J. Electrical Systems 20-2s (2024): 575-586

[1] Milind Kolambe
[2] Dr.Sandhya Arora

Forecasting the Future: A Comprehensive Review of Time Series Prediction Techniques

Abstract: - Time series forecasting is a critical aspect of data analysis, with applications ranging from finance and economics to weather prediction and industrial processes. This review paper explores the evolution of time series forecasting techniques, analyzing the progression from classical methods to modern approaches. It synthesizes key advancements, discusses challenges, future directions and provides insights into emerging trends. Traditional forecasting methods often struggle with capturing the complex patterns and dynamics present in real-world time series data. This study explores the efficacy of cutting-edge models, such as long short-term memory (LSTM) networks, and recurrent neural networks (RNNs), in capturing intricate temporal dependencies. It also aims to guide researchers and practitioners in selecting appropriate methods for diverse time series forecasting applications. We categorize existing approaches, discuss their strengths and limitations, and highlight emerging trends in the field.

Keywords: Time series, Forecasting, Machine learning, Hybrid method

I. INTRODUCTION

Time series forecasting is a branch of predictive analytics that involves predicting future values of a variable based on its past observations or measurements. In a time series, data points are collected, recorded, or observed over time, and the goal of forecasting is to make predictions about future values. Time series forecasting finds applications in various fields, including finance [1][2], economics [3][4], weather forecasting [5][6], stock market analysis [7][8][9], energy consumption prediction [10][11], and more.

Key components and concepts in time series forecasting play important role in forecasting. Time Series Data is a series of data points indexed or ordered chronologically. Examples include daily stock prices, monthly sales figures, hourly temperature readings, etc. Trend is a long term changes in the data, like values rising or falling over time. Seasonality is repeating patterns or cycles that occur at regular intervals, often influenced by factors like seasons, holidays, or days of the week. Noise is random fluctuations or irregularities in the data that do not follow a specific pattern.

Various mathematical models and algorithms are used for time series forecasting, including autoregressive integrated moving average (ARIMA) [12], exponential smoothing methods [13][14], and machine learning techniques like Long Short-Term Memory (LSTM) [15][16] networks and recurrent neural networks (RNN) [17]. Time series forecasting helps businesses anticipate future trends and make informed decisions, such as inventory planning [18], resource allocation [19], and marketing strategies. Time series forecasting is a method employed in the financial domain to predict financial metrics, including stock prices and fluctuations in currency exchange rates [20], assisting investors and traders in making investment decisions. Forecasting demand for products helps optimize inventory levels [21], reduce costs, and improve overall supply chain efficiency [22]. Predicting energy consumption [23] patterns aids in optimizing energy production and distribution, leading to cost savings and sustainability. Time series analysis is crucial in meteorology for predicting weather conditions [24], which is vital for agriculture, disaster management, and public safety.

Table 1. Methods applied for various applications domain

Application domain	Number of Methods Used
Business Decision-Making	Regression Analysis, ARIMA, Exponential Smoothing, Machine Learning Models (e.g., LSTM, Random Forest)
Financial Markets	Time Series Analysis, GARCH Models, Autoregressive Models, Neural Networks, Monte Carlo Simulation
Supply Chain Management	Seasonal Decomposition, Holt-Winters Exponential Smoothing, Long Short-Term Memory (LSTM),

[1] Milind Kolambe
[2] Dr.Sandhya Arora
Copyright © JES 2024 on-line : journal.esrgroups.org

Application domain	Number of Methods Used
	Prophet
Energy Management	Autoregressive Integrated Moving Average (ARIMA), Support Vector Machines (SVM), Deep Learning Models, Genetic Algorithms
Weather Forecasting	Numerical Weather Prediction Models, Time Series Regression, Ensemble Methods, Markov Models
Healthcare	Long Short-Term Memory (LSTM), Gaussian Processes, Bayesian Methods, Hidden Markov Models
Traffic Management	Time Series Clustering, Dynamic Time Warping, Recurrent Neural Networks (RNN), Kalman Filters

Forecasting can be applied to predict disease outbreaks [25], patient admission rates, and resource requirements in healthcare systems. Predicting traffic patterns [26] helps optimize transportation systems, reduce congestion, and enhance overall urban planning. Table 1 shows various methods used for forecasting in various application areas.

II. TIME SERIES FORECASTING METHODS

Time series forecasting methods can be broadly categorized into traditional statistical approaches, modern machine learning techniques, deep learning approaches and hybrid methods.

2.1 Classical Methods

Time series forecasting involves predicting future values based on historical data. Traditional methods for time series forecasting include:

Moving Averages:

Simple Moving Average (SMA) [27] calculates the average of a fixed number of recent data points. Exponential Moving Average (EMA) [28] gives more weight to recent observations, allowing the model to adapt to changes faster. Autoregressive Integrated Moving Average (ARIMA) [29] combines auto regression (AR), differencing (I), and moving averages (MA).

Seasonal Decomposition of Time Series (STL):

It decomposes the time series into seasonal, trend, and remainder components. It helps in analysing and forecasting each component separately.

Seasonal-Trend decomposition using LOESS (STL-LOESS):

Similar to STL but uses locally weighted regression (LOESS) for smoother trend and seasonal components [30].

Holt-Winters Exponential Smoothing:

It incorporates trends and seasonality in the data and includes three smoothing parameters (α, β, γ) for level, trend, and seasonality [31].

SARIMA (Seasonal ARIMA):

It is the extension of ARIMA that considers seasonality [32]. It involves additional seasonal parameters similar to ARIMA.

Theta Method:

It’s a simple exponential smoothing method with a parameter called theta (θ) and can be seen as a generalization of the exponential smoothing methods [33]

Prophet:

Developed by Facebook, Prophet is designed for forecasting with daily observations that display patterns on different time scales [34]. It can handle missing data and outliers well.

Box-Jenkins Methodology:

It’s a systematic approach to time series analysis and forecasting developed by George Box and Gwilym Jenkins [35][36]. It involves model identification, parameter estimation, and diagnostic checking.

2.2 Machine Learning Approaches

Time series forecasting methods based on machine learning leverage algorithms to analyse historical data patterns and make predictions about future values.

Regression-based Methods:

Regression-based methods, including linear regression [37], polynomial regression [38], and time series decomposition [39], are widely used for forecasting. These techniques model the connection or association between the characteristics provided as input and the outcome variable, making them simple and interpretable. However, they may struggle to capture non-linear patterns and complex dependencies.

Decision Trees and Ensemble Methods:

Decision trees [40] are versatile models capable of capturing non-linear relationships. Ensemble methods, such as bagging and boosting, enhance the predictive performance by combining multiple decision trees. Random Forests [41][42], an ensemble of decision trees, provide robustness and are less prone to over fitting [43] compared to individual trees.

Support Vector Machines (SVM):

Support Vector Machines [44][45] are effective in time series forecasting, particularly in situations with high dimensionality. SVMs try to locate a hyper plane that best divides data points in feature space. While they may require careful tuning and pre-processing, SVMs can handle complex relationships in time series data.

Random Forests:

Random Forests [46], an ensemble method discussed earlier, deserve special attention due to their effectiveness in capturing complex relationships and providing robust predictions. They excel in handling large datasets and are less sensitive to noise in the data.

Gradient Boosting:

Gradient Boosting algorithms, such as XGBoost [47] and Light GBM [48], have become popular for time series forecasting. These algorithms build a strong predictive model by iteratively combining weak learners. They are known for their high accuracy and ability to handle missing data.

2.3 Deep Learning Approaches

Time series forecasting methods based on deep learning leverage neural networks, such as recurrent neural networks (RNNs) and long short-term memory networks (LSTMs), to model complex temporal dependencies in sequential data. These methods excel in capturing patterns, trends, and seasonality in time series data, offering a powerful approach for predicting future values.

Recurrent Neural Networks (RNN):

RNNs [49] are foundational deep learning models for sequential data. However, they suffer from the vanishing and exploding gradient problems, limiting their effectiveness in capturing long-range dependencies. Despite these limitations, RNNs serve as the building blocks for more advanced architectures.

Long Short Term Memory (LSTM) Networks:

LSTM networks [50][51] address the vanishing gradient problem by introducing memory cells that can store and retrieve information over long sequences. LSTMs have demonstrated superior performance in capturing temporal dependencies and are widely applied in time series forecasting tasks.

Gated Recurrent Units (GRU):

GRUs [52] is a variant of RNNs designed to simplify the architecture while retaining the capability to capture long-range dependencies. They have shown comparable [53] performance to LSTMs in various applications and are computationally more efficient.

Transformer-based Models:

Originally designed for natural language processing tasks, Transformer architectures, such as the Attention [54] is All You Need model, have been adapted for time series forecasting. These models use self-attention mechanisms to capture dependencies across different time steps simultaneously, enabling parallel processing and scalability.

2.4 Hybrid Approaches

Hybrid approaches for time series prediction aim to leverage the strengths of classical methods, machine learning (ML) techniques, and deep learning (DL) approaches to improve the accuracy and robustness of predictions. Here are some common hybrid approaches:

Classical Methods and Machine Learning:

Classical methods like ARIMA or Exponential Smoothing can be used to generate traditional time series features. These features can then be fed into machine learning methods like random forests and decision trees [55], or gradient boosting machines [56]

Combining the forecasts of classical methods with those of ML models through ensemble methods like bagging or boosting can often result in more accurate predictions.

Machine Learning and Deep Learning:

ML models can be employed to extract features and selection before inputting the data into deep learning models. This can help capture important patterns and reduce the dimensionality of the input space [57][58].

Pre-trained ML models can be used as feature extractors for deep learning models. The knowledge gained by the ML model on one task can be transferred to the deep learning model for better performance [59]

Classical Methods, Machine Learning and Deep Learning:

Using classical methods to generate initial predictions and then refining them with machine learning models, followed by deep learning models, can create a cascade of models [60] that progressively refines the predictions at each stage.

It's an approach of creating models that have both classical and deep learning components [61], for example, using a neural network to learn the residuals from an ARIMA model.

Method is to combine predictions from different types of models into a model for the ultimate forecast [62]. This can include stacking the outputs of classical models, machine learning models, and deep learning models.

Idea is to assign different weights [63] to the predictions from classical, ML, and DL models based on their historical performance or confidence levels.

Table 2 summarizes various methods and their characteristics for their applicability.

Table 2. Forecasting methods and their characteristics

Method	Characteristics
Classical Methods	
- Moving Averages	-Simple trends and seasonality
- Exponential Smoothing	Short-term forecasting, minimal noise
- ARIMA	General-purpose forecasting, linear trends, and seasonality
- SARIMA	Time series with strong seasonality
Machine Learning Methods	
- Linear Regression	- Linear trends and seasonality
- Decision Trees	- Non-linear patterns, multiple variables
- Random Forests	- Non-linear patterns, ensemble learning
- Support Vector Machines	- Non-linear patterns, small to medium-sized datasets
Deep Learning Methods	
- RNNs	- Sequential patterns, long-term dependencies
- LSTM	- Improved handling of long-term dependencies
- GRU	- Similar to LSTM, simpler architecture
- Transformer-based Models	- Sequence-to-sequence modelling, attention mechanisms
Hybrid Methods	
- ARIMA-X	- Incorporates external factors
- STL	- Separates time series into trend, seasonality, and remainder
- Ensembling	- Improves accuracy by combining multiple models
- Prophet	- Daily observations with strong seasonal patterns

III. Challenges in Time Series Forecasting

Time series forecasting poses several challenges, including the presence of seasonality and trends, making it difficult to discern underlying patterns. Additionally, handling missing or irregularly spaced data points can complicate model training and prediction accuracy. The dynamic nature of many real-world time series data further adds complexity, as models must adapt to changing patterns over time.

3.1 Data-related Challenges

Data related challenges in time series forecasting often revolve around issues such as missing values, irregular sampling intervals, and the presence of outliers, which can complicate the training of accurate models and hinder the extraction of meaningful patterns from the temporal data.

Noisy Data and Outliers:

Noisy data [64] refers to random fluctuations or errors in the time series that do not contribute to the underlying patterns. Outliers [65], on the other hand, are data points significantly deviating from the general pattern. Noisy data and outliers can distort the learning process of forecasting models, leading to inaccurate predictions.

Several techniques can be used to mitigate these issues, such as data cleaning [66], which involves identifying and removing outliers from the data to lessen the impact of noisy data; smoothing [67], which involves applying moving averages or other smoothing methods to reduce noise; and using robust models [68], which include robust regression or ensemble methods, which are less prone to anomalies.

Missing Values:

Time series data often contain missing values due to various reasons, such as sensor malfunctions, human errors, or system failures. Missing values can disrupt the temporal patterns crucial for forecasting models.

Mitigation measures include imputation approaches, which estimate missing values based on neighbouring data by applying techniques like mean imputation [69], forward-fill [70], or advanced methods like k-nearest-neighbours (KNN) imputation [71], interpolation [72].

Synthetic data points [73] are created in data augmentation to improve model training and make up for missing variables.

Non-Stationary:

Non-stationary refers to a time series where statistical properties, such as mean and variance, change over time. Many time series forecasting models assume stationary, and violating this assumption can lead to inaccurate predictions.

One solution to these problems is to use differencing. A non-stationary time series can be made stationary by using certain methods [74].

By employing de-trending techniques like polynomial fitting [75] or moving averages, trend components are eliminated.

To properly manage non-stationary data, time series are decomposed into trend, seasonal, and residual components using seasonal decomposition [76].

3.2 Model-related Challenges

Model related challenges include the selection of appropriate algorithms for different types of time series, determining optimal model hyper parameters, and addressing the sensitivity of models to changes in the training data, as well as the need for continuous model updating to adapt to evolving patterns in the time series.

Model Complexity and Interpretability:

Time series forecasting models often face the trade-off between model complexity and interpretability. Complex models, such as deep neural networks, may achieve high accuracy but lack interpretability, making it challenging to understand and trust the predictions.

One of the mitigation strategies is to use simpler models. For interpretability, more straightforward models like Exponential Smoothing or Autoregressive Integrated Moving Average (ARIMA) are taken into consideration.

utilizing interpretability techniques, such as SHAP (Shapley Additive explanations) [77] values, to measure the influence of input features on predictions in order to interpret the model.

Using ensemble approaches, predictions from several interpretable models are pooled to improve accuracy without compromising interpretability.

Raw time series data is converted into meaningful features using feature engineering, which improves interpretability and captures pertinent patterns.

Over fitting and Under fitting:

Over fitting and under fitting are common challenges in time series forecasting, where models may perform well on training data but fail to generalize to unseen data.

Cross-validation is used as part of mitigation technique; for example, time series are separated to evaluate model performance over several time periods.

Regularization techniques, such as L1 or L2 regularization [78], work by penalizing large coefficients in the model in order to prevent overfitting.

In order to avoid overfitting, early stopping is an option, which involves tracking the model's performance on a validation set during training and halting the process when performance reaches a plateau [79].

Hyper parameter Tuning:

Selecting optimal hyper parameters is crucial for achieving optimal model performance. Poorly tuned hyper parameters can lead to suboptimal forecasts.

To overcome these challenges grid search and random search cam be applied where systematically hyper parameter space is explored using grid search or random search [80] to find the combination that yields the best performance.

We can utilize Bayesian optimization [81] techniques to efficiently search for optimal hyper parameters, reducing the computational cost.

Automated hyper parameter tuning can be done i.e. to leverage automated hyper parameter tuning tools like Hyper opt to streamline the process and discover optimal hyper parameters.

Using robust validation techniques, such as time series cross-validation, to ensure hyper parameter tuning decisions are based on reliable performance estimates can be done.

3.3 Temporal Challenges

Temporal challenges involve the dynamic nature of time series data, encompassing issues like seasonality, trend shifts, and abrupt changes in the underlying patterns, making it crucial for forecasting models to adapt and capture these temporal variations accurately.

Handling Seasonality and Trends:

Frequent patterns in time series data, such as trends and seasonality, can have a big influence on forecasting accuracy. Seasonal variations occur at regular intervals, while trends represent long-term patterns. To address these challenges, several techniques can be employed:

Mitigating is possible using decomposition where time series is broken down into its trend, seasonality, and residual components can assist in identifying patterns. This facilitates the application of forecasting models on individual components, enhancing accuracy.

In seasonal adjustment [82], differencing or seasonal decomposition of time series (STL) is done which helps in removing seasonality, making it easier for models to capture underlying patterns.

Adaptive models can be applied [83] to automatically adjust to changing patterns in the data helps in capturing evolving seasonality and trends.

Time Series with Irregular Intervals:

Many real-world time series datasets exhibit irregular intervals between observations, posing a challenge for traditional forecasting models designed for equally spaced data points. Strategies to handle irregular intervals include:

Interpolation techniques can be employed to fill in missing values [84] and regularize the time intervals, making the data suitable for traditional forecasting models.

Developing models that can handle events triggering irregular observations, providing a more realistic representation of the underlying process can be done.

Resampling is another way to mitigate challenges such as aggregation or down sampling to convert irregular intervals [85] into regular ones, facilitating the application of conventional time series forecasting techniques.

Dynamic and Evolving Patterns:

Time series data often exhibit dynamic and evolving patterns, making it challenging for static models to capture changing behaviours. Mitigation strategies include:

To overcome these challenges adaptive learning can be implemented. Adaptive learning algorithms can be used that can continuously update model parameters based on new data, allowing the model to adapt to changing patterns over time.

Leveraging ensemble methods, such as ensemble of models or model ensembling with rolling forecasts [86], to combine the strength of multiple models and improve robustness in capturing dynamic patterns.

Employing incremental learning approaches to update models with new data efficiently, enabling the model to evolve and instantly adjust to shifting trends can done.

IV. EVALUATION METRICS

Evaluation metrics can be categorized as accuracy metrics, forecasting performance metrics, and coverage probability.

4.1 Accuracy Metrics:

Accuracy metrics in time series forecasting, such as Mean Absolute Error (MAE) or Root Mean Squared Error (RMSE), quantify the difference between predicted and actual values over a period, providing a quantitative measure of the model's performance in capturing temporal patterns and trends.

Mean Absolute Error (MAE):

MAE is a widely used metric that measures the average absolute difference between the predicted and actual values. It is calculated as the mean of the absolute differences between predicted and actual values for each observation in the time series. MAE is particularly useful for evaluating the amount of predicted mistakes without taking direction into account.

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y_i}| \qquad (1)$$

Where :

n is the number of observations.

y_i is actual value at time i.

$\hat{y}_i$ is forecasted value at time i.

Mean Squared Error (MSE):

MSE computes the average of the squared differences between predicted and actual values. Squaring the errors emphasizes larger errors and penalizes them more than smaller errors. MSE provides a measure of the overall variance of forecast errors.

$$MSE = \frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y_i})^2 \qquad (2)$$

Where :

n is the number of observations.

y_i is actual value at time i.

$\hat{y}_i$ is forecasted value at time i.

Root Mean Squared Error (RMSE):

RMSE is the square root of the MSE and is often used to express errors in the same units as the original time series data. It provides a more interpretable measure of the average forecast error.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y_i})^2} \qquad (3)$$

Where :

n is the number of observations.

y_i is actual value at time i.

$\hat{y}_i$ is forecasted value at time i.

Forecasting Performance Metrics:

Performance metrics quantitatively assess the accuracy of predictive models by measuring the difference between predicted and actual values over a given time period.

Mean Absolute Percentage Error (MAPE):

MAPE is a percentage-based metric that calculates the average absolute percentage difference between predicted and actual values. MAPE is useful for assessing the accuracy of forecasts relative to the scale of the observed values.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n} |\frac{y_i - \hat{y_i}}{y_i}| \times 100 \qquad (4)$$

Where :

n is the number of observations.

y_i is actual value at time i.
$\hat{y}_i$ is forecasted value at time i.

Forecast Bias:

Forecast Bias measures the systematic overestimation or underestimation of forecasts. It is the average of the estimated and actual values, and it gives information about the general trend of forecast mistakes.

$$Bias = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i) \qquad (5)$$

Where :
n is the number of observations.
y_i is actual value at time i.
$\hat{y}_i$ is forecasted value at time i.

Coverage Probability:

Coverage Probability is particularly relevant when dealing with prediction intervals or confidence intervals. It assesses the proportion of observed values falling within a specified interval, providing a measure of the model's ability to capture uncertainty.

V. FUTURE DIRECTIONS

Future directions in time series forecasting may involve the integration of advanced machine learning techniques, such as deep learning and reinforcement learning, to enhance the accuracy and robustness of predictions.

Explainable AI in Time Series Forecasting:

Future directions in time series forecasting include the integration of Explainable AI techniques to enhance the transparency and interpretability of models, allowing users to understand and trust the predictions generated.

Interpretability of Complex Models:

Interpretability becomes more important as machine learning models get more complicated. Time series forecasting models' decision-making process must be understood and communicated through the use of explainable AI (XAI) [87] approaches. Techniques such as SHAP (Shapley Additive explanations) and LIME (Local Interpretable Model-agnostic Explanations) [88] can be employed to interpret black-box models, providing insights into how specific input features contribute to the model's predictions.

Building Trust in Forecasting Models:

Trust in forecasting models is crucial for their adoption in decision-making processes. Transparent and interpretable models help stakeholders understand the reasoning behind predictions, fostering trust. Communicating uncertainty and model limitations transparently is also vital in building trust among users.

5.1 Incorporating Uncertainty:

The evolution of time series forecasting involves a focus on incorporating uncertainty measures, enabling more realistic and probabilistic predictions that account for the inherent unpredictability in complex systems.

Probabilistic Forecasting:

Traditional point-wise predictions may not capture the inherent uncertainty in time series data. Probabilistic forecasting provides a richer understanding of uncertainty by generating probability distributions over future values. Techniques like Gaussian Processes [89] and ensemble methods can be employed to provide probabilistic forecasts, allowing decision-makers to evaluate the possibility of various results.

Bayesian Approaches:

Bayesian methods offer a principled way to incorporate prior knowledge and update predictions as new data becomes available. Bayesian time series models, such as Bayesian Structural Time Series (BSTS) [90], can capture complex patterns and provide uncertainty estimates. These approaches help in making more informed decisions, especially in dynamic and changing environments.

5.2 Handling Big Time Series Data:

Addressing the challenges of handling big time series data is a crucial future direction, necessitating the development of scalable algorithms and efficient processing techniques to analyze massive datasets with increasing volume and complexity.

Scalability and Efficiency:

As the volume of time series data continues to grow, scalability and efficiency become critical challenges.

Implementing scalable algorithms and distributed computing frameworks, such as Apache Spark [91], can enable efficient processing of large datasets, facilitating real-time forecasting and analysis.

Parallel and Distributed Computing:

Leveraging parallel and distributed computing architectures can accelerate model training and prediction tasks. Techniques like data parallelism and model parallelism can be employed to distribute computation across multiple nodes, addressing the computational demands of big time series data.

5.3 Advanced Feature Engineering:

Advancements in time series forecasting will likely involve the exploration of advanced feature engineering methods, leveraging domain knowledge and innovative techniques to extract relevant information and improve the accuracy of predictive models.

Extracting Informative Features from Time Series Data:

A key factor in time series forecasting models' performance is feature engineering. Advanced techniques, such as time-domain and frequency-domain transformations [92], signal processing, and dimensionality reduction methods, can help extract informative features from raw time series data, enhancing the model's ability to capture underlying patterns.

Feature Importance in Forecasting Models:

Understanding the importance of features aids in model interpretation and decision-making is essential. Techniques like permutation importance [93] and SHAP values can be used to measure how different features affect the predictions made by the model. This information can guide feature selection and refinement, leading to more effective forecasting models.

VI. CONCLUSION

This survey paper has delved into the intricate realm of time series forecasting, shedding light on various methodologies and challenges prevalent in the field. The overview of existing methods has provided a comprehensive understanding of the diverse approaches employed to predict future trends. However, the identified challenges underscore the complexity of the task at hand, emphasizing the need for robust and adaptive techniques.

Looking ahead, potential areas for future research have been highlighted, aiming to address the limitations observed in current methodologies. These include the exploration of advanced machine learning algorithms, incorporation of domain-specific knowledge, and the development of ensemble models to enhance predictive accuracy. It is evident that the evolving landscape of time series forecasting demands continuous exploration and adaptation to keep pace with the dynamic nature of data and real-world scenarios.

In closing, the significance of ongoing research and innovation in time series forecasting cannot be overstated. As technological advancements and data availability continue to burgeon, the field must remain agile, embracing novel techniques and refining existing ones. The continuous pursuit of excellence in time series forecasting is paramount for informed decision-making, be it in finance, healthcare, or other domains where accurate predictions are pivotal. By fostering collaboration, sharing insights, and embracing emerging technologies, researchers can contribute to the advancement of this critical field, ensuring its relevance and efficacy in an ever-changing world.

REFERENCES

[1] Yan, H., & Ouyang, H. (2018). Financial time series prediction based on deep learning. *Wireless Personal Communications*, *102*, 683-700.

[2] Van Gestel, T., Suykens, J. A., Baestaens, D. E., Lambrechts, A., Lanckriet, G., Vandaele, B., ... & Vandewalle, J. (2001). Financial time series prediction using least squares support vector machines within the evidence framework. *IEEE Transactions on neural networks*, *12*(4), 809-821.

[3] Bai, J., & Ng, S. (2008). Forecasting economic time series using targeted predictors. *Journal of Econometrics*, *146*(2), 304-317.

[4] Soloviev, V., Saptsin, V., & Chabanenko, D. (2011). Markov chains application to the financial-economic time series prediction. *arXiv preprint arXiv:1111.5254*.

[5] Gneiting, T., & Raftery, A. E. (2005). Weather forecasting with ensemble methods. *Science*, *310*(5746), 248-249.

[6] Abhishek, K., Singh, M. P., Ghosh, S., & Anand, A. (2012). Weather forecasting model using artificial neural network. *Procedia Technology*, *4*, 311-318.

[7] Gandhmal, D. P., & Kumar, K. (2019). Systematic analysis and review of stock market prediction techniques. *Computer Science Review*, *34*, 100190.

[8] Jiang, W. (2021). Applications of deep learning in stock market prediction: recent progress. *Expert Systems with*

Applications, *184*, 115537.
[9] Nabipour, M., Nayyeri, P., Jabani, H., Mosavi, A., & Salwana, E. (2020). Deep learning for stock market prediction. *Entropy*, *22*(8), 840.
[10] Chou, J. S., & Tran, D. S. (2018). Forecasting energy consumption time series using machine learning techniques based on usage patterns of residential householders. *Energy*, *165*, 709-726.
[11] Singh, S., & Yassine, A. (2018). Big data mining of energy time series for behavioral analytics and energy consumption forecasting. *Energies*, *11*(2), 452.
[12] Shumway, R. H., Stoffer, D. S., Shumway, R. H., & Stoffer, D. S. (2017). ARIMA models. *Time series analysis and its applications: with R examples*, 75-163.
[13] Hyndman, R. J., Koehler, A. B., Snyder, R. D., & Grose, S. (2002). A state space framework for automatic forecasting using exponential smoothing methods. *International Journal of forecasting*, *18*(3), 439-454.
[14] Ostertagova, E., & Ostertag, O. (2012). Forecasting using simple exponential smoothing method. *Acta Electrotechnica et Informatica*, *12*(3), 62.
[15] Yu, Y., Si, X., Hu, C., & Zhang, J. (2019). A review of recurrent neural networks: LSTM cells and network architectures. *Neural computation*, *31*(7), 1235-1270.
[16] Smagulova, K., & James, A. P. (2019). A survey on LSTM memristive neural network architectures and applications. *The European Physical Journal Special Topics*, *228*(10), 2313-2324.
[17] Li, S., Li, W., Cook, C., Zhu, C., & Gao, Y. (2018). Independently recurrent neural network (indrnn): Building a longer and deeper rnn. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5457-5466).
[18] Zhou, Q., Han, R., Li, T., & Xia, B. (2019). Joint prediction of time series data in inventory management. *Knowledge and Information Systems*, *61*, 905-929.
[19] Yadav, M. P., Pal, N., & Yadav, D. K. (2021, January). Workload prediction over cloud server using time series data. In *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)* (pp. 267-272). IEEE.
[20] Kamruzzaman, J., & Sarker, R. A. (2003, December). Forecasting of currency exchange rates using ANN: A case study. In *International Conference on Neural Networks and Signal Processing, 2003. Proceedings of the 2003* (Vol. 1, pp. 793-797). IEEE.
[21] Guo, X., Liu, C., Xu, W., Yuan, H., & Wang, M. (2014, July). A prediction-based inventory optimization using data mining models. In *2014 Seventh International Joint Conference on Computational Sciences and Optimization* (pp. 611-615). IEEE.
[22] Aviv, Y. (2003). A time-series framework for supply-chain inventory management. *Operations Research*, *51*(2), 210-227.
[23] Tso, G. K., & Yau, K. K. (2007). Predicting electricity energy consumption: A comparison of regression analysis, decision tree and neural networks. *Energy*, *32*(9), 1761-1768.
[24] Teague, K. A., & Gallicchio, N. (2017). The evolution of meteorology: a look into the past, present, and future of weather forecasting.
[25] Rees, E., Ng, V., Gachon, P., Mawudeku, A., McKenney, D., Pedlar, J. & Knox, J. (2019). Early detection and prediction of infectious disease outbreaks. *CCDR*, *45*(5).
[26] Xiong, H., Vahedian, A., Zhou, X., Li, Y., & Luo, J. (2018, November). Predicting traffic congestion propagation patterns: A propagation graph approach. In *Proceedings of the 11th ACM SIGSPATIAL International Workshop on computational transportation science* (pp. 60-69).
[27] Johnston, F. R., Boyland, J. E., Meadows, M., & Shale, E. (1999). Some properties of a simple moving average when applied to forecasting a time series. *Journal of the Operational Research Society*, *50*(12), 1267-1271.
[28] Klinker, F. (2011). Exponential moving average versus moving exponential average. *Mathematische Semesterberichte*, *58*, 97-107.
[29] Shumway, R. H., Stoffer, D. S., Shumway, R. H., & Stoffer, D. S. (2017). ARIMA models. *Time series analysis and its applications: with R examples*, 75-163.
[30] Chen, D., Zhang, J., & Jiang, S. (2020). Forecasting the short-term metro ridership with seasonal and trend decomposition using loess and LSTM neural networks. *IEEE Access*, *8*, 91181-91187.
[31] Kalekar, P. S. (2004). Time series forecasting using holt-winters exponential smoothing. *Kanwal Rekhi school of information Technology*, *4329008*(13), 1-13.
[32] Vagropoulos, S. I., Chouliaras, G. I., Kardakos, E. G., Simoglou, C. K., & Bakirtzis, A. G. (2016, April). Comparison of SARIMAX, SARIMA, modified SARIMA and ANN-based models for short-term PV generation forecasting. In *2016 IEEE international energy conference (ENERGYCON)* (pp. 1-6). IEEE.
[33] Thomakos, D. D., & Nikolopoulos, K. (2015). Forecasting multivariate time series with the theta method. *Journal of Forecasting*, *34*(3), 220-229.
[34] Toharudin, T., Pontoh, R. S., Caraka, R. E., Zahroh, S., Lee, Y., & Chen, R. C. (2023). Employing long short-term memory and Facebook prophet model in air temperature forecasting. *Communications in Statistics-Simulation and Computation*, *52*(2), 279-290.
[35] Makridakis, S., & Hibon, M. (1997). ARMA models and the Box–Jenkins methodology. *Journal of forecasting*, *16*(3),

147-163.

[36] Dritsakis, N., & Klazoglou, P. (2018). Forecasting unemployment rates in USA using Box-Jenkins methodology. *International Journal of Economics and Financial Issues*, *8*(1), 9.

[37] Williams, C. K. (1998). Prediction with Gaussian processes: From linear regression to linear prediction and beyond. In *Learning in graphical models* (pp. 599-621). Dordrecht: Springer Netherlands.

[38] Ekum, M., & Ogunsanya, A. (2020). Application of hierarchical polynomial regression models to predict transmission of COVID-19 at global level. *Int J Clin Biostat Biom*, *6*(1), 27.

[39] Sen, J., & Chaudhuri, T. D. (2016). Decomposition of time series data of stock markets and its implications for prediction: an application for the Indian auto sector. *arXiv preprint arXiv:1601.02407*.

[40] Spiliotis, E. (2022). Decision trees for time-series forecasting. *Foresight*, *1*, 30-44.

[41] Tyralis, H., & Papacharalampous, G. (2017). Variable selection in time series forecasting using random forests. *Algorithms*, *10*(4), 114.

[42] Lin, L., Wang, F., Xie, X., & Zhong, S. (2017). Random forests-based extreme learning machine ensemble for multi-regime time series prediction. *Expert Systems with Applications*, *83*, 164-176.

[43] Qiu, X., Zhang, L., Suganthan, P. N., & Amaratunga, G. A. (2017). Oblique random forest ensemble via least square estimation for time series forecasting. *Information Sciences*, *420*, 249-262.

[44] Müller, K. R., Smola, A. J., Rätsch, G., Schölkopf, B., Kohlmorgen, J., & Vapnik, V. (1997, October). Predicting time series with support vector machines. In *International conference on artificial neural networks* (pp. 999-1004). Berlin, Heidelberg: Springer Berlin Heidelberg.

[45] Meyer, D., & Wien, F. T. (2001). Support vector machines. *R News*, *1*(3), 23-26.

[46] Tyralis, H., & Papacharalampous, G. (2017). Variable selection in time series forecasting using random forests. *Algorithms*, *10*(4), 114.

[47] Wang, Y., & Guo, Y. (2020). Forecasting method of stock market volatility in time series data based on mixed model of ARIMA and XGBoost. *China Communications*, *17*(3), 205-221.

[48] Lv, J., Wang, C., Gao, W., & Zhao, Q. (2021). An economic forecasting method based on the LightGBM-optimized LSTM and time-series model. *Computational Intelligence and Neuroscience*, *2021*, 1-10.

[49] Zhang, J., & Man, K. F. (1998, October). Time series prediction using RNN in multi-dimension embedding phase space. In *SMC'98 Conference Proceedings. 1998 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No. 98CH36218)* (Vol. 2, pp. 1868-1873). IEEE.

[50] Siami-Namini, S., Tavakoli, N., & Namin, A. S. (2019, December). The performance of LSTM and BiLSTM in forecasting time series. In *2019 IEEE International conference on big data (Big Data)* (pp. 3285-3292). IEEE.

[51] Karim, F., Majumdar, S., Darabi, H., & Chen, S. (2017). LSTM fully convolutional networks for time series classification. *IEEE access*, *6*, 1662-1669.

[52] Zhang, X., Shen, F., Zhao, J., & Yang, G. (2017). Time series forecasting using GRU neural network with multi-lag after decomposition. In *Neural Information Processing: 24th International Conference, ICONIP 2017, Guangzhou, China, November 14–18, 2017, Proceedings, Part V 24* (pp. 523-532). Springer International Publishing.

[53] Yamak, P. T., Yujian, L., & Gadosey, P. K. (2019, December). A comparison between arima, lstm, and gru for time series forecasting. In *Proceedings of the 2019 2nd international conference on algorithms, computing and artificial intelligence* (pp. 49-55).

[54] Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., & Zhang, W. (2021, May). Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 35, No. 12, pp. 11106-11115).

[55] Kumar, M., & Thenmozhi, M. (2014). Forecasting stock index returns using ARIMA-SVM, ARIMA-ANN, and ARIMA-random forest hybrid models. *International Journal of Banking, Accounting and Finance*, *5*(3), 284-308.

[56] Lu, W., Rui, Y., Yi, Z., Ran, B., & Gu, Y. (2020). A hybrid model for lane-level traffic flow forecasting based on complete ensemble empirical mode decomposition and extreme gradient boosting. *IEEE Access*, *8*, 42042-42054.

[57] Zhang, G. P. (2003). Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing*, *50*, 159-175.

[58] Panigrahi, S., & Behera, H. S. (2017). A hybrid ETS–ANN model for time series forecasting. *Engineering applications of artificial intelligence*, *66*, 49-59.

[59] Fawaz, H. I., Forestier, G., Weber, J., Idoumghar, L., & Muller, P. A. (2018, December). Transfer learning for time series classification. In *2018 IEEE international conference on big data (Big Data)* (pp. 1367-1376). IEEE.

[60] Muzy, J. F., Delour, J., & Bacry, E. (2000). Modelling fluctuations of financial time series: from cascade process to stochastic volatility model. *The European Physical Journal B-Condensed Matter and Complex Systems*, *17*, 537-548.

[61] Hu, S., Corry, E., Curry, E., Turner, W. J., & O'Donnell, J. (2016). Building performance optimisation: A hybrid architecture for the integration of contextual information and time-series data. *Automation in Construction*, *70*, 51-61.

[62] Allende, H., & Valle, C. (2017). Ensemble methods for time series forecasting. *Claudio Moraga: A passion for multi-valued logic and soft computing*, 217-232.

[63] Wu, T. J., & Sepulveda, A. (1998). The weighted average information criterion for order selection in time series and regression models. *Statistics & probability letters*, *39*(1), 1-10.

[64] Tresp, V., & Hofmann, R. (1998). Nonlinear time-series prediction with missing and noisy data. *Neural computation*, *10*(3), 731-747.

[65] Blázquez-García, A., Conde, A., Mori, U., & Lozano, J. A. (2021). A review on outlier/anomaly detection in time series data. *ACM Computing Surveys (CSUR)*, *54*(3), 1-33.

[66] Liu, H., Shah, S., & Jiang, W. (2004). On-line outlier detection and data cleaning. *Computers & chemical engineering*, *28*(9), 1635-1647.

[67] Han, M., Liu, Y., Xi, J., & Guo, W. (2006). Noise smoothing for nonlinear time series using wavelet soft threshold. *IEEE signal processing letters*, *14*(1), 62-65.

[68] Stockinger, N., & Dutter, R. (1987). Robust time series analysis: A survey. *Kybernetika*, *23*(7), 1-3.

[69] Moritz, S., & Bartz-Beielstein, T. (2017). imputeTS: time series missing value imputation in R. *R J.*, *9*(1), 207.

[70] Lipton, Z. C., Kale, D. C., & Wetzel, R. (2016). Modeling missing data in clinical time series with rnns. *Machine Learning for Healthcare*, *56*(56), 253-270.

[71] Sridevi, S., Rajaram, S., Parthiban, C., SibiArasan, S., & Swadhikar, C. (2011, June). Imputation for the analysis of missing values and prediction of time series data. In *2011 international conference on recent trends in information Technology (ICRTIT)* (pp. 1158-1163). IEEE.

[72] Ding, Z., Mei, G., Cuomo, S., Li, Y., & Xu, N. (2020). Comparison of estimating missing values in iot time series data using different interpolation algorithms. *International Journal of Parallel Programming*, *48*, 534-548.

[73] Ramponi, G., Protopapas, P., Brambilla, M., & Janssen, R. (2018). T-cgan: Conditional generative adversarial network for data augmentation in noisy time series with irregular sampling. *arXiv preprint arXiv:1811.08295*.

[74] Salles, R., Belloze, K., Porto, F., Gonzalez, P. H., & Ogasawara, E. (2019). Nonstationary time series transformation methods: An experimental review. *Knowledge-Based Systems*, *164*, 274-291.

[75] Shao, Y. H., Gu, G. F., Jiang, Z. Q., & Zhou, W. X. (2015). Effects of polynomial trends on detrending moving average analysis. *Fractals*, *23*(03), 1550034.

[76] Bandara, K., Hyndman, R. J., & Bergmeir, C. (2021). MSTL: A seasonal-trend decomposition algorithm for time series with multiple seasonal patterns. *arXiv preprint arXiv:2107.13462*.

[77] Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, *30*.

[78] Wu, D., Yan, H., & Yuan, S. (2018). L1 regularization for detecting offsets and trend change points in GNSS time series. *GPS Solutions*, *22*, 1-5.

[79] Caruana, R., Lawrence, S., & Giles, C. (2000). Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping. *Advances in neural information processing systems*, *13*.

[80] Zhao, Y., Zhang, W., & Liu, X. (2024). Grid search with a weighted error function: Hyper-parameter optimization for financial time series forecasting. *Applied Soft Computing*, 111362.

[81] Du, L., Gao, R., Suganthan, P. N., & Wang, D. Z. (2022). Bayesian optimization based dynamic ensemble for time series forecasting. *Information Sciences*, *591*, 155-175.

[82] Gómez, V., & Maravall, A. (2001). Seasonal adjustment and signal extraction in economic time series. *A course in time series analysis*, 202-247.

[83] Tran, N., & Reed, D. A. (2001, June). ARIMA time series modeling and forecasting for adaptive I/O prefetching. In *Proceedings of the 15th international conference on Supercomputing* (pp. 473-485).

[84] Terry, W. R., Lee, J. B., & Kumar, A. (1986). Time series analysis in acid rain modeling: Evaluation of filling missing values by linear interpolation. *Atmospheric Environment (1967)*, *20*(10), 1941-1943.

[85] Broersen, P. M., & Bos, R. (2006). Estimating time-series models from irregularly spaced data. *IEEE transactions on instrumentation and measurement*, *55*(4), 1124-1131.

[86] Yuan, S., Luo, X., Mu, B., Li, J., & Dai, G. (2019). Prediction of North Atlantic Oscillation index with convolutional LSTM based on ensemble empirical mode decomposition. *Atmosphere*, *10*(5), 252.

[87] Vielhaben, J., Lapuschkin, S., Montavon, G., & Samek, W. (2024). Explainable ai for time series via virtual inspection layers. *Pattern Recognition*, 110309.

[88] Pedersen, T. L., & Benesty, M. (2018). lime: Local interpretable model-agnostic explanations. *R package version 0.4*, *1*.

[89] Van der Meer, D. W., Shepero, M., Svensson, A., Widén, J., & Munkhammar, J. (2018). Probabilistic forecasting of electricity consumption, photovoltaic power generation and net demand of an individual building using Gaussian Processes. *Applied energy*, *213*, 195-207.

[90] Qiu, J., Jammalamadaka, S. R., & Ning, N. (2018). Multivariate Bayesian Structural Time Series Model. *J. Mach. Learn. Res.*, *19*(1), 2744-2776.

[91] Krome, C., & Sander, V. (2018). Time series analysis with apache spark and its applications to energy informatics. *Energy Informatics*, *1*(1), 337-341.

[92] Keil, A., Bernat, E. M., Cohen, M. X., Ding, M., Fabiani, M., Gratton, G., ... & Weisz, N. (2022). Recommendations and publication guidelines for studies using frequency domain and time-frequency domain analyses of neural time series. *Psychophysiology*, *59*(5), e14052.

[93] Huang, N., Lu, G., & Xu, D. (2016). A permutation importance-based feature selection method for short-term electricity load forecasting using random forest. *Energies*, *9*(10), 767.