Modelling Early User-Game Interactions for Joint Estimation of Survival Time and Churn Probability

V. Bonometti†‡, C. Ringer†*, M. Hall‡, A.R. Wade§, A. Drachen†
†Department of Computer Science, University of York, York, UK YO10 5DD
*Department of Computing, Goldsmiths, University of London, London, UK SE14 6NW
‡Square Enix Limited, London, UK SE1 8NW
§Department of Psychology, University of York, York, UK YO10 5DD
Email: {vb690, cr1116, anders.drachen, alex.wade}@york.ac.uk; analyticsl@eu.square-enix.com

***Abstract*—Data-driven approaches which aim to identify and predict player engagement are becoming increasingly popular in games industry contexts. This is due to the growing practice of tracking and storing large volumes of in-game telemetries coupled with a desire to tailor the gaming experience to the end-user's needs. These approaches are particularly useful not just for companies adopting Game-as-a-Service (GaaS) models (e.g. for re-engagement strategies) but also for those working under persistent content-delivery regimes (e.g. for better audience targeting). A major challenge for the latter is to build engagement models of the user which are data-efficient, holistic and can generalize across multiple game titles and genres with minimal adjustments.**

This work leverages a theoretical framework rooted in engagement and behavioural science research for building a model able to estimate engagement-related behaviours employing only a minimal set of game-agnostic metrics. Through a series of experiments we show how, by modelling early user-game interactions, this approach can make joint estimates of long-term survival time and churn probability across several single-player games in a range of genres. The model proposed is very suitable for industry applications since it relies on a minimal set of metrics and observations, scales well with the number of users and is explicitly designed to work across a diverse range of titles.

***Index Terms*—Churn Prediction, Survival Estimation, Machine Learning, Engagement, Player Modelling, Game Analytics**

I. Introduction

The video game industry has gained the ability to draw insights on the playing activity from extremely large cohorts of users. This is made possible by the increasing practice of recording a wide range of in-game telemetries coupled with the possibility to store and process massive amounts of data. One of the most important use-case for this type of data is to develop solutions for assessing and predicting engagement-related behaviours [1]–[4]. Indeed, given that the ultimate goal of a game is to deliver a specific entertaining experience to the end user, understanding if, how, and when players are engaged with a game has a pivotal importance in many different areas of applications such as: play-testing [5], stakeholder reporting [2] and automated in-game behaviour estimation [1]. This last one has become perhaps one of the most important goal in contemporary Game Analytics with a major focus on tasks such as churn and survival estimation. Despite the literature on the topic being relatively narrow, a clear pattern of collaboration between academia and industry seems to emerge where one tries to propose solution useful for the other [1]–[4], [6]. In this view, the present work tries to close some of the gaps in the aforementioned literature proposing a novel 'industry-friendly' approach for estimating engagement-related behaviours. We designed a model for joint estimation of survival time and churn probability working across a range of games contexts. Our model requires minimal pre-processing, it employs a restricted set of game-agnostic features, it is suitable for large scale applications and it is able to incorporate uncertainty in its estimations. Moreover, through a series of 3 experiments, we showed how employing a 'hybrid approach' [7] and integrating insights from the engagement and behavioural science literature into the model design allowed us to achieve consistently better results compared to baseline and competing approaches.

II. Engagement

Due to space constrains, this section cannot present an extensive critical review of engagement, indeed our aim is not to provide a theoretical contribution to the construct but rather to operationalize it from a behavioural point of view. Highlighting connections between engagement and behavioural science, we tried to develop a guiding framework for the modelling attempts carried out in our experiments.

A. A video-game perspective on engagement

Defining engagement in digital games is a non-trivial task, with past attempts relying on constructs such as: Flow [8], Self Determination Theory [9], Immersion [10] and Uses and Gratification Theory [11]. However, due to variation in theoretical formulations and the scarcity of empirical validation, the underlying framework appear to be too heterogeneous and not strictly formalized [12]. Therefore, in the present work we decided to adopt a conceptualization, proposed by O'Brien and Toms [13], which we believed formalized engagement in a more precise and most importantly operationalizable way. The authors describe engagement as a multiphasic process arising from the continuous interaction between the user and

This work was supported by the EPSRC Centre for Doctoral Training in Intelligent Games & Games Intelligence (IGGI) [EP/L015846/1] and the Digital Creativity Labs (digitalcreativity.ac.uk), jointly funded by EPSRC/AHRC/Innovate UK under grant no. EP/M023265/1.

the game. Three stages of the aforementioned process are particularly relevant to this work: point of engagement, period of engagement and disengagement. The point of engagement is the moment in which a user directs their attention towards a game due to its ability to fulfill specific drives the user has. The period of engagement is defined by the sustained interaction of a user with a game and is maintained by the ability of the game to provide satisfying feedback. Finally, disengagement defines the moment in which a user makes an active decision to stop interacting with a game due to either external factors or the inability of the game to keep providing rewarding experiences.

Summarizing, engagement seems to be a dynamic process in which the interaction between user and game has a central role. Moreover, this process seems to be initiated, sustained and terminated in accordance with the capability of the user-game interaction to provide rewarding experiences, as highlighted also by [14]–[16].

B. A behavioural perspective on engagement

Considering the previous section, we could argue one way to model the state of the user during the engagement process could be to assess whether they are receiving a sufficiently rewarding experience. This becomes a complex task when performed on a large scale and without the direct involvement of the user because it requires assessing latent, non-observable, states (i.e. the experience of rewarding consequences) influencing measurable outcomes (i.e. behaviour). We can adopt some simple yet powerful concepts from behavioural science (i.e. operant conditioning) to assess whether an activity has reinforcing consequences on an individual by observing the individual's behaviour during the interaction between the two. A comprehensive discussion on 'operant conditioning' would be impossible here given space constraints but, in principle, we say that a specific behaviour tends to increase in amount, frequency and duration when precise reinforcing (i.e. rewarding) consequences are associated to it [17], [18].

Re-framing this in a video game context we could hypothesize that, in absence of external interference, a user will produce in-game behaviour for as long as the game is able to provide rewarding experiences. On the contrary if the game environment fails to provide these experiences, the user would reduce the amount and frequency of in-game behaviour until eventually complete disengagement occurs.

C. Survival Time and Churn Probability as Engagement Approximation

Engagement, while a complex construct, must be circumscribed to simple and quantifiable behaviours when employed for data-driven applications. For this reason we propose survival time and churn as behavioural approximations of future sustained engagement and disengagement. Generally speaking, survival time can be defined as the amount of playing activity occurring between the end of an observation period and the last activity recorded for a specific user [6], [19]–[22]. Churn can be defined as the decision of a user to stop interacting with a specific service due to internal or external reasons, usually formalized as a user entering a prolonged period of inactivity [1]–[4], [22]. While GaaS can only rely on an inactivity period for determining churn, titles with a defined life cycle (e.g. AAA 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 summary, a model estimating survival time and churn from the early stages of the engagement process should include a number of characteristics: 1) Integrate the user-game interaction from the point of engagement through to the end of the on-boarding phase [5], an initial period of sustained engagement critical for assuring long-term engagement; 2) Integrate in-game metrics indicative of behavioural activity to try to infer to what extent the game is providing rewarding experiences; 3) Explicitly model temporality since engagement appears to be a dynamic process which develops over time.

III. State of the art and Current Contribution

A. Churn and Survival Estimation

Given space restrictions, it is not possible to exhaustively describe all the works related to survival and churn estimation, we will therefore focus on some key examples closely linked to our work. When it comes to estimating churn, in particular for industry applications, it is relevant to develop and test models considering different titles and genre. In this view [3], [4], [22], [23] are notable examples where a range of different modelling techniques (i.e. Linear Models, Decision Tress, Naive Bayes, Support Vector Machines and Deep Neural Networks (DNN)) were tested on a churn estimation task across multiple game titles. However, they often employed game specific features (sometimes carefully engineered) and build and test separate models for each game. A notable exception to that is the work done by [24], where churn was formalized as edge prediction in a dynamic graph and modelled through a DNN. This produced a single model able to generalize across multiple game titles however with the limitation of them being all mobile titles. Another important characteristics for a churn estimation model, is to be able to produce predictions even when minimal observations are provided. In this view the works done by [1], [2] highlights the effort made in the literature for designing methodologies able to rapidly provide estimations of churn probability via minimal amount of metrics, this was done employing traditional machine learning algorithms (same as above) but exclusively considering metrics recorded during the initial stages of the user-game interaction. One of the major drawbacks in these works was that the initial period of observation was arbitrarily chosen and fixed for all the considered users making it difficult to take inter individual differences into account. In regard to the literature on survival analysis, we found that most works employed Cox Regression [25], or some variation of it, for estimating the probability to survive (i.e. not have churned) after a specific period of time [19]–[21]. 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 [24], where the goal was to estimate both churn probability and survival time, this also highlights the growing interest for richer assessments of user engagement and for models able to perform both tasks. On top of what is illustrated so far, we also individuated a series of limitations regarding the employed data-sources. The number of the considered users rarely goes beyond 10^4 [24] and when it comes to churn estimation the class distribution is usually greatly imbalanced, both of these factors can pose limitations on the interpretation and generalization of results.

B. Aims and contributions of the present work

The contributions of the present work are two-fold: 1) The three experiments carried out provide insights on the validity of hybrid approaches in engagement modelling while trying to create a bridge between theoretical formulations and data-driven application. 2) The new modelling approach presented tries to fill some of the aforementioned gaps in the literature maintaining at the same time characteristics that makes it appealing for industry applications. While in previous works estimating survival time and churn probability was handled by separate models, our approach can perform both tasks in conjunction, providing a more holistic assessment of user engagement. The model was validated across four separate game genres, deviating from previous works which focused on a single game or a single game genre, potentially limiting the generalizability of the results. Furthermore, in contrast with previous works often utilizing large sets of human-engineered or game-specific features, our proposed model employs only a minimal collection of almost unprocessed and completely game-agnostic metrics, factor that can help reducing overhead in model deployment. In addition, we included in our model the capability to incorporate uncertainty in the predictions, allowing for more cautious interpretation of its estimates when employed in production pipelines.

IV. METHODOLOGY

A. Data

To conduct our experiments, we gathered data from six different games published by our partner company, *Square Enix Limited*. 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 I. Data were gathered from any user playing between the game's release and February 2019, allowing us to adopt more robust sampling strategies which 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 99th 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 applications.

1) Defining the Observation Period: Because we were interested in estimating survival time and churn probability based only on early user-game interactions it was important to define a cut-off at which point interactions were no longer be considered 'early'. We call the period from the user's first interaction till this cut-off the observation period (OP). Choosing the length for the OP was not trivial 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 [1], [2] to take into account natural inter-individual differences. Therefore, we defined the cut-off as:

$$\text{cutoff} = \left\lceil \frac{min(S_t, S_c)}{3} \right\rceil \tag{1}$$

Where S_t is the total number of game play sessions and S_c is the number of game play sessions before the user completed the game for the first time. In this way we take the first ⅓ of all played sessions for players who churned and the first ⅓ 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 for Experiment 2 and 3 where the length of the OP could leak information in the churn probability estimation task. Summarizing, if a user for example had 9 total sessions recorded, we considered the first 3 for making estimations on what happened after the 9^{th}. It goes without saying that at production time the OP is defined only for generating the training samples, the model can be deployed at various stages of previously unseen time series which we simulate in our experiments with the test set.

2) Defining the Behavioural Metrics and Targets: We considered a set of 5 metrics, easily generalizable across games and indicative of behavioural activity, and retrieved them temporally (i.e. over each game session during the OP), see Table II 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 [1]–[4], [19], [22], [23], [26] and survival analysis [6], [20], [21], [26], 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-GaaS games). We took advantage of having access to the complete session history for all users to create a churn definition which

TABLE I: **Data-set Description**. 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		Descriptive Tags
	Min	Max			Min	Max	
hmg	11	260	80,000	80,000	1	7	Mobile, Single Player, Strategy
hms	2	454	80,000	80,000	1	15	Mobile, Single Player, Shooting Gallery
jc3	32	12,695	80,000	80,000	1	20	Console & PC, Single Player, Open World, Action
jc4	7	1,135	80,000	80,000	1	9	Console & PC, Single Player, Open World, Action
lis	5	704	80,000	80,000	1	6	Console & PC, Single Player, Story Driven, Graphic Adventure
libf	14	1,214	80,000	80,000	1	10	Console & PC, Single Player, Story Driven, Graphic Adventure

was robust to the variance in play patterns across games, as it takes into account all the recorded inter-session distances. Therefore, 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:

$$\text{inactivity} = mean(\mathbf{x}) + 2.5 \cdot std(\mathbf{x}) \qquad (2)$$

For better adjusting for inter-individual differences, we could have applied formula 2 to each user individually but this could have created accuracy issue 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 2 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 II: **Considered Metrics over Sessions**

Metric	Description
Session Time	Overall session duration (minutes)
Play Time	Session Time spent actively playing (minutes)
Delta Session	Temporal distance between sessions (minutes)
Activity Index	Count of user initiated game-play-related actions. E.g. ‘Talk to NPC’ or ‘Acquire Upgrade’ were considered valid actions while ‘Click Menu’ or ‘NPC Attacks You’ were not.
Activity Diversity	Count of unique voluntarily initiated actions
Context	Name of the game taken into consideration

3) Data Preparation: We adopted specific data preparation procedures for each experiment. For the first analysis we collapsed the data over the temporal dimension retrieving mean and standard deviation of each considered features, to this concatenating a one-hot encoded transformation of the context metric. For the second and third experiments we kept the data in the original temporal form. In Experiment 3 only we treated the game context slightly differently, numerically encoding it and separating it from the other feature matrix. Since in Experiment 2 and 3 the length of the OP differed between users, we zero padded each sequence of considered sessions to the length of the longest sequence in the data-set. For each experiment we created a tuning and validation subsets (i.e. 20 and 80 % of the original data-set) via stratified shuffle split [27], employing the first for hyper-parameters searching and the second for model evaluation.

B. Experiments

For all experiments we applied the same procedure: first, determined the best hyper-parameters via grid search 10-fold stratified cross validation [27] on the tuning set then evaluated performance via 10-fold stratified cross validation on the validation set. In all experiments, we re-scaled the considered metric separately for each game in outliers-robust way, as in:

$$\text{RobustRescale} = \frac{\mathbf{x} - Q_2(\mathbf{x})}{Q_3(\mathbf{x}) - Q_1(\mathbf{x})} \qquad (3)$$

where $\mathbf{x}$ is the feature vector to be re-scaled and Q_n is the n^{th} quartile for this game. The performance metric that we chose for our survival task was the Symmetric Mean Absolute Percentage Error (SMAPE), defined as:

$$\text{SMAPE} = \frac{\sum_{i=1}^{N} \mid \hat{y}_i - y_i \mid}{\sum_{i=1}^{N} (y_i + \hat{y}_i)} \qquad (4)$$

where N is the collection of all the users in the considered set and $\hat{y}_i$ and y_i are respectively estimated survival time and ground truth value for user *i*. SMAPE was implemented because its scale invariance allowed better comparisons of results across game contexts. For the churn estimation task the chosen metric was the F1 score (F1), defined as:

$$\text{F1} = 2 \cdot \frac{(precision \cdot recall)}{(precision + recall)} \qquad (5)$$

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. We chose the macro-averaged F1 (i.e. employing the unweighted mean of precision and recall for both classes) since our data-set was perfectly balanced.

C. Models

As well as our novel model for joint survival time and churn probability estimation we discuss several models for disjoint estimation, learning only survival time or churn probability, in order to conduct our experiments and compare our model with existing techniques. Furthermore, for providing a baseline comparison in our experiments we employed a mean model (MM), which generates predictions based on the average of the targets in the training set.

1) Models for disjoint estimation: The choice of disjoint estimation models was dictated by a series of needs: widespread usage in research and industry settings, ability to capture linear and non-linear interactions between features and most importantly capability to train on large data-sets (e.g. matrix of dimension $\approx 10^6 \times 10^2$). Four models were employed in Experiments 1 and 2. Firstly, a variant of Regularized Regression, ElasticNet (EN) [28], for survival estimation and Logistic Regression (LR) for churn probability estimation. Secondly, a pair of similar Multi-Layers Perceptron Neural Networks, one tasked to perform survival time regression, MLPr, and one to perform churn classification, MLPc. We felt that given the similarities between linear models and NNs, which can be seen as a stacked version of the former with more 'expressive power', the chosen algorithms constituted a natural progression in the modelling approach. For EN the best hyper-parameters were $\alpha = 0.1$ and a ratio of 0.5 between l1 and l2 regularization. For LR an l1 regularization with $C = 0.01$. Both MLPr and MLPc employed an l2 penalty of 0.01 and utilized a 3 layers architecture with 200, 100 and 50 hidden units. For all hidden units a $ReLU(z) = max(0, z)$ activation function was used, while an $identity(z) = z$ and $sigmoid(z) = \frac{1}{1+\epsilon^{-z}}$ functions were respectively used as final activations for the MLPr and MLPc, where z is a weighted sum of the hidden units of the previous layer. When training the MLP based models a small sub-set was extracted from the training set which represented 10% of the data. This sub-set was used to evaluate convergence of the model and stop the training phase before over-fitting could occur. For both models convergence was determined if the loss did not improve for 3 epochs. The networks were trained using a batch size of 256 and optimized using the Adaptive Moment Estimation (ADAM) optimizer [29]. Because survival time estimation is a regression task and churn prediction is classification task different loss function were used, Mean Squared Error (MSE) and Binary Cross Entropy (BCE) respectively. These are defined as:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{6}$$

$$\text{BCE} = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot log(\hat{y}_i) + (1 - y_i) \cdot log(1 - \hat{y}_i) \tag{7}$$

where N is the size of the batch, and $\hat{y}_i$ and y_i are respectively estimations provided by the model and ground truth value for the i_{th} element in the batch.

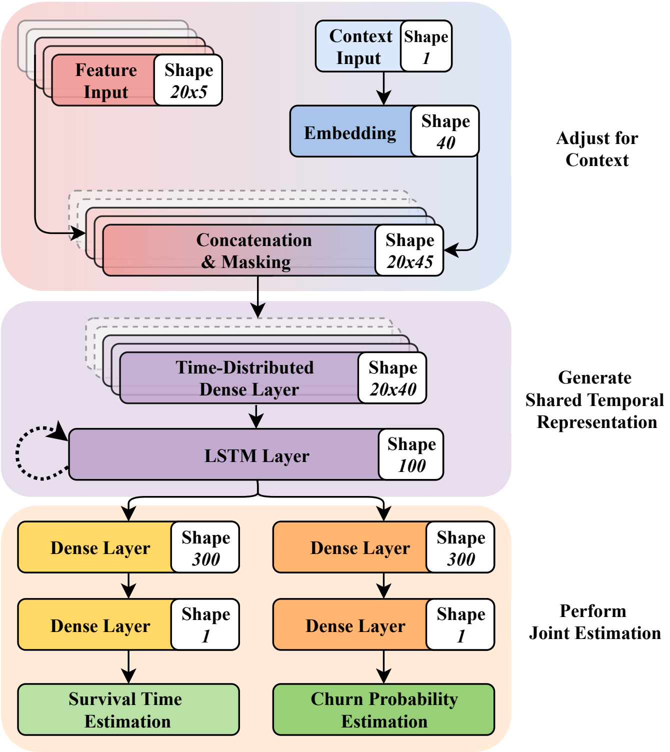


Fig. 1: **Bifurcating Model (BM) Architecture.** The first section learns an embedding for each game context and fuses it, via concatenation, with the feature set. The embedding allows the model to learn a rich multi-dimensional representation of the game context projecting similar games into closer points in the latent space. The second section takes these fused representation over time and models them temporally using an LSTM. The LSTM is particularly suitable because it can handle time series of different lengths and explicitly model temporal dependencies. We e thought to use this part of the model for extracting a high level representation of the player state which could be used for predicting measures of future disengagement and sustained engagement.Inspired by the results of *Experiment 1*, this is achieved by 'branching' two shallow NNs tasked to perform churn probability and survival time estimation.

2) Bifurcating Model for Joint Estimation: For Experiment 3 we present a novel deep neural network architecture, loosely inspired by the winning entry in [26], for jointly estimate survival time and churn probability. This architecture, the 'Bifurcating Model' (BM), is demonstrated in Fig. 1. The model receives, as input, both a vector of unfolded features, as in Table II, as well as a context vector containing a numerical encoding of the game (e.g. jc3 = [1], lis = [2] etc.). The game context is then embedded into a vector of $l = 40$, similarly to what is done in words embedding for sentiment analysis [30]. Differently from a one-hot encoding, this approach provides a non-sparse representation of the input while also projecting it 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). Using an embedding for encoding the game contexts allows to have a representation that grows richer and richer the more categories are included into it. Obviously this would require to re-train the model whenever a new unseen context is added, practice however not just advisable but also routinely done in production. Next, the raw behavioural input and the embedded game context vector are concatenated along the temporal dimension into a single feature vector and a zero-padding re-applied where needed. At this point, a masking layer allows the model to more efficiently work with time-series of different lengths (i.e. skipping the computations for the zero-padded time-steps) and a dense layer, applied to each time step, to combine raw behavioural metrics and context in a new vector of $l = 40$. These newly obtained features are then modelled across time using a Long Short-Term Memory (LSTM) recurrent layer with $n = 100$ units. Therefore, the output of this LSTM Layer is a feature vector of $l = 100$ which is a latent representation of the input features across time and can be seen as providing a high-level representation of the behavioural state of the user during the OP. The final step of this architecture is to then take this high-level latent representation and pass it to a pair of shallow NNs, one tasked with estimating survival time and the other churn probability. These estimators are formed of a pair of densely connected layers, where the first layer has $n = 300$ units and the last has $n = 1$ units, the output of which will constitute the survival time and churn probability estimates. Like the two MLP models the BM was batch trained with a batch size of 256 until convergence using the ADAM optimizer, with learning rate adjusted through a cyclical policy [30], [31], minimizing the sum of the two losses. Similarly to the MLP models, the hidden layers used $ReLU$ as activation function whereas the two outputs units used respectively an $identity$ and $sigmoid$ functions for producing the survival time and churn probability estimates. For the survival time branch SMAPE was used as an objective function while for the churn estimation branch BCE was adopted. We applied two regularization techniques after the computations of the first layers of each shallow NN, batch normalization [32] and dropout [33] ($rate = 0.1$). Additionally, following the intuition from [34], we employed dropout also at inference time for sampling from the model parameters and obtaining a distribution over the posterior so to be able to represent uncertainty in the model estimates. 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 roughly followed a normal distribution the mean could be seen as the value with highest probability. All the experiments were implemented in Python 3.6, with the algorithms for Experiment 1 and 2 provided by the library scikit-learn [27] and our novel BM architecture developed using Keras with Tensorflow as a back-end [30].

V. Results

We will first present results for each disjoint model as well as for a baseline model. Next we will illustrate in detail the performance of the BM model both in terms of it's raw accuracy as well as its capability to include uncertainty in it's output. 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 how often the trained model made the correct classifications without false alarms. The probability threshold employed for discriminating between classes was set to 0.5.

A. Experiment 1

TABLE III: **Performance Baseline Mean Model**

Game	Model	SMAPE	F1
hmg	**MM**	0.767 ± 0.001	0.500 ± 0.003
hms		0.581 ± 0.001	0.507 ± 0.003
jc3		0.632 ± 0.003	0.499 ± 0.004
jc4		0.366 ± 0.002	0.499 ± 0.001
lis		0.404 ± 0.001	0.500 ± 0.003
lisbf		0.244 ± 0.002	0.500 ± 0.005

The results from the first experiment, Table IV, showed how all the 4 models strongly outperformed the MM baseline, Table III, 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 estimation across all games.

TABLE IV: **Performance Collapsed Format**

Game	Model	SMAPE	Model	F1
hmg	**EN**	0.513 ± 0.043	**LR**	0.591 ± 0.004
hms		0.331 ± 0.020		0.624 ± 0.004
jc3		0.423 ± 0.008		0.601 ± 0.004
jc4		0.351 ± 0.006		0.663 ± 0.002
lis		0.287 ± 0.004		0.626 ± 0.003
lisbf		0.239 ± 0.003		0.591 ± 0.003
hmg	**MLPr**	0.304 ± 0.008	**MLPc**	0.660 ± 0.006
hms		0.241 ± 0.007		0.670 ± 0.006
jc3		0.360 ± 0.003		0.654 ± 0.004
jc4		0.334 ± 0.002		0.678 ± 0.004
lis		0.256 ± 0.003		0.664 ± 0.003
lisbf		0.219 ± 0.002		0.622 ± 0.003

B. Experiment 2

Following the results of Experiment 1 we tested the same modelling approaches on the unfolded version of the features, where all data points are provided rather than summary statistics. We observed a similar pattern of results, see Table V, regarding baseline and inter-models comparisons. However, it was clear that using unfolded, temporal data lead to only small improvements over the aggregated data from Experiment 1. This might be explained by the fact that the chosen modelling approaches are not explicitly designed for taking temporal structure into account, for example they have no explicit mechanics for temporal modeling such as those provided by a LSTM.

C. Experiment 3

Informed by the results of Experiment 1 and 2, we proceeded in evaluating the performance of our BM, Table VI, on the unfolded data. We observed how our model achieved a modest but consistent improvements in both churn probability and survival time estimation in all game contexts compared to the previous best model (MLPr and MLPc). From a visual inspection of Figure 2 we can see the presence of a positive linear relationship between estimated and ground

TABLE V: **Performance Unfolded Format**

Game	Model	SMAPE	Model	F1
hmg	**EN**	0.545 ± 0.024	**LR**	0.612 ± 0.004
hms		0.550 ± 0.020		0.626 ± 0.004
jc3		0.384 ± 0.003		0.607 ± 0.003
jc4		0.349 ± 0.002		0.660 ± 0.003
lis		0.302 ± 0.001		0.641 ± 0.004
lisbf		0.235 ± 0.002		0.578 ± 0.003
hmg	**MLPr**	0.293 ± 0.004	**MLPc**	0.683 ± 0.005
hms		0.226 ± 0.004		0.682 ± 0.004
jc3		0.360 ± 0.003		0.643 ± 0.004
jc4		0.331 ± 0.002		0.681 ± 0.003
lis		0.256 ± 0.002		0.673 ± 0.005
lisbf		0.218 ± 0.001		0.627 ± 0.003

truth survival time (indicative of accordance between the two), with a roughly even distribution of error along the entire range of values. In Table VII 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 3a and 3b we can see how the model was able to encode different levels of uncertainty through the distribution's variance of estimated values.

TABLE VI: **Performance Bifurcating Model**

Game	Models	SMAPE	F1
hmg	**BM**	0.275 ± 0.001	0.693 ± 0.002
hms		0.200 ± 0.001	0.701 ± 0.003
jc3		0.344 ± 0.003	0.671 ± 0.005
jc4		0.325 ± 0.002	0.685 ± 0.002
lis		0.246 ± 0.002	0.688 ± 0.003
lisbf		0.208 ± 0.001	0.645 ± 0.003

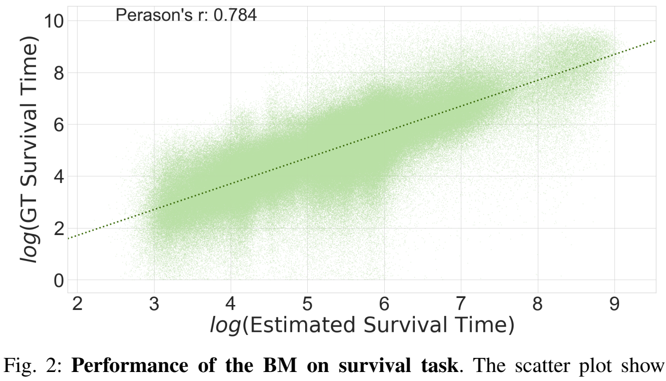
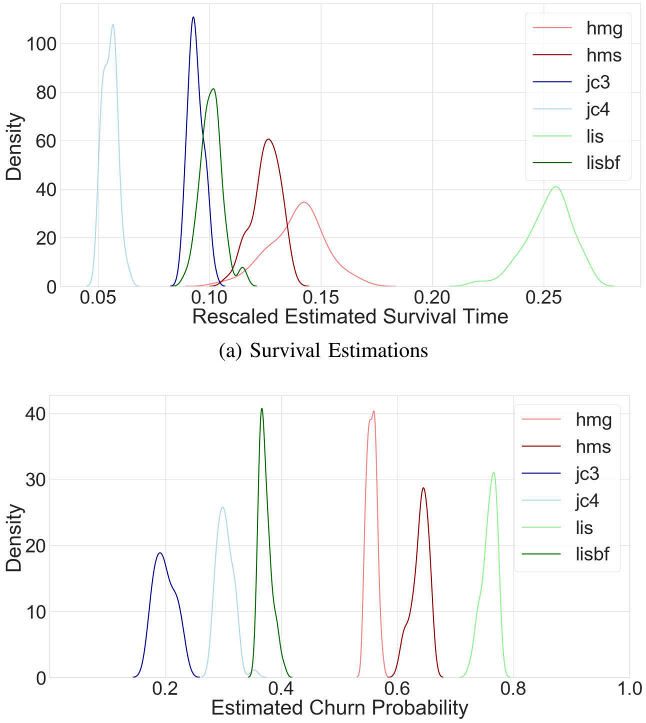


Fig. 2: **Performance of the BM on survival task**. The scatter plot shows the relationship between the survival estimates provided by the BM and the ground truth values. Since the relationship is evaluated on the *log* of both variables, due to the presence extreme outliers in the ground truth, this acts as mostly as a qualitative complement to the more reliable SMAPE measure.



(a) Survival Estimations

(b) Churn Probability Estimation

Fig. 3: **Distribution of the BM estimates for six random users, one for each game**. 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 estimated 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).

TABLE VII: **Performance of the BM on churn task.** Here the diagonal shows the % of correctly predicted users for each label across all games.

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

VI. Discussion, limitations and future work

The results of our experiments highlight how employing metrics indicative of behavioural activity in early user-game interactions allowed our model to estimate proxy measures of future disengagement and sustained engagement. This suggests that the early user-game interactions might be relevant for characterizing long-term engagement as well as that measures of behavioural activity could be a useful index for its inference [1], [5]. We also found how the use of non-parametric models, able to capture non-linear interactions between features provided substantial improvements in estimating proxy measures of engagement when compared to simpler, although computationally cheaper, parametric ones. We also show that including temporal structure explicitly provides a slight edge over metrics representations which are collapsed over time, moreover we noticed that this improvement is more pronounced and consistent when employing approaches that explicitly model temporality, i.e. the BM. This is in accordance with the aforementioned theoretical formalization of engagement as a dynamic process rather than a static construct [13]. Finally the visual representation of the performance of 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.

While the work presented here crosses various game genres, it does not include all the major ones (e.g. multi-player titles). Moreover, despite acknowledging the complexity of the chosen estimation task, better model performance would have been desirable. Finally, the heavy dependence on a supervised approach for learning the context embedding and the inability to fully exploit the LSTM potential (i.e. our time series were at maximum 20 steps long) limited the potential of our approach. Future work will try to improve on these drawbacks considering more game genres, integrating approaches for learning context in an unsupervised way and taking into consideration longer streams of sessions. We will also try to explicitly model the contribution of elements external to the game environment for taking into account the impact of real-world factors (e.g. day of the week or time of the day).

VII. Acknowledgments

We would like to thank Ivan Bravi, Simon Demediuk and Peter York for their invaluable insights provided during the realization of this work and the *Square Enix Limited* Analytics and Insight Team for the continuous support provided.

References

[1] M. Milošević, N. Živić, and I. Andjelković, "Early churn prediction with personalized targeting in mobile social games," *Expert Systems with Applications*, vol. 83, pp. 326–332, 2017.

[2] A. Drachen, E. T. Lundquist, Y. Kung, P. Rao, R. Sifa, J. Runge, and D. Klabjan, "Rapid prediction of player retention in free-to-play mobile games," in *Twelfth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2016.

[3] J. Runge, P. Gao, F. Garcin, and B. Faltings, "Churn prediction for high-value players in casual social games," in *2014 IEEE conference on Computational Intelligence and Games*. IEEE, 2014, pp. 1–8.

[4] F. Hadiji, R. Sifa, A. Drachen, C. Thurau, K. Kersting, and C. Bauckhage, "Predicting player churn in the wild," in *2014 IEEE Conference on Computational Intelligence and Games*. IEEE, 2014, pp. 1–8.

[5] P. Mirza-Babaei, L. E. Nacke, J. Gregory, N. Collins, and G. 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*. ACM, 2013, pp. 1499–1508.

[6] M. Viljanen, A. Airola, J. Heikkonen, and T. Pahikkala, "Playtime measurement with survival analysis," *IEEE Transactions on Games*, vol. 10, no. 2, pp. 128–138, 2018.

[7] G. N. Yannakakis, P. Spronck, D. Loiacono, and E. André, "Player modeling," in *Dagstuhl Follow-Ups*, vol. 6. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2013.

[8] J. L. Sherry, "Flow and media enjoyment," *Communication theory*, vol. 14, no. 4, pp. 328–347, 2004.

[9] R. M. Ryan, C. S. Rigby, and A. Przybylski, "The motivational pull of video games: A self-determination theory approach," *Motivation and emotion*, vol. 30, no. 4, pp. 344–360, 2006.

[10] C. Jennett, A. L. Cox, P. Cairns, S. Dhoparee, A. Epps, T. Tijs, and A. Walton, "Measuring and defining the experience of immersion," *International journal of human-computer studies*, vol. 66, no. 9, pp. 641–661, 2008.

[11] K. Lucas and J. L. Sherry, "Sex differences in video game play: A communication-based explanation," *Communication research*, vol. 31, no. 5, pp. 499–523, 2004.

[12] E. A. Boyle, T. M. Connolly, T. Hainey, and J. M. Boyle, "Engagement in digital entertainment games: A systematic review," *Computers in human behavior*, vol. 28, no. 3, pp. 771–780, 2012.

[13] H. L. O'Brien and E. 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*, vol. 59, no. 6, pp. 938–955, 2008.

[14] H. Wang and C.-T. Sun, "Game reward systems: Gaming experiences and social meanings." in *DiGRA Conference*, 2011, pp. 1–15.

[15] C. Phillips, D. Johnson, and P. Wyeth, "Videogame reward types," in *Proceedings of the First International Conference on Gameful Design, Research, and Applications*. ACM, 2013, pp. 103–106.

[16] D. Ašeriškis and R. Damaševičius, "Computational evaluation of effects of motivation reinforcement on player retention," *Journal of Universal Computer Science*, vol. 23, no. 5, pp. 432–453, 2017.

[17] B. F. Skinner, *Science and human behavior*. Simon and Schuster, 1953, no. 92904.

[18] K. C. Berridge, T. E. Robinson, and J. W. Aldridge, "Dissecting components of reward: liking, wanting, and learning," *Current opinion in pharmacology*, vol. 9, no. 1, pp. 65–73, 2009.

[19] Á. Periáñez, A. Saas, A. Guitart, and C. 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)*. IEEE, 2016, pp. 564–573.

[20] S. Demediuk, A. Murrin, D. Bulger, M. Hitchens, A. Drachen, W. L. Raffe, and M. Tamassia, "Player retention in league of legends: a study using survival analysis," in *Proceedings of the Australasian Computer Science Week Multiconference*. ACM, 2018, p. 43.

[21] P. Bertens, A. Guitart, and Á. Periáñez, "Games and big data: A scalable multi-dimensional churn prediction model," in *2017 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 2017, pp. 33–36.

[22] S. Kim, D. Choi, E. Lee, and W. Rhee, "Churn prediction of mobile and online casual games using play log data," *PloS one*, vol. 12, no. 7, p. e0180735, 2017.

[23] H. Xie, S. Devlin, D. Kudenko, and P. Cowling, "Predicting player disengagement and first purchase with event-frequency based data representation," in *2015 IEEE Computational Intelligence and Games*. IEEE, 2015, pp. 230–237.

[24] X. Liu, M. Xie, X. Wen, R. Chen, Y. Ge, N. Duffield, and N. 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)*. IEEE, 2018, pp. 277–286.

[25] D. R. Cox, "Regression models and life-tables," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 34, no. 2, pp. 187–202, 1972.

[26] E. Lee, Y. Jang, D.-M. Yoon, J. Jeon, S.-i. Yang, S. Lee, D.-W. Kim, P. P. Chen, A. Guitart, P. Bertens *et al.*, "Game data mining competition on churn prediction and survival analysis using game log data," *IEEE Transactions on Games*, 2018.

[27] 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*, vol. 12, pp. 2825–2830, 2011.

[28] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the royal statistical society: series B (statistical methodology)*, vol. 67, no. 2, pp. 301–320, 2005.

[29] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[30] F. Chollet *et al.*, "Keras," https://github.com/fchollet/keras, 2015.

[31] L. N. Smith, "Cyclical learning rates for training neural networks," in *2017 Winter Conference on Applications of Computer Vision*. IEEE, 2017, pp. 464–472.

[32] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[33] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[34] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*, 2016, pp. 1050–1059.