Kathmandu University Journal of Science, Engineering and Technology, Vol. 19, No. 1, March 2025

Kathmandu University
Journal of Science, Engineering and Technology

Comparative analysis of time series forecasting for Nepal Airlines passenger data: ARIMA vs. LSTM model

Saishab Bhattarai*[a] and Ishan Panta[b]

[a]Department of Mathematics, Kathmandu University, Dhulikhel, Kavre, Nepal.
[b]Department of Computer Science and Engineering, Kathmandu University, Dhulikhel, Kavre, Nepal.

Abstract

This paper presents a comparative study of time series forecasting methods applied to Nepal Airlines passenger data, focusing on the ARIMA and LSTM models. The study aims to analyze the forecasting performance of these models and identify the most accurate approach for predicting future airline passenger numbers. The ARIMA model captures linear trends and seasonality, while the LSTM neural network is employed for its ability to model complex patterns and non-linear relationships within the data. Both models are evaluated using standard performance metrics, and the results provide insights into the strengths and weaknesses of each forecasting technique. The results indicate that ARIMA provided more accurate forecasts with MAE: 0.74 and RMSE: 1.78, compared to LSTM having MAE: 0.87 and RMSE: 2.02, underscoring its suitability for datasets with linear trends and seasonality.

Keywords: ARIMA; LSTM; Nepal Airlines; Passenger data; Neural networks.

1. Introduction

Time series data forecasting provides a special set of challenges, particularly in sectors like aviation where market dynamics and economic swings are unexpected. Both preserving operational effectiveness and guaranteeing financial stability depend on accurate forecasting. The nation's first and primary airline, Nepal Airlines, is essential to both its economic growth and its ability to connect to the outside world. Initially established as Royal Nepal Airlines Corporation in 1958, it operated with a single Douglas DC-3 Dakota aircraft, it initially served domestic destinations such as Simara, Pokhara, and Biratnagar, along with Indian cities including Delhi and Kolkata. Over the decades, Nepal Airlines expanded its deft force to include turboprop engines and jet aircraft like the Boeing 727 and 757. Despite its early recognition as a key player in the South Asian aviation industry, the airline had a variety of operational challenges, including declining passenger volumes, unstable markets, and loss to its brand by service interruptions and instability in the country.Regression analysis and other traditional forecasting techniques may fail in dynamic and unpredictability. These techniques are effective for simple correlations, but they are unable to identify the complex and non-linear patterns that are frequently found in real-life data. In order to improve forecast accuracy, sophisticated methods such as ARIMA (Autoregressive Integrated Moving Average) and LSTM (Long Short-Term Memory) models have gained popularity.

Traditional methods for forecasting, including regression analysis, have lim- its in an unstable situation. The complicated and non-linear patterns seen in real-world data are frequently missed by these approaches. ARIMA (Autoregressive Integrated Moving Average) takes trends, seasonality, and patterns into organized account, it is especially well-suited for evaluating time series data. ARIMA explicitly incorporates the impact of historical values and relationships over time, in contrast to typical regression models that assume linear correlations and frequently ignore time-based dependencies. This increases its accuracy when working with data that shows cycles or trends. Without manually adding lagged variables, which takes time and can result in overfitting, standard regression models like linear regression are unable to handle such complexities [1].

Additionally, ARIMA makes it easier to deal with stationarity and autocorrelation problems that other regression models are unable to effectively handle. Because of these benefits, ARIMA is a preferable option for modeling the passenger data from Nepal Airlines, which exhibits distinct seasonal and trend patterns. However, LSTM (Long Short-Term Memory), a kind of recurrent neural network, is particularly good at identifying long-term patterns in data and capturing complex non-linear connections.

The purpose of this study is to determine how effectively the ARIMA and LSTM models forecast operational data for Nepal Airlines. Accurate projections are essential for making strategic decisions, streamlining processes, and raising customer satisfaction levels. This study is relevant because international air travel is recovering from the pandemic and Nepal Airlines is working to recover market share. Through the use of rigorous forecasting methods, the study aims to compare the performance of traditional ARIMA model with LSTM model in a highly volatile sector. The findings of this study have significant implications for Nepal Airlines. Accurate forecasting using ARIMA can aid in optimizing flight schedules, managing crew allocation, and improving inventory control for high-demand seasons. Moreover, these insights can inform marketing strategies to maximize passenger engagement during peak periods, thus enhancing operational efficiency and customer satisfaction.

A time series is just a collection of observations arranged chrono-

*Corresponding author. Email: saishab.bhattarai.75@gmail.com

logically. Time series forecasting models forecast demand using mathematical methods derived from data from the past. It is based on the idea that the future is an extension of the past, which is why we estimate future demand using historical data. Demand forecasting using time series analysis has been the subject of numerous studies in a variety of fields.

Gupta et al. [2] introduced an LSTM-based Recurrent Neural Network(RNN) for predicting airline passenger numbers, highlighting its ability to model long-term patterns better than linear statistical models like ARIMA. Using a four-step process data preparation, model building, training, and evaluation the approach achieved MAPEs of 8.79% (training) and 10.8% (testing), outperforming methods like Holt-Winters and basic neural networks.

Pan et al. [3] developed an LSTM-based model for daily airline demand forecasting, leveraging horizontal time series for short-term predictions and vertical time series for longer-term trends. By addressing limitations of traditional methods, the model achieved 89.11% accuracy for horizontal series and outperformed alternatives like Support Vector Regression and Random Forest Regression. This dual-series approach proved effective for improving revenue management in the airline industry.

Andreoni and Postorino [4] employed univariate and multivariate ARIMA models to forecast air transport demand at Reggio Calabria Airport, effectively handling non-stationarity with differencing and logarithmic transformations. The multivariate model included variables such as per capita income and flight movements, which enabled the assessment of the impact of airport policies and fare changes. Both models provided reliable forecasts, with the multivariate approach offering broader insights despite data limitations.

Asrah et al. [5] compared time series forecasting methods for Malaysia Airlines (MAS) and AirAsia, finding that AirAsia's data followed a Geometric Brownian Motion (GBM) process, while MAS data required a Seasonal ARIMA (SARIMA) approach. Models SARIMA(0,0,1)(1,0,0) and SARIMA(2,0,0)(0,1,1) were used for MAS in 2009 and 2012, respectively, achieving MAPE values below 10%. This study highlighted distinct passenger trends tied to the operational differences between the two airlines.

Siami-Namini et al. [6] examined the performance of ARIMA versus LSTM models for time series forecasting, with a focus on financial and economic data. With an average Root Mean Square Error (RMSE) reduction of 84% to 87%, their study demonstrated that LSTM models consistently outperformed ARIMA in error reduction. The authors emphasized that while ARIMA relies on linear assumptions, LSTM's ability to capture non-linear patterns and long-term dependencies contributed to its superior performance. Additionally, they observed that LSTM's performance did not improve with epochs greater than one, suggesting the potential for overfitting in rolling forecast scenarios.

Zhang et al. [7] developed an LSTM-based model for short-term stock price prediction using historical data from the Vanguard Total Stock Market Index Fund (VTI) between 2018 and 2021. The study compared the performance of LSTM against four other models: Linear Regression, eXtreme Gradient Boosting (XGBoost), Moving Average, and Last Value. The results showed that LSTM, with an RMSE of 1.750 and MAPE of 0.633%, did not outperform simpler models like XGBoost (RMSE 1.647, MAPE 0.593%) and Last Value (RMSE 1.689, MAPE 0.598%). These findings highlighted that LSTM's predictive capability was limited in short-term forecasting due to the restricted prediction range and data size.

2. Methodology

The goal of this research is to compare the forecast number of airline passengers based on past data, which consists of 192 observations of monthly passenger counts. The approach followed a systematic process involving data collection, cleaning, and the use of ARIMA and LSTM to extract meaningful trends and patterns. Additionally the section also explains the structures we used for the LSTM model.

2.1. Data preprocessing

The raw dataset used in this study is the Nepal Airline Passenger Dataset, consisting of 192 monthly records of passenger counts from July, 2008 to June, 2024 . This data spans several years and is structured chronologically, with each record representing the number of passengers for a specific month. The dataset is univariate, which focuses on one dependent variable, passenger-count and the time column is used as the independent variable. This data provides a rich context for understanding the seasonal trends and periodic behavior of passenger travel, which are common in transportation industries. However, the raw dataset contains some challenges, such as missing values or irregularities, which are addressed in the data preprocessing step.

Before performing any analysis, it is critical to preprocess the data to ensure accuracy and reliability. The preprocessing steps include the following:

2.1.1. Handling missing data

Any missing values in the dataset were carefully inspected and replaced using interpolation to maintain the temporal consistency of the time series. Missing values in a time series can disrupt the ARIMA model's accuracy, so this step was crucial [8].

2.1.2. Outlier detection and stationarity check

Outliers were identified using box plots and visual inspection of the data. In cases where significant deviations from the general trend were detected, a decision was made to either smooth or adjust these outliers based on domain knowledge from the expertise of the airlines company, and in some cases, by averaging the trend of the data [9].

Outlier detection and Stationarity Checks were chosen as preprocessing criteria because of their direct relevance to the requirements of the ARIMA model and the characteristics of the dataset. ARIMA specifically requires a stationary time series to ensure the reliability of its autoregressive and moving average components. Stationarity, characterized by constant mean and variance over time, is essential for accurate forecasting. Achieving this was a priority, and LogDiffShifting which is a combination of Logarithmic Transformation and Differencing was selected due to its effectiveness in stabilizing variance and removing linear trends.

While other preprocessing techniques, such as Exponential Decay Transformation, are viable alternatives, they are better suited for data with exponential growth patterns, which were not observed in the data of Nepal Airlines. Additionally, ARIMA does not require the normality or homoscedasticity assumptions typically associated with general regression models, so these were not considered. The focus on stationarity and outlier handling aligned with the specific requirements of ARIMA, ensuring the model's suitability for the Nepal Airlines passenger dataset. Future studies could explore alternative methods like Exponential Decay Transformation to evaluate their efficacy in similar contexts.

2.1.3. Data analysis

To gain a better understanding of the data, data visualization of passenger-count over time was done. This visual representation highlights key components in the data, such as trend and seasonality. The trend plot, as shown in Fig. 1 isolates the underlying trend in the data. Over time, there is a clear upward trend, indicating a general increase in passenger counts. The rise between 2014 and 2019 is particularly significant, showing that more people traveled during this period due to some crucial marketing strategies by airlines and buying two airbuses namely Airbus A330-200s [10]. However, after 2019, there's a sharp decline around 2020, which aligns with major global events like the COVID-19 pandemic. Subsequently, a recovery begins after 2021. This plot effectively captures long-term shifts in travel behavior.

Similarly, the seasonality plot, as shown in Fig. 2 illustrates the number of passengers over time, with clear patterns of peaks and valleys that appear at regular intervals. These peaks likely correspond to high-travel seasons, such as holidays and vacation periods. For example, the number of passengers surges in late 2019 and early 2020, which could reflect increased travel during winter holidays. Similarly, there are sharp declines around the end of early 2020 and mid-2021, possibly due to events like the COVID-19 pandemic that impacted travel demand significantly. This pattern reveals a strong seasonal component in travel behavior, with predictable cycles of higher and lower demand throughout the year. Understanding these trends is essential for travel-related industries, as it helps them prepare for busier periods and adjust operations to meet demand.

2.2. ARIMA model

The ARIMA model is a generalized model of Autoregressive Moving Average(ARMA) which combines Autoregressive (AR) process and Moving Average (MA) processess and builds a composite model of time series.The key elements of model are:

- AR: AR is the Autoregression model that uses the dependencies between an observation and a number of lagged observations (p).
- I: Integrated is used to make the time series stationaly by measuring the diffrerences of observations at different time (d).
- MA: Moving Average is an approach that tackels the dependencies between observations and the residual error terms when a moving average model is used to lagged observations (q).

Auto-regressive process (AR): The relationship between an observation and a specific number of lag observations is represented by the AR component. The parameter (p), which indicates how many lag words to include in the mode, defines it. Mathematically, the AR(p) model is defined as [11]:

$$X_t = c + \sum_{i=1}^{p} \phi_i X_{t-i} + \epsilon_t \tag{1}$$

where X_t is the actual value also known as stationary variable at time t, ϕ_i are the auto correlation coefficients for each lagged observation of 1,2,...,(p), and ϵ_t is the residual also known as Gaussian white noise series having mean zero.

Integrated process (I): A time series which is determined by the cumulative effect of an activity belongs to the class of integrated processes.The stationary of the differences series of an integrated process is a critical characteristic from the perspective of statistical analysis. Time series integrated processes serve as a model for non stationary series. The integrated component is responsible for differencing the time series to attain stationary, eliminating any trends or seasonality present in the data. The parameter d indicates the number of differences needed to achieve stationary. The differencing process can be expressed as [11]

$$Y_t = X_t - X_{t-1} \tag{2}$$

where Y_t is the differences series.

Moving Average (MA): The MA component models the relationship between an observation and a residual error from a moving average model applied to lagged residuals. The current moving average value is the linear combination of the current disruption with one or more which occurred on previous state. Mathematically, the MA(q) model is defined as:

$$X_t = \mu + \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \cdots + \theta_q \epsilon_{t-q} \tag{3}$$

where:

- X_t is the value of the time series at time t,
- μ is the mean of the series,
- ϵ_t is the white noise (error) term at time t,
- $\theta_1, \theta_2, \ldots, \theta_q$ are the coefficients of the model,
- $\epsilon_{t-1}, \epsilon_{t-2}, \ldots, \epsilon_{t-q}$ are the lagged error terms, and
- q is the number of lagged error terms (the order of the MA model) [11].

2.3. ARIMA: parameter selection

Model identification: The ARIMA model has three components: AR (Auto-Regressive), I (Integrated), and MA (Moving Average).To identify the correct parameters for each, an Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) were plotted. The ACF indicated the appropriate lag for the MA component, while the PACF suggested the lag for the AR component [1].

Order of differencing(d): Based on the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) plots, as shown in Fig. 3, the data required one differencing operation to become stationary, resulting in $p = 2$, $d = 1$, and $q = 2$. Differencing is applied as follows to achieve stationarity:

$$\Delta^d y_t = y_t - y_{t-d} \tag{4}$$

where $\Delta^d y_t$ represents the differenced time series, and d is the number of times differencing is applied to make the series stationary [12].

Order of AR (p) and MA (q): The PACF and ACF plots (Fig. 3) suggested an optimal AR order of 2 and MA order of 2, leading to an ARIMA(2,1,2) model. The ACF helps determine the MA order (q), while the PACF helps determine the AR order (p) [11].

Model fitting: After selecting the parameters, the ARIMA model was fitted to the data. The model was trained on a portion of the data, and forecasts were made for the remaining time points. The parameters $\phi_1, \ldots, \phi_p$ and $\theta_1, \ldots, \theta_q$ are estimated by minimizing the residual sum of squares (RSS):

$$RSS = \sum_{t=1}^{n} (y_t - \hat{y}_t)^2 \tag{5}$$

where y_t is the observed value at time t, and $\hat{y}_t$ is the predicted value from the ARIMA model at time t [13].

Residual analysis: Once the ARIMA model was fitted, the residuals (the differences between the observed and predicted values) were analyzed to ensure they followed a white noise pattern. If the residuals showed no autocorrelation and followed a normal distribution, it confirmed that the model had captured the underlying structure of the data effectively. The autocorrelation function (ACF) of the residuals should be close to zero:

$$ACF_\epsilon(h) = \frac{\text{Cov}(\epsilon_t, \epsilon_{t+h})}{\text{Var}(\epsilon_t)} \quad \text{for} \quad h = 1, 2, 3, \ldots \tag{6}$$

Additionally, the Ljung-Box test can be used to test for white noise:

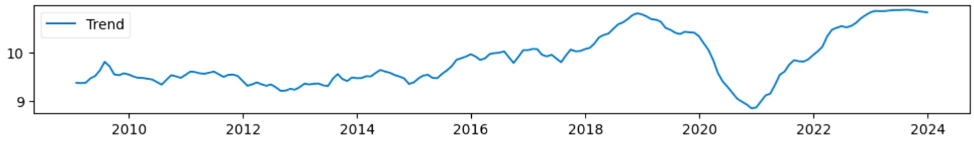

Figure 1: Trend analysis.

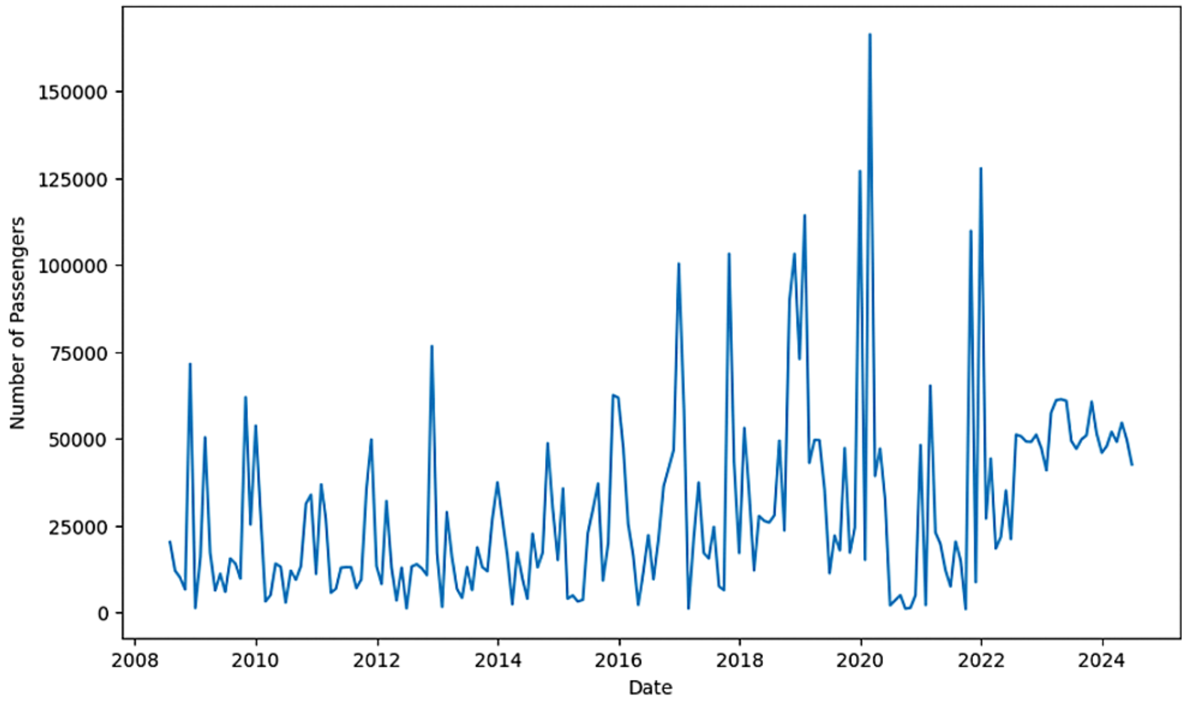

Figure 2: Seasonality analysis.

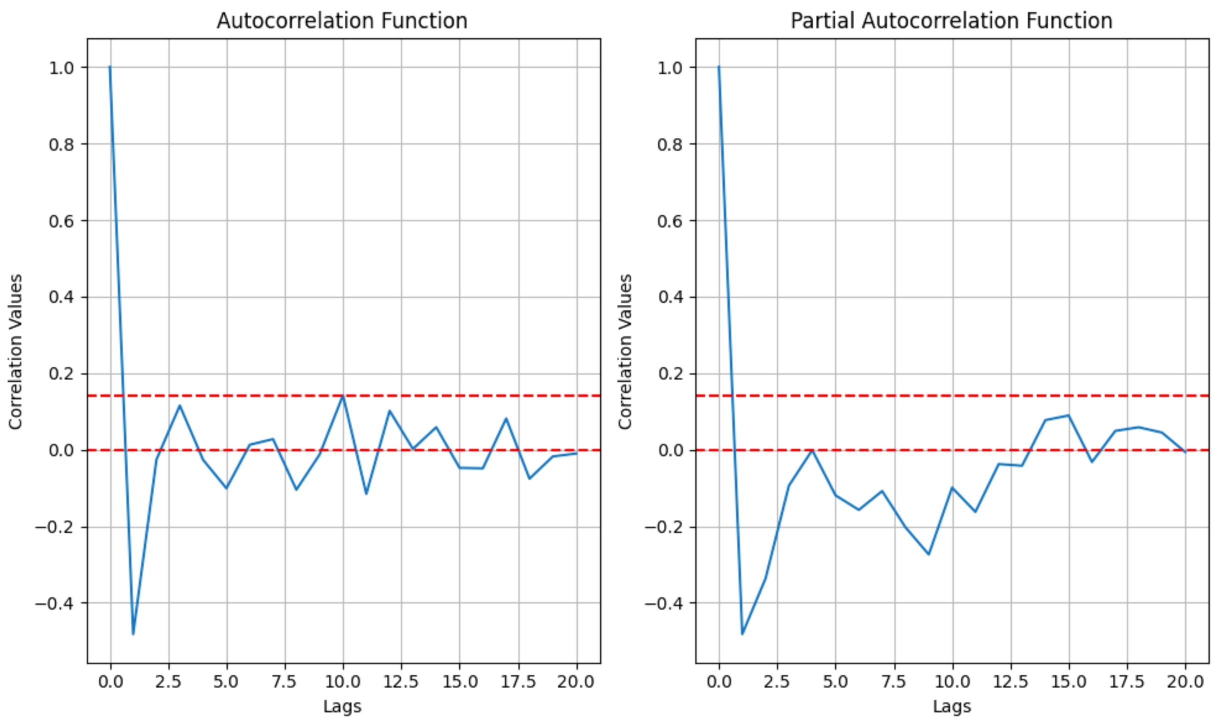

Figure 3: ACF and PACF lags.

$$Q = n(n+2)\sum_{h=1}^{H}\frac{\hat{\rho}^2(h)}{n-h} \tag{7}$$

where $\hat{\rho}(h)$ is the autocorrelation at lag h, n is the number of observations, and H is the maximum lag for which the test is performed. If Q is not significant, the residuals are considered to be white noise [14].

Forecasting: Using the ARIMA model, forecasts were made for future time periods, with confidence intervals calculated to estimate the range of likely values. These forecasts were plotted alongside the actual values, showing that the ARIMA model was able to accurately capture both the trend and seasonal variations present in the dataset. The forecast for time $t+h$ is given by:

$$\hat{y}_{t+h} = \mu + \sum_{i=1}^{p}\phi_i y_{t+h-i} + \sum_{j=1}^{q}\theta_j \epsilon_{t+h-j} \tag{8}$$

where $\hat{y}_{t+h}$ is the forecasted value at time $t+h$ and ϵ_{t+h-j} are the residuals at time $t+h-j$ [15].

2.4. LSTM model

2.4.1. LSTM overview

LSTM network, introduced by Hochreiter and Schmidhuber (1997), is a specialized form of Recurrent Neural network (RNN) designed to capture long-range dependencies in sequential data, addressing the vanishing gradient issue common in traditional RNNs. This enhancement is achieved by employing "gates" that control information flow, enabling the network to retain relevant information over time and selectively update its internal state as needed. This adaptability has established LSTMs as a prominent choice in time-series forecasting, as they effectively capture both short-term variations and long-term trends in data [16].

In this study, we leverage the LSTM model's unique capabilities to forecast monthly passenger counts for Nepal Airlines, aiming to model the seasonal and trend-based fluctuations evident in the dataset spanning 15 years. Given the sequential nature of the data, LSTMs are particularly suited to modeling dependencies in both seasonal peaks and troughs, as well as long-term growth patterns.

An LSTM cell is built upon several essential components that facilitate the management of information across time steps:

Cell state (c_t): The cell state is the core memory of the LSTM cell, responsible for preserving long-term dependencies across the sequence [17]. Information flows along this cell state path, while gates control modifications, allowing the cell to retain or discard specific information as required [18].

Hidden state (h_t): The hidden state is the output of the LSTM cell at each timestep, encapsulating the cell's immediate output and short-term memory. This hidden state is used for generating predictions and is passed to the next timestep along with the updated cell state [18].

Forget gate (f_t): The forget gate determines which information should be discarded from the cell state, based on the current input and the previous hidden state. The forget gate is defined by the equation:

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{9}$$

where σ is the sigmoid function, W_f is the weight matrix, and b_f is the bias vector [19].

Input gate (i_t): The input gate decides which new information should be added to the cell state. The gate generates two values: the input gate vector and the candidate cell state. The equations are:

$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] + b_i\right) \tag{10}$$

$$\tilde{c}_t = \tanh\left(W_c \cdot [h_{t-1}, x_t] + b_c\right) \tag{11}$$

The input gate vector controls the addition of the candidate cell state to the cell state, introducing new information to the LSTM cell [17].

Output gate (o_t): The output gate determines the information from the cell state that should be passed to the hidden state (h_t) as an output and as an input to the next timestep. It is calculated as:

$$o_t = \sigma\left(W_o \cdot [h_{t-1}, x_t] + b_o\right) \tag{12}$$

The hidden state is then updated using:

$$h_t = o_t \times \tanh(c_t)$$

where o_t allows selective output from the cell state to be passed forward in the sequence [18].

2.5. LSTM operation in time-series forecasting

In the context of time-series forecasting, the LSTM model processes each timestep sequentially, maintaining long-term dependencies across multiple timesteps to learn both seasonal patterns and trends. For our dataset of monthly airline passengers, the LSTM leverages its sequential design to capture dependencies critical for understanding fluctuations, including peaks (e.g., during holiday seasons) and troughs [16].

Input representation: Each monthly passenger count (x_t) serves as input to the LSTM cell at each timestep, enabling the model to learn temporal dependencies by updating cell and hidden states throughout the sequence [18].

Training: During training, the LSTM model optimizes its gates to capture both long- and short-term trends in passenger counts. This optimization process enables the model to develop a robust understanding of seasonal variations and general trends in the dataset [16].

Prediction: For forecasting, the LSTM model uses past values to predict future passenger counts by leveraging the hidden and cell states to generate forecasts for subsequent months [20]. By propagating both cell and hidden states through the sequence, the LSTM captures complex dependencies, offering advantages over static models such as ARIMA, particularly in scenarios with non-linear or highly variable patterns [21].

Through this architecture, the LSTM adapts its internal states based on recent changes and long-term patterns in the data, making it a strong choice for capturing seasonal and trend-based variations, as evidenced in airline passenger forecasting tasks [2].

2.6. LSTM model architecture and hyperparameters

The LSTM model architecture was designed to capture both short-term and long-term dependencies in Nepal Airlines' monthly passenger data. The architecture comprises an input layer, two LSTM layers, a dropout layer, and a dense output layer. We optimized the hyperparameters through grid search and evaluated configurations based on performance on a validation set, prioritizing model stability and accuracy.

The final architecture consists of two LSTM layers, each with 50 units, with a dropout rate of 0.4 applied between the layers to prevent overfitting. The dense output layer includes a single neuron with a linear activation function for producing the final forecast value. Hyperparameters such as batch size, learning rate, and the number of epochs were selected based on empirical evaluation.

Early stopping was also applied to terminate training when validation loss plateaued.

The hyperparameter values, detailed in Table 1, provided a good balance between training efficiency and generalization on the test set. This configuration enabled the LSTM model to effectively learn complex patterns in the time series data, resulting in improved forecast accuracy over baseline models.

Table 1: LSTM model hyperparameters.

Hyperparameter	Value
LSTM units	50 (per layer)
Batch size	16
Learning rate	0.001
Dropout rate	0.4
Epochs	100
Loss function	MSE
Optimizer	Adam

3. Results and discussion

Nepal Airlines data was found to exhibit a trend and seasonality, therefore differencing techniques were applied to make the data stationary. The result of the Dickey-Fuller test is provided below:

Table 2: Results of the Dickey-Fuller test for stationarity.

Dickey-Fuller test results	
Test statistic	-1.276
p-value	0.640
Lags used	1
Observations	100
Critical value (1%)	-3.50
Critical value (5%)	-2.88
Critical value (10%)	-2.57

In Table 2, the p-value 0.640 is much higher than typical significance thresholds, such as 0.05 or 0.01. This means there is strong evidence that the null hypothesis of the test, which suggests the series is non-stationary, cannot be rejected. Furthermore, the test statistic of -1.276 is much greater than the critical values at all significance levels (1%, 5%, and 10%). If the test statistic were smaller (more negative), we could have concluded that the series is stationary. Since it's not, it indicates that the data likely exhibits non-stationary. Thus, we have used LogDiffShifting approach, where we combine Log Transformation and Differencing. Log Transformation is applied first to reduce the variability of the series and stabilize variance. Differencing (with shifting) is applied afterward to remove any trend or seasonality and make the series stationary [22].

In Table 3, The calculated value of the Augmented Dickey-Fuller (ADF) test statistic is -3.09. The associated p-value is 0.028, which is less than the common significance levels of 0.05, indicating evidence against the null hypothesis of non-stationarity. The test used 1 lag to account for autocorrelation in the time series, and it was performed on 100 observations. At the 1% significance level, the critical value is -3.47, at the 5% level it is -2.88, and at the 10% level it is -2.58. Since the test statistic -3.09 is more negative than the critical value at the 5% significance level -2.88, we reject the null hypothesis and conclude that the time series is now stationary after log transformation and we move forward with ARIMA model.

Table 3: Results of the Dickey-Fuller test for stationarity after log transformation.

Dickey-Fuller test results (log transformed)	
Test statistic	-3.09
p-value	0.028
Lags used	1
Observations	100
Critical value (1%)	-3.47
Critical value (5%)	-2.88
Critical value (10%)	-2.58

Furthermore, based on the comparative analysis of ARIMA and LSTM models for forecasting Nepal Airlines passenger data, the results as shown in Table 4 demonstrate that the ARIMA model outperforms the LSTM model. The evaluation metrics, including Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE), are presented below:

Table 4: Performance metrics comparison.

Model	Mean Absolute Error (MAE)	Root Mean Squared Error (RMSE)
ARIMA	0.74	1.78
LSTM	0.87	2.02

The findings from the research show that ARIMA performed better than LSTM at forecasting passenger numbers for Nepal Airlines. The better performance of LSTM was reported by Gupta et al. (2019)[2], who obtained MAPEs of 8.79% and 10.8% for training and testing data, respectively. The disparity might be explained by the fact that Nepal Airlines' data primarily exhibits seasonal and linear trends, which align more closely with ARIMA's strengths. In contrast, the dataset used by Gupta et al. favored LSTM, as it included non-linear and long-term relationships. Additionally, the smaller size of the Nepal Airlines dataset likely limited LSTM's ability to generalize complex patterns effectively. Previous research by Siami-Namini et al. (2018)[16] also highlighted LSTM's challenges with limited data and potential overfitting in rolling forecasts, which is consistent with our findings.

For Nepal Airlines, ARIMA successfully captured the underlying seasonal and trend components of the data, which supports findings by Andreoni et al.[4]. Their study demonstrated that multivariate ARIMA models effectively forecast air transport demand by capturing the interactions among multiple influencing variables, leading to more accurate and comprehensive predictions. This strength in forecasting has practical implications for airline operations and market strategy. Accurate forecasts from ARIMA enable the optimization of flight schedules, ensuring resources are allocated efficiently during peak travel seasons which is beneficiary for the corporation like Nepal Airlines . The model's insights can also enhance inventory management and operational planning by helping the airline prepare for fluctuations in passenger demand. Furthermore, ARIMA forecasts can inform targeted marketing campaigns to boost passenger engagement during low-demand periods, thereby supporting revenue growth and improving overall operational efficiency.

4. Conclusion

The effectiveness of the LSTM and ARIMA models for predicting Nepal Airlines passenger statistics is compared in this study. The findings demonstrate that ARIMA is an effective approach for aviation forecasting since it performs exceptionally well on datasets with seasonality and linear trends. However, despite LSTM's ability to handle intricate and non-linear patterns, the dataset's size and more straightforward structure restricted its performance.

The results have significant applications. In order to improve operations and customer satisfaction, Nepal Airlines may use ARIMA to build more accurate forecasts, which will help them optimize flight schedules, manage resources effectively, and create better marketing campaigns.

This study emphasizes how crucial it is to choose models that complement the features of the dataset and the forecasting objectives. By penalizing excessive parameters, metrics like AIC and BIC, which balance model fit and complexity, could be used in future research to improve the choice of ARIMA models. A promising method for utilizing both linear and non-linear patterns is to combine ARIMA with LSTM. Furthermore, including outside factors like economic indicators could improve forecast accuracy and offer a more thorough modeling framework.

Acknowledgment

We are thankful to Mrs. Supriya Tuladhar, Deputy Director, Corporate Fleet Planning and Bilateral Division, Nepal Airlines for providing valuable inputs.

References

[1] Box G E P & Jenkins G M. *Time Series Analysis: Forecasting and Control (Revised Edition)*. Holden-Day (1976).

[2] Gupta V, Sharma K & Sangwan M S, Airlines passenger forecasting using LSTM based recurrent neural networks, *International Journal Information Theories and Applications*, 26(2) (2019) 178–187. URL http://www.foibg.com/ijita/vol26/ijita26-02-p06.pdf.

[3] Pan B, Yuan D, Sun W, Liang C & Li D. A novel LSTM-based daily airline demand forecasting method using vertical and horizontal time series. In: Ganji M, Rashidi L, Fung B & Wang C, eds., *Trends and Applications in Knowledge Discovery and Data Mining. PAKDD 2018*, vol. 11154 of *Lecture Notes in Computer Science*. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-04503-6_17.

[4] Andreoni A & Postorino M N. A multivariate ARIMA model to forecast air transport demand. In: *Proceedings of the Association for European Transport and Contributors* (2006), pp. 1–14. URL https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=53964452afb9c4846b344c6cbafba0a637384db4.

[5] Asrah N M, Nor M E, Rahim S N & Leng W K, Time series forecasting of the number of Malaysia Airlines and AirAsia passengers, *Journal of Physics: Conference Series*, 995 (2018) 012006. https://doi.org/10.1088/1742-6596/995/1/012006.

[6] Siami Namini S, Tavakoli N & Siami Namin A, A comparison of ARIMA and LSTM in forecasting time series (2018) 1394–1401. https://doi.org/10.1109/ICMLA.2018.00227.

[7] Zhang R. LSTM-based stock prediction modeling and analysis. In: *Proceedings of the 2022 7th International Conference on Financial Innovation and Economic Development (ICFIED 2022)*. Atlantis Press (2022). ISBN 978-94-6239-554-1. ISSN 2352-5428, pp. 2537–2542. https://doi.org/10.2991/aebmr.k.220307.414.

[8] Canale R P & Chapra S C. *Numerical Methods for Engineers with Programming and Software Applications*. McGraw-Hill, Hoboken, 3rd edn. (1998).

[9] Williamson D, Parker R & Kendrick J, The box plot: A simple visual method to interpret data, *Annals of internal medicine*, 110 (1989) 916–21. https://doi.org/10.7326/0003-4819-110-11-916.

[10] Nepal Airlines. Profile of aircrafts used in NAC services - Nepal Airlines. https://www.nepalairlines.com.np/history (2024). Accessed: November 2024.

[11] Brockwell P J & Davis R A. *Introduction to Time Series and Forecasting (2nd ed.)*. Springer (2002). URL https://link.springer.com/book/10.1007/b97391.

[12] Hyndman R J & Athanasopoulos G. *Forecasting: Principles and Practice (2nd ed.)*. OTexts (2018). Available online: https://otexts.com/fpp2/.

[13] Shumway R H & Stoffer D S. *Time Series Analysis and Its Applications: With R Examples (4th ed.)*. Springer (2017). URL https://link.springer.com/book/10.1007/978-3-319-52452-8.

[14] Ljung G M & Box G E P, On a measure of a lack of fit in time series models, *Biometrika*, 65(2) (1978) 297–303. https://doi.org/10.2307/2335207.

[15] Hamilton J D. *Time Series Analysis*. Princeton University Press (1994). https://doi.org/10.2307/j.ctv14jx6sm.

[16] Siami-Namini S, Tavakoli N & Siami Namin A. A comparison of ARIMA and LSTM in forecasting time series. In: *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)* (2018), pp. 1394–1401. https://doi.org/10.1109/ICMLA.2018.00227.

[17] Hochreiter S & Schmidhuber J, Long short-term memory, *Neural Computation*, 9(8) (1997) 1735–1780. ISSN 0899-7667. https://doi.org/10.1162/neco.1997.9.8.1735.

[18] Graves A. Generating sequences with recurrent neural networks (2014). URL https://arxiv.org/abs/1308.0850.

[19] Gers F A, Schmidhuber J & Cummins F, Learning to forget: Continual prediction with LSTM, *Neural Computation*, 12(10) (2000) 2451–2471. ISSN 0899-7667. https://doi.org/10.1162/089976600300015015.

[20] Bai S, Kolter J Z & Koltun V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling (2018). URL https://arxiv.org/abs/1803.01271.

[21] Khandelwal I, Adhikari R & Verma G, Time series forecasting using hybrid ARIMA and ANN models based on DWT decomposition, *Procedia Computer Science*, 48 (2015) 173–179. ISSN 1877-0509. https://doi.org/https://doi.org/10.1016/j.procs.2015.04.167. International Conference on Computer, Communication and Convergence (ICCC 2015).

[22] Box G E P, Jenkins G M, Reinsel G C & Ljung G M. *Time Series Analysis: Forecasting and Control*. Wiley, Hoboken, NJ, 5th edn. (2015). ISBN 978-1118675021. https://doi.org/10.1111/jtsa.12194.