Pamukkale Univ Muh Bilim Derg, 32(6), 1293-1303, 2026

Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi

Pamukkale University Journal of Engineering Sciences

Characterization of common temporal features for forecasting of financial time series

Finansal zaman serilerinin tahmininde yaygın zamansal özelliklerin karakterizasyonu

Ata Eren Arslan[1], *Selami Beyhan*[2*]

[1]Çelebi Ground Handling Türkiye, General Directorate, Financial Affairs Directorate, Istanbul, Türkiye.
ata.arslan@celebiaviation.com

[2]Artificial Intelligence and Data Engineering, Electrical and Electronics Engineering Faculty, Yıldız Technical University, Istanbul, Türkiye.
selami.beyhan@yildiz.edu.tr

Received/Geliş Tarihi: 06.10.2025 Revision/Düzeltme Tarihi: 26.01.2026 doi: 10.65206/pajes.68246
Accepted/Kabul Tarihi: 03.02.2026 Research Article/Araştırma Makalesi

Abstract

Purpose: *In forecasting of financial time series, stock prices have similar dynamics due to the sectoral dependencies and common disturbances from the financial market. Therefore, associated stock prices can be used to create features in addition to the past values of the target stock prices.* ***Methods:*** *In this study, a novel common dynamics analysis is proposed between financial time series using five temporal patterns such as seasonality, regime switching, trend, volatility and instantaneous change. These potential dynamics are computed over sliding windows, and a binary covariance matrix is constructed based on the direction of their changes. Subsequently, rule mining is applied to construct the ruleset with highest confidence values where stock prices in rule antecedents are used to construct the feature set. A multiple-input single-output model is proposed and trained using fundamental Long Short-Term Memory (LSTM) architecture.* ***Results:*** *For the unknown future horizons, the values of input stock prices are predicted autoregressively by separate LSTM models since their values are needed when forecasting the primary target stock price values.* ***Conclusion:*** *As a summary, this paper has two major contributions. First, stock prices are analysed with common temporal pattern functions that they provide rule sets for input features. Second, the forecasting performance is highly improved using common dynamics such that the application results confirm that the proposed method improves the forecasting performance up to 28% using the same model.*

Keywords: Stock market prediction, Temporal patterns, Data mining, Long short-Term memory, Common dynamics analysis.

Öz

Amaç: *Finansal zaman serilerinin tahmininde, hisse senedi fiyatları sektörel bağımlılıklar ve finansal piyasalardan kaynaklanan ortak dalgalanmalar nedeniyle benzer dinamikler göstermektedirler. Bundan dolayı, ilişkili hisse senetlerinin fiyatları geçmiş değerlerine ek olarak özellikler üretiminde kullanılabilir.* ***Yöntem:*** *Bu çalışmada, finansal zaman serilerinde bulunan beş zamansal desen (mevsimsellik, rejim değişimi, trend, oynaklık ve ani değişim) kullanılarak yeni bir ortak dinamik analizi önerilmektedir. Bu potansiyel dinamikler kayan pencereler üzerinden hesaplanmakta olup değişim yönlerinin değerlerine bağlı olarak ikili bir kovaryans matrisi oluşturulmaktadır. Ardından en yüksek kural değerine sahip kural kümeleri hakkında bilgi sahibi olmak amacıyla kural madenciliği uygulanmakta, bu kurallardaki öncül hisse senetlerinin fiyatları ise özellik kümesinin oluşturulmasında kullanılmaktadır. Bu doğrultuda, temel bir Uzun Kısa Süreli Bellek (LSTM) kullanılarak çok girişli tek çıkışlı bir model önerilmiş ve uygulanmıştır.* ***Bulgular:*** *Bilinmeyen gelecek görüşleri için, giriş hisse senedi fiyatları, temel hisse senetleri fiyatları tahmininde gerekli olduğundan, ayrı LSTM modelleri ile otoregresif olarak tahmin edilmiştir.* ***Sonuç:*** *Özetle, bu çalışmanın iki ana katkısı vardır. Birincisi, hisse senedi fiyatları, giriş özellikleri için kural kümelerini sağlayan ortak zamansal desen fonksiyonları ile analiz edilmiştir. İkincisi, tahminleme performansı ortak dinamikler ile geliştirilmiştir. Uygulama sonuçları göstermektedir ki önerilen yöntemin aynı model ile kıyasla performansı %28 oranında artırdığını doğrulamaktadır.*

Anahtar kelimeler: Borsa tahmini, Zamansal desenler, Veri madenciliği, Uzun-Kısa süreli hafıza, Ortak dinamikler analizi.

1 Introduction

The stock market is one of the hardest time series for analysis and prediction which is always changing and unpredictable about what will happen next due to unforeseen reasons. Stock markets are determined by a host of factors, including economic indicators and company performance, not forgetting investor sentiment. A correct prediction for the stock prices offers advantages to investors as making informed decisions on the management of associated risks [1]. However, conventional methods applied within statistical models and basic machine learning algorithms are normally incapable of dealing with the complexity and non-linearity of financial data. For this reason, people are looking for new methods that produce more detailed and accurate forecasts to understand upward and downward market changes. Association rule mining (ARM) is one of the fundamental methods for data mining that provides the association rules against a large database of customer transaction patterns [2].

ARM identifies such meaningful relationships between the items efficiently by embedding buffer management with novel estimation and pruning techniques. Results on sales data from a major retail company prove that the algorithm is very effective in uncovering valuable insights into customer purchase [3] improved the ARM methods for sequential patterns. Recently, [4] addresses the limitations of traditional sequential rule mining algorithms by proposing the mining of ”partially ordered sequential rules” (POSR), which generalize sequential rules by allowing unordered items in both the

*Corresponding author/Yazışılan Yazar

antecedent and consequent. The application of the sliding-window constraint significantly reduces the number of discovered rules, and a real-world application confirms that POSR offers higher prediction accuracy than regular sequential rules. ARM methods use in various industrial areas. ARM helps identify frequently co-occurring products and increasing productivity in data mining techniques [5],[6]. Nowadays, ARM has regained attention for the mining of faults for predictive maintenance of large machines.

Long Short-Term Memory (LSTM) cell was introduced in [7] that addresses the vanishing gradient problem in Recurrent Neural Networks (RNN). LSTM based neural network models are particularly suitable for extracting long-term dependency of data with learning over long time intervals. Using constant error flow through special units and multiplicative gate units, Recently, more complex neural network models have been preferred because of their immense effectiveness in time series forecasting and feature extraction. Recent studies highlight the limitations of traditional methods like Linear Regression, AutoRegressive Integrated Moving Average (ARIMA) in stock market predictions due to the complexity and randomness of stock prices. Hybrid models have been developed to enhance classical ARIMA technique by extracting temporal components such as trend and seasonality. For instance, wavelet-based ARIMA combined with seasonal features and results show that decomposing time series into temporal features give better accuracy in forecasting [8].

Deep learning models, particularly LSTM, overcome these issues by effectively using historical data for future forecasting [9]. LSTMs are preferred over Convolutional Neural Network in financial time-series predictions because of their ability to capture long-range temporal patterns through recurrent mechanisms [10]. Multi-input LSTM models further enhance prediction accuracy by processing multidimensional data, making them vital in fields like finance and natural language processing [11].

Beyond recurrent architecture, recent studies emphasize that temporal model strategies are crucial for identifying informative time segments and financial time series forecasting. Specifically, temporal attention mechanisms improve prediction accuracy by dynamically weighting time periods rather than treating all historical dataset equally. These kinds of approaches highlight the importance of extracting temporal features from time series data, even though they do not explicitly focus on common temporal pattern extraction across assets [12].

Recent comprehensive reviews on financial time series forecasting highlight the limitations on traditional statistical models. The research emphasized the importance of additional signal-based feature extraction optimization techniques and handling in volatile, nonlinear, and non-stationary financial data. Reviewers note that raw data into learning models is not sufficient for capturing temporal dynamics due to its noise, non-stationarity, and model parameter uncertainties that integrated approaches combining feature extraction and intelligent learning offer improved forecasting accuracy and robustness. In the review, several categories of temporal features are identified, including trend- and momentum-based, volatility- and risk-based, structural and regime-based, short-term and local dynamics-based, and periodic or recurrent-based temporal characteristics [13].

Recently, significant contributions are published on the time series forecasting. Graph neural networks with contrastive learning have been proposed for representation of multivariate time series under self-supervising approach where temporal graph-based contrastive learning frameworks represent assets as nodes and evolving relationships as edges, enabling the extraction of structured temporal representations for forecasting and anomaly detection [14]. While these models demonstrate strong predictive performance, they often rely on complex graph construction procedures and high computational costs, which may limit their interpretability and practical applicability [15]. An adaptive sliding window normalization method is introduced for time varying sliding window based on anomaly detection with clustering methods and statistical modelling in [16]. Stock prices have been predicted and compared with literature methods.

[17] focuses on the Chinese stock market where it uses deep learning algorithms to forecast stock prices under various market conditions. [18] highlights the human factor in financial markets by showing that collective analyst attention plays an important role in reducing uncertainty as analysts focus more, markets become clearer, price discovery improves, and volatility diminishes.

Develops machine learning models using multimodal data to forecast sudden stock price changes [19]. The combination of textual, financial, and behavioural data improves the performance of the deep learning models. [20] proposes a DLWR-LSTM model for forecasting stock market indices, which integrates dynamic local weighting with recurrent neural-networks. The approach achieves superior accuracy compared to conventional LSTM and statistical models. [21] demonstrates how incorporating causal relationships into time series forecasting models can significantly improve prediction accuracy. Advanced models like LSTM-HIT combine high-frequency market data and volatility information for improved financial volatility forecasting. Like the present study, a novel methodology presents a competitive approach by integrating low-frequency indicators into high-frequency volatility models that enables more accurate long-term forecasts [22]. By extracting the mixed-frequency structure of returning and trading data, this study [23] provides a novel method to offer new results into how foreign investors react to information at different times of the day. Based on the generative modelling approach, Generative Adversarial Networks based data augmentation and transformer-driven attention have been integrated to capture the nonlinear dependencies in stock price dynamics [24]. In this research, a constant sliding window was initially adopted for simplicity and interpretability. Nevertheless, recent studies suggest that adaptive sliding-window mechanisms based on fuzzy systems are more capable of modelling the non-stationary and regime-dependent behaviour of financial time series [25]. Furthermore, recent literature emphasizes that causal and structural temporal representations provide deeper insights than purely correlation-based modelling approaches.

Despite the extensive use of deep learning and feature extraction methods in financial time series forecasting, existing models show us they are predominantly focused on individual stock dynamics or static pairwise relationships. Most approaches rely on basic correlation-based structures and treat multivariate inputs as raw features, without explicitly modelling synchronized structural and statistical changes across multiple stocks. As a result, the extraction of

interpretable and dynamic cross-asset temporal patterns remains limited.

In this paper, a new approach is proposed with common dynamics analysis and deep learning to forecast stock prices in the NASDAQ (USA) and BIST100 (Turkey) markets. The datasets include daily closing prices of stocks from March 2020 to March 2024, respectively. Five temporal patterns were calculated over sliding time windows such that increments or decrements of the dynamics are binary coded in a matrix form. In order to rule the common frequent dynamics for both stock markets, association rules were generated using the well-known Apriori algorithm. The association rules based on sufficient support and confidence levels produced common movements among the stock prices to be base for input stocks of the prediction phase. In forecasting part, LSTM models were trained for individual stocks using both their past values and input stocks from the rule bases. According to the application results, using rule-based relations with five common temporal patterns along with LSTM improves the prediction performances compared to using LSTM model alone. The rule-based input patterns of stock prices simply helped the LSTM model catch interactions between stock prices. The application results confirm that the proposed method improves the forecasting performance up to 28% using the same model.

Based on the discussed literature gaps, this paper addresses the following research questions.

- RQ1: Can common temporal dynamics be extracted from multiple financial time series using pattern functions and association rule mining?
- RQ2: Do rule-based common temporal dynamics provide more useful and interpretable input features compared to using only the past values of a single stock in LSTM-based forecasting models?
- RQ3: How do different temporal dynamics (seasonality, regime switching, trend change, volatility change, and instantaneous change) affect forecasting performance across different stock markets (NASDAQ and BIST100)?

2 Common temporal features in financial forecasting approach

This section highlights the methodological framework of the proposed model. The main framework consists of four main stages. First, time-series functions are applied to stock price series using sliding windows to extract dynamic properties. Second, these dynamics are converted into binary form to create common dynamic matrices that capture simultaneous or inverse movements between assets. Third, relationship rule mining is performed on the bivariate matrices to determine reliable co-movement models based on support, confidence, and lift values. Finally, a multi-input single-output LSTM model integrates the target stock's past values with rule-based related stocks to improve prediction performance. In Figure 1 the proposed model is charted.

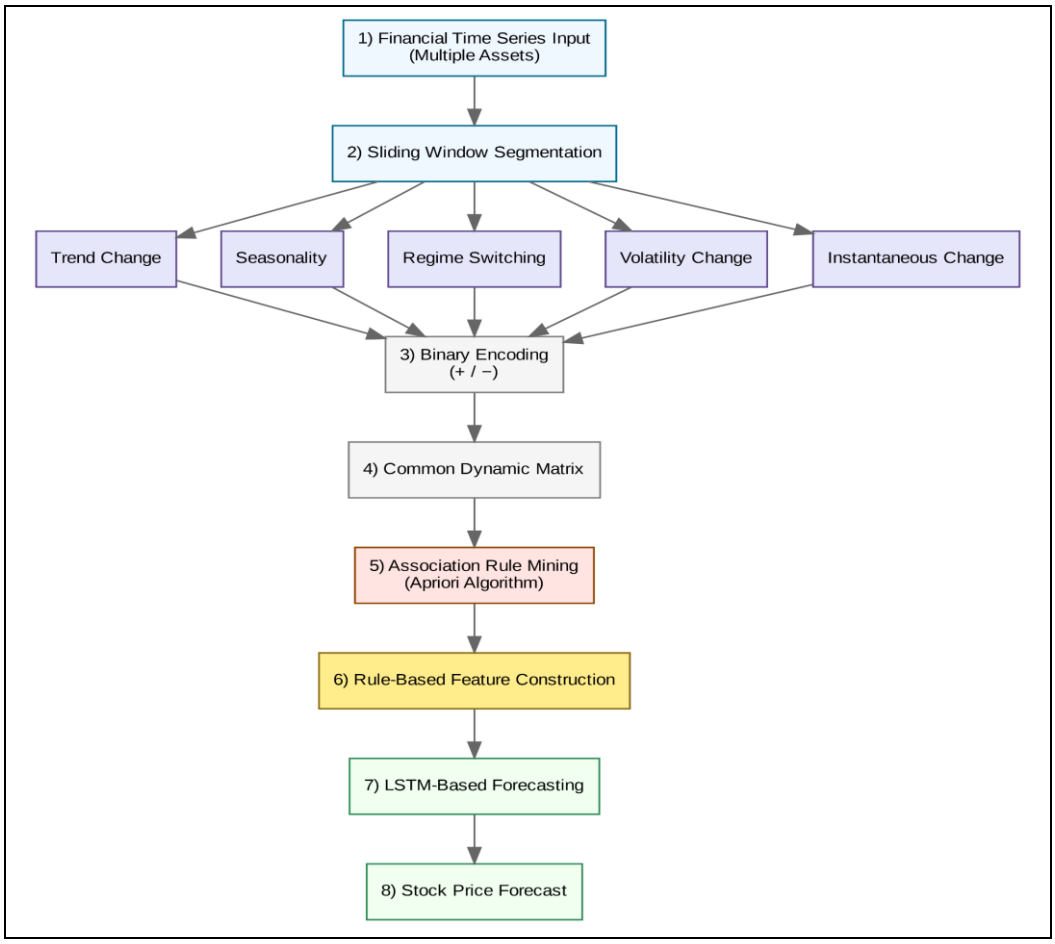


Figure 1. Proposed network architecture.

1.1 Temporal patterns of time series

This section introduces five functional common dynamics of time series. These functions will be used to calculate properties such that their increment and decrement will be coded as binary matrix to find a common change matrix. Their calculations might be seen similar but they produce different common dynamics of time series at the end.

Seasonality: Seasonality is the regular fluctuations in time series that recur at regular intervals. These fluctuations are usually associated with periodic time periods, such as certain periods of the year and days of the week.

i) Moving Average-Based Detection: Let X = {x1, x2, . . . , xT } be a univariate time series of length T. Apply a moving average filter like Equation 1 with window size w to smooth short-term fluctuations:

$$MA_t = (1/w)\sum_{i=t-w+1}^{t} x_i \quad (1)$$

Calculate the shifted difference to detect sudden regime change as shown in Equation 2.

$$\Delta MA_t = MA_t - MA_{t-1} \quad (2)$$

A binary representation can be used for pattern discovery using association rule mining algorithms. If the change exceeds a threshold τ,a seasonal change is considered to be in Equation 3.

$$S_t = \{1, if|\Delta MA_t| > \tau\ 0, \quad otherwise \quad (3)$$

ii) Fourier Series-Based Detection: For detecting periodicity, Fourier series decomposition is formulized in Equation 4.

$$x_t = a_0 + \sum_{k=1}^{K}\left[a_k cos\left(\frac{2\pi kt}{T}\right) + b_k sin\left(\frac{2\pi kt}{T}\right)\right] + \varepsilon_t \quad (4)$$

Here, T is the seasonal period (e.g., 12 months), a_k and b_k are Fourier coefficients, and K is the number of harmonics. This method captures the cyclical structure of the series in the frequency domain, making it useful for modeling complex seasonal behaviors. In this study, seasonality is detected by the moving average-based method. Time series are smoothed by moving-average and meaningful regime changes are detected based on a threshold on first-order differences. The resulting binary matrix is inspected using association rule mining techniques.

Volatility Change: Volatility refers to the degree of variation in a time series, indicating the level of risk or uncertainty. In financial applications, it is commonly measured as the standard deviation of log returns over a rolling window. Let P_t be the price at time t, and define log return as shown in Equation 5.

$$r_t = ln\,(P_t/P_{t-1}) \quad (5)$$

The rolling volatility (standard deviation) over a window of size N is calculated as shown in Equation 6.

$$\sigma_t = \sqrt{(1/N)\sum_{i=t-N+1}^{t}(r_i - \mu)^2} \quad (6)$$

where μ is the mean of the log returns within the window. Binary transformation for pattern mining is given as follows. Volatility change is detected by taking the first-order difference as shown in Equation 7.

$$\Delta\sigma_t = \sigma_t - \sigma_{t-1} \quad (7)$$

A binary signal is generated based on whether the volatility is increasing as show in Equation 8.

$$VC_t = \{1, \quad if \quad \Delta\sigma_t > 0, \quad otherwise \quad (8)$$

In this study, volatility is calculated using the rounded standard deviation of log returns. The change in volatility over time is used to construct a binary matrix where an increase in volatility corresponds to a value of 1. This binary matrix is then used in association rule mining to discover patterns of co-movement between assets.

Regime Switching: Regime switching refers to structural changes in the behavior of a time series, where the series transitions between distinct states such as high and low volatility. These shifts often represent different market dynamics and can be modeled using probabilistic or empirical methods.

i) Markov *Regime Switching Model (MSM):* A probabilistic approach that assumes that the time series operates under multiple latent regimes as shown in Equation 9. The state of the regime at time t is denoted by S_t and follows a first-order Markov process:

$$P(S_{t-1} = j) = P_{jk} \quad (9)$$

Each regime can have its own statistical properties, such as mean or variance. The transition probabilities P_{jk} determine the likelihood of moving from regime *j* to regime *k*. Widely used in financial modeling to capture shifts in market volatility or policy regimes.

ii) *Moving Average-Based Volatility Detection:* A simpler and practical alternative to MSM that relies on volatility estimation from returns. Let $r_t = (p_t - p_{t-1})/p_{t-1}$ be the return series.

Compute the rolling standard deviation of returns over a window *w* as shown in Equation 10.

$$\sigma_t = std(r_{t-w+1}, \dots, r_t) \quad (10)$$

Define the regime shift signal as the first-order difference of the rolling volatility as shown in Equation 11.

$$\Delta\sigma_t = \sigma_t - \sigma_{t-1} \quad (11)$$

If this change exceeds a specified threshold τ, a regime shift is indicated as shown in Equation 12.

$$RS_t = \{1, \quad if\ |\Delta\sigma_t| > \tau, \ otherwise \quad (12)$$

The binary regime matrix can be used in pattern discovery tasks such as association rule mining. In this study, regime change detection is performed using the volatility method based on moving averages. The rolling standard deviation of returns is calculated, and significant volatility changes are identified using a threshold. Binary indicators are then analysed using association rule mining algorithms to uncover common movement patterns among financial assets.

Trend Change: Trend analysis identifies whether a time series exhibits a general increase, decrease, or stationarity over a specific period. Fundamental for understanding long-term movements in financial, economic, or social time series.

i) *Linear Regression-Based Trend Detection*: A simple linear model can be fitted to the time series like Equation 13.

$$y_t = \alpha + \beta_t + \varepsilon_t \quad (13)$$

Here, β the slope coefficient has meaning as follows: $\beta > 0$ implies an upward trend, it $\beta < 0$ implies a downward trend, $\beta \approx 0$ implies no significant trend (stationary). Equation 14 shows the trend signal can be binarized.

$$T_t = \{1, \quad if\ \beta > 0, \quad if\ \beta \leq 0 \tag{14}$$

ii) *Moving Average-Based Trend Change Detection:* Equation 15 the rolling average is computed to capture the short-term trend as

$$MA_t = (1/w) \sum_{i=t-w+1}^{t} x_i \tag{15}$$

The change in trend is calculated as the first-order difference using Equation 16.

$$\Delta MA_t = MA_t - MA_{t-1} \tag{16}$$

where if this change exceeds a defined threshold τ, a trend change is calculated as shown in Equation 17.

$$TC_t = \{1, \quad if\ |\Delta MA_t| > \tau, \quad otherwise \tag{17}$$

This binary signal can be used for pattern mining, such as with association rule learning. In this study, the trend change is detected using a moving average-based method. A specific moving average was performed to the original time series and the change in this average was tracked. When the absolute change exceeded a predefined threshold, the trend transition was recorded as a binary signal. The resulting matrix was examined by mining association rules to identify common trend patterns across different assets characteristics.

Instantaneous Change: Stock instantaneous change analysis is one of the methods used to trace the behaviour of stock prices over time, focusing on the identification of the increase and decrease patterns. Such an approach will be of immense help in comprehending the co-movements of the stocks, which is very critical in the management of a portfolio, assessment of risk, and the building of investment strategies. Simply stated, stock change analysis furnishes observations, allowing for a formal

analysis of dynamic stock price behaviour over time. Its outcomes enable better-informed financial decisions. For the given time series x_t, $t \epsilon [0, T]$ the instantaneous change is calculated as $|\Delta t| = x_t - x_{t-1}$ where this dynamic is based on each time change. Therefore, it has more common dynamics compared to the previous structural dynamics.

$$IC_t = \{1, \quad if\ |\Delta t| > 0, \quad otherwise \tag{18}$$

Equation 18 can be considered with respect to both increment and decrement of the series.

1.2 Construction of a common dynamic matrix using temporal patterns

In time series, stock price data are collected from a financial market, where each column in the dataset represents a different stock and each row corresponds to a specific point in time. The binary matrix is constructed using a sign-based encoding scheme that extracts and captures the change of direction in temporal dynamics. Increased and decreased values of stocks are encoded as 1 and 0. This strategy aims to avoid using magnitude-based thresholds and provides robustness and co-movement across financial time series. As an example, the instantaneous change property is here used for the common dynamics' matrix. For every stock, price changes are calculated by taking the difference between the current price and its value at the previous time step. Based on these changes, two binary indicators are created for each stock: one indicating whether the price has increased (assigned 1 if true, 0 otherwise), and another showing whether the price has decreased (again 1 if true, 0 otherwise). These binary indicators form a new dataset that captures the upward and downward movement patterns of all stocks over time. This transformed data set is then used to identify stocks that exhibit similar behaviour, either moving in the same direction at the same time (synchronous movement) or in opposite directions (inverse movement).

Example: Assume that a simple case of three stocks, A, B, and C are collected at five time points. Performing the instantaneous change analysis, we can create the following binary indicator columns given in Table 1. Followings can be observed in this new dataset patterns: Stocks that have often jointly increased: Stock B and Stock C jointly increased from T3 to T4 and T4 to T5. Stocks that often decrease: Stock A and Stock B decreased from T1 to T2. Inverse movement of stocks: An increase in one stock corresponds to a decrease in another stock.

Common dynamics matrix between stocks can be utilized for market trend analysis to identify and ascertain future market movements, and for investment strategies to ensure that the co-movement of stocks can help maximize returns or minimize losses, and for portfolio diversification to know that the stocks that 'move together' will help diversify portfolios to manage associated risk.

1.3 Association rule mining based on common dynamics

Association Rule Mining (ARM) is one of the important data mining techniques that discover interesting relationships between various variables in large databases. Since, binary matrix is constructed bi-directional movements the resulting ARM rules give us emphasizing of structural synchronization among stocks. It is crucially important for the extraction of patterns, correlations, and structures from data, and is immensely useful in domains like market basket analysis, medical diagnosis, and financial forecasting. ARM relies on the notion of frequent itemset discovery and generating association rules from these itemsets.

Itemset and Transactions: Let $I = \{I1, I2, \ldots, In\}$ be a set of items. Let D be a database consisting of a set of transactions. Every transaction T has a unique transaction ID and contains a subset of items in I.

Table 1. example stock prices and common dynamics matrix with ınstantaneous change.

Time	Stock A	Stock B	Stock C	Increase A	Increase B	Increase C
T1	100	200	300	-	-	-
T2	105	195	310	1	0	1
T3	102	205	305	0	1	0
T4	110	210	315	1	1	1
T5	108	215	320	0	1	1

Support (s) of an itemset is defined as the proportion of transactions in the database D that contain the itemset. For the rule X ⇒ Y, support is given in Equation 19.

$$support(X \Rightarrow Y) = P(X \cup Y) \quad (19)$$

Confidence (c) of a rule is the proportion of transactions containing X that also contain Y. It is a measure of the reliability of the inference made by the rule in Equation 20.

$$confidence(X \Rightarrow Y) = \frac{support(X \cup Y)}{support(X)} \quad (20)$$

Lift is used to evaluate the strength of a rule over the random co-occurrence of X and Y, assuming independence. A lift value greater than 1 suggests that X and Y appear together more frequently than expected by chance as formuled in Equation 21.

$$lift(X \Rightarrow Y) = \frac{support(X \cup Y)}{support(X) \times support(Y)} \quad (21)$$

Minimum support (minsup) and minimum confidence (minconf) are user-defined thresholds used to filter out uninteresting rules. Commonly, the generation of association rules is composed of the following steps: Frequent Itemset Generation: Identify all the itemsets with support greater than or equal to the minimum threshold for support; this can be computationally expensive, but algorithms such as Apriori, Eclat, and FP-Growth optimize this step. Rule Generation: Generation of all possible association rules from the frequent itemsets such that confidence of every rule is above the minimum confidence threshold. This step involves confidence evaluation for every possible rule.

1.4 Long short-term memory model with data-driven features

Long Short-Term Memory (LSTM) networks, a special type of recurrent neural networks (RNN), have gained wide application in time series forecasting due to the possibility of learning long-term terms and ideally avoiding the loss gradient problem commonly present with traditional RNNs. They were first introduced by [7] such that they include memory cells and the addition of gating mechanisms to control information flow, making them effective learners of patterns over extended intervals. An illustrative LSTM cell is shown in Figure 2.

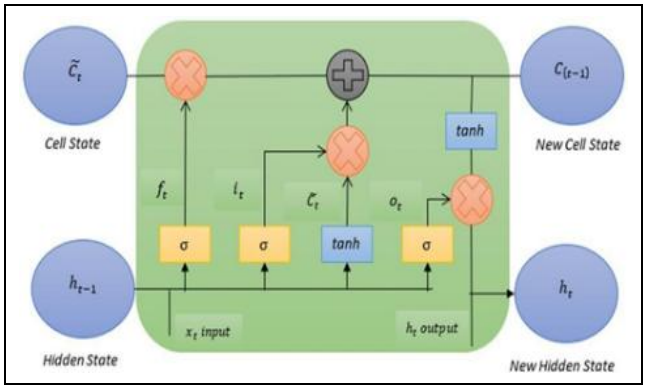


Figure 2. LSTM network architecture.

An LSTM network contains several layers, and each layer has interconnected memory cells. The architecture has input gates, forget gates and output gates that together control the flow of information. The equations of an LSTM cell illustrated in Figure 2 are

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad (22)$$

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad (23)$$

$$\widetilde{C}_t = tanh\,(W_C[h_{t-1}, x_t] + b_C) \quad (24)$$

$$C_t = f_t \times C_{t-1} + i_t \times \widetilde{C}_t \quad (25)$$

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (26)$$

$$h_t = o_t \cdot tanh\,(C_t) \quad (27)$$

such that at each time step t, the model processes the current input x_t and the previous hidden state h_{t-1}. The input gate determines how much of the new information should enter the memory, calculated using a sigmoid activation applied to the linear combination of h_{t-1} and x_t. Similarly, the forget gate f_t decides how much of the previous cell state C_{t-1} should be retained. A candidate cell state $\widetilde{C}_t$ is generated using the tanh function, which introduces non-linearity. The actual cell state Ct is then updated by combining the retained past memory and the newly proposed candidate, weighted by their respective gates. The output gate ot regulates how much of the updated cell state should influence the current output. Finally, the hidden state ht is obtained by applying tanh to the updated memory and scaling it by the output gate. Together, these gated operations allow the LSTM to preserve and adapt information over long sequences without suffering from vanishing or exploding gradient problems. Equations between 22 and 27 formulized the LSTM processes.

In the present study, LSTM model is used to predict the closing prices of stocks. The dataset contained daily stock prices from the past several years. The architecture of the model includes an input layer, two LSTM layers, and a dense output layer. The model was trained using the Adam optimizer with a mean squared error (MSE) loss function. The performance is evaluated on a test set and the results show a significant improvement in forecast accuracy compared to conventional methods. The MSE score of the test set was significantly lower than that of the baseline autoregressive model, highlighting the superior ability of the LSTM model to capture non-linear dependencies and long-term patterns. Hyperparameters of present study as shown in Table 2.

Table 2. Hyperparameters of present study.

Categories	Parameters	Values/Descriptions
Data	Dataset	Historical Dataset
Model Architecture	Window Size	Stable Window Size/ 7
Model Architecture	LSTM Layers	2 Staked Layers /50
Model Architecture	Batch Size	32
Model Architecture	Learning Rate	0.05
Model Architecture	Epoch	50
Model Architecture	Loss Function	Adam
Model Architecture	Activation Function	Sigmoid

Developing machine learning models is splitting the dataset into portions used for training and testing. This ensures that the model's performance and generalization ability accurately.

Training Data: This is the dataset used to train the model. The model adjusts its parameters based on this data.

Validation Data: This dataset is used to tune the model's hyperparameters and prevent overfitting. It provides an indication of how well the model is training.

Test Data: After training and tuning with validation data, the test data is used to evaluate the model's final performance. Since the model has never seen this data before, it helps in assessing its generalization capability. In the present study, we split the dataset into training data (%70), validation data (%10), and testing data (%20) portions. Proper dataset partitioning is necessary to fine-tune the neural network's hyperparameters and achieve the best modelling accuracy.

Data Normalization: Process of transforming features to a common scale, typically within a fixed range such as [0, 1] or [-1, 1]. This technique enhances the performance and efficiency of machine learning algorithms. In addition, input-output data must be normalized to a proper interval to prevent the saturation of network activation functions.

Min-Max Normalization: Scales the data between the minimum and maximum values. Formulized in Equation 28.

$$X_{norm} = (X - X_{min})/(X_{max} - X_{min}) \quad (28)$$

Z-Score Normalization (Standardization): Transforms the data to have a mean of 0 and a standard deviation of 1 seen in Equation 29.

$$X_{std} = (X - \mu)/\sigma \quad (29)$$

where μ is the mean, σ is the standard deviation. For presenting the advantage of proposed methodology, the following performance metrics are used to evaluate the model's prediction accuracy as shown in Equation 31 and Equation 32.

MAE (Mean Absolute Error): Measures the average absolute difference between predicted and actual values.

$$MAE = (1/N)\sum_{i=1}^{N}|y_i - \hat{y}_i| \quad (30)$$

Equation 30 where,y_i represents the actual value, and $\hat{y}_i$ is the predicted value.

RMSE (Root Mean Squared Error): The square root of the average of the squared differences between predicted and actual values, giving more weight to larger errors.

$$RMSE = \sqrt{(1/N)\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \quad (31)$$

MAPE (Mean Absolute Percentage Error): Measures the average absolute percentage difference between predicted and actual values.

$$MAPE = (100/N)\sum_{i=1}^{N}|(y_i - \hat{y}_i)/y_i| \quad (32)$$

3 Results and discussion

In this section, first, rule sets between the stock prices are constructed and analyzed according to common dynamics. Second, using high confidence and support of rules, the input feature data set is created for target stock price forecasting.

3.1 Analysis on common temporal patterns

For the comprehensive data analysis and prediction, TSLA (from NASDAQ-100) and ARCLK (from BIST-100) stock prices have been used with recent five-years data. Constant sliding window is determined as 7 days. Table 3 and Table 4 present the extracted association rules for the chosen stock prices, each one analysed under five different structural and statistical market dynamics with 7 days sliding-windows. The chosen window size reflects a short-term temporal horizon commonly adopted in financial time series analysis, allowing the method to capture local co-movement patterns while avoiding excessive smoothing effects. Several window size during forecasting simulation phases to estimate the sensitivity of the parameter. The experimental results indicated that increasing the window size of the deep learning model leads to higher forecasting ratios. Therefore, it tends to dilute temporal dynamics and reduce the effectiveness of rule based future extraction. The rule with highest confidence values is listed in the tables by considering five different structural and statistical dynamics of the time series. The tables highlight how the antecedent sets derived from different stocks contribute to predicting the target stock movements in terms of support, confidence, and lift. Notice that these rules will be used for feature design of the time series prediction. Followings are the main outcomes from the common dynamics analysis.

For ARCLK, the highest confidence (1.0) and a remarkably high lift (15.90) exhibit at the regime switching. It means that whenever the antecedent stocks (ASELS, SAHOL, TCELL) has a regime shift, ARCLK follows these stocks with absolute certainty.

On the other hand, the instantaneous change provides the highest support value (0.2076), indicating that this condition frequently occurs in the data and is still highly predictive, with a confidence (0.8927) and a respectable lift (1.65). The dynamics seasonality and volatility-based have weak associations both in confidence and lift, implying that smoother, recurring patterns or relations may not be as strongly predictive for ARCLK.

Table 3. Association rules with highest confidence values for *ARCLK* stock price using five different dynamics.

Dynamics	Antecedents	Consequent	Confidence	Support	Lift
Seasonality	(SAHOL, KCHOL)	ARCLK	0.1098	0.6215	2.2400
Regime Switching	(ASELS, SAHOL, TCELL)	ARCLK	0.0110	1.0000	15.9048
Trend Change	(KCHOL, SISE, BIMAS, GOLD, TOASO)	ARCLK	0.2006	0.9054	1.6435
Volatility Change	(GARAN, EREGL, ISCTR, SAHOL)	ARCLK	0.1058	0.7794	1.5557
Instantaneous Change	(TOASO, KCHOL, EREGL, SISE)	ARCLK	0.2076	0.8927	1.6534

Table 4. Association rules with highest confidence values for *TSLA* stock price using five different Dynamics.

Dynamics	Antecedents	Consequent	Support	Confidence	Lift
Seasonality	(AMZN, NFLX)	TSLA	0.2185	0.8059	1.2601
Regime Switching	(AMZN, MSFT, ADBE, AAPL)	TSLA	0.0109	1.0000	3.5333
Trend Change	(AVGO, ADI, AMZN, QCOM, AAPL)	TSLA	0.2075	0.9905	1.1108
Volatility Change	(ADBE, EA, AAPL, MSFT)	TSLA	0.1013	0.7445	1.4558
Instantaneous Change	(COST, MSFT, NVDA, ADI, ADBE)	TSLA	0.2016	0.8423	1.5825

In the case of TSLA, similar associations are observed with the regime switching dynamics. It achieves a confidence of 1.0, with a lift value of 3.53, points that a strong and reliable link between structural market dynamics in the antecedent stocks (AMZN, MSFT, ADBE, AAPL) and TSLA's dynamics. Interestingly, seasonality shows the highest support value (0.2185), though its confidence and lift are relatively small compared to regime switching. The trend change of dynamics yields very high confidence (0.9905) but relatively low lift value, presents a generally reliable relationship but not especially unique pattern of co-movement.

Instantaneous and volatility-related patterns provide balanced performances, with decent support and moderately strong lift values, reinforcing the idea that sudden, sharp market movements carry highly informative signals for forecasting TSLA's near-term behaviour. Pulling the lens back, a compelling narrative emerges from these results. It appears that regime-based dynamics, while less frequent, act like high-impact economic earthquakes—their occurrences are rare, but they provide the most structurally informative and decisive signals for both stocks. They are the key, albeit occasional, drivers of deep-seated co-movement. So far considering them together, these results suggest that regime-based dynamics have strong associations although less frequent these events tend to be the most structurally informative for both stocks.

Meanwhile, other dynamics with high support values are the trend or instantaneous changes that offer more consistent but less exceptional predictive power. This behaviour is valuable in model design, helping balance between rare but powerful signals and more frequent but weaker indicators.

3.2 Forecasting results based on feature engineering

This section indicates the modelling and prediction results of applying the LSTM model to forecast the stock prices with extracted association rules. The antecedent stock prices of the association rules are employed as input stock prices with past values of the chosen stock price. Tables 3 and 4 are completely used and compared for predictions then we will be able to outline such conclusions about common dynamic analysis-based feature construction for prediction. For a core comparison, stock prices are initially predicted without the use of common dynamics analysis. The first rows of Table 5 and Table 6 give the performance values of predictions without common dynamics analysis. It is expected that common dynamics analysis extracts input information from other stock prices. The proposed methodology leads to a significant improvement in forecasting accuracy compared to without CDA case.

Table 5 presents the forecasting performance of ARCLK stock under different market conditions based on selected association rule mining (ARM) results with MAE, MAPE and RMSE metrics. For each of the market dynamics such as seasonality, regime change, trend reversals, volatility change and instantaneous change, the most significant rule (with the highest confidence value) is selected from the relevant ARM outputs, and the forecasting model is constructed. The antecedents in all columns represent the characteristics employed in the model and ARCLK is the modelled target variable (consequent).

The table demonstrates the contribution of different dynamics effects towards forecasting accuracy, indicating the importance of cause-and-effect patterns in financial forecasting. After evaluating the association rules using an LSTM prediction framework, a different pattern have been obtained in Table 5 considering the highest confidence value. The trend change of dynamic delivers the most acceptable forecasting metrics across all three evaluation criteria MAE, MAPE, and RMSE, respectively. It indicates that trend change dynamic enables the model to learn and generalize effectively. Notice that the trend change has both higher values of support and confidences compared to regime switching. Interestingly, the regime switching scenario, despite its strong two association rule indicators, performs less impressively in the LSTM setting, with relatively higher error values.

This discrepancy may cause from the sparse nature of regime-based events so that LSTM model cannot learn without overfitting. Another important finding is the consistent utility of instantaneous change dynamic, which provide both high support and decent predictive performance. Application results here suggest that frequent, short-term market dynamics provide both reliable associations and useful input feature signals with other time series for sequence-based prediction models like LSTM.

Table 5. Prediction performances for *ARCLK* under different market dynamics.

Dynamics	Antecedents	Consequents	MAE	MAPE	RMSE	Improvement (%)
Without CDA	Past Values	ARCLK	4.56	3.62%	5.58	-
Seasonality	(SAHOL, KCHOL)	ARCLK	3.51	2.71%	4.56	18.32%
Regime Switching	(ASELS, SAHOL, TCELL)	ARCLK	3.59	2.77%	4.42	20.86%
Trend Change	(KCHOL, SISE, BIMAS, GOLD, TOASO)	ARCLK	3.07	2.33%	4.03	27.74%
Volatility Change	(GARAN, EREGL, ISCTR, SAHOL)	ARCLK	4.18	3.24%	5.08	8.96%
Instantaneous Change	(TOASO, KCHOL, EREGL, SISE)	ARCLK	3.22	2.43%	4.25	23.87%

Table 6. Prediction performances for *TSLA* under different market dynamics.

Dynamics	Antecedent	Consequents	MAE	MAPE	RMSE	Improvement (%)
Without CDA	Past Values	TSLA	6.17	2.79%	8.24	-
Seasonality	(AMZN, NFLX)	TSLA	6.82	3.00%	8.74	-6.07%
Regime Switching	(AMZN, MSFT, ADBE, AAPL)	TSLA	6.77	3.02%	8.70	-5.59%
Trend Change	(AVGO, ADI, AMZN, QCOM, AAPL)	TSLA	6.10	2.76%	7.95	3.51%
Volatility Change	(ADBE, EA, AAPL, MSFT)	TSLA	5.71	2.61%	7.35	10.80%
Instantaneous Change	(COST, MSFT, NVDA, ADI, ADBE)	TSLA	5.51	2.51%	7.29	11.47%

Table 6 presents how TSLA's prediction performance varies under different market dynamics. Including past values of time series, the model integrates structural and statistical dynamic indicators. For TSLA stock, the instantaneous change dynamics scenario yields the lowest error metrics overall, with the smallest RMSE, MAE, and MAPE performance values. It suggests that patterns associated with price variability in stocks like COST, MSFT, NVDA, ADI and ADBE provide substantial predictive information for TSLA. Interestingly, the volatility change dynamics also delivers applicable results. Both mean that sudden co-movements among antecedent stocks might serve as timely predictors. In contrast, dynamics based on seasonality or regime switching show relatively higher error rates, meaning that recurring or structural market dynamics may not align as tightly with TSLA's short-term movements. The findings emphasize the benefit of incorporating dynamic and abrupt signals especially tied to market instantaneous change into data- driven forecasting frameworks for TSLA. Figure 3 and Figure 4 presents the training and testing performance of the developed prediction model for ARCLK stock including the CDA (Common Dynamics Analysis). Figure 5 and Figure 6 presents the training and testing performance of the developed prediction model for TSLA stock including CDA.

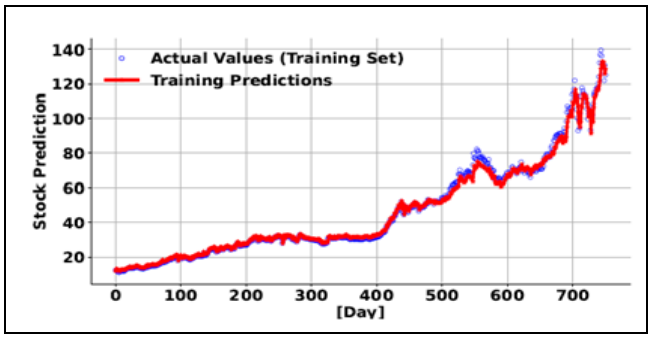


Figure 3. ARCLK: Training data predictions.

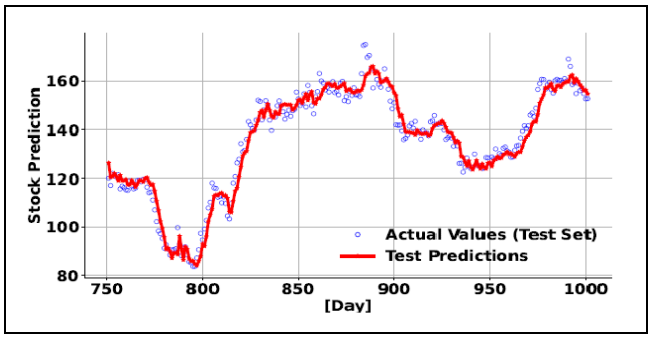


Figure 4. ARCLK: Testing data predictions.

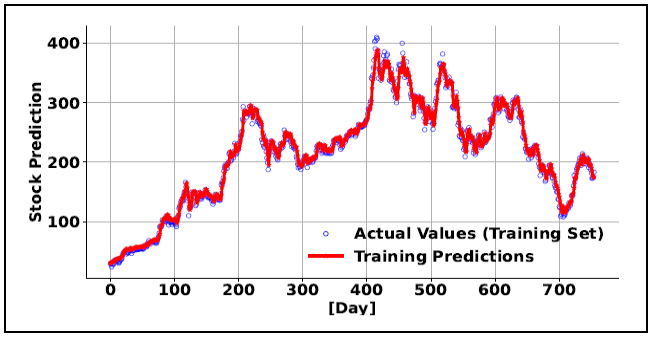


Figure 5. TSLA: Training data predictions.

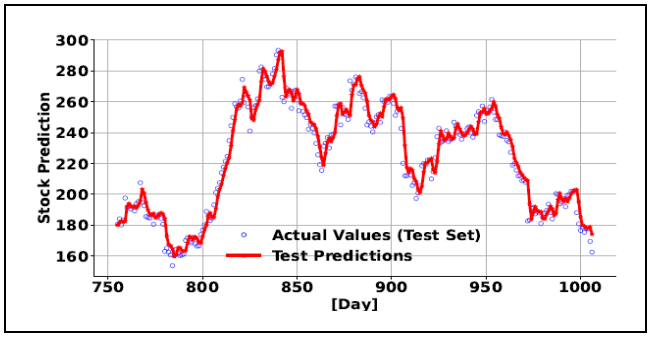


Figure 6. TSLA: Testing data predictions.

These plots are corresponding to the best performance values of the predictions Table 5 and Table 6, respectively. Overall, the contrast between the association rule and prediction tables highlights the difference between rule-based interpretability and model-driven learning capacity. While association rules reveal strong one-shot dependencies (especially in rare events like regime shifts), LSTM models benefit more from patterns that are both meaningful and statistically frequent such as trend and instantaneous change.

3.3 Discussion of research questions

RQ1 is addressed by constructing binary co-dynamic matrices derived from five predefined temporal model functions. The association rules extracted with high support, confidence, and lift values (Tables 3 and 4) demonstrate that common temporal behaviours among stocks can be systematically identified without relying on correlation-based measures.

RQ2 is answered by comparing the prediction performance of LSTM models trained and untrained with co-dynamic analysis. As shown in Tables 5 and 6, including rule-based co-dynamics as input features consistently improve prediction accuracy, providing a 27.74% RMSE improvement for ARCLK and an 11.47% RMSE improvement for TSLA.

RQ3 was examined by evaluating prediction performance under different temporal dynamics. The results show that trend change and instantaneous change dynamics provide the most reliable and frequently occurring prediction signals, while regime change, despite its high reliability, suffers from sparsity-related learning limitations in LSTM models.

3.4 Discussion of advantages and limitations the proposed study

The proposed common temporal dynamics-based forecasting model offers several advantages for financial time series analysis. One of the main strengths of the approach relies on its explicit modelling multiple dynamics. These dynamics are listed as trend change, seasonality, regime switching, volatility change, and instantaneous change. Unlike classic forecasting methods that rely on raw price sequences and statistical changes across multiple assets by encoding its dynamics into covariance matrix.

Another advantage of the proposed framework is the integration of association rule mining in feature extraction. By generating rule sets(antedecents, consequent) characterized with support, confidence and lift metrics, the method identifies cross-asset temporal relationships meaningful and interpretable. As demonstrated in Table 3 and Table 4, the

extracted association rules provide valuable insight into how different inputs interact under temporal dynamics. Unlike classical forecasting models, this rule-based mechanism enables the model to move beyond individual asset behaviour and utilize collective market structures.

Furthermore, the experimental results show that using common temporal features enhances forecasting performance. As reported in Table 5 and Table 6 the proposed model leads to improvement over conventional LSTM model that rely on only historical variables. In particular, the proposed approach achieves up to a 27.74% reduction in RMSE for ARCLK and an 11.47% reduction for TSLA.

Despite these advantages, the proposed study also gives some limitations that should be acknowledged. As observed in Table 3 and Table 4 association rule ratios from regime switching tend to exhibit high confidence and lift metrics. However, their corresponding support values are low compared to other dynamics such as trend change and instantaneous change. This indicates that regime-switching events occur infrequently within the analysed datasets.

Another limitation of the proposed framework is its sensitivity to data characteristics as liquidity and trading activity. For low-liquidity assets with low volume, irregular price movements and low number of investors simulation results may be noisier and less reliable.

Overall, proposed architecture provides a robust, interpretable framework for feature extraction in financial time series analysis; these limitations provide more inspiration for future research in time series analysis. Adaptive window mechanisms, learning strategies that account for sparsity, or extensions incorporating liquidity-sensitive preprocessing can further enhance the robustness and generalization ability of the proposed methodology.

4 Conclusions

In literature, there exist correlation and time-change based feature extraction methods for time series forecasting. However, in this paper, common dynamics are found based on a novel approach with temporal pattern functions then stock prices with similar behaviour are used as features for target stock price forecasting. With the help of feature engineering, LSTM-based prediction model improved RMSE of 27.74% for BIST100 and 11.47% for NASDAQ stock prices. It presents that temporal patterns change at each sliding window, and the common dynamics matrix captures more informative patterns from related stock prices. In future studies, the promising common patterns trend and instantaneous change dynamics can be used construct feature datasets for general time series forecasting. In the case of highly correlated may generate overlapping antecedent sets. However, using lift and confidence metrics helps mitigate non-random associations. For low-liquidity stocks, irregular trading activity and lackness of volume, fundamental conditions may cause decreasing the reliability of the generated association rules. By focusing on liquid and actively traded stocks with a high number of investors the method ensures that the extracted common dynamics are meaningful and representative of underlying market behavior. In ongoing work, for electricity demand forecasting, we are planning to analyse the other time series such as water demand, per capita income, industrial development index, the gross national product etc.

5 Acknowledgement

This work was supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK) under Grant No. 125E284 of the TÜBİTAK-1001 Program.

6 Author contribution statements

Writing-original draft preparation, literature review, and investigation of results were conducted by Ata Eren Arslan. Conceptualization, methodology, and evaluation of results were performed by Selami Beyhan.

7 Ethics committee approval and conflict of interest statement

"There is no need for ethics committee approval for the prepared article". "There is no conflict of interest with any person/institution in the prepared article".

8 References

[1] Teräsvirta T, Tjøstheim D, Granger CWJ. "Modelling Nonlinear Economic Time Series", *1st ed. Oxford, Oxford University Press*, 2010.

[2] Agrawal R, Imielinski T, Swami A. "Mining association rules between sets of items in large databases". *ACM SIGMOD International Conference on Management of Data*, Washington DC, USA, 25-28 May 1993.

[3] Agrawal R, Srikant R. "Mining sequential patterns". *Proceedings of the 11th International Conference on Data Engineering (ICDE)*, Taipei, Taiwan, 6-10 March 1995.

[4] Fournier-Viger P, Wu CW, Tseng VS, Cao L, Nkambou R. "Mining partially-ordered sequential rules common to multiple sequences". *IEEE Transactions on Knowledge and Data Engineering,* 27(8), 2203–2216, 2015.

[5] Çelik C, Aladağ Z, Alkan A. "Veri madenciliği ve stok tutma süresi teknikleri ile depolar için dinamik ürün analizi modelinin geliştirilmesi". *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, 29(2), 172-182, 2023.

[6] Yıldırım P, Birant D. "Bulut bilişimde veri madenciliği tekniklerinin uygulanması: Bir literatür taraması". *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, 24(2), 336-343, 2018.

[7] Hochreiter S, Schmidhuber J. "Long short-term memory". *Neural Computation,* 9(8), 1735–1780, 1997.

[8] Aladağ E. "Forecasting of particulate matter with a hybrid ARIMA model based on wavelet transformation and seasonal adjustment". *Urban Climate,* 39(1), 100930, 2021.

[9] Md AQ, Kapoor S, Chris Junni AV, Sivaraman AK, Tee KF, Sabireen H, Janakiraman N. "Novel optimization approach for stock price forecasting using multi-layered sequential LSTM". *Applied Soft Computing,* 134(1), 1-11, 2023.

[10] Park HJ, Kim Y, Kim HY. "Stock market forecasting using a multi-task approach integrating long short-term memory and the random forest framework". *Applied Soft Computing,* 114(1), 1-17, 2022.

[11] Tang P, Tang C, Wang K. "Stock movement prediction: A multi-input LSTM approach". *Journal of Forecasting,* 43(5), 1199–1211, 2024.

[12] Shabani M, Tran DT, Magris M, Kanniainen J, Iosifidis A. "Multi-head temporal attention-augmented bilinear network for financial time series prediction". *30th European Signal Processing Conference (EUSIPCO),* Belgrade, Serbia, 29 August-2 September 2022.

[13] Praveen M, Dekka S, Sai DM, Chennamsetty DP, Chinta DP. "Financial time series forecasting: A comprehensive review of signal processing and optimization-driven intelligent models". *Computational Economics*, 67(2), 963-989, 2026.
[14] Wang X, Xing Q, Xiao H, Ye M. "Contrastive learning enhanced by graph neural networks for universal multivariate time series representation". *Information Systems,* 125(1), 1-13, 2024.
[15] Barazandeh I, Haratizadeh S, Sermpinis G. "A temporal graph-based contrastive approach for financial time series forecasting". *Engineering Applications of Artificial Intelligence*, 153(1), 110834, 2025.
[16] Papageorgiou G, Tjortjis C. "Adaptive sliding window normalization". *Information Systems,* 129(1), 1-13, 2025.
[17] Liu Q, Tao Z, Tse Y, Wang C. "Stock market prediction with deep learning: The case of China". *Finance Research Letters,* 46(1), 1-7, 2022.
[18] Hou Y, Hu C. "Understanding the role of aggregate analyst attention in resolving stock market uncertainty". *Finance Research Letters,* 57(1), 1-10, 2023.
[19] Sheng Y, Qu Y, Ma D. "Stock price crash prediction based on multimodal data machine learning models". *Finance Research Letters,* 62(1), 1-8, 2024.
[20] Yao D, Yan K. "Time series forecasting of stock market indices based on DLWR-LSTM model". *Finance Research Letters,* 68(1), 1-11, 2024.
[21] Kristjanpoller W, Michell K, Llanos C, Minutolo MC. "Incorporating causal notions to forecasting time series: a case study". *Financial Innovation*, 11(1), 1-22, 2025.
[22] Niu H, Liu T. "Forecasting the volatility of European Union allowance futures with macroeconomic variables using the GJR-GARCH-MIDAS model". *Empirical Economics,* 67(1), 75–96, 2024.
[23] Eroğlu BA, İkizlerli D, Ülkü N. "A mixed-frequency VAR application to studying joint dynamics of foreign investor trading and stock market returns". *Empirical Economics*, 67(1), 47–73, 2024.
[24] Li S, Xu S. "Enhancing stock price prediction using GANs and transformer-based attention mechanisms*". Empirical Economics*, 68(1), 373–403, 2025.
[25] Özen F, Arslan AE, Beyhan S. "Adaptive sliding window size based sequential learning of LSTM, GRU and transformer models applied to financial time series prediction". *Savunma Bilimleri Dergisi,* 22(1), 97-108, 2026.