

A Distribution-Adaptive Attention Transformer for Blood Pressure Prediction

Chen Wang^{1, *}

¹ School of Software Engineering, Chengdu University of Information Technology, Chengdu, 610225, China

* Corresponding author

Abstract: Accurately predicting intraoperative blood pressure to prevent intraoperative hypotension is of significant importance. However, the task of intraoperative blood pressure prediction often faces the challenge of substantial sample distribution disparities due to individual physiological differences among patients and the diversity of surgical conditions. These disparities can cause a decline in model performance when encountering inconsistently distributed samples, thus limiting its generalizability in real-world clinical settings. To address this challenge, this paper develops a distribution-adaptive attention strategy aimed at mitigating distributional differences by incorporating statistical information within the model's internal design. The strategy automatically adjusts the model's attention weights by learning statistical information from input windows, enhancing the capability to learn the inherent distribution patterns within non-stationary sequences. Consequently, it reduces performance fluctuations caused by distributional differences. The effectiveness of the proposed distribution-adaptive attention strategy is demonstrated through multiple comparative experiments in the experimental section.

Keywords: Blood Pressure Prediction, Distribution-Adaptive Attention, Non-Stationary Sequences, Distribution Disparities.

1. Introduction

Intraoperative hypotension is a common clinical risk that can lead to long-term harm and postoperative complications for patients. Accurately predicting the blood pressure state of patients during surgery to prevent intraoperative hypotension plays a positive role in improving surgical safety and reducing postoperative complications[1].

The task of intraoperative blood pressure management

faces a challenge as shown in the figure. Figure 1. During preprocessing and training, individual patient differences aren't accounted for, and physiological responses to external stimuli can differ among patients, causing inconsistent sample distributions[2]. These inconsistencies arise from individual variability. Additionally, physiological states can shift significantly during surgery due to factors such as medication, surgical differences, and individual physiological reactions, leading to non-stationarity in blood pressure series and distribution shift problems.

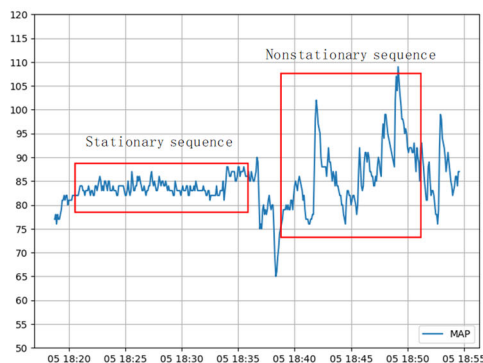
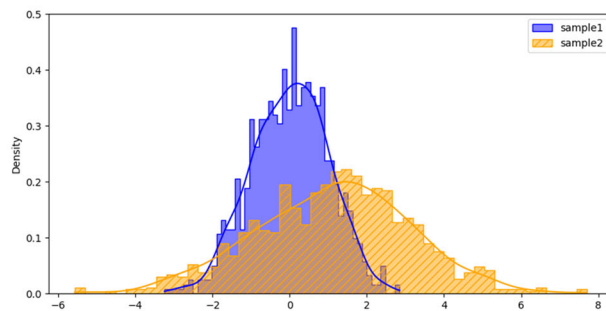


Figure 1 Distributional Variability



These issues result in blood pressure prediction models potentially struggling to capture and adapt to individual variability, non-stationarity, and dynamic changes during surgery, affecting the model's generalization capability and accuracy. In response, traditional statistical methods like ARIMA[3] use differencing to stabilize time series and mitigate distribution shifts. For deep learning models, dealing with non-stationarity-induced distribution changes is more complex[4]. Standardization methods are commonly used as preprocessing in deep learning time series forecasting models, including min-max and Z-score normalization[5]. However, blood pressure data exhibits significant variability between individuals, and methods like Min-Max Normalization and Z-

score Normalization typically require the use of minimum, maximum, mean and standard deviation values computed over the entire dataset. Relying solely on these methods overlooks the differences between individuals and is not suitable for dynamic situations. Subsequent normalization techniques have been proposed, like Adaptive Norm by Ogasawara et al.[6], which uses global statistics to perform Z-score normalization. But this still relies on a static and global statistical description of data, insufficient for capturing the dynamism and complexity of time series. To address this, research has attempted to dynamically adjust normalization parameters to adapt to data changes. DAIN by Passalis et al[7] uses a simple nonlinear neural network to adaptively

normalize based on the observed training distribution. Du et al[8] introduced adaptive RNNs for dealing with distribution shift in time series. Recently, RevIN[9] introduced two-stage instance normalization, transforming both model inputs and outputs to reduce variations within each sequence, aiming to standardize features while preserving useful information. These methods focus on adaptive normalization for inputs and outputs, but they do not consider how the model internally handles the normalized data.

In past blood pressure prediction tasks[10,11], normalization was performed during preprocessing by calculating the dataset's variance and mean. The aim was to minimize distribution differences between samples and mitigate the non-stationarity of the original time series, thereby enhancing the stability of model predictions. Experimental validation has shown that this normalization process can, to some extent, improve the predictability of models in intraoperative blood pressure prediction tasks. However, it also overlooks the unique non-stationary information within samples. To overcome the issue of distribution differences while effectively utilizing the dynamic information in physiological sequences, this paper develops Distribution Adaptive Attention Transformer (DABPformer) for Blood Pressure Prediction based on a Non-stationary Transformer[12] model which integrates the statistical characteristics of input sequence blocks into the attention mechanism, enabling it to adjust weight distribution according to the statistical properties of the data.

The experimental results demonstrate that the distribution

adaptation strategy proposed in this paper performs exceptionally well in intraoperative blood pressure prediction tasks. Under prediction lengths of 5, 10, and 15 minutes, the actual blood pressure Mean Absolute Error (MAE) was 3.221, 3.636, and 3.846, respectively. Compared to the original baseline, the Root Mean Square Error (RMSE) decreased by 4.07%, 4.60%, 4.47%, and 3.846%, respectively, indicating that the model can predict intraoperative blood pressure changes more accurately.

2. Methods

2.1. Problem Definition

The main goal of intraoperative blood pressure prediction tasks is to forecast the continuous values of blood pressure for a future period based on a segment of historical physiological data. Specifically, suppose at time t there are N historical physiological sequences \mathbf{X}^t of length L_x serving as input, where $\mathbf{X}^t = \{\mathbf{x}_1^t, \dots, \mathbf{x}_{L_x}^t \mid \mathbf{x}_i \in \mathbb{R}^N\}$. Under a prediction span of length L_y , the model outputs $\mathbf{Y}^t = \{\mathbf{y}_1^t, \dots, \mathbf{y}_{L_y}^t \mid \mathbf{y}_i \in \mathbb{R}^p\}$, with $p = 3$ representing the three blood pressure measures to predict: Systolic Blood Pressure(SBP), Mean Arterial Pressure (MAP), Diastolic Blood Pressure (DBP).

2.2. Model Structure

The blood pressure prediction model constructed in this paper with distribution adaptive attention is illustrated in the figure 2:

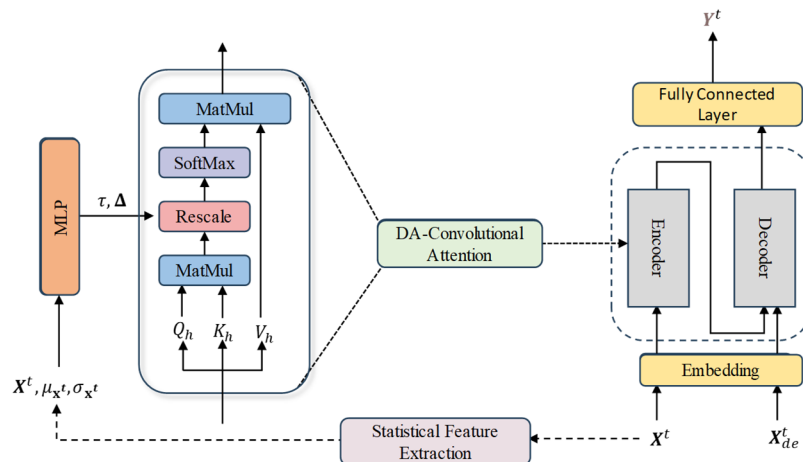


Figure 2 Distribution-Adaptive Attention Architecture

The architecture adopts an informer-style[13] encoder-decoder which mainly includes convolutional attention layers, normalization layers, and self-attention distillation layers. Initially, the input sequence \mathbf{X}^t is a sample undergone global normalization, which serves as the input to the encoder. The input passes through two attention layers. The attention layer employs a convolutional attention mechanism[14], acting as a feature extraction layer. To further reduce the complexity of the attention mechanism, the Informer's *ProbSparse* Self-attention is employed to extract significant queries and keys from the convolutional attention mechanism, decreasing the spatiotemporal complexity of the attention mechanism to $O(L \log L)$. Finally, through the self-attention distillation layer, redundancy in the feature maps is eliminated, further reducing the temporal dimension of the input and allowing the model to focus more on the dominant features.

The input to the decoder \mathbf{X}_{de}^t , serves as a representation of the predicted target blood pressure series. The input goes through two multi-head convolutional attention layers, one of which is a cross-attention layer that receives information from the encoder. This cross-attention layer correlates the current time step representation of the decoder with the encoder's output representation, thereby capturing information related to the target blood pressure prediction in the source sequence. Finally, a fully connected layer is used to output the predicted target sequence \mathbf{Y}^t all at once. This approach avoids the error accumulation that can occur with the original decoder architecture's step-by-step inference and is also used to alleviate the decrease in inference speed that can occur with long sequence predictions.

2.3. Convolutional Attention

During intraoperative blood pressure prediction tasks, the evolution of blood pressure sequence patterns may be significantly influenced by various events, such as drug injections or changes in the external environment[15]. Therefore, whether an observation point is considered an anomaly, a change point, or part of a pattern largely depends on its surrounding context. The segments of the blood pressure sequence extracted from the data are shown in Figure 3, where there is an evident local similarity in the blood pressure sequence, that is, similar rhythms exist at different times.

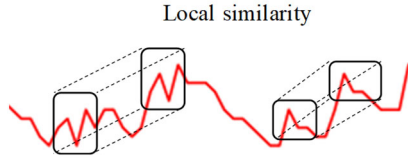


Figure 3 Blood pressure segment

However, in the traditional Transformer[16] self-attention layer, the similarity between queries and keys is calculated based on their pointwise numerical values, which does not fully utilize information such as the shape of the local context.

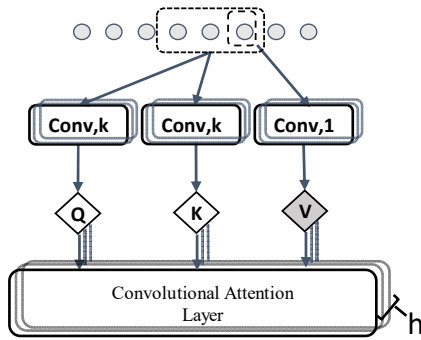


Figure 4 Schematic of Convolutional Attention

To introduce local information from the sequence to better capture temporal correlations, trends, and anomalies, a convolutional attention mechanism is constructed as shown in the figure 4. In the convolutional attention layer[14], the multi-head attention sub-layer transforms the hidden layer $H \in \mathbb{R}^{L \times d_{model}}$ into h different query matrices Q_h , key matrices K_h , and value matrices V_h , Q_h, K_h use a one-dimensional convolution with kernel size k and stride 1 (with zero-padding as appropriate to maintain dimension size), that is, $Q_h = HW_{conv(k)}^q$, $K_h = HW_{conv(k)}^k$, while V_h is constructed using a convolution kernel size of 1, $V_h = HW_{conv(1)}^v$, which is equivalent to a linear transformation. W represents the parameter matrix, with $W_{conv(k)}^q, W_{conv(k)}^k \in \mathbb{R}^{d_{model} \times d_k}$ and $W_{conv(1)}^v \in \mathbb{R}^{d_{model} \times d_v}$. The Q_h and K_h generated in this way can calculate similarity through local context information rather than pointwise values, thus better capturing local features in blood pressure.

Secondly, when calculating attention on sequences, only some dot product pairs contribute to the main attention, while others produce negligible attention. Therefore, *ProbSparse* Self-attention is adopted to extract the

important ones, with the formula as follows:

$$O_h = ProbSparse Attention(Q_h, K_h, V_h) = softmax\left(\frac{\overline{Q_h K_h^T}}{\sqrt{d_k}}\right) \cdot V_h \quad (1)$$

The *ProbSparse* Self-attention mechanism, after processing through the convolutional attention layer, can result in feature maps with redundant values. Therefore, a Self-attention Distillation layer is employed to extract dominant features, and at the same time, it reduces the temporal dimension of the input, which decreases the amount of memory used. Its definition is as follows:

$$X_{j+1}^t = MaxPool(ELU(Conv1d([X_j^t]_{cs}))) \quad (2)$$

Here, $[\cdot]_{cs}$ denotes the operations of the convolutional attention layer. "ELU"[17] is the activation function, which applies a one-dimensional convolution and max pooling operation to the input feature map to focus the feature map.

2.4. Distribution Adaptive Strategy

In blood pressure prediction tasks, it is common to transform all sequence data into a format with uniform mean and standard deviation, which helps reduce the impact of different scales and dimensions on the model, making it easier for the model to learn the interactions between various features[18]. However, due to the distributional differences in physiological sequences, the simple application of global normalization may weaken the dynamic characteristics of blood pressure data. For instance, during surgery, rapid changes in blood pressure are often critical indicators of changes in a patient's condition. Global normalization can reduce the model's sensitivity to these key events, potentially leading to attention weight distributions that are too uniform or focused on noise within the normalized data. This can result in the model producing overly smooth outputs that lack event responsiveness. Therefore, a method capable of capturing the unique dynamic characteristics of blood pressure data is needed.

To address this problem, the paper constructs a distribution adaptation strategy as shown in Figure 2, which combines the convolutional sparse attention mechanism in its encoder with non-stationary attention. For the input sequence of the encoder X^t , it calculates its statistical features μ_{x^t} and σ_{x^t} , which respectively represent the mean and variance of the input time window at moment t . Compared to the original non-stationary attention architecture, the distribution adaptation architecture removes the normalization and denormalization modules for the input. Instead, it directly uses the globally normalized input, allowing it to perform calculations and adjustments of mean and standard deviation based on the data within each fragment. To be able to learn the unique distribution information from local fragments, as shown in equation (3),(4):

$$\log \tau = MLP(\sigma_{x^t}, X^t) \quad (3)$$

$$\Delta = MLP(\mu_{x^t}, X^t) \quad (4)$$

a multilayer perceptron (MLP) module is used to learn the distribution adaptation factors τ and Δ from X^t . τ is a positive scalar factor corresponding to the learned result of

local variance, and Δ is a shift vector corresponding to the learned result of local mean.

The final adjustment of the attention weight coefficients is done in a weighted manner, as shown in equation (5).

$$\text{Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}, \tau, \Delta) = \text{Softmax}\left(\frac{\tau \mathbf{Q} \mathbf{K}^T + \mathbf{1} \Delta^T}{\sqrt{d_k}}\right) \mathbf{V} \quad (5)$$

Here, Attn represents the convolutional attention mechanism, and $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ stand for the input queries, keys, and values, respectively, while d_k denotes the dimensionality of the vectors. In this manner, the attention weights within each sequence block can be independently adjusted based on their respective statistical values. For instance, if a block has a high variance, indicating that the data points are more dispersed, the model should focus more on those points that significantly differ from the average. This is achieved by increasing the scaling factor τ of the attention weights. When there is a change in the mean, it signifies a shift in the overall trend of the sequence. By adjusting an offset Δ , the model takes into account this change in trend when calculating attention weights. With this approach, the model can not only take advantage of the stability of the globally normalized data but also adapt to the potential non-stationary characteristics within each input window, enhancing its ability to learn the inherent distribution patterns of non-stationary sequences.

2.5. Generative Prediction

In the context of long-term blood pressure prediction, a generative decoder design has been adopted, aiming to address the issues of decaying velocity and error accumulation present in traditional prediction methods[19]. The detailed architecture of the generative decoder is shown in the figure 5:

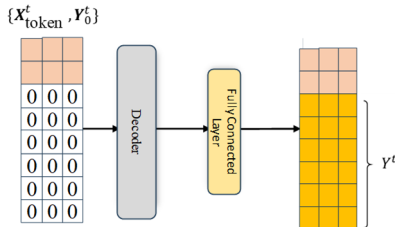


Figure 5 Generative Prediction Schematic

The definition of the decoder's input X_{de}^t is as follows:

$$X_{de}^t = \text{PE}(\text{Concat}(X_{\text{token}}^t, Y_0^t)) \in \mathbb{R}^{(L_{\text{token}} + L_y) \times d_{\text{model}}} \quad (6)$$

where PE represents positional encoding. Here, $X_{\text{token}}^t \in \mathbb{R}^{L_{\text{token}} \times N}$ indicates that output predictions are guided by the starting sequence. It represents using a sequence slice of length L_{token} prior to the predicted target blood pressure sequence as a starting marker for the predictions. $Y_0^t \in \mathbb{R}^{L_y \times p}$ is a placeholder for the target sequence to be predicted (set to 0). The generative decoder directly generates the target blood pressure sequence through a fully connected layer, avoiding the error accumulation associated with the step-by-step inference of traditional autoregressive decoders.

2.6. Loss Function

When predicting the target sequence, the Mean Square Error (MSE) is chosen as the loss function, where Y_i and \hat{Y}_i represent the actual value and the predicted value,

respectively. The loss function is defined as follows:

$$\text{loss} = \frac{1}{L_y} \sum_{i=1}^{L_y} (Y_i^t - \hat{Y}_i^t)^2 \quad (7)$$

3. Experiment

3.1. Dataset

The data description is presented in Table 1. The dataset originates from intraoperative physiological monitoring data of all hepato-pancreato-biliary surgery patients collected via monitors at a hospital, comprising a total of 150 surgeries. To ensure the universality of the research, only 11 common physiological indicators that are typically monitored during surgeries were selected. These indicators cover the majority of those used in current intraoperative hypotension prediction research and all data were collected at a frequency of every 5 seconds. The study focuses on non-invasive blood pressure as the prediction target.

Table 1 Data Description

Data Type	Data Content
Target Data	Diastolic Blood Pressure (DBP)
	Mean Arterial Pressure (MAP)
	Systolic Blood Pressure (SBP)
Related Data	Heart Rate
	Pulse Rate
	Oxygen Saturation
	Perfusion Index
	Inspired Oxygen Concentration
	End-tidal Carbon Dioxide Concentration
	End-tidal Oxygen Concentration
	Tidal End Carbon Dioxide Concentration
	Diastolic Blood Pressure
	Mean Arterial Pressure
	Systolic Blood Pressure

A common sliding window technique is used to construct samples by dividing continuous time data into fixed-length data blocks to create a sample set using a sliding window. As the sliding window moves backward, each new data block is obtained by shifting the previous block back by one step length, allowing each data block to contain data points from before and after in time, ensuring a certain degree of overlap and continuity between data blocks. The sliding window consists of two parts: the retrospective window and the prediction window. The retrospective window is used for input and includes historical physiological sequence data for prediction, while the prediction window corresponds to the predicted label, containing observed values of the current physiological sequence. Around 50,000 samples were divided in total.

3.2. Metrics

To evaluate predictive performance, the Mean Absolute Error (MAE) and the Root Mean Square Error (RMSE) are used to measure the error of predicted values. MAE (see Equation (9)) is used to assess the absolute value of the error between the predicted blood pressure values by the model and the true values, while RMSE (see Equation (8)) imposes a greater penalty on outlier values, both units being mmHg. The smaller these two metrics are, the better. Additionally, the Correlation Coefficient (CORR) (see Equation (10)) is used to evaluate the linear trend correlation between the actual blood pressure sequence and the predicted blood pressure

sequence. The value of CORR ranges from -1 to 1, representing negative and positive correlation between sequences, respectively. The closer the CORR is to 1, the stronger the correlation.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (8)$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (9)$$

$$CORR = \frac{\sum_{i=1}^n (y_i - \hat{y}_i)(\hat{y}_i - \text{mean}(\hat{y}_i))}{\sqrt{\sum_{i=1}^n (y_i - \text{mean}(y_i))^2} \cdot \sqrt{\sum_{i=1}^n (\hat{y}_i - \text{mean}(\hat{y}_i))^2}} \quad (10)$$

3.3. Experimental Settings

In the task of intraoperative blood pressure prediction, the experiment is set up to predict blood pressure for 5, 10, and 15 minutes into the future, corresponding to prediction time steps of 60, 120, and 180, respectively. These three prediction ranges are compared in the experiment. The 5-minute mark is the benchmark for most hypotension prediction tasks, allowing clinical doctors to take timely measures based on the prediction results within this window. The 10 and 15-minute marks are used to evaluate the model's effectiveness for longer-sequence predictions within an acceptable error range, providing more decision-making time for clinical practice. All experiments use 5-fold cross-validation, with the final results being the average.

In terms of model parameter settings, the model training uses an Adam optimizer with an initial learning rate of 10^{-4} , a batch size of 64, and a maximum number of training epochs set at 100, with an early stopping mechanism in place. The DA-BPformer model parameters, adjusted through experimentation, are set with a convolutional kernel size of 3 for the convolutional attention, a model hidden layer

dimension of 512, encoder layers set to 2, decoder layers set to 1, and a dropout rate of 0.05.

3.4. Experimental Results and Analysis

3.4.1. Comparison with Standardization Methods

To visually demonstrate the effectiveness of the distribution adaptation strategy in blood pressure prediction tasks, the study designed the following comparative experiments. Firstly, the Informer and the BPformer from this article were selected as the baseline models. In the comparative experiments, the following notations were used to represent different model configurations: "--" indicates the performance data of the baseline model from the model comparison experiments in the previous chapter; "-Z" refers to the method where the model is trained directly with raw data without any global standardization; "+NS" denotes that the model employs a non-stationary attention architecture[12]; "+DA" signifies that the model incorporates the distribution adaptive attention architecture proposed in this paper; "+Re" represents the use of Revin[9] for adaptive normalization at the input and output stages. With such comparative setups, it is possible to accurately evaluate the effectiveness of the distribution adaptive architecture in blood pressure prediction tasks. It also showcases the specific impacts of different data processing methods and architectural choices on the predictive performance. The final results are obtained by averaging the prediction errors for DBP, SBP and MAP.

The experimental results from Table 2 indicate that both Informer and BPformer saw significant performance improvements when integrated with the distribution adaptive (+DA) architecture, with the DA-BPformer from this paper showing enhancements across all evaluation metrics compared to the original baseline. Specifically, for prediction lengths of 5, 10, and 15 minutes, the MAE reduced by 1.53%, 3.17%, and 6.92% respectively, while the RMSE decreased by 4.07%, 4.60%, and 4.47%. Given that RMSE is a metric

Table 2. Comparative Experimental Results of Different Methods

Model Metric	5 minutes			10 minutes			15 minutes			
	MAE	RMSE	CORR	MAE	RMSE	CORR	MAE	RMSE	CORR	
Informer	--	3.439	6.476	0.881	4.092	7.222	0.844	4.511	7.732	0.814
	-Z	5.597	9.144	0.741	7.048	10.719	0.612	7.066	10.748	0.608
	+NS ^[12]	3.721	6.601	0.878	4.372	7.354	0.835	4.785	7.912	0.803
	+Re ^[9]	4.281	7.159	0.855	4.567	7.590	0.831	4.672	7.647	0.816
	+DA	3.487	6.244	0.891	3.929	6.782	0.865	4.127	7.073	0.848
DPformer	--	3.271	6.300	0.886	3.755	6.846	0.860	4.132	7.293	0.836
	-Z	5.596	9.177	0.739	6.792	10.462	0.634	7.154	10.831	0.595
	+NS ^[12]	3.532	6.428	0.884	3.947	6.981	0.858	4.301	7.403	0.831
	+Re ^[9]	4.007	6.846	0.867	4.161	7.118	0.852	4.323	7.238	0.836
	+DA	3.221	5.990	0.898	3.636	6.470	0.876	3.846	6.757	0.860

more sensitive to outliers, its significant reduction compared to MAE suggests that the distribution adaptive architecture is capable of identifying and handling non-stationary events.

Furthermore, the paper also examined the performance of the models on non-standardized data (-Z), where the models were applied directly to the raw, unstandardized data for training. The results indicated a significant drop in performance for both Informer and BPformer, further emphasizing the importance of data standardization for model performance in blood pressure prediction tasks.

In the +NS experiment group, the results showed that the performance of the distribution adaptive strategy in this paper

was superior to that of the original non-stationary attention architecture. The performance difference likely stems from the way distribution information is extracted. In the non-stationary attention architecture, statistical information is directly extracted from the raw sequence to integrate non-stationary features before normalization is performed. The issue with this approach is that if the raw data contains significant disparities or outliers, the early-extracted statistical information might cause the attention mechanism to overemphasize these anomalous data points, thus overlooking other important information. In contrast, the distribution adaptive architecture in this paper takes a

different approach: it first globally standardizes the data and then extracts distribution information for attention weighting. The normalized data can effectively reduce the impact of disparities and outliers, allowing the attention mechanism to focus more evenly across all features. However, in the +Re experiments, despite using the reversible adaptive normalization module RevIN at the input and output stages, its effects were not as good as the aforementioned approaches, further indicating that performing adaptive normalization externally to the model has limited benefits for enhancing the model's ability to recognize and handle non-stationary information.

3.4.2. Ablation of Adaptive Factors

To verify the impact of the adaptation factors τ and Δ on the prediction results, the following ablation experiments were designed: $+\Delta$ indicates the integration of only the Δ factor, meaning only the data's offset adjustment is considered, while $+\tau$ represents the integration of only the τ factor, meaning only the data's scale adjustment is considered. Through such experimental settings, it is possible to independently evaluate the contribution of each factor to the overall prediction performance when acting alone.

The experimental results from Table 3 indicate the inclusion of both the scaling factor τ and the shift vector Δ significantly enhances the model's predictive performance. It is noticeable that the integration of τ alone slightly outperforms Δ , which may be due to the fact that changes in local variance could contain important clues about physiological state changes. By capturing the nuances of local variance fluctuations in physiological sequences, τ endows the model with vital insights into volatility, which could be particularly pivotal for accurate blood pressure forecasting. When Δ and τ are applied together, the model's performance improves on short to medium-term predictions of 5 and 10 minutes, showing the advantage of capturing local statistical characteristics. However, when the prediction span

is extended to long-term predictions of 15 minutes, the contribution of the combined $\Delta + \tau$ to performance improvement seems to become marginal. In such cases, the performance of the $\Delta + \tau$ model configuration is similar to when using $+\tau$ alone, suggesting that over longer prediction intervals, the dynamics and distribution of physiological sequences may undergo more complex changes. The shift factor Δ , which primarily adjusts the local mean of the data, may not be sufficient to capture these complex changes. To better predict the complex distribution changes in long-term blood pressure data, future research may need to explore more advanced model structures or strategies.

4. Conclusion

Addressing the significant variability in physiological data distributions among different patients, as well as the issue of distributional shifts in physiological sequences for blood pressure prediction tasks, this study developed a distribution-adaptive strategy. The approach begins with global data standardization to minimize patient-to-patient differences, followed by an adaptive learning mechanism that adjusts the focus of the attention mechanism based on the mean and variance within the input window, allowing the model to concentrate more on essential non-stationary information. A series of comparative experiments and ablation studies confirmed the effectiveness of this strategy in enhancing the attention-based model's ability to process non-stationary physiological data, and demonstrated the importance of distribution-adaptive factors in improving blood pressure prediction accuracy. These results offer new perspectives and methodologies for the field of medical time series, holding significant theoretical and practical application value. Future work will focus on collecting more data and conducting experiments and clinical validations to more comprehensively assess the model's performance in clinical practice.

Table 3 Ablation Experiment Results

Model Metric	5 minutes			10 minutes			15 minutes		
	MAE	RMSE	CORR	MAE	RMSE	CORR	MAE	RMSE	CORR
--	3.271	6.300	0.886	3.755	6.846	0.860	4.132	7.293	0.836
$+\Delta$	3.244	6.015	0.897	3.645	6.484	0.875	3.861	6.775	0.860
$+\tau$	3.254	6.028	0.897	3.638	6.489	0.876	3.844	6.758	0.860
$\Delta + \tau$	3.221	5.990	0.898	3.636	6.470	0.876	3.846	6.757	0.860

References

- [1] van der Ven W H, Veelo D P, Wijnberge M, et al. One of the first validations of an artificial intelligence algorithm for clinical use: The impact on intraoperative hypotension prediction and clinical decision-making[J]. *Surgery*, 2021, 169(6): 1300-1303.
- [2] Parati G, Ochoa J E, Lombardi C, et al. Assessment and management of blood-pressure variability[J]. *Nature Reviews Cardiology*, 2013, 10(3): 143-155.
- [3] Ariyo A A, Adewumi A O, Ayo C K. Stock price prediction using the ARIMA model[C]//2014 UKSim-AMSS 16th international conference on computer modelling and simulation. IEEE, 2014: 106-112.
- [4] Eesa A S, Arabo W K. A normalization methods for backpropagation: a comparative study[J]. *Science Journal of University of Zakho*, 2017, 5(4): 319-323.
- [5] Patro S, Sahu K. Normalization: A preprocessing stage[J]. arXiv preprint arXiv:1503.06462, 2015.
- [6] Ogasawara E, Martinez L C, De Oliveira D, et al. Adaptive normalization: A novel data normalization approach for non-stationary time series[C]//The 2010 International Joint Conference on Neural Networks (IJCNN). IEEE, 2010: 1-8
- [7] Passalis N, Tefas A, Kannianen J, et al. Deep adaptive input normalization for time series forecasting[J]. *IEEE transactions on neural networks and learning systems*, 2019, 31(9): 3760-3765.
- [8] Du Y, Wang J, Feng W, et al. Adarnn: Adaptive learning and forecasting of time series[C]//Proceedings of the 30th ACM international conference on information & knowledge management. 2021: 402-411.
- [9] Kim T, Kim J, Tae Y, et al. Reversible instance normalization for accurate time-series forecasting against distribution shift [C]//International Conference on Learning Representations. 2021.

- [10] Su P, Ding X R, Zhang Y T, et al. Long-term blood pressure prediction with deep recurrent neural networks[C]//2018 IEEE EMBS International conference on biomedical & health informatics (BHI). IEEE, 2018: 323-328.
- [11] Jeong Y S, Kang A R, Jung W, et al. Prediction of blood pressure after induction of anesthesia using deep learning: A feasibility study[J]. Applied Sciences, 2019, 9(23): 5135.
- [12] Liu Y, Wu H, Wang J, et al. Non-stationary transformers: Exploring the stationarity in time series forecasting[J]. Advances in Neural Information Processing Systems, 2022, 35: 9881-9893.
- [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] Li S, Jin X, Xuan Y, et al. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting[J]. Advances in neural information processing systems, 2019, 32.
- [15] Kheng C P, Rahman N H. The use of end-tidal carbon dioxide monitoring in patients with hypotension in the emergency department[J]. International journal of emergency medicine, 2012, 5(1): 1-7.
- [16] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.
- [17] Clevert D A, Unterthiner T, Hochreiter S. Fast and accurate deep network learning by exponential linear units (elus)[J]. arXiv preprint arXiv:1511.07289, 2015.
- [18] Bhanja S, Das A. Impact of data normalization on deep neural network for time series forecasting[J]. arXiv preprint arXiv:1812.05519, 2018.
- [19] Ing C K. Accumulated prediction errors, information criteria and optimal forecasting for autoregressive time series[J]. 2007.