# Weekly Sales Forecast using Prophet Model

Beom Su Mun1[0009-0001-3717-6814], Do Yoon Kwon1[0009-0006-5823-520X], Jae Hyeon Lee1[0009-0007-4145-0162], Yerin Choi1[0009-0007-2644-7731], Seol Young Jeong1[0000-0003-0175-7710] and Chan Sik Park2[0009-0002-0140-8975]

[1] Kyungpook University, 80, Daehak-ro, Buk-gu, Daegu, Republic of Korea
[2] CSP Mobile Lab, Hyeonchung-ro, Nam-gu, Daegu, Republic of Korea

**Abstract.** In recent markets, establishing strategies to predict and respond to the future is required to deal with unpredictable situations such as COVID-19 and Russia-Ukraine war. However, data analysts worldwide lack manpower, and it is not easy to secure a high level of manpower in all companies.

To solve these problems, a predictive model that can be fully utilized by non-experts must be developed, and the necessary data must be intuitively represented. Also, upgrading work that increases accuracy in the future should be possible. Researchers has learned and studied various predictive models that are currently available to present and effective sales forecasting model which helps companies. In addition, to derive realistic results, the weekly forecast values were verified through various verification methods using actual corporate data and the prophet model was selected accordingly. This study talks about future directions based on the selection background and actual results.

**Keywords:** Prophet, Sales Forecasting, Deep Learning.

## 1 Introduction

Coffee chains such as cafes are one of the industries with a lot of data available through floating population, sales information, and surrounding commercial analysis services. Due to the nature of the industry, it has a structure that is highly affected by time zone, gender, age, and floating population, and if the relationship between these information can be found, it can provide predicted profit information to stores or companies.

The pandemic COVID-19, which began in 2020, has also affected commercial districts as social distancing, gathering restrictions, and public transportation cuts have been implemented worldwide. Kiosks that can satisfy these regulations have drawn attention as they prefer non-face-to-face orders rather than face-to-face orders, focusing on takeout rather than in-store meals. As orders through kiosks have expanded, it has become a necessity, not a recommendation, across the restaurant industry, and it has naturally become easier to collect detailed data such as sales information and order types.

Measures were required to cope with unpredictable situations such as COVID-19, and naturally, demand and services related to prediction increased. In the case of cafes, models and services that can provide a stable profit structure are essential now when daily life cannot be fully recovered due to the rapidly changing frequency of startups and the closure rate of existing companies.

In this research, we select some branches of Hammersmith Coffee and predict future weekly sales data using Prophet based on data from January to October 2023, and introduce related models.

## 2 Related Research

### 2.1 Prophet

$$y(t) = g(t) + s(t) + h(t) + e_t \tag{1}$$

Prophet (1) basically follows three elements. ***g(t)*** is a trend that can check the overall flow, ***s(t)*** is a repetitive change in seasonality such as days, weeks, and months, ***h(t)*** is a factor that sometimes affects irregularly, such as holidays and $e_t$ is an assumption that follows a normal distribution. Setting these variables has several advantages [1].

First, predictions about different periods and seasonality are easily applied to the model, providing analysts with various opportunities to try. Next, unlike other models, there is no need to divide and normalize the model, and there is no reason to add missing values.

As a result, analysts can execute Prophet by adjusting conditions such as Capitals, Changepoints, Holiday and Seasonality, and Smoothing Parameters. If analysts know how to use visualization tools, they can reduce the error rate by adjusting the following conditions.

### 2.2 ARIMA

ARIMA model is short for Auto-Regressive-Integrated-Moving-Average and is a differentiated form of two models. First, AR is based on intuitive common sense to predict the future through past data, and the most important decision when estimating an AR model is how far past data will be considered in the model. This AR model is not suitable for situations where trends change.

Next, the MA model is called the moving average model that is suitable in situations where the trend changes.

The ARMA model is a combination of AR and MA. It is like the concept of increasing the terms of independent variables in regression analysis, meaning that it reflects all the past time's self and trend to grasp the state of the present time.

The ARIMA model aims to make better predictions in non-stationary situations through the concept of difference, if the previous models were conducted under the assumption that the time series is normal. To name a big drawback of this ARIMA model, it is difficult to use various features because predictions are made with only dependent variables. Therefore, the error rate is formed high in time series analysis in which various factors are considered.

### 2.3 LSTM

The LSTM model is a modified version of the RNN and is a model created to compensate for the problems of the vanishing gradient problem and exploding gradient problem [2, 3].

The LSTM is explicitly designed to avoid the problem of long dependence periods. To introduce only the key idea, unlike the RNN, one layer contains four interaction layers, and each repetition module has a different structure. Instead of a simple layer of natural network layer, four layers are supposed to exchange information with each other in a special way.

Since LSTM increases nonlinearity of information from the past through various gates and utilizes more efficiently, it can control information from a longer time point more finely than RNN.

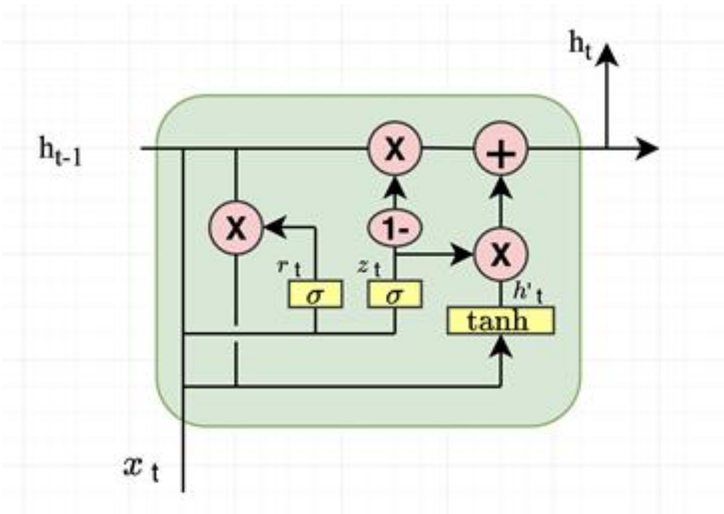


**Fig 1.** GRU structure

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

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t]) \quad (2)$$

$$h' = \tanh\,(W \cdot [r_t * h_{t-1}\,, x_t]) \tag{3}$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * h'_t \tag{4}$$

The LSTM introduced in this paper is a GRU method modified from the existing LSTM as shown in **Fig. 1.** In the structure, forget gate and input gate are combined into one update gate, and there are various changes such as combining cell state and hidden state. As a result, the GRU has a simpler structure than the existing LSTM, and the GRU is better when there are few parameters [4].

The process performed in each step is expressed in an equation. First, in the case of (1), the sigmoid function is used as an output to multiply the value of (0, 1) to the previous hidden layer for the purpose of properly resetting past information with the Reset Gate. It may be obtained by multiplying the value of the hidden layer at the immediately preceding point and the information at the present point by a weight.

(2) determines the ratio of past and present information updates by feeling like the combination of LSTM's forget and input gate as an Update Gate. If the result output in sigmoid determines the amount of information at the present time, subtracts this value from 1 and multiplies the hidden layer information($h_{t-1}$) at the previous point, each is like the input gate and forget gate.

(3) is the Candidate step of calculating the current group of information candidates. The key is not to use the information of the past hidden layer as it is, but to multiply the result of the reset gate.

(4) is the step of calculating the current hidden layer by combining the update gate result and the candidate result. The result of the sigmoid function determines the amount of information in the current result, and the result of the 1-sigmoid function determines the amount of information in the past.

GRU has no significant difference in structure from LSTM, and there is no significant difference in analysis results. In other words, the performance of each model may vary depending on how it is used, and it is a clear advantage that GRU has less weight to learn.

### 2.4 Final Analysis Method Selection

Since the data provided by the company was monthly sales data for each branch, it was necessary to be able to predict small data efficiently, and two aspects that were not difficult for various variables were considered.

Prophet model, one of the libraries provided by the current Meta, is a business-time-series tool created to solve the current problem which there are few series experts. Time series analysis is limited by the fact that fully automated time series are difficult to tune and those with excellent corporate domain knowledge lack knowledge of time series. Prophet has been developed as a time series capable of scaling the model itself and aims to provide a time series tool that can be used by as many people as possible and can be touched by anyone who can consider multiple features.

Therefore, the Prophet model is fast in training and can adjust various detailed model specifications. As a result, the Prophet model was selected as an analysis model because it is like regression analysis and could be adapted quickly rather than other difficult and unfamiliar time series analyses.

# 3 Analyzing Experimental Models

## 3.1 Used Models

The data used in this model are as follows.

**Table 1.** Using Data

| **Index** | **ds[Date]** | **y[Actual Daily Sales]** |
|---|---|---|
| i | 0000-00(datetime64[ns]) | Sales(float64) |

According to **Table 1**, the index uses the base type, and ***ds*** and ***y*** are used as data. ***Ds*** uses the corresponding 11 months from January to November in datetime64 format, and ***y*** uses an Actual Daily Sales in float64 format. The Prophet model analyzes only through two data values. Other additional settings are performed by adjusting the parameter.

## 3.2 Modeling

According to the thesis parameter capable of adjusting g(t), s(t), and h(t) described above is introduced [1]. In this study, we intend to derive realistic predictions through actual sales.

1) Adjusting Trend

**Table 2.** Trend Parameter

| **Parameter** | **Description** |
|---|---|
| changepoints | List value specifying when the trend changes |
| changepoint_prior_scale | Changepoint (trend) flexibility set |
| n_changepoints | Number of changepoints |
| changepoint_range | changepoint configurable range |

**Table 2** describes parameters that can adjust the trend of the model. These values play a role in detecting trends more flexibly, paying attention to and adjusting over-fitting. You can specify when the trend changes at changepoints and determine how flexible to detect at changepoint_prior_scale.

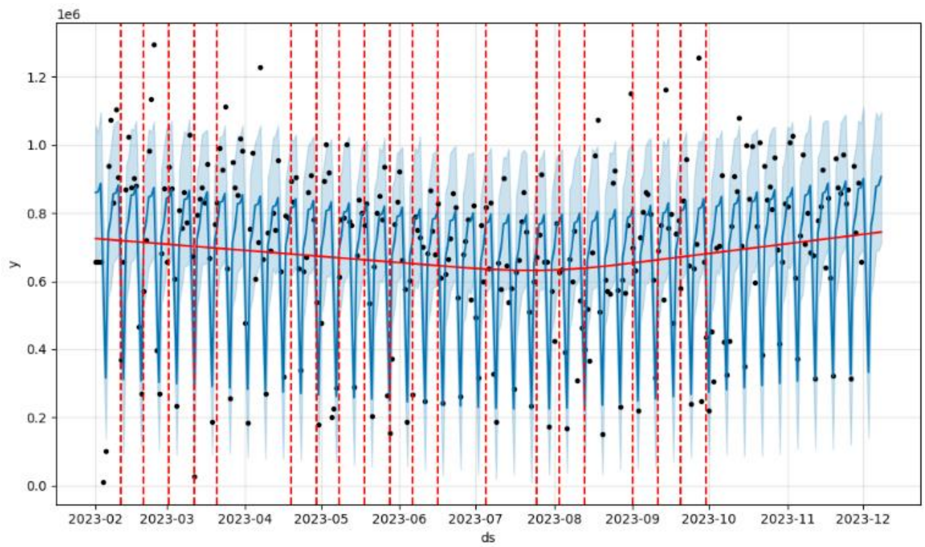


**Fig 2.** Fitting Graph (Using trend parameters)

**Fig 2** is a graph that predicts data over the next eight days and expresses where the change points are. The solid red line also represents the overall trend of the data. By setting the value of the changepoint_prior_scale, you can control how sensitive the change point is to be detected, and you can identify the trend while changing the value to avoid overfitting or underfitting.

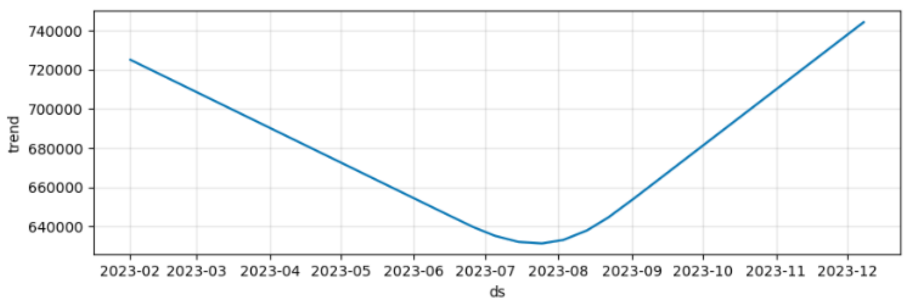


**Fig 3.** Trend of **fig 2**

**Fig 3** is a graph representing the trend of sales data of the corresponding branch. As can be seen in Fig 2, overall sales during the summer period fell briefly, and sales rose at the beginning of the year and the end of the year.

2) Adjusting Seasonality

**Table 3.** Seasonality Parameter

| Parameter | Description |
|---|---|
| yearly_seasonality | Yearly seasonality |
| weekly_seasonality | Weekly seasonality |
| daily_seasonality | Daily seasonality |
| seasonality_prior_scale | Seasonal reflection Weight |
| seasonality_mode | additive or multiplicative |

The **Table 3** parameters play a role of flexibly catching patterns for each period. There are no parameters for each month, so you can add them if you need them. Seasonality mode specifies whether the amplitude of the data is constant (Additive), increasing or decreasing (Multiplicative).

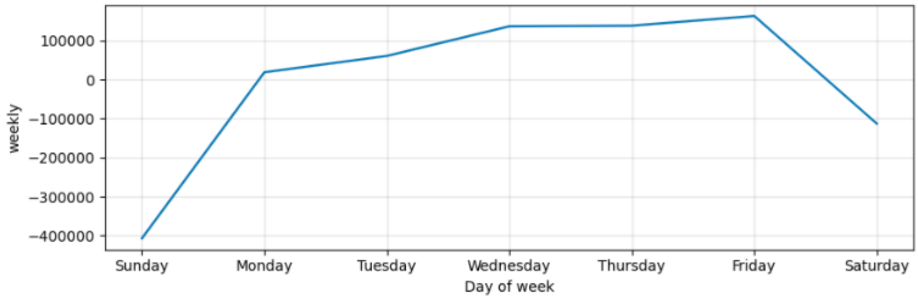


**Fig 4.** Weekly Seasonality Trend

**Fig 4,** the flow of data can be checked by setting a specific period using the seasonality parameter. The graph represents the weekly trend of sales data, and compared to actual sales, the area is an office area, that is, a store where sales decrease due to a decrease in the floating population on weekends, and the weekly seasonality is well understood.

Since the analyzed data was not one year's worth of data, the annual analysis could not be performed, but the periodicity could be confirmed through other available weekly seasonality parameters.

3) Adjusting Holiday

**Table 4.** Holiday Parameter

| Parameter | Description |
|---|---|
| holidays | Data frames that specify the dura tion of a holiday or event |
| holiday_prior_scale | Holiday Reflective Weight |

**Table 4** introduces parameters that can increase accuracy by reflecting them in the model if you know them. This parameter plays a role in detecting the part of the data where the numerical value changes rapidly. In other words, it helps to predict similarly about sales generated during holidays and event periods in the future.

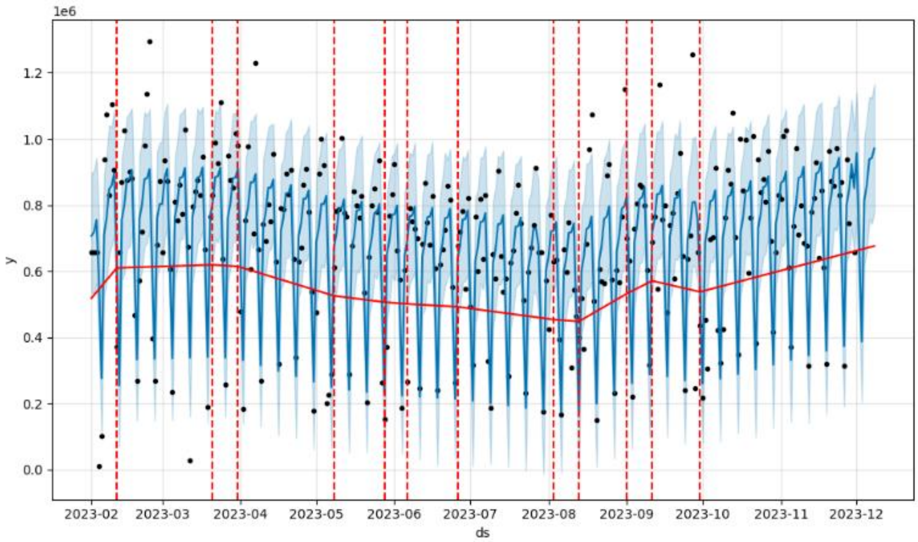


**Fig 5.** Fitting Graph (Using Trend, Seasonality and Holidays parameters)

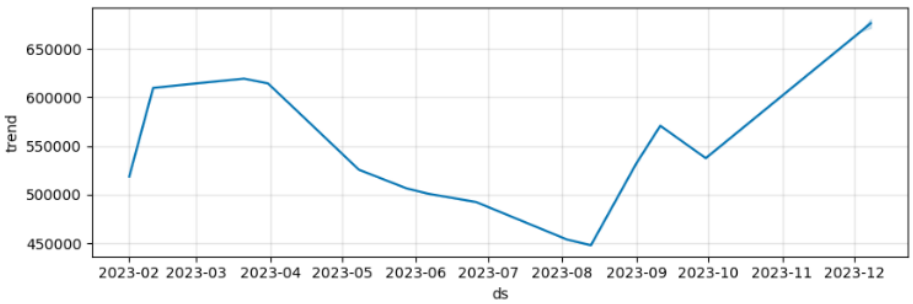


**Fig 6.** Trend of **Fig 5**

**Fig 5** as can be seen from the two graphs above, after setting the holiday parameter, it shows how the trend is reflected more flexibly. Although the current data could not utilize a year-long holiday, the date of the weekend was combined with the holiday data so that the value over the weekend could be predicted uniformly.

### 3.3 Verification Results

The values and equations of MSE (1), RMSE (2), MAE (3), and MAPE (4) corresponding to the evaluation indicators of the predictive model are as follows.[5]
[In the equation, y = actual value, x = predicted value and N = amount of Data.]

$$E = \frac{1}{N}\sum_{i=1}^{N}(y_i - x_i)^2 \tag{1}$$

$$E = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - x_i)^2} \tag{2}$$

$$E = \frac{1}{N}\sum_{i=1}^{N}|(y_i - x_i)| \tag{3}$$

$$E = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - x_i}{y_i}\right| \tag{4}$$

MSE (1) is the sum of the squared deviations divided by the total amount of data. This allows equity between data of different sizes to be matched and is used as an estimate of the expected error square value.

RMSE (2) is an intuitive indicator of error. Because the root is used, the values can be expressed much smaller.

MAE (3) is an evaluation index optimized for the Laplace-distributed error using an absolute value instead of a square value. Because it is an absolute value, it is less sensitive to outliers than when using squares.

MAPE (4) is a method of taking an absolute value to the error rate to match the error rate. This prevents scale-invariant error.

The accuracy verification results of the model, which analyzed daily sales over a 10-month period at a total of seven branches and calculated the estimated sales in the first week of November, are shown in the following graph. **Fig. 2**

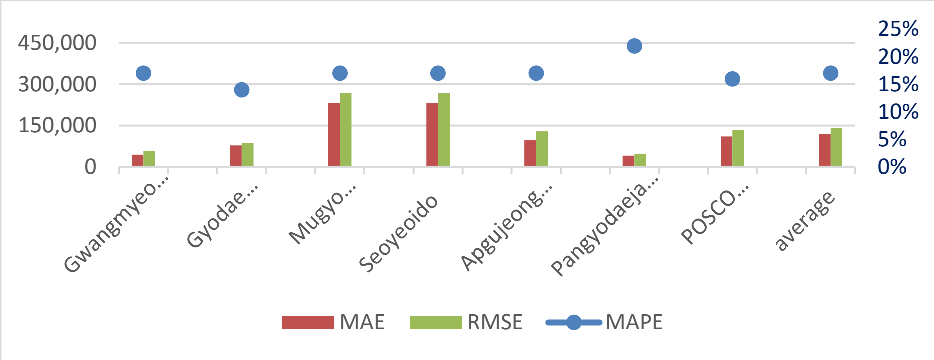


**Fig 6.** MAPE, MAE and RMSE figures for each point

According to **Fig 6** the MSE value was not expressed as too large. According to the graph, the error rate is in the range of 10-20%, and the overall error value is in the range of 200,000 won to 4.5 million won.

## 4 Conclusion

All evaluation indicators do not guarantee perfect values, but due to the nature of the store, it is considered highly valuable to use if an approximate amount can be provided.

The core of this study is to learn about various time series data prediction models and to select the best model for a given situation. Therefore, the future direction and complementary tasks are clear, and this study is expected to provide simple predictive information and simple indicators for future investment directions to companies. In the future, it is expected that research can be conducted to increase accuracy by changing the parameters to upgrade the model and adding various conditions.

In addition, comparisons between other models that are insufficient in this study are expected to be additional tasks. In selecting the model, the logical approach problem is expected to be solved only when the performance analysis and differences from the objective comparison model can be clearly identified.

Finally, if companies continue to provide data, it means that there is more data that can be learned, which leads to an increase in the evaluation index of the model. There is also a need to check whether simply increasing the amount of data leads directly to the performance of the model.

## References


1. Taylor, Sean J., and Benjamin Letham. "Forecasting at scale." The American Statistician 72.1 (2018): 37-45.

2. Author, Lee Sehee., Author, Lee Ji-hyung.: "Customer Churn Prediction Using RNN" Proceedings of KSCI2016 Conference 2016.7 (2016): 45-48.

3. Author, Jiyoung Ko., Author, Yung-Cheol Byun.: "Electric Mobility Sales Forecast Using Tree-Based Ensemble Models." Proceedings of KIIT Conference 2023.6 (2023): 9-11. Author, F., Author, S., Author, T.: Book title. 2nd edn. Publisher, Location (1999).

4. Lee JuHyung and Hong Joongi. "Comparative Analysis of Prediction Performance of Aperiodic Time Series Data using LSTM and Bi-LSTM." The journal of Bigdata v. 7 no.2 (2022): 217-224.

5. Author, Ko Hyeong Seok, Lee Dong Kyu, Lee Seul Gi, Lee Jun Seok, Ha Jeong Won, and Kim Jae Hyun.: "Predicting the Number of Daily Subway Passengers using the Prophet Model " Proceedings of KIIT Conference 2023.6 (2023): 427-431.


"This research was supported by the Korean MSIT (Ministry of Science and ICT), under the National Program for Excellence in SW) (2021-0-01082) supervised by the IITP (Institute of Information & communications Technology Planning & Evaluation)"(2021-0-01082)

This study has been conducted with the support of the Ministry of Science and ICT and the Korea Information and Communication Industry Promotion Agency's Project to Create a Foundation for Developing SW Talent in Local Industries.