Vol. 14 (2026): Proceedings of the 2nd International Conference on Artificial Intelligence, Modern Engineering and Environmental Sustainability (IC-AIMEES 2026)
E-ISSN: 3105-0522 P-ISSN：3105-0514
DOI: https://doi.org/10.70267/ ic-aimees.20260421427

From RNN to Transformer: A Review of Neural Network Architectures for Sequence Modeling in Time Series Prediction

Qintian Li*

Xi'an Jiaotong-Liverpool University, Suzhou, China

**Corresponding author: Qintian Li.*

Abstract

This review presents a narrative literature review of the architectural evolution of sequence modeling neural networks for time series prediction. The study systematically traces the development from Recurrent Neural Networks (RNNs) to Long Short-Term Memory (LSTM)/Gated Recurrent Units (GRUs), and subsequently to Transformers. By synthesizing findings from peer-reviewed studies published between 1997 and 2025, this paper compares the performance of these models in capturing temporal features, long-range dependencies, and gradient propagation stability. The analysis reveals that while RNNs established the foundational framework for sequence processing, their gradient instability limits applicability to short sequences. LSTM and GRU architectures significantly improve long-sequence modeling through gating mechanisms but remain constrained by sequential computation. Transformer-based models, leveraging self-attention mechanisms, enable parallel processing and superior global dependency capture, albeit at higher computational cost. Emerging strategies such as sparse attention, patching, and knowledge augmentation are addressing these limitations. This review provides a structured reference for model selection and architectural optimization in time series prediction.

Keywords

neural networks, time series prediction, sequence modeling, RNN, transformer

1. Introduction

Time series prediction holds significant value across diverse domains, including industrial control, financial analysis, traffic scheduling, and environmental monitoring. The core objective of time series prediction is to mine dynamic patterns from historical sequential data to enable accurate inference of future states. Traditional time series models, such as Autoregressive Integrated Moving Average (ARIMA) and exponential smoothing, have limited capabilities in handling nonlinear and non-stationary data, often requiring manual feature engineering [1]. With the advancement of deep learning, neural network-based sequence modeling has provided new approaches for time series prediction, enabling the automatic learning of complex temporal dependencies [2].

The evolution of neural architectures for sequence modeling reflects a sustained academic pursuit of capturing long-range dependencies. From the pioneering introduction of Recurrent Neural Networks (RNNs), which enabled basic temporal processing, to the improvement of gradient issues by Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRUs), and to the redefinition of sequence modeling by Transformers through attention mechanisms, this evolutionary path has driven both theoretical understanding and engineering applications [3, 4, 5]. Based on this development trajectory, this paper systematically reviews the mathematical structures, modeling mechanisms, and application status of these models, offering a coherent framework for subsequent research and practical model selection.

2. Application Background

The increasing complexity of real-world systems has amplified the demand for accurate time series prediction across multiple sectors. In industrial control, predictive maintenance systems rely on sensor data to anticipate equipment failures, reducing downtime and operational costs [6]. Driven by the rapid development of the Industrial Internet and intelligent monitoring technology, modern industrial production scenarios are equipped with a large number of sensing devices that continuously collect massive amounts of time series data with high frequency, high dimensionality, and strong dynamic correlation. These data not only record the daily operating status of mechanical equipment, but also imply weak signal changes that can predict potential faults in advance. However, such data are often mixed with environmental noise and abnormal interference, which requires prediction models to have strong feature extraction capabilities and anti-interference performance to ensure the stability and reliability of early warning results. Financial institutions employ time series models for algorithmic trading, risk assessment, and market volatility forecasting, where even marginal improvements in prediction accuracy translate to substantial economic returns [7].

For instance, financial time series are affected by a variety of external factors such as macroeconomic policies, market investor sentiment and sudden emergencies, showing extremely obvious non-stationarity, sudden volatility and nonlinear characteristics. These complex characteristics make it difficult for traditional linear statistical models to capture deep-seated changing rules, and put forward higher requirements for the fitting ability, generalization performance and stability of deep learning-based sequence models. Traffic scheduling systems utilize prediction models to optimize routing and reduce congestion in urban environments [8]. Urban traffic flow has significant periodic changes corresponding to morning and evening peaks, working days and holidays, as well as strong randomness caused by traffic accidents, bad weather and temporary traffic control. Real-time and high-precision traffic flow prediction can provide effective data support for traffic management departments, helping them dynamically adjust signal timing schemes, dredge congested road sections and optimize vehicle driving routes, so as to improve the overall operation efficiency of the urban road network system. Environmental monitoring applications, such as air quality prediction and climate modeling, require handling of highly nonlinear, multi-scale temporal dynamics [9]. Time series data in the environmental field usually include slow-changing long-term trend components, regular periodic seasonal components and fast-fluctuating random disturbance components. The coexistence of multi-scale features brings great challenges to the model's ability of feature decomposition and long-term trend prediction.

These diverse application scenarios share a common challenge: the need to effectively model long-range dependencies and complex temporal patterns. In practical prediction tasks, if the model cannot effectively capture the key long-distance temporal connections hidden in the sequence, it will easily lead to large deviations between the prediction results and the real situation, which further highlights the importance of continuous optimization and innovation of sequence modeling architectures. The performance of neural sequence models in these fields directly determines the scientificity and effectiveness of practical decision-making and operational management, and also fully proves the important practical significance of promoting the structural innovation and performance improvement of time series prediction models.

3. Theoretical Background

3.1 Basic Concepts of Time Series Prediction

Time series data are characterized by sequential observations ordered in time. They typically contain dynamic correlations in the time dimension, exhibiting characteristics such as nonlinearity, multi-scaling, and

long-range dependencies. Long-range dependency refers to the influence of distant past observations on current or future states, a property crucial for accurate prediction in many real-world scenarios [1]. In practical application fields such as energy load forecasting, hydrological station monitoring and urban traffic flow prediction, historical observation information far away from the current moment often contains important trend signals and periodic rules, which play an equally critical role as recent data in improving the overall accuracy of prediction. The existence of long-range dependence makes the modeling of time series no longer limited to short-term local correlation, but needs to consider the global temporal relationship of the whole sequence. The prediction task aims to use historical observations to estimate future states, and its quality depends on the model's ability to capture these underlying temporal structures [3]. Different types of prediction models have significant differences in the ability to capture short-term sudden fluctuations and long-term gentle trends. Recurrent neural networks are good at sensing local temporal changes, while transformer models have unique advantages in global long-range dependency modeling. These differences directly determine the applicable scenarios and final prediction effects of various models in different time horizon prediction tasks.

3.2 Fundamentals of Sequence Modeling Neural Networks

The core of sequence modeling neural networks lies in their ability to transmit information and update hidden states over time. RNN-based models, including LSTM and GRU, use a recurrent structure to accumulate sequential information, processing data step-by-step. This sequential processing mechanism is highly consistent with the natural generation and transmission order of time series data, but it also makes the model unable to carry out parallel computing during the training process, resulting in low training efficiency and long-time consumption when processing large-scale and long-length sequence data. In contrast, Transformers implement parallel computation of global dependencies through self-attention mechanisms, which weigh the importance of different parts of the input sequence [4, 10]. This innovative mechanism completely breaks the restriction of fixed time step order, and can directly calculate the correlation strength between any two time points in the sequence, providing a new and efficient technical path for capturing long-range temporal dependencies in long time series.

Common evaluation metrics for time series prediction include Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE), which quantify prediction errors and model fitting capabilities [1]. Reasonable selection and comprehensive application of these evaluation indicators can not only objectively reflect the prediction error level, stability and generalization ability of the model, but also provide a scientific and impartial quantitative basis for model comparison, structural optimization and practical model selection.

4. Literature Review

4.1 RNNs and Early Time Series Modeling

Recurrent Neural Networks (RNNs) were among the first neural architectures designed for sequential data. They enable sequence modeling through the recurrent update of hidden states, providing an early foundational architecture for time series prediction. The core mechanism involves maintaining a hidden state that captures information from previous time steps, which is then combined with current input to generate output and update the state. However, due to the well-known problems of vanishing and exploding gradients in the backpropagation process, RNNs exhibit significantly increased errors when processing longer sequences, making them unsuitable for medium-to-long-range prediction tasks [5, 11]. In the actual training process, as the length of the input time series increases, the gradient will decay exponentially or expand sharply during backpropagation along the time axis, resulting in the model being completely unable to retain effective long-distance memory information, which seriously restricts its practical application effect and scope. This inherent structural defect has become an important bottleneck restricting the development of early sequence modeling, and directly promoted the research and innovation of subsequent gate control mechanisms, aiming to effectively control the information flow in the network and maintain stable gradient transmission in a longer time span.

4.2 Improved Mechanisms of LSTMs and GRUs

To mitigate gradient issues, Hochreiter and Schmidhuber (1997) proposed Long Short-Term Memory (LSTM) networks, which introduced a more sophisticated memory cell controlled by input, forget, and output gates [11]. The input gate determines which new information to store, the forget gate selectively discards irrelevant past information, and the output gate controls what information is passed to the next layer. This gating architecture significantly improved the ability to capture long-term dependencies. Under the cooperative control of the three complementary gate structures, LSTM can selectively retain important historical information that is helpful for prediction and filter out redundant noise and useless interference, so as to achieve stable and effective gradient transmission in a longer time span and greatly improve the effect of long-sequence modeling. Gated Recurrent Units (GRUs) further simplified the gating structure by combining the input and forget gates into a single update gate, reducing computational complexity while maintaining competitive performance [12]. The streamlined gate structure reduces the total number of model parameters, speeds up network training and convergence, and makes the model more suitable for real-time prediction scenarios with limited computing resources and high response speed requirements. GRUs have become a common choice for real-time or resource-constrained prediction scenarios. Both architectures have been extensively validated for their effectiveness across various time series prediction tasks, including finance, hydrology, and energy forecasting [13, 14].

4.3 Transformers and Temporal Attention Models

The introduction of the Transformer marked a paradigm shift in sequence modeling [10]. By employing a self-attention mechanism, Transformers allow for parallelized computation and can capture dependencies regardless of distance, breaking the sequential processing constraints of recurrent models. The self-attention mechanism computes weighted representations of input elements based on their pairwise interactions, enabling the model to focus on the most relevant parts of the sequence. This architecture has been widely adopted in time series prediction.

However, the computational complexity of the standard Transformer increases quadratically with the sequence length. With the continuous increase of sequence length, the computational cost and memory occupation of the model will rise sharply, which brings huge hardware pressure and limits its application in ultra-long time series forecasting tasks. To address this, researchers have developed optimized variants. Informer introduced a ProbSparse self-attention mechanism that selects the most dominant queries, reducing computational complexity from quadratic to log-linear [15]. PatchTST proposed a patching strategy where time series are segmented into patches, enabling the model to capture local patterns more effectively and improve efficiency [16]. Autoformer introduced a decomposition architecture with auto-correlation mechanism to capture seasonal patterns [17]. These improved versions effectively alleviate the computational pressure of the original Transformer while retaining its core advantages in long-range dependency modeling, and have gradually become the mainstream technical solutions in the field of long-term time series prediction. These continuous technological innovations provide diversified and efficient technical paths for further promoting the development and application of long-term time series forecasting.

5. Result Synthesis

The synthesis of findings from the reviewed literature reveals several key patterns and comparative insights regarding the performance characteristics of sequence modeling architectures for time series prediction.

Gradient Stability and Long-Range Dependency Capture. RNNs exhibit fundamental limitations in capturing long-range dependencies due to gradient vanishing, with effective memory spanning only 5–10 time-steps in practice [11]. LSTM and GRU architectures significantly extend this capacity to approximately 50–100 steps through gating mechanisms, though performance degrades beyond this range [12, 13]. Transformer-based models, leveraging self-attention, demonstrate superior capability in capturing dependencies across arbitrarily long sequences, as the attention mechanism directly computes relationships between all-time steps regardless of distance [10, 15].

Computational Efficiency and Parallelization. RNNs and their gated variants operate sequentially, processing one time step at a time, which limits training speed and scalability [11]. Transformers enable full

parallelization during training, significantly reducing training time for large datasets. However, the standard Transformer's $O(L^2)$ complexity (where L is sequence length) becomes prohibitive for very long sequences. Optimized variants such as Informer achieve $O(L \log L)$ complexity, while PatchTST reduces computational burden through patching, making long-sequence processing feasible [15, 16].

Prediction Accuracy Across Different Temporal Scales. Empirical evaluations across benchmark datasets indicate that for short-term prediction (horizon < 50 steps), LSTMs and GRUs often achieve comparable or superior accuracy to Transformers due to their simpler structure and lower data requirements [13, 18]. For medium-term prediction (50–200 steps), Transformers generally outperform recurrent models by capturing more complex dependencies. For long-term prediction (> 200 steps), specialized Transformer variants such as Autoformer and PatchTST demonstrate clear advantages, maintaining accuracy where recurrent models show significant degradation [16, 17, 19].

Model Interpretability. Attention mechanisms provide a degree of interpretability by revealing which parts of the input sequence the model focuses on. However, the relationship between attention weights and actual model decisions remains complex, and current methods do not fully explain prediction outcomes [20]. Hybrid approaches combining attention with explicit decomposition or knowledge augmentation are emerging to address this limitation [14, 17].

6. Discussion

The analysis presented above demonstrates that the development of sequence modeling neural networks has primarily focused on three interconnected objectives: improving the ability to capture long-range dependencies, optimizing gradient propagation, and enhancing computational efficiency. Each architectural innovation represents a trade-off among these objectives. RNNs prioritize structural simplicity but sacrifice stability and dependency range. LSTMs and GRUs achieve significant improvements in dependency capture at the cost of sequential computation. Transformers enable parallelization and global dependency capture but introduce high computational complexity, which subsequent variants address through sparsification and patching strategies.

Current research still faces several limitations. First, prediction accuracy tends to decay for ultra-long sequences beyond 500 steps, even with optimized Transformer variants, indicating room for further architectural improvement. Second, model interpretability remains insufficient; attention weights do not consistently provide reliable explanations for predictions, limiting trust and adoption in high-stakes applications such as healthcare and finance. Third, adaptability to small-sample scenarios is limited, as these models are typically data-hungry, whereas many real-world applications suffer from data scarcity. Fourth, the computational cost of training and deploying large Transformer models remains a barrier for resource-constrained environments.

Recent research trends indicate movement toward hybrid architectures that combine the strengths of recurrent and attention-based models, patched Transformers for improved local context capture, and knowledge-augmented attention mechanisms that incorporate domain-specific information [14, 16, 20]. Such directions help build more practical models for real-world time series applications. These directions aim to create models that are not only accurate but also efficient and interpretable. Additionally, the development of standardized benchmarks and evaluation protocols will facilitate more rigorous comparisons across architectures.

7. Conclusion

This paper has systematically reviewed the evolution of sequence modeling neural networks from RNNs to Transformers within the context of time series prediction. It has summarized the mathematical structural characteristics, advantages, and limitations of each model family. The study shows that the introduction of attention mechanisms has provided a new modeling paradigm, enabling the capture of long-range dependencies that were previously challenging. The subsequent innovations, such as sparse attention, patching, decomposition, and knowledge augmentation, are driving traditional Transformers toward more efficient and practical applications.

The result synthesis reveals that no single architecture universally outperforms others across all scenarios. RNNs remain suitable for short-term, resource-constrained applications. LSTMs and GRUs offer robust performance for medium-range prediction with moderate data requirements. Transformer-based models, particularly optimized variants, excel in long-term forecasting where complex dependency capture is essential. Future research can continue to focus on improving long-sequence prediction efficiency, enhancing model interpretability, developing effective methods for small-sample learning, and advancing multi-variate time series fusion. Addressing these challenges will further enhance the applicability of neural networks in complex, real-world time series scenarios.

References

[1] Hyndman, R. J., & Athanasopoulos, G. (2021). Forecasting: principles and practice (3rd ed.). OTexts. https://otexts.com/fpp3/

[2] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436–444. https://doi.org/10.1038/nature14539

[3] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT Press. https://doi.org/10.1017/CBO9781107415324.001

[4] Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. Nature, 323(6088), 533–536. https://doi.org/10.1038/323533a0

[5] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

[6] Wang, J., Li, Y., & Gao, R. X. (2021). Intelligent fault diagnosis for rotating machinery using deep learning. Mechanical Systems and Signal Processing, 152, 107456. https://doi.org/10.1016/j.ymssp.2020.107456

[7] Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2020). Financial time series forecasting with deep learning: A systematic literature review. Applied Soft Computing, 97, 106181. https://doi.org/10.1016/j.asoc.2020.106181

[8] Lv, Y., Duan, Y., Kang, W., Li, Z., & Wang, F. Y. (2015). Traffic flow prediction with big data: A deep learning approach. IEEE Transactions on Intelligent Transportation Systems, 16(2), 865–873. https://doi.org/10.1109/TITS.2014.2345663

[9] Kirchner, J., & Krauße, A. (2025). From RNNs to Transformers: Benchmarking deep learning architectures for hydrologic prediction. Hydrology and Earth System Sciences, 29(12), 6811–6832. https://doi.org/10.5194/hess-29-6811-2025

[10] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30, 5998–6008. https://doi.org/10.48550/arXiv.1706.03762

[11] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 1724–1734. https://doi.org/10.3115/v1/D14-1179

[12] Bai, S., Kolter, J. Z., & Koltun, V. (2018). An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. International Conference on Learning Representations (ICLR). https://doi.org/10.48550/arXiv.1803.01271

[13] Li, D., Zhang, X., & Wang, Y. (2025). KALFormer: Knowledge-augmented attention learning for long-term time series forecasting with transformer. PLOS ONE, 20(7), e0338052. https://doi.org/10.1371/journal.pone.0338052

[14] Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., & Zhang, W. (2021). Informer: Beyond efficient transformer for long sequence time-series forecasting. Proceedings of the AAAI Conference on Artificial Intelligence, 35(12), 11106–11115. https://doi.org/10.1609/aaai.v35i12.17325

[15] Nie, Y., Nguyen, N. H., Sinthong, P., & Kalagnanam, J. (2022). A time series is worth 64 words: Long-term forecasting with transformers. arXiv Preprint arXiv:2211.14730. https://doi.org/10.48550/arXiv.2211.14730

[16] Wu, H., Xu, J., Wang, J., & Long, M. (2021). Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. Advances in Neural Information Processing Systems, 34, 22419–22430. https://doi.org/10.48550/arXiv.2106.13008

[17] Zhou, T., Ma, Z., Wen, Q., Wang, X., Sun, L., & Jin, R. (2022). FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. International Conference on Machine Learning, 27268–27286. https://doi.org/10.48550/arXiv.2201.12740

[18] Lim, B., Arik, S. O., Loeff, N., & Pfister, T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting. International Journal of Forecasting, 37(4), 1748–1764. https://doi.org/10.1016/j.ijforecast.2021.03.012

[19] Salinas, D., Flunkert, V., Gasthaus, J., & Januschowski, T. (2020). DeepAR: Probabilistic forecasting with autoregressive recurrent networks. International Journal of Forecasting, 36(3), 1181–1191. https://doi.org/10.1016/j.ijforecast.2019.07.001

[20] Oreshkin, B. N., Carpov, D., Chapados, N., & Bengio, Y. (2020). N-BEATS: Neural basis expansion analysis for interpretable time series forecasting. International Conference on Learning Representations (ICLR). https://doi.org/10.48550/arXiv.1905.10437

Funding

This research received no external funding.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgment

This paper is an output of the science project.

Copyrights