Journal of Computer Science

Original Research Paper

Sales Forecasting Models: Comparison between ARIMA, LSTM and Prophet

Dmitry Brykin

Department of Corporate Information Systems, Moscow Institute of Physics and Technology, Moscow, Russia

Article history
Received: 29-02-2024
Revised: 16-04-2024
Accepted: 27-05-2024

Email: brykin.dmitry@mail.ru

Abstract: Sales forecasting is crucial for business planning and resource allocation. Data-driven approaches have become popular in this field. This study compares the performance of three forecasting models: Autoregressive Integrated Moving Average (ARIMA), Long Short-Term Memory (LSTM) and Prophet within the context of specific sales categories derived from acquiring data provided by a bank. This study uses a time series dataset provided by Tink off data, which covers various sales categories and time intervals. These categories, including pharmacies, railway tickets, books, sporting goods and fuel stations, present unique forecasting challenges because of their distinct demand patterns and potential for high volatility. Through a comparative analysis focusing on accuracy, robustness and computational efficiency, the study reveals that while all models demonstrate efficacy in certain scenarios, their performance varies depending on the specific category and forecasting horizon. ARIMA exhibits consistent accuracy across categories, particularly for daily predictions, aligning with its strength in capturing trends and seasonality. LSTM, on the other hand, shows promise for hourly predictions in categories like fuel stations, leveraging its ability to learn long-term dependencies. However, the LSTM model shows inconsistent results, sometimes outperforming others, but with varying performance across runs. This study provides insights for practitioners within the banking and financial sectors seeking to select the most appropriate forecasting model based on their specific sales categories and forecasting needs.

Keywords: Sales Forecasting, Arima, Lstm, Prophet, Time Series, Acquiring

Introduction

Accurate sales forecasting is essential for effective business planning, allowing organizations to align resources with anticipated demand. In today's data-rich landscape, the appeal of data-driven approaches to sales forecasting has grown exponentially, offering the opportunity to optimize operations, reduce waste and improve customer satisfaction. Precise sales forecasting can help organizations improve inventory optimization, reducing waste and enhancing customer satisfaction. The rising abundance of data has resulted in a growing interest in the use of data-driven approaches to sales forecasting (Raban and Gordon, 2020).

This study aimed to evaluate the performance of three forecasting models, namely ARIMA, LSTM and Prophet, across different sales categories and time intervals. ARIMA is a common time series forecasting model that has been widely used for sales forecasting. LSTM represents a type of recurrent neural network that has shown promising results in several applications, such as sales forecasting. Prophet is a relatively new forecasting model developed by Facebook that incorporates seasonal patterns, holidays and other events to improve prediction accuracy.

We utilized a time-series dataset provided by Tinkoff Data, the analytics division of TCS Group, which encompasses impersonal sales in several categories, including pharmacies, train tickets, books, sports goods and petrol stations. Our dataset covers a one-year period and contains payment data at the category level.

Our objective is to objectively compare the accuracy, robustness and computational efficiency of various algorithms for sales forecasting in different categories.

Our aim is to compare the performance of these algorithms in terms of accuracy, robustness and computational efficiency and to provide insights into which algorithm may be best suited for sales forecasting

DOI: 10.3844/jcssp.2024.1222.1230

in these categories. With the diversity of sales categories, organizations face the daunting task of anticipating future demand dynamics. This study aims to evaluate the performance and comparative characteristics of three popular forecasting models: ARIMA, LSTM and Prophet. We aim to identify the nuances of their application across different timeframes and sales categories, such as those derived from acquiring data provided by a large bank, including pharmacies, railway tickets, books and sporting goods, recognizing that a one-size-fits-all approach may not be optimal. Existing research on sales forecasting often focuses on broader retail sectors or product categories, leaving a gap in knowledge regarding the unique characteristics and challenges presented by these specific domains and the data source. This study contributes to the field by offering a focused comparative analysis of ARIMA, LSTM and prophet models within these unique sales domains. Although these models have demonstrated their capabilities in various forecasting tasks, their performance nuances within these specific domains remain underexplored. Our research conducts a targeted comparative analysis of these models, focusing on accuracy and robustness across various forecasting horizons. This granular evaluation provides valuable insights for practitioners within the financial and banking sectors in selecting the most suitable model based on their specific sales category and forecasting needs. It is hoped that this data will help business entities choose the best forecasting model to meet their specific operational requirements.

In addition, in recent years, banks have increasingly turned to data analytics and machine learning algorithms to help them make more informed decisions and improve their performance. This article provides more information on predicting acquisition operations, which can be useful for banks in planning the load on their infrastructure, as well as forecasting card cash flows by purchase categories.

Literature Review

Time series forecasting has been a subject of significant interest in the fields of data science and machine learning for decades. Numerous statistical and machine learning techniques have been devised to address the challenge of time series forecasting and they have been applied in a range of fields such as finance, energy and retail. There are many algorithms for time series forecasting. Some common examples of time-series sales forecasting are ARIMA, LSTM, prophet, K Nearest Neighbor (KNN) and Support Vector Machine (SVM). We decided to use the first three models because of some limitations that KNN and SVM models have.

KNN does not create a generalized separable model and instead relies solely on the training data for predictions (lazy classifier). KNN does not provide insight into the importance of each predictor as ARIMA does. The algorithm is computationally intensive, especially when dealing with numerous predictors or training records (Hachcham, 2023). In this study, we have an initial dataset with over 43000 rows. The process of finding nearest neighbors can be slow in this case. Another limitation is the choice of distance metric during prediction. There is no general way to determine the best distance metric, which can affect the accuracy of the predictions. At the same time, KNN performance relies on this metric a lot (Chomboon *et al*., 2015). Overall, KNN is best suited for situations with a small number of training records and predictors and when the data can be stored in active memory.

Another model that we decided to skip is SVM. The effectiveness of SVM, especially in non-linear cases, heavily depends on the choice of the kernel function, which is not always straightforward and may require domain-specific knowledge (Bennett and Campbell, 2000). This may be problematic because we have a wide range of domains in our dataset. Training SVMs can be computationally intensive, particularly for large datasets, making them less efficient for large-scale time series data. Similar to many machine learning models, SVMs lack straightforward interpretability, especially with non-linear kernels. Understanding how feature values contribute to the outcome can be non-intuitive, making it difficult to explain the model’s decisions in simple terms. The same can apply to the LSTM model; however, it usually provides better results (Lakshminarayanan and McCrae, 2019; Sabyrzhan *et al*., 2021; Tripathi, 2023).

ARIMA has been widely used in various applications. It has been used successfully in applications such as stock market prediction (Idrees *et al*., 2019) and electricity load forecasting (Nepal *et al*., 2020). Some authors use ordinary linear regression (Vyas and Hemrajani, 2021); however, ARIMA has a greater ability to identify patterns in time series due to additional components.

LSTM has gained popularity because of its ability to capture long-term dependencies in time series data. Cryptocurrency price forecasting (Hamayel and Owda, 2021) and weather forecasting (Kim *et al*., 2020) are among the numerous fields in which LSTM has been employed.

Prophet is easy to use and can handle seasonality and trend changes. Prophet has been successfully used in applications such as website traffic forecasting (Subashini *et al*., 2019) and retail sales forecasting (Lisova, 2021).

About the Used Algorithms

ARIMA

The ARIMA algorithm has three components: (AR) Autoregressive, (I) Integrated and (MA) Moving Average. The AR component models the linear dependence between the current and past values of the time series. The order of the AR component is represented by p, which is the number of values used in the model (Hyndman and Athanasopoulos, 2018).

The *I* component removes the trend by differentiating the data. The order of the *I* component is denoted by the letter *d*, which represents the number of differentiations.

The *MA* component models the linear dependence between the current value and past residual errors. The order of the *MA* component is denoted by *q*, which represents the number of errors in past predictions.

The entire model can be represented by formula 1, which can be found in Hyndman and Athanasopoulos (2018). ϕ_p and θ_q are the autoregression and moving average coefficients, respectively:

$$y_t^{'} = c + \phi_1 y_{t-1}^{'} + \cdots + \phi_p y_{t-p}^{'} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t \quad (1)$$

The ARIMA algorithm selects suitable values of *p*, *d* and *q* by analyzing the autocorrelation and partial autocorrelation of the time series. After determining the values of *p*, *d* and *q*, the ARIMA model is optimized for the data using Maximum Likelihood Estimation (MLE). Then, this model is used to make forecasts for future periods.

In general, ARIMA is a powerful algorithm for forecasting time series, which has a long history and extensive areas of application. It was selected due to not stationary nature of our dataset.

LSTM

The architecture commonly referred to as Long Short-Term Memory (LSTM) networks belongs to the domain of artificial neural networks known as Recurrent Neural Networks (RNNs), which are highly effective for time series forecasting. LSTM is specifically designed to identify and remember long-term relationships in time series data (Al Musawi *et al*., 2023). This is achieved through the inclusion of a memory cell and three regulatory gates: The input, forget and output gates. These components are outlined in the source (Brownlee, 2017).

The LSTM model consists of three layers: An input layer, an LSTM layer and an output layer.

The input layer receives sequential data, which represent discrete time intervals and prepares it for processing by the LSTM layer.

The LSTM layer contains LSTM cells that store previous states and control gates to manage data flow.

Finally, the output layer interprets the processed signals from the LSTM layer and generates predictions for future values in the time series.

Training the LSTM algorithm is an elaborate process that uses Backpropagation Through Time (BPTT), a methodical technique that involves the meticulous calculation of gradients of the network's loss function concerning the variable weights within the neural architecture. Following this detailed computation, these weights undergo an adjustment process through the application of an optimization strategy, such as the common Stochastic Gradient Descent (SGD).

The training phase is critical for the LSTM algorithm because it is during this period that the network learns the optimal adjustment of weights to minimize the error between the predicted output and the actual observed values of the time series data.

Despite its robustness and efficacy in time series forecasting, it is important to acknowledge that the LSTM network demands intensive computational resources during training. In addition, achieving peak performance typically necessitates precise calibration of hyperparameters.

In our implementation, we employed an LSTM architecture with four Keras layers. The initial layer comprises 128 LSTM units, followed by a second layer with 64 LSTM units. The third layer comprised 25 dense units and the last layer featured a single dense output unit. This architecture was chosen to balance model complexity with computational efficiency while still allowing for the capture of long-term dependencies within the sales data. This structure showed promising results during the initial testing and was not so resource-intensive. There were 10 training epochs for 12 and 24-h time frames and 5 epochs for the rest to ensure adequate compute time. The model was trained with the Adam optimizer and mean squared error as the loss function.

Prophet

Prophet is a forecasting algorithm developed by Facebook. It is designed to handle various time series data with various characteristics, including nonlinear trends, seasonality and the impact of holidays. The algorithm adopts a decomposable model that breaks down the time series into three distinct components: Trend, seasonality and holidays.

The prophet algorithm consists of three main components:

1. Trend component: Models non-periodic changes in the time series data over time
2. Seasonality component: Models periodic changes in the time series data over time, such as daily, weekly, or yearly. Prophet can model multiple seasonality components simultaneously and can handle seasonality with changing frequency or amplitude
3. Holidays component: Models the influence of holidays on data

The prophet employs a Bayesian approach during training, using Markov Chain Monte Carlo (MCMC) methods for more accurate parameter estimation of forecasts. This allows the algorithm to account for uncertainties and provide probabilistic forecasts. During the training process, the prophet iteratively adjusts the model parameters to minimize the difference between the forecasted values and the actual time series data (Letham and Taylor, 2017). It was selected as one of the simplest realization forecasting methods. Prophet was implemented with its default settings due to its automated nature and focus on capturing trends, seasonality and holidays with minimal parameter tuning.

Materials

About the Dataset

The dataset used to compare the algorithms was provided by Tinkoff data and consists of anonymized relative sales in five categories: Pharmacies, railway tickets, books, sports goods and fuel. The data were recorded on an hourly basis from February 1, 2022, to January 31, 2023. To assess forecast quality, the data were aggregated into sets covering 4/8/12-h periods and 1 day. Tables 1-2 demonstrate the data structure.

Data Overview

The provided data are shown in Figs. 1-3. Each of the five categories exhibits a plummet in sales on January 1. This dip can be explained by the fact that New Year’s Day is the main holiday in Russia, which significantly reduces consumer activity. In the final days of December, there is a noticeable increase in the purchase of books, sports equipment and fuel. This can be explained by the fact that books are a traditional New Year’s gift, while sports equipment can also be used for this purpose. Pre-New Year purchases also increase the demand for fuel due to increased logistics volumes. Seasonal illnesses and people's desire to stock up on necessary supplies before the holidays also contribute to a slight increase in medicine sales. On the other hand, sales of train tickets declined as they are usually purchased in advance.

Table 1: Field types

Field	Field Type	Description
Hour	Date time (%Y-%m-%d %H)	Operation date and time (by hours)
Category	String	Category name
Cnt	float	Relative number of operations for that hour (from 0-1)
y	int	Numerical expression of relative sales, the value of 10000 is taken as 1, it is used for aggregation in other periods

Source: Developed by the author

Table 2: Data examples

2022-02-01 05	Pharmacies	0.0732817	0733
2022-02-01 06	Pharmacies	0.1284307	1284
2022-02-01 07	Pharmacies	0.1973727	1974
2022-02-01 08	Pharmacies	0.3881773	3882

Source: Developed by the author

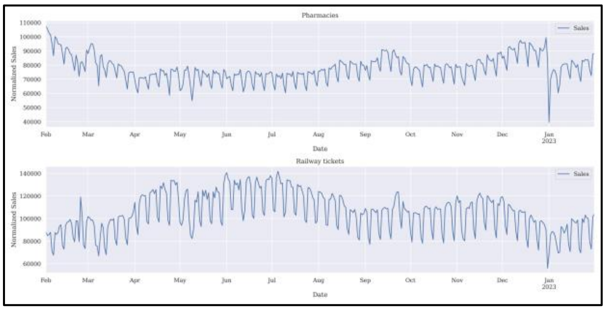

Fig. 1: Pharmacies and railway ticket sales (Source: Developed by the author)

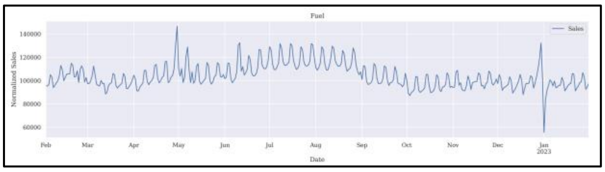

Fig. 2: Fuel sales (Source: Developed by the author)

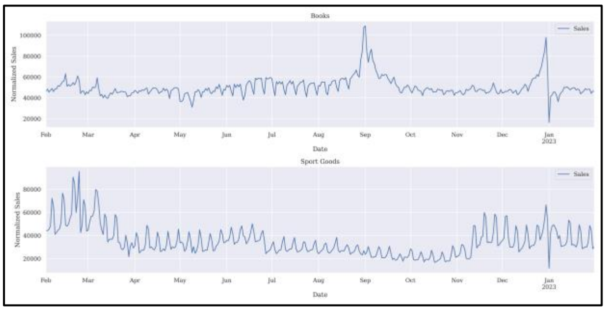

Fig. 3: Books and sports goods sales (Source: Developed by the author)

About Hardware

The forecasting analyses were conducted on a virtual machine equipped with 8 cores of an Intel Xeon Gold 6226R processor, 30GB of RAM and NVIDIA Quadro P6000 GPU.

Methods

About Metrics

The authors used three metrics to measure the precision of time series predictions: Mean Absolute Percentage Error (MAPE), Mean Absolute Error (MAE) and coefficient of determination, commonly referred to as R_2. These metrics are widely recognized and cited in the literature (Hyndman and Koehler, 2006) for their effectiveness in assessing the accuracy of forecasting models:

- *MAPE* is a popular metric for evaluating the accuracy of time series forecasts. It measures the average absolute percentage difference between the actual and forecasted values. *MAPE* is easily interpretable and provides a clear representation of the percentage error in the forecast. It can be used to compare forecasts involving diverse units of measurement or varying scales, making it a useful tool for comparison across different datasets

- *MAE* measures the average absolute difference between the actual and forecasted values of a time series. *MAE* is easy to interpret and provides an idea of the average magnitude of errors in the forecast. It was normalized to an hourly basis for time intervals larger than 1 h

Pseudocode

The implementation of the forecasting models and evaluation metrics used various Python libraries. Specifically, LSTM was implemented using Keras version 2.14 Prophet with fbprophet version 1.15, ARIMA with stats models version 0.14 and evaluation metrics with scikit-learn version 1.2.2.

To elucidate the methodology employed for our comparative analysis, we present a structured pseudocode outline below. The following pseudocode outlines the overall algorithm for data preprocessing, model training, prediction and evaluation. This schematic algorithm encapsulates our data preprocessing and modeling approach, which includes outlier removal, model training and performance evaluation across various categories and time frames. To determine the ARIMA parameters, each time frame was checked on every possible p value between 2 and 14, d value between 0 and 2 and q value between 0 and 4.

```
begin
 data = read_file('sales_data.csv') // Read the dataset from a file
  // Loop through each category in the dataset
 for the category in data.categories:
    // Loop through each specified time frame
    for time_frame in [1H, 4H, 8H, 12H, 1D]:
      // Preprocess data: Remove outliers from the current category and time frame
      processed_data = remove_outliers(data, category, time_frame)
      // Loop through each forecasting model
      for model_name in ['arima', 'lstm', 'prophet']:
        // Initialize the model based on the model name
        model = initialize_model(model_name)
        train(model, processed_data)
        // Make predictions with the trained model
        predictions = model.predict(processed_data)
        // Compute the metrics for the model's performance
        mape = compute_mape(processed_data.true_values, predictions)
        Mae = compute_mae(processed_data.true_values, predictions)
        r2 = compute_r2(processed_data.true_values, predictions)
        print(f"Category: {category}, Time Frame: {time_frame}, Model: {model_name}")
        print(f"MAPE: {mape}, MAE: {mae}, R2: {r2}")
      end
    end
  end
end
```

Results

The process of assessing forecast accuracy involved conducting predictions for each sales category using the chosen algorithms, spread out across different time intervals: 1, 4, 8, 12 and 24 h. To accomplish this, the data was partitioned, with 95% being used for training the models and the remaining 5% set aside for testing. The accuracy of these predictions, as determined by the computational models, is documented by the authors in Tables 3-4 for review and comparison.

An example of a highly accurate (MAPE is less than 5%) forecast is presented in Fig. 4. The blue line depicts the actual data used for model training, which accounts for 95% of the dataset. The red and orange lines represent the actual data for the remaining 5% of the dataset, along with the predicted values.

The plot in Fig. 5 illustrates the *MAPE* scores for each model across various sales categories.

The results demonstrate that ARIMA consistently exhibits superior performance as the forecasting horizon extends, particularly for daily predictions (Table 4). This aligns with the findings of Hyndman and Athanasopoulos (2018), who highlight ARIMA's ability to effectively capture trends and seasonality over longer periods. However, our study further reveals that LSTM outperforms ARIMA in hourly forecasts for certain categories, such as fuel stations (Fig. 4). This suggests that LSTM's capability to learn long-term dependencies might be advantageous when dealing with short-term fluctuations and potentially less pronounced seasonal patterns. Future research could explore hybrid models that combine the strengths of ARIMA and LSTM to achieve robust and accurate forecasts across various time frames. Figure 5 provides a visual representation of the MAPE for each forecasting model across the different sales categories when using a one-day forecasting horizon. The results reveal interesting insights into the models' performance variations depending on the specific category.

Table 3: Results for 1/4/8 timeframes

ARIMA

	8H			4H			1H		
	MAPE	MAE	R^2	MAPE	MAE	R^2	MAPE	MAE	R^2
Pharmacies	055.74	585.93	0.904	014.08	295.09	0.965	008.93	240.06	0.973
Railway tickets	043.48	859.97	0.736	025.12	141.79	0.858	015.04	438.64	0.889
Books	037.04	342.34	0.911	013.16	726.01	0.966	013.05	110.02	0.986
Sport goods	147.02	620.03	0.683	044.31	381.07	0.789	016.16	299.06	0.081
Fuel stations	043.52	924.01	0.664	013.13	414.07	0.908	007.21	279.04	0.944

LSTM

	1H			4H			8H		
	MAPE	MAE	R^2	MAPE	MAE	R^2	MAPE	MAE	R^2
Pharmacies	026.02	206.05	0.994	022.08	387.05	0.958	019.09	366.0	0.965
Railway tickets	009.06	229.03	0.981	013.02	412.02	0.943	020.00	638.4	0.953
Books	014.01	133.00	0.988	018.02	174.01	0.098	014.05	280.7	0.983
Sport goods	020.06	097.08	0.972	076.06	308.07	0.906	019.02	198.1	0.897
Fuel stations	004.06	154.03	0.974	016.06	482.02	0.947	007.07	228.7	0.971

Prophet

	1H			4H			8H		
	MAPE	MAE	R^2	MAPE	MAE	R^2	MAPE	MAE	R^2
Pharmacies	049.02	408.0	0.966	037.07	360.03	0.974	019.01	227.3	0.981
Railway tickets	031.07	552.4	0.875	026.00	474.08	0.899	021.00	415.2	0.908
Books	028.04	310.9	0.941	023.00	274.03	0.951	018.03	242.4	0.094
Sport goods	235.09	558.2	0.819	217.01	579.01	0.791	114.01	494.1	0.764
Fuel stations	011.09	347.8	0.949	007.09	258.06	0.964	006.09	184.1	0.978

Source: Developed by the author

Table 4: Results for 12/24 timeframes

ARIMA

	12H			1D		
	MAPE	MAE	R^2	MAPE	MAE	R^2
Pharmacies	4.2	145.9	0.985	3.2	108.04	0.664
Railway tickets	10.2	355.8	0.898	5.3	180.72	0.788
Books	7.2	131.6	0.97	3.6	69.03	0.708
Sports goods	14.2	252.2	0.848	13.6	250.01	0.037
Fuel stations	4.3	156.1	0.959	2.9	120.07	0.531

LSTM

	12H			1D		
	MAPE	MAE	R^2	MAPE	MAE	R^2
Pharmacies	6.2	165.4	0.971	6.6	225.0	0.239
Railway tickets	8.3	238.0	0.828	9.7	324.2	0.45
Books	9.3	138.2	0.968	12.8	253.4	0.152
Sports goods	12.8	156.8	0.914	20.8	294.0	0.294
Fuel stations	4.7	166.1	0.931	3.2	132.4	0.306

Prophet

	12H			1D		
	MAPE	MAE	R^2	MAPE	MAE	R^2
Pharmacies	8.5	191.1	0.981	6.3	205.2	0.074
Railway tickets	11.1	336.0	0.888	3.6	140.7	0.882
Books	12.9	211.8	0.094	2.5	48.8	0.144
Sport goods	45.1	447.0	0.747	24.2	328.5	0.019
Fuel stations	5.3	179.3	0.095	2.2	89.1	0.707

Source: Developed by the author

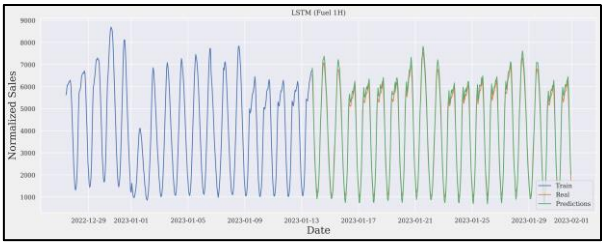

Fig. 4: LSTM forecast for fuel stations (1H) (Source: Developed by the author)

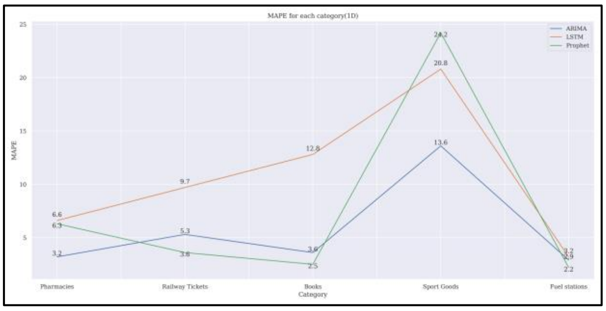

Fig. 5: Dependence of MAPE on model and category (source: Developed by the author)

ARIMA demonstrates consistent performance across all categories, maintaining a relatively low MAPE, indicating its ability to effectively capture the underlying patterns and trends in diverse sales contexts.

LSTM exhibits greater variability in performance. While it achieves a low MAPE for categories like fuel stations and railway tickets, the error increases notably for categories like sports goods. This suggests that the model's effectiveness might be influenced by the specific characteristics of the sales data, like higher volatility or less pronounced seasonal patterns.

Prophet shows a similar trend to LSTM, performing well in certain categories like books and fuel stations but encountering higher MAPE values for sports goods.

These observations highlight the importance of considering the specific sales domain and data characteristics when selecting a forecasting model.

Conclusion

Assessment of ARIMA, LSTM and prophet models in time series sales forecasting reveals a range of informative results. Firstly, it is evident that all three models demonstrate a remarkable ability to produce accurate forecasts across sales categories. This ability holds great importance as businesses strive to predict fluctuations in demand and optimize their resource allocation strategies.

However, the sports category is a notable exception here. This contrast may be related to the range of organizations represented by sports Merchant Category Codes (MCCs), which can pose a challenge for accurate predictions if an MCC does not adequately capture the underlying business characteristics.

When focusing on forecasts with hourly intervals, a decline in performance appears, which likely results from the significant fluctuations in sales that can occur within a single day. In the case of hourly sales predictions, the models may face significant difficulties due to the high variability of sales at shorter time frames. However, the LSTM model, which can capture long-term dependencies, proved to be a reliable performer in this context. However, the LSTM model showed inconsistent results in repeated runs, with better performance sometimes and poor performance in others. In certain instances, the LSTM method demonstrated superior performance compared with the other two approaches during the initial attempt. However, the results were poorer in subsequent runs. Therefore, an average of two runs was calculated.

As the forecast horizon extends, ARIMA consistently shows better performance, particularly for daily forecasts. This reflects the ability of ARIMA to effectively model trends and patterns over longer periods. Beginning at the 4-h interval, ARIMA outperformed the other models in several categories, with the most significant improvement observed in daily predictions.

While all models exhibited strong performance in certain scenarios, our comparative analysis revealed distinct advantages depending on the forecasting context. ARIMA proved particularly effective for longer forecasting horizons, especially in daily predictions, aligning with its strength in capturing trends and seasonality. LSTM, on the other hand, demonstrated potential for accurate hourly predictions, particularly within categories like fuel sales, where its ability to learn long-term dependencies becomes beneficial. Prophet, with its ease of use and ability to incorporate holidays and other events, also presents a valuable tool, particularly for businesses seeking a rapidly implementable solution.

The study's multiple findings not only provide a nuanced understanding of the comparative strengths and weaknesses of ARIMA, LSTM and prophet in forecasting sales but also highlight the critical role of choosing the right time frames and careful model selection depending on the forecasting area. Such insights are anticipated to assist businesses in making data-driven decisions and optimizing sales forecasting operations.

There are still possibilities for future research. One promising direction involves exploring hybrid models that combine the strengths of ARIMA, LSTM and Prophet. For instance, a hybrid ARIMA-LSTM model could leverage ARIMA's ability to capture long-term trends and seasonality while using LSTM's capability to learn complex temporal dependencies from short-term data. Furthermore, exploring alternative deep learning architectures, such as convolutional neural networks or attention-based models, might reveal new approaches for effectively capturing the unique characteristics of data acquisition.

Acknowledgment

I am grateful to Andrey Ring (1C Company) for inspiring this research and providing valuable feedback throughout the writing process. I extend my sincere thanks to the TCS group for generously providing the acquired dataset that made this research possible.

The dataset used in this study was deposited into Harvard dataverse at https://doi.org/10.7910/DVN/CLDQCQ.

Funding Information

The authors have not received any financial support or funding to report.

Ethics

This study is an original research effort with no ethical concerns raised.

References

Al Musawi, S. S., Hamodat, Z. M., & Basheer, A. G. (2023). Forecasting of Solar Radiation for Solar System Under Different Climatic Condition. *International Journal on "Technical and Physical Problems of Engineering" (IJTPE)*, *15*(57), 270-277. http://www.iotpe.tabaelm.com/IJTPE/IJTPE-2023/IJTPE-Issue57-Vol15-No4-Dec2023/34-IJTPE-Issue57-Vol15-No4-Dec2023-pp270-277.pdf

Bennett, K. P., & Campbell, C. (2000). Support vector machines: Hype or hallelujah? *ACM SIGKDD Explorations Newsletter*, *2*(2), 1-13. https://doi.org/10.1145/380995.380999

Brownlee, J. (2017). A gentle introduction to long short-term memory networks by the experts. *Machine Learning Mastery*, *19*. https://machinelearningmastery.com/gentle-introduction-long-short-term-memory-networks-experts/ (Accessed on December 15, 2023).

Chomboon, K., Chujai, P., Teerarassamee, P., Kerdprasop, K., & Kerdprasop, N. (2015, March). An empirical study of distance metrics for k-nearest neighbor algorithm. In *Proceedings of the 3rd International Conference on Industrial Application Engineering* (Vol. *2*, p. 4). https://pdfs.semanticscholar.org/815f/22363962afda432435cdbb478857547a17b5.pdf

Hachcham, A. (2023). The KNN Algorithm–Explanation, Opportunities, Limitations. *neptune. ai*. https://neptune.ai/blog/knn-algorithm-explanation-opportunities-limitations

Hamayel, M. J., & Owda, A. Y. (2021). A novel cryptocurrency price prediction model using GRU, LSTM and bi-LSTM machine learning algorithms. *Ai*, *2*(4), 477-496. https://doi.org/10.3390/ai2040030

Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: Principles and Practice, 1st Ed.* pp. 380, OTexts. ISBN-10: 9780987507112.

Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, *22*(4), 679-688. https://doi.org/10.1016/j.ijforecast.2006.03.001

Idrees, S. M., Alam, M. A., & Agarwal, P. (2019). A prediction approach for stock market volatility based on time series data. *IEEE Access*, *7*, 17287-17298. https://doi.org/10.1109/ACCESS.2019.2895252

Kim, K. S., Lee, J. B., Roh, M. I., Han, K. M., & Lee, G. H. (2020). Prediction of ocean weather based on denoising autoencoder and convolutional LSTM. *Journal of Marine Science and Engineering*, *8*(10), 805. https://doi.org/10.3390/jmse8100805

Lakshminarayanan, S. K., & McCrae, J. P. (2019, December). A Comparative Study of SVM and LSTM Deep Learning Algorithms for Stock Market Prediction. In *AICS* (pp. 446-457). https://ceur-ws.org/Vol-2563/aics_41.pdf

Letham, B., & Taylor, S. J. (2017). Prophet: forecasting at scale. In *Meta Research*. https://research.facebook.com/blog/2017/2/prophet-forecasting-at

Lisova, N. (2021). Keynesian and Monetarist Approaches to Regulation of the Labor Market in the Transition Economies of EECCA Countries. *ELIT–Economic Laboratory for Transition Research Dz. Washingtona 4/5*, *17*(4), 75. https://doi.org/10.14254/1800-5845/2021.17-4.7

Nepal, B., Yamaha, M., Yokoe, A., & Yamaji, T. (2020). Electricity load forecasting using clustering and ARIMA model for energy management in buildings. *Japan Architectural Review*, *3*(1), 62-76. https://doi.org/10.1002/2475-8876.12135

Raban, D. R., & Gordon, A. (2020). The evolution of data science and big data research: A bibliometric analysis. *Scientometrics*, *122*(3), 1563-1581. https://doi.org/10.1007/s11192-020-03371-2

Sabyrzhan, A., Andarova, R., Kanatbek, A., & Lisova, N. (2021). Keynesian and monetarist approaches to regulation of the labor market in the transition economies of EECCA Countries. *Monten J Econ, 17*(4), 75–83. https://doi.org/10.14254/1800-5845/2021.17-4.7

DOI: 10.3844/jcssp.2024.1222.1230

Subashini, A., Sandhiya, K., Saranya, S., & Harsha, U. (2019). Forecasting website traffic using prophet time series model. *Int Res J Multidiscip Technovation*, *1*(1), 56-63. https://doi.org/10.34256/irjmt1917

Tripathi, S. (2023). Can stock prices be predicted? Comparative study of LSTM and SVR for financial market forecast. *Authorea Preprints*. https://doi.org/10.36227/techrxiv.21158932.v2

Vyas, M. & Hemrajani, N. (2021). Predicting effort of agile software projects using linear regression, ridge regression and logistic regression. *Int J Techn Phys Probl Eng* (*IJTPE*), *13*(2), 14-19. http://www.iotpe.com/IJTPE/IJTPE-2021/IJTPE-Issue47-Vol13-No2-Jun2021/3-IJTPE-Issue47-Vol13-No2-Jun2021-pp14-19.pdf