İSTANBUL UNIVERSITY CERRAHPAŞA

ORIGINAL ARTICLE
Electrica 2024

Comparison of Time Series Forecasting for Intelligent Transportation Systems in Digital Twins

Mehmet Ali Ertürk

Department of Computer Engineering, Faculty of Computer and Information Technologies, İstanbul University, İstanbul, Türkiye

Cite this article as: M. A. Ertürk, "Comparison of time series forecasting for intelligent transportation systems in digital twins," *Electrica*, Published online May 13, 2024. doi:10.5152/electrica.2024.23200.

ABSTRACT

The rapid development in transportation systems has created the need to renew transportation with scientific calculation methods and techniques. In this context, the current methods used in public transit need to be improved and have the ability to provide instantaneous responses. In public transportation applications, the primary data consists of passengers transported based on a time series, and although it follows a specific pattern, this pattern can vary. This study evaluates the time series analysis methods using Long Short-Term Memory (LSTM), Autoregressive Integrated Moving Average (ARIMA), and Facebook Prophet for forecasting in public transit with a Digital Twin (DT). The DT model enables instant data delivery and analysis for time-specific events. A metro line in Istanbul city is selected to evaluate the model. The performance of the time prediction algorithms is evaluated by the Mean Absolute Percentage Error, Mean Absolute Error, and Root Mean Squared Error metrics. The results show that both algorithms have almost identical performance; however, Facebook's Prophet is slightly better at every horizon. Long short-term memory can be used in more complex scenarios since training the network requires more resources. Autoregressive Integrated Moving Average and Facebook Prophet can be utilized for single-variable time-series prediction with short horizons.

***Index Terms*—**Autoregressive Integrated Moving Average (ARIMA), Time-series forecast, long short-term memory (LSTM), Facebook prophet, ITS, Digital Twin

I. INTRODUCTION

Transportation is a crucial issue in metropolitan cities, affecting our daily lives. Building transportation infrastructures can be expensive, and their efficiency depends on how we use them. Intelligent transportation systems can help us optimize transportation and make our lives easier. With the help of these applications, we can achieve maximum efficiency while using transportation.

Several challenges need to be addressed for smart transportation systems. These include safety against cybersecurity threats, tuning vehicle functions, predictive maintenance for vehicles and traffic infrastructures, effectively managing vehicle malfunctions, planning disaster scenarios, handling secure communication between vehicle infrastructures, and managing the mobility of vehicles and pedestrians.

These issues can be addressed by renewing classic traffic and transportation management centers with next-generation technologies such as Digital Twins (DT) and machine learning (ML) approaches for public safety [1]. A DT is a virtual representation of a system, process, or any physical entity to make real-time analysis. The essential components of DT are the Internet of Things (IoT) and ML. Machine learning approach can detect hidden patterns in data and predict future events depending on the historical data. Time series data can be analyzed to predict future actions in transportation systems. This study evaluates time series prediction algorithms with a transit data set obtained from Istanbul Metropolitan Municipality. The three different approaches, Long-Short Term Memory (LSTM) neural networks, Facebook's Prophet, and Autoregressive Integrated Moving Average (ARIMA) models, are evaluated for public transportation passenger flow prediction.

The study's main contribution is to evaluate LSTM, Facebook Prophet, and ARIMA models for Public Intelligent Transportation Systems using a Digital Twin architecture. Section II discusses related work about time series data and ITS use cases; Section III presents the time series models

Corresponding author:
Mehmet Ali Ertürk

E-mail:
mehmetali.erturk@istanbul.edu.tr

Received: December 26, 2023

Revision Requested: January 5, 2024

Last Revision Received: January 15, 2024

Accepted: January 31, 2024

Publication Date: May 13, 2024

DOI: 10.5152/electrica.2024.23200

and the proposed Digital Twin approach. Section IV discusses the findings of this study, and Section V concludes the paper.

II. RELATED WORK

Artificial intelligence and LSTM have been used in forecasting with time series data; several studies focus on LSTM and LSTM driver methods to predict future time series.

A trajectory-based approach for forecasting in transportation has been studied to select road links [2]. In the study, authors presented data pre-processing to convert raw Global Positioning System trajectories to form a traffic matrix to propose a graph-based convolutional neural network (CNN). In the study, a hybrid deep neural network is proposed to obtain spatial and temporal relations in traffic data, and experiments are carried out on the TRANSFOR 2019 dataset. Authors show that the presented hybrid method has better performance when compared to LSTM in time horizons from 5 minutes to 4 hours with multi-step predictions.

The spatial and temporal efficiency of the LSTM presented by [3] for dynamic traffic prediction scenarios in transportation. The authors focus on short-term traffic prediction by using data from Eastern Freeway, Melbourne, Australia, with eight upstream and downstream detectors. The study compares the performance of the LSTM with alternative methods of General Regression Neural Networks, Deep Learning Backpropagation neural networks, Modular Neural Networks, Radial Basis Function Networks (RBFNs), and Recurrent Neural Networks (RNNs). It is shown that the overall performance of the LSTM outperforms other methods.

Long short-term memory networks can be utilized to predict different types of data. The research shows they can be employed to create a model for air pollution prediction [4]. The study used 17 attributes that are used between 2012 and 2017 by the Taiwan Environmental Protection Agency. The results are evaluated through MAE, Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE).

In [5], the authors use electrical loads with temperature, humidity, and wind speed to evaluate the efficiency of the LSTM networks. In the study, classical methods were compared with deep learning approaches with different horizons such as 1 day, 2 days, 7 days, and 30 days, and MAPE concerns 1.55, 2.16, 5.97, and 9.75. The study demonstrates that LSTM outperforms other approaches with MAPE and RMSE.

Facebook's Prophet was initially developed with R language for time series forecasting and ported to Python to analyze and predict future events. It can deal with seasonality in the time series data and is used in different fields such as cloud resource predictions[6], sales prediction in supermarkets [7], or high-accuracy air temperature forecasting [8].

Seasonal ARIMA (SARIMA) and Facebook Prophet models are compared and evaluated with time series accident data containing seasonal patterns [9]. The authors indicate that SARIMA has better performance in traffic accidents than Facebook Prophet.

In [10], failure predictions in a fleet of vehicles are presented with different sets of algorithms, including Facebook's Prophet library. The study aims to propose intelligent maintenance methods based on different algorithms such as Facebook Prophet, Simple Feed-forward, and Nbeats. The feed-forward approach predicts 90% of failures.

Autoregressive Integrated Moving Average technique is used in time series prediction [11]. In the study, authors compare Arima, Guru, and LSTM for time series prediction of Bitcoin prices. According to the study, Arima has better performance when compared to other methods with MAPE 2.76% and RMSE 302.53.

In [12], ARIMA is used to predict road traffic congestion for non-stationary, which means the data series is not uniform and includes seasonality. The results show that ARIMA's performance is measured with a 0.1263 MASE accuracy metric within a 95% confidence interval.

In [13], the Kaggle web traffic dataset is used to evaluate the performance of ARIMA and LSTM. The study illustrates that LSTM is slightly better than ARIMA. However, deep learning networks like LSTM can be complex for smaller time series data, and ARIMA is more appropriate for such tasks.

In the [14] study, the authors aim to propose a hybrid model that combines both non-linear and linear models for time-series forecasting. For linear modeling, ARIMA is used, and the non-linear part is implemented using ANN. It is outlined that the hybrid model has better performance when compared to single approaches.

III. MODELS

This study presents three different time series analysis methods with a digital twin architecture. The first part presents the time-series analysis and forecasting methods, and the second part illustrates the digital twin concept.

A. Time Series Analysis

Time-series forecasting requires attention in intelligent transportation systems where things are highly tied to a series of events with time resolution. To understand time series data, this section will introduce time-processing tasks.

A stationary time series data is in the form of statistical equilibrium with statistical properties that change over time, meaning data has a constant mean and variance over time [15]. Nonstationary data contains a seasonality effect where data will turn over in specific periods, such as on weekends.

1) Facebook Prophet

Facebook's Prophet is a library that forecasts time series data fields [16]. The algorithm uses decomposable series which has elements of trend, seasonality, and holidays.

$$y(t) = g(t) + s(t) + h(t) + \varepsilon_t \quad (1)$$

Where $g(t)$ represents the trend function, $s(t)$ is a periodic function, and $h(t)$ represents holidays, and ε is errors. Growth is modeled using the logistic growth model [16]. The Prophet library was initially developed in R statistical language and ported to Python. The library automatically handles the model fitting on non-stationary data. It also provides metrics to measure the forecast performance.

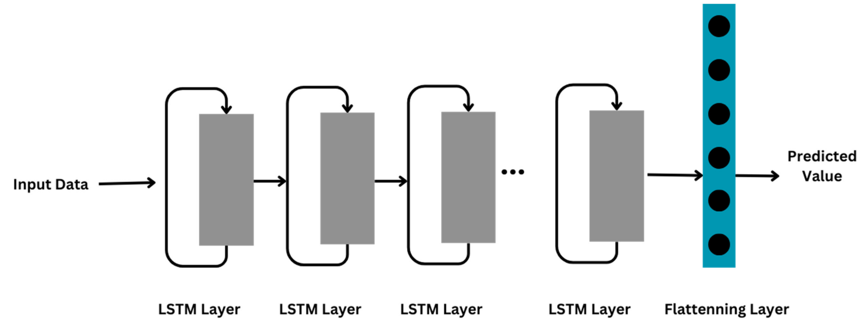

Fig. 1. LSTM architecture [17]. LSTM, long short-term memory.

2) Long Short-Term Memory

Long short-term memory is a RNN with considerable language and machine translation modeling performance. The generic architecture of an LSTM network is given in Fig. 1. Long short-term memory layers take input from the previous layer, which enables accounting for short historical data on the learning fields[17]. Long short-term memory employs a nonlinear learning approach to solve complex problems.

3) Autoregressive Integrated Moving Average

Autoregressive Integrated Moving Average linear stationary model and its general form is given in Eq. 4 by [15]:

$$\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \ldots \phi_p B^p \tag{2}$$

$$\theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \ldots \theta_q B^q \tag{3}$$

$$\varphi(B) z_t = \phi(B) \nabla^d z_t = \theta_0 + \theta(B) a_t \tag{4}$$

where $\phi(B)$ is autoregressive operator and assumed to be stationary, $\varphi(B) z_t = \phi(B) \nabla^d z_t$ is generalized operator and nonstationary, and $\theta(B)$ is a moving average. In the case $d = 0$, the model is stationary but $\phi(B)$ but $\theta(B)$ is not in the same order. Examples are given for p and q are given as $p = 1, 2$ and $q = 1, 2$.

B. The Digital Twin Architecture

The digital twin technology enables real-time what-if analysis by representing physical entities, buildings, systems, or processes through two-way communications. The infrastructure highly relies on the IoT and Artificial intelligence (AI) techniques. A digital twin system basically has three fundamental layers: data, virtualization, and service. A digital twin can be a physical system as well as a process.

Fig. 2 overviews a typical DT technology. The physical layer can be a building, system, human, vehicle, or process, and two-way communication is provided by technologies such as 6G. Machine learning (ML) and other AI-based strategies are used to handle large data tasks and make analyses.

In this study, a digital twin system is designed to represent an ITS railway passenger flow as a digital process. This system tracks the status of passenger loads in real-time and performs time-series analysis. The analysis provides operators with additional information to decide whether to extend vehicle capacity or add additional trains for a given period.

To simulate physical processes, a real dataset is utilized with the help of a Python client library. This application generates passenger flows for each hour using the Poisson distribution, which helps simulate a real system for the digital twin. The generated data is then passed into the digital twin system, which processes the data within the data layer. The virtualization layer represents the virtual state of the transition line, whereas the service layer enables time series analysis in a real-time fashion.

A generic overview of the designed model is given in Fig. 3. In this work, a three-layer model is selected and real-time data generation in the figure is simulating a real physical process for evolution purposes.

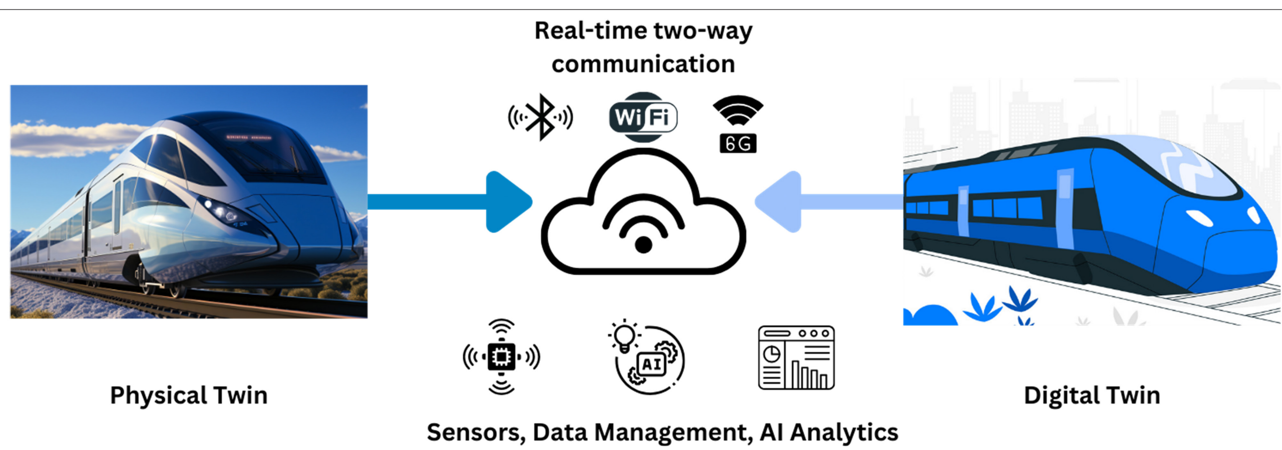

Fig. 2. Digital twin architecture.

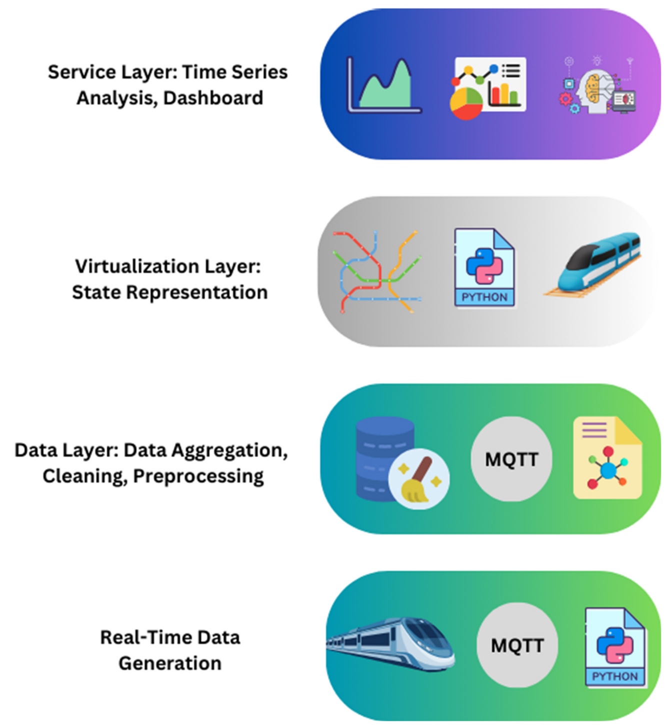

Fig. 3. Architecture of presented model.

1) Data Layer

The data layer consolidates data from different sources; it can connect various sources, such as HTTP Rest API, to MQTT Brokers. The data layer handles necessary tasks on the data, such as detecting duplicates, removing missing values, and cleaning data. Various tasks can be handled on the data layer; for example, the data can be represented as Graphs to perform graph-based signal processing to deal with missing data [18].

As mentioned, a Python client library is implemented to mimic physical processes to deliver data from a data source in a realistic scenario that pushes data for each event to the data layer. The experimentation details and the example data set are presented in Section 4.

The data layers handle data passed from different sources with active preprocessing and aggregation. In this study, a railway line has several stations, each with its own passenger flow pattern. In this design, a uniform distribution of the passengers is evaluated with passion arrivals. In brief, each station generates equal passenger flow events with a Poisson distribution in hourly slots. The design overview of the presented data layer is shown in Fig. 4.

2) Virtualization Layer

The virtualization layer is responsible for creating a virtual representation of the physical twin and its functions in a DT system. This layer responds to events to replicate the physical twin. In this study, the metro transit process was replicated and virtualized in this layer.

Fig. 4. Data layer.

Fig. 5. Virtualization layer.

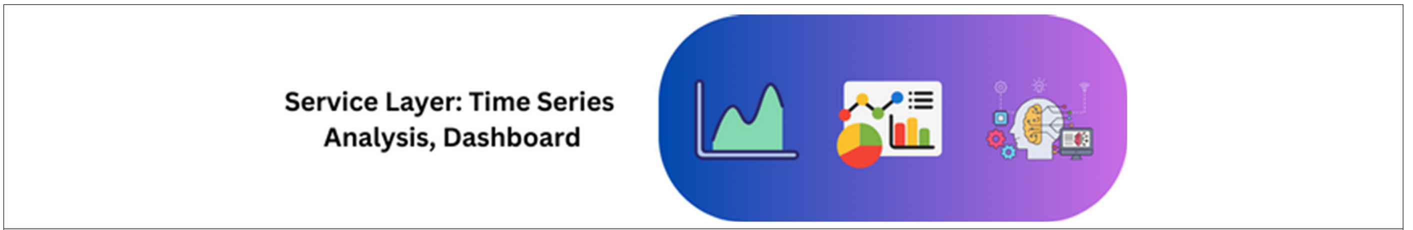

Fig. 6. Service layer.

The virtual state of the physical twin is emulated in this layer. All properties and statuses of the physical process are digitally represented. This layer emulates the transition line process as DT using Python (as shown in Fig. 5). Additionally, this layer can also simulate the DT system's status.

3) Service Layer

The last layer is service, it is the place where operators interact with the system. It enables users to perform analysis tasks and what-if simulation analysis. This layer controls the algorithmic perspective and different methods and techniques can be deployed for system evaluation. Analysis and algorithmic perspective of the DT reside in the service layer.

In this study, three (Facebook Prophet, LSTM, and ARIMA) different time-series prediction algorithms and their configurations are implemented in the service layer, and with its modular structure, any additional algorithm can be adopted without extra effort. Also, this layer has a dashboard where all system-related configurations, monitoring, and management are performed.

In Fig. 6, service layer is described briefly. It contains a dashboard with time series analysis tools and services.

Algorithm 1: Continuous model training

$initialData \leftarrow historicalData;$
$splitDate \leftarrow initialTrainTestSplitPoint;$
$hasNewData \leftarrow True;$
while $hasNewData$ **do**
  $dataTrain \leftarrow initialData < splitDate;$
  $dataTest \leftarrow initialData > splitDate;$
  $modelTrain(dataTrain);$
  $error \leftarrow modelTest(dataTest);$
  if $newDataRecieved$ **then**
    $initialData \leftarrow initialData + newData;$
    $hasNewData \leftarrow True;$
  else
    $hasNewData \leftarrow False;$

Fig. 7. Continuous model training algorithm.

One of the main challenges with time series data is that models become outdated over time. To address this issue, an algorithm has been developed (as shown in Fig. 7) that updates the model with every new event triggered in the DT system. During the model training phase, which runs in the background, the old model is gradually replaced with the new one. This ensures that the model stays up-to-date and accurate over time.

IV. EXPERIMENTATION AND RESULTS

A DT system is developed with three layers described in the previous section to validate three different time series frameworks. For evaluation purposes, actual transit data from the Istanbul Metropolitan Municipality Open Data Portal is used to test the algorithms. Any different transit scenario can be implemented; however, for evaluation "GEBZE-HALKALI" metro line is selected [19]. The metro line and the passenger flow are the process virtually represented in this DT. Actual metro line and stations are given in Fig. 8.

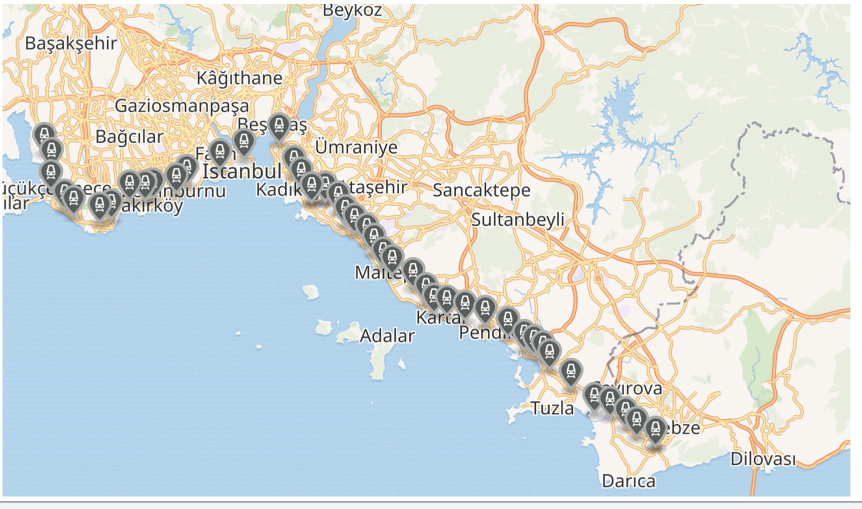

Fig. 8. GEBZE-HALKALI metro line.

TABLE I. DATASET INFORMATION

Attribute	Type	Description
transition_date	Text	Holds transition date of the transportation vehicle
transition_hour	Text	Hour value of the transition
transport_type_id	Text	1: highway 2: rail 3: marine transportation
road_type	Text	Transportation area
transfer_type	Text	Passenger transition type: transfer or normal
number_of_passage	Text	Count of number of trips per hour
number_of_passanger	Text	Individual passenger using vehicle given hour

A. Data Set

The Data Set is the Hourly Public Transport Data Set of Istanbul City, which is obtained from IMM Open Data Portal and it contains all transportation data from May 2023 to September 2023 [20]. The filtered data has 11 315 rows. The description of the data set is given in Table I. Fig. 9 previews the raw data obtained from the source.

The dataset is divided into two parts: training and test datasets. The training dataset contains 82% of the entire dataset, and the remaining part is included in the test dataset for future performance analysis. Performance metrics are presented using the test dataset, which was not used in the training process. The data split point for the two datasets is September 10, 2023.

The data is preprocessed before training to remove outliers such as holidays and weekends. The study focuses only on rush hours for commuting in Istanbul at 07:00 AM, 08:00 AM, and 09:00 AM.

Fig. 10 shows the daily passenger flow count from May 2023 to September 2023. The study aims to predict the following days for passenger flow that could help public transportation management be more efficient.

The study on time series analysis is being evaluated using three models—Facebook Prophet (FB Prophet), LSTM, and ARIMA. The FB Prophet library is being used with standard options and no additional configurations. Autoregressive Integrated Moving Average (1,1,0) is being used for the p and q values, which were described in the previous sections. The LSTM network is being designed as follows:

The attribute forecasting is normalized before the training and testing stages. The neural network has two hidden layers and accepts one input parameter. It is configured to use mean squared error for regression and Adam optimizer. The training is done with a learning rate of 0.01 over 2000 epochs.

For performance comparison, MAPE, MAE, RMSE metrics are used with different horizons. Table II compares LSTM, ARIMA, and Facebook Prophet with these metrics. Based on the findings, all the algorithms perform similarly in the same forecast horizon. However, in all horizons, Facebook's Prophet library shows slightly better performance than LSTM and ARIMA. Prophet uses a linear approach, which has superior computation performance compared to others and can be used for single-value time forecasting tasks. On the other hand, LSTM networks use a deep non-linear approach, making LSTM a better alternative for multi-parameter complex scenarios.

In addition, DT systems rely on real-time data processing and analytics. For this reason, Facebook Prophet is more suitable for DT forecasting scenarios.

Also, the actual data points and predictions are given in Fig. 11, and the red line crosses the splitting point where train and test data are separated into two datasets. The actual data and the forecasting points are given with a higher resolution and are shown in Fig. 12.

	transition_date	transition_hour	transport_type_id	road_type	line	transfer_type	number_of_passage	number_of_passenger	transition_time
23825	2023-05-02	7	2	RAYLI	HALKALI - GEBZE	Normal	43458	43090	2023-05-02 07:00:00
25146	2023-05-02	8	2	RAYLI	HALKALI - GEBZE	Normal	49355	48741	2023-05-02 08:00:00
27317	2023-05-02	9	2	RAYLI	HALKALI - GEBZE	Normal	32003	31099	2023-05-02 09:00:00
47102	2023-05-03	7	2	RAYLI	HALKALI - GEBZE	Normal	42754	42386	2023-05-03 07:00:00
48714	2023-05-03	8	2	RAYLI	HALKALI - GEBZE	Normal	50637	50024	2023-05-03 08:00:00
...	...	...	...	...	...	...	...	...	...
4288238	2023-08-29	8	2	RAYLI	HALKALI - GEBZE	Normal	13265	12786	2023-08-29 08:00:00
4291125	2023-08-10	7	2	RAYLI	HALKALI - GEBZE	Normal	11443	11437	2023-08-10 07:00:00
4291887	2023-08-31	9	2	RAYLI	HALKALI - GEBZE	Normal	1000	995	2023-08-31 09:00:00
4293949	2023-08-22	8	2	RAYLI	HALKALI - GEBZE	Normal	82	80	2023-08-22 08:00:00
4295317	2023-08-31	7	2	RAYLI	HALKALI - GEBZE	Normal	23	23	2023-08-31 07:00:00

Fig. 9. Data preview: GEBZE-HALKALI.

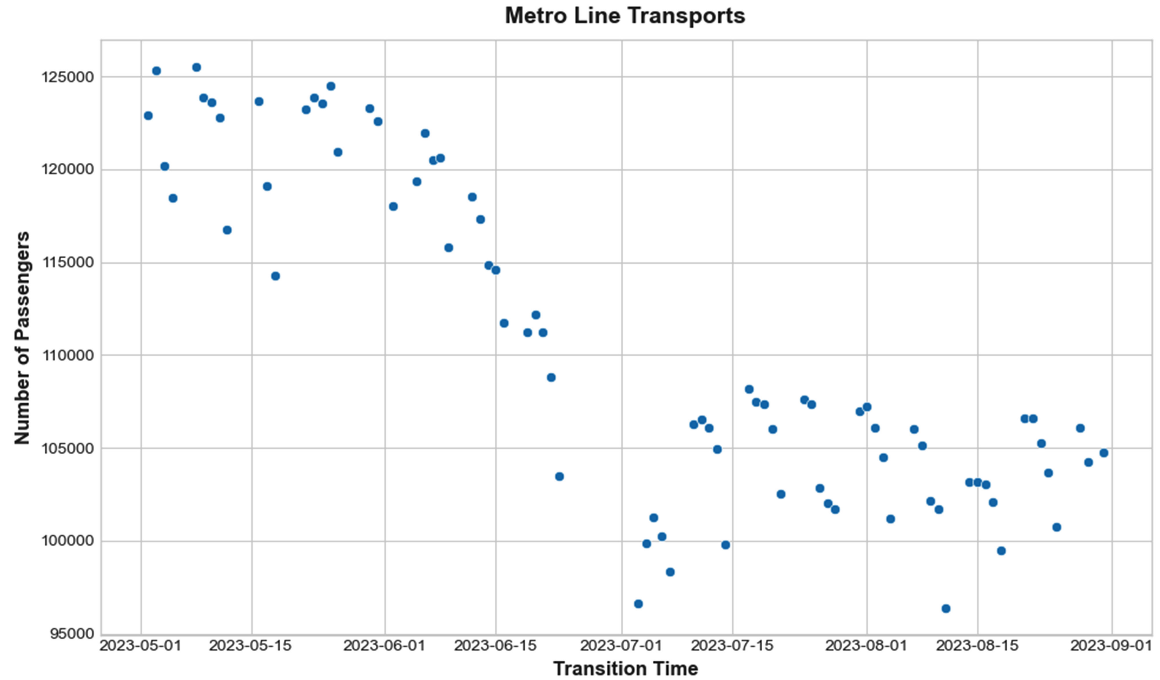

Fig. 10. Daily passenger flow in metro line: GEBZE- HALKALI.

TABLE II. PERFORMANCE COMPARISON

	FB Prophet			LSTM			ARIMA		
Horizon	**MAPE**	**MAE**	**RMSE**	**MAPE**	**MAE**	**RMSE**	**MAPE**	**MAE**	**RMSE**
1 day	0.0226	2354.76	2354.76	0.024	2638.84	3337.40	0.027	2659.49	3759.12
5 days	0.032	3364.84	3518.94	0.029	3081	3726	0.019	1921.67	2632.24
15 days	0.01940	1985.61	2438.27	0.021	2226.34	2844.27	0.0214	2165.04	2952.66

ARIMA, autoregressive integrated moving average; FB, Facebook Prophet; LSTM, long short-term memory; MAE, mean absolute error; MAPE, mean absolute percentage error; RMSE, root mean squared error.

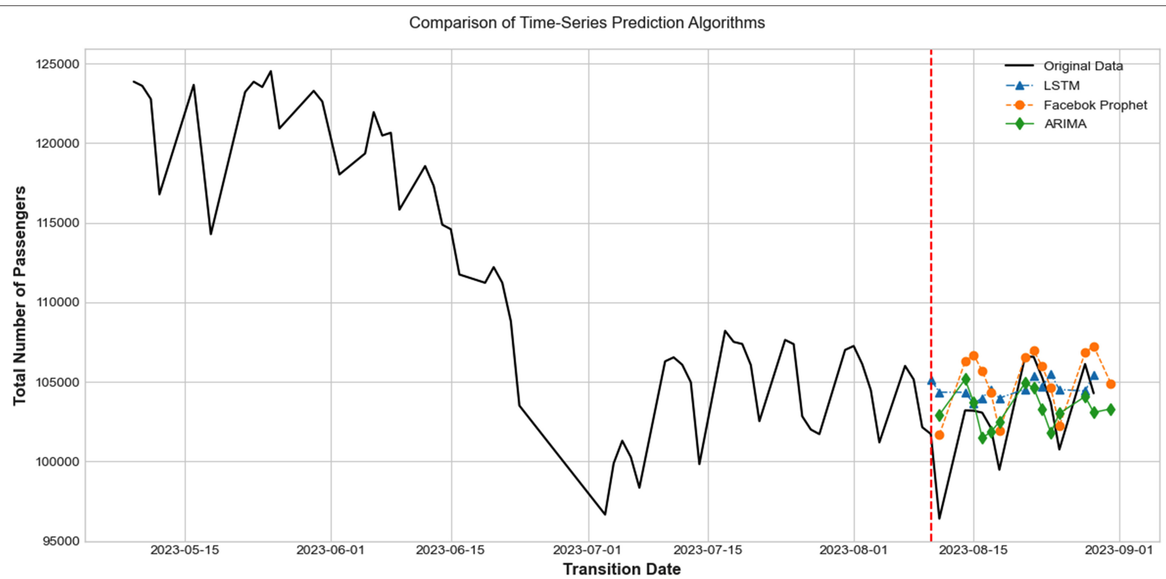

Fig. 11. Comparison of predictions LSTM, Facebook Prophet, and ARIMA. ARIMA, Autoregressive Integrated Moving Average; LSTM, long short-term memory.

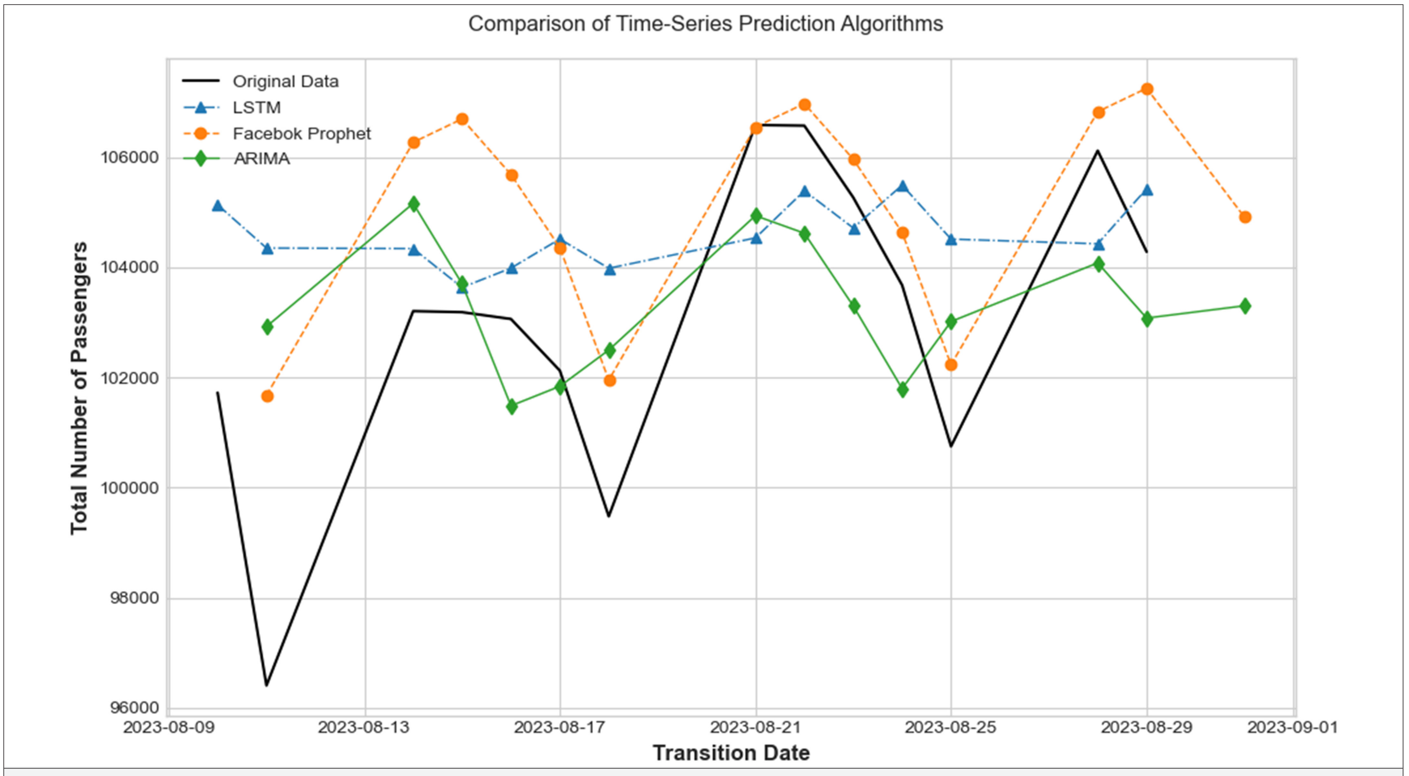

Fig. 12. Passenger count prediction.

V. CONCLUSION AND FUTURE WORK

Public transportation systems can be renowned by DT systems to increase efficiency with real-time management strategies. This study presents a DT approach to improve public transportation infrastructures by predicting passenger flow using a metro transit dataset. Additionally, the DT system is evaluated with time series analysis methods using LSTM, ARIMA, and Facebook Prophet for forecasting passengers using the metro within rush hours of a metropolitan city. The obtained results show that Facebook Prophet slightly outperforms LSTM and ARIMA. Thus, it can be used for forecasting depending on a single variable. Long short-term memory can be used in more complicated scenarios since training the network requires more resources. As a future work, it is planned to design a time-series prediction model with multiple variables in a real-time scenario with traffic and vehicular network simulators to benefit DTs for more complex scenarios. Also, as a future direction, time-series forecasting for ITS with federated learning will be studied for privacy concerns.

Peer-review: Externally peer-reviewed.

Declaration of Interests: The author has no conflicts of interest to declare.

Funding: The author declared that this study has received no financial support.

REFERENCES

1. A. Haydari, and Y. Yılmaz, "Deep reinforcement learning for intelligent transportation systems: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 1, pp. 11–32, 2022. [CrossRef]
2. T. Bogaerts, A. D. Masegosa, J. S. Angarita-Zapata, E. Onieva, and P. Hellinckx, "A graph CNN-LSTM neural network for short and long-term traffic forecasting based on trajectory data," *Transp. Res. C*, vol. 112, pp. 62–77, 2020. [CrossRef]
3. R. L. Abduljabbar, H. Dia, P. Tsai, and S. Liyanage, "Short-term traffic forecasting: An LSTM network for spatial-temporal speed prediction," *Future Transp.*, vol. 1, no. 1, pp. 21–37, 2021. [CrossRef]
4. Y.-S. Chang, H. Chiao, S. Abimannan, Y. Huang, Y. Tsai, and K. Lin, "An LSTM-based aggregated model for air pollution forecasting," *Atmos. Pollut. Res.*, vol. 11, no. 8, pp. 1451–1463, 2020. [CrossRef]
5. S. Muzaffar, and A. Afshari, "Short-term load forecasts using LSTM networks," *Energy Procedia*, vol. 158, pp. 2922–2927, 2019. [CrossRef]
6. M. Daraghmeh, A. Agarwal, R. Manzano, and M. Zaman, "Time series forecasting using Facebook prophet for cloud resource management," in 2021 IEEE International Conference on Communications Workshops (ICC Workshops). [CrossRef].
7. B. Kumar Jha, and S. Pande, "Time series forecasting model for supermarket sales using FB-prophet,", in 2021 5th International Conference on Computing Methodologies and Communication (ICCMC). [CrossRef]
8. T. Toharudin, R. S. Pontoh, R. E. Caraka, S. Zahroh, Y. Lee, and R. C. Chen, "Employing long short-term memory and Facebook prophet model in air temperature forecasting," *Commun. Stat. Simul. Comput.*, vol. 52, no. 2, pp. 279–290, 2023. [CrossRef]
9. E. F. Agyemang, J. A. Mensah, E. Ocran, E. Opoku, and E. N. N. Nortey, "Time series based road traffic accidents forecasting via SARIMA and Facebook Prophet model with potential changepoints," *Heliyon*, vol. 9, no. 12, p. e22544, 2023. [CrossRef]
10. C. Foké, J.-P. Kenné, and N. S. B. Diego, "Failure prediction and intelligent maintenance of a transportation Company's urban fleet," *J. Transp. Technol.*, vol. 13, no. 1, pp. 1–17, 2023. [CrossRef]
11. P. T. Yamak, L. Yujian, and P. K. Gadosey, "A comparison between ARIMA, LSTM, and GRU for time series forecasting," in Proceedings of the 2019 2nd International Conference on Algorithms, Computing and Artificial Intelligence. Association for Computing Machinery. [CrossRef]
12. T. Alghamdi, K. Elgazzar, M. Bayoumi, T. Sharaf, and S. Shah, "Forecasting traffic congestion using ARIMA modeling," in 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC). [CrossRef]
13. K. Zhou, W. Y. Wang, T. Hu, and C. H. Wu, "Comparison of time series forecasting based on statistical ARIMA model and LSTM with attention

mechanism," *J. Phys. Conf. S.*, vol. 1631, no. 1, pp. 012141–012141, 2020. [CrossRef]
14. D. S. de O. Santos Júnior, J. F. L. de Oliveira, and P. S. G. de Mattos Neto, An intelligent hybridization of ARIMA with machine learning models for time series forecasting. *Knowl.-Based Syst.*, vol. 175, pp. 72–86, 2019.
15. G. E. P. Box et al., *Time Series Analysis: Forecasting and Control*. Chichester, UK: John Wiley & Sons, 2015.
16. S. J. Taylor, and B. Letham, "Forecasting at Scale," *The American Statistician*, vol. 72, no. 1, 37–45, 2018. [CrossRef]
17. S. M. J. Jalali, S. Ahmadian, A. Kavousi-Fard, A. Khosravi, and S. Nahavandi, "Automated deep CNN-LSTM architecture design for solar irradiance forecasting," *IEEE Trans. Syst. Man Cybern. Syst.*, vol. 52, no. 1, pp. 54–65, 2022. [CrossRef]
18. M. A. Erturk, and L. Vollero, "GSP for virtual sensors in ehealth applications," in 2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC). [CrossRef]
19. M. Istanbul, *Istanbul Metropolitan Municipality Open Data Portal*, 2023.
20. Metropolitan-Municipality, I, *Hourly Public Transport Data Set*. 2023.

Mehmet Ali Erturk received his BS in Computer Engineering from Doğuş University, Türkiye 2007, his MSc (2010) and PhD (2016) in Computer Engineering from İstanbul University, Türkiye. He is currently an Assistant Professor at the Department of Computer Engineering, at Istanbul University. He is interested in researching wireless network protocols, Low Power Wide Area Networks, Artificial Intelligence, and Cyber Security.