JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 30 No. 12, pp. 113-130, December 2025
https://doi.org/10.9708/jksci.2025.30.12.113

Short-term Forecasting of Handysize freight rates and DNN architecture for time series classification

Sang-Hyeok Lee*, Changho Son**

*Researcher, Dept. of Big Data, Small Enterprise and Markets Service, Daejeon, Korea
**Professor, Dept. of AI·System Science, Korea Army Academy, Yeong-Cheon, Korea

[Abstract]

This study forecasts weekly changes in Handysize freight rates using a simple up-or-down prediction framework. We utilize daily Baltic Handysize Index (BHSI) data from 2006 to 2024 to construct five-day weekly patterns, subsequently predicting whether the average rate for the following week will be higher or lower than that of the current week. To accomplish this, we compare a comprehensive set of models, including 48 compact deep neural networks—comprising multilayer perceptrons, fully convolutional, and residual architectures—as well as standard benchmarks such as Bi-LSTM, Transformer, support vector machine, and random forest classifiers. Out-of-time tests conducted under various market conditions demonstrate that a residual network with moderate sensitivity to intra-week ordering delivers the most accurate and stable forecasts, yielding well-calibrated probabilities. These findings indicate that weekly freight patterns encompass exploitable directional information and that the proposed residual network can function as an effective tool for chartering decisions, freight hedging, and market monitoring within the dry bulk shipping industry.

▸ **Key words**: Residual network, Weekly freight-rate forecasting, Baltic Handysize Index, Time series classification, Deep neural networks, Dry bulk shipping market, Probability calibration, Out-of-time validation

[요 약]

이 연구는 Handysize 운임의 주간 변화를 단순한 상승·하락 예측(framework)에 기반해 분석한다. 2006년부터 2024년까지의 일별 Baltic Handysize Index(BHSI) 데이터를 활용해 5일치 주간 패턴을 구성하고, 다음 주 평균 운임이 현재 주 평균 운임보다 높을지 낮을지를 예측한다. 이를 위해 다층 퍼셉트론, 완전 합성곱 신경망, 잔차 신경망 구조로 이루어진 48개의 컴팩트한 심층신경망과 더불어 Bi-LSTM, Transformer, 서포트 벡터 머신, 랜덤 포레스트 분류기와 같은 표준 벤치마크 모형을 포괄적으로 비교한다. 서로 다른 시장 국면에서 수행한 out-of-time 검증 결과, 주중 순서 변화에 대해 적정 수준의 민감도를 지닌 잔차 신경망이 가장 정확하고 안정적인 예측 성능을 보이며, 동시에 잘 보정된(probability calibration) 예측 확률을 산출하는 것으로 나타났다. 이러한 결과는 주간 운임 패턴에 활용 가능한 방향성 정보가 내재해 있음을 시사하며, 제안한 잔차 신경망이 건화물선 시장에서 용선 의사결정, 운임 헤지, 시장 모니터링을 위한 효과적인 도구로 기능할 수 있음을 보여준다.

▸ **주제어**: 잔차 신경망, 주간 운임 예측, Baltic Handysize Index, 시계열 분류, 심층신경망, 건화물선 시장, 확률보정, 시점외 검증

• First Author: Sang-Hyeok Lee, Corresponding Author: Changho Son
*Sang-Hyeok Lee (sanghyeoke@semas.or.kr), Dept. of Big Data, Small Enterprise and Markets Service
**Changho Son (c13981@kaay.ac.kr), Dept. of AI·System Science, Korea Army Academy
• Received: 2025. 09. 29, Revised: 2025. 10. 23, Accepted: 2025. 12. 15.
• This paper is an extended version of the work presented at the Korea Society of Computer and Information Summer Conference 2025, entitled "Short-term Forecasting of Handysize Freight Rates and DNN Architecture for Time Series Classification." (Best Paper Award)

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

Forecasting freight rates in the dry bulk market has long been a central issue in maritime economics, as short-term fluctuations directly influence chartering decisions and investment outcomes. Among dry bulk carriers, Handy vessels — ships transporting 30,000 to 35,000 tons of bulk commodities such as grain and minerals — make up approximately 20% of the global dry bulk fleet [1,2]. Given their importance, Clarksons Research publishes market information on Handy vessels through the Baltic Handysize Index (BHSI) [4]. As shown in Fig. 1, BHSI exhibits abrupt intra-week fluctuations that occasionally exceed US $1,000 per day. Such volatility implies that shippers, brokers, and investors can gain or lose substantial capital simply depending on when they fix a Handy charter. Consequently, short-term (weekly) forecasting has gained increasing significance in both contemporary maritime practice and academic research [4-6].

Fig. 1. Time series of the daily Baltic Handysize Index (BHSI), illustrating substantial short-term volatility that motivates our weekly directional forecasting approach

Although short-term freight prediction is desirable, obtaining sufficiently high-frequency explanatory variables is extremely difficult. For example, seaborne volumes of raw materials such as coking coal or iron ore are available only on a monthly basis through Clarksons Research [4]. Daily or weekly data on commodity-specific loads, port congestion, or logistics constraints are generally not publicly observable. Thus, while exogenous drivers likely influence Handy rates, constructing a feature space with daily/weekly auxiliary predictors is infeasible. This dataset limitation inevitably leads to a univariate forecasting setup. While this simplifies the modeling design, it should be regarded as a limitation of the present study rather than a conceptual choice; future work could incorporate higher-frequency external predictors when they become obtainable.

Previous research on freight rate forecasting has primarily utilized regression models to predict specific future values of indices such as the Baltic Dry Index (BDI) or the Baltic Handysize Index (BHSI). Yang and Mehmed (2019) employed an Artificial Neural Network (ANN) and demonstrated that the 'Non-linear AutoRegressive with eXogenous' inputs model enhances accuracy when incorporating Forward Freight Agreements [8]. Zeng et al. (2016) combined empirical mode decomposition (EMD) with an artificial neural network (ANN) to forecast the BDI, outperforming both Autoregressive Integrated Moving Average (ARIMA) and Vector Autoregression (VAR) models [9]. Kim et al. (2019) employed Long Short-Term Memory (LSTM) architectures and demonstrated that incorporating exogenous variables improves performance [10]. More recently, Feng (2022) analyzed container freight rate levels using macroeconomic and supply-demand variables, reporting that machine-learning models such as random forests outperform benchmark ARIMA and VAR specifications [11]. Hirata and Matsuda (2022) examined the Shanghai Containerized Freight Index by comparing ARIMA, LSTM, and random forest models, and found that forecasts based on LSTM achieve the highest predictive accuracy, supporting hedging strategies against freight-rate volatility [12]. Li et al. (2023) proposed a hybrid model based on Bidirectional Long Short-Term Memory (Bi-LSTM) that decomposes shipping indices into more regular sub-series and subsequently forecasts each component using an optimized Bi-LSTM network, achieving significant improvements over conventional LSTM and tree-based baselines [13]. Han et al. (2024) further

connected maritime shipping indices with Chinese financial market indicators and demonstrated that LSTM-based deep learning models more effectively capture nonlinear co-movements between freight rates and financial variables than traditional VAR models [14]. Overall, this literature primarily treats freight-rate prediction as a form of 'point forecasting' [15] for future index levels, with limited focus on directional or distributional forecasting.

However, three important gaps remain in the existing freight rate forecasting literature.

• The existing freight rate literature largely focuses on precise numeric (point) forecasts of index levels rather than directional information, even though direction is often more relevant for real-time decision making under uncertainty.

• Short-term predictability at the weekly frequency, and in particular the role of structured noise within such short time windows, has received very little attention.

• Most papers emphasize statistical accuracy measures, without systematically linking forecast performance to practical economic consequences for shippers, brokers, or policymakers.

To address these gaps, the present study reformulates short-term freight forecasting as a time-series classification problem that focuses on weekly direction rather than point predictions. Instead of forecasting exact index levels, this paper predicts whether the average freight rate in the upcoming week will be higher or lower than that of the current week, with each calendar week represented as a five-dimensional vector of Monday–Friday BHSI values. This fixed-length weekly pattern is represented as a small 1×5 array, serving as the fundamental input for our classification models and establishing a direct connection between directional predictive signals and economically meaningful trading and hedging decisions within the weekly prediction period.

The contributions of this study are as follows:

• We reformulate weekly Handysize freight-rate prediction as a binary time-series classification task, enabling models to learn directional patterns rather than numeric index values.

• We construct and compare 48 compact deep neural network variants—covering Multilayer Perceptron, Fully Convolutional Network, and Residual Network architectures—to investigate how architectural inductive biases affect directional predictability under limited weekly observations.

• We develop a residual architecture with calibrated phase sensitivity that selectively responds to informative temporal reorderings within a week. This design preserves useful structural asymmetries that invariant models, such as Transformers, fail to exploit.

• We introduce an evaluation pipeline combining standard classification measures (accuracy, balanced accuracy, Macro F1), cost-sensitive error analysis, and probability calibration metrics. This allows us to assess not only prediction correctness but the reliability of the model's confidence scores.

• Using extensive out-of-sample validation across multiple market regimes, we demonstrate that ReN consistently outperforms the Bi-LSTM, Transformer, SVM, and Random Forest baselines. These results confirm that weekly directional signals in Handysize freight rates are persistent, recoverable, and structurally embedded in weekday rate patterns.

The remainder of this paper is organized as follows. Section II details the proposed classification framework, data preprocessing procedure, and the deep neural network architectures evaluated in this study. Section III reports the empirical results, including comparative performance analysis, out-of-time validation, and interpretive insights into model behavior. Section IV summarizes the main findings and discusses potential directions for future research.

II. The Proposed Scheme

Since the forecasting objective in this study

involves a binary time-series classification problem for directional prediction, we first define the data representation prior to presenting the algorithms. For each calendar week $t = 1,2,\cdots T$, each observation comprises a five-dimensional vector representing daily values from Monday through Friday $\boldsymbol{x}_t$ expressed as the Eq. (1),

$$\boldsymbol{x}_t := (x_{t,Mon}, x_{t,Tue}, x_{t,Wed}, x_{t,Thu}, x_{t,Fri}) \in X, \quad (1)$$

where := denotes 'is defined as'. The binary target $\hat{y_t} \in Y := \{0,1\}$ encodes the direction of the subsequent week shown in the Eq. (2),

$$\hat{y_t} = \begin{cases} 1, \text{if} \frac{1}{5}\sum_{i=Mon}^{Fri} x_{t,i} > \frac{1}{5}\sum_{i=Mon}^{Fri} x_{t+1,i}, \\ 0, otherwise. \end{cases} \quad (2)$$

Let $f : X \rightarrow Y$ denote the predictive mapping; given $\boldsymbol{x}_t$, it outputs $f(\boldsymbol{x}_t) = \hat{y_t}$. This labeling captures directional movement and defines the classification task studied here.

1. DNN for the time series classification

Deep neural networks for time-series classification were first systematized by Wang et al. (2017) [16]. However, the current task involves a notably limited number of weekly observations relative to the model's potential capacity, which may lead to unstable learning dynamics and unreliable model selection. As demonstrated by Lee (2023), this setting frequently yields highly volatile validation-accuracy curves, complicating the identification of architectures that generalize consistently [17]. To overcome this limitation, we employ compact architectures specifically designed for limited-data settings. Specifically, we assess three families of deep neural networks—Multilayer Perceptrons (MLP), Fully Convolutional Networks (FCN), and Residual Networks (ReN)—each developed to extract directional patterns from five-day weekly segments. Although structurally diverse, all models $f(x)$ employ a common formulation comprising stacked transformation blocks $g_1, \cdots, g_l, \cdots, g_L$ followed by a softmax classifier $S_{oft}m_{ax}$ or global average pooling GAP is expressed as the Eq. (3),

$$f(x) = \begin{cases} S_{oft}m_{ax} \circ g_L \circ \cdots g_l \cdots \circ g_1(x) \\ S_{oft}m_{ax} \circ GAP \circ g_L \circ \cdots g_l \cdots \circ g_1(x). \end{cases} \quad (3)$$

Each block g_l is a dense operator, either a convolutional or residual module, depending on the architectural class. The objective is to compare the effects of various structural biases—point-wise, convolutional, and residual—on directional predictability under conditions of severe data sparsity.

To facilitate the reader's comprehension of the models' high-level behavior, we first summarize their functional roles prior to presenting further details. MLPs stack dense layers and process each time point independently; they are well suited for learning hierarchical relationships but tend to smooth out fine-grained curvature patterns observed in weekly sequences [18].

In contrast, FCNs utilize one-dimensional convolutions over the weekly window to extract local slopes, peaks, and curvature motifs [19], while global average pooling ensures translation invariance across the five-day pattern, rendering FCNs particularly well-suited for short-term shape detection [20].

ReNs advance the approach by integrating residual blocks with skip connections that capture perturbations around an identity mapping. This design maintains the original waveform structure of the weekly segment while facilitating deeper hierarchical representations, and this inductive bias results in more stable shape discrimination compared to both MLPs and FCNs [21].

Each architectural family—MLP, FCN, and ReN—was constructed by stacking a limited number of substructures, with each model comprising up to three layers. To systematically investigate the impact of architectural depth and block composition on directional prediction performance, we constructed 16 variants each of MLP, FCN, and ReN, resulting in a total of 48 distinct architectures. These variants differed solely in the

depth and configuration of components such as fully connected layers, rectified linear activation units, one-dimensional convolutional layers, batch normalization layers, dropout regularization, and residual blocks. All architectures were deliberately kept compact to accommodate the constraints of the limited weekly dataset.

Comprehensive architectural specifications for all models are provided in Appendix A. Appendix A details the exact layer configurations and operator definitions, allowing the main text to focus on conceptual comparisons without lengthy formulas.

To compare architectures exhibiting fluctuating learning curves, we employ the stability-aware metric originally proposed by Lee (2023) [17]. For a given model f, let $Acc_i^{train}(f)$ and $Acc_i^{test}(f)$ denote the classification accuracy at epoch i on the training and test sets, respectively, computed over a fixed stabilization window W. From these sequences, we calculate the mean and standard deviation of accuracy for each split, and define the metric j_{NN} as the Eq. (4).

$$j_{NN}(f) := \min\left\{ \underset{i\in W}{mean}\left(Acc_i^{train}(f)\right) - \underset{i}{std}\left(Acc_i^{train}(f)\right), \underset{i\in W}{mean}\left(Acc_i^{test}(f)\right) - \underset{i\in W}{std}\left(Acc_i^{test}(f)\right) \right\}. \quad (4)$$

Intuitively, a higher mean accuracy indicates greater predictive power, while a lower standard deviation suggests that the model's performance remains consistently stable across epochs rather than exhibiting sporadic fluctuations. By taking the minimum of the train-based and test-based scores, j_{NN} penalizes models that demonstrate stability on only one split while rewarding architectures that simultaneously achieve both high accuracy and low volatility. In this way, the metric emphasizes consistently robust models over those that attain high peak accuracy in only a limited number of epochs.

2. Comparative Algorithms

We compared the predictive performance of the model generated by our algorithm with that of established baseline models. First, to compare our classification method with alternative approaches, we selected Support Vector Machines (SVMs), which have consistently exhibited robust performance in short-term financial forecasting [22] and have been extensively applied to financial market time series [23-25], establishing them as a suitable benchmark.

Second, to compare with sequence-model baselines, we employed a variant of Long Short-Term Memory (LSTM): a bidirectional LSTM (Bi-LSTM) enhanced with a lightweight temporal attention head [26]. Processing the 32-week sequence using a bidirectional LSTM enabled each time step's representation to incorporate both preceding and subsequent observations, allowing the model to more effectively capture lead-lag and short-term reversal patterns within the window compared to a unidirectional LSTM [27]. A Bi-LSTM typically enhances short-term discrimination by adding a reverse-direction LSTM with shared design parameters (such as hidden size and depth) and combining the two directions (e.g., via concatenation), thereby increasing the number of parameters and computational cost only linearly without altering the overall architectural scale [28].

Third, we employed a Transformer encoder. Bui, Chien, Kovács, and Bognár (2025) demonstrate that encoder-only Transformers with self-attention effectively capture both long- and short-range dependencies in financial time series while maintaining favorable parameter efficiency, and that time encodings such as Time2Vec can further enhance performance compared to standard positional encoding [29].

Fourth, we incorporated a tree-based ensemble baseline using a Random Forest classifier. Random forests combine numerous decision trees trained on bootstrap samples of the data, incorporating random feature sub-sampling at each split, thereby enhancing robustness to noise and heavy-tailed predictors while effectively capturing nonlinear interactions without relying on strong parametric assumptions [30]. This renders them a natural

benchmark for short-term directional classification utilizing a limited set of weekly summary features.

3. Performance Analysis

We assess directional forecasting performance utilizing multiple complementary metrics. Accuracy represents the proportion of weeks that were correctly classified. Due to potential imbalances in upward and downward movements, we also report balanced accuracy, which averages the hit rates for both classes to prevent either class from disproportionately influencing the evaluation [31]. Furthermore, we present the macro-averaged F1 score (Macro F1), which equally considers the up and down classes and penalizes models that either fail to detect numerous up weeks or produce a substantial number of erroneous up signals [32].

In addition to these standard metrics, we propose a straightforward sensitivity measure to capture the asymmetry of errors in shipping decisions. Let FP and FN denote the numbers of false positives and false negatives on a test set of size N. Given error costs c_{FP} and c_{FN} for false positives and false negatives, respectively, and decision threshold τ on the predicted probability of an up week, we define the average misclassification cost $Cost(\tau)$ [33] as the Eq. (5),

$$Cost(\tau) := \frac{c_{FP}FP(\tau)+ c_{FN}FN(\tau)}{N} \quad . \qquad (5)$$

In practice, a false positive (predicting an increase when rates decline) may result in committing to an unfavorable charter, whereas a false negative (predicting no increase when rates rise) corresponds to a missed profit opportunity. To capture these trade-offs, we consider two stylized scenarios in which false positives are twice as costly as false negatives $(c_{FP} = 2 \text{ and } c_{FN} = 1)$ and vice versa $(c_{FP} = 1 \text{ and } c_{FN} = 2)$. For each classifier, we report the cost at the default threshold $\tau = 0.5$ and examine how $Cost(\tau)$ changes over a grid of thresholds, which allows us to align the evaluation with different economic preferences.

4. Probability Calibration

Although the directional classifiers presented in this study generate class labels indicating weekly market movements, effective decision-making frequently necessitates information beyond discrete predictions. A forecast indicating a 51% probability of an upward movement conveys substantially different information from one indicating 95%, despite both resulting in the same class label. To ensure that the predicted probabilities represent meaningful confidence levels, we evaluate and, when necessary, refine their calibration—that is, the correspondence between the predicted probabilities and the observed frequencies of outcomes.

To obtain calibrated probabilities, we utilize Platt scaling, a post-hoc calibration method that transforms raw predictive scores into well-calibrated posterior probabilities [34, 35]. To quantify calibration performance, we employ three complementary metrics that are commonly used in probabilistic forecasting.

(i) Brier Score [36, 37]

We calculate the mean squared error between the predicted probabilities and the binary outcomes. Lower values signify greater accuracy in probability estimates.

(ii) Expected Calibration Error [38]

Partitions predicted probabilities into equally spaced bins and compares, for each bin, the average predicted probability with the empirical frequency of positive outcomes. Expected Calibration Error (ECE) quantifies systematic miscalibration across the entire probability spectrum.

(iii) Maximum Calibration Error [39]

Identifies the probability bin exhibiting the greatest deviation between predicted confidence and actual accuracy, thereby offering insight into potentially catastrophic overconfidence.

All metrics are computed on the test set both before and after Platt scaling. A model is considered well-calibrated when only minor adjustments are needed, the Brier score is low, and

both the ECE and MCE remain close to zero.

5. Out-of-time Validation Framework

To assess whether the proposed directional forecasting model captures enduring predictive structures rather than sample-specific artifacts, we employ an out-of-time validation framework in which the training set comprises earlier observations, while the validation and test sets consist of subsequent observations [40, 41]. This maintains the inherent chronological sequence of the data. Its advantages become more evident when compared to the more commonly used alternatives [42, 43].

The central concept is to divide the entire sample into multiple non-overlapping test windows, each representing distinct market environments. For each window, the model is trained solely on historical data—without access to any observations from the corresponding test period—and is subsequently evaluated on that unseen future segment. All preprocessing steps, including missing-value imputation, scaling, and normalization, were performed exclusively on the training dataset and subsequently applied to the test set, thereby maintaining the causal structure essential for realistic forecasting.

To ensure comparability across different temporal regimes, the model architecture and hyperparameters are kept constant, and the model is re-estimated from scratch for each out-of-time split [44, 45]. Predictive performance is evaluated using widely accepted classification metrics, including Accuracy, Balanced Accuracy, and Macro F1, enabling the assessment of both the correctness and distributional fairness of directional forecasts.

This framework facilitates a rigorous evaluation of temporal robustness: if the model consistently demonstrates stable performance across various historical market conditions, the observed predictability can be interpreted as fundamentally embedded in the data rather than resulting from a specific sample period. Consequently, out-of-sample validation offers more robust evidence against the likelihood that the reported accuracy is a result of overfitting to recent observations.

6. Data

This study utilized data from the Baltic Handysize Index (BHSI) spanning the period from September 4, 2006, to December 20, 2024 [4]. The dataset consisted of 955 weekly observations. Among these, 128 weeks contain missing weekday values: 100 weeks have one missing day, 8 have two, 6 have three, 5 have four, and 9 have five. We excluded weekend observations and retained only business-day (Monday–Friday) data.

We performed imputation only for weeks in which exactly one weekday value was missing. In all such cases, the missing day was either Monday or Friday, and it was estimated through linear extrapolation based on the adjacent weekdays.

- If Monday's data was unavailable, it could be estimated as Mon = 2 × Tue − Wed.
- If data for Friday was missing, then Friday's value could be calculated as twice that of Thursday minus Wednesday.

Imputation was not conducted for weeks with two or more missing weekday observations; these weeks were excluded from the sample. Furthermore, if an entire week was missing, indicated by the absence of all five weekdays, we also excluded the immediately preceding week, as the data necessary to determine the label for that prior week was unavailable. After applying these criteria, 38 of the 955 weeks were excluded, and the experiments were conducted on the remaining sample. As indicated in Table 1, the ratio of training to test observations was approximately 5 to 4.

Table 1. Summary of Training and Test Set Periods and Weekly Observations

	Training set	Test set
Range (month - day - year)	Sep-04-2006~ Aug-26-2016	Aug-29-2016~ Dec-20-2024
# of weeks	500 weeks	417 weeks

Fig. 2 illustrates six representative test-set sequences to demonstrate the labeling mechanism for weekly forecasting. Each panel displays data for two consecutive weeks—the current week (solid blue) and the subsequent week (dashed orange)—with horizontal dotted lines indicating the mean BHSI for each week. Labels 0 and 1 denote whether the mean of the following week is lower or higher than that of the current week, respectively.

According to our labeling rule, a label of 1 is assigned if the average of the following week is greater than or equal to that of the current week; otherwise, a label of 0 is assigned. The x-axis represents the weekdays (Monday through Friday), while the y-axis indicates the BHSI values.

In the layout, the left column, consisting of three panels, presents examples labeled as 0, indicating an anticipated decline in the BHSI. The right column displays instances labeled as 1, indicating that the index is predicted to increase or remain stable. These visualizations provide intuitive insights into the temporal patterns identified by the model and the underlying rationale of the binary classification framework.

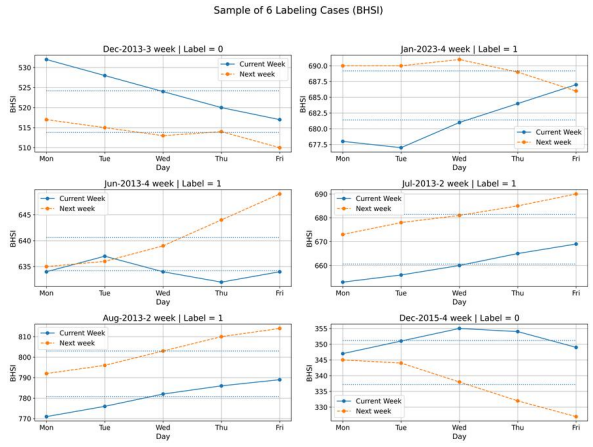

Fig. 2. Six representative weekly BHSI patterns illustrating how binary labels are assigned by comparing whether the following week's average index level rises above or falls below that of the current week.

III. Results

1. DNN Architecture

All experiments were performed using Python 3.10, TensorFlow 2.15, and Scikit-learn 1.4. We trained all neural network models with the Adam optimizer using a learning rate of 0.001 for 2,000 epochs and a batch size of 32. To control random initialization, we set the global seed to 42, ensuring that the training and evaluation results were reproducible. Hyperparameters and training configurations were kept consistent across all models to enable a fair comparison.

Under this standardized experimental setup, we evaluated 48 deep neural network variants introduced in Section II ("Proposed Scheme"). We selected the optimal model based on the performance metrics outlined in Subsection II-1 ("DNN for time series classification"). For convenience, the metric of a DNN configuration J_{NN} is defined as the Eq. (6)

$$J_{NN} := \min_{0 \le n \le 4} \{ j_{NN}(F^i_{DNN} \mid i \in [a + n\Delta, b + n\Delta]) \}, \qquad (6)$$

where $a = 1501$, $b = 1600$ and $\Delta = 100$. In other words, each DNN architecture is evaluated using the sequence of checkpoint models generated during training epochs 1501 to 2000.

We examined three categories of deep neural networks (DNNs): MLP, FCN, and ReN. Within each category, the 16 candidate architectures are indexed by $MLP_n(x)$, $FCN_n(x)$ and $ReN_n(x)$, representing the output of the n-th architecture. (See Appendix B for details). For each DNN architecture, we trained the model and preserved 500 snapshot predictors from the final training epochs. From these 500 predictive models, we computed a stability-aware performance metric and ultimately selected the architecture that attained the highest value of this metric.

Table 2. J_{NN} Performance of 48 DNN Architectures Comprising 16 Variants Each of MLP, FCN, and ReN

n	$MLP_n(x)$	$FCN_n(x)$	$ReN_n(x)$
1	0.674	0.868	0.911
2	0.703	0.881	0.889
3	0.748	0.900	0.895
4	0.783	0.908	0.913
5	0.681	0.856	0.882
6	0.634	0.907	0.896
7	0.651	0.899	0.873
8	0.637	0.887	0.875
9	0.717	0.857	0.899
10	0.855	0.888	0.894
11	0.857	0.896	0.894
12	0.859	0.896	0.898
13	0.780	0.779	0.894
14	0.760	0.801	0.897
15	0.749	0.701	0.895
16	0.741	0.810	0.895

As illustrated in Table 2, the ReN variant achieved the highest value of the performance metric J_{NN}. We elucidate its operational mechanism. While Section II presented the general family of residual classifiers, the specific ReN configuration detailed in Table 2 represents the variant that maximizes the metric J_{NN}. Based on the residual block formulation presented in Eqs (C.1)-(C.3) of Appendix C, the pseudocode in Table 3 outlines the core inference procedure of the selected ReN model.

Table 3. Residual Network-Based Weekly Directional Forecasting Procedure

Algorithm		
Input	: Weekly observation $\boldsymbol{x}_t$ $\boldsymbol{x}_t := (x_{t,Mon}, x_{t,Tue}, x_{t,Wed}, x_{t,Thu}, x_{t,Fri})$	
Output	: $\hat{y}_t \in \{0, 1\}$ indicating the direction of next week's average level	
1.	$h \leftarrow x$	▷ Initialize representation
2.	For $k = 1, \cdots, K$ do	▷ Residual blocks
3.	$z \leftarrow Conv(h)$	▷ Extract short -term patterns
4.	$z \leftarrow BN(z)$	
5.	$z \leftarrow ReLU(z)$	
6.	$h \leftarrow Conv(h)$	
7.	$h \leftarrow BN(h)$	
8	$h \leftarrow h + z$	
7	end for	▷ Residual skip connection
8	$y \leftarrow GAP(h)$	▷ Compress temporal structure
9	$\hat{y} \leftarrow \underset{\max}{Soft}(y)$	▷ Obtain posterior probability
return $\hat{y}$		

2. Predictive models

Using the five algorithms outlined above, we trained predictive models and evaluated them on the held-out weekly test set covering August 29, 2016, to December 20, 2024 (417 observations). We evaluated performance using accuracy, balanced accuracy, and the Macro F1 score. To assess temporal stability, we also calculated a 50-week rolling accuracy series.

Table 4 provides a comparative analysis of accuracy, balanced accuracy, and Macro F1 scores across the five models. The ReN model demonstrates superior overall performance, attaining an accuracy of 0.92, balanced accuracy of 0.92, and a Macro F1 score of 0.92, thereby indicating consistently robust predictive capability. The SVM demonstrates consistently high and stable scores (0.87/0.87/0.87), whereas the Bi-LSTM also performs well, albeit at a slightly lower level (0.85 accuracy, 0.85 balanced accuracy, and 0.86 Macro F1). The Random Forest (RF) model produces moderate results, with all three metrics registering a value of 0.74. In contrast, the Transformer performs near chance level with respect to accuracy and balanced accuracy (0.52 and 0.50, respectively), although its Macro F1 score of 0.69 indicates that its class-level discrimination is somewhat superior to what overall accuracy alone would suggest.

Table 4. Accuracy, Balanced Accuracy, and Macro F1 Scores of the Five Competing Models

Model	Accuracy	Balanced Accuracy	Macro F1
ReN	0.92	0.92	0.92
Bi-LSTM	0.85	0.85	0.86
Trans-Former	0.52	0.5	0.69
SVM	0.87	0.87	0.87
RF	0.74	0.74	0.74

The ROC analysis offers further insight into each model's capacity to distinguish between "up" and "down" weeks. The ReN model once again demonstrates the highest performance, attaining an

AUC of 0.97, thereby confirming the robustness of its predictive ranking capability. The SVM closely follows with an AUC of 0.95, thereby underscoring its competitive classification performance. The Bi-LSTM also demonstrates strong performance, achieving an AUC of 0.91, which aligns with its robust statistical metrics presented in Table 4. The random forest model achieves an AUC of 0.82, demonstrating moderate yet meaningful discriminative capability. In contrast, the Transformer model demonstrates limited ranking capability, with an AUC of only 0.42, which corresponds to its low accuracy and balanced accuracy scores, indicating that its predictions approximate random ordering in this context.

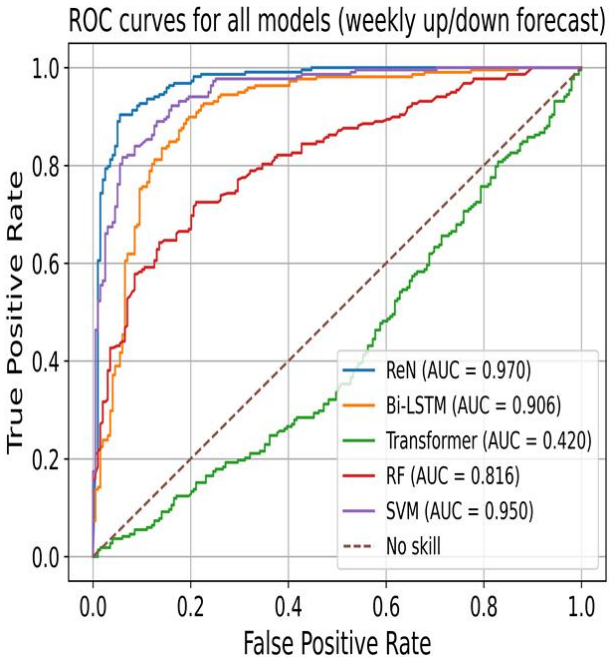

Fig. 3. ROC curves comparing the directional forecasting performance of all models, showing that ReN achieves the highest discriminative ability with the largest AUC value among the classifiers.

Table 5 presents the cost-sensitive evaluation results for all classifiers under two alternative misclassification schemes, $(c_{FP} = 2 \text{ and } c_{FN} = 1)$ and $(c_{FP} = 1 \text{ and } c_{FN} = 2)$, both calculated using the default decision threshold. In both contexts, the ReN model attains the lowest total misclassification cost, demonstrating that its robust statistical performance effectively translates into economic value. The SVM baseline consistently ranks second, exhibiting relatively low costs in both scenarios.

In contrast, the Transformer model entails significantly higher costs in both contexts. Due to the generation of an excessive number of false positives (TP = 218, FP = 199, TN = 0, FN = 0), its total cost is several times greater than that of ReN, especially in scenarios where false positives are more heavily penalized. Consequently, although the Transformer achieves a notable Macro F1 score, it fails to provide economically meaningful forecasts when asymmetric error penalties are considered.

Table 5. Confusion matrix statistics (TP, FP, TN, FN) and scenario-based misclassification costs for all models at the default decision threshold.

Model	TP	FP	TN	FN	$Cost(\tau = 0.5)$	
					$c_{FP} = 2,$ $c_{FN} = 1$	$c_{FP} = 1,$ $c_{FN} = 2,$
ReN	197	12	187	21	0.13	0.11
Bi-LSTM	192	37	162	26	0.21	0.24
Trans-former	218	199	0	0	0.48	0.95
SVM	186	22	177	32	0.21	0.18
RF	160	52	147	58	0.4	0.34

3. Calibration performance

Table 6 presents post-hoc calibration results on the test set using the Brier score, Expected Calibration Error (ECE), and Maximum Calibration Error (MCE). Overall, ReN provides the most reliable probabilistic forecasts. Even before calibration, ReN achieves the lowest Brier score (0.064) with a low ECE (0.033). After applying Platt scaling, ReN changes only marginally (Brier: 0.064 → 0.063; ECE: 0.033 → 0.033; MCE: 0.252 → 0.272), indicating that its raw outputs already behave as stable probability estimates.

In contrast, the other deep learning models exhibit clearer calibration limitations. The Bi-LSTM benefits from calibration in terms of both Brier score and ECE (0.181 → 0.140; 0.237 → 0.143), although its MCE slightly increases (0.297 → 0.313), suggesting that some probability regions remain

misaligned with empirical frequencies. The Transformer does not benefit from Platt scaling in this setting: its Brier score increases (0.178 → 0.218) and both ECE and MCE deteriorate substantially (0.065 → 0.202; 0.153 → 0.455), implying that post-hoc scaling can be unstable when the underlying scores are poorly ordered for calibration.

Baseline methods show a different pattern. Random Forest is severely miscalibrated before calibration (MCE = 0.725), but Platt scaling sharply improves its calibration errors and yields the lowest post-calibration ECE in Table 6 (ECE: 0.055 → 0.023; MCE: 0.725 → 0.059); however, its Brier score remains the highest and improves only marginally (0.253 → 0.250). The SVM baseline remains relatively stable (Brier: 0.089 → 0.088), although its ECE slightly increases after calibration (0.037 → 0.046).

Collectively, these results indicate that ReN offers the strongest overall combination of probabilistic accuracy (lowest Brier score) and consistently low calibration error with minimal post-hoc adjustment, which is desirable for decision-making contexts where the magnitude of confidence carries economic significance.

Table 6. Test-Set Probability Calibration Performance of Competing Models

Model	Calibration	Brier	ECE	MCE
ReN	before	0.064	0.033	0.252
	after	0.063	0.033	0.272
Bi-LSTM	before	0.181	0.237	0.297
	after	0.140	0.143	0.313
Trans former	before	0.178	0.065	0.153
	after	0.218	0.202	0.455
SVM	before	0.089	0.037	0.259
	after	0.088	0.046	0.264
RF	before	0.253	0.055	0.725
	after	0.250	0.023	0.059

4. Out-of-time validation

To implement the aforementioned out-of-time validation framework, the entire sample period was divided into multiple non-overlapping test windows. The weekly dataset, covering the period from September 2006 to December 2024, encompassed various distinct market environments, including pre-crisis conditions, post-crisis recovery, and the recent COVID-19 and post-pandemic periods. Based on this temporal framework, we constructed a sequence of evaluation windows in which each test window was preceded by a corresponding training period. The models were trained exclusively on observations preceding each test window and were subsequently evaluated on the unseen future segment, thereby enabling a realistic assessment of directional forecasting performance amid evolving market regimes.

(i) Test A (2012–2015): Models were trained using observations from 2006 to 2011 and evaluated on data from 2012 to 2015. This period reflected the early post-crisis environment marked by persistent volatility and structural realignment, presenting a challenging benchmark for directional forecasting.

(ii) Test B (2016–2019): Models were trained using data from 2006 to 2015 and evaluated on data from 2016 to 2019. This period corresponded to a relatively stable expansion phase characterized by market movements that were less influenced by short-term shocks, allowing for an examination of whether predictive signals persisted under more tranquil conditions.

(iii) Test C (2020–2024): Models were trained using data from 2006 to 2019 and evaluated on observations from 2020 to 2024. The most recent period encompassed the COVID-19 disruption and subsequent regime transitions, constituting the most challenging out-of-sample environment due to abrupt structural breaks and increased uncertainty.

The out-of-sample evaluation results, summarized in Tables 7, 8, and 9, offered compelling empirical evidence supporting the superiority of the proposed ReN architecture. Each table corresponded to a distinct structural market environment: Table 7 reported performance during the post-crisis adjustment period (2012–2015), Table 8 presented results for the pre-pandemic expansion phase (2016–2019), and Table 9 documented performance amid

the COVID-19 shock and the ensuing regime transition (2020–2024).

Across all three market regimes, ReN consistently attained the highest Accuracy, Balanced Accuracy, and Macro F1 scores compared to all competing models. Notably, although traditional baselines such as SVM and Random Forest occasionally achieved acceptable performance, neither model outperformed ReN in any metric or evaluation window. Deep sequence-based architectures performed even more poorly: the Bi-LSTM model completely failed during the earliest period, resulting in a Macro F1 score of 0 as reported in Table 7, while the Transformer model exhibited unstable behavior, showing near-random accuracy despite occasional increases in Macro F1 scores.

In contrast, ReN consistently demonstrated outstanding predictive accuracy across all scenarios, achieving scores of 0.88, 0.93, and 0.94 in the three respective test periods. The model's performance remained not only stable but also improved as structural shocks intensified, as clearly shown in Table 9, where ReN continued to outperform competing models despite the unprecedented volatility caused by the COVID-19 disruption. This temporal robustness indicated that ReN captured enduring directional signals inherent in freight market dynamics rather than relying on incidental patterns specific to any particular sample period.

Collectively, the results presented in Tables 7, 8, and 9 corroborated the primary contribution of this study: ReN was the sole architecture capable of generalizing directional predictability across heterogeneous market regimes, thereby establishing a new performance benchmark for short-term forecasting in freight-rate prediction.

Table 7. Out-of-time Performance During the Post-Crisis Adjustment Period (2012–2015)

Test A			
Model	Accuracy	Balanced Accuracy	Macro F1
ReN	0.88	0.88	0.88
Bi-LSTM	0.51	0.50	0.00
Trans-former	0.49	0.50	0.66
SVM	0.75	0.75	0.74
RF	0.54	0.55	0.53

Table 8. Out-of-time Performance During the Pre-Pandemic Expansion Phase (2016–2019)

Test B			
Model	Accuracy	Balanced Accuracy	Macro F1
ReN	0.93	0.92	0.93
Bi-LSTM	0.56	0.50	0.72
Trans-former	0.56	0.50	0.72
SVM	0.89	0.89	0.89
RF	0.79	0.80	0.79

Table 9. Out-of-time Performance During the COVID-19 Shock and Post-Pandemic Regime (2020–2024)

Test C			
Model	Accuracy	Balanced Accuracy	Macro F1
ReN	0.94	0.94	0.94
Bi-LSTM	0.81	0.81	0.79
Trans former	0.50	0.50	0.34
SVM	0.86	0.86	0.86
RF	0.75	0.75	0.75

5. Interpretation & Application

To elucidate how the proposed ReN architecture attains its superior directional accuracy, we analyze the extent to which each model responds to cyclic reorderings of intra-week observations. Perturbation analyses—commonly employed in robustness and invariance studies of time-series data and neural representations—examine whether a model depends on meaningful temporal structures or simply memorizes superficial input patterns [46]. Formally, let $x \in R^T$ denote the weekly input and $rot(x)$ be cyclic phase rotation of x. For a classifier f, the absolute change in

posterior probabilities is measured as follows.

$$\begin{cases} S_k(x;f) := |f(x) - f(rot(x))| \\ S(f) := \frac{1}{N}\sum_{i=1}^{N}\frac{1}{|K|}\sum_{k\in K} S_k(x^{(i)};f), \end{cases} \quad (5)$$

where $S(f)$ summarizes the model-level responsiveness to economically neutral phase perturbations. Lower values indicate strong invariance to phase rotations, whereas higher—but moderately stable—values correspond to a controlled capacity to detect structural reconfigurations within the week.

Although invariance is typically advantageous in conventional recognition tasks, our findings, in conjunction with previous studies, indicate that excessive invariance may be detrimental to short-term directional forecasting. Our dataset clearly indicates that intra-week permutations do not preserve labels. Weeks characterized by a pronounced Monday-to-Friday increase and those marked by a steep Thursday-to-Friday decline may display comparable weekly levels and volatility; their primary distinction lies in the intra-week sequencing of shocks. In principle, a permutation of daily observations can transform one pattern into another. However, their labels differ systematically: weeks exhibiting a pronounced Monday-to-Friday increase are almost invariably labeled "up," whereas weeks characterized by a sharp decline from Thursday to Friday are consistently labeled "down." Therefore, in short-term financial data, intra-week permutations do not preserve the directional label, as the timing of shocks within the week is economically significant and directly influences the sign of the subsequent return.

In this context, it is pertinent to analyze how each model responds to cyclic reordering of intra-week observations. Table 10 and Fig. 4 present a summary of the resulting phase sensitivity profiles. The Transformer, whose predictions remain largely unaffected by phase rotations, demonstrates the greatest invariance yet yields the poorest performance in directional forecasting. By eliminating all intra-week rearrangements, it unintentionally discards the very temporal asymmetries and ordering effects that convey directional information. In contrast, ReN exhibits a moderate degree of non-invariance: it responds to phase shifts that modify informative structure while remaining stable when such perturbations do not contain predictive information. This controlled responsiveness allows ReN to leverage meaningful weekly configurations and accounts for its superior directional accuracy compared to all other tested architectures. In summary, for predicting weekly returns, algorithms that exhibit excessive invariance to intra-week phase changes are at a disadvantage, whereas models that maintain sensitivity to economically relevant sequential patterns—such as ReN—are better equipped to forecast the direction of movements in the following week.

Table 10. Mean phase sensitivity of competing models under cyclic intra-week perturbations

Model	Mean Phase Sensitivity
ReN	0.42
Bi-LSTM	0.09
Trans-former	0.00
SVM	0.33
RF	0.19

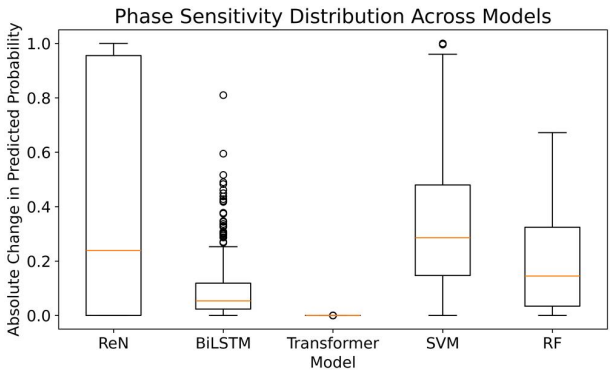

Fig. 4. Phase sensitivity distributions for all models, illustrating substantial differences in how predicted probabilities change under cyclic permutations of weekly inputs.

These properties are important because, in operational contexts, directional forecasts can directly impact maritime decision-making. A shipper uncertain about next week's freight volumes may adjust the timing of charter contracts by postponing fixture decisions if the model forecasts an increase. Brokers can incorporate the model's calibrated probabilities into rate-negotiation dashboards, prioritizing counterparties or vessel classes when confidence levels surpass predetermined thresholds. Policymakers tasked with monitoring logistics bottlenecks could integrate weekly directional signals into early warning systems to facilitate preemptive interventions when persistent upward trends indicate capacity constraints or market overheating. Unlike point forecasts, which necessitate close alignment with realized index levels, directional forecasts are immediately actionable and correspond to the discrete, time-sensitive characteristics of freight contracting.

IV. Conclusion

This study presents the Residual Network (ReN) architecture to mitigate instability and poor generalization frequently encountered in time-series forecasting models, especially when predicting short-term, noisy, and non-stationary sequences. Throughout extensive comparative experiments, ReN consistently demonstrated superior directional forecasting performance, outperforming benchmark models such as the Transformer, Bi-LSTM, and traditional machine learning classifiers. These findings suggest that consistent weekly predictability can be derived from sparse time-series data when the architecture incorporates an appropriate structural bias.

A primary factor contributing to ReN's superiority is its calibrated phase-sensitivity profile. In contrast to highly invariant architectures—most notably the Transformer—which demonstrated negligible changes in predictive outputs following cyclic reordering of intra-week observations, ReN maintained meaningful responsiveness to phase-dependent perturbations. This regulated sensitivity enables the model to preserve directional cues inherent in temporal ordering while mitigating the excessive volatility characteristic of overreactive models. Consequently, ReN is distinctively equipped to leverage structural asymmetries in weekly patterns, facilitating precise directional inference over short-term forecasting.

Nonetheless, several limitations persist. First, the enhanced sensitivity that underpins ReN's superior accuracy results in increased computational costs during training and hyperparameter tuning. Future research ought to explore adaptive regularization strategies that effectively balance phase responsiveness with training efficiency. Second, while ReN exhibits robustness to naturally occurring cyclic shifts, its vulnerability to adversarial or deliberately engineered phase manipulations has yet to be evaluated. Incorporating adversarial perturbation frameworks could provide deeper insights into the temporal robustness of residual architectures. Third, the present study concentrates on weekly economic data; thus, validating ReN across a wider range of domains—such as clinical monitoring, industrial sensor networks, and high-frequency financial series—would improve understanding of its generalizability.

Taken together, this study emphasizes a crucial methodological insight: partial, domain-aligned phase sensitivity—not complete invariance—is essential for accurate short-term temporal prediction. By demonstrating that residual architecture can preserve and utilize phase-dependent structures without overfitting, ReN establishes a foundation for the development of next-generation phase-aware deep learning models designed for environments in which subtle temporal configurations influence decision outcomes.

Appendices

For the sake of completeness, several mathematical details are provided in the appendices. Appendix A compiles the operator definitions and notations employed throughout the deep neural networks. Appendix B provides a summary of the forward mappings of the MLP, FCN, and ReN architectures used for weekly classification. Appendix C formally defines the stability-aware performance metric and outlines its application in selecting the optimal architecture.

A. Operator definitions and notation

Below the table 11. we summarize the basic operators used in the MLP, FCN, and ReN architectures. These definitions allow the main text to focus on high-level modeling ideas without repeatedly introducing notation.

Table 11. Layers in Deep Neural Networks

Layer	Notation	Definition
Dense Operator	Ds	When $x:=(x_1,x_2,\cdots,x_N)$, $Ds(L,f(\cdot))(x):=f(Wx^T+b^T)$. W is a $L\times N$ matrix, b is a $(b_1,b_2,\cdots,b_N)$ N-vector, f is an activation function.
Dropout	$Drop\ Out$	$Drop\ Out(x;W,r(p),b):=W(r(p)*x)^T+b^T$ $r(\cdot)$is the Bernoulli random vector, p is a probability, * is a element-wise product
Linear Unit	LU	$LU(x)=(x_1,x_2,\cdots,x_N)$
Rectified Linear Unit	$ReLU$	$ReLU(x):=(\max(0,x_1),\cdots,\max(0,x_N)$ where $\max(a,b):=a$, if $a>b$.
Convolutional Operator	$Conv$	$Conv(x)_i:=\sum_n^N X_{i+n}K_n$ X is an one-dimensional image and K is the Toeplitz operator.
Block Operator	$Block$	$Block_g\circ F(x)=F(x)+g(x)$.
Batch Normalization	BN	μ_β and σ_β are an average of all elements of vector and its standard deviation, respectively. $BN(x):=\gamma\left(\frac{x-\mu_\beta}{\sqrt{(\sigma_\beta)^2+\epsilon}}\right)+\beta$.
Global Average Pooling	GAP	$GAP(X):=\frac{1}{T}\sum_i^T X_i$.
Softmax	$s_{oft}m_{ax}$	$S_{oft}m_{ax}(\boldsymbol{x})=\frac{e^{x_j}}{\sum_i^N e^{x_i}}$.

B. Network architectures for weekly classification

This appendix provides a summary of the precise forward mappings of the three deep neural network (DNN) families employed in the experiments: MLP, FCN, and ReN. Throughout, the input corresponding to calendar week t is a five-dimensional vector $x\in R^5$ composed of Monday through Friday BHSI values, as detailed in Section II. For each architectural family, we first describe the overall structure and then formalize the corresponding forward mapping in the equations. (B.1)-(B.7).

B.1 Multilayer Perceptron (MLP)

- Let the number of substructures be denoted by i, where $i=1,2,3\text{ or }4$. When $j=0,1,2\text{ or }3$, each DNN architecture within the MLP set $MLP_{4j+i}(x)$ is defined as the Eq. (B.1)

$$MLP_{4j+i}(x) := s_{oft}m_{ax}\circ \overbrace{F_j^{(MLP)}\circ\cdots\circ F_j^{(MLP)}}^{i}(x) \tag{B.1}$$

where the substructure $F_j^{(MLP)}$ is as follows:

$$F_j^{(MLP)}(x) := \begin{cases} Ds(L=500,ReLU)(x), & \text{if } j=0,\\ Ds(L=500,ReLU)\circ DropOut(x), & \text{if } j=1,\\ Ds(L=500,LU)(x), & \text{if } j=2,\\ Ds(L=500,LU)\circ DropOut(x), & \text{if } j=3. \end{cases} \tag{B.2}$$

B.2 Fully Convolutional Network (FCN)

When $i=1,2,3\text{or}4$ and $j=0,1,2\text{ or }3$, each DNN architecture within the $FCN_{4j+i}(x)$ is described as the Eq. (B.3),

$$FCN_{4j+i}(x) := s_{oft}m_{ax}\circ GAP\circ \overbrace{F_j^{(FCN)}\circ\cdots\circ F_j^{(FCN)}}^{i}(x), \tag{B.3}$$

where the substructure $F_j^{(FCN)}$ is as the Eq. (B.4),

$$F_j^{(FCN)}(x)=\begin{cases} Conv\circ ReLU(x), & j=0,\\ Conv\circ BN\circ ReLU(x), & j=1,\\ Conv\circ LU(x), & j=2,\\ Conv\circ BN\circ LU(x), & j=3. \end{cases} \tag{B.4}$$

B.3 Residual Network (ReN)

When $i=1\text{ or }2$, $j=0,1,2\text{or }3$ and $k=0\text{ or }1$ each DNN architecture in ReN set $ReN_{8k+2j+i}(x)$ is as the Eq. (B.5),

$$ReN_{8k+2j+i}(x) := \begin{cases} s_{oft}m_{ax} \circ GAP \circ ReLU \circ Block_{g_j} \circ \overbrace{F_k^{(ReN)} \circ \cdots \circ F_k^{(ReN)}}^{2+i}(x), & \text{if } 8k+2j+i \le 8, \\ s_{oft}m_{ax} \circ GAP \circ LU \circ Block_{g_j} \circ \overbrace{F_k^{(ReN)} \circ \cdots \circ F_k^{(ReN)}}^{i+2}(x), & \text{if } 8k+2j+i > 8, \end{cases} \quad \text{(B.5)}$$

where the substructure $F_j^{(ReN)}$ and activation function g_k is as the Eqs. (B.6) and (B.7),

$$F_k^{(ReN)}(x) := \begin{cases} Conv \circ ReLU(x), & \text{if } k=0, \\ Conv \circ LU(x), & \text{if } k=1, \end{cases} \quad \text{(B.6)}$$

$$g_j := \begin{cases} Conv, & \text{if } j=0, \\ Conv \circ ReLU, & \text{if } j=1, \\ ReLU, & \text{if } j=2, \\ LU, & \text{if } j=3. \end{cases} \quad \text{(B.7)}$$

C. Optimal Architecture

Applying Eqs.~(B.5)--(B.7) with $i=2$, $j=1$ and $k=0$ (hence $8k+2j+i=4$), the resulting architecture, denoted by ReN_4 is given by the Eq. (C.1),

$$ReN_4(x) := s_{oft}m_{ax} \circ GAP \circ ReLU \circ Block_{g_{j=1}} \circ F_{k=0}^{(ReN)} \circ F_{k=0}^{(ReN)} \circ F_{k=0}^{(ReN)} \circ F_{k=0}^{(ReN)}(x), \quad \text{(C.1)}$$

The constituent mappings are defined as the Eqs. (C.2) and (C.3),

$$F_{k=0}^{(ReN)}(x) := Conv \circ ReLU(x) \quad \text{(C.2)}$$

$$g_{j=1}(x) := Conv \circ ReLU(x). \quad \text{(C.3)}$$

where $Conv$ denotes a one-dimensional convolution operator and GAP denotes global average pooling shown in the Table 11.

ACKNOWLEDGEMENT

This work was supported by a grant from the National Research Foundation of Korea (NRF), funded by the Korean government (MSIT) (NRF-2022R1F1A1062959). This study was revised and supplemented based on the report titled “The Operation of Business Forecasting for the Maritime Industry” (January-December 2022), conducted by the Korea Maritime Institute (KMI).

REFERENCES

[1] Stopford, M., "Maritime Economics, 3rd ed.," Routledge, p. xxii, 2009.

[2] UN Trade and Development (UNCTAD), "Review of Maritime Transport 2024," UN Trade and Development, https://unctad.org/publication/review-maritime-transport-2024

[3] Banchero Costa Research, "Dry Bulk Market Outlook," Hellenic Shipping News, https://www.hellenicshippingnews.com/wp-content/uploads/2023/11/2023-11-Dry-Bulk-Outlook.pdf

[4] Clarksons Research, "Shipping Intelligence Network (SIN)," Clarksons Research, https://sin.clarksons.net/

[5] Riviera Maritime Media, "Shipping investments in 2024: Handysize and Aframax vessels lead ROI rankings," Riviera Maritime Media, https://www.rivieramm.com/news-content-hub/news-content-hub/shipping-investments-in-2024-handysize-and-aframax-vessels-lead-roi-rankings-81866

[6] D. Lei, H. Hu, and D. Zhang, "An Empirical Analysis of Freight Rate and Vessel Price Volatility Transmission in Global Dry Bulk Shipping Market," Journal of Traffic and Transportation Engineering (English Edition), Vol. 2, No. 5, pp. 353-361, May 2015. DOI: 10.1016/j.jtte.2015.08.007

[7] S.-H. Bae and K.-S. Park, "Analysis of Causality of Baltic Drybulk Index (BDI) and Maritime Trade Volume," Korea Trade Review, Vol. 44, No. 2, pp. 127-141, Apr. 2019. DOI: 10.22659/KTRA.2019.44.2.127

[8] Z. Yang and E. E. Mehmed, "Artificial Neural Networks in Freight Rate Forecasting," Maritime Economics & Logistics, Vol. 21, No. 3, pp. 390-414, Sept. 2019. DOI: 10.1057/41278-019-00121-x

[9] Q. Zeng, C. Qu, A. K. Y. Ng, and X. Zhao, "A New Approach for Baltic Dry Index Forecasting Based on Empirical Mode Decomposition and Neural Networks," Maritime Economics & Logistics, Vol. 18, No. 2, pp. 192-210, Feb. 2016. DOI: 10.1057/mel.2015.2

[10] D. Kim, H. Kim, S. Sim, Y. Choi, H. Bae, and H. Yun, "Prediction of Dry Bulk Freight Index Using Deep Learning," Journal of the Korean Institute of Industrial Engineers, Vol. 45, No. 2, pp. 111-116, Apr. 2019. DOI: 10.7232/JKIIE.2019.45.2.111

[11] Y. Feng, "Container freight rate level forecasting with machine learning methods," International Journal of Business Performance and Supply Chain Modelling, Vol. 13, No. 3, pp. 245-263, Sept. 2022. DOI: 10.1504/IJBPSCM.2022.125689

[12] E. Hirata and T. Matsuda, "Forecasting Shanghai Container Freight Index: A deep-learning-based model experiment," Journal of Marine Science and Engineering, Vol. 10, No. 5, Article 593, Apr. 2022. DOI: 10.3390/jmse10050593

[13] C. Li, X. Wang, Y. Hu, Y. Yan, H. Jin, et al., "Forecasting shipping index using CEEMD-PSO-BiLSTM model," PLOS ONE, Vol. 18, No. 2, Article e0280504, Feb. 2023. DOI: 10.1371/journal.pone.0280504

[14] Z. Han, X. Zhu, and Z. Su, "Forecasting maritime and financial market trends: Leveraging CNN-LSTM models for sustainable shipping and China's financial market integration," Sustainability, Vol. 16, No. 22, Article 9853, Nov. 2024. DOI: 10.3390/su1622 9853

[15] P. Schmidt, M. Katzfuss, and T. Gneiting, "Interpretation of point forecasts with unknown directive," Journal of Applied Econometrics, Vol. 36, No. 6, pp. 728-743, July 2021. DOI: 10.1002/jae.2833

[16] Z. Wang, W. Yan, and T. Oates, "Time Series Classification from Scratch with Deep Neural Networks: A Strong Baseline," 2017 International Joint Conference on Neural Networks (IJCNN), pp. 1578-1585, Alaska, USA, May 2017. DOI: 10.1109/IJCNN.2017. 7966039

[17] S.-H. Lee, "Deep Neural Network Architectures for Momentary Forecasting in Dry Bulk Markets: Robustness to the Impact of COVID-19," IEEE Access, Vol. 11, pp. 1541 9-15448, Feb. 2023. DOI: 10.1109/ACCESS.2023.244680

[18] J. Yin, W. Li, X. Wang, X. Ye, and Y. Ouyang, "4G/5G Cell-level multi-indicator forecasting based on dense-MLP," ITU Journal on Future and Evolving Technologies, Vol. 3, No. 2, pp. 108-116, June 2022. DOI: 10.52953/KBAL8913

[19] C. Ji, Y. Hu, S. Liu, L. Pan, B. Li, and X. Zheng, "Fully convolutional networks with shapelet features for time series classification," Information Sciences, Vol. 612, pp. 835-847, Sept. 2022. DOI: 10.1016/j.ins.2022.09.009

[20] M. Perslev, M. Jensen, S. Darkner, P. J. Jennum, and C. Igel, "U-time: A fully convolutional network for time series segmentation applied to sleep staging," Advances in Neural Information Processing Systems 32 (NeurIPS 2019), Vancouver, Canada, Dec. 2019. DOI: 10.48550/arXiv.1910. 11162

[21] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, "Deep learning for time series classification: A review," Data Mining and Knowledge Discovery, Vol. 33, No. 4, pp. 917-963, July 2019. DOI: 10.1007/s10618-019-00619-1

[22] N. I. Sapankevych and R. Sankar, "Time Series Prediction Using Support Vector Machines: A Survey," IEEE Computational Intelligence Magazine, Vol. 4, No. 2, pp. 24-38, May 2009. DOI: 10.1109/MCI.2009.932254

[23] F. Guan, Z. Peng, K. Wang, X. Song, and J. Gao, "Multi-step Hybrid Prediction Model of Baltic Supermax Index Based on Support Vector Machine," Neural Network World, Vol. 26, No. 3, pp. 219-232, June 2016. DOI: 10.14311/NNW.2016.26.012

[24] A. Dingli and K. S. Fournier, "Financial Time Series Forecasting – A Machine Learning Approach," Machine Learning and Applications: An International Journal, Vol. 4, No. 1/2/3, pp. 11-26, Sept. 2017. DOI: 10.5121/mlaij.2017. 4302

[25] E. A. Gerlein, M. McGinnity, A. Belatreche, and S. Coleman, "Evaluating Machine Learning Classification for Financial Trading: An Empirical Approach," Expert Systems with Applications, Vol. 54, pp. 193-207, July 2016. DOI: 10.1016/j. eswa.2016.01.018

[26] M. K. Das, C. C. Chinnappan, and E. Elakiya, "A temporal attention-based SARIMA-BiLSTM residual learning model tuned by grey wolf optimizer for parallel urban traffic forecasting," IEEE Access, Vol. 13, pp. 136073-136086, July 2025. DOI: 10.1109/ACCESS.2025.3590104

[27] R. L. Abduljabbar, H. Dia, and P.-W. Tsai, "Unidirectional and bidirectional LSTM models for short-term traffic prediction," Journal of Advanced Transportation, Vol. 2021, Article 5589075, Mar. 2021. DOI: 10.1155/2021/5589075

[28] S. Siami-Namini, N. Tavakoli, and A. S. Namin, "The performance of LSTM and BiLSTM in forecasting time series," Proceedings of the 2019 IEEE International Conference on Big Data (Big Data), pp. 3285-3292, Los Angeles, USA, Dec. 2019. DOI: 10.1109/BigData47090 .2019.9005997

[29] Nguyen K. H. Bui, N. D. Chien, P. Kovács, and G. Bognár, "Transformer Encoder and Multi-features Time2Vec for Financial Prediction," Proceedings of the 33rd European Signal Processing Conference (EUSIPCO 2025), pp. 1682-1686, Palermo, Italy, Sept. 2025. DOI: 10.48550/arXiv.2504.13801

[30] L. Breiman, "Random Forests," Machine Learning, Vol. 45, No. 1, pp. 5-32, Oct. 2001. DOI: 10.1023/A:101093340432 4

[31] K. H. Brodersen, C. S. Ong, K. E. Stephan, and J. M. Buhmann, "The balanced accuracy and its posterior distribution," Proceedings of the 20th International Conference on Pattern Recognition (ICPR 2010), pp. 3121-3124, Istanbul, Turkey, Aug. 2010. DOI: 10.1109/ICPR.2010.764

[32] J. Opitz and S. Burst, "Macro F1 and macro F1," arXiv preprint arXiv:1911.03347, 2019. URL: https://arxiv.org/abs/ 1911.03347

[33] C. Zhang, K. C. Tan, H. Li, and G. S. Hong, "A cost-sensitive deep belief network for imbalanced classification," IEEE Transactions on Neural Networks and Learning Systems, Vol. 30, No. 1, pp. 109-122, Jan. 2019. DOI: 10.1109/TNNLS.2018. 2832648

[34] J. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in Advances in Large Margin Classifiers, MIT Press, pp. 61-74, 1999.

[35] X. Ma and M. B. Blaschko, "Meta-Cal: Well-controlled post-hoc calibration by ranking," Proceedings of the 38th International Conference on Machine Learning (ICML 2021), pp. 1-11, Vienna (Virtual), Austria, July 2021. DOI: 10.5555/3495724.3495746

[36] G. W. Brier, "Verification of forecasts expressed in terms of probability," Monthly Weather Review, Vol. 78, No. 1, pp. 1-3,

Jan. 1950.

[37] L. Hoessly, "On Misconceptions About the Brier Score in Binary Prediction Models," arXiv preprint arXiv:2504.04906, 2025. URL: https://arxiv.org/abs/2504.04906

[38] M. P. Naeini, G. F. Cooper, and M. Hauskrecht, "Obtaining well calibrated probabilities using Bayesian binning," Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI-15), pp. 2901-2907, Austin, Texas, USA, Jan. 2015. DOI: 10.1609/aaai.v29i1.9602

[39] J. H. Shen, E. Vitercik, and A. Wikum, "Algorithms with calibrated machine learning predictions," Proceedings of the 42nd International Conference on Machine Learning (ICML 2025), Vienna, Austria, July 2025. OpenReview: https://openreview.net/forum?id=Obet2x6GNl

[40] J. Mushava and M. Murray, "Comprehensive credit scoring datasets for robust testing: out-of-sample, out-of-time, and out-of-universe evaluation," Data in Brief, Vol. 54, Article 110262, Mar. 2024. DOI: 10.1016/j.dib.2024.110262

[41] A. Inoue and L. Kilian, "In-sample or out-of-sample tests of predictability: Which one should we use?," Econometric Reviews, Vol. 23, No. 4, pp. 371-402, 2004. DOI: 10.1081/ETC-200040785

[42] M. T. Hibbeln, C. Jentsch, R. M. Kopp, and N. Urban, "Model validation for pooled cross-sectional data: Out-of-sample vs. out-of-time," SSRN working paper, April 2025. DOI: 10.2139/ssrn.5241694. URL: https://ssrn.com/abst ract=5241694

[43] A. T. Yang, "Temporal cross-validation impacts multivariate time series subsequence anomaly detection and benchmarking," arXiv preprint arXiv:2506.12183, 2025. URL: https://arxiv.org/abs/2506.12183

[44] H. Arian, D. N. Mobarekeh, and L. Seco, "Backtest overfitting in the machine learning era: A comparison of out-of-sample testing methods in a synthetic controlled environment," Knowledge-Based Systems, Vol. 305, Article 112477, Dec. 2024. DOI: 10.1016/j.knosys.2024.112477

[45] G. Loterman, M. Debruyne, K. Vanden Branden, T. Van Gestel, and C. Mues, "A proposed framework for backtesting loss given default models," Journal of Risk Model Validation, Vol. 8, No. 1, pp. 69-90, 2014. DOI: 10.21314/JRMV.2014.117

[46] Z. Wang, "Validation, robustness, and accuracy of perturbation-based sensitivity analysis methods for time-series deep learning models," Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence (AAAI-24) Undergraduate Consortium, pp. 23768-23770, Vancouver, Canada, Feb. 2024. DOI: 10.1609/aaai.v38i21.30559

Authors

Sang-Hyeok Lee earned an M.S. degree in Physics from Chung-Ang University in 2009 and a Ph.D. degree in Financial Engineering from Ajou University in 2018. He served as a research professor in Applied Mathematics at Gachon University from 2018 to 2019 and as a postdoctoral researcher at the National Institute for Mathematical Sciences (NIMS) from 2019 to 2020. Between 2020 and 2025, he served as a senior researcher at the Korea Maritime Institute (KMI). Dr. Lee currently holds the position of Associate Research Fellow at the Small Enterprise and Market Service (SEMAS). His research interests encompass time series forecasting, deep neural networks, volatility modeling, and the analysis of maritime and small-business markets.

Changho Son received the B.S. degree in Mechanical Engineering from the Korea Military Academy in 2002, the M.S. degree in Industrial Engineering from North Carolina State University in 2006, and the Ph.D. degree in Industrial Engineering from Seoul National University in 2012. Dr. Son is currently a professor in the Department of AI·System Science at the Korea Army Academy at Yeong-cheon, where he also serves as the Director of the Industry-Academic Cooperation Foundation. His research interests include technology management, big data analysis, and artificial intelligence.