Predicting Air Ticket Demand using Deep Neural Networks

Kodai Imanaka
Graduate School of Systems Engineering
Wakayama University
Wakayama, Japan
imanaka.kodai@g.wakayama-u.jp

Chiaki Sakama
Department of Systems Engineering
Wakayama University
Wakayama, Japan
sakama@wakayama-u.ac.jp

***Abstract*—Predicting air ticket demand is crucial for both airline companies and travel agencies, while the task is generally hard due to its dynamic nature and few attempts have been made to apply machine learning techniques for this purpose. This paper provides an empirical study for predicting airline tickets sales using deep neural networks. A new learning model is introduced by extending the Long Short-Term Memory (LSTM) for handling non-time series data as well as time series data. The proposed model is compared with the SARIMAX model that is used for forecasting time series data with seasonal patterns. We perform experiments using real data and show that the proposed model captures demand changes better than the SARIMAX. In particular, features related to the day of the week and different airlines are well predicted.**

***Index Terms*—airline tickets, time series prediction, deep neural network, long short-term memory, time series model**

I. Introduction

Nowadays airline tickets are mostly booked on the Internet and ticket sales change every moment. Airline companies use dynamic pricing to maximize their profits. The price for the same service class on the same flight may rise or fall within several days. Recent big data analysis enables airlines offering personalized prices to customers [9]. On the demand side, customers become more price sensitive as time to departure nears and the number of active customers increases closer to departure [6]. Air ticket demands are also of particular interest for travel agencies. They purchase tickets from airline companies and sell to customers, then it is crucial for agencies to predict market demand from the viewpoint of financial management. In addition to dynamic change in the booking status, however, the types of airline tickets sold by travel agencies are diverse, which makes it difficult to predict the sales of airline tickets. Moreover, ticket sales data are business sensitive and most airlines do not reveal their data. There is lack of publicly available datasets that could enable researchers to conduct prediction effectively. As a result, most studies use small datasets that are available on the Web [3].

Recent development of machine learning techniques, especially deep neural networks, enables to automatically discover patterns in data and make predictions. Several researchers have applied artificial neural network techniques (ANN) for predicting air ticket or air travel demand. Huang [8] introduces a model that combines ANN and the genetic algorithm for predicting air ticket sales revenue of travel agencies. They use a back propagation architecture of ANN. Mostafaeipour $et\,al.$ [4] use a three-layered ANN for predicting air travel demand in airports. They use evolutionary meta-heuristic algorithms to improve ANN and show that it increases adaptation rate between prediction and real data. Those two studies use simple neural networks with a single hidden layer. Yuan $et\,al.$ [14] develop a model for predicting air ticket demand to increase revenue of an online travel agent. The model estimates the effect of customer calls as internal factors and customers' search engine query history as external factors. It then compares historical weekly ticket price data fluctuations with those factors. The model consists of ANN and two types of support regressions. Pan $et\,al.$ [11] use the long short-term memory (LSTM) model that is a type of recurrent deep neural networks. It then forecasts daily airline demand using two time series data: a horizontal time series for short-term forecasting, and a vertical time series for long-term forecasting. It is shown that the model achieves the best prediction accuracy compared with other classification techniques. The dataset used in its evaluation is restricted to one route by one airline.

In this study, we use deep neural networks to predict air ticket demand based on the past ticket sales. We introduce a new learning model called LSTMX that extends the LSTM by introducing exogenous variables for non-time series data. We evaluate on real data provided by a travel agency to predict air ticket demands of several flights by different airlines. We then compare the results with those predicted using a time series model SARIMAX [2]. Our experimental results show that the LSTMX well captures demand changes in contrast to the SARIMAX. The LSTMX also distinguishes trends of different airlines on the same route. The rest of this paper is organized as follows. Section 2 addresses description of data used in this study. Section 3 introduces a learning model and a method of learning. Section 4 presents experimental results and analyses. Section 5 summarizes the paper.

II. Data Description

In this study we use real data provided by a travel agency. The data present reservations of international flights arriving and departing the major airports of Japan. The number of

978-1-6654-3902-2/21/$31.00 ©2021 IEEE

TABLE I: Data attributes

Attributes	Value	Data type
outbound/inbound	0 or 1	non-time series
airline company	airline code	non-time series
departure date	Y/M/D	non-time series
departure time	0:00–23:59	non-time series
booking class	A–Z	non-time series
remaining seats	integer from 0 to 9	time series
day of the week	integer from 0 to 6	non-time series
holiday	0 or 1	non-time series

flights is approximately 800 per day and the data contain the reservation status of each flight for the next 6 months on each day.[1] Among them, we use data of four air routes: Tokyo-Paris, Osaka-Taipei, Osaka-Helsinki, and Nagoya-Honolulu, where each route has different airlines. Departure day is from 10/6/2018 to 30/9/2019 (approximately 16 months). The data contain several attributes for each flight such as outbound/inbound flight distinction, airline company, flight number, aircraft type, departure airport, arrival airport, departure date, departure time, arrival date, and booking class. Among those attributes, we use outbound/inbound flight distinction (0 or 1), airline company, departure date, departure time, and booking class. Booking classes are given to tickets by each airline and are represented by the letters A to Z. Generally, international flights have seat classes such as first class, business class, and economy class. The same seat class is further classified into different booking classes, depending on time of reservation, availability of changes after reservation, mileage accrual rate, etc. The number of remaining seats in each booking class is represented by an integer from 0 to 9. By the specification of data, 9 is used if the number of remaining seats is 10 or more.

In addition to these attributes, we add indices that represent the day of the week and the holiday flag of the departure date for each flight. The day of the week is specified by integers, e.g., Mon:=0, Tue:=1, ... Sun:=6. The holiday flag is set to 1 for departure on weekends, national holidays, and busy seasons such as summer vacation or the year-end and New Year holidays. Otherwise, it is set to 0. Table I summarizes attributes/values of dataset used in this study. Among those attributes, remaining seats are time series data (i.e., a series of data points indexed in time order) and others are non-time series data.

III. Learning Model

A. LSTMX

We apply a deep neural network to predict the sales of airline tickets. Since the fluctuation of the remaining number of tickets is expressed as time series data, we use the Long Short-Term Memory (LSTM) that is suitable for learning time series data. On the other hand, we also handle non-time series data as presented in Section II. Input of the LSTM is assumed to be time series data that are meaningful in its order, then the

[1]For each day, the data contain approximately 140 thousands of tuples in the csv format (22MB).

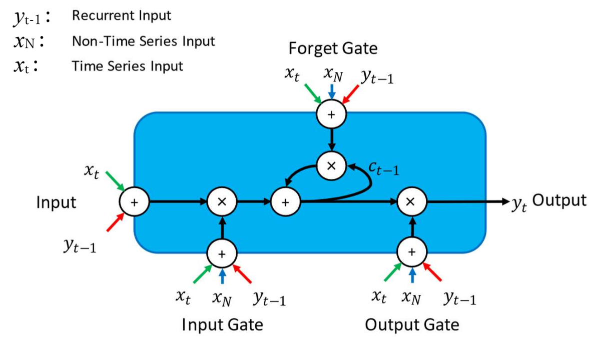


Fig. 1: LSTM for non-time series data

prediction accuracy will be decreased if non-time series data whose order are not meaningful are given as input. To cope with the problem, we first extend an LSTM model to handle non-time series data as well as time series data (Fig. 1).[2] In this model, time series data, the current input x_t and the recurrent input y_{t-1}, are put into the block input, the input gate, the output gate, and the forget gate in the same way as the conventional LSTM. By contrast, non-time series data x_N are represented as exogenous variables and are not fed into the block input. c_{t-1} denotes the activation of the memory cell at time step $t-1$ and circles containing $\times$ represent an element-wise multiplication between its inputs. As a result, non-time series data are not stored in the memory cell and are handled separately from time series data.

We use this extended type of LSTM to build a learning model for the current purpose. The architecture of our learning model (called LSTMX) is illustrated in Fig. 2. Data are processed in this learning model as follows.

1) Time series data and non-time series data are processed separately in the LSTM block as in Fig. 1. The number of LSTM cells is 128. Data are processed with a batch size of 64 and an epoch of 100 training steps.
2) Time series input is also fed into the autoregressive component that consists of two affine layers. The first layer is supposed to detect linear features in each series and the second layer is supposed to reflect those features. The first layer has 32 output units and the second has one output unit.
3) The output of the LSTM is input to the Dropout layer in order to prevent overfitting to training data. The dropout rate is set as 0.2. After the Dropout layer, data are combined into 32 output units in the first affine layer, then again combined into one output unit in the second affine layer.
4) The output of the LSTMX is obtained by combining the outputs of the LSTM part and the autoregressive component. In each layer the ReLU function is used for activation. We use the Adam optimization method with the learning rate of 0.01.

[2]A similar model is introduced in [7].

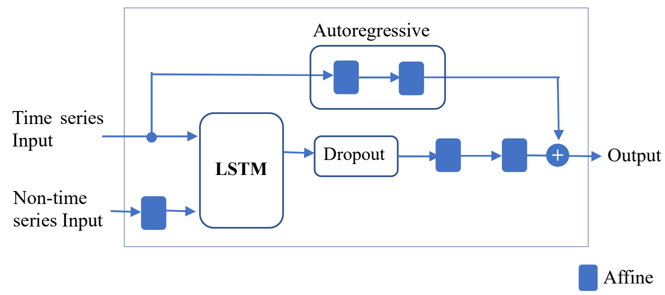


Fig. 2: LSTMX

The idea of introducing the autoregressive component apart from the neural network is introduced in the LSTNet [10]. It successfully captures both short-term and long-term patterns in time series data.

B. SARIMAX

The SARIMAX [2] is one of the most popular techniques currently used to forecast a time series. The SARIMAX extends the ARIMA model [5] by introducing seasonality and exogenous variables as

$$y_t = c + \sum_{i=1}^{p} \phi_i y_{t-i} + \sum_{i=1}^{q} \theta_i \epsilon_{t-i} + \beta_t^T \boldsymbol{x}_t + \epsilon_t$$

where $\boldsymbol{x}_t$ is an exogenous input vector at time t and β_t^T its coefficient, y_t is time series data, ϵ_t is the noise at t, c is a constant, and ϕ_i and θ_i are coefficients for AR (autoregressive) and MA (moving average) models, respectively.

The SARIMA model is represented by $\textsc{Sarima}(p,d,q)(P,D,Q)[s]$ where (p,d,q) (resp. (P,D,Q)) represents non-seasonal (resp. seasonal) part of the model. p and P are the order of AR processes, q and Q are the order of MA processes, and d and D are the degree of differencing (number of times it is differenced). s represents a season's cycle. In this study we assume the seasonality as the weekly cycle. So we use the SARIMAX model with $\textsc{Sarima}(p,1,q)(0,0,0)[s]$ where $s = 7$ when there is a weekly cycle; otherwise, set $s = 0$. We assume that data have an unstable seasonal pattern over time, and set $P = D = Q = 0$. The parameter p and q are selected in the range of $0 \leq p,\, q \leq 4$ to make the AIC (Akaike's Information Criterion) parameter minimal. Here, the AIC parameter is defined as: "$AIC = -2\ln L + 2k$" where L is the maximal value of the likelihood function of the model and k is the number of parameters. We set $d = 1$ to have the difference $y_t - y_{t-1}$.

C. Learning Method

The dataset presented in Section II is preprocessed to feed into a neural network as follows.

- Time series data are classified based on flight destinations and booking classes. An input has one time series data that contain the number of remaining tickets for each booking class on 14 days before the departure date. Some flights are not scheduled every day. If there is no flight on a day, the number of remaining seats is set to -1.
- Airlines, the month of departure date, departure time, the day of the week, and types of booking classes are distinguished by dummy variables that assign 1 to applicable parts. For instance, if there are three airlines A, B, C and data represent a flight by A then the airline information is represented by 100.
- Different airlines have different booking classes. To compare prediction results among different airlines, booking classes are ordered by the price in each airline.
- Input data are represented by vectors. Time series data are represented by (number of dataset, window size, number of remaining tickets), and non-time series data are represented by (number of dataset, outbound/inbound, airline company, departure month, departure time, day of the week, booking class, holiday).

TABLE II: Air routes, airlines, flight and booking classes

Air routes	airlines	flights	Business	Economy
Tokyo – Paris	3	8	40	40
Osaka – Taipei	3	20	78	100
Nagoya – Honolulu	2	4	20	12
Osaka – Helsinki	1	2	10	10

As stated above we predict the number of remaining tickets on 14 days before the departure date. This is because we mainly focus on demand for leisure travel and customers are more likely to purchase tickets for this purpose more than 14 days before departure. As addressed in Section II, we use data from 10/6/2018 to 30/9/2019. Sequences of data for 14, 28, 56 or 84 days in this period are set as window sizes and fed into the LSTM. The number of remaining tickets of departure on the next day after a sequence is used as training data. This window size assumes fluctuations in the number of remaining tickets for the last two weeks, one month, two months, and three months before departure. If the number of remaining ticket is -1 in training data, it is removed. The same time series data is used for the SARIMAX model, while a model is generated for each booking class. It has variables for the month of departure date, departure time, the day of the week of departure date, and the holidays.

We construct a model for each air route. Table II shows air routes, the number of airlines, the number of flights, and the number of booking classes that are considered in this study. In each route, a half of the flights are inbound and the other half are outbound. We use every booking class in business class. For economy class, on the other hand, 3 to 5 booking classes are selected from each flight that have different characteristics of fluctuation. In Table II, Business and Economy represent the number of booking classes in business class and economy class, respectively.

We predict the number of remaining tickets for each booking class during the period from 1/10/2019 to 31/12/2019 (92 days). The prediction is done as follows.

1) Using the latest time series data $y_{t-1}, \ldots, y_{t-p}$ with

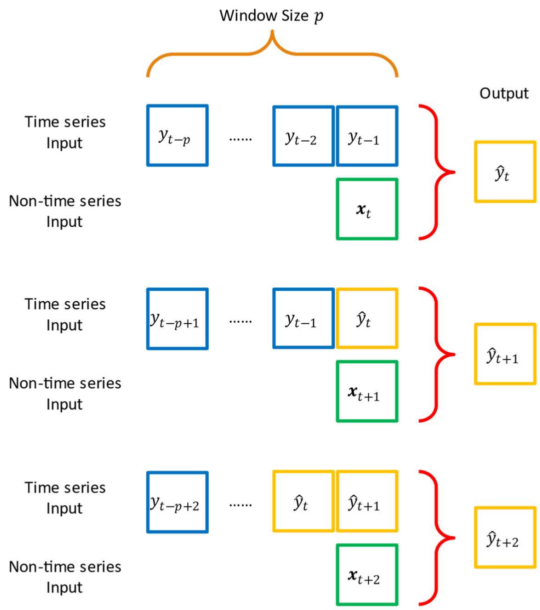


Fig. 3: Prediction Method

the length p of a window size and non-time series data x_t, predict the number $\hat{y}_t$ of remaining tickets for each flight.

2) Incorporating the predicted value $\hat{y}_t$ at Step 1 into the time series input, predict the number $\hat{y}_{t+1}$ of remaining tickets for each flight on the next day as Step 1.
3) Repeat Steps 1–2 for days during the period.

The process of prediction is illustrated in Fig. 3. For instance, if we want to predict the number of remaining tickets 14 days before the departure on May 31, then set (in case of $p = 14$):

x_t : non-time series data of the flight on May 31;

$\hat{y}_t$: the number of remaining tickets on May 17 for the flight departing on May 31;

y_{t-1} : the number of remaining tickets on May 16 for the flight departing on May 30;

y_{t-2} : the number of remaining tickets on May 15 for the flight departing on May 30;

...

y_{t-14} : the number of remaining tickets on May 3 for the flight departing on May 30.

The predicted value $\hat{y}$ is a real number then set: $\hat{y} = 9$ if $\hat{y} \geq 9$; otherwise, $\hat{y}$ is rounded to the nearest integer. If there is no flight on a day, the predicted value is set to $\hat{y} = -1$.

We use the RMSE (Root Mean Squared Error) as a loss function to evaluate the prediction accuracy:

$$RMSE = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(y_n - \hat{y}_n)^2}$$

where N is the number of days to be predicted, y_n is the actual value on a day, and $\hat{y}_n$ is the predicted value on the same day. If there is no flight on a day, the day is not counted in N. Since we predict values in 92 days, set $N = 92$. We perform learning 10 times and take the average values of RMSE. The value of RMSE represents the average of the difference between the real value and the predicted value of the number of remaining tickets. So smaller RMSE reflects greater accuracy.

IV. Experimental Results

In this experiment, we apply two prediction models, the LSTMX and the SARIMAX, to each flight in Table II. The number of data used in this experiment is shown in Table III. For instance, for flights in Tokyo–Paris 17,160 is the total number of time series data with the window size of 14 for 40 booking classes in business class. We use the Tensorflow/Keras library of Python for implementing the LSTMX and the Statsmodels for the SARIMAX.[3] All experiments are done using Google Colaboratory.

A. Comparison of RMSE

We calculate the average values of RMSE by changing the window size for different classes of each flight. For instance, Table IV provides RMSE values for each booking class of different flights in Tokyo–Paris. In the table, "size" means a window size. Table V shows the number of booking classes classified by the range of RMSE for all flights in Table III. In the table, ≤ 3, ≤ 4, and ≤ 5 mean that the range of an RMSE value x is $x \leq 3$, $3 < x \leq 4$, and $4 < x \leq 5$, respectively. By Table V, the following facts are observed.

- In business class, the LSTMX and the SARIMA make prediction within RMSE≤ 3 in almost the same number of booking classes. On the other hand, the LSTMX makes prediction within RMSE≤ 4 or RMSE≤ 5 in more booking classes than the SARIMAX.
- In economy class, the LSTMX makes prediction within RMSE≤ 3, RMSE≤ 4 and RMSE≤ 5 in more booking classes than the SARIMAX.
- Each model makes more precise prediction in business class than in economy class by comparing values in RMSE≤ 3. We use every booking class in business class while select some in economy class, which would result in more precise prediction in business class.
- There is no big difference in the results of prediction by changing the window size. This indicates that there are no distinctive features in the size of windows considered here. A longer-term fluctuation could be found by making the window size larger, while it takes much time for learning.

One learning model is considered better than the other if the model predicts the number of remaining tickets with lower RMSE in more booking classes. The RMSE of the LSTMX becomes lower when there are less fluctuation. This would be

[3]https://www.statsmodels.org/dev/generated/statsmodels.tsa.statespace.sarimax.SARIMAX.html

TABLE III: The number of data

Window size	Business class 14	28	56	84	Economy class 14	28	56	84
Tokyo – Paris	17,160	16,640	15,600	14,560	17,160	16,640	15,600	14,560
Osaka – Taipei	28,358	27,406	25,574	23,670	42,490	41,090	38,410	35,610
Nagoya – Honolulu	8,200	7,920	7,360	6,840	4,920	4,572	4,416	4,104
Osaka – Helsinki	4,480	4,340	4,060	3,780	4,480	4,340	4,060	3,780

explained by that the LSTMX has the autoregressive layer that extracts linear features in time series data. Even if the average RMSE is small, however, there could be a considerable difference between the real value and the predicted value in a particular time or periods. We then provide detailed analyses on the results of different flights in the following.

B. Comparison between LSTMX and SARIMAX

We compare the LSTMX and the SARIMAX in different flights. Fig. 4 shows prediction results for different booking classes of two flights. In each case, it is observed that the LSTMX takes wide ranged values, while the SARIMAX takes narrow ranged values. Two reasons are considered for the difference. First, the LSTMX can predict short-term fluctuations using time series data, while learning features of the day of the week with extrinsic variables. Second, a time-series model like SARIMAX tends to converge to the average of the time-series values of input data when a long-term prediction is made. As a result, the LSTMX makes "brave" prediction and captures dynamics better than the SARIMAX.

Generally, time series models require data to be stationary. If data is not stationary, it must be converted into a (weak) stationary series by differencing or logarithmic transformations. Moreover, even if data is stationary, the optimum parameters differ for each time series. In this study, we difference data and decide optimal parameters by restricting the range of p and q of $\textsc{Sarima}(p, d, q)(P, D, Q)[s]$ using the autocorrelation coefficient and the partial autocorrelation coefficient. It will take much time to seek optimal parameters outside those ranges. In contrast, there is no need to make data stationary in the LSTMX and the same model can be used for flights having different features. As such, the LSTMX can easily handle input data than the SARIMAX.

C. Prediction results

The LSTMX has extrinsic variables for handling non-time series data. As a result, the LSTMX often captures weekly fluctuations or features in different months. Fig. 5 shows two prediction results in a week. In DL611 no seat is available for departure on Tuesday and Friday, while there are some on other days. In AY78, on the other hand, five or more seats are available from Tuesday to Friday. The LSTMX well captures these dynamics. Fig. 6 shows two prediction results in October-December. AF279 has large amplitude in October and November, while CI173 has one in December. The LSTMX successfully predicts these dynamics. These results show that the month of departure data and the day of the week are effective in predicting fluctuation. By contrast, we observe no specific feature depending on departure time. The effect of holiday flags is also limited in our experiments. This is probably because there are not so many holidays (except Saturday and Sunday) in data that are enough to affect the results of prediction.

When the value of RMSE is small, we observe that the LSTMX often predicts timing of changes even if the number of remaining tickets does not match. For instance, in NH215 it does not well predict the number of tickets in October and November, while some coincidences are observed for timing of increase/decrease. A similar observation is done in AF279 (Fig. 7).

Different airlines on the same route often have different dynamics. Fig. 8 shows the results of business class of two airlines on the same route in the same period. As observed in the figure, fluctuations in two flights are quite different—there is large amplitude in October–November while small amplitude in December in AF272. In NH215, on the other hand, there is relatively smaller fluctuation in November while some big fluctuation in December. The LSTMX captures these different characteristics due to the existence of an extrinsic variable representing airlines.

There are cases where the prediction fails significantly (Fig. 9). One reason for this is that the same flight has different fluctuation in the same period (Fig. 10). In such cases, it is hard to predict dynamics of change using two years' data. To predict fluctuation more accurately, longer-term training data of several years would be needed.

V. Conclusion

In this study, we attempt to predict the number of remaining airline tickets in different booking classes. We use real data of 15 months as training data and compare prediction results of 3 months using two different learning models. Experimental results show that the LSTMX has smaller RMSE than the SARIMAX in general, and also captures dynamics in short periods better than the SARIMAX. The LSTMX also successfully predicts weekly/monthly changes and distinguishes different airlines on the same route. Experimental results also suggest that the prediction accuracy depends on flights and booking classes. We used data from 2018 to 2019 but could not use data of 2020 due to the COVID-19 pandemic. To improve the prediction accuracy, it is necessary to increase the amount of data over several years.

In [12] the authors argue whether newly developed deep neural network based algorithms are superior to traditional algorithms for forecasting time series data. They then show that an LSTM based algorithm improved the prediction by 85%

TABLE IV: RMSE for each booking class of different flights in Tokyo–Paris (bold letters show the lowest values)

AF279		Business class					Economy class				
model	size	C	D	J	I	Z	O	W	S	Y	B
LSTMX	14	3.33	3.62	3.24	3.85	4.19	**2.79**	3.78	3.76	1.53	1.61
	28	**3.30**	**3.48**	**3.21**	3.79	**3.97**	3.04	3.80	3.83	1.49	**1.51**
	56	**3.30**	3.49	**3.21**	3.75	4.06	2.88	**3.77**	**3.73**	1.48	1.54
	84	3.38	3.59	3.24	3.83	4.16	3.07	3.94	3.95	**1.46**	**1.51**
SARIMAX	–	4.40	4.77	4.84	**3.29**	4.36	3.30	5.32	5.28	3.48	4.50
JL45		Business class					Economy class				
model	size	C	D	J	X	I	Y	B	H	K	M
LSTMX	14	4.50	4.51	4.50	3.64	3.50	**4.26**	4.27	**4.27**	4.30	**4.20**
	28	4.36	4.44	4.32	3.70	**3.26**	4.30	**4.26**	**4.27**	**4.29**	4.32
	56	**4.31**	**4.33**	**4.30**	3.52	3.33	4.41	4.48	4.52	4.47	4.48
	84	4.47	4.57	4.38	3.72	3.43	4.37	4.43	4.37	4.48	4.49
SARIMAX	–	4.99	4.44	5.02	**3.36**	3.25	5.49	6.37	5.77	6.06	5.35
NH215		Business class					Economy class				
model	size	C	D	J	P	Z	Y	E	N	B	M
LSTMX	14	4.26	**4.26**	3.19	**3.02**	**3.32**	1.85	3.59	2.74	**2.67**	**2.70**
	28	4.22	4.43	3.35	3.14	3.51	1.87	3.62	2.84	2.79	2.81
	56	**4.13**	4.29	3.29	3.10	3.50	**1.79**	**3.58**	2.86	2.75	2.86
	84	4.20	**4.26**	3.30	3.36	3.66	1.85	3.64	**2.71**	2.77	2.82
SARIMAX	–	5.60	5.33	**2.46**	4.25	3.76	6.94	5.39	3.01	6.78	6.72

TABLE V: The number of booking classes classified by the range of RMSE

		Business class			Economy class		
model	size	≤ 3	≤ 4	≤ 5	≤ 3	≤ 4	≤ 5
LSTMX	14	38	46	38	24	32	42
	28	37	51	34	22	35	44
	56	38	48	38	24	34	42
	84	37	48	41	22	33	45
SARIMAX	–	38	30	26	18	30	39

on average compared to the ARIMA model using financial data. The current study provides yet another verification that an LSTM based algorithm works well against the SARIMAX under some circumstances.

The proposed model has a room for further extension. Several studies extend LSTM by taking into account the seasonality, for instance, building a specialized LSTM model for each season [1] or combining LSTM and SARIMA [13]. The LSTMX could be extended by incorporating those techniques. The proposed model has potential applications for predicting tickets sales for sports events, hotel reservations, train tickets, etc. In case of sports events, match or start time is handled as extrinsic variables. In case of hotel reservations, location or events are used variables to distinguish features. Moreover, it could be applied for managing stock products. These extensions and applications are left for future study.

Acknowledgment

We thank the STWorld, Inc. for providing data of airline ticket sales.

References

[1] H. Abbasimehr and M. K. Nahari, "Improving demand forecasting with LSTM by taking into account the seasonality of data", Journal of Applied Research on Industrial Engineering 7(2): 177–189, 2020.

[2] T. Agostino and I. L. Amerise, "Short-term load forecasting using a two-stage sarimax model", Energy 133: 108–114, 2017.

[3] J. A. Abdella, N. Zaki, K. Shuaib, and F. Khan, "Airline ticket price and demand prediction: a survey", Journal of King Saud University - Computer and Information Sciences, 2019.

[4] A. Mostafaeipour, A. Goli, and M. Qolipour, "Prediction of air travel demand using a hybrid artificial neural network (ANN) with Bat and Firefly algorithms: a case study", Journal of Supercomputing 74(2):5461–5484, 2018.

[5] G. E. P. Box and G. M. Jenkins, Time-series analysis:. forecasting and control, Holden-Day, San Francisco, 1970.

[6] E. Diego, "Estimating dynamic demand for airlines", Economics Letters 124(1): 26–29, 2014.

[7] T. Haneda, K. Nakama, T. Okamoto, S. Koakutsu, T. Shimobara, and T. Ito, "Long Short-Term Memory model for time series prediction problem including non-time series data" (in Japanese), Proc. Annual Meeting on Electronics, Information and Systems, IEEJ Japan, 2019.

[8] H.-C. Huang, "A hybrid neural network prediction model of air ticket sales", TELKOMNIKA Indonesian Journal of Electrical Engineering 11:6413–6419, 2013.

[9] A. Krämer, M. Frisen, and T. Shelton, "Are airline passengers ready for personalized dynamic pricing? A study of German consumers", Journal of Revenue and Pricing Management 17:115–120, 2018.

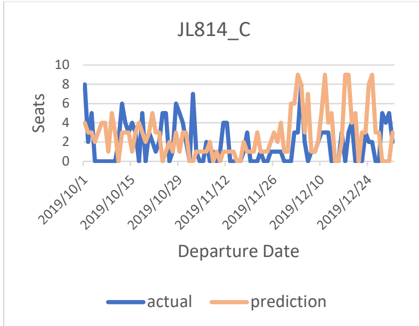
[10] G. Lai, W.-C. Chang, Y. Yang, and H. Liu, "Modeling long- and short-term temporal patterns with deep neural networks", Proc. 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, pp. 95–104, ACM, 2018.

[11] B. Pan, D. Yuan, W. Sun, C. Liang, and D. Li, "A novel LSTM-based daily airline demand forecasting method using vertical and horizontal time series", Pacific-Asia Conference on Knowledge Discovery and Data Mining, LNAI 11154, pp. 168-173, Springer, 2018.

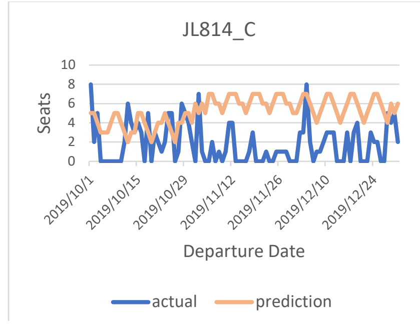
[12] S. Siami-Namini, N. Tavakoli, and A. S. Namin, "A comparison of ARIMA and LSTM in forecasting time series", Proc. 17th IEEE International Conference on Machine Learning and Applications, pp. 1394–1401, IEEE, 2018.

[13] D. C. W. Wu, L. Ji, K. He, and K. F. G. Tso, "Forecasting tourist daily arrivals with a hybrid Sarima-Lstm approach", Journal of hospitality & tourism research 45(1): 52–67, 2021.

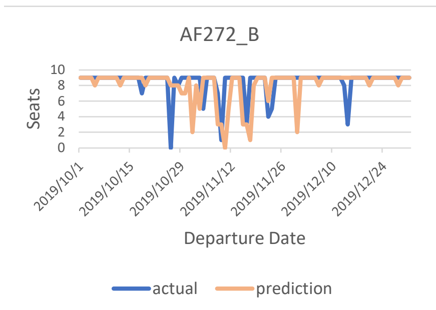
[14] H. Yuan, W. Xu, and C. Yang, "A user behavior-based ticket sales prediction using data mining tools: an empirical study in an OTA company", Proc. 11th International Conference on Service Systems and Service Management, 2014.



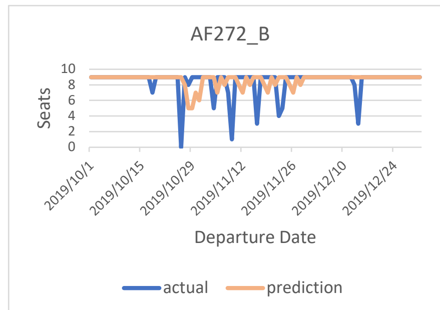
(a) LSTMX



(b) SARIMAX



(a) LSTMX



(b) SARIMAX

Fig. 4: Prediction Results (upper: JL814 Osaka-Taipei, Business Class; booking class=C. lower: AF272 Tokyo-Paris, Economy Class; booking class=B)

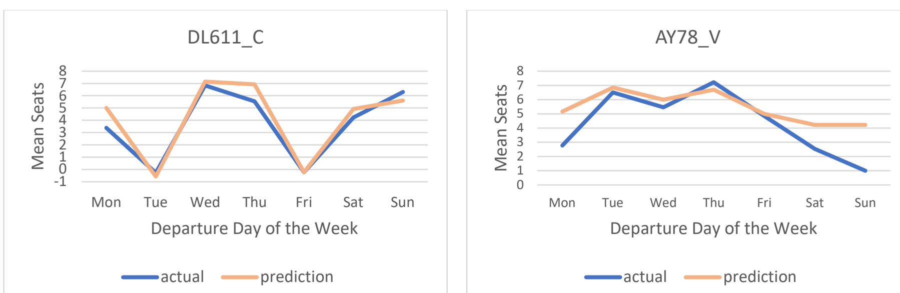


Fig. 5: Mean values of prediction by the day of the week (DL611: Nagoya-Honolulu, Business; AY78: Osaka-Helsinki, Economy)

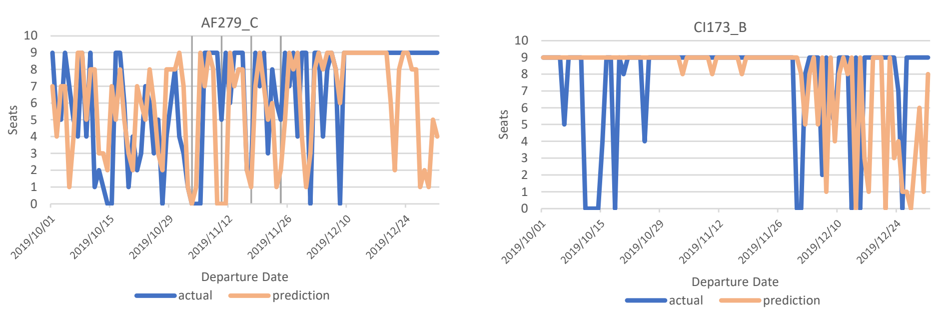


Fig. 6: Different fluctuations of months (AF279: Tokyo-Paris, Business; CI173: Osaka-Taipei, Economy)

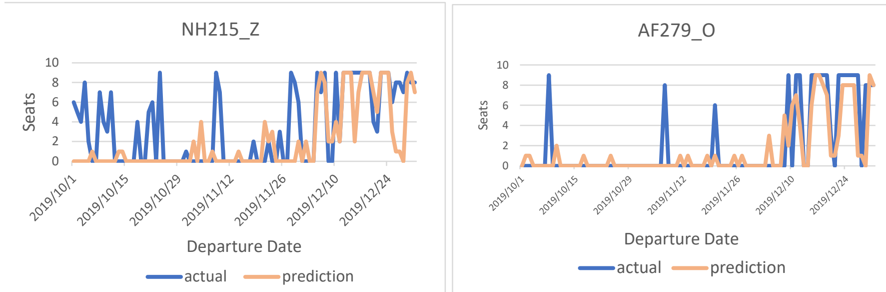


Fig. 7: Match on Timing of Changes (NH215: Tokyo-Paris, Business; AF 279: Tokyo-Paris, Economy)

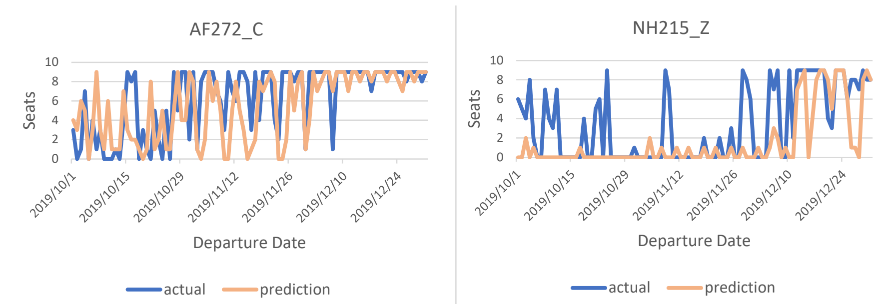


Fig. 8: Fluctuations on the same route by different airlines (AF 272/NH215: Tokyo-Paris, Business)

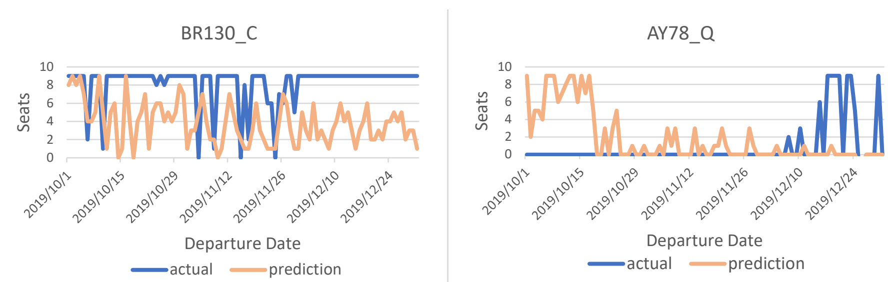


Fig. 9: Failure of Prediction (BR130: Osaka-Taipei, Economy; AY78: Osaka-Taipei, Business)

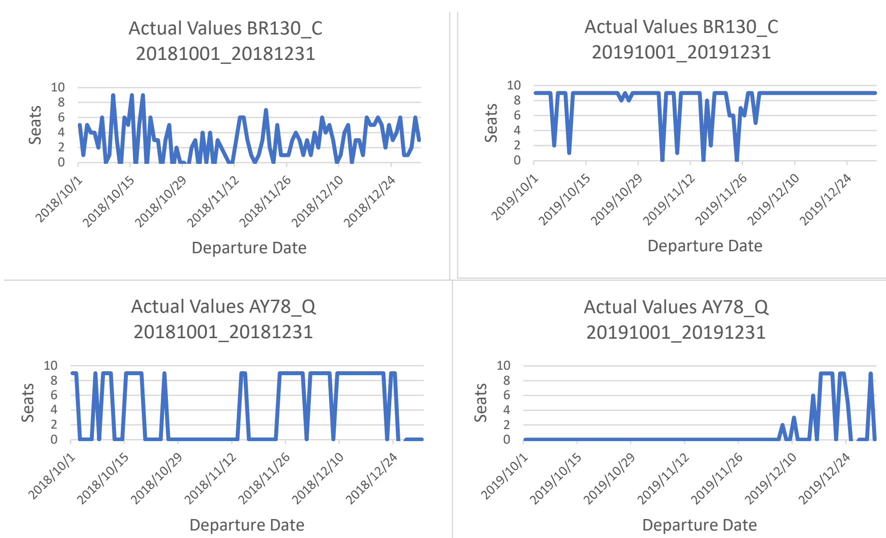


Fig. 10: Different fluctuations in the same period (BR130 and AY78) left: from 1/10/2018 to 31/12/2018; right: from 1/10/2019 to 31/12/2019)