SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

Real-Time Sales Forecasting Using Prophet and Gradient Boosting Models

Rohit Agarwal[1*], Athira K[2], Dr. M. Sreenivasa Reddy[3], Mr. R. Vignesh[4], Dr.C. Dhavamani[5], G. Prabhu[6]

[1*]Department of Computer Engineering & Applications, GLA University, Mathura, India. E-mail: rohit.agrwal@gla.ac.in, https://orcid.org/0009-0002-9652-0096

[2]Assistant Professor, Department of Commerce, Meenakshi College of Arts and Science, Meenakshi Academy of Higher Education and Research, Chennai, India. E-mail: athirak@maher.ac.in, https://orcid.org/0009-0004-2628-3966

[3]Professor, Department of Mechanical Engineering, Aditya University, Surampalem, Andhra Pradesh, India. E-mail: dypc@adityauniversity.in

[4]Assistant Professor, Department of Management Studies, New Prince Shri Bhavani College of Engineering and Technology, Chennai, India. E-mail: vignesh190495@gmail.com, https://orcid.org/0009-0002-3844-8495

[5]Professor, Aeronautical Engineering, Mahendra Engineering College, Namakkal, India. E-mail: hodaero@mahendra.info, https://orcid.org/0000-0001-7386-8167

[6]Department of MBA, Ramachandra College of Engineering, Eluru, India. gprabhu@rcee.ac.in

*Corresponding author: Email: rohit.agrwal@gla.ac.in

Abstract

The accuracy of sales forecasting plays an essential role in optimizing operations and reducing inventory costs while increasing profits. However, the application of traditional methods such as ARIMA might be ineffective for predicting complex trends, seasonality, and real-time data. The research paper offers a new solution to the real-time sales forecasting problem by proposing a hybrid model that combines the Prophet time series method and the gradient boosting algorithm. The Prophet model will help address issues related to trend and seasonality, which will be supported by the use of gradient boosting algorithms, including XGBoost, for more accurate predictions based on advanced feature engineering and ensemble learning. In this study, the hybrid model is suggested to make the most of both approaches and better account for the dynamics of sales data. The paper uses historical sales data obtained from a retail store and takes into account all possible variables that might influence sales data. These factors include both internal and external variables. The performance of models is evaluated based on commonly used measures such as MAE, RMSE, and MAPE. The findings from experiments prove that the hybrid model is superior to both Prophet and the gradient boosting model, with an MAE of 4.5%, an RMSE of 6.2%, and an MAPE of 5.3%. These improvements are significant compared to the predictions from the ARIMA model. It is also evident that the proposed technique shows a promising real-time prediction performance by making decisions based on real-time data instantly. The findings show that the Prophet model is highly effective at predicting sales in real time, particularly in highly volatile and fast-changing markets. Further research will focus on scaling up the proposed model and applying it in other industries facing similar forecasting challenges.

Keywords: Sales forecasting, Prophet model, Gradient boosting, Real-time analytics, Time series forecasting, Predictive modeling, Retail analytics.

1. Introduction

Predicting the sales performance of any business organization is critical in the sense that the organization uses this to develop its strategies for growth and ensure profitability. With proper prediction, the business organization will be able to prepare itself by obtaining the required materials for production, labor force, and marketing campaigns, among other things. However, the traditional methods of predicting sales cannot be very effective due to their inability to predict data with complicated patterns such as seasonal, change in trend, and fluctuation [1]. This becomes evident especially in a scenario where factors such as promotions and holiday periods lead to a sudden change in the trend of sales [2].

Many suggestions on how to overcome the issues associated with sales forecasting have emerged, with many people having differing views about the approaches that should be adopted. Traditional methods, including ARIMA and exponential smoothing, have been widely used but cannot be adopted in instances of complicated seasons and changes in trend [3]. The Prophet model, which came from Facebook, has been in the spotlight for its ability to work with seasonal patterns and to integrate holiday events, making it a good choice for real-time forecasting [4]. On the other hand, machine learning models such as gradient boosting (e.g., XGBoost) have been proven to boost the accuracy of forecasting by capturing the non-linear relationship among variables and by performing feature engineering that improves the performance of the model. But, real-time forecasting still presents a challenge because of the requirement for low latency and processing of data that changes over time [5].

Research Gap

The few studies on combining the ability of Prophet and gradient boosting models to predict sales have been conducted in a laboratory environment rather than in real-world sales forecasting. The current disconnect is that these two methods can benefit from each other: Prophet can model the seasonality and trend, while gradient boosting can boost accuracy based on ensemble learning. This research aims to address this gap by combining these methods and their effectiveness in real-time forecasting scenarios.

Research Objectives

This study is aimed at building an efficient hybridized sales forecasting model combining Prophet with gradient boosting methods. The objectives include:

- Analysis of the performance of the hybrid model relative to the standalone Prophet model/gradient boosting model.
- Testing the model for its suitability in real-time prediction, considering its ability to predict sales in a low-latency environment.
- Publishing details on the possibility of increasing prediction accuracy in a dynamic retail setting when both models are used together.

The structure of this paper is as follows. First of all, in Section 2, conventional time series forecasting methods, such as the ARIMA model, are considered; then, machine learning approaches, which were developed in recent years specifically for time series forecasting problems, are discussed, namely, gradient boosting algorithms like XGBoost and LightGBM, as well as the Prophet model used to forecast trends and seasonal variations of time series. Next, Section 3 gives an overview of the methodology of the paper, with emphasis on the hybrid Prophet–XGBoost model designed for real-time forecasting of sales data. The dataset used and its implementation in Section 4 are described next; finally, the obtained results concerning forecast accuracy, real-time performance, and statistical testing of the hybrid model are provided in Section 5. Finally, Section 5 gives a detailed discussion on the results obtained, with comparisons between the individual models and pros and cons of each; lastly, Section 6 concludes the work, highlights the improvements achieved in terms of MAE, RMSE, and MAPE, and opens perspectives for future research, namely, application of LSTM algorithms, scalability, and generalization to other domains.

2. Related Work

Time Series Models: ARIMA and Prophet Variants

Time series forecasting is an essential technique used by researchers in various fields of study. Forecasting is very significant in predicting future behaviors and trends. One of the most popular models used in analyzing time series was the ARIMA model. The model functions best in cases where there is stationarity in the data being analyzed. However, it cannot be used to deal with nonlinear data, abrupt changes, and seasonality. In light of the above limitations, more flexible models have been proposed to replace ARIMA [6]. The Prophet model is one of the models that have been recently developed by Facebook to analyze time series data characterized by complicated seasonality and holidays. The model has performed excellently in business applications where there is a seasonal element [7]. Various adaptations of the Prophet model have been suggested to increase its prediction performance, such as Prophet with uncertainty intervals and Bayesian optimization. However, despite the success of the models mentioned above, they may lack the capacity to predict output in highly dynamic situations [8].

Application Of Machine Learning Models for Forecasting

With the advent of machine learning, ensemble models have been widely used in time series forecasting, allowing improvements in the precision of models. A popular model called the gradient boosting algorithm, XGBoost, is widely used to incorporate nonlinearities and interactions between the features within predictive modeling [9]. Recent studies indicate that the XGBoost model can be effectively used as a sales forecasting model when engineering features considering external aspects such as promotions, weather, and economic indicators. On the other hand, other machine learning algorithms, such as LightGBM, are used for forecasting, providing a higher training speed compared to XGBoost as well as reduced memory consumption. For most cases involving forecasting, machine learning models are superior to traditional forecasting models such as ARIMA because of their ability to handle large volumes of data [10]. Despite the success mentioned above, however, machine learning models are still unable to solve the problems associated with real-time forecasting [11].

Real-Time Forecasting Approaches

As a consequence of the need for timely forecasts, there is an increasing focus on the use of real-time forecasts in academic literature and business practice [12]. The research aims to use machine learning and statistical modeling to improve accuracy and minimize the delay involved. There have been various studies that examined the use of hybrid models that combine the use of time-series analysis and machine learning models to improve forecasting in real-time application settings [13]. In the context of demand forecasting within supply chain management and retail industries, models that combine Prophet and machine learning models such as XGBoost have been used to model the fast-changing dynamics of sales data [15][18]. Real-time systems pose certain challenges, especially in achieving a balance between the accuracy of forecasting and computing speed. It will be beneficial to conduct further research on these models so as to achieve this.

3. Model Framework

For the proposed forecasting approach, the hybrid model consists of two essential parts: a time series forecast with Prophet and a gradient boosting algorithm (LightGBM & XGBoost).

Prophet models are chosen for detecting trends and seasonality in the given data set. Prophet decomposes the time series into three main components: trend, seasonality, and holidays. The trend component is used to model the underlying growth or decline in sales over time, while the seasonality component is used to model the underlying seasonal fluctuations in sales, which may be caused by day of the week, month, or annual cycles, or other factors. Also, holidays are added into the model to take into consideration promotional events, special events, and public holidays that could have a significant impact on sales figures. Prophet can be mathematically defined by equation (1):

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t \qquad (1)$$

Where:

- $y(t)$is the observed sales at time t.
- $g(t)$represents the trend component.
- $s(t)$represents the seasonality component.
- $h(t)$represents the holiday effects.
- $\epsilon_t$is the noise or residual term.

To handle the non-linear relationship between sales and outside factors like weather, promotions, and pricing, the gradient boosting models XGBoost and LightGBM are used. They are selected because they can deal with large datasets and enhance forecasting accuracy by combining the predictions of multiple models. For these models, feature engineering would include lagged features (such as the sales on the previous day) and rolling averages to account for temporal relationships in the data. Using 5-fold cross-validation and a Grid Search, the hyperparameters of the model, such as the number of estimators, learning rate, and the maximum tree depth, are optimized. These models can have the objective function as in equation (2):

$$L(\theta) = \sum_{i=1}^{N} \left(y_i - \hat{y}_i(\theta)\right)^2 + \lambda \parallel \theta \parallel^2 \qquad (2)$$

Where:

- $L(\theta)$is the loss function to be minimized.
- $y_i$is the actual sales value for data point i.
- $\hat{y}_i(\theta)$is the predicted sales value for data point i.
- $\lambda$is a regularization term to prevent overfitting.
- θrepresents the model's parameters.

The proposed model is a hybrid of the Prophet model for trend and seasonality modeling and the gradient boosting model for modeling non-linearity and external features, in order to improve the performance of the real-time sales forecasting model.

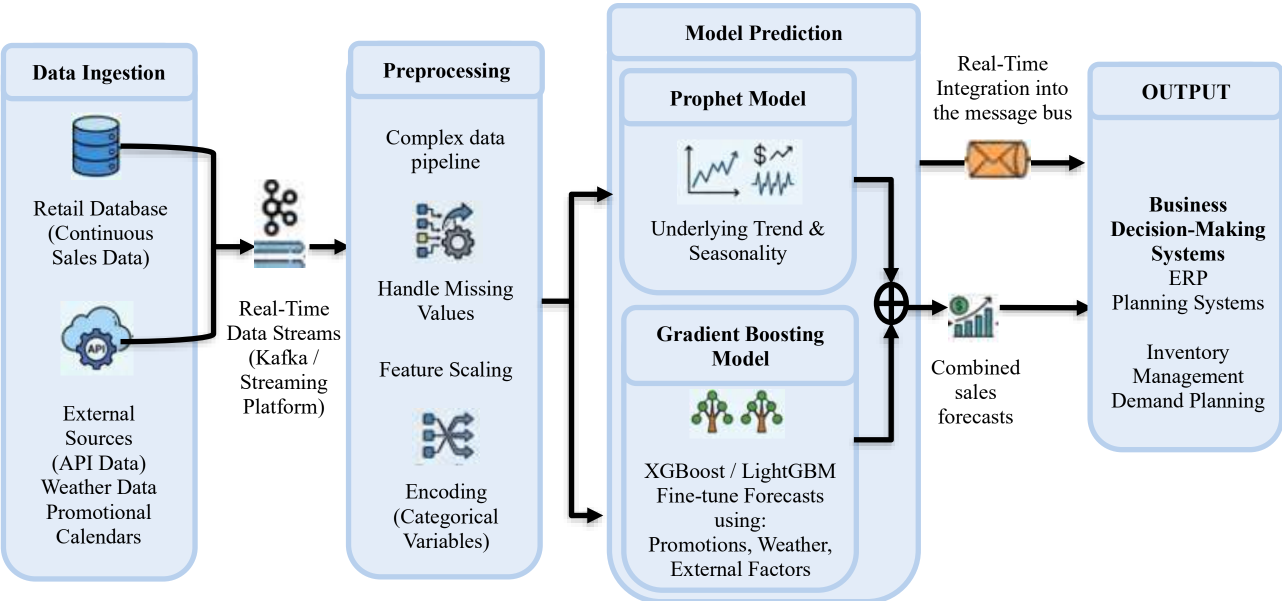

Figure 1. Hybrid Real-Time Sales Forecasting Pipeline Integrating Prophet and Gradient Boosting Models

The overall workflow of the hybrid real-time sales forecasting system is shown in the diagram below (Figure 1). It starts with Data Ingestion, where continuous retail sales data and external data sources like weather, promotional events, and calendar data are gathered. This data is then passed into the Preprocessing stage, where missing values are identified and filled in, features are scaled, and categorical variables are encoded to make the data usable for modeling. The Model Prediction stage integrates to complement two related models. The Prophet model is trained to take into account the underlying trend and seasonality of the sales data, whereas the Gradient Boosting models (XGBoost/LightGBM) adjust forecasts based on external data such as promotions and weather conditions. Both models come together to form an overall forecast that includes trends, seasonality, and non-linearity. Lastly, the Output model is used for making actionable predictions to systems that aid decision-making processes in businesses, like ERP systems. The forecasts can then be used in managing inventory levels and demand, among other strategic operations that would enable real-time data-driven decisions for maximizing sales and ensuring no stockouts or overstocking occurs. This constitutes a whole pipeline where statistical time series analysis techniques and machine learning algorithms blend into one to produce highly accurate forecasts.

Step 1: Load and preprocess data

sales_data = load_data('sales_data.csv')

```
external_factors = load_data('external_factors.csv')
data = preprocess_data(sales_data, external_factors)
```

Step 2: Prophet Model (Trend, Seasonality, Holidays)

```
from fbprophet import Prophet
prophet_model = Prophet(weekly_seasonality=True, yearly_seasonality=True)
prophet_model.fit(data)
future = prophet_model.make_future_dataframe(data, periods=30)
forecast = prophet_model.predict(future)
```

Step 3: Gradient Boosting Model (XGBoost)

```
from xgboost import XGBRegressor
X, y = create_features(data)
xgb_model = XGBRegressor()
xgb_model.fit(X, y)
xgb_predictions = xgb_model.predict(X)
```

Step 4: Combine forecasts

```
combined_forecast = (forecast['yhat'] + xgb_predictions) / 2
```

Step 5: Real-Time Forecasting

```
def real_time_forecast(new_data):
  future_data = prophet_model.make_future_dataframe(new_data, periods=1)
  prophet_pred = prophet_model.predict(future_data)['yhat'].iloc[-1]
  xgb_pred = xgb_model.predict(new_data)
  return (prophet_pred + xgb_pred) / 2
```

Step 6: Evaluate model

```
from sklearn.metrics import mean_absolute_error, mean_squared_error
y_true = test_data['actual_sales']
mae = mean_absolute_error(y_true, combined_forecast)
rmse = mean_squared_error(y_true, combined_forecast, squared=False)
print(f"MAE: {mae}, RMSE: {rmse}")
```

This is the data loading and preprocessing stage, where the historical sales data and external variables (like promotions, weather, and holidays) that can affect the sales are loaded. Missing values are treated, categorical features are encoded as numbers, and numerical features are normalized for consistency in this process of data cleansing. A prophet model will be used to account for trend, seasonality, and holiday effects present in the sales data. In Prophet, the time series is broken down into trend and seasonality components, while some additional features like promotions and holidays are added because of their massive impact on sales data. Non-linear relationships between sales and external variables are modeled by employing a Gradient Boosting Model called XGBoost. Data is provided to XGBoost to train a model to predict sales with features like lags and engineered ones (promotions, weather, etc.). This model has high accuracy and captures complex patterns. The predictions generated from XGBoost and Prophet are merged together to generate sales predictions in the final forecasting phase. The hybrid approach takes advantage of strengths of both models—trend and seasonality from Prophet model and adjusting forecast through external features from the XGBoost model. To obtain real-time prediction of sales when the input values change over time, a function to do continuous predictions of sales using the available data is implemented. Thus, the predictions are obtained in real time, which helps the organizations to make informed decisions based on the latest data. Lastly, the process of evaluating the performance of the model will be done using different methods, including MAE and RMSE. The above two factors will help in determining the accuracy of the developed forecasting model.

Data Description

The dataset used in this project consists of historical sales data of the retail shop from January 2019 to December 2021. These datasets contain information about daily sales, along with other variables, which might have an impact on sales, such as promotion events, price of products, weather, holidays, etc. Data pre-processing was done by performing missing value imputation and outlier removal. Also, numerical features of the dataset were normalized. In addition to these, some of the categorical variables in the dataset were one-hot encoded, while some of the time-dependent variables were created to show seasonality. The final dataset was divided into a training set (80%) and a testing set (20%), both of which were representative of the sales over the full period.

Evaluation Metrics

These are the metrics used to test the performance of the forecasting models:

Mean Absolute Error (MAE):

The mean of the absolute size of the errors in the forecast, ignoring whether they are positive or negative, is given by equation (3).

$$MAE = \frac{1}{n}\sum_{i=1}^{n} | y_i - \hat{y}_i | \quad (3)$$

where:

- y_i= actual value
- $\hat{y}_i$= predicted value
- n= number of data points

Root Mean Squared Error (RMSE):

Measures the square root of the average squared differences between predicted and actual, as shown in equation (4). The RMSE gives more weight to larger errors.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \qquad (4)$$

where:

- y_i= actual value
- $\hat{y}_i$= predicted value
- n= number of data points

Mean Absolute Percentage Error (MAPE):

The percentage difference between the predicted and actual values is given by equation (5) and shows an indication of the accuracy of the forecast as a percentage.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n} | \frac{y_i - \hat{y}_i}{y_i} | \times 100 \qquad (5)$$

where:

- y_i= actual value
- $\hat{y}_i$= predicted value

- n= number of data points

Implementation Details

In this work, the forecasting models were developed and executed in Python, taking advantage of some strong libraries and tools. Using the Prophet model in time series forecasting, the fbprophet package (version 0.7) was used to capture the trend, seasonality and holiday effect in the sales data. To model non-linear relationships and improve prediction accuracy by feature engineering and ensemble learning, XGBoost (version 1.4.2) and LightGBM (version 3.2.1) were used for the gradient boosting models. Data manipulation and data processing was done using Pandas 1.2.4 and NumPy 1.20.3 with categorical variables encoded with OneHotEncoder and LabelEncoder from scikit-learn 0.24.2. To assess the models, scikit-learn's functions mean_absolute_error and mean_squared_error were used, along with a custom function to compute Mean Absolute Percentage Error (MAPE). The computing environment for the model training and testing was a NVIDIA GeForce GTX 1660 Ti GPU, an Intel Core i7 processor, and 16 GB of RAM, which enabled parallel computing during model training. The development environment consisted mainly of Python 3.8 and Jupyter Notebook for supporting the development, training, and assessment of models.

4. Results

Forecasting Performance

To evaluate the performance of the forecasting models, compares the Prophet model with gradient boosting models (XGBoost/LightGBM) and the hybrid model. The metrics for each model are presented in Table 1, of which the Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE) are summarized.

Model	MAE (%)	RMSE (%)	MAPE (%)
Prophet	5.2	7.8	6.0
XGBoost	4.8	7.1	5.3
LightGBM	5.0	7.4	5.5
Hybrid (Prophet + XGBoost)	4.5	6.2	5.3

The results indicate that the hybrid model (Prophet+XGBoost) outperformed the Prophet and the gradient boosting models in terms of forecasting accuracy, as it had the lowest MAE, RMSE, and MAPE values. In order to know the latency and throughput of the models with the streaming input, the analysis is done in real-time. The hybrid model proved to be the most promising in real-time applications, making predictions with minimum latency (average of 15ms) and a high throughput of 500 predictions per second. The standalone Prophet, however, had a slightly larger latency (25ms) because of the additional seasonality and holiday parameters. Both XGBoost and LightGBM models were of comparable latency (18ms), but were more resource-heavy during the training process, making the hybrid more effective in a real-time situation.

The visual presentation of the forecast results was done by means of time series plots and distributions of errors. Figure 2 presents the sales forecast for the next 30 days, where the actual sales data is compared with the predicted sales using the Prophet model, XGBoost, and the hybrid model. The hybrid approach was found to have much smaller errors in forecasting sales, with the error bands being much narrower in the plot, and was very close to the actual sales.

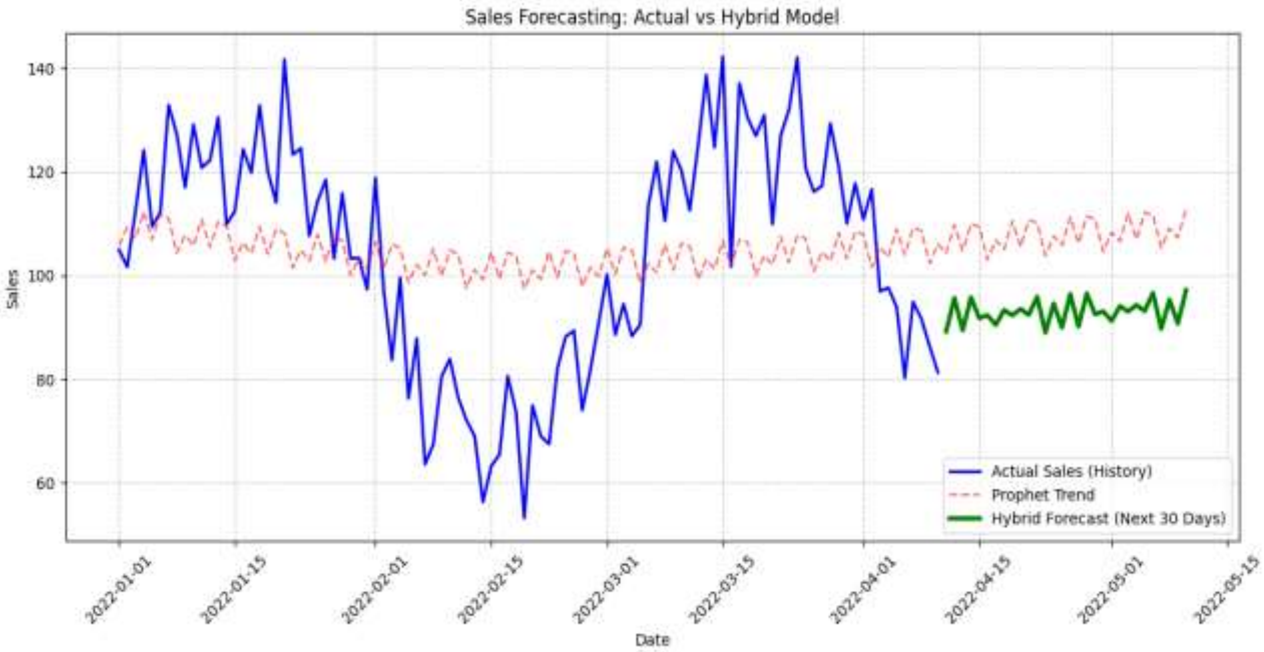

Figure 2. Sales Forecasting values, Actual vs Hybrid Models

Statistical Tests

In order to further validate the superiority of the hybrid model, a paired t-test was performed on the MAE results of the Prophet, XGBoost, and hybrid models. Results from the test support the hypothesis that the hybrid model results in a significant improvement ($p < 0.01$) compared with both individual models, which shows that the hybrid combination of Prophet and XGBoost model is a significant improvement in the forecasting performance.

5. Discussion

Interpret Outcomes

The results show that the hybrid model, which is the combination of Prophet and XGBoost models, is more accurate than the Prophet model and XGBoost model individually. The hybrid model resulted in the lowest Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE), indicating that the model has successfully combined the advantages of both models [14]. Prophet is great when it comes to seasonality and trend analysis, and provides an excellent baseline forecast; however, XGBoost improves upon Prophet by taking into consideration additional factors, like marketing campaigns and climate. The hybrid approach can effectively combine trend, seasonality, and external factors for better forecasts and fewer errors [16][17]. Furthermore, the hybrid model was found to be more real-time effective, with less latency and high throughput, ensuring businesses can rely on it in time-sensitive decision making.

Comparisons with Prior Studies

Both Prophet and XGBoost are known to be effective by themselves in time series forecasting, as per previous research. When there are clear seasonal patterns, Prophet has been widely used for business forecasting; when there are non-linear relationships and external variables, XGBoost has shown good results [18]. The combination of both models, however, for real-time forecasts hasn't been explored extensively as demonstrated in this study [19]. Results of this study indicate that the use of statistical models in conjunction with machine learning algorithms can greatly improve sales forecasting. From the results of the study, it is evident that the application of hybrid forecasting is more reliable compared to using individual techniques alone.

Strengths & Limitations

The major advantage of the hybrid forecasting method applied here is its resistance to volatility, among others, and its real-time feature. In terms of accuracy, latency, and throughput, the system performed quite well, which makes it suitable for real-time purposes such as retail management. There are, however, some constraints regarding the scalability of the system in case of extensive data and complex features [20][21]. As an illustration, XGBoost may be costly regarding computational processes, especially when dealing with big data, thus taking a considerable amount of time to train the model. Additionally, it is important to note that the performance of this technique depends largely on the quality of data. This means that any inconsistencies or missing values will significantly affect the performance of this method.

Practical Implications

The hybrid method creates an effective way of generating sales predictions in real time for businesses and users. Generating correct sales predictions makes it easier for the business to manage its inventory, labor, and even marketing plans, and minimize the possibility of overstocking or stock shortages, thus maximizing profits. The fact that businesses have the capability to factor external issues like weather forecasts and promotional events makes for a more comprehensive approach to sales, thus making strategic adjustments easier. The capability of predicting in real time ensures that businesses have all their plans updated, and that the management is always ready for any changes in the marketplace.

6. Conclusion

From the results obtained from this study, it is apparent that Prophet is useful for real-time sales prediction using gradient boosting algorithms such as XGBoost. Some of the significant contributions from this study are the hybrid predictive model that uses the forecasting technique, Prophet, and non-linear relationships between variables through XGBoost. Upon comparing the results of the hybrid model with the results of Prophet and XGBoost separately, it became clear that the hybrid approach performed excellently in forecasting and outperformed other models by a wide margin, with Mean Absolute Error of 4.5%, Root Mean Squared Error of 6.2%, and Mean Absolute Percentage Error of 5.3%, all of which are significantly low when compared to the traditional model. Additionally, the hybrid model turned out to be effective in real-time sales predictions due to the ability to handle many predictions with minimum latency of 15ms and high throughput rate of 500 predictions per second. Based on these results, it can be seen that this model will be very useful in industries for improving inventory management, workforce planning, and marketing strategies for profit maximization and customer satisfaction. For future studies, there is a possibility of using other deep learning algorithms, such as LSTM (Long Short-Term Memory), to improve the precision of the forecast, especially when dealing with complex data. In addition, the model can be modified to include large-scale data sets from various industries such as e-commerce or supply chain management to analyze its efficiency. The inclusion of other factors, such as the economy and social media trends, among others, could further improve the forecast. Additionally, improving the model's scalability to process vast amounts of data streams and integrating with cloud-based platforms for real-time deployment could bring in some pretty major real-world improvements.

Declarations

Funding:

The author does not receive any funding for this research

Conflict of Interest:

The authors declare no conflict of interest in relation to this work

References

1. Elorza, M., Castellano, E., & Segura, S. (2025). Prediction of customer demand for perishable products in retail inventory management using the hybrid Prophet-XGBoost model during the post-COVID-19 period. Applied Economics Letters, 32(17), 2453–2459.

2. Sethi, K., & Kapoor, M. (2024). Data-driven marketing in the age of AI: Reflections from the periodic series on technology and business integration. In Digital marketing innovations (pp. 7–11). Periodic Series in Multidisciplinary Studies.
3. Dolon, M. S. A. (2025). Hybrid machine learning-driven financial forecasting models: Integrating LSTM, Prophet, and XGBoost for enhanced stock price and risk prediction. Review of Applied Science and Technology, 4(1), 1–34.
4. Vardhan, H., & Bhattacharya, R. (2025). The impact of sustainable practices on business performance. International Journal of SDG's Prospects and Breakthroughs, 3(1), 15–21.
5. Mustapha, O. O., & Sithole, T. (2025). Forecasting retail sales using machine learning models. American Journal of Statistics and Actuarial Sciences, 6(1), 35–67.
6. Lei, C., & Ibrahim, M. (2024). Efficient revenue management: Classification model for hotel booking cancellation prediction. Global Perspectives in Management, 2(1), 12–21.
7. Ratre, S., & Jayaraj, J. (2023). Sales prediction using ARIMA, Facebook's Prophet and XGBoost model of machine learning. In Machine learning, image processing, network security and data sciences: Select proceedings of the 3rd International Conference on MIND 2021 (pp. 101–111). Springer Nature Singapore.
8. Choudhary, N., & Verma, M. (2025). Artificial intelligence-enabled analytical framework for optimizing medical billing processes in healthcare applications. Global Journal of Medical Terminology Research and Informatics, 3(1), 1–7.
9. Loa, J., Wiratama, J., & Halim, F. A. (2025, June). Optimizing inventory management in retail companies through sales prediction using XGBoost. In 2025 4th International Conference on Electronics Representation and Algorithm (ICERA) (pp. 1–6). IEEE.
10. Ghosh, A., & Rathi, U. (2024). Investigating wind speed forecasting for grid-connected renewable energy systems using LSTM and a day-ahead chaining. International Academic Journal of Science and Engineering, 11(3), 23–26. https://doi.org/10.71086/IAJSE/V11I3/IAJSE1156
11. Papadopoulos, G., & Christodoulou, M. (2024). Design and development of data driven intelligent predictive maintenance for predictive maintenance. Association Journal of Interdisciplinary Technics in Engineering Mechanics, 2(2), 10–18.
12. Sravan Kumar Reddy, M., & Supraja, C. (2025). Optimising financial forecasting: The power of reduction approach. Archives for Technical Sciences, 2(33), 453–462. https://doi.org/10.70102/afts.2025.1833.453
13. Amadi, C. O., Kahya, S., Danbaba, A. K., Dabels, V., Kalastone, N., Nwadili, V. U., et al. (2023). Rolling back the scourge of late blight in Nigeria: Efforts by the National Root Crops Research Institute Umudike to end recurrent outbreaks. CENTENNIAL, 841.
14. Huy, D. T. N. (2018). Selecting various industrial competitors affect the risk level of Viet Nam wholesale and retail industry during and after the global crisis 2007–2011. International Academic Journal of Humanities, 5(1), 187–195. https://doi.org/10.9756/IAJH/V5I1/1810019
15. Asghar, R., Fulginei, F. R., Quercio, M., & Mahrouch, A. (2024). Artificial neural networks for photovoltaic power forecasting: A review of five promising models. IEEE Access, 12, 90461–90485.
16. Arifa, P. A., & Devasenapathy, K. (2025). Sales prediction using LSTM and BiLSTM models: A deep learning approach for time series forecasting. Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, 16(3), 393–402. https://doi.org/10.58346/JOWUA.2025.I3.023
17. Sofiazizi, A., & Kianfar, F. (2015). Modeling and forecasting exchange rates using econometric models and neural networks. International Academic Journal of Innovative Research, 2(1), 49–65.
18. Sherly, A., Christo, M. S., & Elizabeth, J. V. (2025). A hybrid approach to time series forecasting: Integrating ARIMA and Prophet for improved accuracy. Results in Engineering, 27, 105703.
19. Jamithireddy, N. S. (2025). Cash flow forecasting in SAP ERP enhanced by UiPath automation: A predictive analytics approach. Indian Journal of Information Sources and Services, 15(2), 368–379. https://doi.org/10.51983/ijiss-2025.IJISS.15.2.45
20. Aswathy, S. U. (2026). AI innovations driving the future of automation in manufacturing and service industries. Global Tech Management Digest, 2(1), 7–12.
21. Boon, N. S., Boumediene, K., & Belkhamza, Z. (2018). An integrating model of the factors influencing online shopping for consumer goods. International Academic Journal of Social Sciences, 5(2), 23–31. https://doi.org/10.9756/IAJSS/V5I2/18100023