Adaptive Meta-Learning Probabilistic Inference Framework for Long Sequence Prediction

Jianping Zhu[1], **Xin Guo**[1], **Yang Chen**[2*], **Yao Yang**[3], **Wenbo Li**[3], **Bo Jin**[1*], **Fei Wu**[2]

[1]Dalian University of Technology
[2]Zhejiang University
[3]Zhejiang Lab
{zhujp, guoxinguo}@mail.dlut.edu.cn, jinbo@dlut.edu.cn, {ychen2014, wufei}@zju.edu.cn, {yangyao, liwenbo}@zhejianglab.com

Abstract

Long sequence prediction has broad and significant application value in fields such as finance, wind power, and weather. However, the complex long-term dependencies of long sequence data and the potential domain shift problems limit the effectiveness of traditional models in practical scenarios. To this end, we propose an Adaptive Meta-Learning Probabilistic Inference Framework (AMPIF) based on sequence decomposition, which can effectively enhance the long sequence prediction ability of various basic models. Specifically, first, we decouple complex sequences into seasonal and trend components through a frequency domain decomposition module. Then, we design an adaptive meta-learning task construction strategy, which divides the seasonal and trend components into different tasks through a clustering-matching approach. Finally, we design a dual-stream amortized network (ST-DAN) to capture shared information between seasonal-trend tasks and use the support set to generate task-specific parameters for rapid generalization learning on the query set. We conducted extensive experiments on six datasets, including wind power and finance scenarios, and the results show that our method significantly outperforms baseline methods in prediction accuracy, interpretability, and algorithm stability and can effectively enhance the long sequence prediction capabilities of base models. The source code is publicly available at https://github.com/Zhu-JP/AMPIF.

Introduction

Long sequence prediction has significant and widespread demand in many fields. For example, in the field of finance, long-sequence prediction can provide adequate decision-making references for investors and risk warning indicators for regulatory authorities (Zhao and Chen 2022). In the field of wind power prediction, long sequence prediction can give adequate support for the operation and maintenance of wind farms (Zhou et al. 2021). Therefore, accurate long-sequence prediction has become a key prerequisite for the above applications. However, due to the highly nonlinear, non-stationary, and uncertain characteristics often present in long sequence prediction tasks, existing research is still limited to fields with obvious periodic features and weak noise disturbances such as climate, transportation, and wind power, and cannot be effectively transferred to more complex long sequence prediction scenarios. This poses a critical challenge to the generalization ability of basic long-sequence prediction algorithms.

Firstly, the time dependence of long sequences exhibits complex entangled patterns of trends and seasonality (Wu et al. 2021). Currently, classic long sequence prediction algorithms include models based on Transformer (Vaswani et al. 2017), such as Informer (Zhou et al. 2021), Reformer (Kitaev, Kaiser, and Levskaya 2020), and Preformer (Du, Su, and Wei 2022), methods based on seasonal and trend term representation learning, such as COST (Woo et al. 2022a) and LaST (Wang et al. 2022), as well as methods that combine sequence decomposition with the Transformer architecture, such as Autoformer (Wu et al. 2021), FEDformer (Zhou et al. 2022), and ETSformer (Woo et al. 2022b). These models have demonstrated powerful predictive capabilities on datasets with strong seasonal regularities and weak noise disturbances. However, existing research often focuses more on seasonal-term modeling while neglecting the impact of long-term trends, which is a limitation of this type of research. Guided by the idea of sequence decomposition, we believe that decoupling complex sequences and modeling seasonal and trend terms separately can better improve the model's generalization ability in long sequence prediction tasks.

Secondly, long sequence prediction tasks in complex application scenarios are often accompanied by domain distribution shift problems (Jin et al. 2022). For the model to adapt to changing environments, meta-learning algorithms that capture shared knowledge between tasks to achieve rapid and general learning have become a viable solution. Existing meta-learning strategies often assume that meta-learning tasks have been completed—for example, MAML (Finn, Abbeel, and Levine 2017), ATS (Yao et al. 2021), VC-BML (Zhang et al. 2021), etc. However, in practical scenarios, most datasets are difficult to complete a reasonable task division based solely on prior knowledge, which also limits the transferability of these methods to other data scenarios. For example, in more complex financial scenarios, VML (Liu et al. 2022), DPML (Chen et al. 2023), and MASSER (Zhan et al. 2022) meta-learning-based stock prediction methods treat each stock's prediction as a meta-learning task. However,

*Corresponding authors.

there are often significant differences between the historical volatility of a stock and its future volatility. Therefore this way of dividing tasks through prior knowledge often introduces additional bias. Although the application of meta-learning methods has made some progress, they are not suitable for long sequence prediction tasks.

To address all these challenges, we propose an Adaptive Meta-Learning Probabilistic Inference Framework (AMPIF) based on sequence decomposition for efficient adaptive learning of complex dynamics in long sequences. Specifically, we first divide a set of different samples into a support set and a query set and use a frequency domain decomposition module to decompose each sequence into two parts: seasonality and trendiness, while performing denoising to reduce the interference of noise on long-term predictions. Then we propose a global representation learning method (ST-GRL) that considers both input sequences and their corresponding long sequence labels to learn global representations for seasonal and trend terms separately and divide them into different tasks based on clustering-matching patterns. Finally, based on the VERSA (Gordon et al. 2018) meta-learning framework, we further design a dual-stream amortized network (ST-DAN) which takes as input feature vectors encoded by Season Encoder and Trend Encoder, respectively and uses support set to generate task-specific parameters for rapid generalization learning on query set.

In summary, the main contributions of this paper are as follows:

- We propose an Adaptive Meta-Learning Probabilistic Inference Framework (AMPIF) based on sequence decomposition, which effectively enhances base models' long sequence prediction capabilities.
- We propose a novel meta-learning task construction mode. It leverages a global representation, considering both input sequences and their long sequence labels via mutual information constraints, to minimize domain distribution shift in long sequence prediction. We innovatively employ sequence clustering-matching, using similar fluctuation characteristics from support set samples, to guide query set predictions.
- AMPIF can effectively enhance the long-sequence prediction capabilities of base models. In addition to our constructed A-share and Cross-Market datasets, we also conducted experiments on publicly available ETT datasets (Zhou et al. 2021). Extensive experimental results show that AMPIF significantly outperforms baseline methods in terms of prediction accuracy and algorithm stability.

Problem

For a given batch of time series $\mathcal{X} = \{X, \tilde{X}\} \in \mathbb{R}^{B\times T_x}$, we partition it and its corresponding labels into N support sets $D = \{(x_n, y_n)\}_{n=1}^{N}$ and M query sets $\tilde{D} = \{(\tilde{x}_m, \tilde{y}_m)\}_{m=1}^{M}$, where $x_n, \tilde{x}_m \in \mathbb{R}^{T_x}$ represent the input sequences and $y_n, \tilde{y}_m \in \mathbb{R}^{T_y}$ represent the corresponding labels. T_x and T_y denote the lengths of the input and prediction sequences, respectively, and B represents the number of samples in a batch, i.e., $B = N + M$.

In the meta-learning paradigm, we capture shared patterns (or parameters) through knowledge transfer from similar samples to ultimately model the conditional distribution $p(\tilde{Y}|D, \tilde{X})$. The architecture of AMPIF is shown in Fig.(1). We denote these shared task parameters as, $\theta = \{\theta_S, \theta_T, \theta_{ST\text{-}GRL}, \theta'\}$, where θ_S and θ_T represent the seasonal encoder and trend encoder respectively, $\theta_{ST\text{-}GRL}$ represents the seasonal trend global representation extractor, and θ' represents the decoder. Specifically, this paper first decomposes the stock time series into seasonality component $\mathcal{X}_\mathcal{S}$ and trend component $\mathcal{X}_\mathcal{T}$, which are encoded by θ_S and θ_T, respectively. Then, under a clustering-matching pattern, the sequences encoded by $\theta_{ST\text{-}GRL}$ are divided into K_S and K_T meta-learning tasks, respectively. Next, based on the task parameters $\{\psi_{k_S}\}_{k_S=1}^{K_S}$ and $\{\psi_{k_T}\}_{k_T=1}^{K_T}$ generated from the support set in each task, we learn the seasonality representation $\tilde{H}_S^\psi = \{\tilde{h}_{k_S}^\psi\}_{k_S=1}^{K_S}$ and trend representation $\tilde{H}_T^\psi = \{\tilde{h}_{k_T}^\psi\}_{k_T=1}^{K_T}$ of the query set. Finally, under θ', we fuse and decode the seasonality and trend components to complete the final prediction. That is:

$$p(\tilde{Y}|D, \tilde{X}) = p(\tilde{Y}|\tilde{H}_S^\psi, \tilde{H}_T^\psi, \theta') \prod_{k_{S/T}=1}^{K_{S/T}} \prod_{m=1}^{M_{k_{S/T}}} p(\tilde{h}_{k_{S/T}}^\psi | \tilde{x}_{k_{S/T}}^m, \psi_{k_{S/T}}, \theta_{S/T}) p(\psi_{k_{S/T}} | D_{k_{S/T}}, \theta_{S/T}), \tag{1}$$

where $D_{k_S} = \{x_{S,n}, y_{S,n}\}_{n=1}^{N_{k_S}}$ and $D_{k_T} = \{x_{T,n}, y_{T,n}\}_{n=1}^{N_{k_T}}$ represent the sample sets of the support set for tasks k_S and k_T, respectively. It is worth noting that we also decomposed the label information into a sequence to fit the complex temporal dynamics of the task. In the scenario of sequence decoupling, a batch of samples can theoretically be divided into $K = K_S \times K_T$ tasks, indicating that after sequence decomposition, a sample has K_S possible task assignments for the seasonal components and K_T possible task assignments for the trend components.

Methodology

In this section, we provide a detailed description of the AMPIF framework. As mentioned earlier, long sequence prediction tasks are limited by domain shift problems, making it difficult to transfer to more complex application scenarios. To effectively address this issue, we first use a frequency domain decomposition module to decompose the original sequence into seasonal and trend terms for separate modeling. Then, we design a global representation learning module (ST-GRL) and adaptively construct meta-learning tasks through clustering-matching. Finally, we extract shared and specific information between seasonal-trend tasks through a dual-stream amortized network (ST-DAN) and generate corresponding task parameters to achieve accurate prediction.

Frequency Decomposition

Since long time series typically exhibit long-term trends and short-term periodic fluctuations, decomposing time series can help the model effectively capture its internal complex

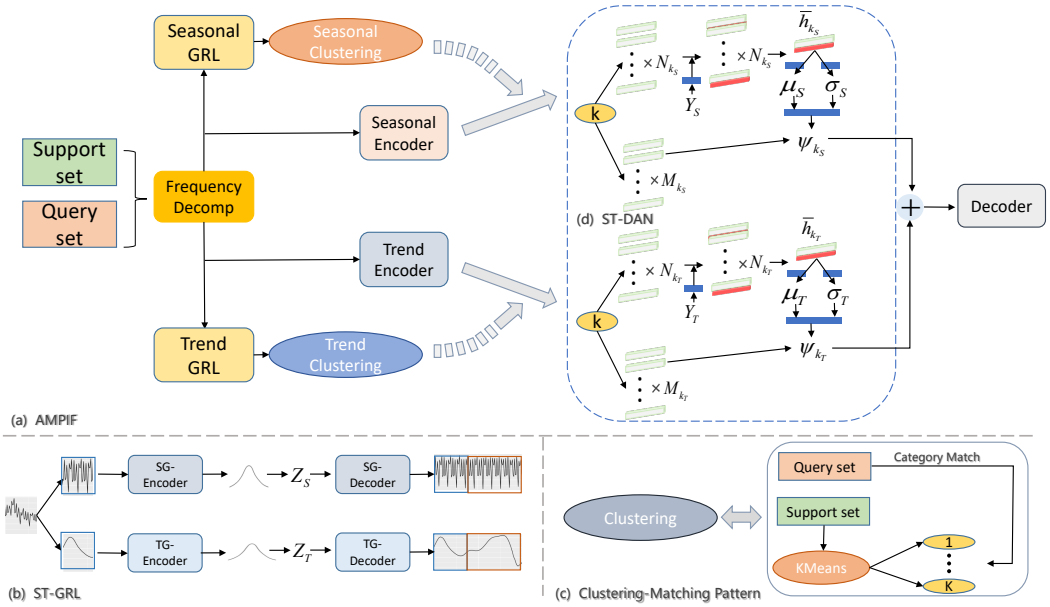

Figure 1: Overview of proposed AMPIF architecture.

temporal dynamics (Lv et al. 2022). Unlike traditional methods that extract trend terms through fixed-window moving averages, we use a frequency-domain-based method to decompose time series. Specifically, we map the sequence to the frequency domain and then separate the high-frequency part as the seasonal components $\hat{\mathcal{X}}_S$ and the low-frequency part as the trend components $\mathcal{X}_T$ through frequency domain masking.

$$\hat{\mathcal{X}}_S, \mathcal{X}_T = \mathcal{F}^{-1}(\mathcal{F}(\mathcal{X})[:V], \mathcal{F}(\mathcal{X})[V:]), \tag{2}$$

where $\mathcal{F}$ denotes the FFT and $\mathcal{F}^{-1}$ is its inverse, $\mathcal{F}(\mathcal{X}) \in \mathbb{R}^{B\times I}$, where $I = \lfloor T_x/2 \rfloor + 1$ and in this paper $V = 3$. At this point, the seasonal component $\hat{\mathcal{X}}_S$ usually contains much noise. In long-sequence prediction tasks, the presence of noise often reduces the generalization ability of the model. Therefore, we further reduce noise in the seasonal component by filtering the Top-K frequencies corresponding to the amplitudes to obtain the final seasonal component $\mathcal{X}_S$.

Meta-Learning Task Construction

Meta-learning captures shared knowledge between tasks to achieve rapid prediction of new tasks. This enables it to adapt to more complex task scenarios, such as in financial markets, where when a new stock suddenly emerges, it can capture shared patterns through knowledge transfer from similar stocks within the same task to predict the new stock rapidly (Chang et al. 2021; Chen et al. 2023). In this paper, we adopt a clustering-matching pattern to ensure each task has samples with known labels as a support set. Specifically, we first divide a batch of samples into a support set and a query set. Then, we cluster the support set to form different tasks. Finally, we calculate the cosine similarity between each sample in the query set and each cluster center in the support set and assign the samples in the query set to the category with the highest similarity:

$$\tilde{k}_m = argmax(\cos(\tilde{z}_m, \bar{C})), \tag{3}$$

where $\tilde{k}_m$ and $\tilde{z}_m$ represent the category and feature vector used for clustering of the m-th sample in the query set, respectively, $\bar{C} = \{\bar{c}_1, \cdots, \bar{c}_K\}$ represents the cluster centers of each category in the support set, and $cos()$ denotes the cosine similarity calculation function.

Global Representations Learning

When clustering by extracting global representation, an intuitive approach is constructing a feature extractor to learn the global representation of the input sequence $\mathcal{X}$. However, this method is still subject to interference from domain shift between different time segments in long-sequence prediction tasks, resulting in large differences in label distributions for samples within the same category. Therefore, learning a global representation containing information about the input sequence $\mathcal{X}$ and its corresponding label $\mathcal{Y}$ becomes the key to the problem. In this paper, we reconstruct the constraints of the VAE (Kingma and Welling 2013) by adding mutual information constraints to learn a seasonal global representation Z_S and a trend global representation Z_T that considers both the input sequence and its corresponding long-sequence label.

Theorem 1 *Under the guidance of the sequence decomposition strategy and global representation learning idea, the ELBO has the following factorized form:*

$$\begin{aligned}\mathcal{L}_{ELBO} &= E_{q_{\theta_S^E}(Z_S|\mathcal{X}_S)}[\log p_{\theta_S^D}(L_S|Z_S)] \\ &+ E_{q_{\theta_T^E}(Z_T|\mathcal{X}_S)}[\log p_{\theta_T^D}(L_T|Z_T)] \\ &- KL(q_{\theta_S^E}(Z_S|\mathcal{X}_S)||p(Z_S)) - KL(q_{\theta_T^E}(Z_T|\mathcal{X}_T)||p(Z_T)),\end{aligned} \tag{4}$$

where $L = [\mathcal{X}, \mathcal{Y}]$ and $[\cdot,\cdot]$ denotes the concatenation operation of two variables, θ_S^E and θ_T^E represent the encoders for

the seasonal and trend components in ST-GRL, respectively, both of which are composed of stacked one-dimensional convolutional layers and linear layers. θ_S^D and θ_T^D represent the decoders for the seasonal and trend components in ST-GRL, respectively, composed of a multilayer perceptron. The detailed inference process of the above formula is provided in Appendix A.1.

We assume the input sequence $\mathcal{X}$ contains sufficient information to predict the target sequence $\mathcal{Y}$. The first two terms of Eq.(4) represent the reconstruction loss of the seasonal and trend components, respectively. Unlike traditional VAEs, our reconstruction target is the entire time series containing the desired prediction $\mathcal{X}$, used to learn a representation with the ability to predict future long sequences. The last two terms are constraints on KL divergence. To improve computational efficiency, we use Monte Carlo sampling to estimate KL divergence and assume that the prior distribution follows a standard normal distribution $\mathcal{N}(0, 1)$.

Theorem 2 *Under the guidance of the idea of clustering through learning global representations, the reconstruction loss $\mathcal{L}_{rec}$ is redefined as:*

$$\mathcal{L}_{rec} = \mathcal{L}_{pre\text{-}rec} + \mathcal{L}_{shape\text{-}rec}, \tag{5}$$

$$\mathcal{L}_{pre\text{-}rec} = -\left\|\hat{L}_S - L_S\right\|^2 - \left\|\hat{L}_T - L_T\right\|^2, \tag{6}$$

$$\begin{aligned}\mathcal{L}_{shape\text{-}rec} = &-\tau_A\left\|A(\mathcal{F}(\hat{L}_S)) - A(\mathcal{F}(L_S))\right\|^2 \\ &- \tau_P\left\|P(\mathcal{F}(\hat{L}_S)) - P(\mathcal{F}(L_S))\right\|^2 \\ &+ \frac{\sum_{i=1}^{T_x+T_y-1}(\hat{L}_T^i - \hat{L}_T^{i-1})(L_T^i - L_T^{i-1})}{\sqrt{\sum_{i=1}^{T_x+T_y-1}(\hat{L}_T^i - \hat{L}_T^{i-1})}\sqrt{\sum_{i=1}^{T_x+T_y-1}(L_T^i - L_T^{i-1})}},\end{aligned} \tag{7}$$

where $A(F) = \sqrt{F_r^2 + F_i^2}$ and $P(F) = \arctan(\frac{F_i}{F_r})$ represent the amplitude and phase of the sequence, respectively. After mapping the sequence to the frequency domain through Fourier transform, F_r and F_i represent the corresponding real and imaginary parts, respectively. τ_A and τ_P are adjustable hyperparameters. The reconstruction loss consists of the prediction value and sequence shape reconstruction loss. We elaborate on the specific details in Appendix A.2.

We found that using only ELBO as the loss function may lead to the problem of invalid clustering. That is, when the modeling ability of VAE is insufficient to reduce the distance between the posterior and prior distributions, VAE will sacrifice variational inference and data fitting (Zhao, Song, and Ermon 2017), resulting in the learned representation containing almost no information about the input sequence, thus making the clustering results random and meaningless. To solve this problem, we introduce an additional constraint: maximizing the mutual information between the overall sequence L and the learned representation Z. We provide proof of its information lower bound in Appendix A.3. In summary, we define the loss function for global representation learning as follows:

$$\mathcal{L}_{GRL} = I(L_S, Z_S) + I(L_T, Z_T) - \mathcal{L}_{ELBO}, \tag{8}$$

where $I(\cdot, \cdot)$ represents the mutual information between two representations.

Seasonal-Trend Dual-Stream Amortization Network

Based on the specific meta-learning task assignment under the dual perspective of seasonality and trend, we propose a seasonality-trend dual amortized network (ST-DAN) to capture shared information between seasonal and trend tasks and generate parameters for specific tasks. Specifically, we design a task-shared amortized network that can output parameters for specific tasks or distributions of random inputs through a single forward channel, thus achieving fast and flexible meta-learning. As described in VERSA (Gordon et al. 2018), forming a posterior distribution for specific task parameters $p(\psi_{k_{S/T}}|D_{k_{S/T}}, \theta)$ is usually difficult to achieve. Therefore, we use two amortized networks ϕ_S and ϕ_T to approximately construct the posterior distributions $q_{\phi_S}(\psi_{k_S}|D_{k_S})$ and $q_{\phi_T}(\psi_{k_T}|D_{k_T})$, respectively. In this paper, we use feed-forward neural networks as amortized networks. First, for the sequence of length T_y to be predicted, we concatenate a zero vector of length T_y and the seasonal term along the time dimension as input to the seasonal encoder and concatenate a mean vector of length T_y and the trend term along the time dimension as input to the trend encoder. After encoding by the encoder, we can obtain the seasonal representation $\mathcal{H}_\mathcal{S}$ and the trend representation $\mathcal{H}_\mathcal{T}$. Then, for each task, we concatenate the representation of its support set with the corresponding label information encoded by a linear layer along the feature dimension. An average instance pooling operation is used to ensure that the network can adapt to any number of support set samples. Finally, a linear operation helps generate the mean and variance of a factorized Gaussian distribution, which is used to generate task-specific parameters. Based on this, an approximate posterior predictive distribution can be represented as:

$$\begin{aligned}q_\phi(\tilde{y}_k|D_k) = \iint p(\tilde{y}_k|\psi_{k_S}, \psi_{k_T}) q_{\phi_S}(\psi_{k_S}|D_{k_S}) \\ q_{\phi_T}(\psi_{k_T}|D_{k_T}) d\psi_{k_S} d\psi_{k_T},\end{aligned} \tag{9}$$

where ϕ represents the dual-stream amortized network, which includes ϕ_S and ϕ_T. k_S and k_T represent the samples in task k according to their corresponding seasonal and trend tasks. Using amortized variational inference and neural networks enables fast prediction of the query set without the need for traditional second-order derivatives during training. Further details can be found in VERSA (Gordon et al. 2018). Finally, we use a fusion function $\Psi()$ to update the representation of the query set for a specific task:

$$\begin{aligned}\tilde{h}_{k_S}^{\psi} &= \Psi(\tilde{h}_{k_S}, \psi_{k_S}), \\ \tilde{h}_{k_T}^{\psi} &= \Psi(\tilde{h}_{k_T}, \psi_{k_T}).\end{aligned} \tag{10}$$

End-to-End Stochastic Training

In AMPIF, the task-shared parameters θ and the dual-stream amortization network ϕ are optimized by the query set's loss in different tasks. Given a new task, our model takes a small training dataset as input and outputs the distribution of random inputs ψ for a specific task in a single forward channel, achieving fast and general learning and inference.

We designed a specific loss function for end-to-end stochastic training. For a specific task, we optimize the model by

Methods	Metrics	Input Length: 24					Input Length: 96				
		Horizon					Horizon				
		24	48	168	336	720	24	48	168	336	720
LSTM	MSE	0.495	0.989	1.032	0.990	1.048	0.985	0.994	1.025	1.021	1.193
	MAE	0.543	0.790	0.794	0.788	0.801	0.770	0.780	0.798	0.791	0.809
Autoformer	MSE	0.496	0.496	0.569	0.628	0.763	0.812	0.894	0.803	0.983	0.874
	MAE	0.530	0.537	0.579	0.615	0.677	0.694	0.733	0.700	0.783	0.751
FEDformer	MSE	0.574	0.519	0.533	0.609	0.813	0.823	0.860	0.837	0.872	0.811
	MAE	0.572	0.544	0.559	0.601	0.695	0.693	0.717	0.695	0.730	0.705
LaST	MSE	0.105	0.102	0.277	0.458	0.670	0.199	0.215	0.283	0.533	0.745
	MAE	0.247	0.248	0.396	0.522	0.629	0.341	0.348	0.402	0.551	0.670
AMPIF	MSE	**0.034**	**0.071**	**0.174**	**0.340**	**0.657**	**0.049**	**0.091**	**0.231**	**0.378**	**0.634**
	MAE	**0.136**	**0.190**	**0.303**	**0.427**	**0.600**	**0.161**	**0.223**	**0.353**	**0.452**	**0.591**

Table 1: Performance comparison of long sequence prediction tasks for new stocks in the A-share dataset.

Datasets	Models	22 trading days		66 trading days		264 trading days		528 trading days	
		MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
Cross-Market	LSTM	0.353	0.217	0.840	0.951	0.799	0.907	0.799	0.916
	Autoformer	1.014	1.466	1.355	2.387	1.448	2.771	1.465	2.963
	FEDformer	0.327	0.177	0.534	0.551	0.646	0.644	0.907	1.251
	LaST	0.297	0.153	0.461	0.346	0.676	0.708	0.774	0.895
	AMPIF	**0.214**	**0.096**	**0.314**	**0.196**	**0.445**	**0.353**	**0.475**	**0.418**

Table 2: Performance comparison of long sequence prediction tasks for new market indices in the Cross-Market dataset.

calculating the KL divergence between the approximate posterior predictive distribution and the true distribution. The learning objective is to minimize the expected KL in the task.

$$\begin{aligned}\phi^* &= \underset{\phi}{argmin}\, \mathbb{E}_K[\mathbb{E}_{p(D_k)}[KL[p(\tilde{y}_k|D_k)||q_\phi(\tilde{y}_k|D_k)]]] \\ &= \underset{\phi}{argmax}\, \mathbb{E}_K[\mathbb{E}_{p(\tilde{y}_k|D_k)}[\log \iint p(\tilde{y}_k|\tilde{x}_k, \psi_{k_S}, \psi_{k_T}) \\ &\quad q_{\phi_S}(\psi_{k_S}|D_{k_S}) q_{\phi_T}(\psi_{k_T}|D_{k_T}) d\psi_{k_S} d\psi_{k_T}]].\end{aligned} \tag{11}$$

After introducing the shared parameter θ, we have

$$\begin{aligned}\mathcal{L}(\phi, \theta) &= -\mathbb{E}_K[\mathbb{E}_{p(D_k, \tilde{x}_k, \tilde{y}_k)}[\log q_\phi(\tilde{y}_k|\tilde{x}_k, \theta)]] \\ &= -\mathbb{E}_K[\mathbb{E}_{p(D_k, \tilde{x}_k, \tilde{y}_k)}[\log \iint p(\tilde{y}_k|\tilde{x}_k, \psi_{k_S}, \psi_{k_T}, \theta) \\ &\quad q_{\phi_S}(\psi_{k_S}|D_{k_S}, \theta_S) q_{\phi_T}(\psi_{k_T}|D_{k_T}, \theta_T) d\psi_{k_S} d\psi_{k_T}]].\end{aligned} \tag{12}$$

We used $\beta_{S/T}$ Monte Carlo samples to approximate the expected values of ψ_{k_S} and ψ_{k_T}. The final end-to-end stochastic training loss function is:

$$\mathcal{L}(\theta, \phi) = \frac{1}{K} \sum_{k=1}^{K} \frac{1}{M_k} \sum_{m=1}^{M_k} \log \frac{1}{\beta_S \beta_T} \sum_{i_S=1}^{\beta_S} \sum_{i_T=1}^{\beta_T} p(\tilde{y}_m|\tilde{x}_m, \psi_{k_S}^{i_S}, \psi_{k_T}^{i_T}, \theta), \tag{13}$$

where M_k represents the number of samples in the query set under task k. The final model loss function is defined as:

$$\mathcal{L}_{AMPIF} = \mathcal{L}(\theta, \phi) + \alpha \mathcal{L}_{GRL}, \tag{14}$$

where α is a tunable hyperparameter.

Experiments

To comprehensively evaluate the performance advantages of AMPIF, we designed a series of experiments to compare it with the current state-of-the-art methods for long sequence prediction. In addition, we further explored the roles and effects of each module of the model through data scarcity experiments and ablation studies.

We conducted a series of experiments and analyses in the Appendix. Specifically, in Appendix E.4, we used t-SNE (Van der Maaten and Hinton 2008) technology to visualize the representation of seasonal and trend components; in Appendix E.5, we performed a visual analysis of sequence decomposition; in Appendix E.6, we demonstrated the effectiveness of global sequence clustering; in Appendix E.7, we presented the results of global representation ablation experiments; and in Appendix E.8, we performed fluctuation analysis.

Settings

Datasets We constructed two real-world stock and index datasets to explore AMPIF's long sequence prediction capabilities in more complex new stock/index scenarios. In addition, we also validated the improvement of AMPIF's long sequence prediction performance on base models on the publicly available ETT datase. **(1) A-share**: This dataset contains historical data of 3579 stocks in the A-share market from 2017 to 2022. We extracted 300 factors as auxiliary information for prediction for each stock, and the specific factor construction process is detailed in Appendix D. **(2) Cross-Market**: This dataset covers index data from 22 markets from March 2018 to March 2023. This dataset only uses closing

Methods		AMPIF		Autoformer		Informer		LogTrans		Reformer		LSTnet		LSTMa	
Metrics		mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae	mse	mae
ETTh1	24	**0.360**	**0.390**	0.384	0.425	0.577	0.549	0.686	0.604	0.991	0.754	1.293	0.901	0.650	0.624
	48	**0.370**	**0.392**	0.392	0.419	0.685	0.625	0.766	0.757	1.313	0.906	1.456	0.960	0.702	0.675
	168	**0.427**	**0.424**	0.490	0.481	0.931	0.752	1.002	0.846	1.824	1.138	1.997	1.214	1.212	0.867
	336	**0.502**	**0.471**	0.505	0.484	1.128	0.873	1.362	0.952	2.117	1.280	2.655	1.369	1.424	0.994
	720	**0.496**	**0.493**	0.498	0.500	1.215	0.896	1.397	1.291	2.415	1.520	2.143	1.380	1.960	1.322
ETTh2	24	**0.230**	**0.323**	0.261	0.341	0.720	0.665	0.828	0.750	1.531	1.613	2.742	1.457	1.143	0.813
	48	**0.240**	**0.345**	0.312	0.373	1.457	1.001	1.806	1.034	1.871	1.735	3.567	1.687	1.671	1.221
	168	**0.311**	**0.392**	0.457	0.455	3.489	1.515	4.070	1.681	4.660	1.846	3.242	2.513	4.117	1.674
	336	**0.399**	**0.430**	0.471	0.475	2.723	1.340	3.875	1.763	4.028	1.688	2.544	2.591	3.434	1.549
	720	**0.437**	**0.449**	0.474	0.484	3.467	1.473	3.913	1.552	5.381	2.015	4.625	3.709	3.963	1.788
ETTm1	24	**0.312**	**0.363**	0.383	0.403	0.323	0.369	0.419	0.412	0.724	0.607	1.968	1.170	0.621	0.629
	48	**0.360**	**0.391**	0.454	0.453	0.494	0.503	0.507	0.583	1.098	0.777	1.999	1.215	1.392	0.939
	96	**0.384**	**0.404**	0.481	0.463	0.678	0.614	0.768	0.792	1.433	0.945	2.762	1.542	1.339	0.913
	228	**0.500**	**0.466**	0.634	0.528	1.056	0.786	1.462	1.320	1.820	1.094	1.257	2.076	1.740	1.124
	672	**0.565**	**0.514**	0.606	0.542	1.192	0.926	1.669	1.461	2.187	1.232	1.917	2.941	2.736	1.555
ETTm2	24	**0.120**	**0.231**	0.153	0.261	0.173	0.301	0.211	0.332	0.333	0.429	1.101	0.831	0.580	0.572
	48	**0.156**	**0.268**	0.178	0.280	0.303	0.409	0.427	0.487	0.558	0.571	2.619	1.393	0.747	0.630
	96	**0.205**	**0.312**	0.255	0.339	0.365	0.453	0.768	0.642	0.658	0.619	3.142	1.365	2.041	1.073
	228	**0.229**	**0.342**	0.342	0.378	1.047	0.804	1.090	0.806	2.441	1.190	2.856	1.329	0.969	0.742
	672	**0.317**	**0.387**	0.434	0.430	3.126	1.302	2.397	1.214	3.090	1.328	3.409	1.420	2.541	1.239

Table 3: Multivariate results on the four ETT datasets with predicted length as 24, 48, 168, 288, 336, 672, 720. We fix the input length of AMPIF as 96.

prices for prediction. We divided the dataset according to the categories of stocks and indices to ensure that the stocks or indices in the test set have not appeared in the training set. We divided the training/validation/test data in a ratio of 6/2/2. **(3) ETT** *: This dataset is often used for long sequence time series prediction. It contains data from two different regions in China, recorded at 2-hour, 1-hour, and 15-minute intervals, respectively, from July 2016 to July 2018. Each data point includes oil temperature and six electricity load indicators.

Baselines Since traditional meta-learning methods assume that tasks have already been divided, it is difficult to find a meta-learning algorithm for comparison that satisfies the criteria of complete code availability, no need for additional auxiliary programs, and a scheme that can be directly transferred to financial long sequence prediction tasks. In the financial long sequence prediction task, we compared AMPIF with the current state-of-the-art long sequence prediction methods, including LSTM (Hochreiter and Schmidhuber 1997) based on RNN recursive prediction, Autoformer (Wu et al. 2021), and FEDformer (Zhou et al. 2022) based on Transformer (Vaswani et al. 2017) long sequence prediction methods, and LaST (Wang et al. 2022) based on sequence decomposition representation learning. In the ETT dataset, we mainly verified the performance improvement of AMPIF on base models by replacing both the Season Encoder and Trend Encoder in AMPIF with the autocorrelation mechanism in Autoformer (Wu et al. 2021). Therefore, we mainly compared the results with Autoformer. In addition, we also introduced several other popular long sequence prediction methods such as Informer (Zhou et al. 2021), LogTrans (Li et al. 2019), Reformer (Kitaev, Kaiser, and Levskaya 2020), LSTnet (Lai et al. 2018), and LSTMa (Bahdanau, Cho, and Bengio 2014) as baselines for comparison.

Implementation Details We optimized our method using the Adam optimizer. In all methods, the learning rate was set to 0.00001, and the batch size was set to 256, while the hyperparameters of the other baselines were consistent with those in their original papers. During training, 50% of the samples in each batch were used as a support set. During testing, the support set was randomly drawn from the training set with equal samples. During training, the model performance was evaluated on the validation set at each epoch. The best model on the validation set during 100 epochs of training was saved for final prediction on the test set. The specific details of the encoder can be found in the Appendix E.3. We set the vector dimension to 128 and the number of clusters for seasonal and trend components $K_{S/T}$ to 10. The number of task-specific parameter samples for seasonal and trend components $\beta_{S/T}$ is set to 20. τ_A and τ_P are set to 0.01 and 1 respectively, while α is set to 1. All models were trained/tested on NVIDIA Tesla V100 32G GPUs.

Main Results

Long Series Forecasting for Financial Markets Table 1 and Table 2 show the prediction results of AMPIF on the A-share and Cross-Market datasets, respectively. For the A-share dataset, we used two input lengths of 24 and 96 to predict sequences of lengths 24, 48, 168, 336, 720. For the Cross-Market dataset, we fixed the input sequence length at 12 to predict sequences of lengths 22, 66, 264, 528. From the results, we can observe that compared to Autoformer and

*https://github.com/zhouhaoyi/ETDataset

Datasets	Models	22 trading days		66 trading days		264 trading days		528 trading days	
		MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
Cross-Market	AMPIF$_{decomp-}$	0.234	0.111	0.345	0.224	0.520	0.440	0.600	0.649
	AMPIF$_{ST\text{-}DAN-}$	0.304	0.199	0.394	0.299	0.682	0.711	0.733	0.849
	AMPIF$_{season-}$	0.242	0.114	0.336	0.217	0.506	0.423	0.575	0.603
	AMPIF$_{trend-}$	0.229	0.104	0.373	0.246	0.518	0.438	0.705	0.779
	AMPIF	**0.214**	**0.096**	**0.314**	**0.196**	**0.445**	**0.353**	**0.475**	**0.418**

Table 4: Conducting ablation studies on the individual components of AMPIF on the Cross-Market dataset.

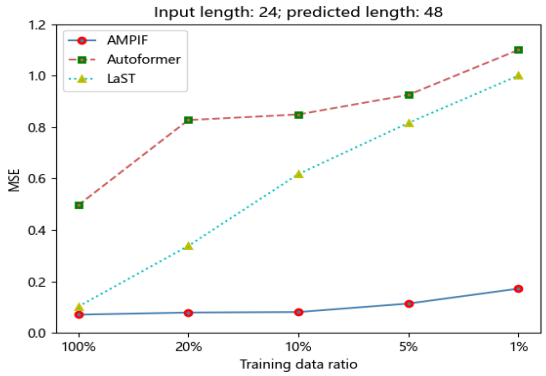

Figure 2: Comparing the stability of models trained with different amounts of training data on the A-share dataset with an input length of 24 and a prediction length of 48.

FEDformer, which focus more on learning seasonal components while ignoring trend changes, LaST is more suitable for learning complex sequences by extracting seasonal and trend representations. Compared to LaST, our model reduced the MSE by 35.8% and the MAE by 24.5% on average on the A-share dataset; on the Cross-Market dataset, our model reduced the MSE by 46.0% and the MAE by 36.1% on average. Further analysis of experimental results can be found in Appendices C.3 and C.4.

More Results on ETT Benchmark In AMPIF, we replace both the Season Encoder and Trend Encoder with the autocorrelation mechanism in Autoformer to verify the improvement of the model effect by removing other complex structures of Autoformer and only using a simple autocorrelation mechanism. We experimented on the publicly available ETT (Electricity Transformer Temperature) dataset. This real-world dataset was collected by (Zhou et al. 2021) from electricity data. For ease of comparison, we keep the training/evaluation/testing data partition consistent with Autoformer, dividing 20 months of data into 12/4/4. At the same time, the length of the input sequence is consistent with Autoformer, both set to 96.

The results are shown in Table 3. From the results, we can observe that AMPIF achieved a significant performance improvement. Compared to Autoformer, on the ETT dataset, for different prediction lengths, AMPIF reduced the MSE by an average of 15.8% and the MAE by 8.5%.

Ablation Studies We conducted ablation experiments on the Cross-Market dataset to further explore the roles and contributions of each module in the framework. The experimental results are shown in Table 4. First, we removed the frequency decomposition module from AMPIF (AMPIF$_{decomp-}$). We directly fed the original sequence into the seasonal and trend encoding modules to test the effectiveness and robustness of sequence decomposition encoding. Second, we removed ST-DAN from AMPIF (AMPIF$_{ST\text{-}DAN-}$) and directly added the outputs of the seasonal and trend encoders as the final prediction to verify the necessity of meta-learning. Finally, we separately constructed seasonal (AMPIF$_{trend-}$) and trend (AMPIF$_{season-}$) components by removing the seasonal module and frequency decomposition module and only inputting the original sequence into the trend encoding module; removing the trend module and frequency decomposition module and only inputting the original sequence into the seasonal encoding module to verify model performance from a single perspective. The results show that all components of the model are indispensable.

Few Data Training We conducted further experiments on the A-share dataset to validate AMPIF's learning ability in low-resource environments. As shown in Fig.(2), in the task with an input sequence length of 24 and an output sequence length of 48, we used only 20%, 10%, 5%, and 1% of the original training data to train the model, respectively. For example, with a batch size of 256, 284 batches can be divided into the training set. When we only use 10% of the training set for training, 28 batches are randomly selected with equal probability. We compared the algorithm stability of AMPIF with Autoformer and LaST, two advanced long sequence algorithms, under conditions of limited training data. The results show that as the scale of training data decreases, AMPIF exhibits stable performance and strong generalization ability.

Conclusion

This paper proposes an Adaptive Meta-Learning Probabilistic Inference Framework (AMPIF) for long sequence prediction. This can address the potential domain distribution shift problem in long sequence prediction tasks. AMPIF provides a solution for the wider application of meta-learning through adaptive meta-learning task construction. Our extensive experiments demonstrate that AMPIF can be well applied to long-sequence prediction tasks in complex scenarios. At the same time, as a general framework, AMPIF can also effectively enhance the long sequence prediction capabilities of other base models. The limitations and broader implications of AMPIF are discussed in Appendices F and G, respectively.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (No. 62172074), and Program of Introducing Talents of Discipline to Universities (Plan 111) (No. B20070).

References

Bahdanau, D.; Cho, K.; and Bengio, Y. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Chang, S.-H.; Hsu, C.-W.; Li, H.-Y.; Zeng, W.-S.; and Ho, J.-M. 2021. Short-Term Stock Price-Trend Prediction Using Meta-Learning. In *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2900–2905. IEEE.

Chen, R.; Li, W.; Zhang, Z.; Bao, R.; Harimoto, K.; and Sun, X. 2023. Stock Trading Volume Prediction with Dual-Process Meta-Learning. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2022, Grenoble, France, September 19–23, 2022, Proceedings, Part VI*, 137–153. Springer.

Du, D.; Su, B.; and Wei, Z. 2022. Preformer: predictive transformer with multi-scale segment-wise correlations for long-term time series forecasting. *arXiv preprint arXiv:2202.11356*.

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, 1126–1135. PMLR.

Gordon, J.; Bronskill, J.; Bauer, M.; Nowozin, S.; and Turner, R. E. 2018. Meta-learning probabilistic inference for prediction. *arXiv preprint arXiv:1805.09921*.

Hochreiter, S.; and Schmidhuber, J. 1997. Long short-term memory. *Neural computation*, 9(8): 1735–1780.

Jin, X.; Park, Y.; Maddix, D.; Wang, H.; and Wang, Y. 2022. Domain adaptation for time series forecasting via attention sharing. In *International Conference on Machine Learning*, 10280–10297. PMLR.

Kingma, D. P.; and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Kitaev, N.; Kaiser, Ł.; and Levskaya, A. 2020. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*.

Lai, G.; Chang, W.-C.; Yang, Y.; and Liu, H. 2018. Modeling Long- and Short-Term Temporal Patterns with Deep Neural Networks. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, 95–104.

Li, S.; Jin, X.; Xuan, Y.; Zhou, X.; Chen, W.; Wang, Y.-X.; and Yan, X. 2019. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. *Advances in neural information processing systems*, 32.

Liu, T.; Ma, X.; Li, S.; Li, X.; and Zhang, C. 2022. A stock price prediction method based on meta-learning and variational mode decomposition. *Knowledge-Based Systems*, 252: 109324.

Lv, P.; Shu, Y.; Xu, J.; and Wu, Q. 2022. Modal decomposition-based hybrid model for stock index prediction. *Expert Systems with Applications*, 202: 117252.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, Z.; Xu, X.; Zhang, W.; Trajcevski, G.; Zhong, T.; and Zhou, F. 2022. Learning Latent Seasonal-Trend Representations for Time Series Forecasting. In *Advances in Neural Information Processing Systems*.

Woo, G.; Liu, C.; Sahoo, D.; Kumar, A.; and Hoi, S. 2022a. CoST: Contrastive learning of disentangled seasonal-trend representations for time series forecasting. *arXiv preprint arXiv:2202.01575*.

Woo, G.; Liu, C.; Sahoo, D.; Kumar, A.; and Hoi, S. 2022b. Etsformer: Exponential smoothing transformers for time-series forecasting. *arXiv preprint arXiv:2202.01381*.

Wu, H.; Xu, J.; Wang, J.; and Long, M. 2021. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in Neural Information Processing Systems*, 34: 22419–22430.

Yao, H.; Wang, Y.; Wei, Y.; Zhao, P.; Mahdavi, M.; Lian, D.; and Finn, C. 2021. Meta-learning with an adaptive task scheduler. *Advances in Neural Information Processing Systems*, 34: 7497–7509.

Zhan, D.; Dai, Y.; Dong, Y.; He, J.; Wang, Z.; and Anderson, J. 2022. Meta-adaptive stock movement prediction with two-stage representation learning. In *NeurIPS 2022 Workshop on Distribution Shifts: Connecting Methods and Applications*.

Zhang, Q.; Fang, J.; Meng, Z.; Liang, S.; and Yilmaz, E. 2021. Variational continual Bayesian meta-learning. *Advances in Neural Information Processing Systems*, 34: 24556–24568.

Zhao, S.; Song, J.; and Ermon, S. 2017. Infovae: Information maximizing variational autoencoders. *arXiv preprint arXiv:1706.02262*.

Zhao, Y.; and Chen, Z. 2022. Forecasting stock price movement: New evidence from a novel hybrid deep learning model. *Journal of Asian Business and Economic Studies*, 29(2): 91–104.

Zhou, H.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; and Zhang, W. 2021. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 11106–11115.

Zhou, T.; Ma, Z.; Wen, Q.; Wang, X.; Sun, L.; and Jin, R. 2022. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International Conference on Machine Learning*, 27268–27286. PMLR.