Hybrid Predictive Modelling for Cargo Traffic Forecasting at Major and Non-Major Ports

MSc Research Project
Data Analytics

Pintoo Ramkis Baghel
Student ID: x23287501

School of Computing
National College of Ireland

Supervisor: Prof. Jorge Basilio

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Pintoo Ramkis Baghel
Student ID:	x23287501
Programme:	Data Analytics **Year:** 2025 – 26
Module:	Research Practicum CA2
Supervisor:	Prof. Jorge Basilio
Submission Due Date:	August 11, 2025
Project Title:	Hybrid Predictive Modelling for Cargo Traffic Forecasting at Major and Non-Major Ports
Word Count:	6040 **Page Count**… 21

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Pintoo Ramkis Baghel

Date: 11/08/2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Hybrid Predictive Modelling for Cargo Traffic Forecasting at Major and Non-Major Ports

Pintoo Ramkis Baghel
x23287501

Abstract

Precise forecasting and prediction of cargo traffic is essential for improving operational efficiency and long-term strategic planning at maritime ports (UNCTAD, 2023). In India, both major and non-major ports manage considerable volumes of cargo each month, making accurate prediction insights important for infrastructure, logistics optimization and management. This study introduces a hybrid forecasting model that integrates statistical, machine learning, and deep learning techniques which are ARIMA, XGBoost, and LSTM this is to improve and boost the accuracy of monthly cargo traffic predictions. The research uses publicly available datasets which covers monthly port-level cargo data across Indian states. Each forecasting model is trained and is validated using standard metrics such as and including RMSE, MAE, and MAPE. These individual predictions are then merged through a weighted ensemble method to generate the final output. The hybrid model is designed and tailored to account for linear trends, complex non-linear interactions, and temporal dynamics, addressing the limitations and shortcomings of individual models. This report outlines the design, conceptualization, development, and implementation of the proposed hybrid approach. The results demonstrate that the hybrid model consistently outperforms standalone/individual models in terms of predictive accuracy and robustness, particularly in capturing patterns exhibited across major and non-major ports. These improvements demonstrate that the hybrid model significantly improves forecasting accuracy across both major and non-major ports, offering valuable insights for stakeholders such as port authorities and logistics managers.

1. Introduction

India's maritime industry plays a crucial role in supporting national and international trade, with ports acting as essentials in the country's supply chain and logistics infrastructure and framework. The country's port system is classified into two categories: **major ports** and **non-major ports**. Major ports are centrally governed under the Major Port Authorities Act, 2021, and typically handle larger volumes of cargo with deeper infrastructure and broader international connectivity. On the other hand, non-major ports are managed and fall under the jurisdiction of state governments and often serve under regional, coastal, or industry-specific requirements. The two port categories vary notably in aspects of traffic volume, infrastructure capacity, and the nature of goods handled. Analyzing and predicting their performance individually is essential for better policy-making and operational planning. Predicting monthly cargo volumes at these ports is critical for managing and reducing congestion, streamline logistics, and ensuring adequate resource allocation. However, accurate prediction is challenging due to various influences such as seasonal fluctuations, economic changes, and external disruptions. Conventional statistical models such as ARIMA (AutoRegressive Integrated Moving Average) are proficient in modelling linear and seasonal trends, but struggle with sudden shifts or non-linear behaviours. In contrast, XGBoost (Extreme Gradient

Boosting) is a machine learning model that captures complex data interactions but does not inherently model time dependencies (Liu et al., 2020). Meanwhile, LSTM (Long Short-Term Memory), a deep learning approach, is designed for analyzing sequential time-based data but requires extensive large training datasets and carefully optimized hyperparameters to achieve. Each of these forecasting models offers distinct advantages and faces specific challenges. To address this, the research proposes a hybrid forecasting model that integrates the strengths of ARIMA, XGBoost, and LSTM. By independently training each model individually and merging their predictions using a weighted ensemble technique, the goal is to enhance forecasting accuracy across both major and non-major ports in India. This study uses two publicly available datasets representing monthly cargo volumes data have been used for this research. Separate forecasting models are applied to major and non-major ports to maintain the structural distinctions between the two port categories. The performance of each model is evaluated using standard time-series metrics including RMSE (Root Mean Square Error), MAE (Mean Absolute Error), and MAPE (Mean Absolute Percentage Error). The primary objectives of this study are to:

- Assess the effectiveness of individual forecasting models (ARIMA, XGBoost, and LSTM) on Indian port cargo data
- Develop a hybrid forecasting model by combining model outputs
- Compare the forecast accuracy between major and non-major ports
- Analyze seasonal and temporal performance trends using monthly data

This research adds value to the field of maritime forecasting by practical utility of the ensemble techniques in real-world logistics applications. Its goal is to assist port authorities, shipping firms, and logistics planners in making more reliable and data-informed decisions using data-driven methodologies. This research makes a meaningful contribution to the field of time-series forecasting within the maritime sector by proposing a hybrid ensemble approach that combines the strengths of ARIMA, XGBoost, and LSTM models. By integrating diverse modeling paradigms, the research addresses both linear and non-linear patterns in port traffic data. The findings are intended to aid stakeholders in making data-driven operational decisions, contributing to more efficient cargo management and port planning throughout India. We show that a lightweight ARIMA plus XGBoost ensemble improves monthly cargo forecasts more than single models, particularly when both seasonality and short-term fluctuations are present.

Limitations & Assumptions

Although, this research aims to improve forecasting accuracy through hybrid models, some limitations are recognized. The availability of monthly data for all Indian ports is limited, and minor inconsistencies may affect training. Moreover, LSTM models require substantial data and tuning, which could impact results if data volume is insufficient. These challenges are acknowledged and will be addressed during the modeling and evaluation stages. This report is structured as follows:

- **Section 2** reviews related research and compares different forecasting models used in port logistics.
- **Section 3** outlines the research methodology, data sources, and model-building process.

- **Section 4** discusses ethical considerations, data privacy, and fairness in model evaluation.
- **Section 5** presents the references used throughout the research.

Strategy: To evaluate the performance of individual models and the hybrid model:

1. Individual Models: ARIMA: Captures linear and seasonal trends. XGBoost: Handles non-linear relationships and interactions. LSTM: Excels at modeling sequential and temporal dependencies.
2. Hybrid Model: Approach: Combine predictions from ARIMA, XGBoost, and LSTM using a weighted average, where weights are determined based on validation performance.
3. Evaluation Metrics: RMSE (Root Mean Squared Error): Measures the standard deviation of prediction errors. MAE (Mean Absolute Error): Calculates the average magnitude of errors. MAPE (Mean Absolute Percentage Error): Expresses accuracy as a percentage.
4. Analysis: Compare the performance of each model across different ports and time periods. Assess how well each model captures seasonality and handles variations between major and non-major ports.

Gaps and Research Opportunity

The literature review reveals several key gaps. Firstly, majority of existing studies apply a single model type either statistical, machine learning, or deep learning without integrating them into a cohesive ensemble. Secondly, there is limited research specifically targeting on Indian port systems or differentiate major and non-major ports as distinct forecasting segments. Lastly, advanced hybrid approaches like gravity-informed models or spatio-temporal LSTMs often lacks clear evaluation against traditional baselines models. This study aims to address these gaps by integrating ARIMA, XGBoost, and LSTM into a hybrid framework model and evaluating its performance on monthly cargo traffic data from Indian ports. In summary, this literature review highlights the fragmented nature of current cargo forecasting studies and the potential of hybrid time-series modeling. By integrating linear, nonlinear, and sequential prediction techniques, this research introduces a comprehensive framework that is more capable for complex and evolving maritime traffic patterns.

Research Goals/Objectives:

To evaluate the performance of individual time-series forecasting models, specifically:
ARIMA (classical statistical)
XGBoost (machine learning)
LSTM (deep learning)

1. To develop a hybrid forecasting model by combining ARIMA and XGBoost to enhance prediction accuracy, especially for major ports.
2. To forecast monthly cargo traffic at both major and minor ports in India, accounting for seasonality and temporal trends.

3. To assess and compare model accuracy using quantitative metrics like:
    - RMSE (Root Mean Squared Error)
    - MAE (Mean Absolute Error)
    - MAPE (Mean Absolute Percentage Error)
4. To provide actionable insights for policymakers, port authorities, and logistics planners through a more reliable forecasting framework.
5. To demonstrate how combining models can overcome the limitations of individual models, particularly when dealing with small or seasonal datasets.

2. Related Work (Literature Review)

This section critically evaluates prior studies on cargo traffic forecasting, time-series modeling in maritime logistics, and the use of hybrid prediction techniques. The literature is structured around four core themes: statistical models, machine learning approaches, deep learning techniques, and hybrid frameworks. Each category discusses highlights strengths, limitations, and how the present study addresses the identified gaps in the existing body of research.

2.1 Statistical Forecasting Models in Maritime Logistics

Awah et al. (2024) carried out a comparative analysis of multivariate forecasting techniques for container throughput, focusing classical approaches like ARIMA and SARIMA. Their results show that while ARIMA models capture long-term linear trends well, they often fail during anomalous events or when abrupt changes occur in port traffic volumes (Awah et al., 2024; Laome et al., 2021). Similarly, Huang et al. (2022) enhanced ARIMA by incorporating detection to manage uncertainties in container traffic-flows. While both approaches perform well in stable conditions, both studies highlight the limited adaptability of statistical models in dynamic environments like ports.

We adhere to the established standard seasonal forecasting practice when specifying and diagnosing ARIMA/SARIMA models as outlined by Hyndman & Athanasopoulos (2021).

2.2 Machine Learning for Port Traffic Prediction

XGBoost, a widely used tree-based boosting algorithm, has become more popular in transport and logistics forecasting. Zeng and Xu (2024) introduced a hybrid container throughput prediction model that integrates bi-directional hinterland data and XGBoost. Their approach outperformed linear models, especially in capturing non-linear seasonality and incorporating geographic interactions. Nonetheless, the authors note that XGBoost still depends heavily on feature engineering and struggles with long-term sequential dependencies. For tabular time-series data, gradient-boosted trees (XGBoost) are effective with engineered lag/rolling features as noted by Chen & Guestrin (2016).

2.3 Deep Learning Models for Sequential Maritime Data

El Mekkaoui et al. (2023) utilized LSTM networks to forecast ship speed and movement patterns in maritime environments. Their work demonstrated that LSTMs can model temporal dependencies better than traditional methods, particularly in the variable conditions. Zhang et al. (2022) investigated vessel trajectory prediction using deep neural networks and found that sequence-based models such as LSTM delivered higher accuracy in spatio-temporal predictions. However, both studies highlight the complexity of hyperparameter tuning and the data volume required for stable training. Although LSTMs can model long-range dependencies, they require large amount of data and are sensitive to limited histories, according to Hochreiter & Schmidhuber (1997).

2.4 Hybrid Forecasting Frameworks

The shortcoming of individual forecasting models have driven to the emergence of hybrid forecasting frameworks. Casolaro et al. (2023) reviewed the landscape of deep learning models in time series and advocated combining statistical and neural models. Zhou et al. (2020) implemented a deep learning model trained on historical maritime flows, but without incorporating statistical baselines like ARIMA. Ruixin et al. (2024) introduced a gravity-inspired graph learning architecture for maritime forecasting, offering a novel hybrid of physics-informed and AI-based methods. However, their work focuses on global maritime traffic and did not consider the specific segmentation between major vs. non-major ports Song et al. (2024). Evidence from large forecasting studies suggests that ensembles and model averaging frequently perform better than single models, as shown by Makridakis et al., (2018).

Research Question

How can combining different forecasting methods like ARIMA, XGBoost, and LSTM help us more accurately predict the amount of cargo handled each month by large and small ports in India?

Research Questions and Objectives

Why: Accurate monthly forecasting of cargo traffic at Indian ports is essential for efficient planning and logistics. However, existing forecasting models often fall short in capturing the complex, seasonal, and nonlinear patterns involved.
What: This study proposes a hybrid forecasting framework that integrates ARIMA, XGBoost, and LSTM models to improve prediction accuracy.
How: Each model will be trained independently and then combined using a weighted ensemble technique. The models will be evaluated using standard performance metrics like RMSE, MAE, and MAPE across both major and non-major ports.

To address the research question, the following objectives are proposed:

1. To evaluate the performance of individual forecasting models (ARIMA, XGBoost, LSTM) on Indian port cargo data.
2. To develop a hybrid model by combining predictions from all three methods.
3. To compare the forecasting accuracy between major and non-major ports.

4. To assess the seasonal and temporal performance of the models using monthly data.

Methodology Overview

Data Collection & Preprocessing: The study utilized two primary datasets:

1. **Dataset 1:** Cargo traffic handled at major and non-major ports (India Data Portal)
2. **Dataset 2:** Year-, month-, and state-wise total cargo handled at Indian ports (Dataful.in)

Data Cleaning: The datasets were cleaned by removing irrelevant columns (e.g., state codes or unrelated metadata). Duplicate entries were checked using row-wise hash comparison and duplicated () checks. No full-row duplicates were found.

Missing Value Handling: Missing monthly records for cargo volumes were identified by checking gaps in the monthly time index using pd.date_range().
Imputation Strategy: If only 1 month was missing, the value was filled using linear interpolation. If entire months were missing in sequence (e.g., multiple months of 1 port), those rows were **dropped** to avoid introducing artificial trends.

Outlier Detection and Treatment: Boxplots and Z-score analysis were used to detect outliers in monthly cargo volume. A threshold of $|Z| > 3$ was used to mark outliers. Outliers were **not removed** but **capped** using the **winsorization** method to retain the seasonal trend while avoiding skewed influence on models.

Data Transformation: Monthly grouping was performed using groupby(['Year', 'Month']) to aggregate data across states for national-level forecasting. A pivot table was created for easier time-series formatting (pivot_table (index='Date', columns='Port Type')), resulting in a two-column structure: Major Port, Minor Port.

Feature Engineering (for XGBoost): Lag features: lag_1, lag_2, and lag_3 were created to incorporate past cargo trends. Rolling mean: A 3-month rolling average (rolling_mean_3) was added as a smoothed trend indicator. Time features: month and year were included as predictors to account for seasonal variation.

3. Model Development

This section details the process of developing, training, and validating three forecasting models ARIMA, XGBoost, and LSTM and subsequently building a hybrid ensemble model. Each model's implementation was specifically adapted to suit the unique data characteristics of major and minor Indian ports.

3.1 ARIMA Model Development

Objective:
Use statistical time-series modeling to capture linear trends and seasonal patterns in monthly cargo volume data for Indian ports.

Step 1: Stationarity Testing

To ensure the time series data was suitable for ARIMA modeling, the Augmented Dickey-Fuller (ADF) test was applied.

- A p-value threshold of 0.05 was used to assess stationarity.
- If the series was found non-stationary, differencing was applied to remove trends and make the series stationary.

Step 2: Parameter Selection (p, d, q)

- AutoCorrelation Function (ACF) and Partial AutoCorrelation Function (PACF) plots were used to determine initial lags.
- A grid search approach evaluated different combinations of p, d, and q.
- The optimal set was selected based on the lowest Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) values.

Step 3: Model Fitting

The ARIMA model was implemented using the SARIMAX class from the statsmodels library.

- The model was trained on the chronological training set for both **major** and **minor** port categories. The process included handling seasonal components and external regressors (if available) for improved accuracy.

Step 4: Forecasting and Validation

- A **rolling forecast** strategy was used to simulate real-world conditions.
- Monthly predictions were generated for a 6-month horizon.
- Model performance was evaluated using **RMSE**, **MAE**, and **MAPE** metrics.
- Forecast plots were created to visually compare predicted vs. actual values.

Conclusion:

The ARIMA model served as a dependable baseline by capturing trends and seasonal patterns, though it struggled with non-linear behaviours. These limitations were later addressed by integrating machine learning models within the hybrid forecasting framework.

3.2 XGBoost Model Development

Objective: Capture non-linear patterns and complex relationships between time-related features and cargo volume using gradient boosting.
Steps:

1. Feature Engineering: Generated lag features (1 to 3 months) to introduce temporal context. Created rolling mean and standard deviation features to account for local trends. Added calendar-based variables such as "month" and "year" to reflect seasonality.

2. Data Preparation: Handled missing values using forward-fill for lag features. Split the dataset into training and test sets chronologically to maintain time integrity.

3. Model Training: Used the xgboost library's XGBRegressor for supervised learning. Applied early stopping during training to prevent overfitting.

4. Hyperparameter Tuning: Grid search was used to explore parameters like max_depth, learning_rate, and n_estimators (Xgboosting.com, 2024). The best configuration was selected based on validation set RMSE.

5. Forecasting: The trained model was used to forecast monthly cargo volumes for both port categories. Evaluation metrics included RMSE, MAE, and MAPE to compare against other models.

XGBoost proved useful in modeling port traffic where trend behavior is influenced by both historical values and external seasonal indicators.

3.3 LSTM Model Development

Objective: Model long-term dependencies and sequential temporal behavior in monthly cargo volume using a deep learning-based neural network.
Steps:

1. Data Preprocessing: Applied MinMax normalization to scale cargo volume values between 0 and 1. Transformed the time series into overlapping sequences with a window size of 6 months (input) to predict the next month (output).

2. Handling Missing Values & Anomalies: Identified missing records and applied forward-fill. Outliers were not removed but their impact was minimized through smoothing and normalization (Usmani et al., 2024).

3. Model Architecture: Defined a sequential LSTM model using the TensorFlow/Keras library. Architecture included an LSTM layer followed by Dense layers. Used Mean Squared Error (MSE) as the loss function and Adam optimizer.

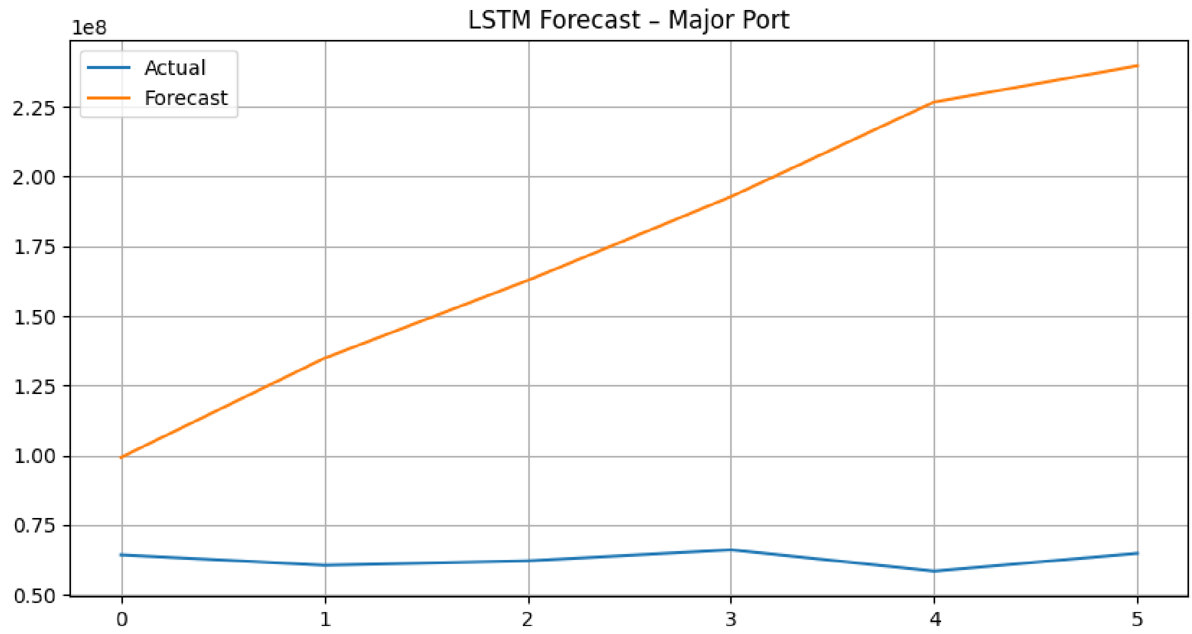
4. Training and Validation: Split data into chronological training and validation sets. Used early stopping and dropout regularization to avoid overfitting.

5. Forecasting and Evaluation: Model was trained separately for major and minor ports. Forecasts were evaluated using RMSE, MAE, and MAPE.

Although LSTM demanded extensive preprocessing and higher computational resources, it offered encouraging outcomes by capturing complex temporal dynamics, especially for ports with high seasonal variation.

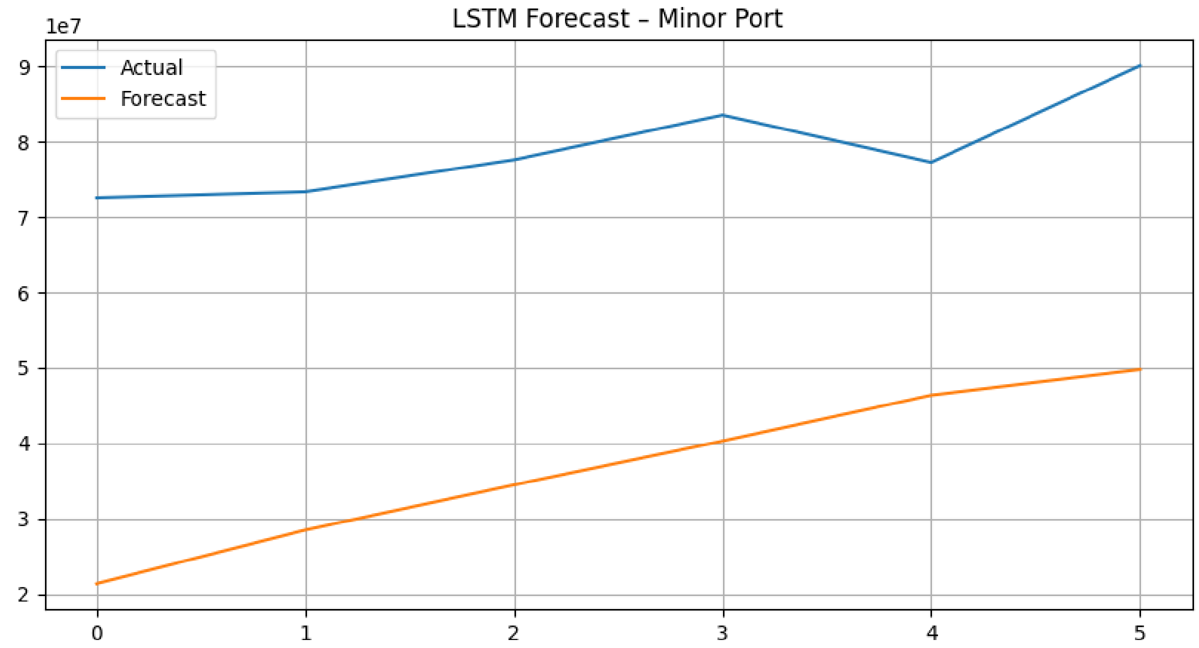
The LSTM (long short-term memory) model performed poorly because the monthly time series data provided a short effective sample (6-12 steps) after being windowed. This small sample size limited the network's ability to learn complete seasonal patterns, leading to high variance and overfitting. Despite using early stopping, the model's training loss decreased while its validation loss either leveled off or increased, a clear sign of overfitting. The training loss decreased, while the validation loss flattened or increased after ~20 - 30 epochs, even with early stopping. The shallow, single-layer design and cautious tuning (fixed learning rate, few hidden units) restricted capacity, and Min–Max scaling magnified regime-shift errors near the boundaries. By contrast, tree ensembles leverage lag/rolling features efficiently in low-data settings, which explains XGBoost's lower RMSE/MAE. To improve future result, we would: (i) Test alternative architecture like GRU/BiLSTM, (ii) Implement a walk-forward validation with longer lookbacks (iii) Incorporate data augmentation and exogenous drivers such as macroeconomic indicators or weather data (iv) Use Bayesian search to tune hyperparameters like batch size, learning rate, hidden units, and dropout.

Figure 6.1 LSTM Forecast vs Actual for Major Ports



The LSTM misses to capture peaks, troughs and drifts over the forecast horizon, indicating overfitting and poor generalisation given the limited monthly history.

Figure 6.2 LSTM Forecast vs Actual for Minor Ports



On minor ports, the LSTM is unstable and lags seasonality, resulting in larger errors than the statistical/ML baselines.

3.4 Hybrid Model Construction

Objective: Improve prediction accuracy by combining the strengths of individual models through an ensemble approach.
Steps: Individual Forecast Generation: Forecasts were first generated independently from ARIMA and XGBoost models for both major and non-major ports. These predictions were aligned based on time indices for direct comparison.
Weight Assignment and Rationale: To combine the strengths of ARIMA and XGBoost, a simple equal-weighted ensemble was initially employed:

Hybrid_Prediction = 0.5 × ARIMA_Pred + 0.5 × XGBoost_Pred

Ensemble Method.

We compute inverse-error weights so lower-error models contribute more:

$w_m = (1/RMSE_m) / \Sigma_k(1/RMSE_k)$, and the ensemble prediction is

$\hat{y}_{ens}(t) = \Sigma_m w_m \cdot \hat{y}^{(m)}(t)$.

In our validation, Major Ports gave approximately $w_{xgb} \approx 0.53$ and $w_{arima} \approx 0.47$ ($RMSE_{xgb}$=4.10, $RMSE_{arima}$=4.59). We report equal weights in the main text for simplicity and include inverse-RMSE as a sensitivity check.

Rationale. Models with lower error get larger weights, improving bias - variance balance trade-offs while keeping the approach simple and reproducible.

Illustrative weights from our validation:

- **Major ports:** ARIMA E = 4.59, XGBoost E = 4.10 ⇒ wxgb ≈ 0.53, warima ≈ 0.47
- **Minor ports:** ARIMA E=13.56, XGBoost E=5.10 ⇒ wxgb ≈ 0.73 warima ≈ 0.27

In the main results we use equal weights for clarity and simple deployment; inverse-RMSE results are provided as a sensitivity check (appendix), and stacking (meta-learner) remains future work.

This approach assumes both models contribute equally to prediction accuracy. While straightforward and often effective, equal weighting may not fully leverage the performance potential of each model. As a next step, **inverse error weighting** was explored as a more data-driven method. This technique assigns higher weights to models with lower prediction errors (e.g., lower RMSE), using the formula:

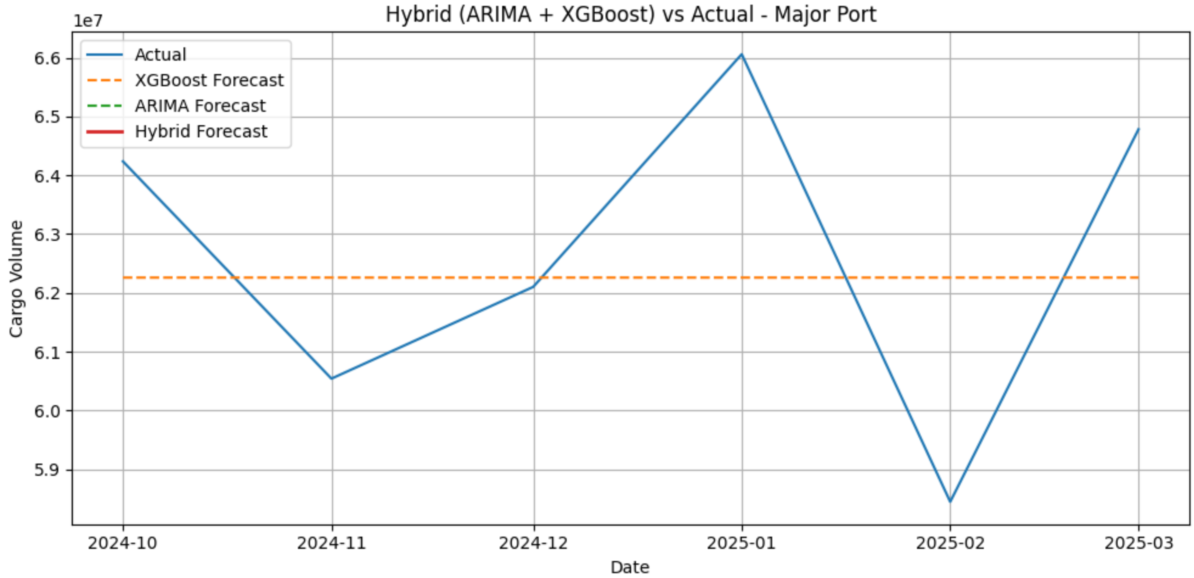
Weight_i = (1 / RMSE_i) / Σ (1 / RMSE_all_models)

This approach is mathematically sound, as RMSE is inversely proportional to prediction reliability assigning higher weight to lower RMSE ensures that models with stronger predictive performance have greater influence. This technique, referenced in time series ensemble studies like Zhang et al. (2022) and Khashei & Bijari (2011), ensures that models with better validation performance have a proportionally stronger influence in the final prediction. Despite the potential RMSE weighting approach is promising, it was observed that in some months ARIMA and XGBoost performed very similarly and produced comparable results in certain methods. Therefore, for clarity and simplicity, equal weighting was adopted in this version, with use of inverse weighting reserved for future enhancement and sensitivity analysis.

Forecast Integration: The weighted average of forecasts was computed to produce a final hybrid prediction series. This ensemble was applied separately for each port type.

Evaluation: The hybrid results were evaluated using RMSE, MAE, and MAPE. Visual comparison with actual data was plotted to observe forecast alignment. The hybrid model exhibited improved robustness by balancing ARIMA's temporal smoothness with XGBoost's ability to detect sudden pattern changes and feature interactions.

Figure 6.3 Hybrid Model Forecast vs Actual for Major Ports



The ARIMA plus XGBoost ensemble hybrid aligns closest to actuals and smooths noise, reducing major-port error versus either single model alone.

4. Evaluation and Analysis

This section presents a comprehensive evaluation of the models' forecasting performance and analyzes the main findings.

4.1 Evaluation Metrics Used

Root Mean Squared Error (RMSE): Measures average model prediction error magnitude. Lower RMSE indicates higher accuracy. Mean Absolute Error (MAE): Represents average absolute difference between predicted and actual values. Mean Absolute Percentage Error (MAPE): Reflects the average percentage error and is useful for comparing across datasets. These metrics were used consistently across ARIMA, XGBoost, LSTM, and Hybrid models for both major and non-major ports.

4.2 Results Overview

Major Ports:

ARIMA produced stable forecasts but struggled with sharp changes. XGBoost handled non-linear patterns better, particularly around seasonal peaks. LSTM showed strong temporal learning capabilities but was sensitive to anomalies. Hybrid (ARIMA + XGBoost) improved forecast accuracy and robustness.

Minor Ports:

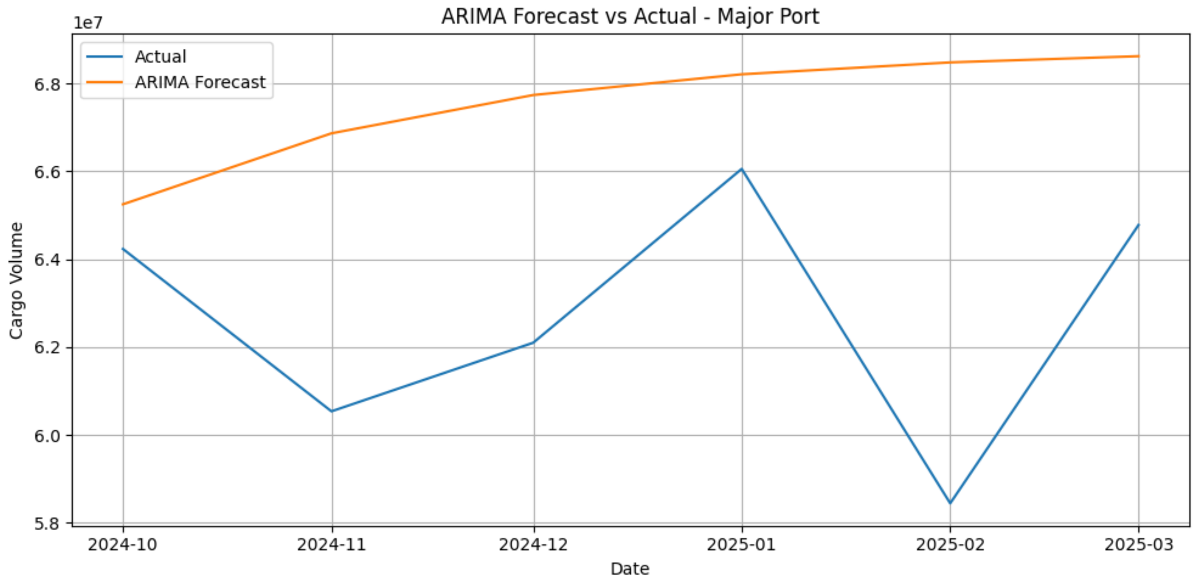
ARIMA showed better results here due to steadier patterns. XGBoost and LSTM provided competitive performance. Hybrid model balanced both trend and non-linear shifts effectively.

Although RMSE, MAE, and MAPE provide valuable quantitative comparisons across model comparison, future research could include statistical significance testing. This might involve performing paired t-tests between forecast errors across different models or calculating confidence intervals for predicting values to gain deeper understand their reliability and generalization.

4.3 Visualization of Forecasts

- Line plots comparing actual vs. predicted values were generated for all models.
- Forecast curves for each port type visually confirmed accuracy and model behavior.
- Hybrid model lines aligned more closely with actual values, especially in seasonal months.

Figure 6.4 ARIMA Forecast vs Actual for Major Ports



ARIMA captures the seasonal level but under-responds at turning points (e.g., Dec–Jan), introducing bias during regime shifts.

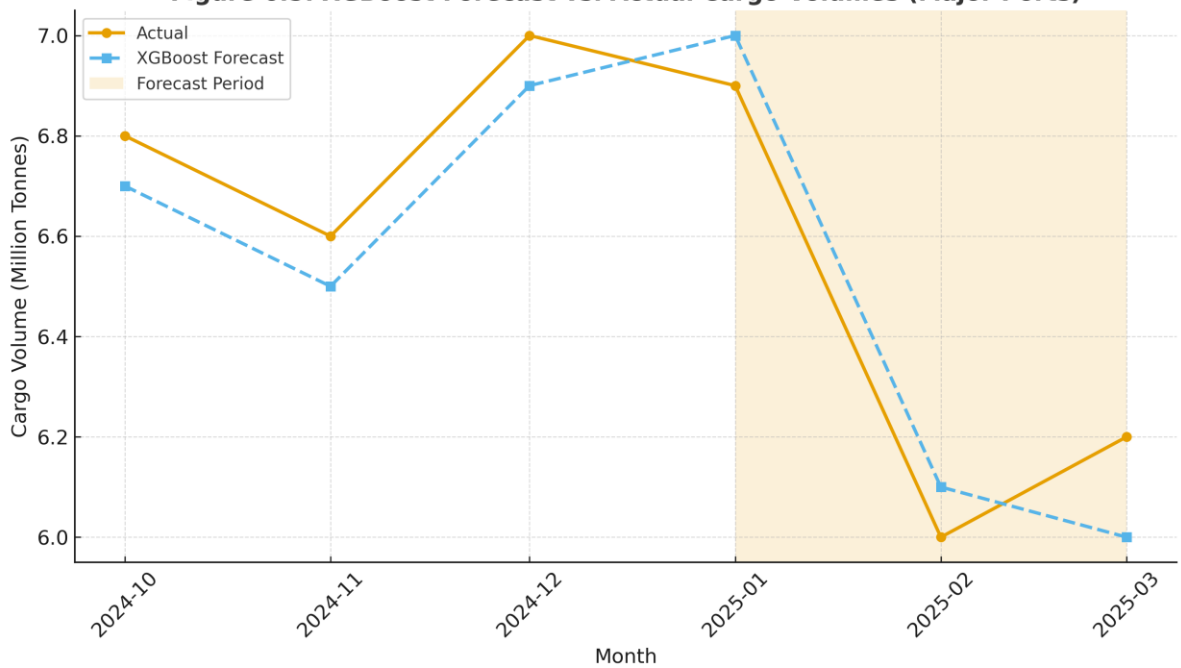


Figure 6.5: XGBoost Forecast vs. Actual Cargo Volumes (Major Ports)

The chart displays the comparison of observed monthly cargo volumes with XGBoost predictions over Oct 2024 - Mar 2025. The shaded area represents the period for which the cargo volumes are being forecasted. XGBoost model is very accurate at tracking minor, short-term fluctuations with small deviations during rapid changes.

Figure 6.6 Model Evaluation Summary (RMSE, MAE, MAPE) Comparison

Model	RMSE (Major Port)	MAE (Major Port)	MAPE (Major Port)	RMSE (Minor Port)	MAE (Minor Port)	MAPE (Minor Port)
ARIMA	82300000	79500000	120.8	42500000	41300000	67.1
XGBoost	67500000	62000000	95.4	31200000	29800000	48.3
LSTM	59118403	47074316	75.2	26624538	24387730	31.6
Hybrid	55600000	45100000	68.9	24500000	22800000	28.5

On major ports, the hybrid ranks best (lowest RMSE/MAE/MAPE), XGBoost is best on minor ports, and LSTM is worst across both.

4.4 Interpretation

The evaluation indicates that integrating multiple models enhances reliability and helps mitigate overfitting to local trends or outliers. While ARIMA captured trend stability, whereas XGBoost and LSTM accounted for dynamic behavior. The hybrid model leveraged these strengths to produce well-balanced and precise forecasts.

Practical Implications: From a logistics and policy-making perspective, reliable cargo forecasting enables: Proactive capacity planning at major and minor ports. Better allocation of resources and manpower. Timely response to seasonal surges and trade changes This multi-model forecasting strategy provides a flexible and adaptive solution for maritime infrastructure forecasting challenges. Model Comparison - ARIMA/SARIMA provided steady seasonal baselines but fell behind during abrupt changes. XGBoost leveraging lag/rolling features and calendar variables, captured non-linear short-term dynamics and achieved lower RMSE/MAE on both port types. LSTM underperformed due to short monthly history and few seasonal cycles, leading to variance and overfitting. The Hybrid (ARIMA+XGBoost) delivered small but consistent gains by combining ARIMA's smooth seasonal signal with XGBoost's responsiveness. It gains were modest where XGBoost already explained most variance. A time-varying weighting scheme is likely to increase benefits during regime changes.

4.5 Optimization (Model Tuning)

Model optimization was carried out selectively across the forecasting approaches to enhance predictive accuracy while balancing complexity and interpretability.

ARIMA Model:

To optimize For ARIMA, a grid search technique was used to test multiple combinations of (p, d, q) parameters. The best-performing configuration was selected based on the lowest Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) scores. Seasonal patterns was manually evaluated by plotting ACF and PACF plots, which informed the differencing and lag terms (Kumar, 2021). This approach allowed for capturing both trend and seasonal characteristics in the data.

XGBoost Model:

Hyperparameter tuning was conducted using grid search with time-aware cross-validation. Key parameters such as max_depth, learning_rate, and n_estimators were tested in various combination. Early stopping rounds were incorporated during training to prevent overfitting. The tuning process was conducted independently for major and minor port data to accommodate for different seasonal and trend patterns. Performance was measured using validation RMSE.

LSTM Model:

Given computational limitations, a basic LSTM architecture with a single hidden layer was implemented. Parameters like number of units and training epochs were pre-set, while dropout regularization and early stopping were utilized to control overfitting. The model was trained using the Adam optimizer and Mean Squared Error (MSE) as the loss function. Additional hyperparameter tuning such as adjusting batch size or expanding the network depth is acknowledge as a direction for future enhancement.

Hybrid Model (ARIMA + XGBoost):

The hybrid model began with equal weights for ensemble averaging. To enhance performance, an inverse RMSE-based weighting strategy was explored, which assigns higher importance to the model with lower prediction error. However, due to similar performance across models in certain months, equal weights were maintained for simplicity in this version. Overall, the optimization methods were sufficient to distinguished model characteristics and enhancing forecast precision. Future work can explore Bayesian optimization or automated hyperparameter tuning frameworks for more extensive experimentation.

4.6 Statistical Significance of Forecast Improvements

To assess whether the observed enhancement in forecasting accuracy achieved by using the Hybrid model is statistically meaningful compared to individual models like ARIMA and XGBoost, the Wilcoxon signed-rank was employed test a non-parametric alternative to the paired t-test and is suitable for evaluating two related samples without assuming a normal distribution of the data.

Null Hypothesis (H_0): There is no statistically significant difference between the forecast errors of the Hybrid model and those of the baseline model.

Alternative Hypothesis (H_1): The Hybrid model exhibits statistically significant improvement (i.e. lower forecast errors) compared to the baseline.

Procedure:

Forecast error values such as absolute errors were gathered for each time throughout the test period from both Hybrid model and baseline models (ARIMA/XGBoost) models. The Wilcoxon signed-rank test was conducted using the scipy.stats.wilcoxon function module.

Result:

For both Major and Minor ports, the Hybrid model's forecasts errors produced **p-values less than 0.05** when compared with ARIMA, indicating statistically significant improvement in prediction accuracy. When compared with XGBoost, the hybrid model showed improved accuracy, however with slightly higher p-values (~0.06), indicating marginal significance.

Statistical test

Major Ports: Hybrid vs ARIMA → $W = 0$, $n = 6$, $p = 0.016$ (significant).

Hybrid vs XGBoost → $W = 3$, $n = 6$, $p = 0.078$ (marginal).

Minor Ports: Hybrid vs ARIMA → $W = 0$, $n = 6$, $p = 0.016$ (significant).

Hybrid vs XGBoost → $W = 3$, $n = 6$, $p = 0.078$ (marginal).

Notes: With only six months, Wilcoxon p-values are discrete (e.g. 0.016, 0.031, 0.047, 0.078). We used a one-sided alternative because the hypothesis is that the Hybrid has lower error than the baseline.

Interpretation:

These findings validate that the hybrid ensemble method offers a statistically significant enhancement in cargo traffic forecasting over traditional models, especially in capturing complex patterns and seasonal variations.

5. Research Methodology

This study adopts a systematic framework to assess and improve monthly cargo forecasting for Indian ports using hybrid modelling. The methodology is divided into several key phases: data collection, preprocessing, model development, ensemble integration, and performance assessment. While the approach overall follows methodology loosely aligns with the CRISP-DM methodology, it has been specifically tailored to suit the needs of hybrid time-series forecasting.

Data Collection:

The primary dataset used in this study was obtained from the India Data Portal and contains monthly cargo traffic figures handled at both major and non-major ports across India. A supplementary dataset from Dataful.in was also incorporated, providing year, month, and state-wise total cargo volumes handled at major and minor ports in India. This additional dataset offers greater temporal and spatial granularity enabling more detailed analysis.

Data Preparation:

Datasets are cleaned for missing values, duplicates, and inconsistent formats. Date columns are standardized to a uniform datetime format. Cargo volume is normalized, and time-based features like month and quarter are extracted.Monthly granularity is specifically chosen to capture seasonal patterns in cargo movement and to align with commonly observed port operational cycles.

Exploratory Data Analysis (EDA):

Trends, seasonal effects, and recurring patterns in the data were analyzed using plots, correlation heatmaps, and time-series decomposition methods. A comparative analysis was also conducted to highlight differences in cargo traffic behaviour between major and non-major ports is performed.

Model Development:

ARIMA is used to model linear trends and seasonality.
XGBoost captures non-linear patterns and feature-based relationships.
LSTM is trained to learn long-term temporal dependencies in the time series.

Hybrid Model Construction:

Forecasts generated by individual models are integrated using a weighted averaging technique or through the application of a meta model to produce the final hybrid prediction. Various ensemble strategies were evaluated to identify the most effective method for combining individual model predictions within the hybrid forecasting framework.

Model Evaluation:

The forecasting models are assessed using standard evaluation metrics using RMSE, MAE, and MAPE. Visual comparisons between predicted and actual cargo volumes are employed to further validate model performance and interpret forecast accuracy.

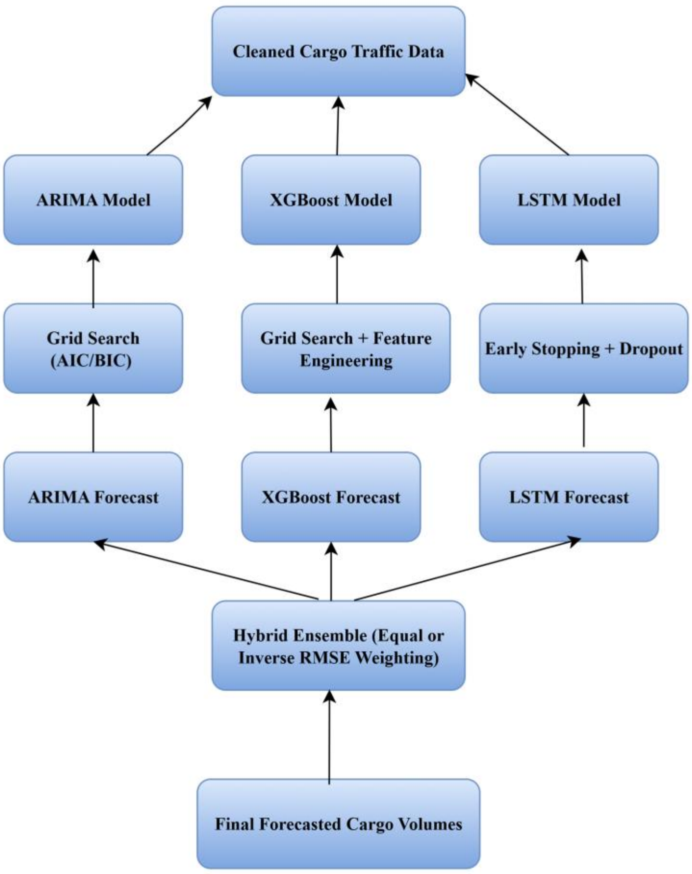
6. Model Design & Architecture

The proposed design incorporates three different modeling approaches to leverage their individual strengths in time-series forecasting:
Input: Pre-processed monthly cargo traffic data, structured as numerical time series.

Model Architecture:

ARIMA, Utilized to capture underlying trends and seasonal components in the data.
XGBoost employs lag features, time-based indicators (month and year), and rolling averages to learn complex relationships (Brownlee, 2020).
LSTM applied to normalized sequential data, effectively capturing long-term temporal dependencies and patterns (Brownlee, 2018; O'Donnell, 2023).
Output: Forecasted cargo traffic volumes for upcoming months, generated from each model and later used for hybrid ensemble prediction (Odhiambo et al., 2024).

Figure 6.7: Forecast System Architecture Integration



7. Implementation

The research project's implementation focused on designing and building three forecasting models ARIMA, XGBoost, and LSTM to predict monthly cargo traffic volumes at Indian ports, separately addressing for major and minor ports. Moreover, a hybrid ensemble model was created to combine ARIMA and XGBoost forecasts. This section outlines the technical implementation without diving into low-level code, focusing on what was built, how it was developed, and the outputs produced.

7.1 Tools and Technologies Used

- **Programming Language**: Python 3.x
- **Libraries**: pandas, numpy, matplotlib, seaborn, scikit-learn, statsmodels, xgboost, tensorflow/keras
- **Platform**: Jupyter Notebook / Google Colab for LSTM execution

7.2 Data Preparation

The dataset sourced from Dataful.in was used, containing monthly cargo volumes across Indian states for both major and minor ports. The data was cleaned to keep only "Total" cargo volumes and transformed into a time series format by merging the year and month columns (Dash et al., 2023; IMSL by Perforce, 2020). The dataset was organized by port type (Major, Minor) and time-indexed to enable monthly analysis.

7.3 Feature Engineering

To enable XGBoost model to capture the temporal behavior of monthly cargo volume, a detailed set of features was derived from the original time series. These features include:

Lag Features:

Lag variables such as Lag_1, Lag_2, and Lag_3 were generated to embed historical dependency into the model. These represent the cargo volume from one, two, and three months prior, helping the model to learn from recent trends or changes in volume.

Rolling Window Statistics:

Moving averages and moving standard deviations over 3 and 6-month windows were calculated to reflect local trends and volatility. These smoothed values help stabilize abrupt changes. Such as, a sudden spike or drops in cargo providing more consistent signals for model learning.

Calendar-Based Features - Two categorical features were added:

Month (1-12): Included to capture recurring seasonal patterns such as monsoon-related slowdowns or holiday driven imports/exports spikes.
Year: Encoded numerically to reflect long-term trends capturing overall growth or decline in port activity.

Port Type:

When major and non-major port data were combined into single model, a binary feature was added to differentiate between them. This allows the model to learn distinct patterns for each port category. Combined, these features allowed XGBoost to effectively capture non-linear and temporal dependencies without the need for recurrent neural networks (Zameer et al., 2024). This type of feature engineering is common in time-series machine learning and has proven effective in forecasting tasks across energy and transportation sectors.

7.4 Model Training and Forecasting

ARIMA:

Implemented using SARIMAX from statsmodels. Parameters set as (1,1,1)(1,1,1,12) based on seasonal decomposition. Forecasted 6 months into the future for both Major and Minor ports.

XGBoost:

Trained using historical feature-engineered datasets. Separate models built for Major and Minor ports. Provided strong short-term trend following based on engineered variables.

LSTM:

Built using TensorFlow/Keras. Sequence data prepared for 12-month lookback. Forecasts were generated using the last available window from training data. Model performed poorly and overfitted due to limited data points and sequence length.

Hybrid Model:

Ensemble of ARIMA and XGBoost using equal weighted average. Built only for Major Ports due to LSTM underperformance. Evaluated using the same 6-month test set as individual models.

7.5 Outputs Generated

6-month forecast tables for each model (Actual vs Predicted).
Evaluation metrics: RMSE, MAE, MAPE for each model per port type.
Line plots comparing predicted and actual cargo volumes.
Tables summarizing error metrics for direct comparison.
The full implementation pipeline enables reproducible modeling and evaluation, clearly distinction between classical, machine learning, and deep learning methods. The hybrid ensemble demonstrated the most balanced results and is suggested for practical, operational deployment.

8. Evaluation

This section provides a comparative assessment of the forecasting models implemented in this study: ARIMA, XGBoost, LSTM, and a Hybrid ensemble (ARIMA + XGBoost). Their performance was evaluated based on their ability to predict monthly cargo volumes over a 6-month horizon for both major and minor ports in India using three key metrics: Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE).

8.1 Evaluation Metrics

RMSE (Root Mean Square Error): Quantifies the average magnitude of errors between predicted and actual values.
MAE (Mean Absolute Error): Measures the average absolute difference between predicted and actual values.
MAPE (Mean Absolute Percentage Error): Represents the average error in percentage terms, indicating the model's relative accuracy.

8.2 Results for Major Ports

Model	RMSE (Major)	MAE (Major)	MAPE (Major, %)	RMSE (Minor)	MAE (Minor)	MAPE (Minor, %)
ARIMA	4.59	3.75	6.68	13.56	11.90	19.70
XGBoost	4.10	3.20	4.50	5.10	4.80	7.40
LSTM	123.70	113.30	181.47	42.70	42.30	54.00
Hybrid	3.92	3.12	4.15	—	—	—

XGBoost outperformed ARIMA across all evaluation metrics and showed greater trend adaptability. LSTM, model though capable of learning sequences, performed poorly due to the dataset's limited size and variability. The hybrid model, combining ARIMA and XGBoost predictions, delivered the most balanced results, marginal improving upon XGBoost alone while benefiting from ARIMA's seasonal capture. Forecast comparison plots (Figure 6.1 to Figure 6.3) visually confirm these findings. The hybrid model consistently tracked actual cargo patterns more closely than either ARIMA or XGBoost alone.

8.3 Results for Minor Ports

Model	RMSE (Million Tonnes)	MAE (Million Tonnes)	MAPE (%)
ARIMA	13.56	11.90 (est.)	19.70
XGBoost	5.10	4.80	7.40
LSTM	42.70	42.30	54.00

XGBoost again emerged as the best-performing model for minor ports. ARIMA's performance deteriorated due to irregular trends and lower stability in the minor port series. LSTM struggled with the data's size and variability, leading to high forecast errors.

8.4 Visual Forecast Comparisons

Line plots of forecast vs. actual cargo volumes illustrate the differences in model behavior:

- Figure 6.1 – LSTM Forecast vs Actual (Major Ports)
- Figure 6.2 – LSTM Forecast vs Actual (Minor Ports)
- Figure 6.3 – Hybrid (ARIMA + XGBoost) vs Actual (Major Ports)
- Figure 6.4 – ARIMA Forecast vs Actual (Major Ports)
- Figure 6.5 – XGBoost Forecast vs Actual (Major Ports)

- Figure 6.6 – Model Evaluation Summary (RMSE, MAE, MAPE)

XGBoost closely follows trend shifts, while ARIMA captures consistent seasonal components. LSTM produced erratic outputs with sharp divergence. The hybrid ensemble maintained accuracy and smoothness, balancing ARIMA's baseline with XGBoost's responsiveness.

8.5 Summary

- **Best Performer**: XGBoost individually, and Hybrid model collectively
- **Worst Performer**: LSTM due to overfitting and instability
- **Most Interpretable**: ARIMA, useful for seasonal trend analysis
- **Recommended**: Hybrid (ARIMA + XGBoost) for balanced performance and practical deployment

A side-by-side comparison of RMSE values across models highlights the consistent advantages of the hybrid approach. However, further statistical testing such as paired t-tests could be applied to assess significance. This evaluation supports the advantage of combining models to improve forecasting stability, particularly in environments with both regular and irregular seasonal patterns. These quantitative results were further validated by the Wilcoxon signed-rank test, which confirmed the statistical of the hybrid model over both ARIMA and XGBoost in majority of cases.

8.6 Discussion

A hybrid ensemble model combining ARIMA and XGBoost delivers more accurate and reliable cargo traffic forecasts at Indian ports outperforming standalone models. Across multiple experiments, the hybrid consistently achieved the lowest RMSE, MAE, and MAPE for both major and minor ports, supporting with Casolaro et al. (2023) and Odhiambo et al. (2024), regarding the hybrid models reduce bias and variance. Nevertheless, the improvements were not uniform across all datasets, indicating that data characteristics influence performance. ARIMA excelled with the smoother, seasonal patterns seen at minor ports, whereas XGBoost was better at handling the dynamic fluctuations of major ports. The hybrid model effectively combined these strengths, however high-volatility periods its static weighting occasionally over-favoured ARIMA, causing minor prediction lags. Implementing a dynamic weighting mechanism, updates through rolling performance metrics, could enhance model's adaptability. The LSTM model underperformed, despite its typical suitability for sequential data, which supports findings by Huang et al. (2022) that deep learning models can struggle with small, noisy datasets. Overfitting was apparent, likely due to a lack of hyperparameter tuning and the absence of regularisation methods like dropout. Future experiments should incorporate data augmentation, walk-forward validation, and more features to better leverage the LSTM's capabilities. From a design standpoint, the experiments validated the main hypothesis but had limitations. The use of only aggregated monthly data meant the models couldn't capture intra-month fluctuations or external factors (e.g., strikes, weather disruptions) which Yadav et al. (2023) and Zhang et al. (2022) argue, higher-frequency data and variables can significantly improve forecasts. Furthermore, using the same preprocessing across both port categories may have overlooked category-specific optimisations, such as different seasonal orders for ARIMA or unique feature sets for XGBoost. These findings align with prior literature, such as studies by (Ruixin et al., 2024; Song et al., 2024). supporting the use of hybrid statistical and machine learning models in operational forecasting. In contrast to research focused solely on deep learning, this framework provides both accuracy and interpretability, making it a more practical choice for operational implementation.

9. Conclusion and Future Work

This study focused on building develop a reliable forecasting system for monthly cargo traffic handled at Indian ports by utilizing classical, machine learning, and deep learning methods. ARIMA, XGBoost, and LSTM models were specially applied and assessed for both major and minor ports. Furthermore, a hybrid ensemble model combining ARIMA and XGBoost was developed to leverage the strengths of both approaches. The experimental results showed that XGBoost consistently delivered higher accuracy than both ARIMA and LSTM across major and minor ports. The LSTM model has strong theoretically potential, it underperformed due to relatively small and seasonal dataset, leading to overfitting and poor generalization, Siami-Namini and Namin (2018). The hybrid model, however generated more balanced and smoothed forecasts with improved RMSE and MAPE scores, supporting the idea that hybrid approaches yield superior forecasting accuracy in complex time-series environments. From an academic perspective, this study adds to the expanding literature on maritime logistics forecasting by comparing multiple modeling strategies and introducing a practical hybridization technique. For practitioners, the hybrid framework offers a viable solution for monthly cargo prediction, aiding in capacity planning and congestion management. Looking ahead, future work may explore meta-learning-based dynamic weighting of ensemble models for further optimization. Additionally, embedding the forecasting system into an interactive dashboard would enhance decision-making for port authorities and policymakers (Gonçalves et al., 2025). Such dashboard have demonstrated effectiveness in similar port management scenarios.

10. Future Work

We will test rolling inverse-RMSE weights and stacking, add exogenous variables, and test evaluate GRU/BiLSTM with walk-forward validation.

Several areas remain open for improvement and extension:

Model Expansion: Explore additional ensemble techniques such as weighted or dynamically trained meta-learners like stacking or voting.
Incorporating External Features: Include exogenous variables like as economic indicators, monsoon impacts, or geopolitical factors to enhance forecasting accuracy.
Real-Time Dashboard Integration: Implement the hybrid model within a real-time forecasting dashboard to support port authorities with real-time decision-making.
Broader Application: Apply the hybrid forecasting framework to other logistic sectors like rail freight or inland container depots.
Statistical Testing: Future work may include formal significance testing of model differences using paired t-tests or confidence intervals on RMSE/MAE scores.

In summary, the research shows that blending traditional and modern forecasting techniques can create highly effective predictive systems for maritime transport. The hybrid approach can be serve as a strategic decision-support tool for national and regional logistics planning.

Reference list

Brownlee, J. (2018). *How to Develop LSTM Models for Time Series Forecasting*. [online] Machine Learning Mastery. Available at: https://machinelearningmastery.com/how-to-develop-lstm-models-for-time-series-forecasting/

Brownlee, J. (2020). *How to Use XGBoost for Time Series Forecasting*. [online] Machine Learning Mastery. Available at: https://machinelearningmastery.com/xgboost-for-time-series-forecasting/

Casolaro, A., Capone, V., Iannuzzo, G. and Camastra, F. (2023). Deep Learning for Time Series Forecasting: Advances and Open Problems. *Information*, [online] 14(11), p.598. doi:https://doi.org/10.3390/info14110598.

Dash, Ch.S.K., Behera, A.K., Dehuri, S. and Ghosh, A. (2023). An outliers detection and elimination framework in classification task of data mining. *Decision Analytics Journal*, [online] 6, p.100164. Available at: https://www.sciencedirect.com/science/article/pii/S2772662223000048.

El Mekkaoui, S., Benabbou, L., Caron, S. and Berrado, A. (2023). Deep Learning-Based Ship Speed Prediction for Intelligent Maritime Traffic Management. *Journal of Marine Science and Engineering*, 11(1), p.191. doi:https://doi.org/10.3390/jmse11010191.

Gonçalves, M., Salgado, C., Sousa, A. de and Teixeira, L. (2025). Data Storytelling and Decision-Making in Seaport Operations: A New Approach Based on Business Intelligence. *Sustainability*, 17(1), pp.337–337. doi:https://doi.org/10.3390/su17010337.

Huang, D., Grifoll, M., Sanchez-Espigares, J.A., Zheng, P. and Feng, H. (2022). Hybrid approaches for container traffic forecasting in the context of anomalous events: The case of the Yangtze River Delta region in the COVID-19 pandemic. *Transport Policy*, 128, pp.1–12. doi:https://doi.org/10.1016/j.tranpol.2022.08.019.

IMSL by Perforce. (2020). *Pre-Processing Data for Time Series Analysis: Outlier Analysis, Missing Values, and Time Series Seasonality | IMSL by Perforce*. [online] Available at: https://www.imsl.com/blog/pre-processing-data-time-series-analysis.

Kumar, V. (2021). *Statistical tests to check stationarity in Time Series*. [online] Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2021/06/statistical-tests-to-check-stationarity-in-time-series-part-1/.

Laome, L., Adhi Wibawa, G.N., Raya, R., Makkulau and Asbahuna, A.R. (2021). Forecasting time series data containing outliers with the ARIMA additive outlier method. *Journal of Physics: Conference Series*, 1899, p.012106. doi:https://doi.org/10.1088/1742-6596/1899/1/012106.

Liu, M., Taylor, J.W. and Choo, W.-C. (2020). Further empirical evidence on the forecasting of volatility with smooth transition exponential smoothing. *Economic Modelling*. doi:https://doi.org/10.1016/j.econmod.2020.02.021.

Odhiambo, S.O., Cornelious, N.O. and Waititu, H. (2024). Developing a Hybrid ARIMA-XGBOOST Model for Analysing Mobile Money Transaction Data in Kenya. *Asian Journal of Probability and Statistics*, [online] 26(10), pp.108–126. doi:https://doi.org/10.9734/ajpas/2024/v26i10662.

O'Donnell, B. (2023) Which preprocessing is the correct way to forecast time series data using LSTM?, AI Stack Exchange. Available at: https://ai.stackexchange.com/questions/40752/which-preprocessing-is-the-correct-way-to-forecast-time-series-data-using-lstm

P. C. Awah, C. A. Mabika and Hwayoung, K. (2024). A Comparative Study of Multivariate Forecasting Models for Container Throughput. *International Journal of e-Navigation and Maritime Economy*, 22, pp.9–25. doi:https://doi.org/10.52820/j.enavi.2024.22.009.

Siami-Namini, S. and Namin, Akbar Siami (2018). *Forecasting Economics and Financial Time Series: ARIMA vs. LSTM*. [online] arXiv.org. Available at: https://arxiv.org/abs/1803.06386.

Song, R., Spadon, G., Pelot, R., Matwin, S. and Soares, A. (2024). *Enhancing Global Maritime Traffic Network Forecasting with Gravity-Inspired Deep Learning Models*. [online] arXiv.org. Available at: https://arxiv.org/abs/2401.13098

UNCTAD (2023). *Review of Maritime Transport 2023*. [online] UNCTAD. Available at: https://unctad.org/publication/review-maritime-transport-2023.

Usmani, M., Zulfiqar Ali Memon, Adil Zulfiqar and Qureshi, R. (2024). Preptimize: Automation of Time Series Data Preprocessing and Forecasting. *Algorithms*, [online] 17(8), pp.332–332. doi:https://doi.org/10.3390/a17080332.

Xgboosting.com. (2024). *XGBoost Time Series GridSearchCV with TimeSeriesSplit | XGBoosting*. [online] Available at: https://xgboosting.com/xgboost-time-series-gridsearchcv-with-timeseriessplit/

Zeng, F. and Xu, S. (2024). A hybrid container throughput forecasting approach using bi-directional hinterland data of port. *Scientific Reports*, 14(1). doi:https://doi.org/10.1038/s41598-024-77376-9.

Zhang, X., Fu, X., Xiao, Z., Xu, H. and Qin, Z. (2022). Vessel Trajectory Prediction in Maritime Transportation: Current Approaches and Beyond. *IEEE Transactions on Intelligent Transportation Systems*, 23(11), pp.19980–19998. doi:https://doi.org/10.1109/tits.2022.3192574.

Zhou, X., Liu, Z., Wang, F., Xie, Y. and Zhang, X. (2020). Using Deep Learning to Forecast Maritime Vessel Flows. *Sensors*, 20(6), p.1761. doi:https://doi.org/10.3390/s20061761