Contents lists available at ScienceDirect

International Journal of Research in Marketing

journal homepage: www.elsevier.com/locate/ijresmar

Full length article

Customer base analysis with recurrent neural networks

Jan Valendin[a,*], Thomas Reutterer[a], Michael Platzer[b], Klaudius Kalcher[b]

[a] Department of Marketing, WU Vienna University of Economics and Business, Welthandelsplatz 1, A-1020 Vienna, Austria
[b] MOSTLY AI Solutions, Hegelgasse 21/3, A-1010 Vienna, Austria

ARTICLE INFO

Article history:
First received on October 15, 2020 and was under review for 7½ months
Available online 11 March 2022

Area Editor: Michael Haenlein
Accepting Editor: Renana Peres

Keywords:
Customer base analysis
Forecasting
Seasonality
Customer covariates
Deep learning
Customer lifetime value

ABSTRACT

One of the primary goals that researchers look to achieve through customer base analysis is to leverage historical records of individual customer transactions and related context factors to forecast future behavior, and to link these forecasts with actionable characteristics of individuals, managerially significant customer sub-groups, and entire cohorts. This paper presents a new approach that helps firms leverage the automatic feature extraction capabilities of a specific type of deep learning models when applied to customer transaction histories in non-contractual business settings (i.e., when the time at which a customer becomes inactive is unobserved by the firm). We show how the proposed deep learning model improves on established models both in terms of individual-level accuracy and overall cohort-level bias. It also helps managers in capturing seasonal trends and other forms of purchase dynamics that are important to detect in a timely manner for the purpose of proactive customer-base management. We demonstrate the model performance in eight empirical real-life settings which vary broadly in transaction frequency, purchase (ir)regularity, customer attrition, availability of contextual information, seasonal variance, and cohort size. We showcase the flexibility of the approach and how the model further benefits from taking into account static (e.g., socio-economic variables, demographics) and dynamic context factors (e.g., weather, holiday seasons, marketing appeals). We make an open-source reference implementation of the newly developed method available at https://github.com/valendin/rfm2lstm.

1. Introduction

Anticipating future customer behavior and making individual-level predictions for a firm's customer base is crucial to any organization that wants to manage its customer portfolio proactively. More precisely, firms following a customer-centric business approach need to know how their clientele will behave on different future time scales and levels of behavioral complexity (Gupta & Lehmann, 2005; Fader, 2020): What are they going to do in the immediate future and when do they make their next transaction with the focal company, if any? Are some of them at risk of stopping doing business with the firm? How exactly do seasonality and other time-based events influence the propensity of customers to buy?

To address these questions and to assist managers in designing their marketing programs accordingly, the marketing discipline has produced a rich stream of literature. These contributions include predictive models and techniques for customer targeting and reactivation timing (Gönül & ter Hofstede, 2006; Simester, Sun, & Tsitsiklis, 2006; Holtrop & Wieringa, 2020),

[*] Corresponding author.
E-mail addresses: valendin@gmail.com (J. Valendin), thomas.reutterer@wu.ac.at (T. Reutterer), michael.platzer@mostly.ai (M. Platzer), klaudius.kalcher@mostly.ai (K. Kalcher).

https://doi.org/10.1016/j.ijresmar.2022.02.007
0167-8116/© 2022 The Author(s). Published by Elsevier B.V.

market response models for firm- and/or customer-initiated marketing actions (e.g., Hanssens, Parsons, & Schultz (2003), Blattberg, Kim, & Neslin (2008), Sarkar & De Bruyn (2021)), methods for churn prediction and prevention (e.g., Ascarza (2018), Ascarza, Iyengar, & Schleicher (2016), Lemmens & Gupta (2020)), as well as a growing literature on customer valuation (e.g., McCarthy, Fader, & Hardie (2017), McCarthy & Fader (2018)) and customer prioritizing (Homburg, Droll, & Totzek, 2008). However, none of these qualify as a (Swiss Army knife-like) general-purpose problem solver that generalizes across the described decision tasks of managing customer relationships. This article makes a first step towards this direction. We propose and implement a flexible methodological framework that provides marketing managers with highly accurate forecasts of fine granularity both in the short and in the long run. Our method also captures seasonal peaks and customer-level dynamics and allows to differentiate between different customer groups.

The challenge to derive such individual-level predictions is particularly demanding in the context of non-contractual settings (such as most retail businesses, online media consumption, charity donations). Contrary to subscription-based or contractual settings where customer "churn" events are directly observable, customer defection in non-contractual business settings is by definition unobserved by the firm and thus needs to be indirectly inferred from past transaction behavior (Reinartz & Kumar, 2000; Gupta et al., 2006). The specific challenge in such settings is to accurately and timely inform managers on the subtle distinction between a pending defection event (i.e., a customer stops doing business with the focal firm) and an extended period of inactivity of their customers, because possible marketing implications are completely different in each of these situations.

Consider, for example, the situation for a few prototypical customer transaction histories depicted in Fig. 1, which are from a customer cohort of a large U.S. charity organization we will study in our empirical evaluation section in more detail. From a managerial perspective, accurately spotting the future activity patterns of such customers is of vital importance because of their value to the company (Blattberg & Deighton, 1996; McCarthy & Fader, 2018). They were all high frequency donors in the past; however, as we will further demonstrate in more detail in the empirical evaluation section, the evaluation of their future with the charity institution will lead us to different conclusions. For instance,

- What would we expect from customers like the first ten individuals 1001–1010, who started out as occasional benefactors, but through an evolving relationship with the firm have developed a more regular transaction[1] behavior? Will they continue this trend; will they eventually turn into the firm's *premium* customers?
- Conversely, how about the next ten individuals 1011–1020, who have all made a number of transactions historically, but recently have been on an unusually long hiatus? Is the customer-firm relationship *at risk* and are these customers potential defectors? A timely response is critical in such a situation, because it is generally easier to regain a customer before their new relationship with a competitor has consolidated.

In this specific domain of customer base analysis, probabilistic approaches from the "Buy 'Till You Die" (BTYD) model family represent the gold standard, leveraging easily observable Recency and Frequency (RF, or RFM when including also the monetary value) metrics together with a latent attrition process to deliver accurate predictions (Schmittlein, Morrison, & Colombo, 1987; Fader, Hardie, & Lee, 2005; Fader & Hardie, 2009). The simple behavioral story which sits at the core of BTYD models – *while "alive", customers make purchases until they drop out* – gives these models robust predictive power, especially on the aggregate cohort level, and over a long time horizon. Extended variants of the original models (e.g., Zhang, Bradlow, & Small (2015), Platzer & Reutterer (2016), Reutterer, Platzer, & Schröder (2021)) improve predictive accuracy by incorporating more hand-crafted summary statistics of customer behavior. However, including customer covariates is cumbersome and an approach to account for time-varying covariates has only just recently been introduced by Bachmann, Meierer, and Näf (2021) at the cost of manual labeling and slower performance. Even advanced BTYD models can be too restrictive to adequately capture diverse customer behaviors in different contexts and the derived forecasts present customer future in an oftentimes too simplified way.

Other options to capture changes between lower- and higher-frequency purchase episodes (as we observe for our customers in Fig. 1), or vice versa, are to adopt a dynamic changepoint model (Fader, Hardie, & Chun-Yao, 2004), a simulation based model of the type presented by Rust, Kumar, and Venkatesan (2011), or to incorporate additional states other than the absorbing, inactive state as in standard BTYD latent attrition models. The latter way of accounting for nonstationarity in transaction sequences can be achieved by applying more general hidden Markov models (see, e.g., Netzer, Lattin, & Srinivasan (2008), Schweidel, Bradlow, & Fader (2011), Romero, van der Lans, & Wierenga (2013)). A Bayesian nonparametric approach to flexibly model purchasing dynamics depending on calendar time effects, inter-event timing and customer lifetime was recently proposed by Dew and Ansari (2018). However, all such approaches come at the cost of additional model complexity, rising computational cost, and a loss in sufficiency.

In this paper, we offer marketing analysts an alternative to these models by developing a deep learning based approach that does not rely on any ex-ante data labelling or feature engineering, but instead automatically detects behavioral dynamics like seasonality or changes in inter-event timing patterns by learning directly from the prior transaction history. This enables us to simulate future transactions at a very fine granular level and attribute them to the right customer (or any sub-

[1] For expositional simplicity, we refer to the act of making an order, a donation, or any type of purchase event as a *transaction*, to charitable contributors, blood donors, bank clients, households, etc. similarly as *customers*, and to the charity, bank, or store as the *firm*.

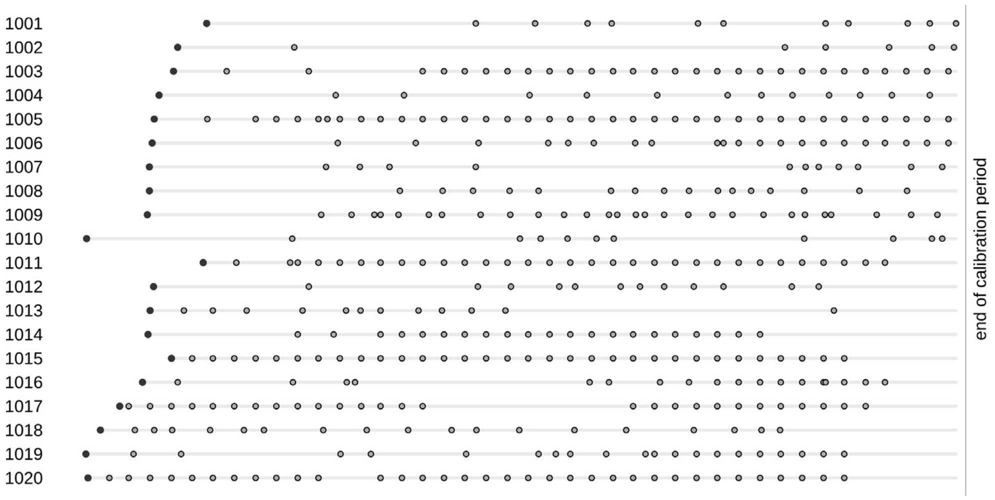


Fig. 1. Transaction timing plots of weekly Charity Contributions - twenty example individuals.

group of the customer-base) and calendar time without prior domain knowledge. We explore the capabilities of this novel forecasting approach to customer base analysis in detail, and benchmark the proposed model against established probabilistic models with latent attrition, as well as a non-parametric approach based on Gaussian process priors, in very diverse non-contractual retail and charity scenarios. Our model raises the bar in predictive accuracy on both the individual customer and the cohort level, automatically capturing seasonal and other temporal patterns.

The paper proceeds as follows. In the next section, we first briefly outline the main characteristics of our approach to learning from and predicting sequential customer behavior. We also relate back to prior work using deep learning and of relevance for customer base analysis. Then we introduce our proposed deep learning based model architecture, training and inference methods. Next, we test the model's forecasting ability in eight real-life empirical settings that vary in terms of cohort size, customer attrition, (ir)regularity of inter-event timing patterns, transaction frequency and seasonal variance, as well as availability of contextual information. Together with the baseline model of simple transaction events, we also show how to easily extend the model with time-varying and time-invariant customer covariates, to further benefit in predictive accuracy. We perform robustness checks by varying model training input and output prediction length, we study individual and group behavior, and demonstrate how these forecasts bring new opportunities for post hoc analysis. Finally, we discuss the merits and limitations of the proposed approach and offer suggestions for further research.

2. Modelling approach

Based on our initial discussion, an ideal model for customer base analysis in data-rich environments would combine a robust forecasting capability both in the short and in the long-term with limited engineering requirements at low computational cost and providing a direct link towards managerial decision-making. Recognizing that traditional statistical forecasting models often suffer from poor efficiency when increasing model complexity and heavily rely on manual feature engineering and data labeling, Table 1 picks up these issues and compares some of the key differences between stochastic BTYD models and the deep learning approach we present in this section.

To circumvent additional feature engineering when increasing model flexibility, Salehinejad and Rahnamayan (2016) and Mena, Caigny, Coussement, Bock, and Lessmann (2019) have introduced a *recurrent neural network* (RNN)[2] approach to the domain of customer base analysis by modeling the evolution of RFM variables over time. However, since the focus still remains on predicting hand-engineered RFM metrics, such an approach does not fully leverage the automatic feature extraction capabilities of deep learning methods. Sheil, Rana, and Reilly (2018) take this one step further by allowing the neural network to derive its own internal representation of transaction histories. The authors demonstrate the performance of several RNN architectures and benchmark them against more conventional machine learning approaches for predicting purchasing intent. In a similar context, Toth, Tan, Di Fabbrizio, and Datta (2017) have shown that a mixture of RNNs can approximate several complex functions simultaneously. More recently, Sarkar and De Bruyn (2021) demonstrate that a special RNN type can help marketing response modelers to benefit from the multitude of inter-temporal customer-firm interactions accompanying observed transaction flows for predicting the most likely next customer action. However, their approach is limited to single point, next-step predictions and to continue with such forecasts into the long-run one must estimate the new model repeatedly with each additional future time step.

[2] In contrast to the more common "feed-forward" neural network where signals propagate from model inputs to outputs all in a single direction, an RNN is a type of neural network that allows previous signals to feed back and combine with the subsequent input. Letting past signals influence future ones means it naturally fits the task of modelling future behavior based on previous history.

Table 1
Comparison of approaches for customer base analysis using historical event data.

	Stochastic BTYD models	*Proposed deep learning approach*
Model assumptions	rigid, probabilistic process parameters	flexible, data-driven machine learning
Transaction flow representation	compressed into simple summary statistics	full flow of historic event history captured
Feature engineering and data labeling	hand-crafted features by model user	no ex-ante labeling, automatic feature detection
Incorporating multiple data streams and (static/dynamic) marketing covariates	increases model complexity and computational cost	easy and highly flexible at no significant additional cost
Detecting purchase dynamics in event histories	complicated, requires additional assumptions and model complexity	flexible, emerges as part of automatic feature detection
Accounting for customer heterogeneity	model-based, parameter estimates of mixture distribution(s)	implicit, distributed across network weight vectors
Detail and quality of forecasts (forecasting quantities)	less detailed summary statistics	fine granular transaction flow characteristics
Prediction horizon of forecasts	focus on long-term predictions	both immediate (next transaction) and long-term perspective

We take a different approach inspired by self-supervised[3] *sequence-to-sequence* model architectures developed originally for Natural Language Processing (NLP) tasks like text generation (Sutskever, Vinyals, & Le, 2014), epitomized by the Google Translate application. The name, often shortened to *seq2seq*, comes from the fact that these models can translate a sequence of input elements into a sequence of outputs. Different *seq2seq* models can be created depending on how we manipulate the input data; i.e., we can conceal certain parts of the input sequence and train the model to predict what is missing, to “fill in the blanks”. If we always blank only the last element in a historical sequence, the model effectively learns to predict the most likely future, conditioned on the observed past. Applying this idea to customer transaction records, we can forecast sequences predicting future behavior. We next present our model architecture in detail.

2.1. Model architecture

To forecast future customer behavior, our model is trained using individual sequences of past transaction events, i.e., chronological accounts of a customer’s lifetime. The example in Table 2 describes one such customer’s transaction history over seven consecutive discrete time periods.[4] This particular individual makes a transaction in the first week, followed by one week of inactivity, then transacting for two consecutive weeks, and so on; in weeks 3 and 4 they also received some form of a marketing appeal. The two calendar components – the month and week indicators – represent time-varying contextual information which is shared across the individuals within a given cohort. In addition, in this example we include also an individual time-invariant covariate (gender) and a time-varying, individual-level covariate (marketing appeals). This particular customer history can then be represented as a sequence of vectors with five elements: the input variable plus the four covariates. Individual-level covariates are strictly optional – in our empirical study, the **Base** model is built without any such variables. Whenever individual covariates are included, we label the model **Extended**. Note that the model is completely agnostic about further extensions: all individual-level, cohort-level, time-varying, or time-invariant covariates are simply encoded as categorical input variables, and are handled equally by the model. This property makes our model extremely flexible in dealing with diverse customer behaviors observed across multiple contexts and platforms.

A schematic high-level representation of the proposed model architecture is shown in Fig. 2. The structure of the model begins with its input layers[5] for (i) the input variable (i.e., transaction counts) and (ii) optional covariates (time-invariant or time-varying inputs). These variable inputs enter the model through dedicated input layers at the top of the model’s architecture and are combined by simply concatenating them into a single long vector. This input signal then propagates through a series of intermediate layers including a specialized LSTM, or *Long Short-Term Memory* RNN neural network component[6], which stores a dynamically updated internal representation of the input sequence presented to the network thus far. This non-stationary representation of the customer’s behavior is distributed across numerous *cell state* values in the LSTM compo-

[3] A *self-supervised* model is learning to perform a task without a set of true labels being provided, typically by learning to reconstruct the input data which has been corrupted by noise or manipulated in another way.

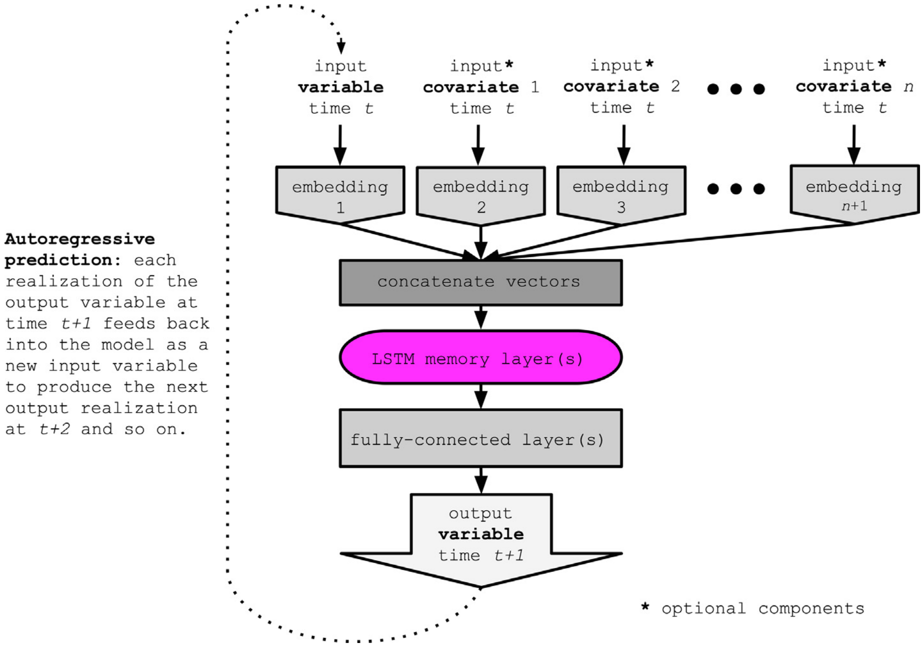
[4] We chose the default unit of time in our empirical studies being one week and aggregate input data into weekly buckets, because such a granularity translates into convenient input data and model size while preserving a high level of dynamic detail. In most practical situations, the choice of granularity level will be guided by managerial considerations. In our empirical model evaluation we will illustrate this using monthly aggregation.

[5] A “layer” is a group of artificial neurons that process signals in parallel. Individual neurons combine signals from a previous layer and transform the result with a non-linear function before passing as output to the neurons in the following layer. A typical network has at least one input layer and an output layer, with several “hidden” layers in between, together forming the “depth” of a neural network, which is why the term “deep learning” is used in the context of neural networks with many layers. We can think of these layers as steps in an execution of a highly parallel computer program.

[6] Note that this memory module is typically composed of several stacked layers of LSTM cells, each layer passing its results to the next one. In Appendix Section A we present details of the properties of the specialized LSTM neural network in the context of modeling customers’ transaction sequences. It also gives name to our proposed model: the Base LSTM model of the transaction stream variable only, and the Extended LSTM with covariates.

Table 2
Transaction history of one sample customer used for network training.

	input	input	input	input	input	output
	individual variable	shared covariate	shared covariate	individual covariate	individual covariate	individual variable
	transactions (time t)	**month**	**week**	**gender**	**appeals**	**transactions** (time t+1)
↓ **time**	**1**	January	1	F	0	**0**
	0	January	2	F	0	**1**
	1	January	3	F	1	**1**
	1	January	4	F	1	**0**
	0	February	5	F	0	**0**
	0	February	6	F	0	**1**
	1	February	7	F	0	**0**



Autoregressive prediction: each realization of the output variable at time *t+1* feeds back into the model as a new input variable to produce the next output realization at *t+2* and so on.

* optional components

Fig. 2. Network schema.

nent layer(s). We then expand the model's computational capacity even further by propagating the LSTM component output through several fully-connected[7] layers.

Finally an output prediction[8] is produced by a *softmax* layer, which normalizes the raw output z derived by a fully connected, k neuron-sized layer into a multinomial (categorical) probability distribution over the set of k classes[9]. These k classes correspond to our target variable and all its k observed outcomes, in our example: *How many transactions will an individual make during the next time period?* Each class label therefore corresponds to the probability of observing $i - 1$ ($i = 1, \ldots, k$) transactions in the next unit of time. Similar to a multinomial logit regression the softmax normalization is given by:

$$softmax_i(z) = \frac{e^{z_i}}{\sum_{i=1}^{k} e^{z_i}}, \tag{1}$$

[7] A fully-connected layer, often also called "dense", is a neural network layer where each of m inputs units connects to each of n output units, forming $m * n$ weighted connections (the trainable parameters of the network).

[8] This model setup can be extended to predict multiple output variables, such as monetary transaction values or intertransaction timing. However, in our empirical study we focus on forecasting a single variable: the number of transactions within a discrete time period, which serves us best for a like-for-like comparison with established benchmark methods.

[9] Softmax is the recommended choice if the goal is to approximate a probability distribution, because of the favourable properties of the error gradient, helping the model adjust incorrect outputs faster (Goodfellow, Bengio, & Courville, 2016).

and the output of the softmax layer at any given time step t is a k-tuple $(p(x_t = c_1), p(x_t = c_2), \ldots, p(x_t = c_k))$ for the probability distribution across the k neurons of the output layer. We set the number of neurons k in the softmax layer to reflect the transaction counts observed across all individuals in the training data: as is the case with any "forward-looking" approach, the model can only learn from events that are observed at some point during estimation; i.e., if in the calibration period individuals only make between zero and three transactions during any of the discrete time periods, then a softmax layer with four neurons is sufficient: the neurons' respective outputs represent the inferred probability of zero, one, two and three transactions.[10]

With each vector read as input, the model's training objective is to predict the target variable, which in this self-supervised training setup is just the input variable shifted by a single time step. Using the example from Table 2, given the sequence of input vectors starting with the first week of January, i.e. `[1,January,1,F,0]`, `[0,January,2,F,0]`, `[1,January,3,F,1]` ..., we train the model to output the target sequence `0,1,1,`... equal to the rightmost column in Table 2. With each input vector processed by the network, the internal memory component is trained to update a real-valued *cell state* vector to reflect the sequence of events thus far.

We estimate the model parameters by minimizing the stochastic mini-batch[11] error between the predicted output and the actual target values. At the time of prediction, we fix the model parameters in the form of weights and biases between the individual neurons in the deep neural network, but the cell state vector built into the structure of the LSTM "memory" component is nonetheless being updated at each step with parts of the latest input, which helps the model learn very long-term transaction patterns. Each prediction is generated by drawing a sample from the multinomial output distribution calculated by the bottom network layer; our model therefore does not produce point or interval estimates, each output is a simulated draw[12]. Each time a draw from this multinomial distribution is made, the observation is fed back into the model as the new transaction variable input in order to generate the following time step prediction, and so on, until we create a sequence of predicted time steps of desired length. This so-called *autoregressive* mechanism in which an output value always becomes the new input is illustrated in Fig. 2 with the dotted arrow bending from the output layer back to the input. Fig. 2 also shows that we feed each input first into a dedicated *embedding* (Mikolov, Sutskever, Chen, Corrado, & Dean, 2013)[13] layer. Using embeddings is not critical to our approach, but by creating efficient and dense (real-valued) vector representations of all variables it already serves to better separate useful signals from noise and to condense the information even before it reaches the memory component (see also Chamberlain, Cardoso, & A (2017) for a similar approach). It should be highlighted that this setup of inputs with associated embeddings is completely flexible and allows for the inclusion of *any* time-varying context or customer-specific static variables by simply adding more inputs together with their respective embedding layers.

2.2. Formal model

The goal is to generate forecasts for sequences of n elements $(X_1, \ldots, X_n)$ where for each time step t the forecast X_t is a multinomial random variable with k possible outcomes $c_1, \ldots, c_k$. We train the model to ultimately make inferences about the conditional probabilities $p(X_t = c_i | X_1 = x_1, \ldots, X_{t-1} = x_{t-1})$ for each time step t and class c_i, where $x_1, \ldots, x_{t-1}$ are the realizations of $X_1, \ldots, X_{t-1}$ at prior time steps. For generated sequences of length n, the joint distribution[14] can then be reconstructed via the chain rule:

$$p(x_1, \ldots, x_n) = \prod_{t=2}^{n} p(X_t = x_t | X_1 = x_1, \ldots, X_{t-1} = x_{t-1}) \times p(X_1 = x_1) \tag{2}$$

During training the model learns to output a multinomial probability distribution so that it maximizes the likelihood of reproducing the true class labels. In a forward pass, the model outputs the predicted distribution $p_\theta(c_i | x_1, \ldots, x_{t-1})$ where θ represents the set of all parameters of the model at this point, and c_i ranges over the k possible outputs at time step t. The loss of a given input sample is then given by the negative log-likelihood of the joint distribution:

$$\mathscr{L} = -\sum_{i=1}^{k} I(x_t = c_i) \times log(p_\theta(c_i | x_1, \ldots, x_{t-1})) \tag{3.1}$$

where I is an indicator function that is equal to 1 when $x_t = i$ and 0 otherwise, and x_t is the realization of X_t, therefore:

[10] We can easily relax this assumption and extend the range of model outputs, for example, to anticipate future additional training examples where values of k might be greater, to later use such an expanded dataset to fine-tune a pre-trained model. When no examples of a given category are observed during training, it is easy for the model to learn that the output probability of such an event is always equal to zero, and the associated part of the model capacity will effectively remain unused.

[11] A *mini-batch* refers to a small subset of training samples we use to calculate the error gradient determining how to adjust the model parameters - the smaller the subset, the less representative and more noisy the gradient.

[12] To make the predicted transaction sequences robust against sampling noise, we repeat this process for each customer several times and take the mean expected number of transactions in a given time step as our final result. We describe how this benefits the prediction accuracy in the Appendix Section B.3

[13] Embedding layers are used to reduce data dimensionality, compressing large vectors of values into relatively smaller ones, to both reduce noise and limit the number of model parameters required.

[14] The product contains a term for every t from 1 to n, either unconditional for $t = 1$ or conditional for $t \geqslant 2$. The reason it starts at $t = 2$ is that only then we need conditional probabilities. We add the final term explicitly: $p(X_1 = x_1)$.

$$\mathscr{L} = -\log(p_\theta(x_t|x_1,\ldots,x_{t-1})) \tag{3.2}$$

Each iteration of the model training begins with the network calculating the predicted output for a small batch of input samples, followed then by the adjustment of all model parameters (weights and biases) using the gradient of the error, starting with the parameters in the final output layer and then propagating backwards through all preceding network layers (Rumelhart, Hinton, & Williams, 1986). After each input sample has been used to update the network parameters once – a segment of training called an *epoch* – we monitor the progress of the training by calculating the *validation error* using a separate set of input samples, which are otherwise not used during training. Although the validation loss is not a perfect approximation of the model's predictive performance, we show in Appendix C that models with lower validation loss often produce more accurate holdout predictions. As long as the validation error keeps decreasing, we continue iterating over batches and epochs, until we reach a minimum. More details about the model's technical implementation, the training and inference procedure, are provided in Appendix B.

2.3. Autoregressive simulation of predictions and endogeneity

Our model lets us simulate future transactions of an individual step-by-step by drawing from the output probability distribution conditional on a previously observed sequence of input time steps. To generate a holdout simulation, we must first input the individual's calibration input sequence into the model, one time step after another. Intuitively, this brings the model's internal state – including the values stored as the *cell state* in the LSTM memory module – to represent the observed customer history. Once the final step of the calibration has been processed, we draw a sample from the output multinomial distribution to create the first holdout prediction time step. This realization feeds back into the model as a new input to produce the next time step prediction and so on, until we create a sequence of interdependent simulated draws of desired length. If the model is trained with additional covariates, we input their values into the model at each simulation time step. Conditioned with the observed covariate values the model simulates the actual holdout as we demonstrate in our empirical study, or one can specify custom covariate values to simulate "what-if" scenarios. We include two examples of such future simulation scenarios conditioned with predefined holdout covariate schedules in the Appendix Section E.

Marketing managers are particularly interested in accounting for marketing interventions when evaluating future decision-making scenarios. The Charity Contributions setting we introduce in the next section includes the contact records from a major US nonprofit organization where reverse causality cannot be completely excluded. Even though we are not aware of any rules applied in this particular business setting, there are cases where not only the donors are affected by the appeals from the charity, but the charity manager might also tailor the marketing interventions according to some targeting rule that is a function of previous charitable transactions. Such situations would imply that the marketing variable is likely endogenous.[15] Endogeneity is important to consider when the focus is on disentangling causal relationships among the variables involved and to derive policy-making implications. However, our key objective is to improve holdout predictive performance, rather than building a descriptive model. As demonstrated already by Ebbes, Papies, and van Heerde (2011, p. 1116), holdout sample validation and superior predictive performance favors regression estimates that are not corrected for endogeneity. Similarly, Schweidel and Knox (2013, p. 479) and more recently McCarthy and Fader (2018) and Bachmann et al. (2021) also find that controlling for endogeneity is of minor importance when high predictive accuracy of future forecasts is the primary objective. However, as opposed to forecasts made by BTYD models that are only conditioned on observed transaction data, we acknowledge that when it comes to planning conditional future marketing actions a more sophisticated approach is necessary than our autoregressive simulation setup. This is particularly relevant in cases where marketing actions are conditioned on previous customer activities, which requires to explicitly link marketing outcome variables to a policy function.

3. Empirical performance evaluation

In this section, we introduce the datasets, present the evaluation metrics, and describe the benchmark models.

3.1. Datasets

To demonstrate the capability of the LSTM approach as a forecasting tool for sequential data and to show its flexibility in incorporating covariates, we select a broad range of empirical scenarios. Two datasets represent philanthropic behavior (Charity Contributions and Blood Donations), the remaining six are purchase transaction datasets with varying calibration lengths, transaction frequencies, and cohort sizes (Electronics Retailer, Multichannel Merchant, CDNOW, Groceries, Yogurt and Sunscreen Purchases). The key descriptive statistics summarized in Table 3 already point at substantial differences between the datasets: the Yogurt Purchases dataset is characterized by high transaction frequency with only moderate levels of inactivity, others represent transactional data for customer cohorts with lower activity rates – a familiar setting for customer base analysts. The Charity Contributions and Multichannel Merchant settings allow us to study the influence of individual-level time-varying marketing interventions, and in the Electronics Retailer data case we observe a rich set of

[15] The other scenario in our empirical applications section with marketing interventions, the Multichannel Retailer dataset, is using pre-scheduled catalog mailings, where endogeneity is certainly not present.

Table 3
Descriptive statistics of the used datasets.

				Calibration period				Holdout period		
Dataset	Cohort size	Clumpy	r_{WM}	Seasonality	Length (weeks)	Mean events	Non-repeaters	Length (weeks)	Mean events	Inactive
Charity Contributions	21,166	2%	2.1	0.8	181	2.2	52%	53	0.3	81%
Electronics Retailer	3,782	24%	0.5	0.3	260	4.5	23%	53	0.6	72%
Blood Donations	11,887	1%	2.2	0.4	116	1.9	55%	105	0.8	67%
CDNOW	23,570	8%	0.9	0.3	39	2.1	59%	39	0.9	70%
Groceries	1,525	24%	1.8	0.3	52	4.7	38%	52	2.2	66%
Yogurt Purchases	31,216	52%	1.5	0.1	156	43.7	6%	52	10.6	46%
Multichannel Merchant	1,379	10%	0.9	0.4	52	1.2	83%	338	0.8	69%
Sunscreen	7,794	3%	0.8	1.6	104	1.8	57%	104	0.8	61%

individual-level customer covariates and background information. These eight real-world empirical settings allow us to demonstrate the robustness of the proposed method.[16]

In addition to the descriptive statistics provided by Table 3, for each dataset we also provide a cohort-level summary statistic r_{WM} proposed by Wheat and Morrison (1990) for measuring inter-event timing regularity, calculated as implemented in the R package `BTYDplus` (Platzer, 2016), as well as the proportion of individuals classified as “clumpy” according to the measure proposed by Zhang et al. (2015).[17] To summarize the seasonal profile of a given transaction setting, we calculate the Seasonality statistic for aggregate weekly transactions during the observed calibration period, calculated as the mean absolute deviation (MAD) from the median weekly aggregate repeat customer transactions (we exclude initial transactions to better capture repeat customers):

$$Seasonality = \frac{1}{n}\sum_{t=1}^{n}\frac{|Ar_t - Ar_{median}|}{Ar_{median}}, \qquad (4)$$

where Ar_{median} is the median of repeat weekly transactions during calibration, Ar_t is the actual volume of transactions at time t, and n is the number of discrete time periods in the calibration period. A transaction setting exhibiting a Seasonality score of less than 0.5 can then be said to have low seasonality (e.g., CDNOW), a Seasonality between 0.5–1.0 describes a mildly seasonal setting (e.g., Charity Contributions), and a Seasonality above 1.0 indicates a strongly seasonal transaction scenario (e.g., Sunscreen).

The used datasets are characterized as follows:

- *Charity Contributions*: A widely used benchmark dataset examined extensively in the customer base analysis literature (see, e.g., Schweidel & Knox (2013) and Platzer & Reutterer (2016)), provided by the Direct Marketing Educational Foundation (see Malthouse (2009) for more details). This dataset contains the contribution histories of a cohort of 21,166 donors to a large U.S. charity organization acquired during the first half of 2002 and observed over a time span of 4.5 years, as well as the associated individual direct marketing solicitation records. It is characterised by a low holdout activity of just 0.3 donations on average together with a relatively high Seasonality of 0.8. The r_{WM} statistic of 2.1 points at a significant degree of timing regularity, which is confirmed by a very low share of “clumpy” customers.
- *Consumer Electronics Retailer*: 3,782 individual household transaction records from a major U.S. durable goods retailer observed from December 1998 through November 2004. The quasi cohort is part of the ISMS durable goods dataset (Ni, Neslin, & Sun, 2012) and consists of customers who made their first transaction during the first year of the observation period. For each household we observe demographic covariates such as age, income, and gender of the head of the household, the number of children present, their age and gender. With a r_{WM} statistic of 0.5 this dataset is an example of irregular transaction behavior.
- *Multichannel Merchant*: A set of transaction histories collected from 1,379 customers of a multichannel catalog merchant. This cohort was examined recently by Bachmann et al. (2021); the data also include time-varying catalog mailings sent to individual clients during the observation period. All customers made their first purchase during the initial six months of the observation period and were observed from the beginning of 2005 until mid-2012. With a r_{WM} parameter of 0.9 the inter-transaction times are close to being randomly distributed. This very low transaction frequency scenario allows us to showcase the longest prediction period of 6.5 years (338 weeks).

[16] Furthermore, in Appendix Section D we present a two by two simulation study using synthetic transaction data generated by the Pareto/NBD (Schmittlein et al., 1987), the canonical BTYD model which also serves us as one of our benchmark models.

[17] *Clumpy* or also “binging” transaction behavior is characterized by inter-event timing patterns that are *clumped* together and typically emerges from switching between “hot” and “cold” activity states (see also Schwartz, Bradlow, & Fader (2014))

- *Blood Donations*: A cohort of 11,887 new blood donors who donated for the first time during the year 2015 and were observed through March 2019. The data are provided by a national branch of the Red Cross organization for the purpose of this study. For this dataset, we used a longer holdout period of 2 years (107 weeks). Similar to the other charitable dataset, the r_{WM} regularity parameter is 2.2, i.e., this cohort is characterized by rather regular donors.
- *CDNOW*: Another well-known dataset from an online music store which includes 23,570 customers acquired in the first quarter of 1997 and observed over the following 1.5 years. This dataset has been benchmarked extensively also in Fader et al. (2005), Abe (2009), Zhang et al. (2015), Platzer and Reutterer (2016). With a calibration period of just 9 months (the shortest in this study), 70% inactivity in the holdout, a r_{WM} parameter of 0.9 indicating random inter-event timing behavior overall, this dataset poses a challenge even for advanced forecasting methods.
- *Groceries*: A quasi cohort was constructed in this case since customer acquisition dates were not available: Following the prescription in Platzer and Reutterer (2016) we select customers who made a first recorded transaction during the initial 3 months of 2006, but have not been active in the 2 years prior. The r_{WM} parameter of 1.8 suggests a significant regularity in the purchasing behavior. This is the smallest dataset in this study, with 1,525 individuals, and also one of the shortest, with a calibration of just 1 year.
- *Yogurt Purchases*: A dataset of household purchase records is provided by a global consumer research company. We constructed a quasi cohort of 31,216 households (the largest in this empirical study) by selecting those with first recorded transaction made within the first observed year, giving us a total calibration period of three years and a one year holdout. Compared to other datasets, these transactional data are characterized by higher activity rates. An r_{WM} statistic of 1.5 indicates customers' Yogurt Purchase timings being more regular than random.
- *Sunscreen*: A cohort containing 7,794 customers who first purchased a bottle of sunscreen during the year of 2015. Sunscreen is a highly seasonal product (Seasonality score of 1.6), with the vast majority of transactions occurring during the summer months. However, transaction frequencies are very low (only 1.8 mean transactions during the estimation period) and according to a r_{WM} parameter value of 0.8 more irregular at the cohort level. We use a 2-year calibration period and predict the behavior in the following 2 years.

3.2. Implementation and benchmarks

To demonstrate how we estimate our model and generate forecasts we provide an open-source reference implementation available at https://github.com/valendin/rfm2lstm. The predictive model can be estimated in minutes on a regular laptop computer and moving from the base to the extended LSTM model requires no significant additional modelling effort. To evaluate the performance of our approach, we use three established benchmark models that are readily available as well-documented open-source analytical tools. These models are as follows:

- The "workhorse" probability model for customer base analysis, the Pareto/NBD (Schmittlein et al., 1987), estimated by the hierarchical Bayesian implementation using Markov Chain Monte Carlo (MCMC) sampling of model parameters from the R package `BTYDplus` (Platzer, 2016).
- A generalization of the Pareto/NBD model, the Pareto/GGG (Platzer & Reutterer, 2016), which can leverage individual differences in inter-event timing patterns for improved holdout predictions in the presence of timing regularity in the data. Here we also used the MCMC implementation from the `BTYDplus` package.
- A Bayesian non-parametric approach to customer base analysis, the Gaussian Process Propensity Model (GPPM; Dew & Ansari (2018)). This model integrates two sets of predictors modelled through latent functions that jointly determine transaction propensity related to calendar time, inter-event time, and customer lifetime. We used the Stan implementation provided in the Web Appendix C of Dew and Ansari (2018).[18]

3.3. Evaluation metrics

To evaluate the predictive performance at the individual level, we report the Root Mean Squared Error (RMSE) as a preferred metric when the aim is to minimize the *mean* forecasting error, which is the case for methods examined here.[19] The RMSE also penalizes larger errors more than smaller ones, which is particularly relevant in scenarios with low event frequencies. For example, in our weekly-aggregated Charity Contributions setting, taking the median value implies forecasting mostly zero events for all individuals, a very poor prediction, which would nevertheless "outperform" all models in this study (with the exception of the LSTM) in terms of MAE.

[18] During the GPPM estimation we faced a performance bottleneck related to Hamiltonian Monte Carlo (HMC) sampling, a serial process from complexity class $\mathcal{O}(n^3)$ where n is the length of calibration. For the Yogurt dataset with 159 calibration time steps, 800 HMC iterations took approximately 8 days on a 5 GHz CPU. This can be circumvented with sub-sampling at some accuracy cost. Because in the Yogurt dataset the total number of transactions strongly correlates with customer lifetime and calendar time, we also removed the lifetime component from the model, otherwise it would not converge. For other datasets, such manual model adjustments are not necessary, however may potentially be beneficial.

[19] The often used Mean Absolute Error (MAE) is only appropriate for forecasts of the median (Hanley, Joseph, Platt, Chung, & Belisle, 2001).

At the aggregate level, we report the forecast bias calculated as the percentage of over- or under-forecasting of the observed accumulated weekly transactions. Unbiased forecasts are considered important to correctly estimate future revenue streams and to facilitate related demand planning. Further, an accurate estimate of the total aggregate customer transactions expected for customer cohorts in a given time horizon is an important indicator for deriving the overall value of a company's customer base (Blattberg & Deighton, 1996; McCarthy et al., 2017).

Weekly transaction volumes often show strong seasonal or other cyclical patterns. To assess how accurately the overall seasonal pattern of transaction flows is captured, we propose the Mean Absolute Percentage Error (MAPE). MAPE is a measure of *variance*, calculated as the mean difference between the actual A_t and predicted P_t transactions at time unit t:

$$M = \frac{100\%}{n}\sum_{t=1}^{n}\left|\frac{A_t - P_t}{A_{mean}}\right| = 100\% \frac{\sum_{t=1}^{n}|A_t - P_t|}{\sum_{t=1}^{n}A_t} \tag{5}$$

The importance of forecasts showing low values of MAPE can hardly be overstated. Managers determine stock based on expected demand and unavailable products may lead to customer switching and lost profits, while excess unsold inventory carries additional labour and storing capacity costs and possibly also losses due to perishability (Balachander & Farquhar, 1994). A lower MAPE also implies a more accurate anticipation of aggregate market volatility, which helps to reduce such over- and under-stocking costs compared to a high MAPE prediction.

4. Results

In this section we report the findings from the performance analysis of our proposed model when applied to the above-described datasets. The evaluation strategy is as follows: In each transaction setting we train a Base LSTM model using the customer event timing sequences alone (together with the calendar time information) and compare the results with those derived from the benchmark models. Where additional covariates are available, we train an additional Extended LSTM model using the training data enriched with individual time-varying covariates (Charity Contributions & Multichannel Merchant *with actual marketing appeals*) or static customer demographic information (Electronic Retailer *with time-invariant covariates*).[20]

A performance summary of our empirical benchmarking study is given in Table 4, with the top score per dataset and performance metric in **bold**, highlighting the leading performance of the LSTM model across all three metrics (RMSE, bias, MAPE). Tables 12 and 13 in Appendix Section F document the performance lift in more detail.

4.1. Individual-level predictions

Next, we take a closer look at the predictive performance of our proposed model at the individual level and discuss how well the LSTM model does to accurately spot customer groups important to the firm. Let us consider the example customers from the Charity Contributions setting we already highlighted in Setion 1. Upon inspection of the Actual Scenario in their respective transaction timing plots in Fig. 3, sub-plot (a), we note that these two groups of customers indeed differ in their holdout period activity: First, individuals 1001 to 1010 who begin their relationship with the firm as sporadic customers, but end up developing an ever more frequent transaction habit as time goes on. Since these customers turn out to be among the firm's future premium customers, they present an opportunity for the manager who wants to ensure this favourable relationship continues to strengthen, perhaps by providing them with the best customer experience possible, or simply by keeping up the current level of service. For the purposes of this study, we define the **opportunity** group as customers who make transactions more frequently in the holdout than during the calibration period.

On the other hand, the remaining individuals 1011–1020: Recurring customers historically, who however turn out to have defected already at the time of the forecast. Losing such previously frequent returning customers means the firm incurs a significant loss, and it is of crucial importance for the manager to know, as soon as possible, whether they are likely to churn. We call this group customers **at risk**, and define it as those individuals with at least average historical transaction frequency who do not return to make a single transaction in the holdout period.

4.1.1. Opportunity customers

Across our eight real-life scenarios the opportunity group accounts for over 59% of total holdout period transactions. The first ten individual members of the Charity Contribution scenario depicted in Fig. 3 (samples 1001–1010) belong to this customer group and together account for 99 transactions during the holdout period. While the Pareto/NBD attributes a significant number of transactions to these customers, it is not able to make a proper distinction between this and the second group of customers at risk (samples 1011–1020), who in fact remain inactive in the holdout: The Pareto/NBD estimates less

[20] Additionally, we present the result of a monthly-aggregated model in the case of Charity Contributions to investigate the effects of input data granularity.

Table 4
Results summary: RMSE, bias, MAPE. Best results per dataset in bold.

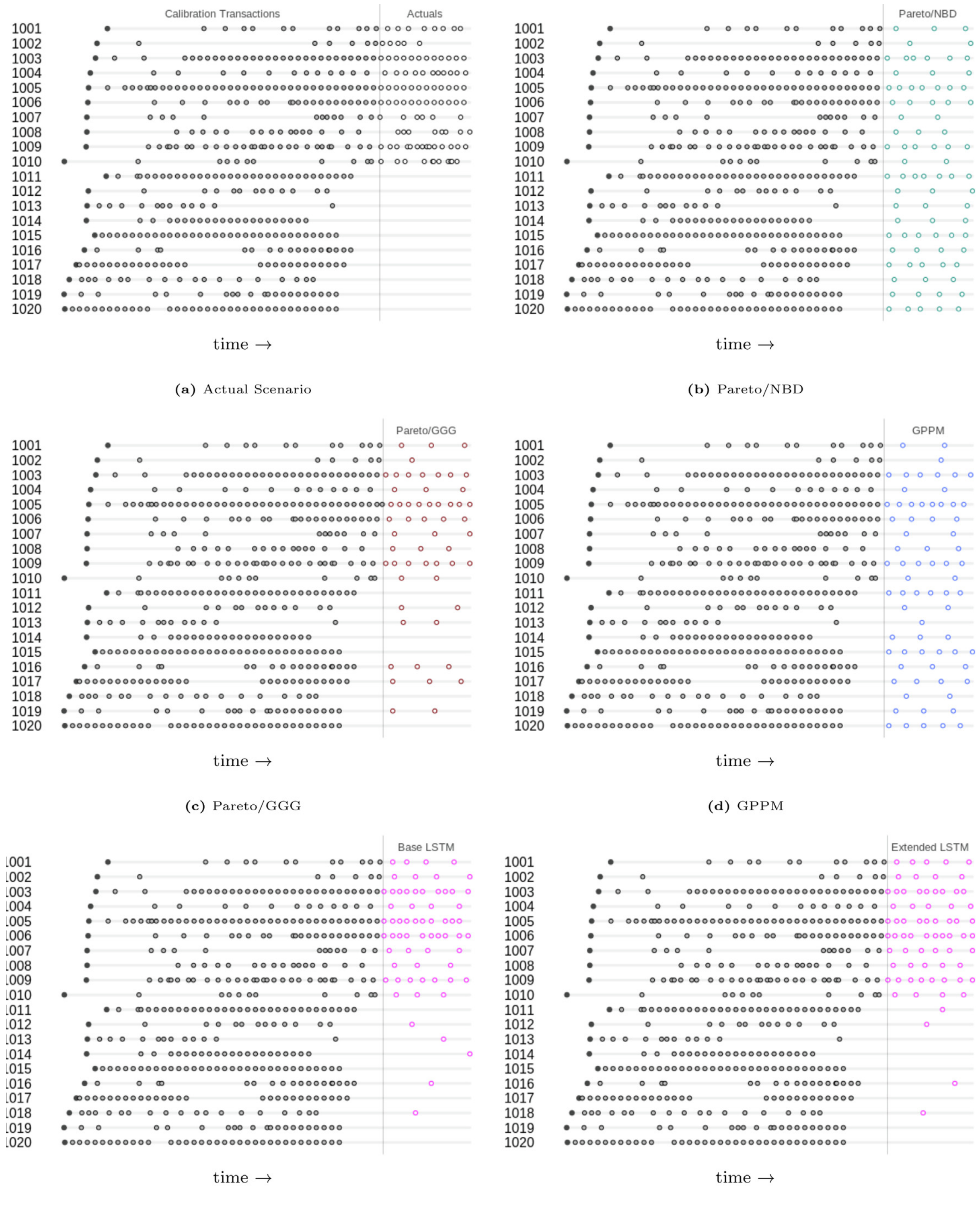
Dataset	Model	RMSE	bias (%)	MAPE (%)
Charity Contributions	Base LSTM	0.62	-2.8	29.3
monthly aggregation	Base LSTM	0.62	-2.9	16.2*
with actual marketing appeals	Extended LSTM	**0.58**	**0.8**	**27.2**
	Pareto/NBD	0.65	16.1	57.4
	Pareto/GGG	0.63	-7.8	49.6
	GPPM	0.69	-32.7	47.6
Electronics Retailer	Base LSTM	1.18	2.7	16.9
with time-invariant covariates	Extended LSTM	**1.17**	**0.1**	**15.5**
	Pareto/NBD	1.26	-14.8	32.2
	Pareto/GGG	1.25	-4.1	36.4
	GPPM	1.27	33.9	51.5
Multichannel Merchant	Base LSTM	2.00	-5.8	54.9
with actual marketing appeals	Extended LSTM	**1.79**	**1.2**	**51.2**
	Pareto/NBD	2.06	17.0	105.9
	Pareto/GGG	2.06	-48.0	90.6
	GPPM	1.99	-9.5	60.6
Blood Donations	Base LSTM	**1.28**	**1.7**	**16.2**
	Pareto/NBD	1.31	26.1	28.9
	Pareto/GGG	1.29	-10.7	16.5
	GPPM	1.31	28.4	32.0
Sunscreen	Base LSTM	**1.20**	**2.1**	**32.1**
	Pareto/NBD	1.21	-21.4	78.8
	Pareto/GGG	1.21	-9.9	81.6
	GPPM	1.29	52.1	123.6
CDNOW	Base LSTM	**1.86**	**-0.7**	**13.8**
	Pareto/NBD	1.97	-18.9	19.8
	Pareto/GGG	1.97	-19.7	20.3
	GPPM	2.11	-36.6	36.6
Groceries	Base LSTM	**2.90**	**-0.7**	**12.9**
	Pareto/NBD	3.25	20.0	21.4
	Pareto/GGG	3.10	11.8	15.4
	GPPM	3.51	-10.7	17.8
Yogurt Purchases	Base LSTM	**7.33**	**0.6**	**2.9**
	Pareto/NBD	8.22	5.6	8.2
	Pareto/GGG	8.28	3.7	7.2
	GPPM	9.20	-4.8	4.9

* Not directly comparable with the other Charity Contributions models due to different aggregation.

(39) transactions for these ten opportunity group customers than for the other ten customers at risk (43). The Pareto/GGG improves on this result (estimated 44 opportunity transactions vs. only 12 for the at risk group), while the GPPM does not (36 opportunity vs. 35 at risk). The Base LSTM offers a much improved prediction, closer to the actual high opportunity transactions (59), with only 5 transactions for the customers at risk. Leveraging also the records of individual-level marketing interventions allows the Extended LSTM model to refine the result further (65/4).

To generalize this perspective and to examine the models' ability to correctly identify the opportunity customers, we compute F1-scores as a measure of classification accuracy and report them along with Precision and Recall in Table 5. Five times out of eight, the Pareto/NBD and Pareto/GGG fail to identify any such customers (denoted by —), which occurred three times for the GPPM, and in just two cases (Multichannel Merchant and Sunscreen) for the Base LSTM model.[21] Across the eight scenarios, on average, the Base LSTM is 57% better at spotting opportunity customers than the best alternative benchmark model (Pareto/GGG) as per the F1 score. The individual RMSE, aggregate bias, and MAPE scores improve by 3.3%, 2.1% and 7.6%, respectively. These findings illustrate the limitations of Pareto/NBD type models in dealing with non-stationary behavior other than attrition, but such dynamics are easily captured by our neural network model without any additional changes in the basic setup. We also notice an extra boost in predictive performance by moving from the Base LSTM model to the Extended LSTM, meaning that including covariates additionally helps in accurately spotting these customers.

[21] None of the examined models is able to identify any of the handful of opportunity customers in the Multichannel Merchant scenario. Both this and the Sunscreen dataset present a challenging setting due to the extremely low transaction frequency of just 1.2 and 1.8 events during calibration per customer on average, respectively.



(a) Actual Scenario

(b) Pareto/NBD

(c) Pareto/GGG

(d) GPPM

(e) Base LSTM

(f) Extended LSTM

Fig. 3. Individual Charity Contributions: actual, Pareto/NBD, Pareto/GGG, GPPM, Base LSTM, Extended LSTM.

4.1.2. Customers at risk

The Fig. 3 top-left plot (a) of the Actual Scenario shows that the individuals “at risk” (sample 1011–1020) remained inactive throughout the holdout period. This is contrary to the expectations of the Pareto/NBD model which is mislead by their high frequency and relatively low recency and assumes that said customers are still “alive” and thus wrongly estimates a high number of future transactions in the holdout. Consistent with the observation of Platzer and Reutterer (2016) that accounting also for the regularity of purchasing helps to better spot defection, the Pareto/GGG improves on this projection again, while the GPPM does not, but the flexible Base LSTM model spots the change in transaction pattern better still, and the

Table 5
Opportunity customers: F1 score, precision and recall.

Dataset	Model	Precision	Recall	**F1**
Charity Contributions	Base LSTM	.42	.02	.04
with actual marketing appeals	Extended LSTM	.57	.06	**.12**
12% of cohort	Pareto/NBD	—	—	—
	Pareto/GGG	—	—	—
	GPPM	—	—	—
Electronics Retailer	Base LSTM	.18	.29	.22
with time-invariant covariates	Extended LSTM	.25	.27	**.26**
19% of cohort	Pareto/NBD	.43	.05	.08
	Pareto/GGG	.43	.15	.22
	GPPM	.17	.43	.24
Multichannel Merchant	Base LSTM	—	—	—
only 1% of cohort	Pareto/NBD	—	—	—
	Pareto/GGG	—	—	—
	GPPM	—	—	—
Blood Donations	Base LSTM	.53	.01	.01
18% of cohort	Pareto/NBD	—	—	—
	Pareto/GGG	—	—	—
	GPPM	.91	.02	**.04**
Sunscreen	Base LSTM	—	—	—
23% of cohort	Pareto/NBD	—	—	—
	Pareto/GGG	—	—	—
	GPPM	.21	.05	**.08**
CDNOW	Base LSTM	.46	.01	**.01**
16% of cohort	Pareto/NBD	—	—	—
	Pareto/GGG	—	—	—
	GPPM	—	—	—
Groceries	Base LSTM	.40	.04	**.07**
13% of cohort	Pareto/NBD	.75	.03	.06
	Pareto/GGG	.50	.03	.05
	GPPM	1.0	.02	.03
Yogurt Purchases	Base LSTM	.54	.46	**.50**
21% of cohort	Pareto/NBD	.47	.23	.31
	Pareto/GGG	.48	.22	.30
	GPPM	.17	.24	.20
mean total	Base LSTM	.31	.10	**.11**
	Pareto/NBD	.21	.04	.06
	Pareto/GGG	.18	.05	.07
	GPPM	.31	.10	.07

Extended LSTM leverages marketing interventions to improve the forecast further. Note that there is no explicit terminal "churn state" modelled by the LSTM, which acknowledges that there remains "always a share" (Rust, Lemon, & Zeithaml, 2004) in non-contractual business settings, so some probability of the customers being alive is always preserved (e.g., in the case of customers No. 1012, 1013 and 1014).

We summarize the prediction accuracy for the customers "at risk" in Table 6, presenting the performance uplift brought by the Base LSTM model compared to the other three benchmark models, averaged across all eight empirical settings. By grouping individuals inactive in the holdout according to their calibration period transaction frequency, we arrive at five groups of customers at risk.[22] The green shading used to highlight the most improved customer groups reveals the fact that the more frequent the churned customers were during calibration, the more does the Base LSTM model improve the individual prediction accuracy (RMSE) as compared to the benchmark models. This is good news for the manager who wants to protect the firm's relationship with its most frequent customers.

[22] The 50th percentile group is formed of customers inactive during holdout with at least average calibration period frequency, the 33rd percentile group contains the most frequent third of inactive customers, and so on. These five customer subgroups represent 19%, 8%, 7%, 2%, and 1% of total customers respectively averaged across our eight empirical scenarios, accounting for 26%, 16%, 14%, 5%, and 3% of all past (calibration period) transactions on average, respectively.

Table 6
Customers at risk: performance lift of Base LSTM vs benchmark models, broken down by calibration period transaction frequency percentile.

		Base LSTM Performance Lift				
		At Risk Frequency Percentile				
Benchmark Model		50th	33rd	25th	10th	5th
Pareto/NBD	**RMSE** (%)	+18	+18	+19	+20	+24
Pareto/GGG		+7	+7	+8	+11	+13
GPPM		+15	+15	+17	+23	+30
Pareto/NBD	**bias** (pp)	+4	+3	+3	+1	+1
Pareto/GGG		-1	0	0	0	0
GPPM		+4	+2	+2	+1	+1
Pareto/NBD	**MAPE** (pp)	+3	+3	+3	+1	+1
Pareto/GGG		+1	+1	+1	+1	+1
GPPM		+3	+1	+2	+1	+1

4.2. Overall performance and capturing purchase dynamics

As summarized in the results Table 4, we find the LSTM model performs best in all eight empirical scenarios, with most individual-level forecasting improvement in higher frequency transaction settings like Groceries and Yogurt Purchases, and less improvement in low frequency settings like Blood Donations and Sunscreen. We also notice that using monthly instead of weekly aggregation of input sequences in the case of the Charity Contribution dataset does not have a significant effect on the prediction accuracy in terms of RMSE and bias. Overall, individual-level RMSE improves by an average 6% across all 8 settings, compared to the Pareto/NBD (4% and 9% compared to Pareto/GGG and GPPM, respectively). As for the two cohort-level accuracy metrics, the Base LSTM model is particularly strong, with an average aggregate bias of just 2% across all scenarios (18%, 15% and 26% for Pareto/NBD, Pareto/GGG and GPPM, respectively). The MAPE is improved most by the Base LSTM in the more seasonal settings like Multichannel Merchant or Sunscreen, and is reduced by 44% on average overall, compared to the next best benchmark, the Pareto/GGG. This means that on an average week, the LSTM is nearly twice as accurate in predicting the expected aggregate weekly transaction volume. For further comments on the relationship between the characteristics of input data and the LSTM model performance, see Appendix Section F.

Let's further examine the Charity Contribution scenario. The plot in Fig. 4 tracks the actual weekly recurring events (solid dark line) on the cohort level and compares them to the expected aggregated numbers estimated by the Base LSTM model (magenta line), along with the corresponding Pareto/NBD estimates (dashed line).[23] From inspecting the aggregate-level plot it is apparent that the Base LSTM model captures the seasonality pattern including the high variance period around December/January very well, which is reflected in lower MAPE scores, while remaining much less prone to over- and under-forecasting bias than any of the benchmark models. In many similar transactional scenarios, periods of high activity – the "peak season" – represent a very significant portion of yearly revenue. We investigate our model's ability to capture such important temporal features of customer activity in the following subsections.

4.2.1. "Christmas Shopping": Forecasting peak season

Being able to project fluctuations such as seasonality accurately and to capture these patterns automatically *without any a priori definition or labeling* is a powerful core benefit of our LSTM model, because the domain expertise required to define seasonal effects labels manually is at best cumbersome if not impossible to accomplish in any given business scenario. For example, seasonal effects are not necessarily "tied" to calendar time, but often depend on some other factors that are not under the control of the analyst, such as weather or temperature conditions, or the firm's marketing actions. Capturing seasonality, however, can be crucial for the financial success of the firm: under-stocking means you will miss out on potential sales and over-stocking represents wasted resources that could have been productively used elsewhere.

Let's illustrate the benefits of automatic feature learning by zooming in on the season that can make or break consumer-focused companies: holiday shopping. In Table 7 we compare the total actual number of transactions encountered in our eight datasets for the five-week period leading up to the holidays each year (denoted as "Christmas Transactions") with those predicted by the models under investigation for these time windows. As Table 7 clearly documents, frequent, seasonal customer behavior typically associated with holiday shopping is captured extremely well by the Base LSTM model compared to all benchmark models, reducing the overall amount of over- or under-prediction (bias) by double digit percentage points (pp) in every scenario except the Multichannel Merchant dataset (with a smaller improvement of 6 pp). Strongly seasonal scenarios such as Charity Contributions and Sunscreen[24] see the largest improvements of 49 pp and 450 pp, whereas scenarios without a significant seasonal pattern like CDNOW and Groceries only improve by 10 pp and 29 pp, respectively. These results

[23] For the tracking plots of the remaining datasets see the Online Appendix, Fig. 1.

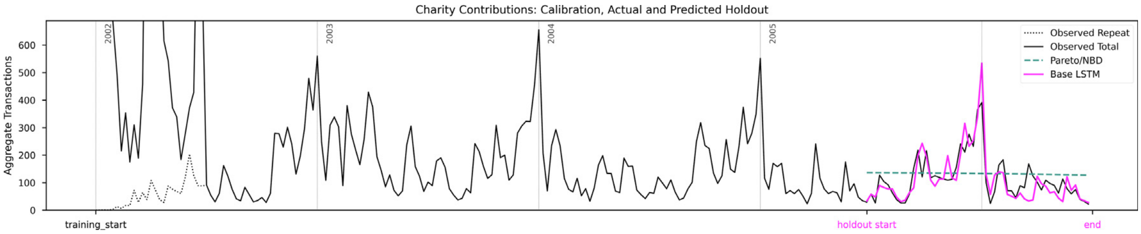


Fig. 4. Tracking plot of weekly Charity Contributions.

Table 7
Christmas Period Forecast: Aggregate transactions, RMSE and bias, LSTM vs benchmark models, best result per dataset in bold.

		Christmas Transactions			
Dataset	Model	Actual	Predicted	RMSE	bias (%)
Charity Contributions	Base LSTM	1326	**1343**	0.23	**+1**
with actual marketing appeals	Extended LSTM	1326	1091	**0.23**	-18
	Pareto/NBD	1326	655	0.25	-51
	Pareto/GGG	1326	524	0.24	-61
	GPPM	1326	447	0.25	-66
Electronics Retailer	Base LSTM	634	**546**	**0.49**	**-14**
with time-invariant covariates	Extended LSTM	634	542	0.49	-15
	Pareto/NBD	634	228	0.52	-64
	Pareto/GGG	634	257	0.51	-60
	GPPM	634	431	0.50	-32
Multichannel Merchant	Base LSTM	176	121	0.44	-31
with actual marketing appeals	Extended LSTM	176	**149**	**0.41**	**-15**
	Pareto/NBD	176	111	0.50	-37
	Pareto/GGG	176	49	0.48	-72
	GPPM	176	99	0.44	-44
Blood Donations	Base LSTM	835	**836**	**0.28**	**0**
	Pareto/NBD	835	1150	0.30	+38
	Pareto/GGG	835	778	0.29	-7
	GPPM	835	1180	0.28	+41
Sunscreen	Base LSTM	70	**117**	**0.10**	**+68**
	Pareto/NBD	70	432	0.18	+517
	Pareto/GGG	70	542	0.21	+674
	GPPM	70	1179	0.19	+1584
CDNOW	Base LSTM	2901	**2564**	**0.44**	**-12**
	Pareto/NBD	2901	2286	0.44	-21
	Pareto/GGG	2901	2266	0.45	-22
	GPPM	2901	1933	0.450	-33
Groceries	Base LSTM	257	**275**	**0.42**	**+7**
	Pareto/NBD	257	349	0.52	+36
	Pareto/GGG	257	311	0.45	+21
	GPPM	257	208	0.45	-19
Yogurt Purchases	Base LSTM	26771	**27392**	**1.13**	**+2**
	Pareto/NBD	26771	32285	1.29	+21
	Pareto/GGG	26771	31717	1.26	+19
	GPPM	26771	26168	1.18	-2

are remarkable, because the deep learning model was not informed that there was something like a ”holiday season” but it directly inferred this from the observed stream of transaction data.

[24] The Sunscreen scenario represents a case where seasonality is “reversed” – most of the product is sold during the summer, with a yearly low around Christmas. This documents our model’s ability to automatically learn to forecast the “valleys” just as well as the “peaks”.

Table 8
When Will They Make The Next Purchase: Mean Next Inter-Transaction Time ($\overline{NITT}$) Prediction, bias, and individual-level NITT RMSE, LSTM vs benchmark models, best result per dataset in bold.

			$\overline{NITT}$ (weeks)		NITT (weeks)
Dataset	Model	Actual	Predicted	bias (%)	RMSE
Charity Contributions	**Base LSTM**	47.3	**47.0**	**-1**	10.7
with actual marketing appeals	**Extended LSTM**	47.3	46.4	-2	**10.3**
	Pareto/NBD	47.3	52.2	+10	13.2
	Pareto/GGG	47.3	52.2	+10	13.2
	GPPM	47.3	52.7	+11	13.8
Electronics Retailer	**Base LSTM**	44.0	42.9	-3	15.8
with time-invariant covariates	**Extended LSTM**	44.0	**43.3**	**-2**	**15.7**
	Pareto/NBD	44.0	50.2	+14	17.4
	Pareto/GGG	44.0	50.1	+14	17.4
	GPPM	44.0	49.9	+13	17.5
Multichannel Merchant	**Base LSTM**	262.3	243.7	-7	122.9
with actual marketing appeals	**Extended LSTM**	262.3	**248.4**	**-5**	**100.3**
	Pareto/NBD	262.3	302.6	+15	134.2
	Pareto/GGG	262.3	324.0	+24	141.5
	GPPM	262.3	309.0	+18	132.2
Blood Donations	**Base LSTM**	81.2	**79.9**	**-2**	**32.8**
	Pareto/NBD	81.2	91.2	+12	35.3
	Pareto/GGG	81.2	92.8	+14	36.0
	GPPM	81.2	93.1	+15	36.0
Sunscreen	**Base LSTM**	77.2	**75.5**	**-2**	**34.2**
	Pareto/NBD	77.2	96.7	+25	40.0
	Pareto/GGG	77.2	97.1	+26	40.1
	GPPM	77.2	94.4	+22	38.2
CDNOW	**Base LSTM**	31.6	**30.8**	**-2**	**11.3**
	Pareto/NBD	31.6	35.4	+12	12.5
	Pareto/GGG	31.6	35.4	+12	12.5
	GPPM	31.6	37.2	+18	13.3
Groceries	**Base LSTM**	38.7	**37.1**	-4	**13.8**
	Pareto/NBD	38.7	38.5	**+0**	15.1
	Pareto/GGG	38.7	39.8	+3	14.9
	GPPM	38.7	40.9	+6	18.0
Yogurt Purchases	**Base LSTM**	31.9	**31.1**	**-2**	**10.9**
	Pareto/NBD	31.9	30.5	-4	12.0
	Pareto/GGG	31.9	31.0	-3	12.1
	GPPM	31.9	15.8	-51	27.1

4.2.2. "When Will The Next Transaction Occur?"

Our proposed self-supervised approach is also flexible enough to detect many other dynamic transaction patterns observed during model calibration, which can be leveraged to improve predictions. As demonstrated previously by Korkmaz, Kuik, and Fok (2013) or Holtrop and Wieringa (2020) using BTYD models, one such feature is the timing of the next purchase: instead of looking into a distant future, managers are eager to know when their customers are expected to make their next transaction (if any), in order to consider possible reactivation initiatives in case a customer is "overdue". Our model can inform managers about this by combining multiple forecast simulations of each customer's future. To examine how well the model performs in correctly detecting the next purchase timing, we calculate the mean Next Inter-Transaction Time $\overline{NITT}$ (in weeks following the end of the respective calibration period) across all customers as an aggregate metric and compare it to the actuals. To measure the individual level performance, we calculate the RMSE for the estimated (or simulated) individual-level NITTs.[25]

The results in Table 8 clearly show the Base LSTM model's exceptional performance in this regard: its predictions deviate from the true $\overline{NITT}$ by just 2.8% on average across the eight empirical scenarios (with an average bias of 11.8%, 13.2% and 19.2% for the Pareto/NBD, Pareto/GGG and GPPM, respectively), and it can also attribute the correct NITTs to individuals quite accurately, as is expressed by the consistently lower NITT RMSE values. We also note that additional covariates again

[25] For LSTM models, we measure the NITT by averaging over the simulated first transactions, for the other benchmarks, we track the cumulative transaction count for each individual customer in the holdout period, marking the first transaction in the period where the count reaches a threshold of one.

help the Extended LSTM model to refine the forecasts further, improving on 7 out of 9 measured scores in the three scenarios with an Extended LSTM model, compared to the Base LSTM model.

4.3. Calibration length sensitivity

The above-demonstrated ability to capture seasonality and other dynamics automatically from the raw transaction histories sets the Base LSTM model clearly apart from the probability benchmark models as well as the GPPM. To get an even better sense of the model's flexibility and to study what is required to arrive at such forecasts, we next conduct a sensitivity analysis using calibration periods of varying length.

Marketing managers oftentimes need to assess a cohort soon after it has been acquired, i.e., when still relatively little historic data is recorded. With a limited amount of information available for model training, this might pose a challenge for a data-driven approach like ours, which does not benefit from prior knowledge like the BTYD models with their built-in parametric assumptions. The Charity Contributions data is already a challenging scenario due to the low levels of activity, and in Fig. 5 we show how the Base LSTM model forecasts changes as we decrease the length of the calibration period, first to 3 years (top plot), 2 years, 1 year, and finally to just 6 months (bottom plot). As an alternative look at each of the data scenarios, we also plot the cumulative transactions as a function of time in the right-hand plots. The holdout prediction length is increased accordingly to take advantage of the entire data available for comparison.

The results summary we present in Table 9 show that irrespective of the calibration length, the Base LSTM model estimates transaction counts for each individual accurately in terms of the RMSE, while keeping the aggregate bias very low. Only when the calibration spans a single year in the third plot in Fig. 5 does the Base LSTM lose some of its power to model

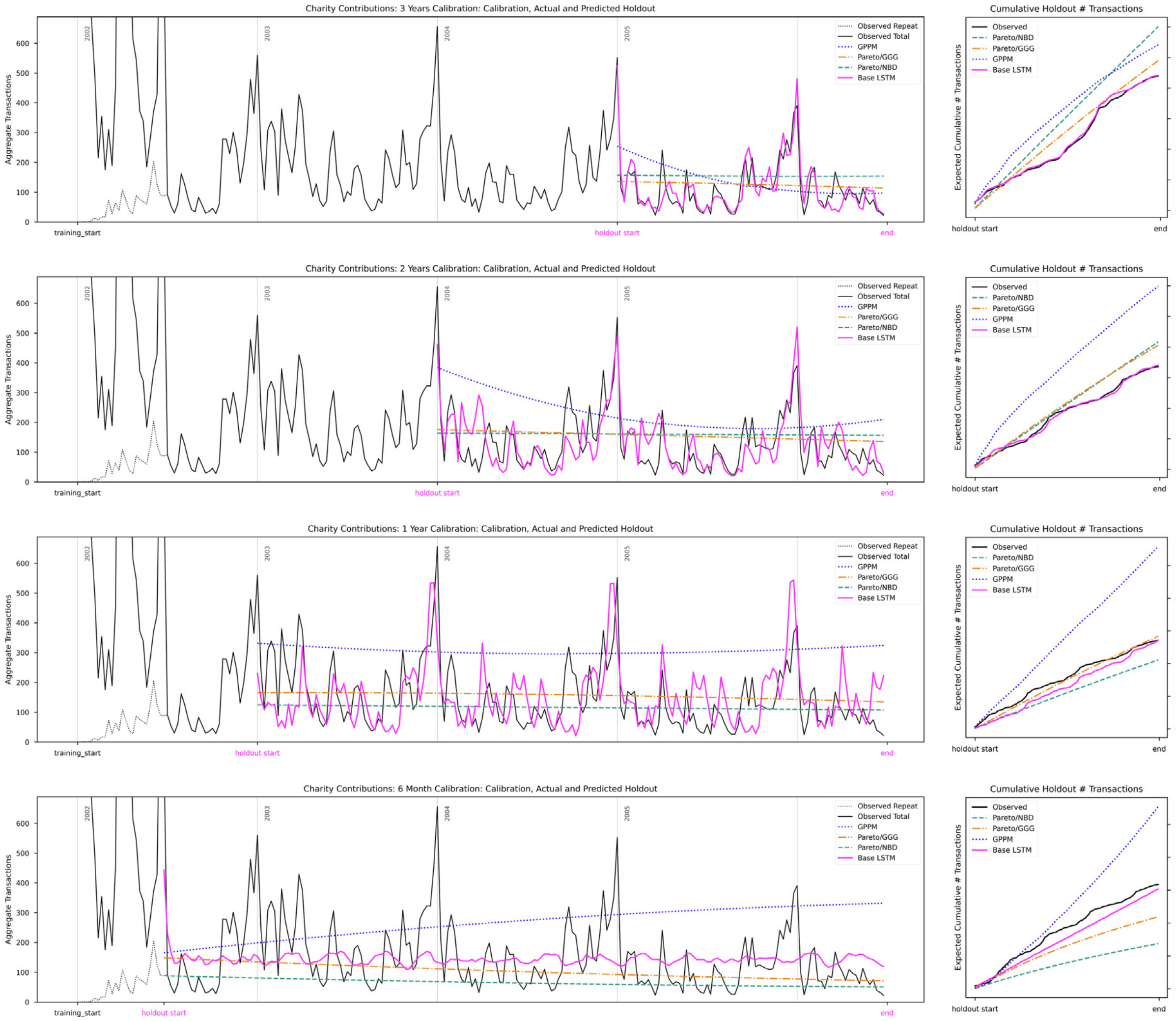


Fig. 5. Charity Contributions: calibration length sensitivity.

Table 9
Charity Contributions: varying the calibration and holdout period length.

Calibration	Holdout	Model	RMSE	bias (%)	MAPE (%)
3 years	1.5 years	**Base LSTM**	**0.84**	**0.6**	**28.2**
		Pareto/NBD	0.88	37.0	71.9
		Pareto/GGG	0.85	11.6	57.9
		GPPM	0.87	23.3	62.1
2 years	2.5 years	**Base LSTM**	**1.44**	**1.2**	**43.4**
		Pareto/NBD	1.48	24.1	65.3
		Pareto/GGG	1.45	20.8	62.8
		GPPM	1.55	78.3	98.1
1 year	3.5 years	**Base LSTM**	**2.21**	**0.1**	58.2
		Pareto/NBD	2.23	-22.0	**54.2**
		Pareto/GGG	2.22	4.8	57.5
		GPPM	2.60	106.4	115.1
6 months	4 years	**Base LSTM**	**2.68**	**-4.2**	58.0
		Pareto/NBD	2.81	-56.8	64.0
		Pareto/GGG	2.69	-31.1	**56.6**
		GPPM	2.98	75.0	102.3

the seasonal pattern. Indeed, in this case only a portion of the customers included in the calibration sample has transaction histories that span the entire year (this is a cohort acquired during the first half of that first year), and consequently the model is not able to observe post-New Year behavior of many individuals who have not become active until later during the first six months, and so the reoccurring yearly activity spikes after the New Year are not captured very well. When the calibration period spans only 6 months (bottom plot in Fig. 5) the aggregate LSTM prediction becomes similar to those of the BTYD models. Here, the overall excellent performance of the Base LSTM model is perhaps most surprising, as there is little evidence of customers becoming inactive in the initial 6 months. In fact, the opposite is true – more individuals are becoming active as the cohort is still growing, and this is where the prior knowledge available to the probability models should help. Indeed, in the six-month calibration scenario the Base LSTM model over-forecasts the inactive group more than twice as much as compared to the Pareto/NBD, but is able to compensate for this error by forecasting the remaining customer groups more accurately, and with much lower bias.

4.4. Customer groups and segments

Companies are usually particularly keen on correctly forecasting the future most active customer groups and those who become inactive (and thus might be subject to reactivation campaigns). To further investigate how well our proposed model does beyond the previously discussed "opportunity" and "at risk" customer groups and to get a finer granular insight into this, we classify the customer base into several subgroups with decreasing holdout period activity. This way we arrive at the Top 1, 5, 25 and 50 percent customer groups. Likewise, we do the same for the less active Low 50 percent group as well as the inactive customers with zero holdout transactions. We showcase the Charity Contributions dataset again, which is characterized by a high degree of dataset-level inter-event timing regularity. Fig. 6 reflects this observation for the various subgroups. The most active group (Top 1%) follows a highly regular monthly donation pattern and the Base LSTM model captures this behavior to a large extent, which is reflected by a 10% RMSE improvement, 11 percentage points of aggregate bias reduction, and a MAPE improved by 24 percentage points, compared to the Pareto/NBD model. These loyal customers deserve special attention and their characteristics can serve as a prime example of the type of audience marketers like to target when crafting an acquisition strategy.

Within the groups of less active individuals, yearly seasonality with emphasis on the holiday period dominates the group behavior. The bottom left chart in Fig. 6 shows that the LSTM model correctly estimates a large portion of the spike(s) coinciding with the holidays for the "occasional" donors (Low 50%) – individuals often at the searchlights of marketers, as they have the potential to become more valuable with the right intervention. In the bottom right plot we see the transactions incorrectly attributed to the large (81.3%) group of inactive donors, and the LSTM reduces this share by 20 pp compared to the Pareto/NBD. In the Appendix Section F, Table 13 we provide details on these segment-level observations for all analyzed datasets.

4.5. The influence of covariates

When individual characteristics or other contextual descriptive covariates are known or made available, we can account for these by fitting an Extended LSTM model. This usually improves (i.e., lowers) the best cross-entropy loss (see Equation 3)

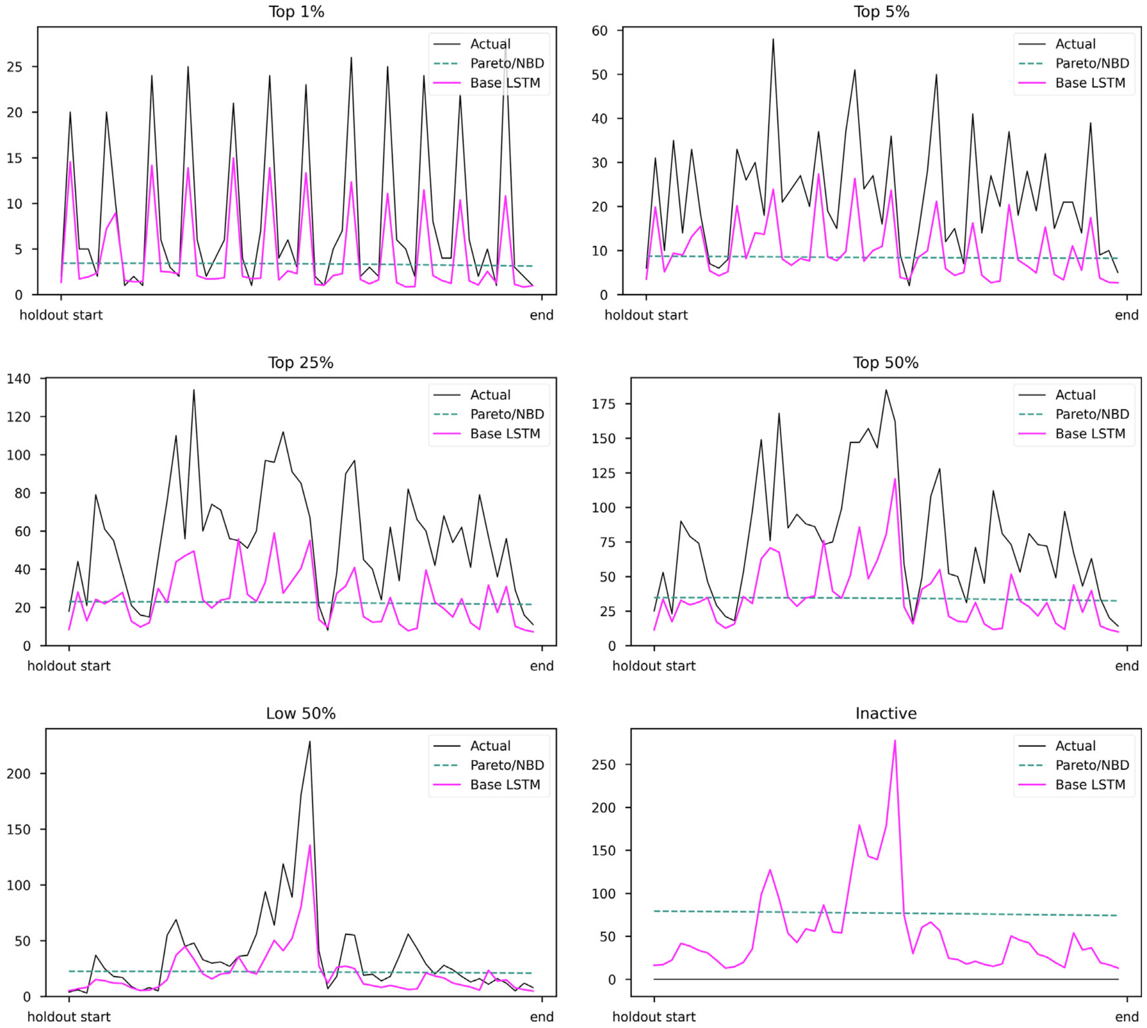


Fig. 6. Segment-level prediction details: Charity Contributions.

achieved during model training, which, in turn, translates into improved out of sample predictive performance. We provide more details about this process in the Appendix Section C.

Take as an example for such covariates the household customers of the Electronics Retailer. They are characterized by some basic demographic background information which we can use as additional inputs for our Extended LSTM model during model training: The age and gender of the ‘‘head of household", an indicator of the presence of children, and the household income group.[26]

Such customer covariates can help the Extended LSTM improve the predictive accuracy and further refine the forecast with respect to the various segments of the customer base. In Fig. 7, we show this effect for the above-mentioned Electronics Retailer customer groups, and plot the evolving cumulative holdout period transactions separately for households with and without children, broken down by age group or the gender of the ‘‘head of household", or by household income group. The full shaded areas depict the actual transaction counts, the magenta lines show predicted values. In the left column is the baseline prediction of the Pareto/NBD model which does not account for the additional covariates. Neither does the forecast derived by the Base LSTM model in the middle column, but it is already very accurate in its ability to identify heterogeneous customer groups. The most important signals are already contained in the basic transaction log. Making the customer covariates available to the Extended LSTM model in the right column brings further polish to the model's ability to accurately attribute evolving transaction counts to the right customers, which enables marketing planners to fine-tune their marketing

[26] Instead of selecting or otherwise preparing the covariates, we simply provide them all to the Extended LSTM model as variable inputs and the model is left to decide which of them are useful to consider, and which are not. This is another notion of the process we earlier denoted as the model's automatic feature extraction capability. Note that the variables can be categorical or continuous; in the latter case we would re-scale the values into a range between 0 and 1 to prevent large values entering parameter estimation.

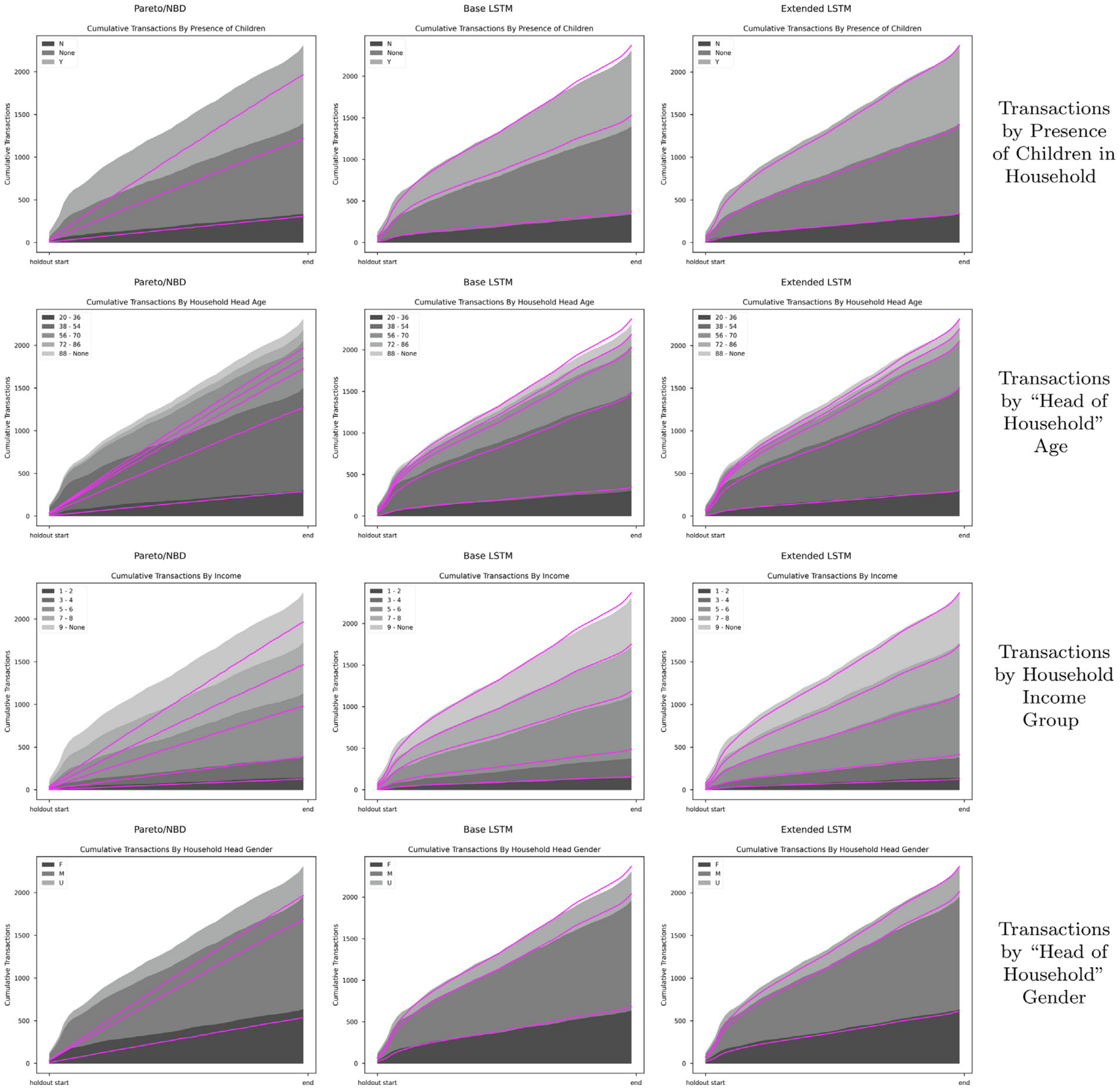


Fig. 7. Cumulative transactions with Electronics Retailer: Pareto/NBD, Base LSTM, Extended LSTM model.

Table 10
Performance lift of Extended LSTM models using customer covariates in prominent segments, reported in percentages (%).

		Percentile by Holdout Transaction Activity								
Dataset		Top 5%	Top 10%	Top 25%	Top 50%	Low 50%	Low 25%	Inactive	Active	**Overall**
Charity Contributions **Extended LSTM** *with actual marketing appeals*	% of total	0.9%	1.9%	4.7%	9.3%	9.3%	4.7%	81.3%	18.7%	100%
	RMSE	+7	+7	+8	+8	+9	+9	-1	+8	**+6**
	bias	+4	+5	+5	+6	+7	+6	+3	+6	**+2**
	MAPE	+4	+5	+5	+5	+6	+3	+1	+5	**+2**
Electronics Retailer **Extended LSTM** *with time-invariant covariates*	% of total	1.4%	2.8%	7.0%	14.0%	14.0%	7.0%	71.9%	28.1%	100%
	RMSE	+2	+1	+1	+1	-1	+5	-2	+1	**+1**
	bias	+3	+2	+1	+1	+2	+3	+4	+1	**+3**
	MAPE	+3	+2	+1	+1	+1	+1	+3	+1	**+1**
Multichannel Merchant **Extended LSTM** *with actual marketing appeals*	% of total	1.5%	3.0%	7.7%	15.4%	15.4%	7.7%	69.3%	30.7%	100%
	RMSE	+7	+10	+13	+14	-53	-65	+13	+12	**+12**
	bias	+8	+12	+20	+27	-1	-20	+26	+33	**+5**
	MAPE	+2	+5	+13	+18	-3	-14	+12	+19	**+4**

programs. The performance lift of the Extended LSTM model compared to the Base LSTM broken down by transaction activity group for the two examined scenarios where customer covariates are available to us is reported in Table 10. In Appendix Section E we demonstrate further how marketing managers can benefit from "what-if" scenario simulations for user-defined holdout covariate schedules, such as customized direct marketing campaigns.

5. Discussion

We demonstrate how firms operating in non-contractual business settings can benefit from the automatic feature extraction capabilities of deep learning models for predictive customer base analysis. Our proposed model informs managers on both short- and long-term forecasts of individual customer behavior and helps to timely uncover business opportunities as well as potential customer defection. As we have shown, it also accurately predicts periods of elevated transaction activity and captures other forms of purchase dynamics that can be leveraged in simulations of future sequences of customer transactions. We highlight our model's flexibility and performance on two groups of valuable customers: those who keep making more and more transactions with the firm (denoted as ”opportunity” customers) and those who are at risk of defection. We demonstrate that the model also excels at automatically capturing seasonal trends in customer activity, such as the shopping period leading up to the December holidays. In Appendix Section F we provide a further characterization of scenarios where our model performs particularly well and where it does not do so relative to the used benchmark methods.

The model brings many practical benefits for the marketing analyst, such as the lack of need for manual encoding of any features in the customer data, a simple optimization objective, and quick estimation on modern computer hardware. We show that incorporating contextual information in the model is straightforward and brings an additional boost in predictive accuracy. However, the model performance is already extremely strong when no context is available beyond the timing of the customer's transactions. This is welcome news for firms that do not wish to collect personal information on principle, to avoid the questionable ethics of harvesting the "behavioral surplus" (Zuboff, 2019): our work shows that this is feasible without a big loss of accuracy. We gather evidence from eight diverse real-life settings to demonstrate the model robustness as a flexible, general purpose prediction tool for customer base analysis.

The proposed approach is agnostic about time-varying or time-invariant covariates: Instead of adapting the data to a model, our model adapts to the data and can simply be left to leverage useful signals automatically without the need to change the model architecture or training procedure. While the incorporation of covariates is in principle possible with so-called "scoring" or regression-like models and, to a certain extent, with advanced probability models as well, our approach comes with another advantage. Regression-type models and traditional ML methods are often criticized for their backward-looking properties and inefficient use of the available data (because they need to hold out the most recent period of transaction histories to construct the dependent variable; cf. Fader & Hardie (2009)). This limitation implies the inability to make projections into the distant future, but despite the "one time step ahead" property of its predictions we show, by means of a calibration length sensitivity study, that the proposed approach can leverage the complete transaction histories and deliver excellent long-term forecasts for individual customers. Such a perspective seems to be particularly useful for the rich stream of information accompanying customer-firm interactions in modern digital business environments (Wedel & Kannan, 2016; Dzyabura & Peres, 2021) where anything including high-dimensional data can become available as a covariate.

The challenge for deep learning models of customer behavior remains their opaque nature and the lack of simple ways to interpret their behavior, which is especially true for the complex temporal dynamics of RNNs. Other frequently contended disadvantages are disappearing: Computational power is more affordable and efficient training methods are advancing at a fast pace, which also facilitates the adaptive fine-tuning of model parameters once ”new" transaction data accrues, and datasets of historical customer transaction records are more commonly available, larger, and more detailed with observed behavior across diverse contexts and platforms. Furthermore, the skills required to build such models are becoming widespread, thanks to the mature open source programming tools and burgeoning research community. Deep neural networks continue to inspire creative new applications, engineering and theoretical advancements, and with more marketing practitioners interested, this trend will continue in the future.

Appendix A. Long Short-Term Memory (LSTM)

In this Appendix, we provide a short explanation of the idea of the Long Short-Term Memory (LSTM) and how it can be utilized for customer base analysis. The specific properties of LSTM cells, which we leverage for forecasting customer activities, have been introduced by Hochreiter and Schmidhuberter (1997) and then further complemented by Gers, Schmidhuber, and Cummins (2000) to address a limitation frequently encountered by RNNs. The activation functions of RNN feedback connections are designed to capture inter-temporal relationships, but they merely establish "short-term" memory. Once the lag between significant input events becomes too big, the influence of earlier signals can get lost to noise and RNNs become "myopic". This problem of *vanishing gradients* (Gers et al., 2000) makes it difficult for RNNs to learn long-term dependencies and aggravates with longer training sequences, because of the extended chaining of mathematical operations used to calculate the error gradient. This instability makes "vanilla" RNNs inappropriate in the context of customer bases analysis, where the task is to train and predict long sequences of transaction histories. These limitations of RNNs

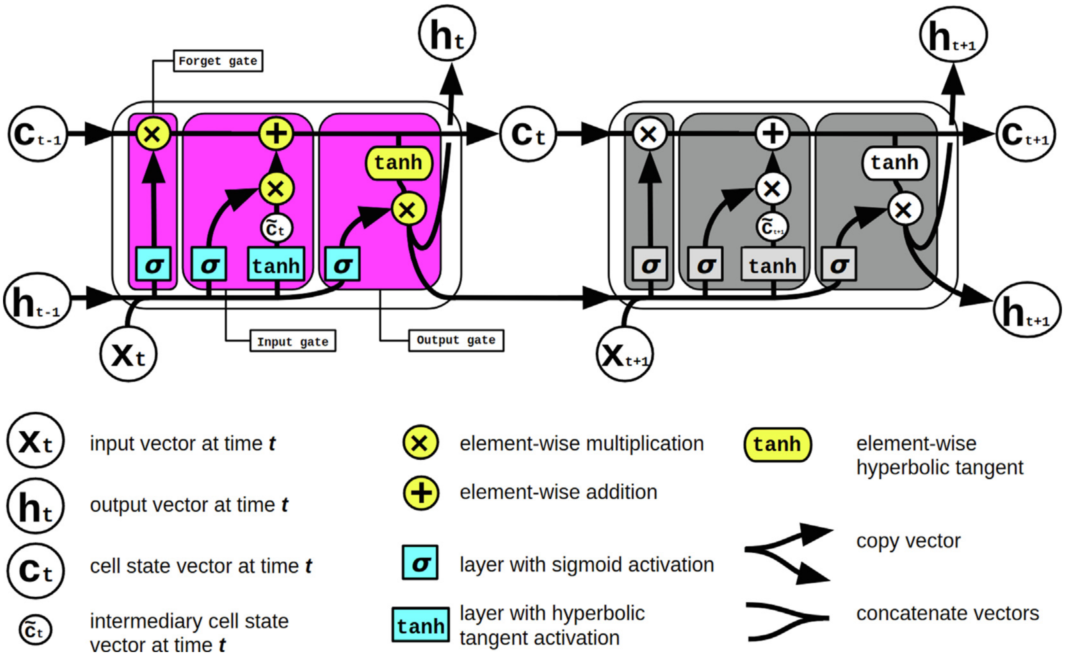

Forget gate
ht
ht+1
Ct-1
Ct
Ct+1
ht-1
Xt
Xt+1
tanh
σ
C̃t
C̃t+1
Input gate
Output gate
Xt input vector at time t
ht output vector at time t
Ct cell state vector at time t
C̃t intermediary cell state vector at time t
element-wise multiplication
element-wise addition
σ layer with sigmoid activation
tanh layer with hyperbolic tangent activation
tanh element-wise hyperbolic tangent
copy vector
concatenate vectors

Fig. 8. The LSTM module.

can be overcome by the LSTM cell. The LSTM is characterized by a set of built-in trainable internal neural networks called "gates" that can selectively update an internal *cell state* with parts of the input signal, while automatically discarding uninformative signals. These cells are usually organized in layers which effectively serve as the network's long-term memory. A single LSTM cell consists of four connected "gates".

In the context of customer base analysis, the LSTM module serves to automatically compress the previously observed history of a customer's actions into its internal cell state. The configuration of cell states is the functional equivalent of the latent behavioral characteristics captured by a probability model. While the latter makes inferences about these latent characteristics given an individual's summary statistics of observed past behavior (i.e., RFM metrics), LSTM-based networks find a suitable representation of observed history through a stochastic optimization process, learning how to continuously update the cell states and retain useful information in order to maximize the model's predictive performance measured by an entropy loss function.[27] This cell state update makes the LSTM conceptually similar to hidden Markov models (Baum & Petrie, 1966), but with higher-order Markov properties.

Fig. 8 illustrates the processing of two consecutive input vectors x_t and x_{t+1}. In our context these vectors represent the individual customer's behavior during two consecutive discrete time periods. At each step t two vectors from the previous time step, the cell state c_{t-1} and the previous output h_{t-1}, enter to update the cell state and produce a new output. First, the so-called *forget* gate determines which part to remove from the previous cell state vector (c_{t-1}), depending on the previous output (h_{t-1}) and the current input (x_t):

$$forget_t = \sigma(W_{forget}[h_{t-1}, x_t] + b_{forget}), \tag{6}$$

where W_{forget} are the forget layer weights and b_{forget} the corresponding unit biases; $\sigma(\ldots)$ denotes a nonlinear sigmoid or squashing function which transforms the argument into a value between 0 and 1. Next, the *input* gate determines what information will be used to update the cell state with its own set of weights (W_{input}) and biases (b_{input}). Then, an intermediary cell state $\widetilde{cell}_t$ is derived by applying a hyperbolic tangent function. The hyperbolic tangent is another smooth non-linear squashing function with output values between −1 and +1. Finally, the actual update of the cell state is achieved by combining the previous cell state with the filter of the forget layer and adding the intermediary cell state scaled by the input layer:

$$input_t = \sigma(W_{input}[h_{t-1}, x_t] + b_{input}), \tag{7.1}$$

$$\widetilde{cell}_t = tanh(W_{cell}[h_{t-1}, x_t] + b_{cell}), \tag{7.2}$$

$$cell_t = forget_t \, * \, cell_{t-1} + input_t \, * \, \widetilde{cell}_t. \tag{7.3}$$

[27] Note that other loss functions can be useful, in particular when the outputs of the network are not categorical.

The *output* gate is responsible for determining how much of the current input and the just updated cell state will be allowed to flow to the current output (h_t) and thus be conserved for future updating cycles. This is achieved by combining the analog operation on the input vector ($output_t$) as in the previous gates with the hyperbolic tangent transformation of the newly derived cell state $cell_t$:

$$output_t = \sigma(W_{output}[h_{t-1}, x_t] + b_{output}) \tag{8.1}$$

$$h_t = output_t \ * tanh(cell_t) \tag{8.2}$$

Appendix B. Technical Implementation Notes

Here we provide more details about the technical aspects of the model and share our experience with training. We implement the model in Python using the open source neural network tools Keras (Chollet, 2015) and Tensorflow (Abadi et al., 2015) and we use a standard desktop computer with an NVidia GeForce 2080 consumer graphics card for training. This hardware setup allows for multiple (typically 4–8, depending on model and data size) LSTM models to train in parallel, so lesser hardware is fine to use. We tried replacing the LSTM memory layer with different and more advanced RNN architectures, but the "vanilla" LSTM speeds up training by a factor of 10 due to its efficient GPU-acceleration, which is an advantage that outweights any other potential incremental benefits. If less granular forecasts are sufficient, an additional training speed-up can be achieved by aggregating the input samples into monthly rather than weekly buckets. As illustrated for the Charity Contributions case (see the second entry in Table 4) this does not result in a significant loss of accuracy. We observe the same findings for different sub-groups of customers and variations in holdout periods.

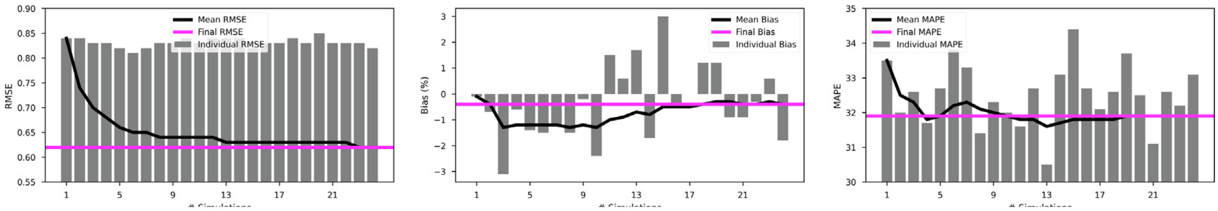
B.1. Network topology and hyperparameters

To find a best-performing neural network topology and set hyperparameters for model training, we implement a random walk (RW) algorithm (see, e.g., Matuszyk, Castillo, Kottke, & Spiliopoulou (2016)), beginning with a small model i.e., one LSTM layer stacked on top of a fully-connected layer, both layers with 64 units each.[28] We then let the RW explore similar configurations progressively towards larger, deeper architectures, varying each new configuration by randomly changing the number and size of LSTM layers and fully-connected layers, adjusting the learning rate and batch size, or choosing from a set of different optimization and regularization methods. This simple RW quickly discovers model architectures that are fast to train (i.e. 10 min for small data sets like Groceries) and produce accurate results, however the different models can exhibit different biases, which is why we recommend forming an ensemble using a set of best performing models with varying hyperparameter profiles, as well as generating multiple independent predictions with each individual model, to further reduce noise.

B.2. Network training

When training neural networks, the norm is to monitor the validation loss – a measure of the error the model makes on an unseen part of the data – to asses the progress of model training. The idea is that good performance on unseen data is evidence of the model's "general" ability to perform a given task, and by ending the training process at the point when the validation loss stops improving, we prevent the model from *overfitting* the training data. There is a downside though: when we take out a (randomly sampled) portion of the calibration data to form this validation set, we end up with less data to learn from overall (typically around 10% of the data is used for validation). This trade-off makes sense in scenarios where we do not know which data will be used as model input in the future. This is not our case though: we make predictions for a specific cohort of customers. This means our training procedure is as follows: we train the model as is common using a validation set first, and once the validation loss stops improving for a number of epochs, we restore the model state to the point with lowest *validation loss* and perform several "fine-tuning" training epochs using the entire calibration data set including the samples previously left out as validation, using a large batch size and a reduced learning rate. The idea is to fine-tune the model to the specific cohort, even if this technically means a small degree of "overfitting". Note that at no point is any of the holdout period data used during the fine-tuning stage. This way, we also assure a like-for-like comparison of the LSTM with probability models, which also use the entire calibration set for model estimation but do not require a separate validation set.

[28] Stacking multiple LSTM layers may be necessary in order to achieve best predictive performance – we study this empirically in the Charity Contributions scenario, recording the performance of various models with one, two, and three stacked LSTM layers, in a range of hyperparameter settings generated by the RW (n = 300). We find that the second LSTM layer brings an average improvement of 2% in terms of the individual level RMSE: 0.854 ± 0.024 for the one-layer networks, 0.840 ± 0.015 for networks with two LSTM layers. Adding a third stacked LSTM layer resulted in an RMSE of 0.845 ± 0.022, which is better than a single-layer LSTM, but worse than the two layer average. A similar effect is observed when measuring the aggregate bias: On average, 8.7%±7.2% for the single-layer network, 4.4%±2.7% for the two-layer setup (a 4.3 pp improvement), and again adding the third stacked LSTM layer does not lead to further improvement (7.0 ± 5.0%, which is still better than the single layer result). This illustrates that there is indeed an optimal network capacity, depending on the amount and complexity of training data – with a larger set of training examples, the optimal capacity would grow larger.



(a) RMSE (b) Bias (c) MAPE

Fig. 9. Evolution of error measures across increasing number of simulations (Charity Contributions dataset).

B.3. Holdout simulation and prediction robustness

As described in Section 2.1, in prediction mode we generate a set of independent simulations for each customer and take the mean expected number of transactions in a given time step as our final result. Fig. 9 shows the prediction error measures (y-axis) evolving with increasing number of averaged simulations (x-axis). The grey bars depict the respective errors of each single simulation, while the average error values across evolving simulation runs are given in black. The magenta line shows the final result after averaging across all simulations. All error metrics converge to a terminal value after a relatively low number of simulations.[29] An additional accuracy improvement is obtained by using an ensemble of multiple LSTM models trained with different model hyperparameters and taking the mean result across predictions generated by the individual models.

Appendix C. The Extended LSTM model

In this section of the Appendix, we comment further on the topic of extending the LSTM with additional customer covariates. Accounting for clearly exogenous covariates, such as customer background characteristics or scheduled marketing campaigns (e.g., quarterly catalogue mailings, periodical reminders, etc.) is straightforward. The situation can become more difficult with marketing interventions that might be subject to some individual-level targeting rules. However, as we discuss in Section 2.3, for a modelling setup like ours where prediction accuracy is the primary goal, correcting for potential endogeneity does not seem to be critical. Furthermore, it is common business practice in larger firms that financial planning and analysis (FP&A) divisions elaborate future marketing activity and spend on forecasts which can be used by the model to condition upon. This is what we did in the Charity Contributions setting, where individual direct mailing intervention records are available, and by conditioning the model on the actual holdout interventions we observe our most accurate forecast in terms of individual-level RMSE – see the Charity Contributions *with actual marketing appeals* entry in Table 4.

The relationship between forecasting performance and numerical optimization objectives can be illustrated by plotting the best validation loss value achievable during training against the individual-level RMSE score in the left-hand plot of Fig. 10, where each dot represents a model created with varying neural network topology, optimization strategy and other hyperparameter settings. The Base LSTM models, trained without any covariates, are plotted in black color and are clustered in the center. The Extended LSTM models, which leverage the individual-level direct marketing interventions, are depicted with red dots, showing that this additional information tends to help the models achieve better (lower) validation loss value, which in turn tends to improve individual out-of-sample prediction accuracy as indicated by the RMSE scores.[30] The validation loss itself is subject to the influence of all the various model training hyperparameters, and to highlight the most important of these factors we use a Boosted Random Forest (Friedman, 2001) to regress model hyperparameters to the final validation loss value in the right hand plot in Fig. 10.

As an alternative look at the Extended LSTM model, the two left-hand column plots in Fig. 11 show the average actual holdout-period transactions along with the conditional expected counts predicted by the Base and the Extended LSTM, broken down by the calibration-period number of repeat transactions. We note that in both reported scenarios, Charity Contributions and Electronics Retailer, the prediction of the Base LSTM model already follows the actual transaction counts closely. To allow for a better inspection of differences, the right-hand plots display the MAE between the estimated and actual conditional counts from the left-hand charts. There are a couple of subtle differences between the predictions derived by the Base LSTM and the Extended LSTM models: In both scenarios we note an improvement in the most frequently purchasing customer group, in the Electronics Retailer case the Extended LSTM leverages household demographic information and is able to largely correct for the over-forecasting bias of the Base LSTM model in the group of zero-repeaters. Both of these customer segments are important to managers: The former because it contains some of the firm's most valuable customers, the latter because it is the largest group.

[29] We find in our empirical study that using around 30 independent simulations renders a good balance between computational cost and the incremental improvement in accuracy, but one can simulate as many scenarios as needed.

[30] In the Charity Contributions scenario depicted in Fig. 10, the validation loss and RMSE are positively correlated with a Pearson's coefficient of 0.56.

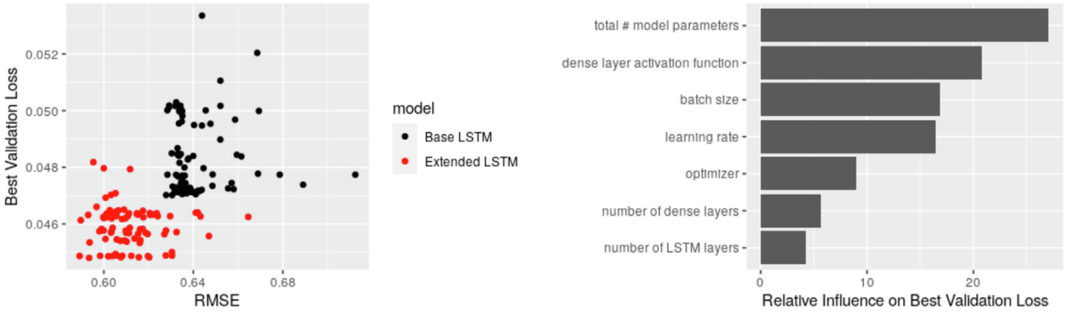


Fig. 10. Charity Contributions: influence of covariates on best validation loss, relative importance of model hyperparameters on final validation loss.

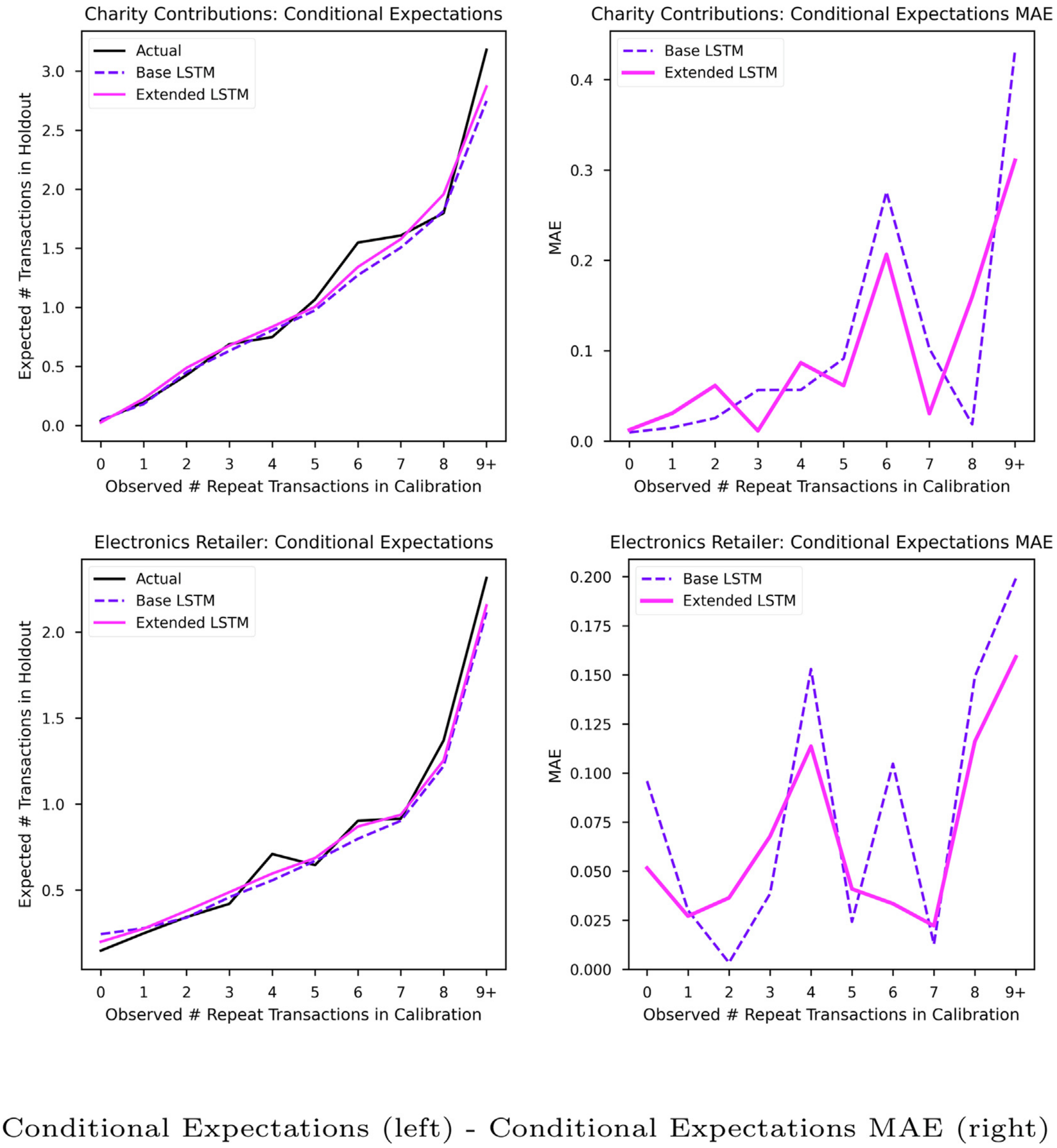


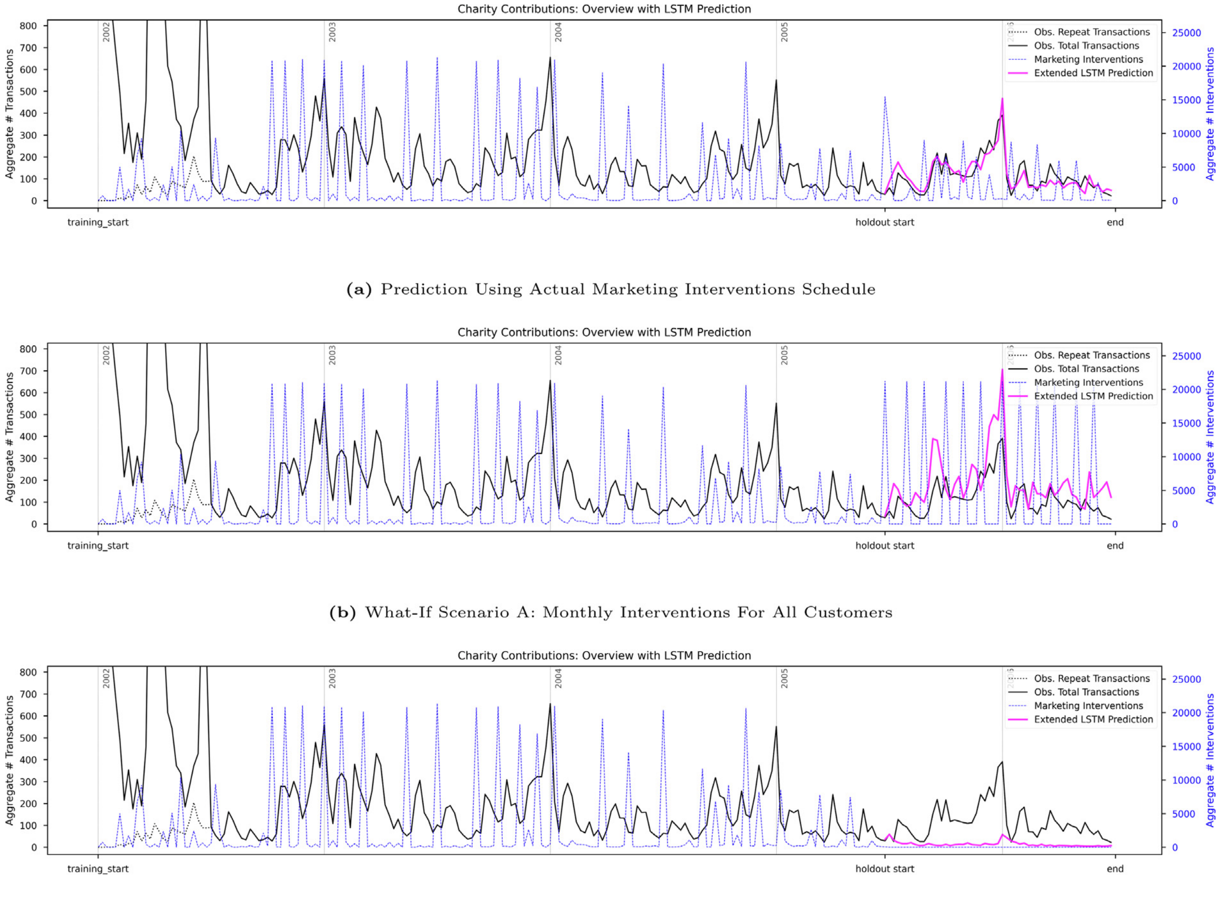
Fig. 11. Conditional expectations for Base LSTM model and Extended LSTM model with covariates.

Appendix D. Simulation Study Of Pareto/NBD-Generated Data Scenarios

Deep neural networks are famous for their ability to discover even very subtle patterns in noisy data. This is, we believe, one of the reasons for the excellent performance of our model since the real-life scenarios we examine are presumably full of such subtle "hints", which however largely remain hidden from the other benchmark models. To demonstrate how our proposed approach performs in the complementary scenario when there is an underlying source model that generates the trans-

Table 11
Base LSTM on synthetic Pareto/NBD transactions: dataset characteristics and prediction accuracy.

				Calibration		Holdout			Accuracy		
Dataset	Clumpy	r_{WM}	Seasonality	Mean events	Non-repeaters	Mean events	Inactive	Model	RMSE	bias	MAPE
High Frequency High Churn	52%	1.2	0.4	5.3	59%	1.1	92%	Base LSTM	**2.35**	**-1.8**	**9.0**
								Pareto/NBD	2.59	4.0	9.7
High Frequency Low Churn	34%	1.1	0.2	18.5	21%	6.6	55%	Base LSTM	**4.95**	-2.1	4.8
								Pareto/NBD	4.99	**-0.3**	**4.0**
Low Frequency High Churn	16%	1.1	0.4	1.5	85%	0.1	96%	Base LSTM	0.58	**-0.1**	**23.0**
								Pareto/NBD	**0.55**	-1.9	24.8
Low Frequency Low Churn	8%	1.0	0.2	2.8	52%	0.7	74%	Base LSTM	1.16	**0.3**	13.4
								Pareto/NBD	**1.14**	1.0	**11.9**



(a) Prediction Using Actual Marketing Interventions Schedule

(b) What-If Scenario A: Monthly Interventions For All Customers

(c) What-If Scenario B: Zero Interventions

Fig. 12. Charity Contributions: what-if scenarios with custom marketing intervention schedules.

Table 12
Relative predictive performance lift between LSTM and benchmark models.

			Relative Lift			
Dataset	Model	Benchmark	RMSE	bias	MAPE	Descriptive Tags
Charity Contributions	Base LSTM	Pareto/NBD	+5	+13	+28	regular large high churn long seasonal
		Pareto/GGG	+2	+5	+20	
		GPPM	+10	+30	+18	
		Pareto/NBD	+11	+15	+30	
with actual marketing appeals	Extended LSTM	Pareto/GGG	+8	+7	+22	
		GPPM	+15	+32	+20	
Electronics Retailer	Base LSTM	Pareto/NBD	+7	+12	+15	irregular small high churn v.long
		Pareto/GGG	+6	+1	+20	
		GPPM	+7	+31	+35	
		Pareto/NBD	+7	+15	+17	
with time-invariant covariates	Extended LSTM	Pareto/GGG	+6	+4	+21	
		GPPM	+8	+34	+36	
Multichannel Merchant	Base LSTM	Pareto/NBD	+3	+11	+51	v. low freq random small high churn
		Pareto/GGG	+2	+42	+36	
		GPPM	0	+4	+6	
		Pareto/NBD	+13	+16	+55	
with actual marketing appeals	Extended LSTM	Pareto/GGG	+13	+47	+39	
		GPPM	+10	+8	+9	
Blood Donations	Base LSTM	Pareto/NBD	+3	+24	+13	regular low freq not clumpy
		Pareto/GGG	+1	+9	0	
		GPPM	+2	+27	+16	
CDNOW	Base LSTM	Pareto/NBD	+6	+18	+6	random large high churn v.short
		Pareto/GGG	+6	+19	+6	
		GPPM	+12	+36	+23	
Groceries	Base LSTM	Pareto/NBD	+11	+19	+9	regular v.small short
		Pareto/GGG	+6	+11	+3	
		GPPM	+17	+10	+5	
Yogurt Purchases	Base LSTM	Pareto/NBD	+11	+5	+5	v.large low churn long high freq clumpy
		Pareto/GGG	+11	+3	+4	
		GPPM	+20	+4	+2	
Sunscreen	Base LSTM	Pareto/NBD	+1	+19	+47	low freq not clumpy very seasonal
		Pareto/GGG	+1	+8	+50	
		GPPM	+6	+50	+91	

Note: RMSE lift is reported in percent (%), bias and MAPE in percentage points (pp).

action data, we conducted an experimental simulation study involving a two-by-two comparison of the Base LSTM model with the Pareto/NBD as baseline in transaction settings generated according to Pareto/NBD assumptions.[31] The four settings differ by high/low values of frequency and churn, each containing 5,000 customers with two years (104 weeks) of calibration, and a holdout forecasting period of one year (52 weeks). We present the remaining descriptive statistics as well as the results in Table 11, showing that even in this simulated setting the Base LSTM model is a strong competitor for the Pareto/NBD in all three metrics.

Appendix E. "What-If Charity Contributions: Conditioning the Extended LSTM Model Prediction With Custom Marketing Schedules

In this Appendix section, we explore how firms can use the Extended LSTM model to generate alternative "what-if" prediction scenarios. To this end, we estimate an Extended LSTM model by providing extra contextual information as additional model input(s) during training. In the case of the Charity Contributions, we use records of direct marketing appeals sent out by the charity as time-varying individual-level covariates. In our main empirical study, we condition the Extended LSTM model with the actual marketing intervention schedule to create our most accurate forecast. However, as we mention in Section 2.3, there is another option available: The manager can determine custom schedules for the contextual information (e.g., a direct marketing campaign) and observe the effect on the predictions for the target variable.

As depicted in Fig. 12 we create three alternative scenarios by manipulating the holdout marketing intervention schedules. In the top figure (a) is the actual scenario as tailored by the charity manager, with a varying aggregate volume of marketing interventions throughout the calibration period that subsides somewhat in the holdout period (dotted blue line). In this case the Extended LSTM model predicts an aggregate total of 6108 transactions over the holdout period, only 0.8% more than the actual total of 6060 transactions. The middle figure (b) What-If Scenario A includes a custom intervention schedule during the holdout period in which every individual is contacted by the firm at the beginning of each month, as visualised by the regular blue dotted line spikes. This leads to an increased transaction activity and a predicted total of 9990 transactions, a +63% increase. Conversely, the bottom figure (c) depicts an extreme scenario in which we stop the direct marketing appeals

[31] We use the `pnbd.GenerateData` function from the `BTYDplus` package (Platzer, 2016) to prepare synthetic data.

Table 13
Relative performance lift for segments by holdout transaction activity – LSTM against Pareto/NBD. Note: RMSE lift is reported in percent (%), bias and MAPE in percentage points (pp).

		Customer Percentile by Holdout Transaction Activity								
Dataset		Top 5%	Top 10%	Top 25%	Top 50%	Low 50%	Low 25%	Inactive	Active	**Overall**
Charity Contributions	% of total	0.9%	1.9%	4.7%	9.3%	9.3%	4.7%	81.3%	18.7%	100%
Base LSTM	RMSE	+7	+6	+4	+3	-6	-7	+17	+2	**+5**
	bias	+7	+5	+3	+2	-2	-1	+20	+1	**+13**
	MAPE	+10	+7	+6	+7	+19	+27	+10	+8	**+28**
Extended LSTM	RMSE	+13	+12	+10	+10	+2	+2	+15	+9	**+11**
with actual marketing appeals	bias	+11	+10	+9	+8	+5	+4	+22	+7	**+15**
	MAPE	+14	+12	+11	+12	+24	+30	+11	+14	**+30**
Calibration Length Sensitivity Study										
3 years calibration only	% of total	1.1%	2.2%	5.6%	11.2%	11.2%	5.6%	77.6%	22.4%	100%
Base LSTM	RMSE	+1	-1	-3	-4	-9	-15	+26	-4	**+5**
	bias	+2	-1	-4	-6	-18	-17	+28	-9	**+36**
	MAPE	+7	+4	+3	+3	29	+34	+14	+5	**+44**
2 years calibration only	% of total	1.6%	3.3%	8.1%	16.3%	16.3%	8.1%	67.4%	32.6%	100%
Base LSTM	RMSE	+3	0	-2	-4	+1	+7	+22	-4	**+2**
	bias	+1	-1	-5	-8	-20	+6	+11	-12	**+23**
	MAPE	+3	+1	-4	-5	+24	+34	+6	-3	**+22**
1 year calibration only	% of total	2.2%	4.4%	11.1%	22.2%	22.3%	11.1%	55.5%	44.5%	100%
Base LSTM	RMSE	+5	+4	+4	+4	+18	-1	-27	+5	**+1**
	bias	+4	+2	+2	+3	+17	-23	-16	+6	**+22**
	MAPE	+5	+3	+3	+3	0	-3	-10	+5	**-4**
6 months calibration only	% of total	2.5%	5%	12.5%	25%	25%	12.5%	49.9%	50.1%	100%
Base LSTM	RMSE	+5	+6	+8	+10	+32	-4	-56	+10	**+4**
	bias	+7	+8	+12	+16	+35	-13	-26	+26	**+53**
	MAPE	+6	+8	+11	+14	-16	-27	-17	+17	**+6**
Electronics Retailer	% of total	1.4%	2.8%	7.0%	14.0%	14.0%	7.0%	71.9%	28.1%	100%
Base LSTM	RMSE	+11	+10	+8	+7	+14	+4	+3	+8	**+7**
	bias	+7	+5	+4	+3	+4	+4	-14	+3	**+12**
	MAPE	+7	+5	+4	+3	+10	+12	-10	+3	**+15**
Extended LSTM	RMSE	+13	+11	+9	+8	+13	+8	+1	+8	**+7**
with time-invariant covariates	bias	+10	+7	+5	+4	+6	+7	-11	+4	**+15**
	MAPE	+10	+7	+5	+4	+11	+13	-7	+4	**+17**
Multichannel Merchant	% of total	1.5%	3%	7.7%	15.4%	15.4%	7.7%	69.3%	30.7%	100%
Base LSTM	RMSE	-8	-7	-7	-6	+65	+75	+31	-3	+3
	bias	-11	-10	-16	-18	-23	+8	+2	-21	+11
	MAPE	+23	+21	+13	+11	+52	+60	+11	+14	+51
Extended LSTM	RMSE	-1	+2	+5	+7	+25	+28	+39	+8	+13
with actual marketing appeals	bias	-3	+2	+5	+9	-24	-12	+28	+12	+16
	MAPE	+25	+26	+26	+29	+50	+57	+22	+33	+55
Blood Donations	% of total	1.6%	3.3%	8.2%	16.4%	16.4%	8.2%	67.2%	32.8%	100%
Base LSTM	RMSE	+0	+0	-2	-4	-10	-4	+19	-4	**+3**
	bias	+1	+1	-1	-3	-12	+7	+18	-7	**+24**
	MAPE	+2	+2	-1	-3	+5	+11	+11	-6	**+13**
CDNOW	% of total	1.5%	3.0%	7.5%	15.0%	15.0%	7.5%	70.1%	29.9%	100%
Base LSTM	RMSE	+7	+6	+6	+6	+17	+19	-9	+7	**+6**
	bias	+1	+0	-1	+0	+9	+12	-17	+1	**+18**
	MAPE	+1	+0	-1	+0	+3	+2	-10	+1	**+6**
Groceries	% of total	1.6%	3.3%	8.5%	17.0%	17.0%	8.5%	66.1%	33.9%	100%
Base LSTM	RMSE	-27	-20	-13	-7	+38	+47	+35	+6	**+11**
	bias	-11	-14	-17	-19	+50	+80	-3	-23	**+19**
	MAPE	-9	-9	-14	-13	+32	+62	-2	-9	**+9**
Yogurt Purchases	% of total	2.7%	5.4%	13.4%	26.9%	26.9%	13.4%	46.2%	53.8%	100%
Base LSTM	RMSE	+5	+5	+6	+8	+17	+15	+10	+11	**+11**
	bias	-3	-3	-4	-5	+9	+9	-1	-1	**+5**
	MAPE	-3	-3	-4	-4	+9	+9	-1	+3	**+5**
Sunscreen	% of total	1.9%	3.8%	9.6%	19.3%	19.3%	9.6%	61.4%	38.6%	100%
Base LSTM	RMSE	+6	+6	+6	+6	+8	+7	-26	+6	**+1**
	bias	+6	+6	+6	+6	+15	+15	-15	+8	**+19**
	MAPE	+15	+14	+15	+18	+41	+41	-9	+26	**+47**
Pareto/NBD Simulated	% of total	0.4%	0.8%	2%	4.1%	4.1%	2.1%	91.8%	8.2%	100%
High Frequency High Dropout	RMSE	-20	-11	-8	-4	+5	-2	+31	-3	**+9**
Base LSTM	bias	-6	-6	-5	-4	+5	+6	+1	-4	**+2**
	MAPE	-2	-2	-4	-3	+9	+23	+1	-2	**+1**
High Frequency Low Dropout	% of total	2.2%	4.5%	11.3%	22.5%	22.5%	11.3%	54.9%	45.1%	100%
Base LSTM	RMSE	-3	-3	-2	-1	0	+1	+8	-1	**+1**
	bias	-2	-1	-1	-2	+2	-1	0	-2	**-2**
	MAPE	-2	-1	-1	-2	+2	+0	0	-2	**-1**
Low Frequency High Dropout	% of total	0.2%	0.4%	1.1%	2.2%	2.2%	1.1%	95.7%	4.3%	100%
Base LSTM	RMSE	-13	-12	-8	-8	-2	+1	0	-7	**-6**
	bias	-6	-5	-1	+1	-2	-4	-1	0	**+2**
	MAPE	-2	-1	0	0	+10	+5	0	0	**+2**
Low Frequency Low Dropout	% of total	1.3%	2.6%	6.4%	12.9%	12.9%	6.4%	74.3%	25.7%	100%
Base LSTM	RMSE	-9	-6	-5	-4	+4	+6	+1	-3	**-2**
	bias	-6	-5	-5	-5	-2	-3	-3	-4	**+1**
	MAPE	-5	-4	-5	-5	-1	+5	-2	-4	**-2**

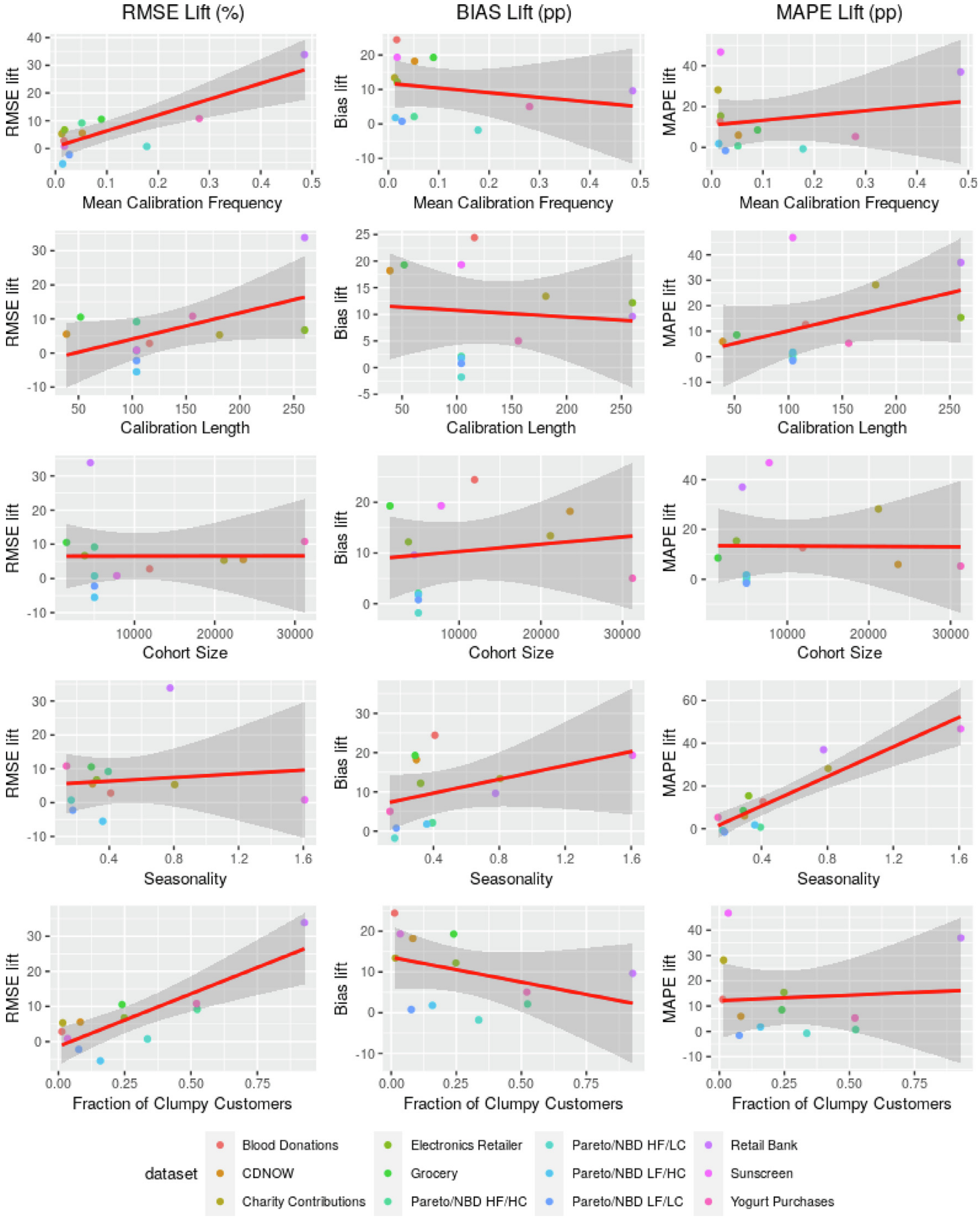


Fig. 13. Influence of selected dataset features on base LSTM model performance lift vs Pareto/NBD.

altogether during holdout. The Extended LSTM model predicts this would result in just 766 transactions being made over the year, a 87% reduction of total volume.

While all these above described scenarios are hypothetical and assume stationary marketing activity effects in both the calibration and forecasting periods, only carefully executed business experiments involving randomized controlled trials could ultimately clarify the credibility of such what-if forecasts. However, they can serve managers as a baseline for benchmarking the impact of certain changes in target marketing actions.

Generally speaking, when using future realizations of covariates as inputs for our simulation model, we need to treat them differently depending on which categories they fall into along the following differentiation: (i) Covariates under the control of the manager (e.g., marketing interventions) can be added according to the strategy/method used to schedule them, while covariates not under the control of the manager need to be generated by a separate model. (ii) Covariates influenced by the customers' future actions (e.g., targeted marketing actions) need to be simulated step-by-step based on the rules or simulation procedure they obey (for example, by linking back to a policy function), while covariates not influenced by the customers' future actions (e.g., non-targeted marketing campaigns) can be simulated in their entirety before running our model. (iii) Covariates with outcome ranges that can potentially increase in the future form a special category. Because it is strictly forward-looking, our LSTM model can only learn from signals that are observed at some point during calibration, which is a limitation not present in BTYD models. However, if we anticipate new signal types in the future, we can reserve "placeholder" variable inputs in the model (e.g., a pandemic indicator variable) which correspond to unknown future events or "shocks". The model will learn that the probability of these placeholder outcomes is zero, and part of its capacity will remain unused, but later we can fine-tune this model using an augmented dataset which includes events of the placeholder kind, to save the computational effort required for a full re-training of the model.

Appendix F. Performance of the LSTM Model and limitations of use

In this Appendix section, we further examine the strengths and weaknesses of out-of-sample predictive performance. Table 12 shows the performance lift relative to all three benchmark models together with a set of descriptive tags for each of the examined real-life transaction data settings. These colored tags represent a simple indication of the overall regularity, randomness, or irregularity of transactions (yellow tag), the cohort size (pink tag), the churn rate (cyan tag), the length of the calibration period (lime tag), the mean transaction frequency (orange tag), the "clumpiness" (teal tag), and the prominence of seasonal patterns (magenta tag). Table 13 reports the performance lift between the LSTM models compared to the Pareto/NBD, broken down by customer transaction activity subgroup, and we note that while the Base LSTM often performs better in forecasting the most active groups of individuals, in a number of cases it is the improved prediction of less active and churned individuals that accounts for most of the performance lift.

For simplicity, we only tag cases where the given feature deviates significantly from the average. This allows us to make a broad recommendation in terms of when **not** to use the LSTM model: The improvement in RMSE is not significant in data settings that combine low-frequency with low proportions of "clumpy" customers, such as Blood Donations and Sunscreen. Even then however, the Base LSTM remains strong on the aggregate level in terms of the aggregate bias and MAPE, and our expectation is that with more available data (Blood Donations contain 11,887 customers, Sunscreen 7,794), the performance gap between the LSTM and the benchmark models would only grow wider. To illustrate the relationship between data characteristics and the Base LSTM model performance further, we regress the selected dataset features on the performance uplift in Fig. 13, showing that conversely, the biggest improvements in individual-level RMSE are observed in scenarios with higher calibration transaction frequency, in cases where longer calibration periods can be observed and also in scenarios where more customers are "clumpy". Understandably, the aggregate MAPE improves most wherever Seasonality is high.

Appendix G. Supplementary material

Supplementary data associated with this article can be found, in the online version, at https://doi.org/10.1016/j.ijresmar.2022.02.007.

References

Abadi, M., & Agarwal, A., et al. (2015). *Tensorflow: Large-scale machine learning on heterogeneous systems*. https://tensorflow.org, [Online; accessed 28 Jan 2019].

Abe, M. (2009). Counting Your Customers One by One: A Hierarchical Bayes Extension to the Pareto/NBD Model. *Marketing Science, 28*(3), 541–553.

Ascarza, E. (2018). Retention futility: Targeting high-risk customers might be ineffective. *Journal of Marketing Research, 55*(1), 80–98.

Ascarza, E., Iyengar, R., & Schleicher, M. (2016). The perils of proactive churn prevention using plan recommendations: Evidence from a field experiment. *Journal of Marketing Research, 53*(1), 46–60.

Bachmann, P., Meierer, M., & Näf, J. (2021). The role of time-varying contextual factors in latent attrition models for customer base analysis. *Marketing Science*.

Balachander, S., & Farquhar, P. H. (1994). Gaining More by Stocking Less: A Competitive Analysis of Product Availability. *Marketing Science, 13*(1), 3–22.

Baum, L. E., & Petrie, T. (1966). Statistical inference for probabilistic functions of finite state markov chains. *The Annals of Mathematical Statistics, 37*(6), 1554–1563.

Blattberg, R. C., & Deighton, J. (1996). Manage marketing by the customer equity test. *Harvard Business Review, 74*(4), 136–144.

Blattberg, R. C., Kim, B. D., & Neslin, S. A. (2008). *Database Marketing: Analyzing and Managing Customers*. Springer.

Chamberlain, B. P., & Cardoso, A, et al. (2017). Customer lifetime value prediction using embeddings. In: *23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*(pp. 1753–1762).

Chollet, F, (2015). *Keras: The python deep learning library*. URL https://github.com/fchollet/keras, [Online; accessed 28 Jan 2019].

Dew, R., & Ansari, A. (2018). Bayesian Nonparametric Customer Base Analysis with Model-Based Visualizations. *Marketing Science, 37*(2), 216–235.

Dzyabura, D., & Peres, R. (2021). Express: Visual elicitation of brand perception. *Journal of Marketing*.

Ebbes, P., Papies, D., & van Heerde, H. (2011). The Sense and Non-Sense of Holdout Sample Validation in the Presence of Endogeneity. *Marketing Science, 30*, 1115–1122.

Fader, P. (2020). *Customer Centricity: Focus on the Right Customers for Strategic Advantage*. Wharton Executive Essentials. Wharton School Press.

Fader, P., Hardie, B., & Chun-Yao, H. (2004). A dynamic changepoint model for new product sales forecasting. *Marketing Science, 23*(1), 50–65.

Fader, P. S., & Hardie, B. G. (2009). Probability models for customer-base analysis. *Journal of Interactive Marketing, 23*(1), 61–69.

Fader, P. S., Hardie, B. G., & Lee, K. L. (2005). Counting your customers the easy way: An alternative to the Pareto/NBD model. *Marketing Science, 24*(2), 275–284.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics, 29*(5), 1189–1232. https://doi.org/10.1214/aos/1013203451.

Gers, F. A., Schmidhuber, J. A., & Cummins, F. A. (2000). Learning to Forget: Continual Prediction with LSTM. *Neural Computation, 12*(10), 2451–2471.

Gönül, F., & ter Hofstede, F. (2006). How to compute optimal catalog mailing decisions. *Marketing Science, 25*, 65–74. https://doi.org/10.1287/mksc.1050.0136.

Goodfellow, I., Bengio, Y., & Courville, A., (2016). *Deep Learning*. MIT Press.

Gupta, S., Hanssens, D., Hardie, B., Kahn, W., Kumar, V., Lin, N., Ravishanker, N., & Sriram, S. (2006). Modeling customer lifetime value. *Journal of Service Research, 9*(2), 139–155.

Gupta, S., & Lehmann, D., (2005). *Managing Customers as Investments: The Strategic Value of Customers in the Long Run*. Wharton school publishing (Wharton School Pub.), ISBN 9780131428959.

Hanley, J. A., Joseph, L., Platt, R. W., Chung, M. K., & Belisle, P. (2001). Visualizing the Median as the Minimum-Deviation Location. *The American Statistician, 55*(2), 150–152.

Hanssens, D. M., Parsons, L. J., & Schultz, R. L. (2003). *Market response models: Econometric and time series analysis* (Vol. 12). (Springer Science & Business Media).

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation, 9*(8), 1735–1780.

Holtrop, N., & Wieringa, J., (2020). *Timing customer reactivation interventions*. SSRN https://doi.org/10.2139/ssrn.3443422.

Homburg, C., Droll, M., & Totzek, D. (2008). Customer prioritization: Does it pay off, and how should it be implemented? *Journal of Marketing, 72*(5), 110–130.

Korkmaz, E., Kuik, R., & Fok, D., (2013). Counting Your Customers: When Will They Buy Next? An Empirical Validation of Probabilistic Customer Base Analysis Models Based on Purchase Timing. *ERIM Report Series* Reference No. ERS-2013-001-LIS, https://ssrn.com/abstract=2198260.

Lemmens, A., & Gupta, S. (2020). Managing churn to maximize profits. *Marketing Science, 39*(5), 956–973.

Malthouse, E. (2009). The results from the lifetime value and customer equity modeling competition. *Journal of Interactive Marketing, 23*, 157–168.

Matuszyk, P., Castillo, R. T., Kottke, D., & Spiliopoulou, M., (2016). A comparative study on hyperparameter optimization for recommender systems. In Lex, E., Kern, R., Felfernig, A., Jack, K., Kowald, D., Lacic, E., (Eds.),*Workshop on Recommender Systems and Big Data Analytics* (RS-BDA'16) @ iKNOW 2016.

McCarthy, D., & Fader, P. (2018). Customer-based corporate valuation for publicly traded noncontractual firms. *Journal of Marketing Research, 55*(1), 617–635.

McCarthy, D. M., Fader, P. S., & Hardie, B. G. (2017). Valuing subscription-based businesses using publicly disclosed customer data. *Journal of Marketing, 81* (1), 17–35.

Mena, C. G., Caigny, A. D., Coussement, K., Bock, K. W. D, & Lessmann, S., (2019) Churn prediction with sequential data and deep neural networks. a comparative analysis.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. In Burges, C. J. C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K. Q., (Eds.), *Advances in Neural Information Processing Systems* (Vol. 26, pp. 3111–3119). Curran Associates, Inc.

Netzer, O., Lattin, J. M., & Srinivasan, V. (2008). A hidden markov model of customer relationship dynamics. *Marketing Science, 27*(2), 185–204.

Ni, J., Neslin, S. A., & Sun, B. (2012). Database submission—the ISMS durable goods data sets. *Marketing Science, 31*(6), 1008–1013.

Platzer, M., (2016). Btydplus: Probabilistic models for assessing and predicting your customer base. https://cran.r-project.org/web/packages/BTYDplus, [Online; accessed 28 Jan 2019].

Platzer, M., & Reutterer, T. (2016). Ticking Away the Moments: Timing Regularity Helps to Better Predict Customer Activity. *Marketing Science, 35*(5), 779–799.

Reinartz, W., & Kumar, V. (2000). On the profitability of long-life customers in a noncontractual setting: An empirical investigation and implications for marketing. *Journal of Marketing, 64*(4), 17–35.

Reutterer, T., Platzer, M., & Schröder, N. (2021). Leveraging purchase regularity for predicting customer behavior the easy way. *International Journal of Research in Marketing, 38*(1), 194–215.

Romero, J., van der Lans, R., & Wierenga, B. (2013). A Partially Hidden Markov Model of Customer Dynamics for CLV Measurement. *Journal of Interactive Marketing, 27*(3), 185–208.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature, 323*, 533–536.

Rust, R., Lemon, K., & Zeithaml, V. (2004). Return on marketing: Using customer equity to focus marketing strategy. *Journal of Marketing, 68*(1), 109–127.

Rust, R. T., Kumar, V., & Venkatesan, R. (2011). Will the frog change into a prince? Predicting future customer profitability. *International Journal of Research in Marketing, 28*(4), 281–294.

Salehinejad, H., & Rahnamayan, S. (2016). Customer shopping pattern prediction: A recurrent neural network approach. In *2016 IEEE Symposium Series on Computational Intelligence (SSCI)* (pp. 1–6).

Sarkar, M., & De Bruyn, A. (2021). Lstm response models for direct marketing analytics: Replacing feature engineering with deep learning. *Journal of Interactive Marketing, 53*, 80–95.

Schmittlein, D. C., Morrison, D. G., & Colombo, R. (1987). Counting your customers: Who they are and what will they do next? *Management Science, 33*(1), 1–24.

Schwartz, E. M., Bradlow, E. T., & Fader, P. S. (2014). Model selection using database characteristics: Developing a classification tree for longitudinal incidence data. *Marketing Science, 33*(2), 188–205.

Schweidel, D., Bradlow, E., & Fader, P. (2011). Portfolio dynamics for customers of a multiservice provider. *Management Science, 57*(3), 471–486.

Schweidel, D. A., & Knox, G. (2013). Incorporating Direct Marketing Activity into Latent Attrition Models. *Marketing Science, 32*(3), 471–487.

Sheil, H., Rana, O., & Reilly, R. (2018) Predicting purchasing intent: Automatic feature learning using recurrent neural networks. *The SIGIR 2018 Workshop On eCommerce, Ann Arbor, Michigan, USA, July 12, 2018, volume 2319 of CEUR Workshop Proceedings* (CEUR-WS.org).

Simester, D. I., Sun, P., & Tsitsiklis, J. N. (2006). Dynamic catalog mailing policies. *Management Science, 52*(5), 683–696.

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, 3104–3112, NIPS'14. Cambridge, MA, USA: MIT Press.

Toth, A., Tan, L., Di Fabbrizio, G., & Datta, A. (2017). Predicting shopping behavior with mixture of RNNs. In *Proceedings of the SIGIR 2017 Workshop on eCommerce (ECOM 17)*.

Wedel, M., & Kannan, P. K. (2016). Marketing analytics for data-rich environments. *Journal of Marketing, 80*(6), 97–121.

Wheat, R. D., & Morrison, D. G. (1990). Estimating Purchase Regularity with Two Interpurchase Times. *Journal of Marketing Research, 27*(1), 87–93.

Zhang, Y., Bradlow, E., & Small, D. (2015). Predicting customer value using clumpiness: From RFM to RFMC. *Marketing Science, 34*(2), 195–208.

Zuboff, S. (2019). *The Age of Surveillance Capitalism: The Fight for a Human Future at the New Frontier of Power* (Profile Books), 1st edition.