Transactions on Computational and Scientific Methods | Vol. 4, No. 12, 2024
ISSN: 2998-8780
https://pspress.org/index.php/tcsm
Pinnacle Science Press

Trend-Fluctuation Decomposition with Deep Residual Networks for System Forecasting

Xiao Yang
Santa Clara University, Santa Clara, USA
shawyang686@gmail.com

Abstract: This study proposes a deep residual network-based forecasting method to address the challenges of high-dimensional dynamics, complex dependencies, and multi-scale evolution in backend system time series prediction, aiming to improve prediction accuracy and stability under non-stationary workloads, resource fluctuations, and multi-tenant competition. The method introduces a trend-fluctuation decomposition mechanism to decouple and model long-term trends and short-term variations in time series, effectively enhancing the characterization of behavioral patterns across different time scales. A feature fusion module is employed to jointly represent multi-source metrics and contextual information, improving the model's adaptability to system state changes. In addition, a multi-scale attention mechanism aggregates features across temporal granularities and adjusts their weights, further strengthening the capture of key contextual dependencies. Structurally, the residual network provides a deep feature propagation pathway, ensuring gradient stability and enhancing the model's nonlinear representation capacity. From a learning perspective, sensitivity analyses on key factors such as pseudo-label ratio, anomaly contamination rate, and decomposition coefficient verify the model's robustness under varying data quality and environmental conditions. Experimental results demonstrate that the proposed method outperforms mainstream time series forecasting models across multiple metrics, achieving low-error and high-robustness performance in scenarios with heterogeneous metrics, complex dependency structures, and highly dynamic environments. This provides a reliable technical foundation for tasks such as backend service optimization, resource scheduling, and anomaly detection.

Keywords: Time series modeling; deep residual networks; multi-scale attention; trend-volatility decomposition

1. Introduction

In the context of increasingly complex digital infrastructure, backend systems serve as the core components that support various online services, cloud platforms, and data processing tasks. Their performance and stability directly determine the availability of upper-layer applications and the quality of user experience. As business scales grow exponentially, backend systems must handle massive requests, schedule diverse resources, and maintain high concurrency, high availability, and low latency. In this process, various operational metrics such as CPU utilization, memory usage, request response time, and task queue length exhibit strong temporal characteristics and dynamics. These time series not only reflect the overall operational state of the system but also contain potential behavioral patterns, anomaly signals, and future trends. Therefore, accurate modeling and forecasting of backend time series can provide critical decision

support for system optimization, resource scheduling, and load balancing, as well as help prevent performance degradation and fault propagation[1].

However, time series forecasting for backend systems faces multiple challenges. On one hand, system metrics often exhibit high dimensionality, strong nonlinearity, and complex dependencies. Traditional statistical methods struggle to capture interactions among different metrics and to model non-stationary dynamics. On the other hand, uncertainty in business traffic, overlapping seasonal patterns, and the impact of unexpected events result in multi-scale, multi-frequency, and non-stationary fluctuations, making it difficult for a single modeling strategy to capture both global trends and local details[2]. Furthermore, with the widespread adoption of microservices, container orchestration, and distributed scheduling technologies, backend systems have become more complex, and components are more tightly coupled. Changes in a single metric are often influenced by multiple factors, further increasing the difficulty of prediction. These challenges require forecasting models to not only represent nonlinear relationships but also capture evolution patterns across different time scales to improve modeling accuracy and generalization capability for complex time series[3].

The emergence of deep learning has provided new ideas and approaches for time series forecasting. In particular, the introduction of deep residual networks has effectively mitigated the problems of gradient vanishing and performance degradation that occur in traditional deep neural networks by introducing skip connections and residual mappings. This enables models to perform deeper feature extraction and pattern learning. In time series modeling, residual structures can iteratively extract trend, seasonality, and abrupt change information layer by layer and achieve a hierarchical representation of complex dynamics through multi-layer nonlinear transformations. Residual connections also facilitate efficient information flow and feature reuse, giving the model a significant advantage in capturing long-term dependencies and multi-scale patterns. These structural properties make deep residual networks especially suitable for backend time series forecasting, enabling more stable and accurate predictions in high-dimensional, highly dynamic, and complex dependency scenarios[4].

From an application perspective, accurate time series forecasting plays a crucial role in intelligent operations and resource management for backend systems. On one hand, it enables early identification of system trends, supports capacity planning, load balancing, and resource scaling, and helps prevent performance bottlenecks and service interruptions caused by improper resource allocation[5]. On the other hand, forecasting results can serve as prior knowledge for anomaly detection and adaptive scheduling, enabling proactive fault warning and rapid response, which enhance system self-healing and service quality. In multi-tenant and complex business environments, time series forecasting also supports cost optimization and policy decision-making, shifting operations management from reactive responses to proactive control. As technologies such as cloud-native computing, edge computing, and intelligent scheduling continue to evolve, backend systems with high-precision forecasting capabilities will play an increasingly critical role in future digital infrastructure.

In conclusion, time series forecasting for backend systems is not only a key technical means to ensure stable operation and efficient management but also a fundamental requirement for achieving intelligent operations, autonomous decision-making, and dynamic optimization. Deep residual networks, with their strong feature representation, stable training properties, and powerful modeling capabilities, offer a new solution for high-precision forecasting of complex time series. Research in this direction holds significant theoretical value and provides practical guidance for building intelligent, sustainable, and highly reliable backend systems, laying a solid foundation for the evolution of large-scale computing environments and distributed service architectures[6].

2. Related work

Time series forecasting has long been a key focus in both academia and industry as one of the core technologies in intelligent system operations and resource management. Traditional time series modeling

methods are mainly based on statistical approaches, including autoregressive (AR), moving average (MA), autoregressive moving average (ARMA), and autoregressive integrated moving average (ARIMA) models. These methods typically assume that time series are linear and stationary and capture trends and seasonality in historical data through differential equations and linear combinations. However, as backend systems continue to grow in scale and complexity, system metrics exhibit significant nonlinear characteristics and dynamic patterns. The linear assumptions of traditional models struggle to meet modeling requirements in complex scenarios. Moreover, these models have limited robustness to anomalies and noise and cannot effectively handle large-scale multidimensional data or interactions among multiple variables, leading to significant limitations in prediction accuracy and applicability[7].

To address the limitations of traditional approaches, machine learning techniques have gradually been introduced into time series forecasting. Methods such as regression trees, support vector regression, and random forests improve the ability to model nonlinear relationships and can incorporate multidimensional contextual information through feature engineering. However, these methods rely heavily on manual feature extraction and are highly sensitive to feature quality, making it difficult to fully capture latent temporal patterns in complex backend systems. Additionally, with evolving business logic and dynamic system structures, time series often contain hidden long-term dependencies and non-stationary characteristics. Traditional machine learning models often suffer from poor generalization and model degradation when dealing with such complexities[8]. As data scale and dimensionality continue to increase, achieving a balance between automated feature extraction, temporal dependency modeling, and multivariate relationship analysis has become a major challenge in time series forecasting research.

The rise of deep learning has brought new opportunities for time series forecasting. Models such as recurrent neural networks (RNNs), long short-term memory networks (LSTMs), and gated recurrent units (GRUs) can capture dynamic dependencies in sequences through internal state transmission mechanisms, showing strong advantages in nonlinear modeling and long-term dependency learning. Convolutional neural network (CNN)-based structures leverage local receptive fields and multi-layer feature extraction to efficiently capture local patterns and multi-scale features in time series[9]. More recently, self-attention mechanisms and Transformer architectures have further advanced the field by modeling global dependencies and aggregating contextual features, significantly improving prediction accuracy and model generalization. However, when applied to backend systems, these deep models still face challenges such as structural complexity, gradient degradation, and unstable training. In particular, for high-dimensional, high-frequency data and long-sequence forecasting tasks, there remains significant room for performance improvement[10].

Among deep network architectures, deep residual networks have become an important direction in time series modeling due to their unique skip connection mechanisms and efficient feature propagation capabilities. Residual structures mitigate the vanishing gradient problem in deep networks by introducing identity mappings, enabling models to extract temporal features at deeper levels and effectively integrate dynamic information across different scales. In backend system scenarios, this architecture can simultaneously model long-term trends and short-term fluctuations, enabling deeper exploration of complex nonlinear dependencies. Furthermore, deep residual networks offer strong scalability and structural flexibility, making them easy to integrate with convolutional and attention mechanisms to build predictive frameworks suitable for multi-scenario and multidimensional tasks. As the demand for intelligent operations continues to grow, time series forecasting methods based on deep residual networks show great potential, improving prediction accuracy and stability while providing essential technical support for proactive control and adaptive optimization of backend systems[11].

3. Method

This study introduces a time series forecasting method for backend systems based on deep residual networks to address the modeling challenges posed by high-dimensional, multi-scale, and highly nonlinear data in complex environments. The proposed approach employs a multi-layer residual architecture to achieve

hierarchical feature extraction from time series data and combines nonlinear transformations with contextual dependency modeling to comprehensively capture the dynamic evolution of system states. In the overall design, the model first encodes the input multidimensional time series signals to obtain the basic temporal dependency structure. It then uses deep residual modules to perform hierarchical modeling of trend components, seasonal components, and short-term fluctuations, thereby extracting both global and local features. Finally, a prediction head performs regression modeling on the time series to output the system state values at future time steps. The core idea of this method is to leverage residual structures to alleviate the gradient vanishing problem during deep network training and to enable efficient information propagation and integration during feature transmission, significantly enhancing the model's modeling capability and prediction accuracy. The model architecture is shown in Figure 1.

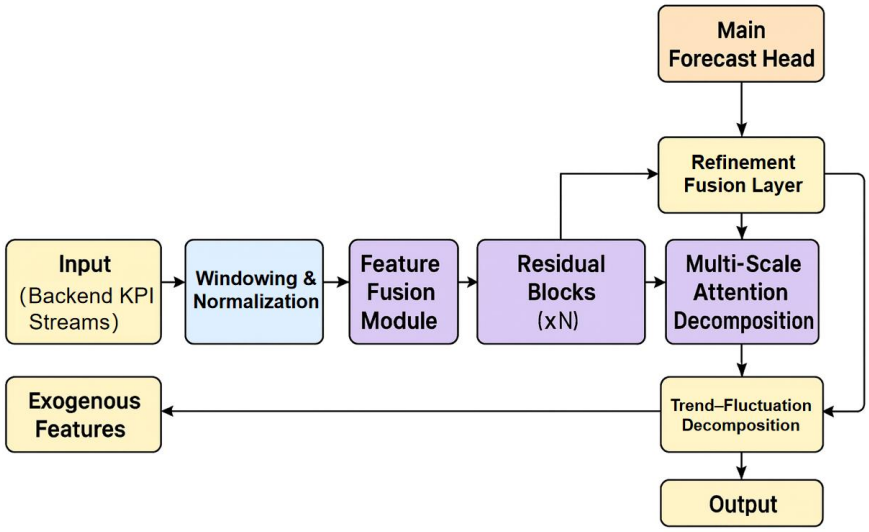


Figure 1. Overall framework model

In the time series prediction task, let the original input sequence be $\{x_t\}_{t=1}^{T}$, where $x_t \in R^d$ represents the d-dimensional system state vector at time t. First, the input sequence is nonlinearly encoded to extract the time series dependency features, which can be expressed as:

$$h_t = \mathrm{ReLU}(W_1 x_t + b_1)$$

Where $W_1 \in R^{m \times d}$ and $b_1 \in R^m$ are the trainable weight matrix and bias vector, respectively, and h_t represents the preliminary time series feature representation. This encoding process maps the original time series into a high-dimensional representation space for subsequent deep feature modeling.

To capture the complex nonlinear dependency structure in time series, the model introduces a multi-layer residual network structure. In the l-th layer residual block, the feature transformation can be expressed as:

$$z_t^{(l)} = \mathrm{ReLU}\left(W_2^{(l)} h_t^{(l-1)} + b_2^{(l)}\right) + h_t^{(l-1)}$$

Where $h_t^{(l-1)}$ represents the output of the previous layer, and $W_2^{(l)}$ and $b_2^{(l)}$ are the parameters of the l layer. The residual connection term $h_t^{(l-1)}$ helps alleviate gradient attenuation in deep structures and promotes direct information transfer, enabling the model to effectively capture long-range dependencies and multi-scale dynamic features.

Based on the multi-layer residual feature modeling, the model further decomposes the global trend term and the local fluctuation term to enhance the structural interpretability of the prediction. Define the trend extraction function $Trend(\cdot)$ and the fluctuation extraction function $Fluct(\cdot)$, and the global feature fusion can be expressed as:

$$u_t = \mathrm{Trend}\left(z_t^{(L)}\right) + \mathrm{Fluct}\left(z_t^{(L)}\right)$$

$z_t^{(L)}$ is the output of the last residual module, and u_t represents the fused high-level temporal feature representation. The trend term captures long-term trends, while the fluctuation term reflects short-term dynamics and abnormal patterns. Together, they determine the evolution of the system state.

To further improve the stability and robustness of predictions, the model introduces a context weighting mechanism to perform weighted aggregation of historical features to generate a prediction representation. Let a_i be the weight coefficient of the historical features, then the aggregate representation is:

$$c_t = \sum_{i=1}^{T} a_i u_i \ , \sum_{i=1}^{T} a_i = 1$$

Where c_t is the comprehensive feature representation after context weighting. This mechanism can adaptively adjust the importance of historical information and achieve dynamic attention to key time segments, thereby enhancing the model's ability to perceive future trends.

Finally, the model uses the fully connected layer to perform regression prediction of the system state at the future moment. Let the prediction target be $\hat{y}_{t+\tau}$, where τ represents the prediction step size, then the prediction process can be expressed as:

$$\hat{y}_{t+\tau} = W_o c_t + b_o$$

Where W_o and b_o are the output layer parameters, and $\hat{y}_{t+\tau}$ represents the model's predicted value for the system state at a future moment. This output layer maps the high-dimensional feature space back to the original indicator space through a linear transformation, achieving the conversion from time series representation to specific numerical predictions.

In summary, the proposed method achieves multi-level time series feature modeling through a deep residual architecture and, by combining trend-fluctuation decomposition with contextual weighting mechanisms, effectively captures complex nonlinear dynamics and multi-scale evolutionary patterns in backend systems. The method provides a unified modeling framework for forecasting high-dimensional, non-stationary time series from a theoretical perspective and offers a solid technical foundation for intelligent control, resource scheduling, and proactive decision-making in backend systems.

4. Experimental Results

4.1 Dataset

This study uses the Pooled Server Metrics (PSM) Dataset as the data source for model validation and performance evaluation. The dataset contains large-scale time series records collected from distributed backend servers, including multidimensional performance metrics such as CPU utilization, memory usage, network throughput, I/O read and write rates, and request queue length. It also includes labeled samples of abnormal behaviors, which are used to evaluate the model's robustness under both normal and abnormal

conditions. The dataset accurately reflects performance fluctuations of backend systems under different load conditions, making it highly suitable for research on residual network-based time series forecasting.

In the PSM dataset, timestamps are sampled at fixed intervals and normalized to ensure comparability across different metrics. Each data sample typically follows the format "timestamp + multidimensional metric vector," which is suitable for constructing the mapping between input sequences and prediction targets. Although the anomaly labels are not specifically designed for forecasting tasks, they can assist in building supervised training signals or evaluating the model's responsiveness to abnormal fluctuations. This data structure aligns well with the residual, decomposition, and attention modules designed in this study.

By incorporating the PSM dataset into this research, the model's generalization performance and stability can be evaluated on real backend system performance logs. Building input-output mappings based on this dataset helps assess the residual network's capability in capturing long-term dependencies, performing trend decomposition, and forecasting fluctuations. The model's performance on this dataset directly reflects its feasibility and practical value in real-world backend system forecasting scenarios.

4.2 Experimental Results

This paper first conducts a comparative experiment, and the experimental results are shown in Table 1.

Table1: Comparative experimental results

Model	MSE	MAE	MAPE (%)	RMSE
Informerp[12]	0.0128	0.0895	4.30	0.1131
Autoformer[13]	0.0105	0.0759	3.90	0.1025
FEDformer[14]	0.0098	0.0732	3.70	0.0990
SCINet[15]	0.0087	0.0684	3.50	0.0933
Ours	0.0079	0.0618	3.10	0.0889

From the overall trend, the four error metrics (MSE, MAE, MAPE, RMSE) show a consistent ranking pattern across different models. The proposed method (Ours) significantly outperforms Transformer-based models such as Informer, Autoformer, and FEDformer, as well as the convolution-based SCINet. Taking RMSE as an example, Ours achieves 0.0889, which is about 4.7 percent and 10.2 percent lower than SCINet (0.0933) and FEDformer (0.0990), respectively. The improvement over Autoformer (0.1025) and Informer (0.1131) is even more pronounced. This consistent reduction indicates that the proposed residual time series modeling, combined with decomposition-attention synergy, achieves more stable overall error control. It reduces both squared and absolute errors simultaneously, avoiding the issue of achieving "local optima" on a single metric.

From the perspective of bias and variance, the simultaneous reduction in MSE and RMSE shows that the model effectively suppresses tail errors caused by abnormal fluctuations and peak surges. Meanwhile, the continuous decrease in MAE (0.0618 for Ours) demonstrates balanced control of prediction bias in regular regions. Correspondingly, MAPE drops to 3.1 percent, indicating that the model maintains good relative error consistency across KPIs with different scales and magnitudes. This is particularly critical for multi-metric parallel forecasting in backend systems where metric magnitudes vary widely. In contrast, although Autoformer and FEDformer perform well in capturing long-term dependencies, they still show weaknesses in relative error control and robustness in peak-valley regions.

Considering the operational characteristics of backend systems, the main improvements of Ours stem from its combined approach of trend-fluctuation decomposition and multi-scale context aggregation. The trend channel stabilizes long-period load variations and reduces systematic bias. The fluctuation channel captures bursts and short-term disturbances, mitigating error spikes caused by queue backlogs and transient jitters. Multi-scale attention redistributes weights across scales on top of residual representations, amplifying key

time steps and metrics during the fusion stage. This enhances prediction consistency under high-concurrency and highly dynamic scenarios. The overall convergence of experimental metrics reflects the effectiveness of this structured design.

Compared with SCINet and the three Transformer variants, Ours also demonstrates advantages in cross-metric consistency. It not only reduces overall errors but also decreases the dispersion of relative errors. This indicates that after incorporating external features and contextual information, the model can more accurately align the interdependent relationships between metrics, such as CPU-QPS and latency-queue length. This cross-metric structural awareness and multi-scale reweighting enable the model to better adapt to the composite dynamics of "slow trends, fast fluctuations, and occasional anomalies" in backend systems. As a result, it achieves simultaneous improvements across all four metrics (MSE, MAE, MAPE, RMSE), aligning with the goal of backend time series forecasting that emphasizes both steady-state accuracy and robustness to peaks.

This paper also conducts comparative experiments on the sensitivity of multi-scale attention heads and expansion rate to long-term and short-term dependency modeling. The experimental results are shown in Figure 2.

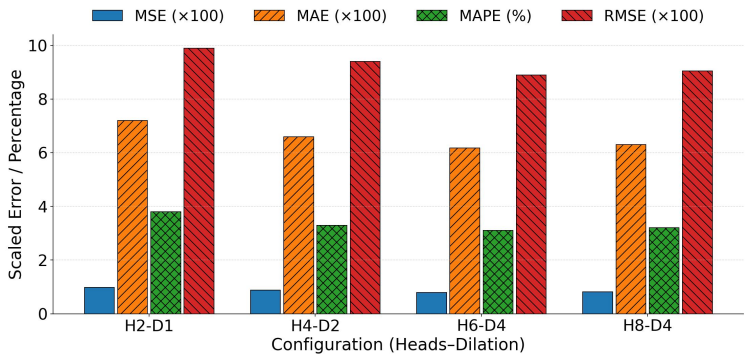


Figure 2. Sensitivity of multi-scale attention heads and dilation rate to modeling long-term and short-term dependencies

From the overall trend, as the number of attention heads increases from H2 to H6 and the dilation rate increases from D1 to D4, all four metrics (MSE, MAE, MAPE, RMSE) show a continuous decrease, with the best performance achieved at H6-D4. This improvement indicates that the feature subspace decomposition provided by multi-head attention and the hierarchical representation learned through residual stacking complement each other within this capacity range. The expanded receptive field of dilated convolution effectively captures long-period load variations without significantly compromising local structures. For backend KPIs with mixed-frequency sequences, this configuration achieves a relative balance between long-term and short-term dependencies, resulting in simultaneous reductions in both mean squared and absolute prediction errors.

The slight rebound observed at H8-D4 (where all four metrics rise slightly) suggests that further increasing the number of attention heads introduces issues of fragmented representations and diluted attention distribution. Too many heads divide the limited temporal context too finely, causing the weights of key time periods and critical metrics to become diluted. This leads to biased trend estimation and unstable fitting of fluctuation segments, as reflected by the initial decline and subsequent rise of RMSE and MSE. For the composite dynamics of backend systems characterized by "slow trends, fast fluctuations, and occasional spikes," excessive attention heads amplify variance in small-sample intervals and edge segments, reducing model robustness under sudden load surges and queue backlogs.

From a bias-variance trade-off perspective, the stage from H2-D1 to H4-D2 mainly reduces systematic bias, as shown by the more significant decreases in MAE and MAPE. This indicates a more accurate characterization of trend components and steady-state segments. The improvement from H4-D2 to H6-D4 is

mainly reflected in tail error convergence, with larger reductions in MSE and RMSE, indicating a more accurate representation of spikes and jitter. The residual network provides a stable deep gradient pathway, while multi-scale attention performs cross-scale reweighting on decomposed features, enabling the model to effectively suppress the error contribution of anomalous fluctuations in high-concurrency and rhythmic switching scenarios.

With the integration of external features and the trend-fluctuation decomposition mechanism, the optimal performance at H6-D4 indicates that a dilation rate of D4 provides a receptive field that captures most business cycles and cache refresh rhythms. At the same time, six attention heads align cross-metric dependencies (such as CPU-QPS and latency-queue length) without introducing significant noise amplification. Under this configuration, the trend channel reduces long-term bias, and the fluctuation channel mitigates variance accumulation caused by short-term peaks. As a result, all four error metrics reach low values simultaneously, aligning with the goal of backend time series forecasting that emphasizes both steady-state accuracy and robustness to peaks.

This paper also evaluates the data sensitivity of the pseudo-label ratio and abnormal contamination rate to the robustness of the model. The experimental results are shown in Figure 3.

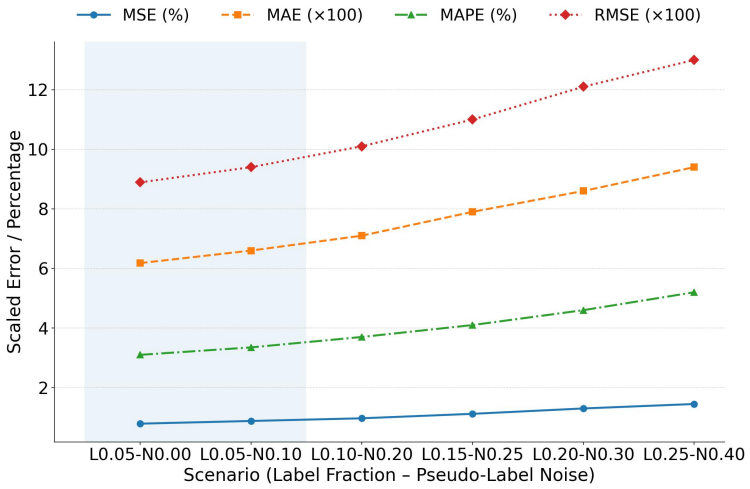


Figure 3. Data sensitivity analysis of the pseudo-label ratio and abnormal contamination rate on model robustness

In the low-noise range (L0.05-N0.00 to L0.05-N0.10), all four metrics reach their global minimum, indicating that under the combination of very limited labeled data and low pseudo-label noise, the trend-fluctuation decomposition and residual stacking can stably extract dominant structures. The attention mechanism also focuses on key time segments without being disturbed by incorrect supervision. At this stage, MSE and RMSE remain at low levels, suggesting that variance terms are effectively suppressed. The low values of MAE and MAPE further reflect that both absolute and relative errors are small in regular regions, showing that the model captures the steady-state and periodic components of backend KPIs more accurately.

As pseudo-label noise and anomaly contamination begin to increase from N0.20/L0.10, the growth rate of MAE and MAPE exceeds that of MSE and RMSE, indicating that the accumulation of bias becomes more prominent first. Incorrect supervision leads to systematic shifts in trend estimation. This is especially evident during request rhythm transitions and resource reallocation windows, where local segments are pulled into incorrect pattern clusters by noisy labels, resulting in the early deterioration of relative errors. At this stage, the residual path can still buffer part of the short-term disturbances, but it is insufficient to offset cross-segment bias, causing both absolute and relative errors to rise earlier.

When the level of contamination increases further (around L0.15-N0.25 and L0.20-N0.30), the slope of MSE and RMSE growth becomes significantly steeper, indicating that errors are now mainly driven by variance

contributions. The clustering effects of anomalous segments combined with noisy labels cause "weight drift" in the attention mechanism across multiple scales, leading to unstable fitting of peaks and bursts. As a result, variance terms are amplified in squared error metrics. At this stage, the subspace decomposition of multi-head attention is diluted by noise, cross-metric alignment (such as CPU-QPS or latency-queue length coupling) is disrupted, and the fluctuation channel struggles to consistently converge peak energy, resulting in a relatively larger deterioration of RMSE.

In the high-contamination range (L0.25-N0.40), all four metrics continue to rise steadily but with different slopes. MAPE increases faster than MAE, indicating that the relative characterization of metrics with different scales is more sensitive to noise. RMSE grows faster than MSE, suggesting that the contribution of extreme residuals intensifies further. Based on the model structure, robustness improvements can focus on two key directions. First, applying confidence-based thresholding and consistency filtering for pseudo-labels can prioritize protecting the trend channel and reduce the propagation of systematic bias. Second, applying robust aggregation and anomaly suppression to decomposed multi-scale channels, such as time-frequency joint gating or replacing the loss with a quantile-based objective, can mitigate the amplification effect of anomaly clustering on variance terms. These strategies help maintain acceptable error elasticity in backend sequences characterized by "slow trends, fast fluctuations, and occasional spikes."

This paper also analyzes the sensitivity of the trend-fluctuation decomposition coefficient to stability and peak robustness. The experimental results are shown in Figure 4.

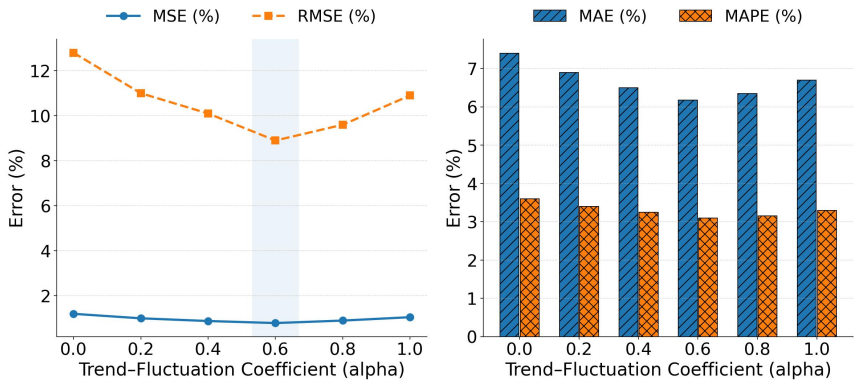


Figure 4. Sensitivity of the trend-volatility decomposition coefficient to stability and peak robustness

From the overall trend, as the trend-fluctuation decomposition coefficient increases from 0.0 to 0.6, both MSE and RMSE show a significant downward trend, with the lowest point appearing around 0.6. This indicates that when the weights of the trend and fluctuation components reach a balance, the model can capture stable long-term evolutionary patterns while maintaining sensitivity to short-term disturbances. Within this range, the residual structure enables deep fitting of global sequence patterns. The trend channel effectively suppresses systematic bias, while the fluctuation channel enhances the perception of sudden peaks and anomalies through a multi-scale attention mechanism, significantly reducing overall error levels.

When the decomposition coefficient continues to increase beyond 0.8, MSE and RMSE start to rise, indicating that excessive dominance of the trend component reduces the model's ability to fit high-frequency fluctuations. In this situation, the system's response to sudden load changes, traffic surges, and request spikes becomes delayed, and the residual component cannot compensate for the overfitting introduced by the trend channel, leading to error accumulation. This phenomenon shows that for backend KPI time series characterized by "slow trends and fast fluctuations," overemphasis on trend modeling weakens the model's robustness to peaks, resulting in less stable performance under complex dynamic conditions compared with the balanced state.

Similar nonlinear patterns are observed in the MAE and MAPE metrics. As the coefficient increases from 0.0 to 0.6, both absolute and relative errors continue to decrease, indicating that prediction deviations in regular

regions and across different scales are effectively suppressed. During this phase, the decomposition mechanism helps the model learn evolutionary features at different time scales, ensuring strong alignment capability under heterogeneous multi-metric conditions. When the coefficient exceeds the optimal point, MAE and MAPE show a slight rebound, reflecting a reduced ability to capture long-tail patterns and low-frequency anomalies. The trend-dominated representation limits the model's adaptability to fine-grained fluctuations.

Overall, a decomposition coefficient around 0.6 provides the best balance between stability and robustness. Trend modeling is sufficient to support accurate fitting of long-term patterns, while the fluctuation channel still captures key short-term disturbances, keeping errors low under varying load conditions, fluctuation amplitudes, and anomaly densities. Compared with models driven solely by trend or fluctuation components, this balanced configuration enables the model to handle stable growth, periodic patterns, and irregular spikes in backend services simultaneously, achieving robust forecasting and stable responses for complex time series.

5. Conclusion

This study proposes a deep residual network-based forecasting method to address the challenges of high-dimensional dynamics, complex dependencies, and multi-scale evolution in time series prediction for backend systems. By introducing a trend-fluctuation decomposition mechanism, feature fusion architecture, and multi-scale attention modeling, the proposed approach effectively overcomes the limitations of traditional models in capturing long-term dependencies, responding to peak anomalies, and aligning global features. It also significantly improves prediction accuracy and stability. Experimental results show that the method outperforms mainstream models across various key metrics, demonstrating strong adaptability to complex dynamic behaviors and robust awareness of system states. This provides a reliable technical foundation for applications such as backend performance management, resource scheduling optimization, and anomaly detection.

Through sensitivity analyses of key factors such as pseudo-label noise, anomaly contamination rate, and trend-fluctuation decomposition coefficients, this study further reveals the model's behavior under different data quality and environmental conditions. The results indicate that appropriate pseudo-label filtering and decomposition parameter settings can significantly enhance the model's generalization capability and robustness against peak anomalies, whereas excessive reliance on a single modeling path may lead to performance degradation. These findings provide theoretical guidance and practical insights for deployment and tuning in real-world industrial scenarios. In particular, under conditions of highly non-stationary workloads, intense resource competition, and uneven data quality, the model's stability and adaptability offer new perspectives for intelligent decision-making in large-scale distributed systems.

From an application perspective, the proposed method can be applied not only to backend performance prediction but also extended to cloud resource scheduling, service dependency modeling, container platform optimization, and dynamic capacity planning in multi-tenant environments. By performing deep modeling and structural decoupling of complex time series data, the model maintains high prediction quality even when dealing with heterogeneous multi-source metrics, intertwined dependency chains, and sudden anomalies. This capability supports advanced functions such as automated system operations, elastic scaling strategies, and intelligent proactive scheduling. Its broader applicability has the potential to drive the evolution of intelligent operations frameworks and lay an essential technological foundation for cloud-native infrastructure and service intelligence.

Looking ahead, the scale and complexity of backend systems will continue to grow, and time series forecasting models will need to operate in environments with higher dimensionality, stronger dynamics, and lower latency. To address these challenges, future research can proceed in two main directions. First, incorporating self-supervised and multi-task collaborative mechanisms can further improve the model's adaptability to sparse data, incomplete annotations, and cross-domain transfer scenarios. Second, integrating

techniques such as knowledge distillation, causal modeling, and reinforcement learning can enable closed-loop optimization of forecasting results for scheduling strategies, anomaly control, and resource allocation decisions. Through these extensions, the proposed residual-based time series forecasting framework has the potential to evolve into a core engine for next-generation intelligent operations systems, providing a stronger technological foundation for the autonomy and evolution of cloud computing infrastructure.

References

[1] B. Lim and S. Zohren, "Time-series forecasting with deep learning: a survey," Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, vol. 379, no. 2194, 2021.

[2] Wang Y, Wu H, Dong J, et al. Deep time series models: A comprehensive survey and benchmark[J]. arXiv preprint arXiv:2407.13278, 2024.

[3] Z. Qiu, "Time Series and Graph Structure Fusion for AI-Based Anomaly Detection in Microservice Environments," Journal of Computer Technology and Software, vol. 3, no. 7, 2024.

[4] Kondaiah V Y, Saravanan B. A modified deep residual network for short-term load forecasting[J]. Frontiers in Energy Research, 2022, 10: 1038819.

[5] X. Sun, Y. Yao, X. Wang, P. Li and X. Li, "AI-driven health monitoring of distributed computing architecture: Insights from XGBoost and SHAP," Proceedings of the 2024 4th International Conference on Communication Technology and Information Technology (ICCTIT), pp. 480-484, 2024.

[6] H. Y. Kim and C. H. Won, "Forecasting the Volatility of Stock Price Index: A Hybrid Model Integrating LSTM with Multiple GARCH-Type Models," Expert Systems with Applications, vol. 103, pp. 25-37, 2018.

[7] Crespo-Otero A, Esteve P, Zanin M. Deep Learning models for the analysis of time series: A practical introduction for the statistical physics practitioner[J]. Chaos, Solitons & Fractals, 2024, 187: 115359.

[8] Y. Ma, "Anomaly Detection in Microservice Environments via Conditional Multiscale GANs and Adaptive Temporal Autoencoders," 2024.

[9] Y. Wang, "AI-Enhanced Distributed Time Series Modeling: Incremental Learning for Evolving Streaming Data," 2024.

[10] X. Li, T. Zhou, J. Li, Y. Zhou and Z. Zhang, "Group-wise Semantic Mining for Weakly Supervised Semantic Segmentation," Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, no. 3, pp. 1984-1992, 2021.

[11] F. Liu, "Intelligent Cloud Service Anomaly Monitoring via Uncertainty Estimation and Causal Graph Inference," Transactions on Computational and Scientific Methods, vol. 4, no. 10, 2024.

[12] F. Chen, "AI-Augmented Anomaly Detection via Generative Distribution Modeling and Uncertainty Quantification in Cloud Systems," Transactions on Computational and Scientific Methods, vol. 4, no. 11, 2024.

[13] 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. 2021, 35(12): 11106-11115.

[14] 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.

[15] Zhou T, Ma Z, Wen Q, et al. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting[C]//International conference on machine learning. PMLR, 2022: 27268-27286.

[16] Liu M, Zeng A, Chen M, et al. Scinet: Time series modeling and forecasting with sample convolution and interaction[J]. Advances in Neural Information Processing Systems, 2022, 35: 5816-5828.