted for the Future Active Time target.

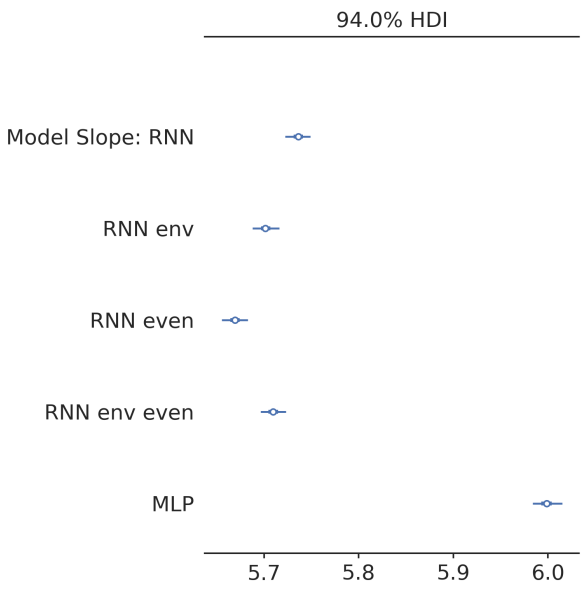


Figure C.34: Forest plot of the marginal distributions for the parameter α (i.e. model slope) estimated by the model fitted for the Future Active Time target.

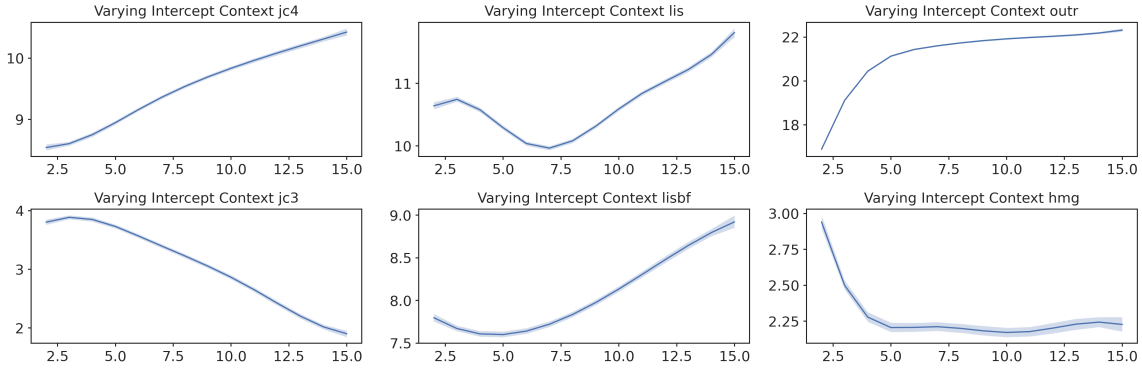


Figure C.35: Time varying random intercept estimated by the model fitted for the Future Active Time target.

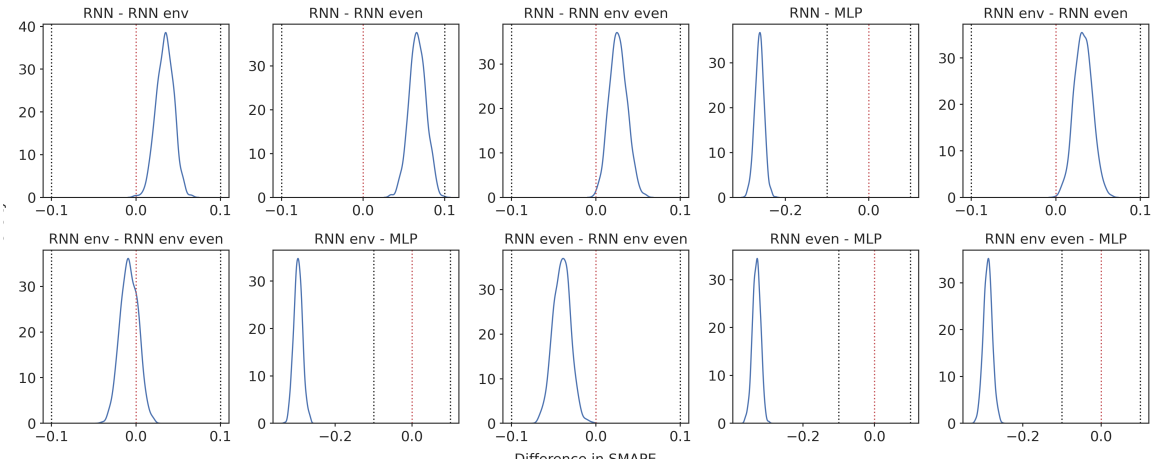


Figure C.36: Pairwise comparisons of the parameter α (i.e. model slope) estimated by the model fitted for the Future Active Time target.

C.3.4 Future Session Time

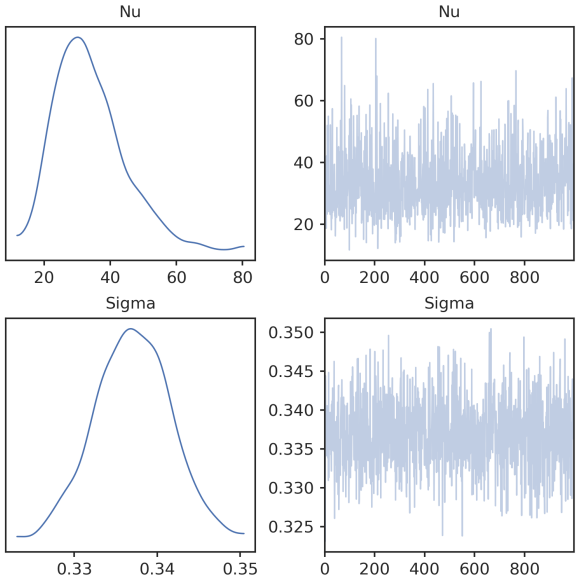


Figure C.37: Marginal distributions for the parameters ν, σ estimated by the model fitted for the Future Session Time target.

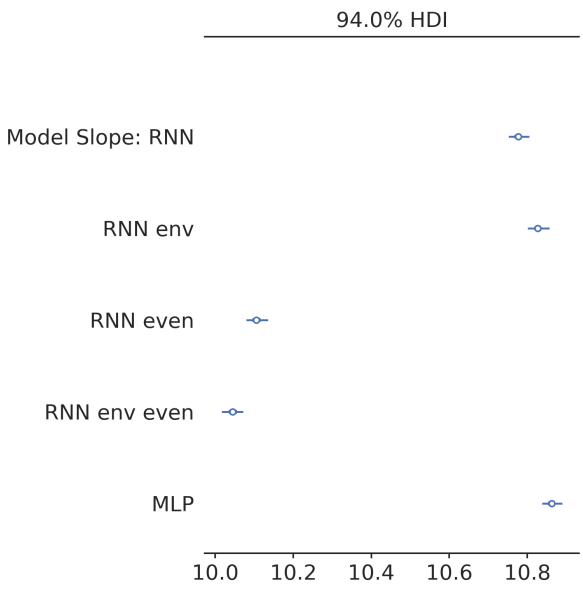


Figure C.38: Forest plot of the marginal distributions for the parameter α (i.e. model slope) estimated by the model fitted for the Future Session Time target.

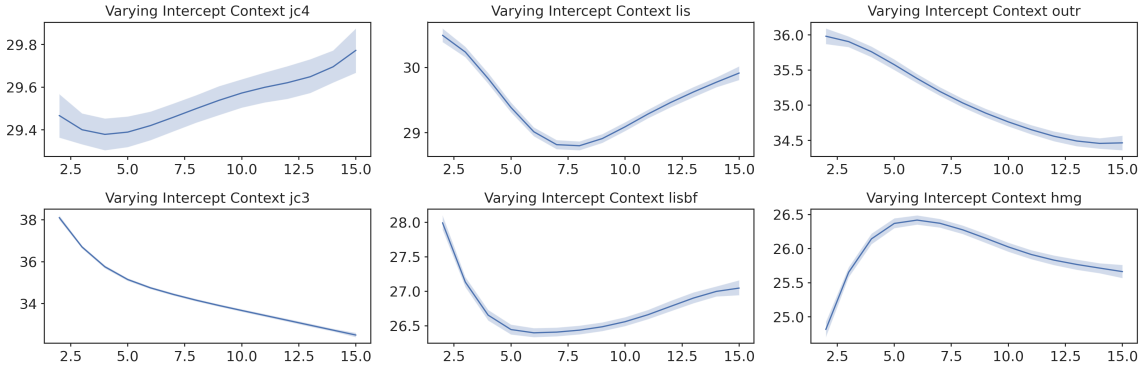


Figure C.39: Time varying random intercept estimated by the model fitted for the Future Session Time target.

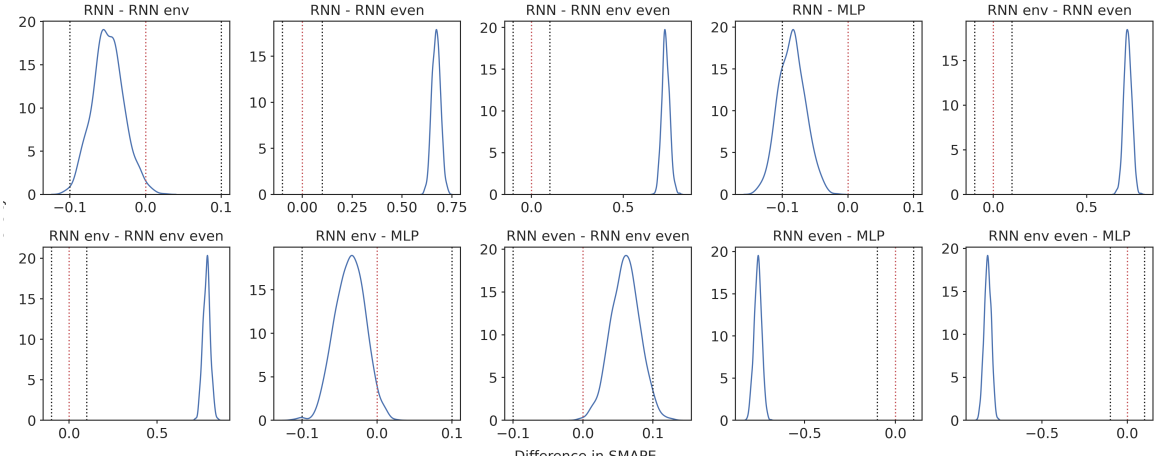


Figure C.40: Pairwise comparisons of the parameter α (i.e. model slope) estimated by the model fitted for the Future Session Time target.

C.3.5 Future Session Activity

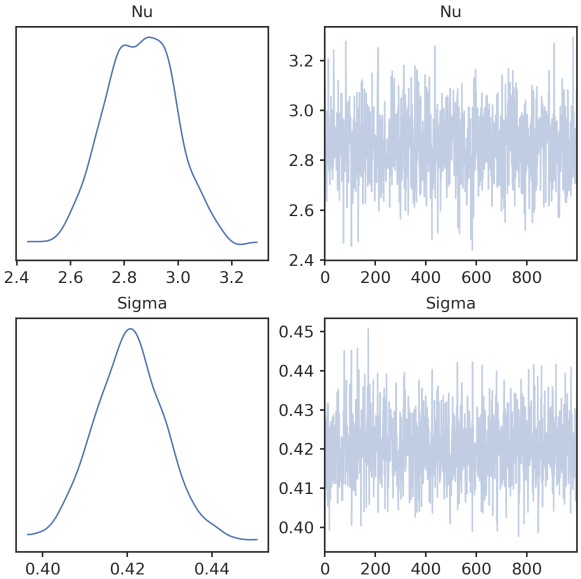


Figure C.41: Marginal distributions for the parameters ν, σ estimated by the model fitted for the Future Session Activity target.

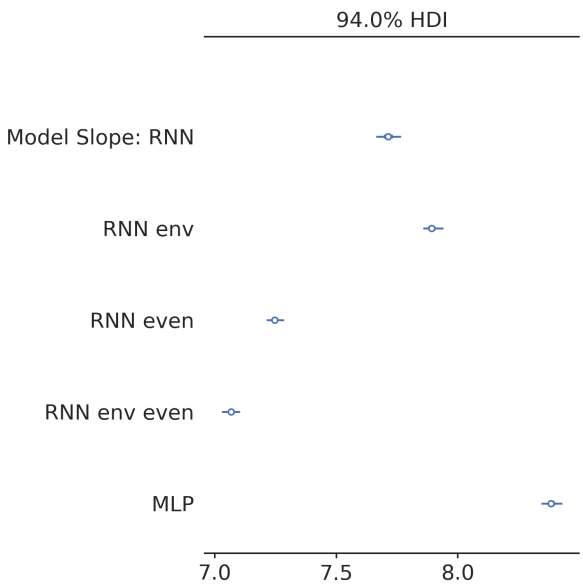


Figure C.42: Forest plot of the marginal distributions for the parameter α (i.e. model slope) estimated by the model fitted for the Future Session Activity target.

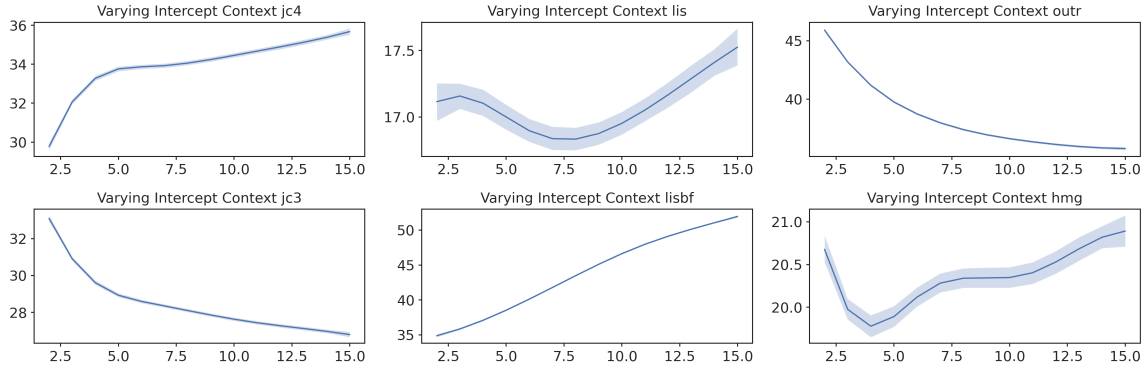


Figure C.43: Time varying random intercept estimated by the model fitted for the Future Session Activity target.

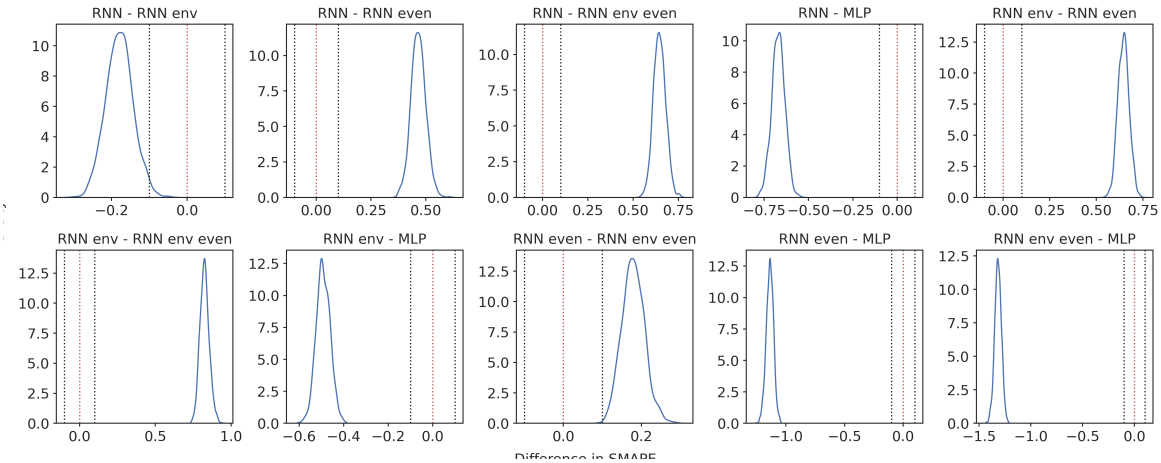


Figure C.44: Pairwise comparisons of the parameter α (i.e. model slope) estimated by the model fitted for the Future Session Activity target.

C.3.6 Future N° Sessions

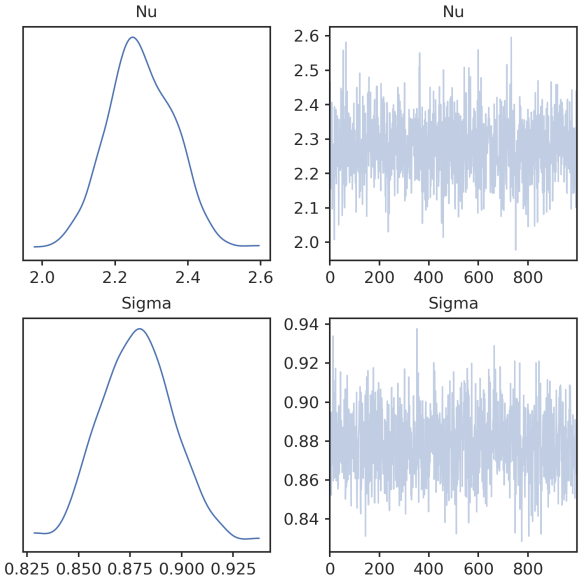


Figure C.45: Marginal distributions for the parameters ν, σ estimated by the model fitted for the Future N°sessions target.

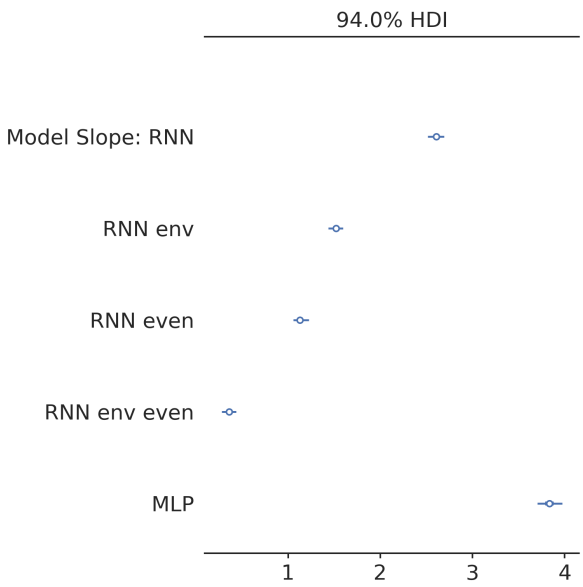


Figure C.46: Forest plot of the marginal distributions for the parameter α (i.e. model slope) estimated by the model fitted for the Future N°sessions target.

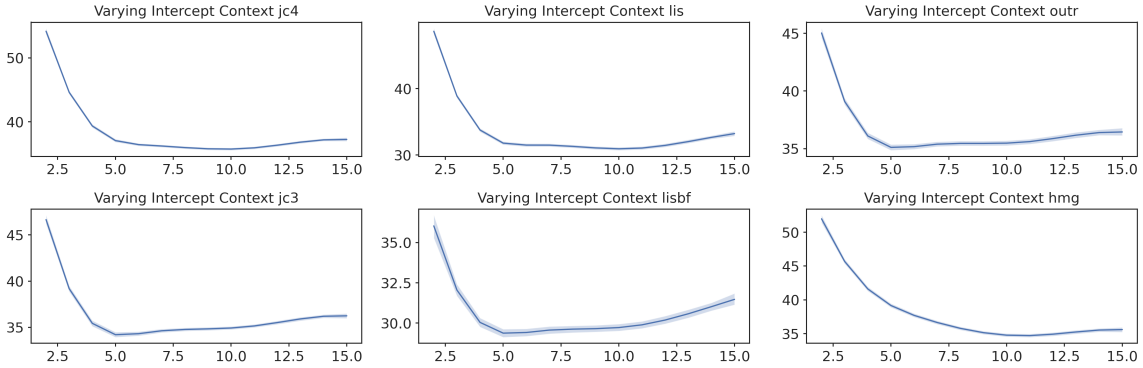


Figure C.47: Time varying random intercept estimated by the model fitted for the Future N°sessions target.

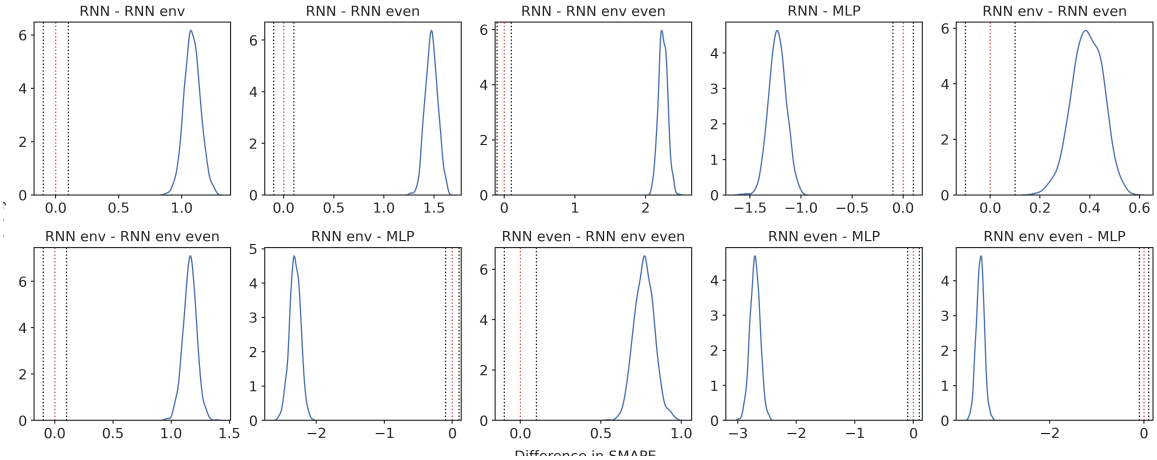


Figure C.48: Pairwise comparisons of the parameter α (i.e. model slope) estimated by the model fitted for the Future N°sessions target.

Appendix D

Ancillary Representation Analyses

D.1 RNN Architecture Learned Representation

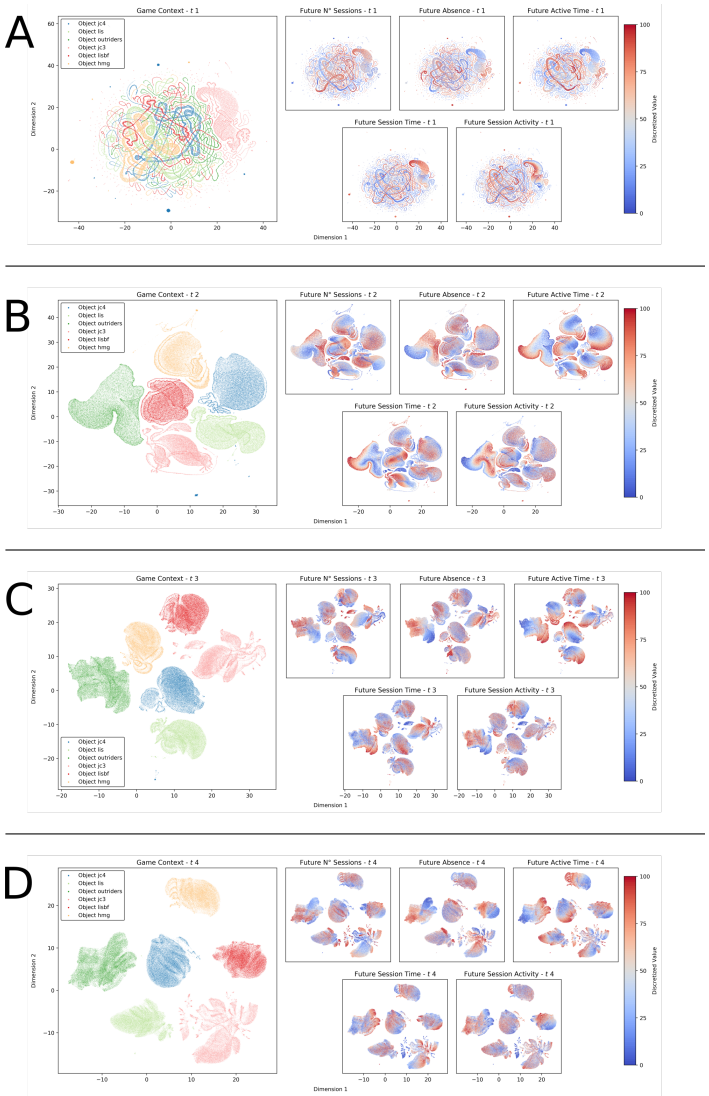


Figure D.1: Each panel shows a two-dimensional projection of the multi-dimensional representation inferred by the RNN architecture. Rows from A to D report the representation inferred at $t1$, $t2$, $t3$ and $t4$. Colours in the large panel indicate which game object the representation is coming from while those in the small panels indicate the discounted sum of future predictions for a single target.

D.2 MLP Architecture Learned Representations

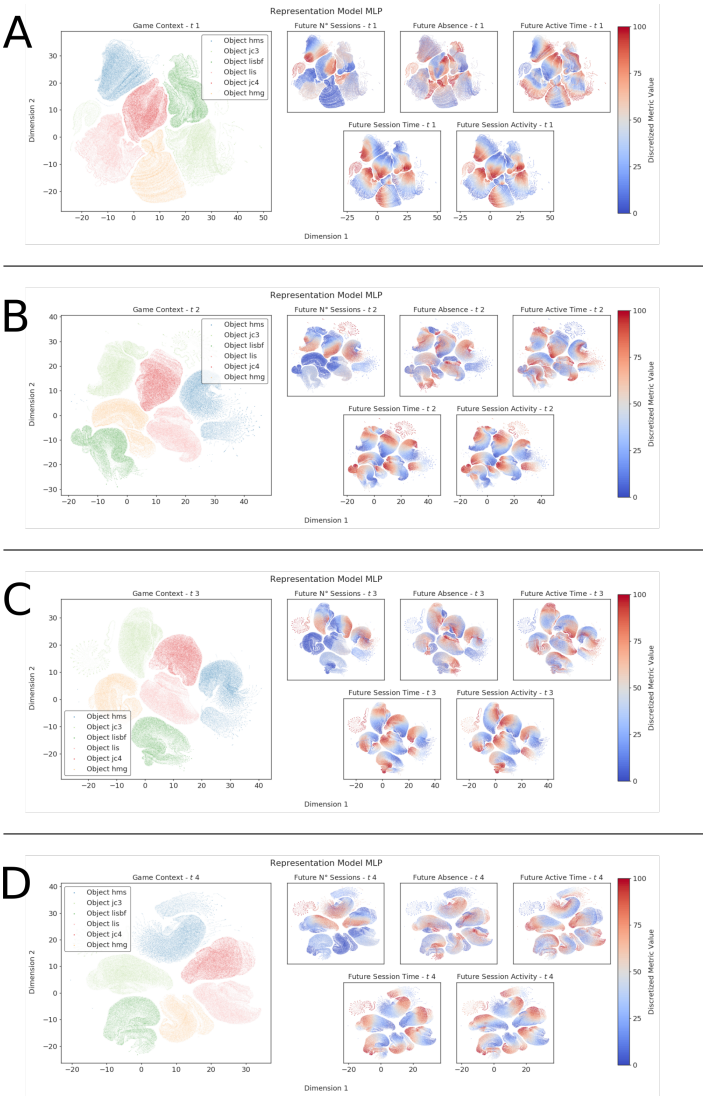


Figure D.2: Each panel shows a two-dimensional projection of the multi-dimensional representation inferred by the MLP architecture. Rows from A to D report the representation inferred at $t1$, $t2$, $t3$ and $t4$. Colours in the large panel indicate which game object the representation is coming from while those in the small panels indicate the discounted sum of future predictions for a single target.

D.3 RNN Architecture with environmental and game events covariates learnt representations

D.3.1 Behavioural Representations

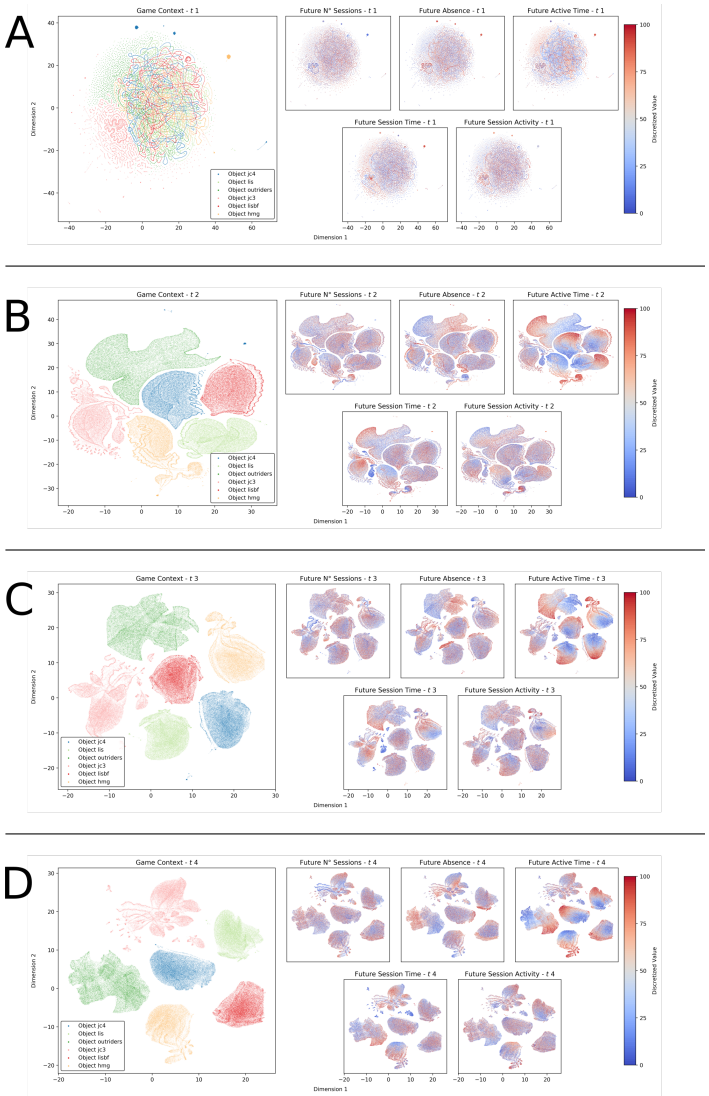


Figure D.3: Each panel shows a two-dimensional projection of the multi-dimensional representation inferred by the improved RNN architecture from the behavioural metrics. Rows from A to D report the representation inferred at $t1$, $t2$, $t3$ and $t4$. Colours in the large panel indicate which game object the representation is coming from while those in the small panels indicate the discounted sum of future predictions for a single target.

D.3.2 Environmental Representations

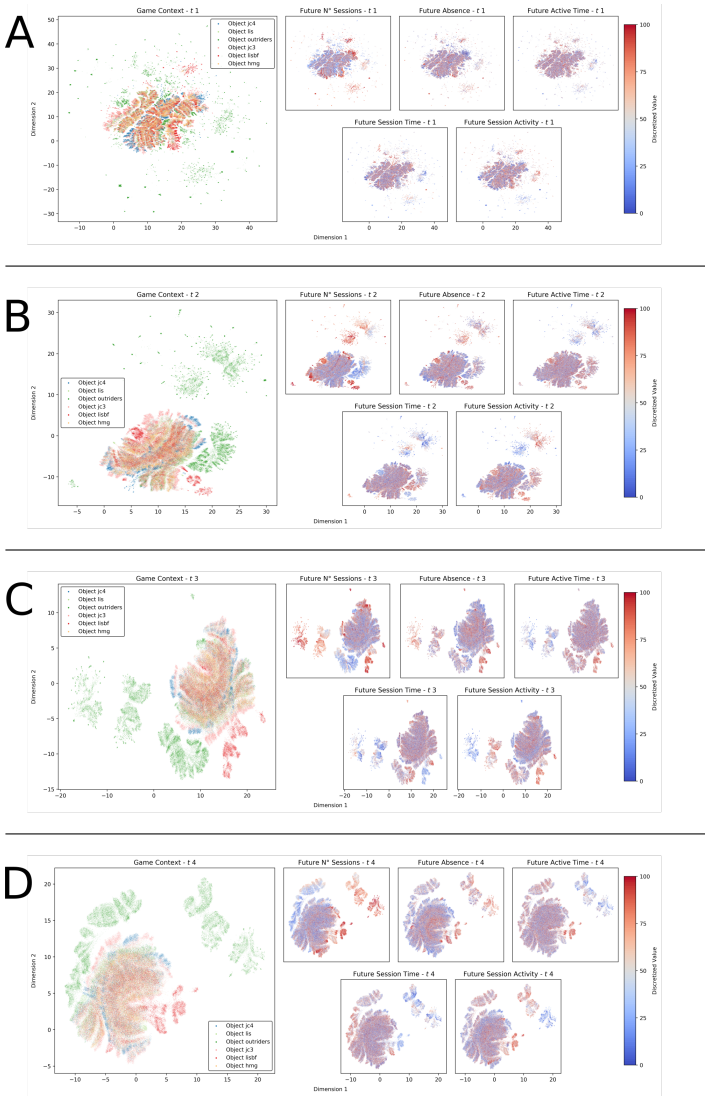


Figure D.4: Each panel shows a two-dimensional projection of the multi-dimensional representation inferred by the improved RNN architecture from the environmental metrics. Rows from A to D report the representation inferred at $t1$, $t2$, $t3$ and $t4$. Colours in the large panel indicate which game object the representation is coming from while those in the small panels indicate the discounted sum of future predictions for a single target.

D.3.3 Game Events Representations

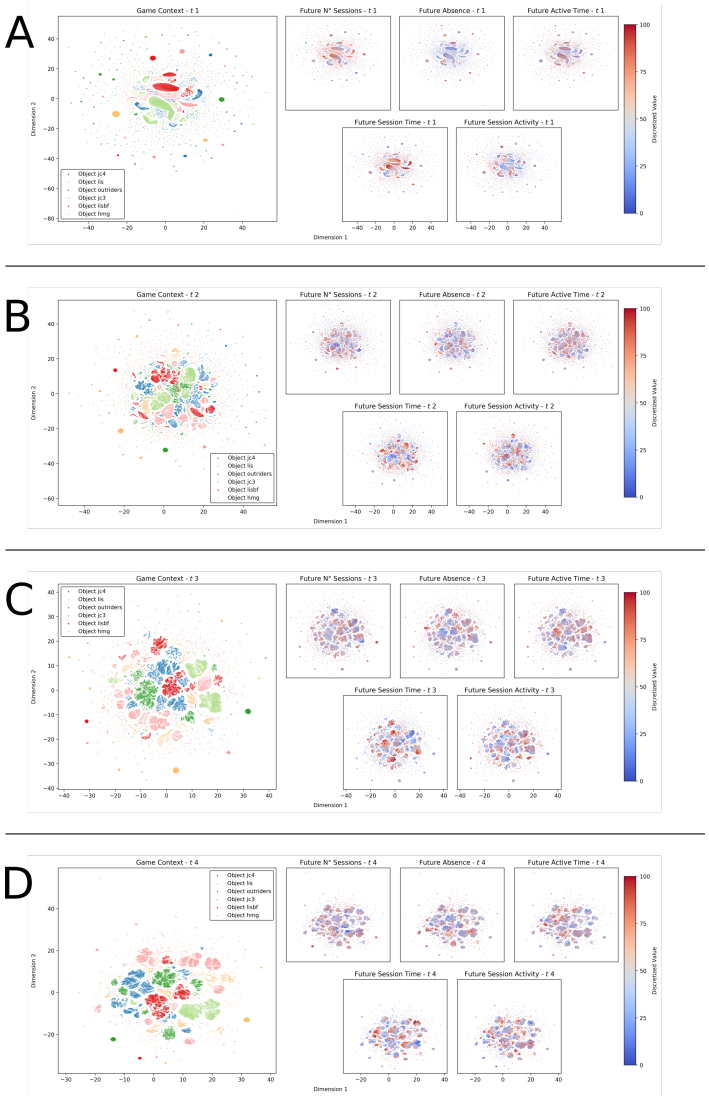


Figure D.5: Each panel shows a two-dimensional projection of the multi-dimensional representation inferred by the improved RNN architecture from the game events metrics. Rows from A to D report the representation inferred at $t1$, $t2$, $t3$ and $t4$. Colours in the large panel indicate which game object the representation is coming from while those in the small panels indicate the discounted sum of future predictions for a single target.

D.3.4 Shared Representations

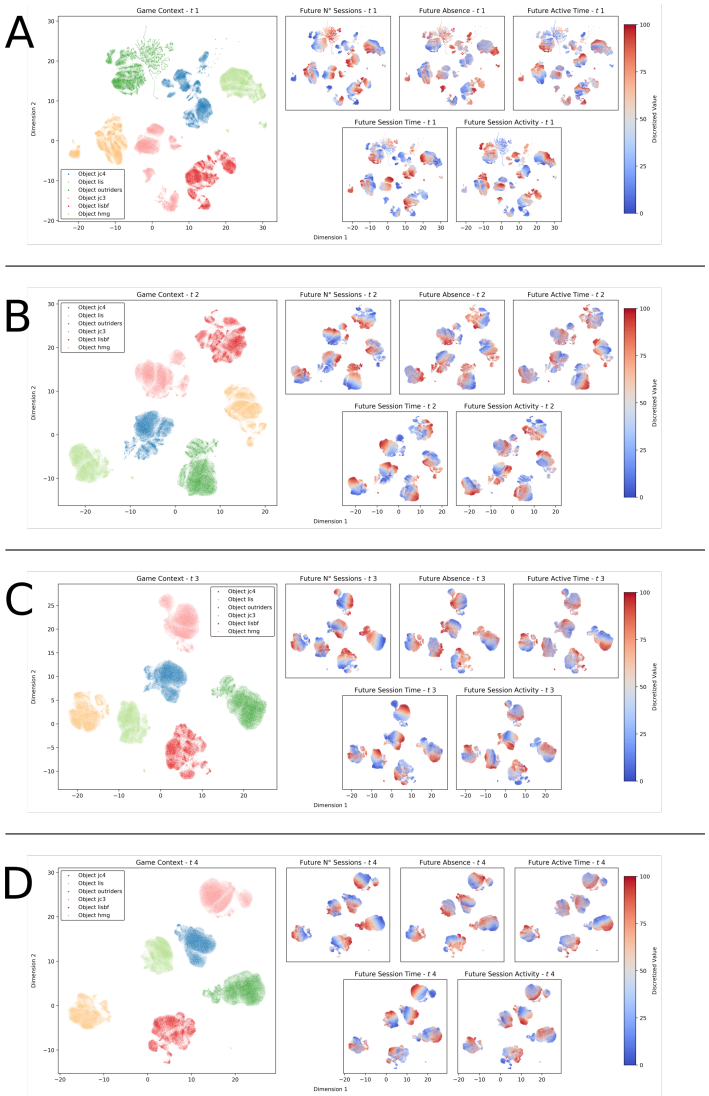


Figure D.6: Each panel shows a two-dimensional projection of the multi-dimensional representation inferred by the improved RNN architecture by combining the behavioural, environmental and game events representations. Rows from A to D report the representation inferred at $t1$, $t2$, $t3$ and $t4$. Colours in the large panel indicate which game object the representation is coming from while those in the small panels indicate the discounted sum of future predictions for a single target.

D.4 Partitions behavioural metrics representations

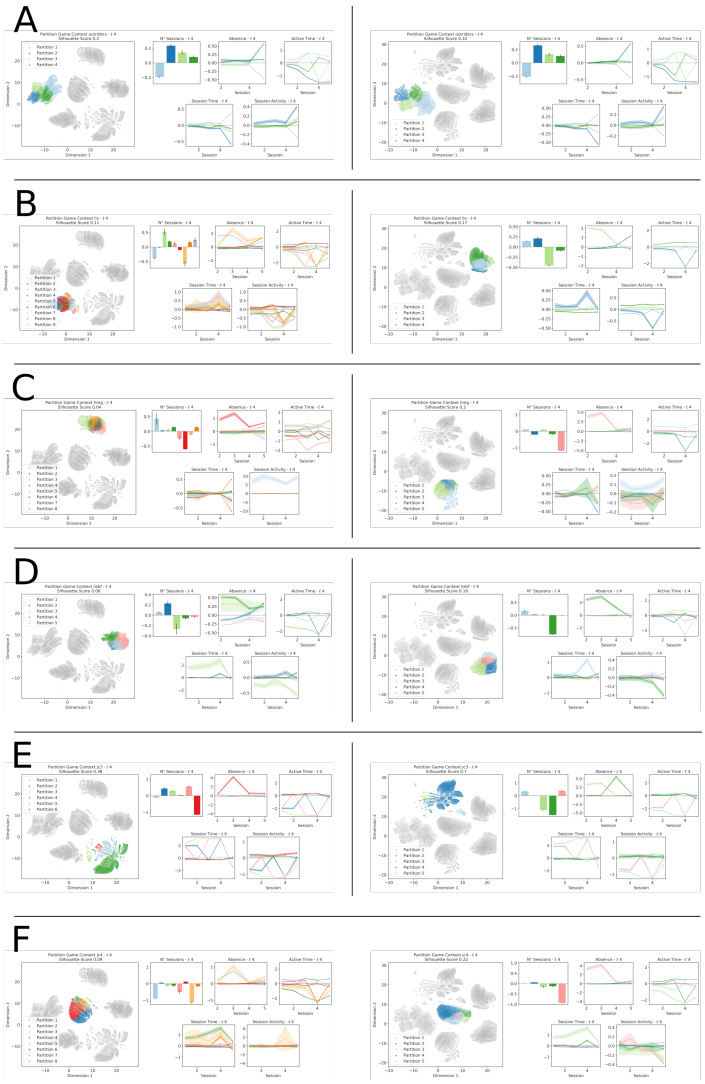


Figure D.7: The two panels show the individuated partitions and associated profiles generated by applying Mini Batch KMeans to the representation generated by the RNN architecture (left) and its improved version (right) using the behavioural metrics. Each row report the partitions associated with each of the considered game contexts.

D.5 Partitions environmental metrics representations

D.5.1 Game Context hmg

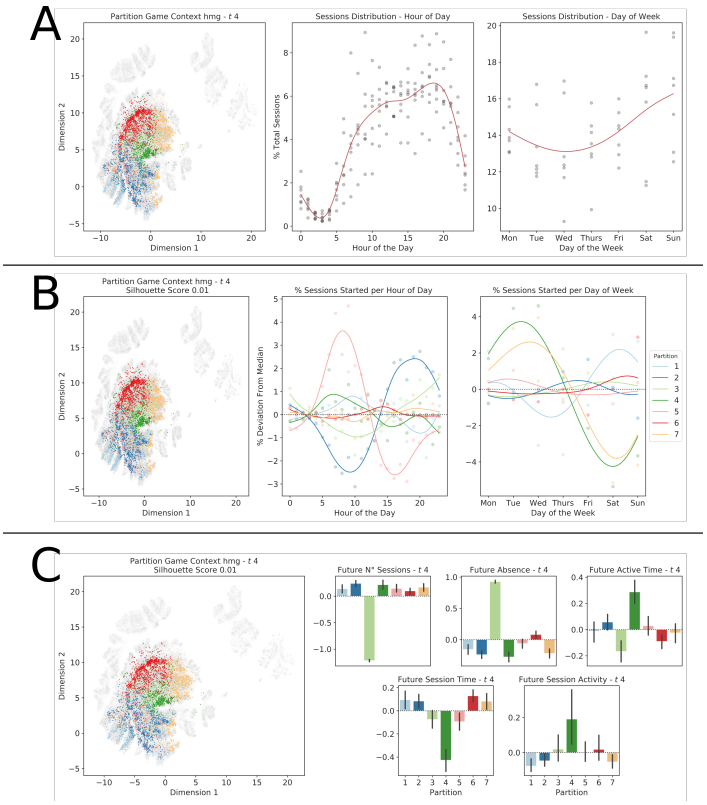


Figure D.8: Partitions of the representation generated from the environmental metrics for the game context hmg

D.5.2 Game Context jc3

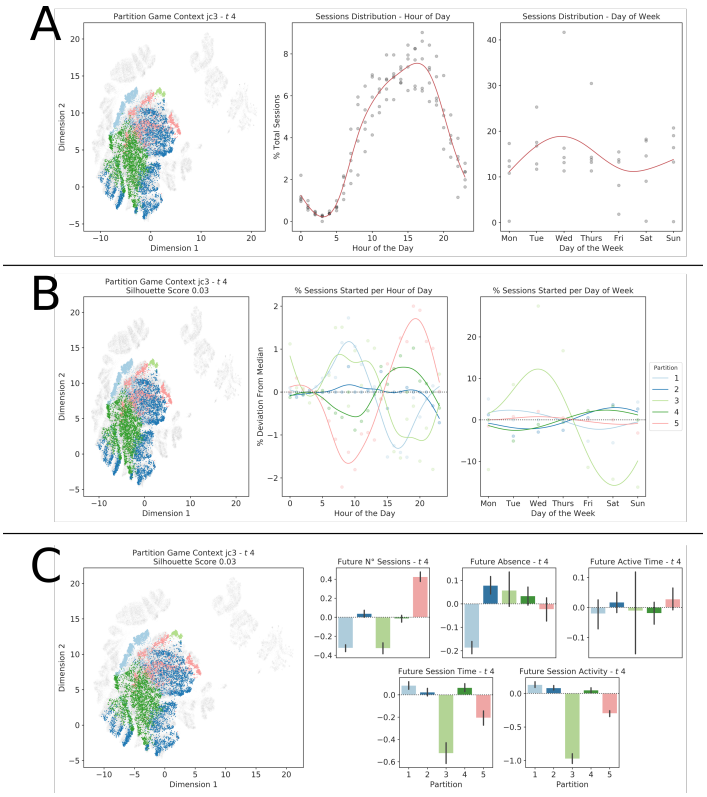


Figure D.9: Partitions of the representation generated from the environmental metrics for the game context jc3

D.5.3 Game Context jc4

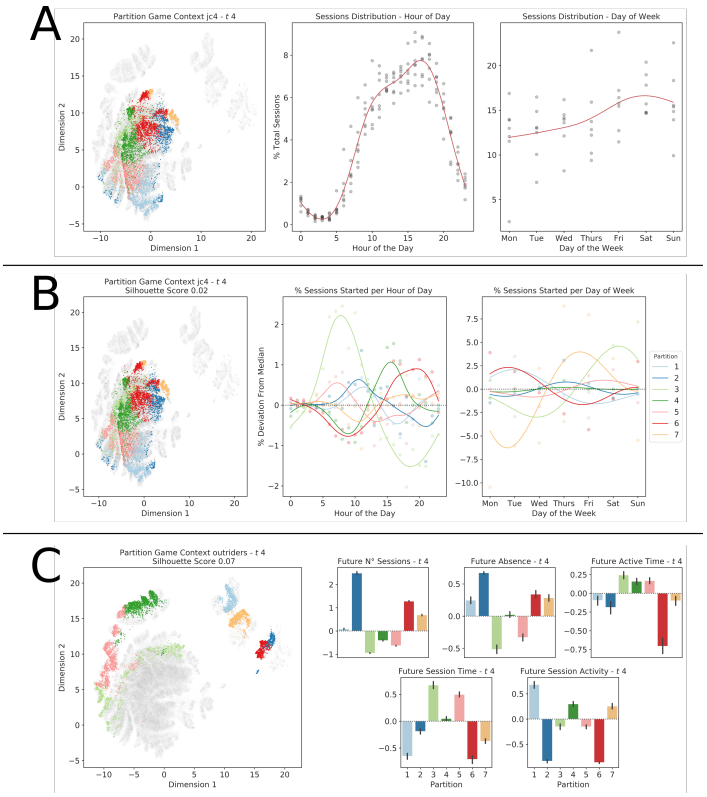


Figure D.10: Partitions of the representation generated from the environmental metrics for the game context jc4

D.5.4 Game Context lis

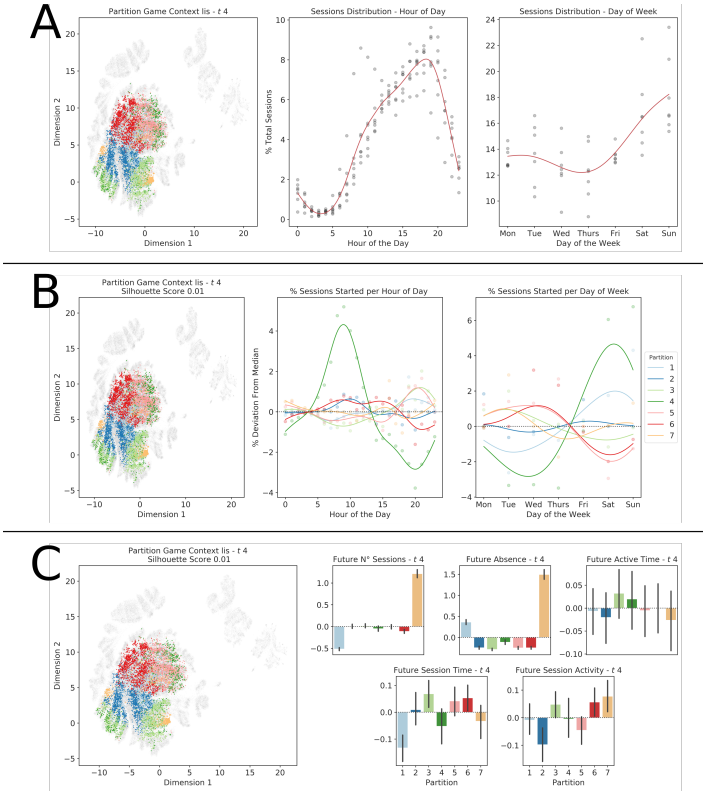


Figure D.11: Partitions of the representation generated from the environmental metrics for the game context lis

D.5.5 Game Context lisbf

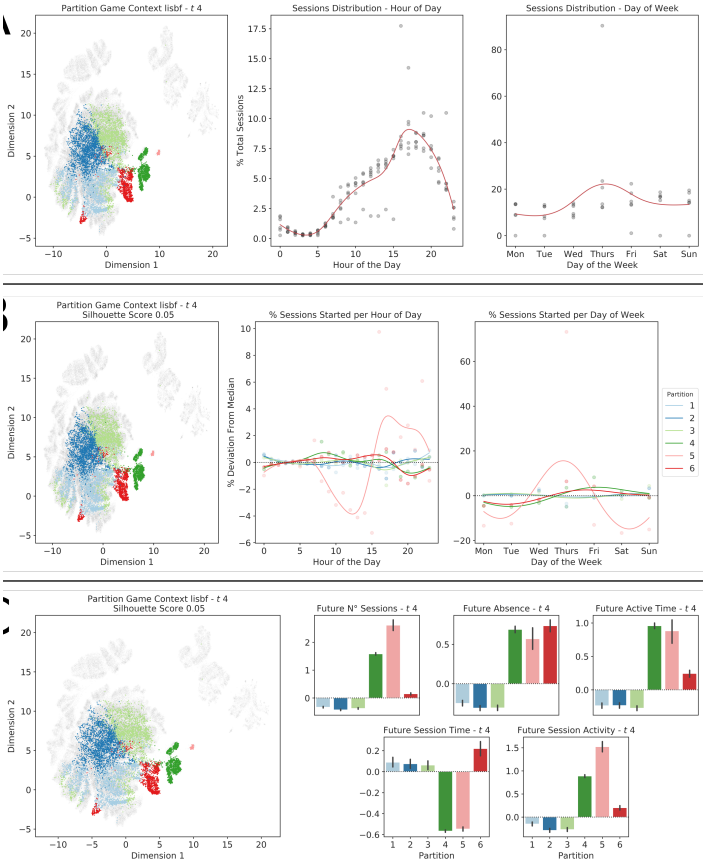


Figure D.12: Partitions of the representation generated from the environmental metrics for the game context lisbf

D.6 Partitions game events metrics representations

D.6.1 Game Context hmg

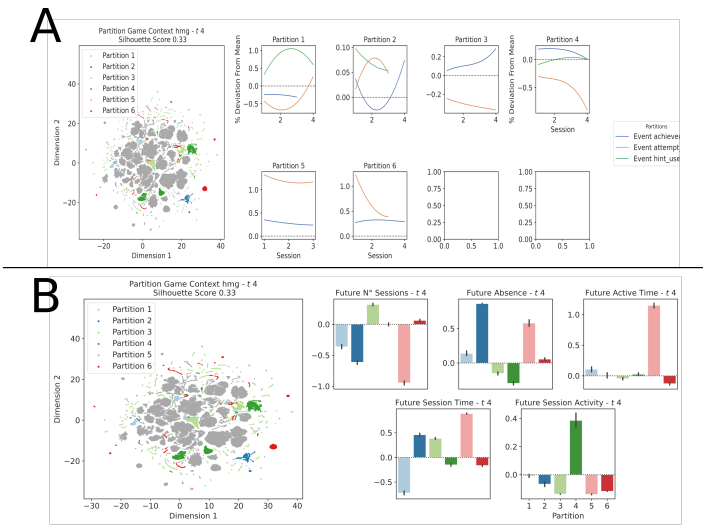


Figure D.13: Partitions of the representation generated from the game events metrics for the game context hmg

D.6.2 Game Context jc3

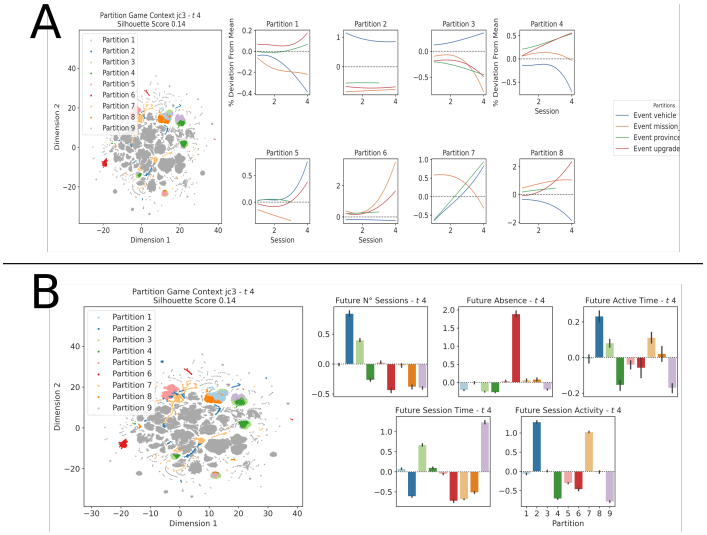


Figure D.14: Partitions of the representation generated from the game events metrics for the game context jc3

D.6.3 Game Context jc4

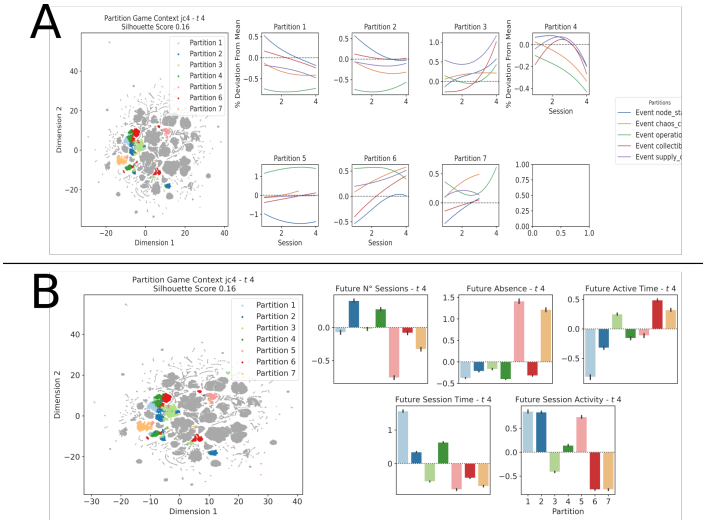


Figure D.15: Partitions of the representation generated from the game events metrics for the game context jc4

D.6.4 Game Context lis

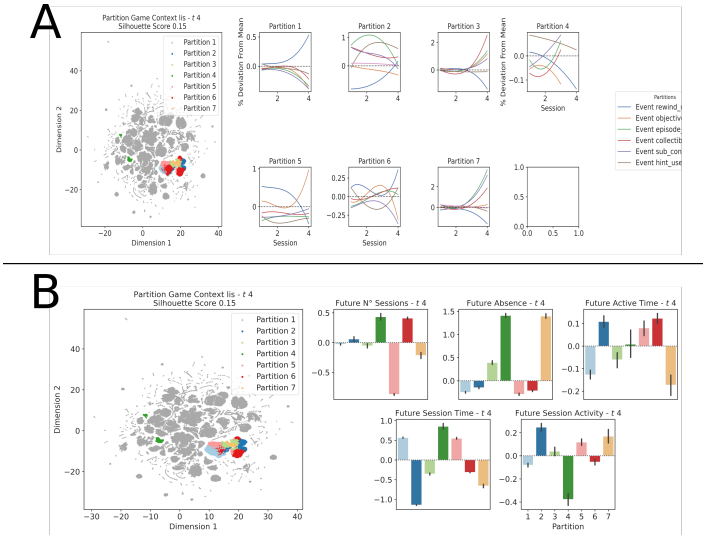


Figure D.16: Partitions of the representation generated from the game events metrics for the game context lis

D.6.5 Game Context lisbf

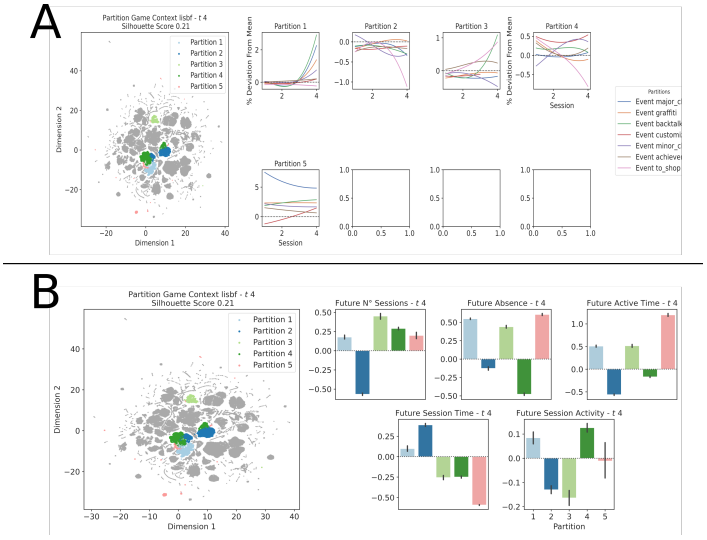


Figure D.17: Partitions of the representation generated from the game events metrics for the game context lisbf