Journal of Computer Technology and Software
ISSN:2998-2383
Vol. 3, No. 3, 2024

Deep Learning Methods for Time Series Forecasting: A Comparative Review

Michael Johnson

College of Computer Science and Engineering, University of Connecticut, USA

Correspondence should be addressed to Michael Johnson; michaejh18@gmail.com

Abstract: Time series forecasting involves analyzing the trends and patterns of variables over time to predict future values, which is crucial for decision-making in diverse fields such as finance, meteorology, agriculture, industry, and medicine. With the rapid advancement of sensor and network technologies, vast amounts of time series data have been generated, emphasizing the need for accurate forecasting methods. This paper provides a comprehensive review of traditional and machine learning approaches for time series prediction, with a focus on deep learning methods. The review highlights the strengths and limitations of these deep learning models, providing insights into their practical applications. Additionally, we discuss the future development of deep learning methods in this domain, suggesting directions for enhancing prediction accuracy and model efficiency.

Keywords: Enter Time series forecasting; Deep learning; Convolutional neural nets; Recurrent neural nets; Transformer nets.

1. Introduction

Time series are sets of random variables generated in time, usually at relatively fixed sampling frequencies. Time series forecasting is the study of the trend and pattern of a random variable or variables over time for a future time or time interval based on available time series data.

Time series data has a wide range of applications in the fields of finance, meteorology, agriculture, industry and medicine[1-3]. In recent years, a large amount of time series data has been generated and accumulated with the rapid development of sensor and network technologies. For the time series data field, it usually involves several research problems such as classification, anomaly detection and prediction, among which time series prediction is the research focus. The core of the study is the extraction of patterns from past time series data and their use in the estimation of future trends. Since time series forecasting research can analyses the patterns contained in time series to predict future trends and provide guidance for the decision-making process in various industries, it is of high academic significance and application value.

1.1. Definition of the problem

The time series forecasting problem wants to predict the future target value $\bar{y}_{t+1}$ from the currently available information, this means that the historical data prior to the time t is used to predict the data after the time t. The mathematical expression is shown in equation:

$$\hat{y}_{t+1} = f(y_{t-k:t}, x_{t-k:t}, s) \quad (1)$$

Where $y_{t-k:t}$ denotes the observed values of the time series from time t-k to time t; $x_{t-k:t}$ denotes the covariate inputs from time t-k to time t; S denotes the static factors that remain constant in the observing process, such as the model and the category of the observed object; Note that y and x in the equation can be univariate or multidimensional vectors.

Time series forecasting tasks can be divided into four categories based on the length of the time span being predicted, as shown in Figure 1:

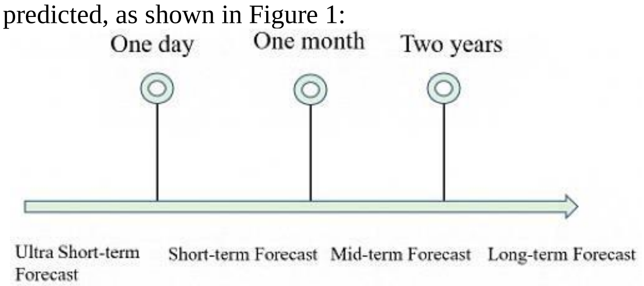

Figure 1: Predicted task time span

This paper begins by discussing the problem of time series forecasting, followed by showing the characteristics and evaluation criteria of time series data, comparing the research methods of time series forecasting, analyzing and discussing the research methods based on deep learning according to three main types, and finally giving further outlook on the development of time series forecasting technology.

2. Characterization of time series data

The core of the time series forecasting problem is the extraction of potential patterns from the data and the estimation of the data values for a specified future time period by learning and analyzing the historical data for the previous t-1 moments. Time series data often exhibit one or more characteristics due to the inherent potential linkages between their variables. This section will introduce these common characteristics in detail.

The main difference between time series data and other types of data, apart from the sheer volume, is that the past data imply the law of the present or future development of the data, namely trend, periodicity, volatility, smoothness and symmetry.

Massive: with the upgrade of IoT sensor devices, the increase of measurement frequency, and the increase of measurement dimensionality, time series data is exploding, and high-dimensional time series data occupy the mainstream [4]. Effective preprocessing of the data set is key to achieving high quality in predicting time series.

Trend: The data at the current time are closely related to the data at the previous time. This feature implies that the time series data are influenced by other factors with a certain pattern of change. They may show a trend of smooth rising, smooth falling or flat over a long period of time.

Periodicity: Data in a time series are affected by external factors and show alternating ups and downs over a long period of time [5].

Volatility: The variance and mean of the time series can also change systematically over time and under the influence of various external factors, which to some extent affects the accuracy of the forecast.

Smoothness: The smoothness of a time series indicates that its mean and variance do not vary systematically over time.

Symmetry: If the distance between the original time series and its inverse time series is controlled within a certain threshold over a certain period of time and the curves are basically in line, the time series is considered to be symmetric [6].

3. Evaluation metrics

Error evaluation index is an important measure of time series forecasting performance. Generally, the larger the error, the lower the prediction accuracy, indicating the poorer performance of the established forecasting method. The evaluation metrics of time series forecasting algorithms commonly in use today are as follows:

Mean Absolute Error(MAE)[7]: The MAE is derived by calculation of the absolute value of the difference between the predicted and true values for each sample, as follows:

$$MAE(y,\hat{y})=\frac{1}{m}\sum_{i=1}^{m}(|y_i-\hat{y}_i|) \tag{2}$$

The formula m is the number of samples, y_i is the true value, $\hat{y}$ is the prediction value of the model, same below.

Mean Square Error(MSE)[8]: MSE is a practical metric derived by calculating the square of the difference between predicted and true for each sample and then averaging, as follows:

$$MSE(y,\hat{y})=\frac{1}{m}\sum_{i=1}^{m}(y_i-\hat{y}_i)^2 \tag{3}$$

Root Mean Square Error(RMSE)[9]: The RMSE is the MSE to open the square to obtain the final calculation results are vulnerable to the extreme values in the dataset, the specific formula is:

$$RMSE(y,\hat{y})=\sqrt{\frac{1}{m}\sum_{i=1}^{m}(y_i-\hat{y}_i)^2} \tag{4}$$

Mean Absolute Percentage Error(MAPE)[10]: MAPE is a relative error metric that prevents positive and negative errors from cancelling each other out. The formula is as follows:

$$MAPE=\frac{100\%}{m}\sum_{i=1}^{m}\left|\frac{\hat{y}_i-y_i}{y_i}\right| \tag{5}$$

R-squared[11]: The result of the calculation is the accuracy of predicting the model, the closer the R^2 value is to 1, the better the model performs. The formula is:

$$R^2(y,\hat{y})=1-\frac{\frac{1}{m}\sum_{i=1}^{m}(y_i-\hat{y}_i)^2}{\frac{1}{m}\sum_{i=1}^{m}(y_i-\bar{y})^2}=1-\frac{MSE}{Var} \tag{6}$$

4. Traditional time series forecasting

In 1927, the autoregressive method(AR) was proposed by the British statistician Yule in his study of sunspots. It laid the foundation for the discipline of time series forecasting. In 1931, the mathematician Walker successively introduced the sliding average method (MA) and the autoregressive sliding average method (ARMA), inspired by the AR method. For time series data with nonstationary processes, in the early 1960s, scientists proposed many methods such as variable difference and exponentially weighted sliding averages, which gradually formed the more mature ARIMA.

AR is an evolution of linear regression in regression analysis for linear forecasting of time series and. The expressions are:

$$Y_t=c+\sum_{i=1}^{p}\varphi_i Y_{t-i}+\varepsilon_t \tag{7}$$

Equation (6) is written as AR(p). where: Y_t is the predicted value; c is a constant term; ε_t is the random error value for a random variable with mean 0 and standard deviation φ ; φ is constant for each t.

MA is a common method in modern spectral estimation and is often used in parametric spectral analysis. The expressions are:

$$Y_t=\delta_t+\sum_{t=1}^{p}\sigma_t\delta_{t-l}-p \tag{8}$$

Equation (8) is written as MA(P). Where: Y_t is predicted value; δ_t is error value; σ_t is weighting factor; p is order.

ARMA is combined AR and MA. It has a more accurate spectral estimation and excellent spectral resolution performance, and is commonly used to study the forecast of sales volume and market size with seasonal variation characteristics. The expressions are:

$$Y_t=\sum_{j=1}^{p}a_jY_{t-j}+\sum_{j=0}^{q}b_j\varepsilon_{t-j} \tag{9}$$

Equation (9) is written as ARMA(p, q).where: p and q are the autoregressive sliding mean orders, respectively; a_j is the autoregressive coefficient; b_j is the sliding mean coefficient; Y_t is the forecast value; Y_{t-j} is the time series value; and ε_{t-j} is the error value.

ARIMA transforms nonstationary time series into stationary time series by differencing and then regressing dependent variable only on lagged values and current and lagged values of random error terms, ARIMA (p, q, d). This approach has been used successfully in many improvements, e.g. SARIMA and SARIMAX [12].

5. Machine learning based methods for predicting time series

There is a close relationship between the nature of forecasting time series data and regression analysis in machine learning methods, so machine learning is also widely used in forecasting time series. The most common are grouped into the following categories.

Support Vector Machines(SVM) can effectively overcome the dimensional catastrophe problem without any increase in computational complexity in the mapping to higher dimensional spaces by means of kernel function methods. Using SVM for time series prediction regression is called SVR, which has stable prediction for nonlinear time series [13].

Gradient Boosting Regression Tree(GBRT) uses gradient descent as a method to solve regression problems. GBRT calculates and iterates using its negative gradient to minimize the loss function and finally obtain the best predicted result [14].

HMM is one of the simplest dynamic Bayesian nets. Das et al [15] proposed a multivariate time series prediction method: semantic Bayesian network (semBnet). The primary goal of the project is to improve the accuracy of weather forecasts by integrating spatial semantics as a form of domain knowledge into a standard Bayesian net.

Machine learning methods are unable to capture the temporal correlation and complex dynamics of data, making it difficult to accurately identify and predict time-series data with long-term dependencies. Therefore, many studies have turned to multilayer neural networks.

6. Deep learning based methods for predicting time series

In recent years, deep learning has attracted a lot of attention from researchers in various fields, and deep learning methods have been widely used in time series prediction because of the superior performance in comparison with traditional methods and machine learning methods. In this section, we introduce three major classes of deep learning methods that can be used to solve time series prediction problems.

6.1. Convolutional Neural Networks(CNN)

CNNs are a class of deep feedforward neural networks based on convolution and pooling operations. They were originally developed to solve image recognition problems in computer vision. CNN retains key information through pooling, uses convolutional kernel's ability to sense changes in historical data over time, and makes predictions based on changes in historical data. Its network structure generally has five layers, as shown in Figure 2:

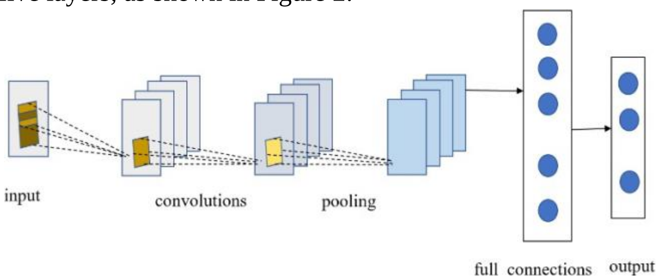

Figure 2: CNN structure schematic

Anastasia Borovykh et al [16] proposed the WaveNet-CNN method inspired by WaveNet, a speech sequence generation model. A ReLU activation function is used and the structure is simplified by using parametric skip connections. Shaojie Bai et al [17] proposed a less memory-consuming and parallelizable temporal convolutional network (TCN) based on CNN. TCN introduce causal convolution to ensure that future information is not acquired in advance during training. They also introduce residual connections to transfer information across layers to solve the information loss problem. By introducing residual connections to pass information across layers, the information loss problem can be solved. The SCINet proposed by Minhao Liu et al [18] uses a hierarchical convolutional network structure to learn effective information with improved predictability by iteratively extracting and aggregating features at different temporal resolutions.

6.2. Recurrent Neural Networks(RNN)

RNNs have achieved good results on various tasks in Natural Language Processing (NLP). RNN can learn the hidden states within all time series before prediction, as a feature representation of past information, and combine with the current input to give the next prediction. The specific structure is shown in Figure 3:

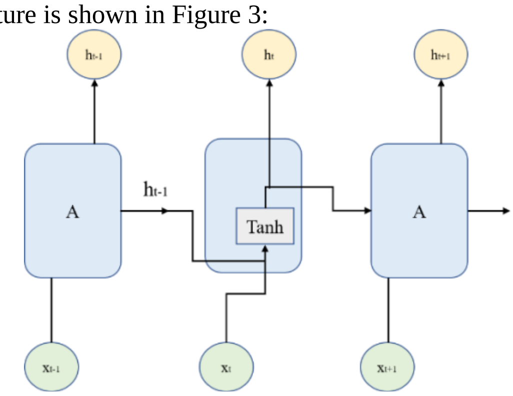

Figure 3: RNN structure schematic

To solve the RNN gradient disappearance or gradient explosion problem, a Long Short-Term Memory (LSTM) is used.The LSTM uses an input gate to decide when to read data into the cell, and a forgetting gate to manage the contents of the reset cell, and weights obtained through training to decide when to remember or ignore inputs in the hidden state. The specific structure is shown in Figure 4:

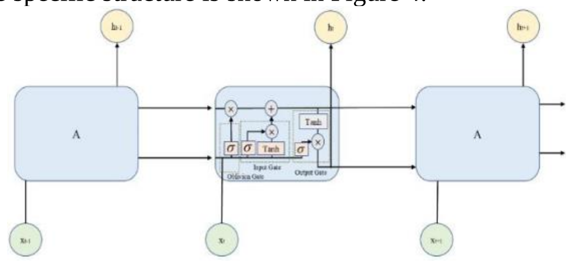

Figure 4: LSTM structure schematic

Gated Recurrent Unit (GRU) is proposed by improving the LSTM, and GRU simplifies the structure. The proportion of information transferred from the previous state to the candidate state is determined by the reset gate. The update gate is a combination of the functions of the forgetting gate and the output gate of the LSTM. LSTM and GRU are important components of RNN-based predicting time series. The specific structure is shown in Figure 5:

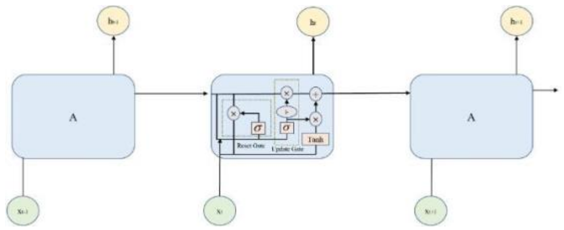

Figure 5: GRU structure schematic

Mike Schuster et al [19] extended RNNs to Bidirectional Recurrent Neural Networks (Bi-RNN), which can obtain both past and future feature information by training in both forward and backward directions.Bi-RNN achieved better prediction results. A Graves et al [20] proposed that Bidirectional Long Short-Term Memory (Bi-LSTM) consists of two independent LSTMs stitched together. Bi-LSTM makes the feature data obtained at time t have both past and future information, and its feature extraction capability is significantly higher than that of LSTM. Qi Y et al [21] proposed a GRU-based approach to explicitly model the competitive relationship between a target product and its alternatives, and applied it to the widespread task of promotional sales forecasting in e-commerce.Another work proposed by Xin S et al [22] fused heterogeneous information into a modified GRU entity to understand the state of the pre-sale phase prior to a promotional campaign.

The problem of gradient disappearance and gradient explosion in RNN has never been completely solved. In the last two years, Transformer has been proposed to solve the above problems well.

6.3. Transformer

Transformer is a self-attention mechanism proposed in Attention is All You Need [23], a paper that experimented with machine translation scenarios and achieved the best results at the time. Due to the similarity between time series prediction and natural language processing, Transformer's approach was soon applied to time series prediction tasks.

Transformer uses a self-attention mechanism to solve the problem that neural networks cannot adequately capture the potential connections between inputs during actual training, resulting in poor model training results. The self-attention module receives n sets of inputs and returns n sets of outputs. All inputs in it interact with each other, and the attention points with obvious roles are mined. The aggregation and attention scores of these interactions are the outputs given by the module. The input (query, key) of the self-attention mechanism, calculated as:

$$A(Q,K,V)=Softmax(\frac{QK^{\cdot}}{\sqrt{d}})V \tag{10}$$

where $Q\in R^{L_Q\times d}$, $K\in R^{L_K\times d}$, $V\in R^{L_V\times d}$, d denote the input dimensions, and the probability formula for the attention factor of the ith request is:

$$A(q_i,K,V)=\sum_j \frac{k(q_i,k_i)}{\sum_l k(q_i,k_l)} v_j = E_{p(k_i/q_i)}[V_j] \tag{11}$$

where, $p(k_i,q_l)=\frac{k(q_i,k_i)}{\sum_l k(q_i,k_l)}$, $k(q_i,k_i)$ choose the asymmetric index $exp(\frac{q_i k_j^{T}}{\sqrt{d}})$.

Transformer is fully dependent on the attention mechanism to represent the global dependencies between method inputs and outputs, as shown in Figure 6:

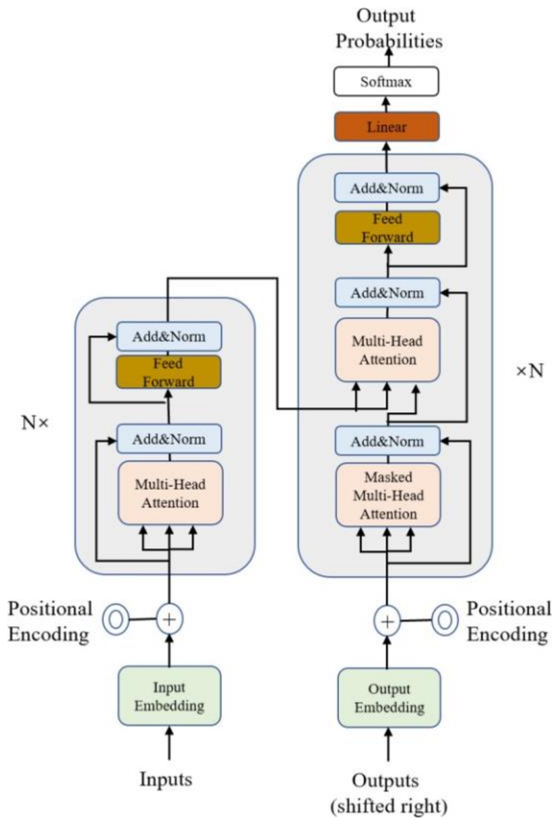

Figure 6: Transformer structure schematic

Sifan Wu et al [24] proposed Adversarial Sparse Transformer (AST) based on Sparse Transformer.The AST model can better represent time series by adversarial training and encoder-decoder structure using discriminators to improve the sequence level prediction performance.

Long time series prediction is particularly important in practical applications such as long-term energy consumption planning, Haoyi Zhou et al [25] proposed Informer based on the classical Transformer encoder-decoder structure, and the paper proposed the ProbSparse self-attention mechanism to efficiently handle long time series input, which can greatly reduce the number of layers of the network through distillation operation, improve the robustness of the layer stacking part, and finally output the desired long time series results immediately.

The TFT proposed by Lim B et al [26] designed a multiscale prediction model that includes a static covariate encoder, a gated feature selection module, and a temporally self-attention decoder.The TFT selects useful information from various covariate information to perform the prediction and retains the interpretability to include global, time-dependent, and event-driven information.

Lin Y et al [27] proposed SSDNet, which combines Transformer and (SSM) to combine the performance advantages of deep learning with the interpretability of SSM.

The Autoformer proposed by Wu H et al [28] developed a simple seasonal trend decomposition method. The unique internal operator used in the paper is able to separate the overall trend of variables from the predicted hidden variables, and this design allows the model to alternatively decompose and refine the intermediate results during the forecasting process.

Qi X et al [29] proposed Aliformer based on bidirectional transformer to solve the problem of accurate time series sales prediction in e-commerce. Aliformer designs a knowledge-driven self-attention layer that uses the consistency of known knowledge to know the transmission of time series information, and proposes a future emphasis training strategy to make the method more focused on the future use of knowledge.

The FEDformer proposed by Tian Zhou et al [30] was designed with two attention modules that deal with the application of attention operations in the frequency domain using Fourier transform and wavelet transform, respectively.The FEDformer incorporates the seasonal trend decomposition method [31], which is widely used in time series analysis, into the transformer-based approach. Since the proposal of FEDformer, the unique property of applying the frequency domain to time series data has attracted much attention in the field of time series forecasting.

Yan Li et al [32] proposed a Conformer for multivariate long-period time series prediction to solve the efficiency and stability problems of long series prediction tasks with obvious periodicity.The network uses fast Fourier transform to do processing on multivariate time as a way to extract the correlation features of multivariate variables and complete the extraction of regularity at different frequencies.

Transformer-like methods are now widely used in various artificial intelligence tasks. Building models based on Transformer can overcome the capacity bottleneck of previous methods, and can effectively solve the long-run forecasting problem by capturing both short-run and long-run dependencies while processing in parallel.

7. Prospecting

This section lists the key issues and further research directions in the field of time series forecasting to facilitate the research and improvement of time series forecasting algorithms.

Choosing the hyperparameters determines whether the model can break out of the local optimum trap to reach the global optimum. To optimise multiple hyperparameters of the deep learning method, a stochastic nature-inspired optimisation algorithm is used. The optimisation algorithm randomly generates a certain number of interpretable steps based on the problem constraints, and then uses the algorithm steps to repeatedly find the global optimum solution and optimal hyperparameters within the bounds to improve model prediction ability. The search for optimal hyper-parameters will be one of the research hotspots in the future.

Transformer shows superior performance on data sets with good periodicity, but does not perform well on data sets with small data size and irregular time intervals. Therefore, the problem of overfitting on small data sets will be a new direction of research in the future.

Introduction of Graph Neural Network (GNN) for Multivariate Time Series Prediction Modelling Recently, many scholars have used temporal polynomial graph neural networks to represent dynamic variable correlations as dynamic matrix polynomials, which have reached an advanced level in both short-term and long-term multivariate time series forecasting. Therefore, the powerful modelling capability of GNN for multivariate time series forecasting deserves further investigation.

8. Conclusion

In this paper, after briefly introducing traditional and machine learning methods for time series prediction, time series prediction methods based on deep learning are systematically summarised using time series data characteristics and evaluation criteria as a guide. The advantages, disadvantages and applications of CNN, RNN and Transformers in predicting time series are reviewed. Finally, the future development of deep learning methods in time series forecasting will be discussed.

References

[1] CI B C, ZHANG P Y. Financial time series forecast based on ARIMA-LSTM model[J]. Statistics & Decision, 2022, 38 (11): 145-149.

[2] CHE C C, WANG H W, NI X M, et al. Residual life prediction of aeroengine based on 1D-CNN and Bi-LSTM[J]. Journal of Mechanical Engineering, 2021, 57 (14): 304-312.

[3] LUO X M, LI J B, HU P. E-commerce inventory optimization strategy based on time series forecasting[J]. Systems Engineering，2014, 32 (6): 91-98.

[4] SONG Y Q, ZHOU G L, ZHU Y L. Present Status and Challenges of Big Data Processing in Smart Grid[J]. Power System Technology, 2013, 37 (04): 927-935.

[5] Wen Q, He K, Sun L, et al. RobustPeriod: Robust time-frequency mining for multiple periodicity detection[C]. Proceedings of the 2021 International Conference on Management of Data, 2021: 2328-2337.

[6] LI P P, SONG S X, WANG J M. Time Series Symmetric Pattern Mining[J]. Journal of Software, 2022, 33 (3): 968−984.

[7] Willmott C J, Matsuura K. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance[J]. Climate research, 2005, 30 (1): 79-82.

[8] Wang Z, Bovik A C. Mean squared error: Love it or leave it ? A new look at signal fidelity measures[J]. IEEE signal processing magazine, 2009, 26 (1): 98-117.

[9] Chai T, Draxler R R. Root mean square error (RMSE) or mean absolute error (MAE) -Arguments against avoiding RMSE in the literature[J]. Geoscientific model development, 2014, 7 (3): 1247-1250.

[10] De Myttenaere A, Golden B, Le Grand B, et al. Mean absolute percentage error for regression models[J]. Neurocomputing, 2016, 192: 38-48.

[11] Cameron A C, Windmeijer F A G. An R-squared measure of goodness of fit for some common nonlinear regression models[J]. Journal of econometrics, 1997, 77 (2): 329-342.

[12] LI S X, LI B G. Comparison of prediction method based on SARIMA model and X-12-ARIMA seasonal adjustment method[J]. Statistics & Decision, 2018, 34 (18): 39-42.

[13] CHEN R, LIANG C Y, XIE F W. Application of nonlinear time series forecasting methods based on support vector regression[J]. Journal of Hefei University of Technology (Natural Science), 2013, 36 (3): 369-374.

[14] ELSAYED S, THYSSENS D, RASHED A, et al. Do we really need deep learning models for time series forecasting? [Z]. arXiv: 2101. 02118, 2021.

[15] M. Das, S.K. Ghosh. semBnet: A semantic bayesian network for multivariate prediction of meteorological time series data[J]. PatternRecognition Letter, 2017, 93 (1): 192-201.

[16] Borovykh A, Bohte S, Oosterlee C W. Conditional time series forecasting with convolutional neural networks[J]. arXiv preprint, arXiv: 1703. 04691, 2017.

[17] Bai S, Kolter J Z, Koltun V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling[J]. arXiv preprint, arXiv: 1803. 01271, 2018.

[18] LIU M H, ZENG A L, CHEN M X, et al. SCINet: Timeseries modeling and Forecasting with sample convolution and interaction[Z]. arXiv: 2106. 09305, 2021.

[19] Schuster M, Paliwal K K. Bidirectional recurrent neural networks[J]. IEEE transactions on Signal Processing, 1997, 45 (11): 2673-2681.

[20] Graves A, Schmidhuber J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures[J]. Neural networks, 2005, 18(5-6): 602-610.

[21] Qi Y, Li C, Deng H, et al. A deep neural framework for sales forecasting in e-commerce[C], Proceedings of the 28th ACM International Conference on Information and Knowledge Management, Beijing, Nov 3-7, 2019. New York: ACM, 2019: 299-308.

[22] Xin S, Ester M, Bu J, et al. Multi-task based sales predictions for online promotions[C], Proceedings of the 28th ACM International Conference on Information and Knowledge Management, Beijing, Nov 3-7, 2019. New York: ACM, 2019: 2823-2831.

[23] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C], Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach, CA, USA: Curran Associates Inc. 2017: 6000-6010.

[24] Wu S, Xiao X, Ding Q, et al. Adversarial sparse transformer for time series forecasting[J]. Advances in neural information processing systems, 2020, 33: 17105- 17115.

[25] Zhou H, Zhang S, Peng J, et al. Informer: Beyond efficient transformer for long sequence time-series forecasting[C], Proceedings of the AAAI Conference on Artificial Intelligence, virtual, Feb 2-9, 2021. California: AAAI,2021, 35 (12): 11106-11115.

[26] Lim B, Arı k S Ö, Loeff N, et al. Temporal fusion transformers for interpretable multi-horizon time series forecasting[J]. International Journal of Forecasting, 2021, 37 (4): 1748-1764.

[27] Lin Y, Koprinska I, Rana M. SSDNet: State space decomposition neural network for time series forecasting[C], 2021 IEEE International Conference on Data Mining (ICDM), Auckland, Dec 7-10, 2021. IEEE, 2021: 370-378.

[28] Wu H, Xu J, Wang J, et al. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting[J]. Advances in Neural Information Processing Systems, 2021, 34: 22419-22430.

[29] Qi X, Hou K, Liu T, et al. From known to unknown:Knowledge-guided transformer for time-series sales forecasting in Alibaba[J]. arXiv preprint, arXiv: 2109. 08381, 2021.

[30] Zhou T, Ma Z, Wen Q, et al. FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting[J]. arXiv preprint, arXiv: 2201. 12740, 2022.

[31] Wen Q, Gao J, Song X, et al. RobustSTL: A robust seasonal-trend decomposition algorithm for long time series[C], Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, Jan 27-Feb 1, 2019. California: AAAI, 2019, 33(01): 5409-5416.

[32] Li Y, Lu X, Xiong H, et al. Towards Long-Term Time-Series Forecasting: Feature, Pattern, and Distribution[J]. arXiv preprint, arXiv:2301.02068, 2023.

[33] WAN C, LI W Z, DING W X, et al. A Multivariate Time Series Forecasting Algorithm Based on Self-Evolutionary Pre-training[J]. Chinese Journal of Computer, 2022, 45(03):513-525.

[34] SALINAS D， FLUNKERT V， GASTHAUS J， et al. DeepAR: probabilistic forecasting with autoregressive recurrent networks[J]. International Journal of Forecasting, 2020, 36 (3): 1181-1191.