

Article

Unveiling the Significance of Individual Level Predictions: A Comparative Analysis of GRU and LSTM Models for Enhanced Digital Behavior Prediction

Burhan Y. Kiyakoglu *  and Mehmet N. Aydin 

Department of Management Information Systems, Kadir Has University, 34083 Istanbul, Turkey;
mehmet.aydin@khas.edu.tr

* Correspondence: burhan.yasin.kiyakoglu@gmail.com

Abstract: The widespread use of technology has led to a transformation of human behaviors and habits into the digital space; and generating extensive data plays a crucial role when coupled with forecasting techniques in guiding marketing decision-makers and shaping strategic choices. Traditional methods like autoregressive moving average (ARMA) can-not be used at predicting individual behaviors because we can-not create models for each individual and buy till you die (BTYD) models have limitations in capturing the trends accurately. Recognizing the paramount importance of individual-level predictions, this study proposes a deep learning framework, specifically uses gated recurrent unit (GRU), for enhanced behavior analysis. This article discusses the performance of GRU and long short-term memory (LSTM) models in this framework for forecasting future individual behaviors and presenting a comparative analysis against benchmark BTYD models. GRU and LSTM yielded the best results in capturing the trends, with GRU demonstrating a slightly superior performance compared to LSTM. However, there is still significant room for improvement at the individual level. The findings not only demonstrate the performance of GRU and LSTM models but also provide valuable insights into the potential of new techniques or approaches for understanding and predicting individual behaviors.



Citation: Kiyakoglu, B.Y.; Aydin, M.N. Unveiling the Significance of Individual Level Predictions: A Comparative Analysis of GRU and LSTM Models for Enhanced Digital Behavior Prediction. *Appl. Sci.* **2024**, *14*, 8858. <https://doi.org/10.3390/app14198858>

Academic Editors: Yujin Lim and Hideyuki Takahashi

Received: 12 August 2024

Revised: 17 September 2024

Accepted: 20 September 2024

Published: 2 October 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: behavioral analytics; individual-level prediction; BTYD; LSTM; GRU; forecasting

1. Introduction

The development of technology has initiated a transformation in our behaviors and habits. In contemporary times, activities such as shopping, watching movies, expressing opinions and managing accounts have migrated to digital platforms. The advent of databases, coupled with enhanced data storage capabilities, has facilitated the seamless recording of individual behaviors.

Businesses, recognizing the value of data, increasingly adopt data-driven methodologies and predictive techniques to optimize user experiences, formulate effective marketing strategies and reach success targets. This data encompasses not only human behaviors but also instances of machine malfunctions, sensor-derived values from machinery, server logs and analogous information. The advancement of data science has resulted in substantial financial benefits for companies engaging in behavior prediction. Despite existing academic studies and applications in behavioral analytics, there is a notable gap necessitating an exploration of the effectiveness of predictive methods applied across diverse domains to anticipate individual behaviors.

In this research, we undertake the task of predicting individual behaviors by leveraging a novel real-world dataset and an online transaction dataset generously shared by UC Irvine. Due to the impracticality of crafting autoregressive moving average (ARMA) models tailored for each person and the inherent limitations of buy till you die (BTYD) models, the

efficacy of deep learning models assumes a vital importance. The precision of individual-level predictions holds significant implications for businesses, enabling them to distinguish between individuals based on their anticipated future behaviors and implement customized actions to enhance overall operational effectiveness. This approach, in contrast to prevalent industry practices relying on past behaviors for segmentation and decision-making, offers a forward-looking perspective. We propose a novel approach and conduct a comparative analysis of the performances of gated recurrent unit (GRU) and long short-term memory (LSTM) models.

This paper is structured as follows: in Section 2, we provide a brief overview of previous studies. Subsequently, in Section 3, we describe the models applied in this study, followed by an explanation of the dataset and evaluation metrics in Section 4. Next, in Section 5, we present the results and compare the performance of the models. Finally, in Section 6, we examine the benefits and constraints of the models and provide recommendations for prospective research in the field.

2. Related Works

One of the important concepts in behavior prediction is behavior patterns which help to detect recurring behaviors and increase the predictability of future behaviors. These patterns have been studied to predict future behaviors by considering them as time series. The main reason for this is the assumption that there is a relationship between recurring behaviors and time. The best example that can be given for this is consumers ordering a certain product at certain intervals or a machine breaking down at certain intervals. Methods such as autoregressive integrated moving average (ARIMA) [1], singular spectrum analysis (SSA) [2] and support vector regression (SVR) [3] have been used to predict the future values of these time series. In the presence of nonlinear and irregular curve shapes in the data, these methods fail to succeed independently. In other words, if an e-commerce site applies a discount to a product that customers repeat buy for a certain period, behaviors become irregular and purchasing behavior in this data creates complex patterns over time. This makes it difficult for the model to perceive different effects. In addition, methods such as ARMA, ARIMA, seasonal autoregressive integrated moving average (SARIMA) or SSA are used to predict the future values of a single time series. Instances of univariate time series, which do not provide individual-level information, are aggregate daily sales or the total number of users accessing the application.

Due to the development of deep learning methods, artificial neural networks (ANN) [1] have been used for behavior prediction. The constraints that affect the performance of the model are that the ANN can take the local minimum as the optimal value and show overfitting to the data. In addition, the ANN makes predictions without establishing the connection between the data and time. Among the deep learning methods, recurrent neural networks (RNN), LSTM [4] and GRU [5] can be shown as models that establish the time connection. RNN has limitations of exploding or vanishing gradient problem [6], and not having long-term memory feature. To overcome these limitations, LSTM or GRU can be used. LSTM and GRU have many applications in different fields, such as natural language processing (NLP) [7,8] and speech recognition [9]. However, LSTM and GRU have been rarely applied in behavior prediction. Examples can be given such as Jiang et al. [10], who examined animal behavior patterns and Damian et al. [11], who proposed a new architecture in recommender systems.

In case one focuses on predictions at the individual level, BTYD [12] models are widely used modeling approaches. This model uses the recency, frequency and monetary (RFM) values which take into account purchase intensity, customer attrition tendency and customer heterogeneity [12]. However, BTYD models also have limitations as they are unable to predict trends; and they cannot leverage features or high-dimensional data.

Table 1 provides a summary of relevant research in behavior analytics, primarily focusing on customer behavior. Notably, Ho et al. [1] and Abbasimehr and Paki [13] delve into univariate time series predictions. In Murray et al. [14], market segmentation precedes

the application of univariate time series for predictions. Fader et al. [15] and Fader and Hardie [16] focus on the RFM approach and BTYD models.

Table 1. A summary of literature on behavior analytics.

Study	Model	Application Area
A comparative study of neural network and Box-Jenkins ARIMA modeling in time series prediction [1]	Box-Jenkins ARIMA, RNN	System Failure Analysis
“Counting Your Customers” the Easy Way: An Alternative to the Pareto/NBD Model [15]	Pareto/NBD	Customer Base Analysis, Repeat Buying
Probability Models for Customer-Base Analysis [16]	RFM	Customer Base Analysis
Customer Shopping Pattern Prediction: A Recurrent Neural Network Approach [17]	RFM, RNN	Customer Behaviour Prediction
Ticking Away the Moments: Timing Regularity Helps to Better Predict Customer Activity [18]	Pareto/NBD	Customer Base Analysis, Purchase Regularity
Forecast of individual customer’s demand from a large and noisy dataset [14]	Market Segmentation, ARIMA	Behavior Patterns, Market Segmentation
Improving time series forecasting using LSTM and attention models [13]	LSTM, Attention	Time series forecasting
Predictive analytics for customer repurchase: Interdisciplinary integration of buy till you die modeling and machine learning [12]	BTYD, Lasso Regression, ANN, LSTM, GRU	Customer Repurchase
Customer base analysis with recurrent neural networks [19]	BTYD, LSTM	Customer Base Analysis

In literature, the application of deep learning frameworks to individual behavioral prediction studies is still limited. Furthermore, these studies exhibit a diversity of approaches when it comes to making predictions at the individual level. Salehinejad and Rahnamayan [17] and Mena et al. [20] proposed a RNN approach to model RFM variable changes over time in the context of customer base behavior. Nonetheless, as the attention continues to center around forecasting RFM metrics that are manually crafted, this method fails to completely exploit the automatic feature extraction potential of deep learning techniques. Chou et al. [12] applied various machine learning and deep learning methods, but they took target values as binary transactions. Sheil et al. [21] used neural networks to create internal representations of transaction histories. Then, they compared the performance of various RNN architectures with traditional machine learning methods to predict purchase intentions. Toth et al. [22] showcased that by employing a combination of RNNs, it is possible to simultaneously approximate multiple intricate functions. Sarkar and De Bruyn [23] demonstrated the utility of a particular type of RNN in aiding marketers who develop response models. They illustrated how this RNN can harness the extensive history of customer-firm interactions associated with observed transaction patterns to predict the most likely future actions of customers. Nevertheless, their methodology is restricted to producing single-point predictions for the upcoming step and for the extension of these predictions over a longer time horizon, it becomes necessary to iteratively re-estimate the model for each subsequent time step. More recently, Valendin et al. [19] demonstrated a flexible framework which we can make individual-level predictions in a future time period and determine the general trend. Li et al. [24] and Lu and Kannan [25] also proposed models for predicting customer churn. Li et al. [24] integrated ChatGPT into the model, while Lu and Kannan [25] utilized Large Language Models (LLM).

3. Models

The BTYD models, which rely on statistical distributions, do not identify patterns through learning. These models generate aggregate predictions at the individual level for the holdout periods, rather than offering predictions for a specific time. Consequently, to obtain weekly predictions, we derive cumulative forecasts and subsequently calculate

the differences for each time point. For instance, assuming a three-week scenario, when calculating predictions for the third week, we first compute forecasts for two and three-week periods and then, calculate the difference between the predictions for three and two-week periods. To overcome this limitation, we use GRU and LSTM.

Figure 1 illustrates the daily login patterns of fifteen randomly selected mobile application users. The terms ‘Past’ and ‘Future’ refer to the calibration and holdout periods, respectively. Within this context, models are employed to predict the future behavior of individual users.

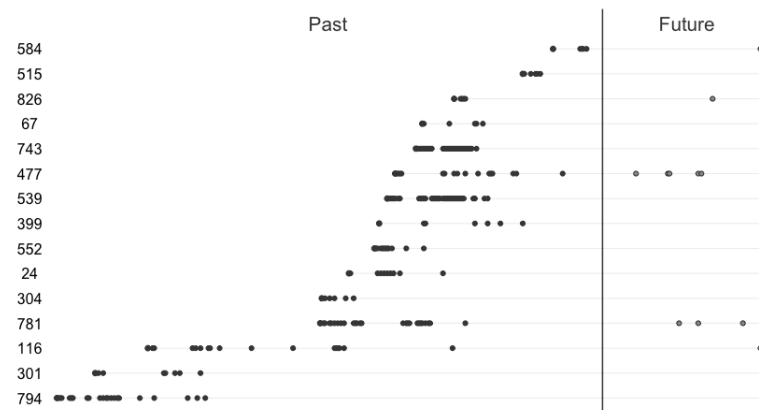


Figure 1. The x-axis in the graph corresponds to the days, with fifteen randomly selected users’ daily logins represented. On the y-axis, the customer IDs are indicated.

3.1. Buy Till You Die (BTYD)

BTYD models are a class of statistical models used in marketing and customer analytics to predict the future behavior of non-contractual customers. These models assume that customers have a certain probability of making purchases in a given time period and that this probability decreases over time. The BTYD models estimate the customer’s purchase probability and the expected number of future purchases. The models discussed in this subsection are all specific implementations within the BTYD framework.

The Negative Binomial Distribution (NBD), Beta Geometric/Negative Binomial Distribution (BG/NBD) and Modified Beta Geometric/Negative Binomial Distribution (MBG/NBD-k) models are the maximum likelihood estimated models. Hierarchical Bayes Extension to the Pareto/NBD (Pareto/NBD (HB)) and Generalized Gamma (Pareto/GGG) models are the Markov Chain Monte Carlo (MCMC) estimated models. To be able to apply these models, recency and frequency values must be created. Monetary values are not always needed as in this study.

NBD model of Ehrenberg [26], assumes that the purchasing process is heterogeneous but remains constant. The time between purchases follows an exponential distribution, while the purchase rate λ varies across customers and is distributed according to a Gamma distribution with parameters r and α .

(M)BG/CNBD-k models [27] make the following assumptions: while active, a customer’s intertransaction times follow an Erlang-k distribution and the customer’s purchase rate λ is distributed across customers according to a Gamma distribution with parameters r and α . After each transaction, there is a constant probability p that the customer will become permanently inactive. This probability is $Beta(a, b)$ -distributed across customers. The only difference between the MBG/CNBD-k and the BG/CNBD-k models is that in the latter, customers are not allowed to drop out after the initial transaction, but only after repeat transactions.

Pareto/NBD model [28] integrates the NBD model with the potential for customers to lapse into inactivity. However, the state of a customer cannot be directly observed and the model relies on drawing inferences from the time that has passed since a customer’s most recent activity. The model assumes that a customer’s lifetime, denoted by τ , follows an ex-

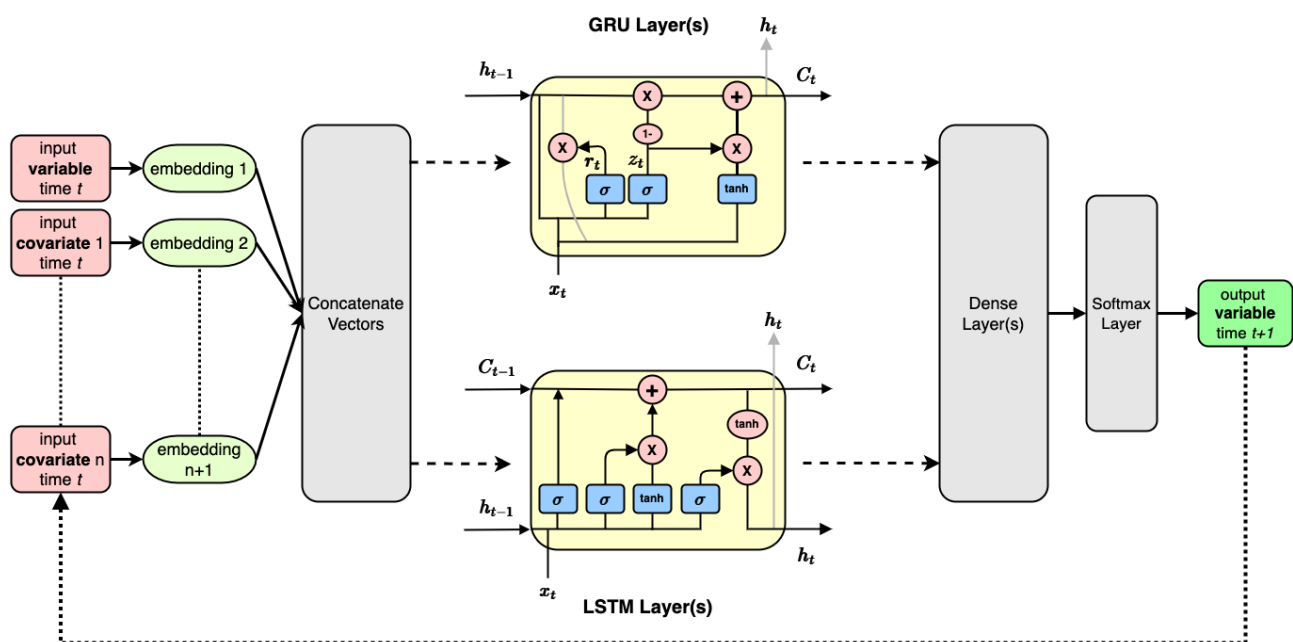
ponential distribution with a parameter of μ and that μ is distributed across customers as a Gamma(s, β) distribution. The only difference between Pareto/NBD (HB) and Pareto/NBD is the utilization of MCMC for parameter estimation in the former.

Platzer and Reutterer [18] presented an additional expansion of the Pareto/NBD model called Pareto/GGG. The shape parameter k differs among customers and follows a Gamma(t, γ) distribution. Consequently, the purchase process conforms to a Gamma-Gamma-Gamma (GGG) mixture distribution that can capture various levels of regularity across customers.

3.2. Devised Deep Learning Architecture

Incorporating insights from NLP research [7,8,29,30] and the work of Valentin et al. [19] on leveraging techniques, we devised a GRU model framework. This pipeline predicts future behavior by analyzing chronological records of the past behaviors, which are individual sequences used for training. In this architecture, many covariates can be added, but in this study we used the base model, which has only the week number as its feature. Unlike BTYD models, which rely on predefined probabilistic distributions and assumptions about customer behavior, the GRU and LSTM models capture complex, nonlinear relationships within the data and allow for more flexibility.

Figure 2 illustrates a high-level diagram of the GRU and LSTM model architectures used in this study. The model framework begins with an input layer that feeds into the embedding layer. The embeddings are then concatenated into a long vector. This vector passes through one or more GRU or LSTM layers, and the output of these layers is fed into one or more dense layers, also known as neural network layers.



In an autoregressive prediction approach, the output variable's value at time $t+1$ is used as an input variable for the model, which then generates the next output value at time $t+2$, and this process continues iteratively.

Figure 2. Proposed Model Framework.

Equation (1) represents the computational flow within a GRU cell, which consists of an update gate, a reset gate, a candidate activation, and the hidden state.

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t]) \quad (1a)$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t]) \quad (1b)$$

$$\tilde{h}_t = \tanh(W \cdot [r_t \odot h_{t-1}, x_t]) \quad (1c)$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \quad (1d)$$

where z_t is the update gate output at time step t , r_t is the reset gate output at time step t , $\tilde{h}_t$ is the candidate activation vector at time step t , h_t is the hidden state at time step t , h_{t-1} is the previous hidden state at time step $t - 1$, x_t is the input at time step t , W_z , W_r and W are weight matrices to be learned, σ denotes the sigmoid function and $\odot$ denotes element-wise multiplication.

Equation (2) represents the computational flow within a LSTM cell, which consists of an input gate, a forget gate, an output gate, a candidate activation, the cell state, and the hidden state.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t]) \quad (2a)$$

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t]) \quad (2b)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t]) \quad (2c)$$

$$\tilde{c}_t = \tanh(W_c \cdot [h_{t-1}, x_t]) \quad (2d)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \quad (2e)$$

$$h_t = o_t \odot \tanh(c_t) \quad (2f)$$

where i_t indicates the input gate output at time step t , f_t determines the forget gate output at time step t . o_t represents the output gate output at time step t . $\tilde{c}_t$ stands for the candidate activation vector at time step t . c_t denotes the cell state at time step t ; and h_t represents the hidden state at time step t . The terms h_{t-1} and x_t represent the previous hidden state and the input at time step t , respectively. The weight matrices are W_i , W_f , W_o and W_c . The functions σ and $\tanh$ indicates the sigmoid and hyperbolic tangent functions, respectively. The symbol $\odot$ represents element-wise multiplication.

At the final step, a softmax layer generates the output prediction by converting the raw output z obtained from a fully connected layer of size k into a multinomial probability distribution over the k observed outcomes of the target variable. The target variables are the number of behaviors, which will be carried out in the upcoming time interval. Each class label determines the probability of observing a specific number of $i - 1$ ($i = 1, 2, \dots, k$) behaviors in the subsequent unit of time. The softmax normalization is computed similarly to multinomial logit regression as follows:

$$\text{softmax}_i(z) = \frac{e^{z_i}}{\sum_{i=1}^k e^{z_i}}, \quad (3)$$

and the softmax layer produces a k -tuple $(p(x_t = c_1), p(x_t = c_2), \dots, p(x_t = c_k))$ at any time step t , which represents the distribution of probabilities among the k neurons in the output layer.

In the final step to generate each prediction, we sample from the multinomial output distribution produced by the lower network layer. As a result, this model does not generate point or interval estimates; each output represents a simulated draw. After each draw, the observation is reinjected into the model as the new behavior variable input to generate the next prediction and this process continues until a sequence of predicted time steps of the desired length is created. In Figure 2, this flow, where each output value becomes the new input, is illustrated with a dotted arrow that bends from the output layer back to the input.

4. Empirical Study

4.1. Dataset

We possess two datasets originating from a mobile application in the financial technology sector, specifically capturing daily user logins.

The second dataset is sourced from UC Irvine and is known as the Online Retail II dataset. This dataset encompasses the comprehensive log of transactions conducted by an online retail business operating in the United Kingdom. The company specializes in retailing unique gift items. Table 2 provides essential details about both datasets in terms of basic information.

Table 2. Descriptive statistics of the used data (These statistics were found after some preprocess).

Dataset	Fintech Mobile Application	Online Retail II
Cohort Size	53,576	4993
Calibration Period Length (Weeks)	49	80
Holdout Period Length (Weeks)	25	25
Minimum Date	4 March 2021	2 December 2009
Training End Date	10 February 2022	16 June 2011
Maximum Date	3 August 2022	8 December 2011

We conducted an empirical study using a weekly time unit as the default, which allowed us to capture the dynamics of the input data while keeping the dataset and model size reasonable. However, we acknowledge that the choice of granularity may depend on the specific context and objectives of the decision makers. Moreover, we note that the mobile application data and the online retail data differ in how they measure user activity: the former uses binary indicators of daily logins and single users may engage up to a maximum of seven times per week, while the latter records the number of transactions per day, which can be higher than 7. Therefore, the results of this study are valuable for understanding behavior across different types of datasets.

4.2. Evaluation Metrics

The evaluation metrics are the mean absolute error (MAE), the root mean square error (RMSE), the bias and the symmetric mean absolute percentage error (SMAPE). These metrics provide different insights into the performance of models at the individual level. Bias determines the percentage of the difference between the total estimations and actuals. It is defined as

$$\text{Bias} = \frac{\sum_{t=1}^n \sum_{i=1}^k (\hat{y}_{i,t} - y_{i,t})}{\sum_{t=1}^n \sum_{i=1}^k y_{i,t}} \times 100 \quad (4)$$

where $\hat{y}_t$ is the predicted value and y_t is the actual value for individual i at point t .

Since we have many 0 values, it is not possible to use MAPE. Thus, we used SMAPE; and its definition is given by Equation (5):

$$\text{SMAPE} = \frac{1}{n} \sum_{t=1}^n \frac{\sum_{i=1}^n |\hat{y}_t - y_t|}{\frac{|\hat{y}_t| + |y_t|}{2}} \times 100 \quad (5)$$

where $\hat{y}_t$ represents the predicted value and y_t denotes the actual value at point t .

We also calculate MAE and RMSE at individual level. We determined them as Equations (6) and (7):

$$\text{MSE} = \frac{1}{n} \sum_{t=1}^n \sum_{i=1}^k (y_{i,t} - \hat{y}_{i,t})^2 \quad (6)$$

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{t=1}^n \sum_{i=1}^k (y_{i,t} - \hat{y}_{i,t})^2} \quad (7)$$

where $\hat{y}_t$ is the estimated value and y_t is the actual value for individual i at point t .

5. Results

We commence with the outcomes that derived from the mobile application data, as summarized in Table 3. In each model, we made predictions for the weekly login counts of the users. For Pareto/NBD (HB) and Pareto/GGG models, some login predictions exceeded 7. To align with the practical constraint, that is, 7 represents the upper limit for weekly logins per user, these predictions were adjusted to this maximum value. Additionally, Pareto/NBD (HB) and Pareto/GGG models yielded negative results, which we converted to 0. In Table 3, we did not include BG/CNBD-k and MBG/CNBD-k models because the optimum k parameters for the BG/CNBD-k and MBG/CNBD-k models were 1, which means that they do not differ from BG/CNBD and MBG/CNBD.

Table 3. Summary of Results for mobile application data: Total number of actuals and estimations, Mean Absolute Error (MAE), Root Mean Square Error (RMSE), bias and Symmetric Mean Absolute Percentage Error (SMAPE).

Model	Actuals	Estimation	MAE	RMSE	Bias (%)	SMAPE
GRU	125,342	122,365	0.178	0.681	−2.375	0.659
LSTM	125,342	129,139	0.183	0.664	3.029	0.628
Pareto/NBD (HB)	125,342	167,959	0.143	0.530	34.000	0.687
MBG/NBD	125,342	168,229	0.139	0.510	34.216	1.960
Pareto/GGG	125,342	168,819	0.145	0.520	34.687	0.704
BG/NBD	125,342	170,077	0.142	0.507	35.691	1.960
NBD	125,342	654,812	0.461	0.980	422.421	1.944

For GRU and LSTM models, we have three hyperparameters which are memory units, dense units and simulation numbers. The GRU model was configured with 256 memory units and 32 dense units, while the LSTM model employed 128 memory units and 128 dense units during training. Additionally, simulation numbers were taken as 23 for both models. The impact of these hyperparameters presented in Figures 3–5. It is apparent that the bias is influenced by the choice of the number of memory and dense units. When simulation numbers increase, SMAPE values increase. This pattern seems plausible as more simulations lead to slightly elevated predictions for zero values, contributing to a greater deviation in SMAPE calculations. Conversely, the stability of MAE and RMSE values remains consistent. The results emphasize the impact of simulation numbers and underscore the persistent advantages of GRU and LSTM, despite variations in simulation numbers compared to BTYD models.

Upon reviewing Figure 6 and Table 3, GRU and the LSTM have the least bias and the lowest SMAPE values; and they capture the trends better than the BTYD models. GRU has the lowest difference between the actuals and estimations, lowest bias and second lowest SMAPE values. However, it is important to note that MAE and RMSE values are relatively higher than most BTYD models.

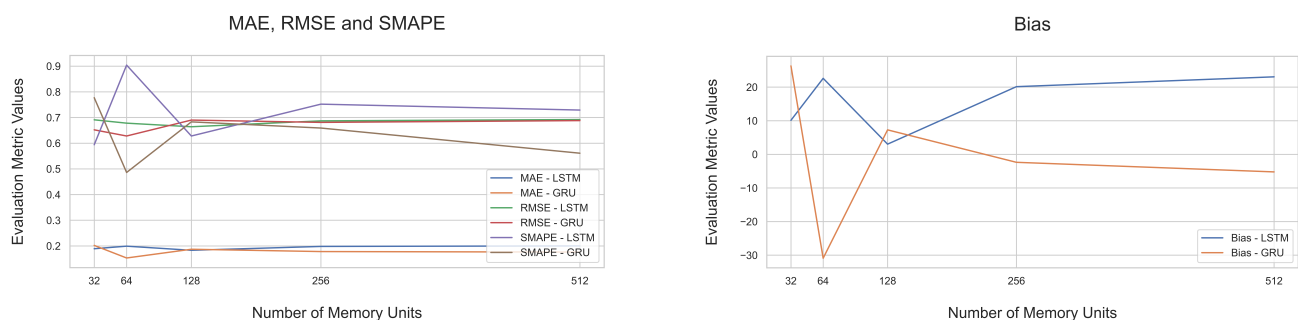


Figure 3. Evaluation metrics vary with the memory units for mobile application data.

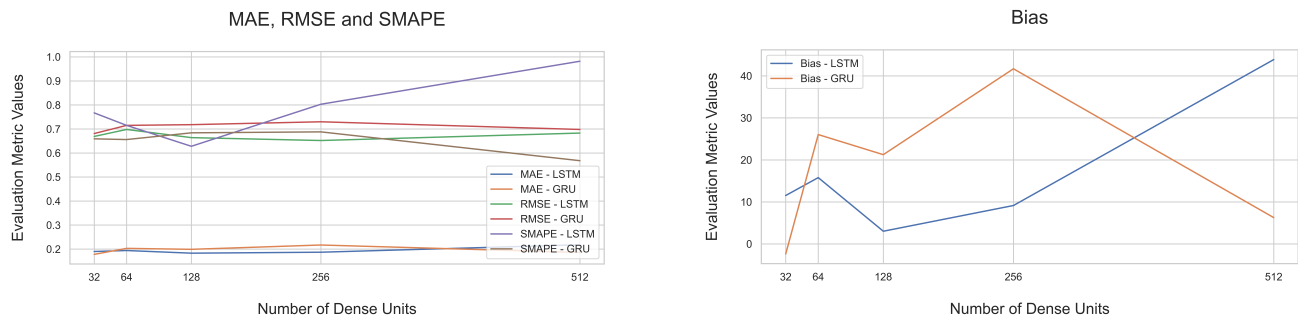


Figure 4. Evaluation metrics vary with the dense units for mobile application data.

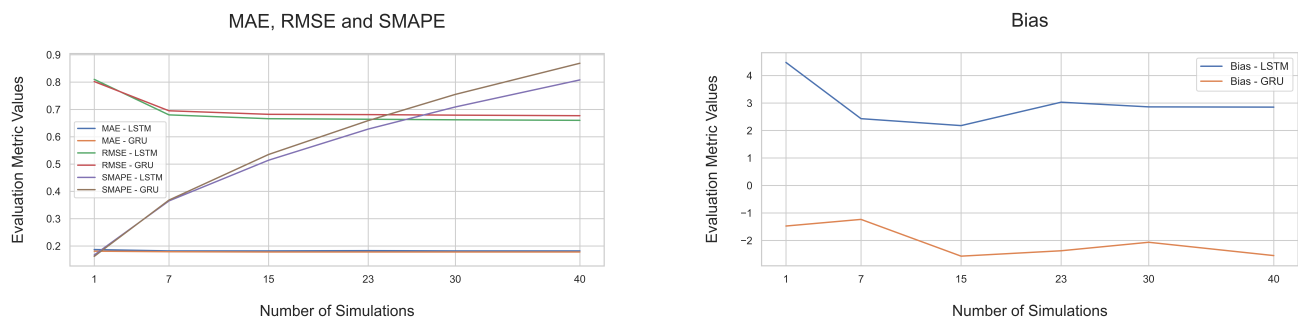


Figure 5. Evaluation metrics vary with the number of simulations for mobile application data.

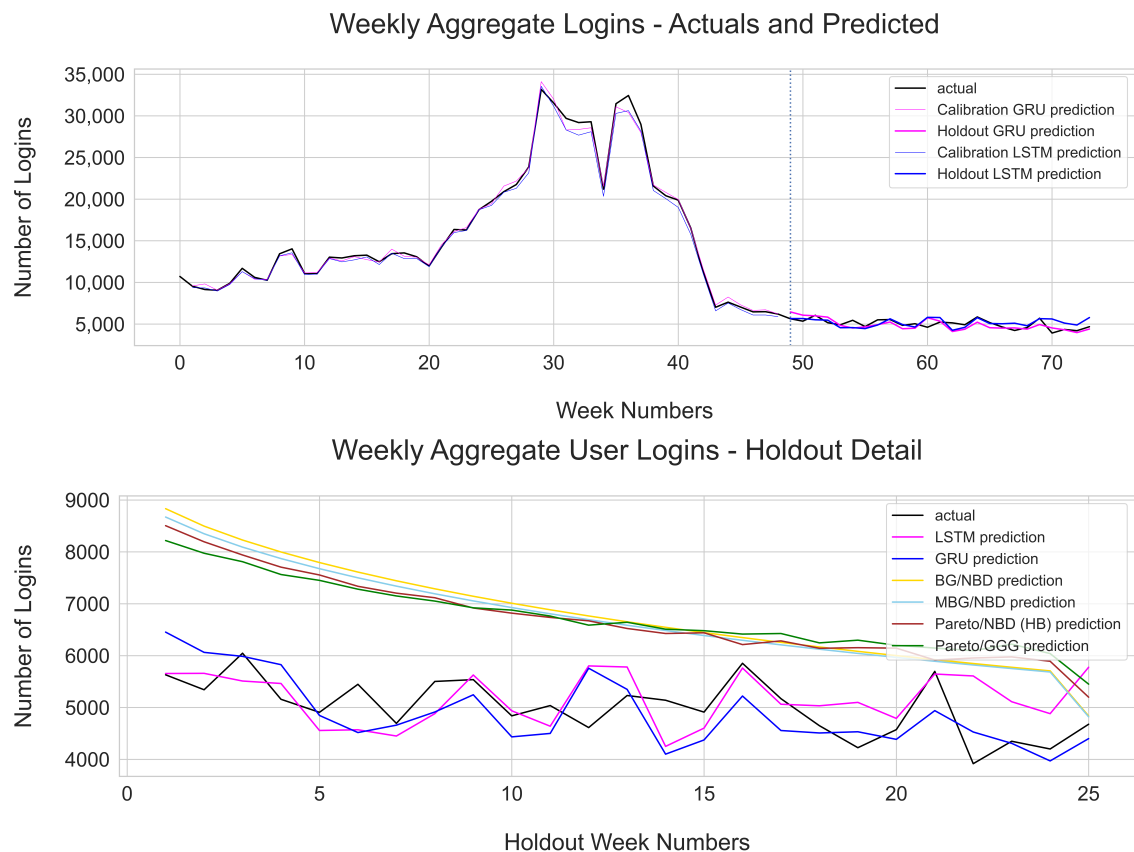


Figure 6. Aggregate weekly actual logins and predictions for mobile application data. The dotted vertical blue line in the upper line chart separates the calibration and holdout periods.

Further, Table 4 presents the evaluation metrics for each model applied on Online Retail II dataset. It is noteworthy that the Pareto/NBD (HB) and Pareto/GGG models produced negative results, which we adjusted to zero.

Table 4. Summary of Results for Online Retail II data: Total number of actuals and estimations, MAE (Mean Absolute Error), RMSE (Root Mean Square Error), bias and SMAPE (Symmetric Mean Absolute Percentage Error).

Model	Actuals	Estimation	MAE	RMSE	Bias (%)	SMAPE
GRU	8469	8330	0.127	0.342	−1.643	1.181
LSTM	8469	8300	0.127	0.349	−2.000	1.120
MBG/NBD	8469	8035	0.109	0.286	−5.123	1.972
BG/NBD	8469	8018	0.109	0.286	−5.321	1.972
Pareto/GGG	8469	8972	0.116	0.299	5.936	1.278
NBD	8469	9072	0.116	0.287	7.125	1.971
Pareto/NBD (HB)	8469	10,414	0.126	0.303	22.966	1.316

In both the GRU and LSTM models, we employed 256 memory units and 32 dense units, conducting 23 training simulations.

Upon examining Figure 7, and Table 4, it becomes apparent that both the GRU and LSTM models perform better than the BTYD models in capturing the underlying trend. GRU exhibits the smallest disparity between actual and estimated values, the least bias and the lowest SMAPE values although the MAE and RMSE values are higher compared to all the BTYD models.

Comparisons between the weekly aggregated actual data and the corresponding predictions for the LSTM and GRU models are illustrated in Figure 7. This visualization clearly demonstrates that LSTM and GRU effectively capture the overall trend in the data.

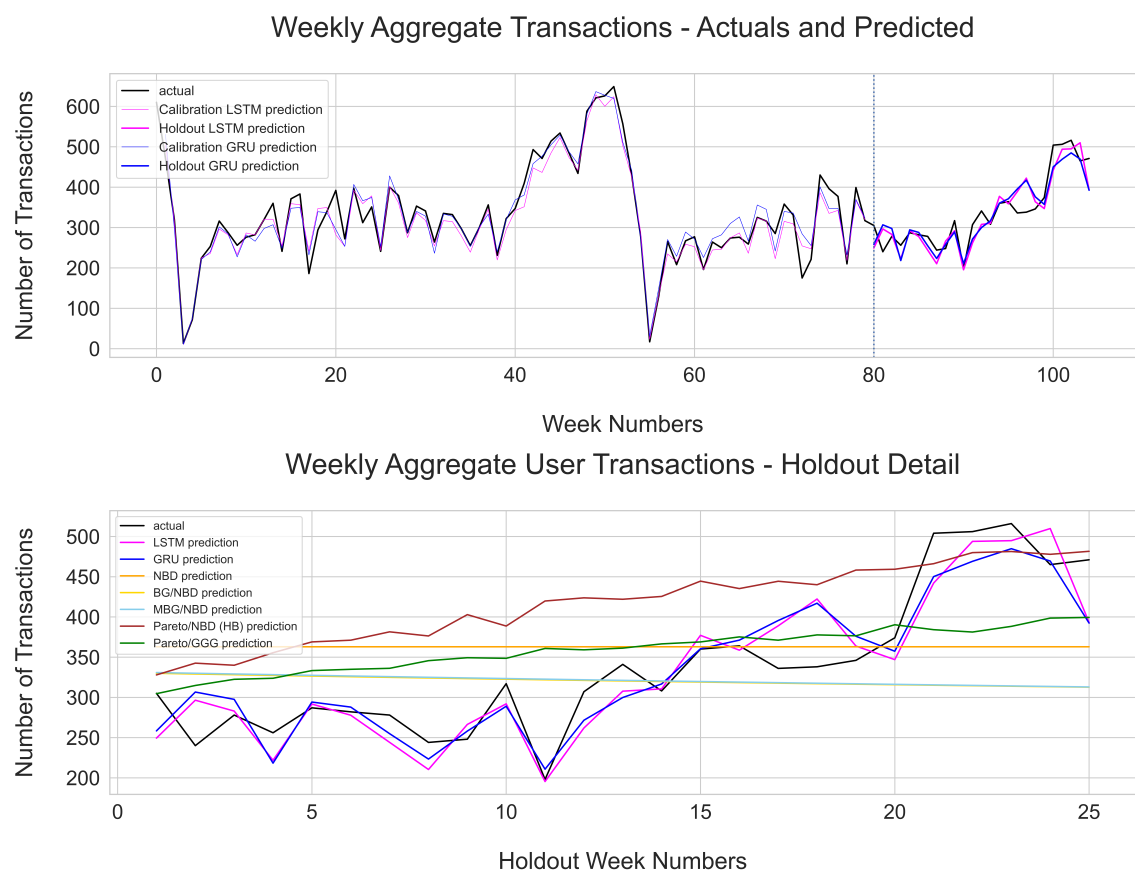


Figure 7. Aggregate weekly transactions and predictions for Online Retail II data. The dotted vertical blue line in the upper line chart separates the calibration and holdout periods.

The MAE and RMSE metrics for the BTYD models were observed to be lower than those of the GRU and LSTM models. As in the user login dataset, users exhibit infrequent

activity in the Online Retail II dataset. The BTYD models demonstrate higher accuracy in terms of MAE and RMSE due to their tendency to underforecast. This means that BTYD models often forecast better near zero transactions or logins, which is probably the main reason for this result.

In relation to performance, both GRU and LSTM models demonstrated superior results in terms of bias and SMAPE compared to the BTYD models across both datasets. The GRU achieves slightly better results compared to the LSTM. Notably, these recurrent neural network models effectively captured the underlying patterns and trends inherent in the data.

6. Discussion

This research focuses on elucidating the primary contributions and limitations, utilizing a deep learning architecture to predict individual-level behavior across datasets with diverse characteristics. We achieved the best results in terms of accuracy with the GRU model, which is a type of recurrent neural network that can capture temporal dependencies. This model allows us to make predictions for each week during the calibration period, which reveals the overall trends of user and customer behavior. These predictions can help enterprises make decisions based on future expectations rather than past patterns. This can enhance the understanding of individual-level behavior and enable better strategies for engagement and retention. For instance, the aggregated results can help identify trends at the managerial level, while the individual-level predictions can enable customer segmentation and churn detection at a more personalized level. Thus, the deep learning approach has great potential for optimizing business strategies and improving customer experiences. However, we also note that scalability issues may arise when dealing with large data and many individuals.

In comparison, the widely used BTYD models, which are statistical models typically based on some statistical distributions, do not learn any patterns. In this study, we observed that the BTYD models provided better MAE and RMSE values. However, the bias results indicated potential misleading tendencies; and these values may still be considered successful. If we apply the proposed methodology on data with less zero values at individual level (e.g., less inactivity of users), we might have better MAE and RMSE values. Also, as Valentin et al. [19] mentioned in their study, if we encounter more frequent cases, employing LSTM could lead to greater benefits. On the other hand, when the model predicts logins or transactions more than zero for the users or customers who have zero logins or transactions, it might determine the users or customers who are more likely to make logins or transactions. These individuals might be treated differently by decision makers. Considering the nature of the datasets, where a significant portion of users had zero logins during most of the calibration period, predicting user activity as active or not may hold more value than calculating the difference between login counts in various cases.

In addition to BTYD, several methods have been proposed for individual-level predictions. Salehinejad and Rahnamayan [17] model the temporal evolution of RFM values, Chou et al. [12] use binary outcomes as targets, and Sarkar and De Bruyn [23] forecast single-point values. These studies address important aspects such as handcrafted features, RFM interdependencies, and target diversity. The framework proposed in this study can be adapted to various target settings and benefit from incorporating handcrafted features. However, integrating such features might be challenging for companies with large datasets and real-time needs. In these cases, invariant features may be more practical than time-varying features. Additionally, the impact of these covariates may vary across different types of datasets (e.g., e-commerce data, mobile application usage data).

Future research could explore how BTYD and deep learning models might complement each other. Investigating hybrid or ensemble models might enhance both individual-level and aggregate results. Furthermore, future studies should focus on effectively integrating covariates across diverse datasets. Incorporating additional features (e.g., variables that determine economic trends, socioeconomic variables) that may influence individual

behavior could also be beneficial. Moreover, addressing scalability challenges is crucial; therefore, future research might investigate various segmentation strategies to manage large datasets more effectively. Exploring different segmentation approaches could facilitate the application of specialized models tailored to distinct data subsets, which could lead to improved overall model performance. These efforts will advance the application of deep learning pipelines in real-world scenarios and contribute to the development of best practices across diverse datasets.

7. Conclusions

This study addresses an important task of predicting individual-level behavior, a critical aspect for businesses across diverse industries. By employing a deep learning architecture, particularly the GRU model, on two distinct datasets, our research significantly contributes to the field. The utilization of GRU not only proves effective in mitigating bias but also offers valuable insights into nuanced individual behavior trends. This predictive approach empowers enterprises to formulate targeted strategies based on anticipated future behaviors, transcending reliance on historical patterns and deepening comprehension of individual-level behavior dynamics.

A pertinent comparison with prevalent BTYD models reveals nuanced trade-offs; while statistical models excel in terms of MAE and RMSE values, the deep learning approach demonstrates superior bias results, shedding light on potential misleading tendencies within BTYD models.

Despite the strides made in individual-level predictions, opportunities for refinement persist. Future research endeavors may explore ways to enhance prediction accuracy and precision, considering the incorporation of attention models. Furthermore, extending the application of deep learning methods to datasets exhibiting diverse characteristics will offer a more comprehensive understanding of individual-level behavior in varied contexts.

Author Contributions: Conceptualization, B.Y.K. and M.N.A.; methodology, B.Y.K. and M.N.A.; software, B.Y.K.; validation, B.Y.K. and M.N.A.; formal analysis, B.Y.K.; investigation, B.Y.K.; data curation, B.Y.K.; writing—original draft preparation, B.Y.K.; writing—review and editing, M.N.A.; visualization, B.Y.K.; supervision, M.N.A. All authors have read and agreed to the published version of the manuscript.

Funding: The APC was funded by Burhan Y. Kiyakoglu.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The Online Retail II dataset used in the study is openly available in the UC Irvine Machine Learning Repository at <https://archive.ics.uci.edu/dataset/502/online+retail+ii>, accessed on 21 June 2024. The mobile application data were obtained from a third party and are subject to access restrictions, requiring permission from the provider.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Ho, S.L.; Xie, M.; Goh, T.N. A comparative study of neural network and Box-Jenkins ARIMA modeling in time series prediction. *Comput. Ind. Eng.* **2002**, *42*, 371–375. [\[CrossRef\]](#)
2. Rocco S, C.M. Singular spectrum analysis and forecasting of failure time series. *Reliab. Eng. Syst. Saf.* **2013**, *114*, 126–136. [\[CrossRef\]](#)
3. das Chagas Moura, M.; Zio, E.; Lins, I.D.; Droguett, E. Failure and reliability prediction by support vector machines regression of time series data. *Reliab. Eng. Syst. Saf.* **2011**, *96*, 1527–1534. [\[CrossRef\]](#)
4. Graves, A. Supervised Sequence Labelling. In *Supervised Sequence Labelling with Recurrent Neural Networks*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 5–13. [\[CrossRef\]](#)
5. Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv* **2014**, arXiv:1406.1078. [\[CrossRef\]](#)
6. Gers, F.A.; Schmidhuber, J.; Cummins, F. Learning to forget: Continual prediction with LSTM. *Neural Comput.* **2000**, *12*, 2451–2471. [\[CrossRef\]](#) [\[PubMed\]](#)

7. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to sequence learning with neural networks. *arXiv* **2014**, arXiv:1409.3215. [\[CrossRef\]](#)
8. Bosco, G.L.; Pilato, G.; Schicchi, D. A neural network model for the evaluation of text complexity in Italian language: A representation point of view. *Procedia Comput. Sci.* **2018**, *145*, 464–470. [\[CrossRef\]](#)
9. Ravanelli, M.; Brakel, P.; Omologo, M.; Bengio, Y. Light gated recurrent units for speech recognition. *IEEE Trans. Emerg. Top. Comput. Intell.* **2018**, *2*, 92–102. [\[CrossRef\]](#)
10. Jiang, W.; Wang, K.; Lv, Y.; Guo, J.; Ni, Z.; Ni, Y. Time series based behavior pattern quantification analysis and prediction—A study on animal behavior. *Phys. A Stat. Mech. Its Appl.* **2020**, *540*, 122884. [\[CrossRef\]](#)
11. Damian, A.; Piciu, L.; Turlea, S.; Tapus, N. Advanced customer activity prediction based on deep hierarchic encoder-decoders. In Proceedings of the 2019 22nd International Conference on Control Systems and Computer Science (CSCS), Bucharest, Romania, 28–30 May 2019; IEEE: New York, NY, USA, 2019; pp. 403–409. [\[CrossRef\]](#)
12. Chou, P.; Chuang, H.H.C.; Chou, Y.C.; Liang, T.P. Predictive analytics for customer repurchase: Interdisciplinary integration of buy till you die modeling and machine learning. *Eur. J. Oper. Res.* **2022**, *296*, 635–651. [\[CrossRef\]](#)
13. Abbasimehr, H.; Paki, R. Improving time series forecasting using LSTM and attention models. *J. Ambient Intell. Hum. Comput.* **2022**, *13*, 673–691. [\[CrossRef\]](#)
14. Murray, P.W.; Agard, B.; Barajas, M.A. Forecast of individual customer’s demand from a large and noisy dataset. *Comput. Ind. Eng.* **2018**, *118*, 33–43. [\[CrossRef\]](#)
15. Fader, P.S.; Hardie, B.G.; Lee, K.L. “Counting your customers” the easy way: An alternative to the Pareto/NBD model. *Market. Sci.* **2005**, *24*, 275–284. [\[CrossRef\]](#)
16. Fader, P.S.; Hardie, B.G. Probability models for customer-base analysis. *J. Interact. Mark.* **2009**, *23*, 61–69. [\[CrossRef\]](#)
17. Salehinejad, H.; Rahnamayan, S. Customer shopping pattern prediction: A recurrent neural network approach. In Proceedings of the 2016 IEEE Symposium Series on Computational Intelligence (SSCI), Athens, Greece, 6–9 December 2016; IEEE: New York, NY, USA, 2016; pp. 1–6. [\[CrossRef\]](#)
18. Platzer, M.; Reutterer, T. Ticking away the moments: Timing regularity helps to better predict customer activity. *Mark. Sci.* **2016**, *35*, 779–799. [\[CrossRef\]](#)
19. Valendin, J.; Reutterer, T.; Platzer, M.; Kalcher, K. Customer base analysis with recurrent neural networks. *Int. J. Res. Mark.* **2022**, *39*, 988–1018. [\[CrossRef\]](#)
20. Mena, C.G.; De Caigny, A.; Coussemont, K.; De Bock, K.W.; Lessmann, S. Churn prediction with sequential data and deep neural networks. a comparative analysis. *arXiv* **2019**, arXiv:1909.11114. [\[CrossRef\]](#)
21. Sheil, H.; Rana, O.; Reilly, R. Predicting purchasing intent: Automatic feature learning using recurrent neural networks. *arXiv* **2018**, arXiv:1807.08207. [\[CrossRef\]](#)
22. Toth, A.; Tan, L.; Di Fabrizio, G.; Datta, A. Predicting Shopping Behavior with Mixture of RNNs. In Proceedings of the eCom@SIGIR, Tokyo, Japan, 11 August 2017.
23. Sarkar, M.; De Bruyn, A. LSTM response models for direct marketing analytics: Replacing feature engineering with deep learning. *J. Interact. Mark.* **2021**, *53*, 80–95. [\[CrossRef\]](#)
24. Li, Y.; Xia, G.; Wang, S.; Li, Y. A deep multimodal autoencoder-decoder framework for customer churn prediction incorporating chat-GPT. *Multimed. Tools Appl.* **2023**, 1–27. [\[CrossRef\]](#)
25. Lu, Z.; Kannan, P. Measuring the Synergy Across Customer Touchpoints Using Transformers. 2024. Available online: <https://ssrn.com/abstract=4684617> (accessed on 1 September 2024).
26. Ehrenberg, A.S. The pattern of consumer purchases. *J. R. Stat. Soc. Ser. C Appl. Stat.* **1959**, *8*, 26–41. [\[CrossRef\]](#)
27. Reutterer, T.; Platzer, M.; Schröder, N. Leveraging purchase regularity for predicting customer behavior the easy way. *Int. J. Res. Mark.* **2021**, *38*, 194–215. [\[CrossRef\]](#)
28. Schmittlein, D.C.; Morrison, D.G.; Colombo, R. Counting your customers: Who-are they and what will they do next? *Manag. Sci.* **1987**, *33*, 1–24. [\[CrossRef\]](#)
29. Li, J.; Xu, Q.; Shah, N.; Mackey, T.K. A machine learning approach for the detection and characterization of illicit drug dealers on instagram: Model evaluation study. *J. Med. Int. Res.* **2019**, *21*, e13803. [\[CrossRef\]](#)
30. Tang, C.; Plasek, J.M.; Zhang, H.; Kang, M.J.; Sheng, H.; Xiong, Y.; Bates, D.W.; Zhou, L. A temporal visualization of chronic obstructive pulmonary disease progression using deep learning and unstructured clinical notes. *BMC Med. Inform. Decis. Mak.* **2019**, *19*, 258. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.