Skip-Timeformer: Skip-Time Interaction Transformer for Long Sequence Time-Series Forecasting

Wenchang Zhang[†1], **Hua Wang**[†2], **Fan Zhang**[*1,3]
[1]School of Computer Science and Technology, Shandong Technology and Business University, Yantai 264005, China
[2]School of Information and Electrical Engineering, Ludong University, Yantai 264025, China
[3]Shandong Future Intelligent Financial Engineering Laboratory, Yantai 264005, China
a592006923@gmail.com, hwa229@163.com, zhangfan@sdtbu.edu.cn

Abstract

Recent studies have raised questions about the suitability of the Transformer architecture for long sequence time-series forecasting. These forecasting models leverage Transformers to capture dependencies between multiple time steps in a time series, with embedding tokens composed of data from individual time steps. However, challenges arise when applying Transformers to predict long sequences with strong periodicity, leading to performance degradation and increased computational burden. Furthermore, embedding tokens formed one time step at a time may struggle to reveal meaningful information in long sequences, failing to capture correlations between different time steps. In this study, we propose Skip-Timeformer, a Transformer-based model that utilizes a skip-time interaction for long sequence time-series forecasting. Specifically, we decompose the time series into multiple subsequences based on different time intervals, embedding various time steps into variable tokens across multiple sequences. The skip-time interaction mechanism utilizes these variable tokens to capture dependencies in the skip-time dimension. Additionally, skip-time interaction is employed to learn dependencies between sequences missed by multiple skip time steps. The Skip-Timeformer model demonstrates state-of-the-art performance on various real-world datasets, further enhancing the long sequence forecasting capabilities of the Transformer variations and better adapting to arbitrary lookback windows.

1 Introduction

Long sequence time-series forecasting (LSTF) holds significant practical relevance in complex real-world scenarios [Demirel *et al.*, 2012; Angryk *et al.*, 2020; Patton, 2013; Zhang *et al.*, 2023b; Wang *et al.*, 2023]. For instance, stock prices are influenced by factors such as the company's performance reports over the past few months, industry trends, news events, and investor sentiment, making the advance prediction of stock prices crucial [Liu *et al.*, 2022]. Given its immense practical value, long sequence time-series forecasting has garnered widespread research interest [Lim and Zohren, 2021]. Diverging from other sequential data types, such as language or video, time series distinguishes itself by its continuous record, where each time step stores only limited scalar information. Since individual time steps often fall short in providing sufficient semantic information for in-depth analysis, researchers have shifted their focus towards the temporal dynamics. This temporal evolution encapsulates richer information, more authentically reflecting the inherent properties of time series, such as continuity, periodicity, and trendiness. Real-world time series variations frequently involve intricate temporal patterns, encompassing diverse forms of changes (e.g., ascent, descent, fluctuation) that intertwine and overlap, posing a substantial challenge in modeling temporal dynamics. Confronted with this complexity, in-depth exploration of time series forecasting methods, especially in the realm of long sequence forecasting model design, becomes an imperative need to address practical challenges. This necessitates delving into the underlying deep patterns to enhance the accuracy of forecasting future trends.

In recent years, deep learning has not only demonstrated outstanding performance in predictive tasks but has also excelled in representation learning. It can extract abstract representations and transfer them to various downstream tasks, such as classification and anomaly detection, yielding state-of-the-art performance. Notably, Transformer-based models have achieved tremendous success in natural language processing [Kalyan *et al.*, 2021] and computer vision [Khan *et al.*, 2022; Zhang *et al.*, 2023a; Zhang *et al.*, 2024a], showing great potential in capturing time dependencies in the field of time series [Zhou *et al.*, 2021; Wu *et al.*, 2021; Nie *et al.*, 2022; Zhang *et al.*, 2024b].

Recently, researchers have raised concerns about the effectiveness of Transformer-based models in long sequence forecasting [Zeng *et al.*, 2023]. Existing studies leveraging Transformer for long-term multivariate time series forecasting have exhibited suboptimal performance. This is largely due to the fact that most existing LSTF methods primarily focus on reducing the computational cost of univariate predictions, lacking research specifically tailored to the characteristics of LSTF data. Applying Transformer models to long-term sequence forecasting faces two main challenges.

*Corresponding Author: zhangfan@sdtbu.edu.cn

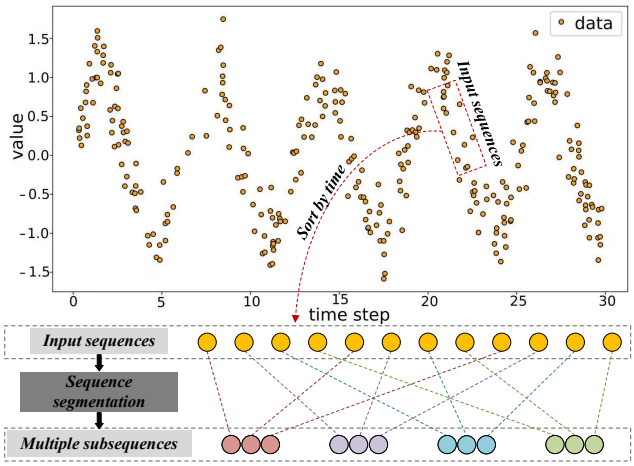

Figure 1: Illustration of Skip-Timeformer embeddings with a skip step of 4. Unlike Transformer, which embeds each time step into time tokens, Skip-Timeformer embeds multivariate long sequence time-series into multiple variable subsequence tokens with different skip steps. This allows the model to more easily capture long-term trends.

Initially, the Transformer was originally designed for language analysis [Devlin *et al.*, 2018], where each word could be regarded as an individual sequence of data. However, in the context of time series forecasting, as illustrated in Figure 1, there are periodic and irregular fluctuations between long-term time sequences. Tokens represented at single time step interval struggle to unveil valuable information, and the model's performance may even decline due to irregular, unforeseen events. Consequently, the model's ability to analyze interactions between different data sequences is limited. Secondly, LSTF is more susceptible to overfitting compared to short-term predictions. Transformer-based models are not suitable for training on longer sequences at once. After multiple rounds of training, these models tend to memorize intricate details from the training data, failing to accurately capture real trends and patterns.

To address the aforementioned issues, we focus is on enhancing the capabilities of transformer-based models in long sequence forecasting at the sequence data level. In the context of long sequence time-series problems, capturing substantial historical information is crucial, especially in the LSTF setting. Specifically, we introduce the multi-skip sequence token embedding, as depicted in Figure 1, where sequences in each dimension are segmented and then embedded into feature vectors. The output of the multi-skip sequence token embedding is a one-dimensional array of multiple sequences. Subsequently, we propose skip-time interaction forecasting to capture dependencies between different skipping times among multiple subsequences based on features at various scales. Utilizing skip-time interaction conditional layer normalization, we combine multi-skip sequence tokens with original sequence tokens to analyze dependencies and global patterns missed by multi-skip time steps. Here, we employ a dynamic Dropout scheme, progressively introducing Bernoulli noise to the training data to prevent model overfitting and enhance robustness. Our contributions are as follows:

- We conducted a critical examination of the Transformer architecture and identified that the inherent capabilities of the native Transformer structure for long sequence time-series have not been fully explored. These models embed data points for all dimensions at each time step, sequentially aggregating them into a single vector, focusing on capturing dependencies between variables across different time steps. Without explicitly and adequately mining and utilizing the dependencies between different time steps in long sequences, their predictive capabilities are limited. To address this, we propose the method of multi-skip sequence token embedding.
- We introduce Skip-Timeformer, a Transformer-based model designed for long sequence forecasting that leverages the dependency on skip-time dimensions. Skip-Timeformer is one of the few Transformer models explicitly exploring and utilizing skip-time dependencies for long sequence forecasting.
- Extensive experimental results on eight real-world benchmarks demonstrate that Skip-Timeformer consistently achieves state-of-the-art performance on real-world prediction benchmarks. We conduct in-depth analyses of token embedding methods and architecture choices, providing new insights for the future development of Transformer-based models in long sequence forecasting.

2 Related Work

In recent years, Transformer-based models have achieved state-of-the-art results in numerous time series tasks [Zeng *et al.*, 2023], particularly in long short-term forecasting (LSTF) problems [Wu *et al.*, 2020; Wu *et al.*, 2021; Jin *et al.*, 2023]. Here, we summarize some of the mainstream models. LogTrans [Li *et al.*, 2019] utilizes convolutional self-attention layers with LogSparse design to capture local information, reducing spatial complexity. Informer [Zhou *et al.*, 2021] introduces ProbSparse self-attention, effectively extracting the most important keywords using extraction techniques. Autoformer [Wu *et al.*, 2021] incorporates decomposition and autocorrelation ideas from traditional time series analysis methods. FEDformer [Zhou *et al.*, 2022] employs a Fourier-enhanced structure to achieve linear complexity. Pyraformer [Liu *et al.*, 2021] applies pyramid attention modules with both inter- and intra-scale connections, also maintaining linear complexity. These models predominantly focus on designing novel mechanisms to reduce the complexity of the original attention mechanism, thereby achieving better predictive performance, especially for longer prediction horizons. PatchTST [Nie *et al.*, 2022] takes a different approach by segmenting time series into subsequence-level patches, serving as input tokens for the Transformer. This enhances the local semantic information of the sequence.

It is noteworthy that, unlike previous approaches, we analyze the applicability of Transformer in handling long sequence time-series. Our approach aims to make the Trans-

former suitable for long sequence time-series forecasting by exploring sequence token embeddings.

3 Skip-Timeformer

In the context of multivariate long sequence time series forecasting, given historical observations $X = \{x_1, ..., x_T\} \in R^{T \times D}$ with T time steps and D variables, we aim to predict the future U time steps $Y = \{y_{T+1}, ..., y_{T+U}\} \in R^{U \times D}$. For convenience, let's denote $x_{1:T}$ as the set of T time steps representing recorded multivariate variables and $\bar{y}_{T:T+U}$ as the set of multiple variables predicting the future U time steps.

3.1 Multi-skip Sequence Token Embedding

For the Transformer, tokens embedded in it are formed only by single time steps, resulting in weaker interaction capability for long sequential data. To address this limitation, we decided to partition the time series into multiple subsequences based on different time intervals, as illustrated in Figure 1. We propose a method for embedding tokens, known as multi-skip sequence token embedding. This method involves dividing the multi-variable long sequence into various subsequences based on a skip step length μ. Subsequently, to emphasize edge semantic information, we use padding to generate a series of sequences $S_1, S_2, ..., S_\mu$. This transformation converts tokens formed by a single time step into tokens formed by multiple skip time steps. It enables the transformer to capture time-related dependencies at different time scales when processing long sequential data. For instance, certain subsequences can aid in capturing short-term variations, while others may be more helpful in capturing long-term trends. The tokens with multiple skip time steps not only assist the model in adapting to variations across various time scales but also enhance the model's generalization ability, helping prevent performance degradation on other time scales while overfitting to a specific one.

3.2 Skip-Time Interaction Forecasting

To further enhance the outstanding performance of our proposed method of multi-skip sequence token embedding in long sequence time series forecasting tasks, we introduce skip-time interaction forecasting. This conceptual framework is specifically designed for Transformer models in the context of long sequence forecasting. Our objective is to augment the Transformer's ability to handle the data complexity, prolonged temporal dependencies, and the identification of hidden temporal patterns within multiple highly similar long sequences. Hence, we propose skip-time interaction forecasting, as illustrated in Figure 2.

Given the output $S \in R^{\mu \times T \times D}$ of a multi-skip sequence token embedding layer as input to the skip-time interaction, where μ and D represent the number of subsequences and dimensions, respectively. For simplicity, we use S_i^μ in the following text to denote the vector of all dimensions of the i-th sub-sequence at time step t. In skip-time interaction forecasting, we directly apply multi-head self-attention (MSA) to all dimensions of each sub-sequence:

$$S^{skip-time} = S_t^i + MSA^{skip-time}\left(S_t^i, S_t^i, S_t^i\right) \quad (1)$$

To further refine the concept of skip-time interaction forecasting, combining the obtained information with the original sequence input helps analyze the dependencies between sequences missed by multi-skip time steps. We employ skip-time at the task level to adapt to time series characteristics. The connection of time interaction information to the original sequence input is achieved through the skip-time interaction conditional layer normalization (STICLN) layer. The general formula for this connection can be expressed as:

$$STICLN\left(h^l, MSA\left(S^{l-1}\right)\right) \quad (2)$$

Here, $STICLN(\cdot)$ represents skip-time interaction conditional layer normalization, $MSA(\cdot)$ denotes the attention layer in the encoder block, and S^{l-1} signifies the output state of the preceding layer $l-1$ and S^0 indicates the multiple subsequences obtained in the embedding layer $S_1, S_2, ..., S_\mu$. h^l represents the embedding token of the original sequence.

3.3 Skip-Time Interaction Conditional Layer Normalization

The conditional layer normalization of skip time interaction integrates the multi-skip sequence tokens into the encoder. In this context, the encoder's input comprises the output from the self-attention layer of the multi-skip sequence tokens and the embedding token information provided from the original sequence. Longer sequences often encompass more time steps, and the model can benefit from global statistical information across the entire sequence. By considering the entire sequence when computing mean and variance, the model can better adapt to the global characteristics of the sequence, aiding in capturing long-term dependencies within the sequence. We normalize the features of the entire sequence information to ensure each temporal feature carries the same weight for generating predictions.

Within $STICLN$, we update α and β based on the multi-skip sequence token embedding, as they are the two most crucial parameters for normalization. For each time step t and each sample $S_{t,j}^{(i)}$ with D feature dimensions, the proposed ST is expressed as follows:

$$\begin{aligned}
\alpha_t &= \alpha + k_l\left(S^l\right) \\
\beta_t &= \beta + k_l\left(S^l\right) \\
\theta &= \frac{1}{l \cdot D} \sum_{i=1}^{l} \sum_{j=1}^{D} S_{t,j}^{(i)} \\
\sigma &= \frac{1}{l \cdot D} \sum_{i=1}^{l} \sum_{j=1}^{D} \left(S_{t,j}^{(i)} - \theta\right)^2
\end{aligned} \quad (3)$$

Here, $k(\cdot)$ represents the linear layer. Scaling and shifting are applied to the normalized values:

$$STICLN\left(h^{l-1}, S^l\right) = \alpha_t \odot \frac{h^{l-1} - \theta}{\sigma} + \beta_t \quad (4)$$

Where θ and σ represent the mean and standard deviation of h^{l-1}.

Taking the above considerations into account, the modeling process for predicting the future sequence of each specific

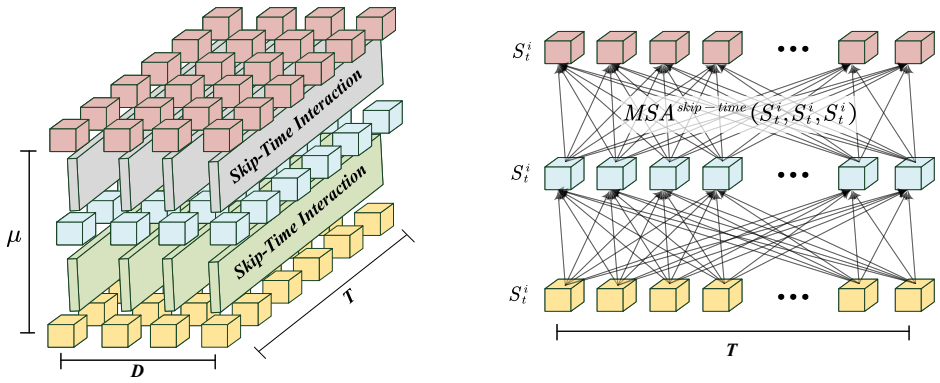

Figure 2: Skip-time interaction layer. Used to process arrays of vectors representing multiple subsequence segments of long sequence time-series, where each vector denotes a fragment of the original sequence. The entire vector array undergoes skip-time interaction to obtain corresponding dependencies (left). Multi-head self-attention (MSA) is employed within the skip-time interaction to establish connections between multiple subsequences (right).

variable based on backtracking sequences can be succinctly described as follows:

$$\begin{aligned} &h_t^0, \left[S_1^0, S_2^0, ..., S_n^0\right] = Embedding\left(x_{1:T}\right) \\ &H^l = STICLN\left(MSA\left(S^l, H^{l-1}\right), h^l\right) \\ &H^l = Droput\left(H^l\right) + S^l \\ &\overline{y}_{T:T+U} = Project\left(h_t^L\right) \end{aligned} \tag{5}$$

Where $H = \{h_1, ..., h_t\}, H \in R^{t \times D}$ consists of t embedding tokens with D dimensions, h^l is the embedding token of the original sequence, the superscript denotes the layer index, $\left[S_1^0, S_2^0, ..., S_n^0\right]$ represents the generated n multi-skip sequence tokens. These tokens are connected through a fully connected layer in the embedding, transitioning from R^T to R^D, and from R^D to R^C in the projection. The multi-skip sequence tokens and the output of the previous layer $STICLN$ interact through an MSA layer, and the obtained information is passed into the l-th layer along with the original sequence tokens. We propose an approach that gradually introduces Bernoulli noise to the training data through dropout, progressively increasing the difficulty of the training task and the diversity of the data. This can effectively reduce overfitting and enhance model performance. The output of the L-th layer is linearly projected, denoted as h_t^L to obtain the final prediction. The overall framework is illustrated in Figure 3, where the proposed Skip-Timeformer leverages the simpler encoder architecture of Transformer [Vaswani *et al.*, 2017], consisting of embedding, $STICLN$, projection, and Transformer blocks.

4 Experiments

We thoroughly evaluated the proposed Skip-Timeformer across various time series forecasting applications, validating the framework's generality. Additionally, we conducted an in-depth investigation into the impact of the Transformer components' skip-time interaction on the long sequence time-series dimension.

4.1 Datasets

In our experiment, we extensively incorporated eight real-world datasets, comprising four ETT datasets employed by Autoformer [Wu *et al.*, 2021] (ETTh1, ETTh2, ETTm1, ETTm2), as well as datasets related to Weather, Electricity, Exchange and Traffic. These datasets have been widely utilized for benchmarking.

4.2 Baselines

We meticulously selected eight widely recognized predictive models as our benchmarks, encompassing (1) Transformer-based approaches: Informer [Zhou *et al.*, 2021], Autoformer [Wu *et al.*, 2021], FEDformer [Zhou *et al.*, 2022], ETSformer [Woo *et al.*, 2022], PatchTST [Nie *et al.*, 2022], Crossformer [Zhang and Yan, 2022]; (2) Linear-based methods: DLinear [Zeng *et al.*, 2023]; (3) TCN-based approaches: TimesNet [Wu *et al.*, 2022a].

4.3 Main Results

The comprehensive prediction results are presented in Table 1, with the superior outcomes highlighted in red. Lower values of MSE and MAE indicate more accurate predictions. The proposed Skip-Timeformer achieves consistently state-of-the-art performance. It is noteworthy that PatchTST, considered the current best model for time series forecasting, falls short on multiple datasets compared to our model. This discrepancy can be attributed to the high-frequency fluctuations in long sequences of the datasets, and the patching mechanism of PatchTST may emphasize local attention, leading to failures in predicting long-term trends.

In contrast, our proposed approach, leveraging an aggregation-based representation of the entire series variation, proves more adept at handling such scenarios. Additionally, as a representative explicitly designed to capture periodicity, TimesNet's performance still lags behind that of Skip-Timeformer. This suggests that representing time series as a 2D structure still falls short in effectively capturing the long sequence trends and patterns in the sequences.

Therefore, Transformer-based components demonstrate competence in long sequence time-series modeling, and the

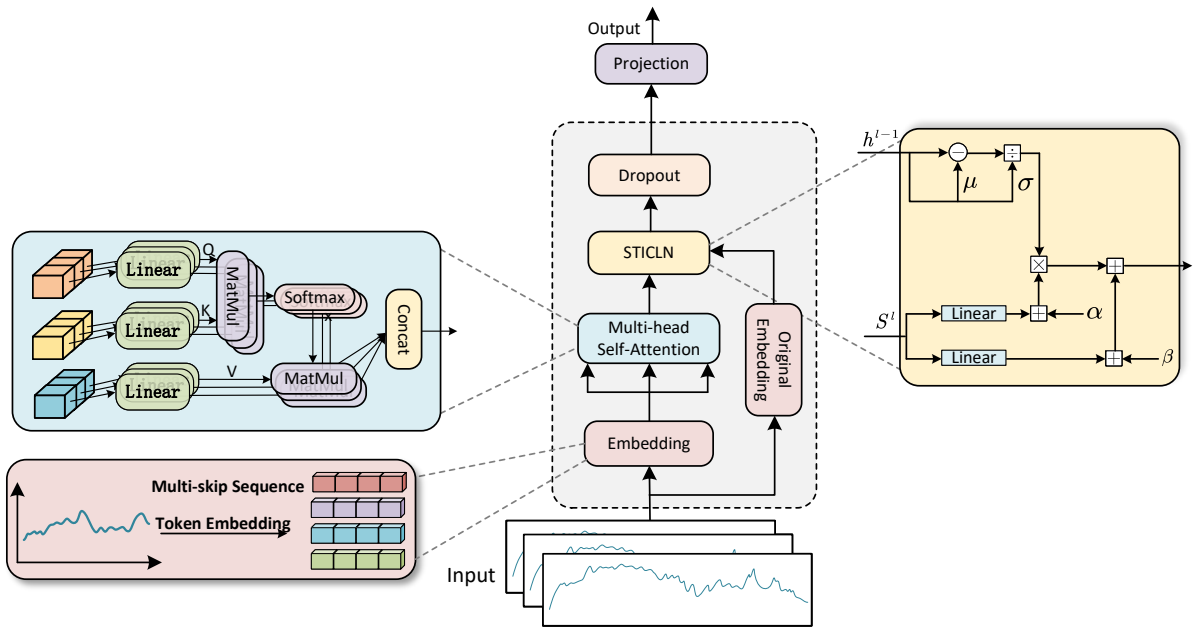

Figure 3: Overall framework of Skip-Timeformer. Our model comprises three main elements: (1) Multiple subsequences embedded into tokens based on time intervals. (2) Applying multi-head self-attention to multi-skip sequence token embedding to enhance interpretability and reveal correlations among multiple subsequences. (3) Employing skip-time interaction conditional layer normalization (STICLN) to reduce differences between subsequences. Dropout is used to introduce Bernoulli noise to training data, effectively reducing overfitting.

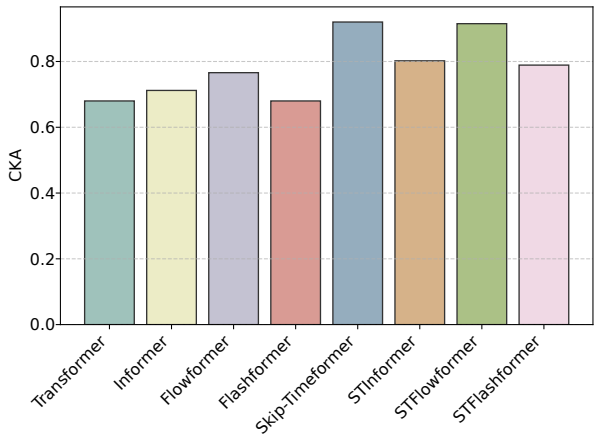

Figure 4: CKA similarity for different models on Weather dataset.

proposed Skip-Timeformer's architecture with skip-time interactions proves effective in real-world long sequence time-series forecasting scenarios.

4.4 Skip-Timeformer Generality

In this section, we evaluate the Skip-Timeformer by applying our framework to Transformer and its variants, which commonly deal with the secondary complexity of self-attention mechanisms, including Transformer, Informer, Flowformer [Wu *et al.*, 2022b], and FlashAttention [Dao *et al.*, 2022]. Surprising and promising findings are demonstrated, indicating that skip-time forecasting can enhance the performance and efficiency of transformer-based predictors, generalize to unseen variables, and better utilize historical observations.

We assess the Transformer and the corresponding Skip-Timeformer based on the performance improvements reported in Table 2 and 3. It is noteworthy that the framework continuously improves various Transformers. Overall, the average enhancement rates for Transformer, Informer, Flowformer, and Flashformer on the ETTh1, Weather, and Exchange datasets are 57.3%, 42.6%, 28.5%, 57.8%, and 47.8%, respectively, revealing the inappropriate use of the Transformer architecture in time series forecasting. Furthermore, due to the skip-time interaction forecasting structure, our framework makes it easier for attention mechanisms to capture periodic patterns and trends between long sequences. Therefore, the idea of Skip-Timeformer can be widely applied to transformer-based predictors to leverage the flourishing efficient attention mechanisms.

4.5 Increasing Lookback Length

Some previous studies have found that the predictive performance of the Transformer may not necessarily improve with an increase in lookback length [Nie *et al.*, 2022; Zeng *et al.*, 2023]. This can be attributed to the continuous growth of data, making it challenging for attention mechanisms to capture true trends when dealing with long sequences. Existing research focuses on performance improvement, usually based on linear predictions, with theoretical statistical support [Box and Jenkins, 1968], utilizing expanded historical information to enhance predictive performance. As we partition the sequence into multiple subsequences controlled by skip-time length, we evaluate the performance of both the Transformer and Skip-Timeformer as the lookback length increases, as

Models		Skip-Timeformer		PatchTST		TimesNet		DLinear		ETSformer		Crossformer		FEDformer		Autoformer	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	96	**0.322**	**0.362**	0.332	0.370	0.338	0.375	0.345	0.372	0.375	0.398	0.374	0.400	0.379	0.419	0.505	0.475
	192	**0.366**	0.387	0.371	**0.385**	0.374	0.387	0.380	0.389	0.408	0.410	0.400	0.407	0.426	0.441	0.553	0.496
	336	**0.395**	0.408	0.404	**0.406**	0.410	0.411	0.413	0.413	0.435	0.428	0.438	0.438	0.445	0.459	0.621	0.537
	720	**0.454**	**0.443**	0.462	0.456	0.478	0.450	0.474	0.453	0.499	0.462	0.527	0.502	0.543	0.490	0.671	0.561
	Avg	**0.384**	**0.400**	0.392	0.404	0.400	0.405	0.403	0.406	0.429	0.424	0.434	0.436	0.448	0.452	0.587	0.517
ETTm2	96	**0.177**	**0.258**	0.180	0.262	0.187	0.267	0.193	0.292	0.189	0.280	0.209	0.308	0.203	0.287	0.255	0.339
	192	**0.241**	**0.301**	0.252	0.313	0.249	0.308	0.284	0.362	0.253	0.319	0.311	0.382	0.269	0.328	0.281	0.340
	336	**0.300**	**0.339**	0.318	0.350	0.321	0.351	0.369	0.427	0.314	0.357	0.442	0.466	0.325	0.366	0.339	0.372
	720	**0.400**	**0.390**	0.411	0.405	0.408	0.403	0.554	0.522	0.414	0.413	0.675	0.587	0.421	0.415	0.433	0.432
	Avg	**0.279**	**0.322**	0.290	0.332	0.291	0.332	0.350	0.400	0.292	0.342	0.409	0.435	0.304	0.349	0.327	0.370
ETTh1	96	**0.372**	**0.396**	0.376	0.400	0.384	0.402	0.386	0.400	0.494	0.479	0.305	0.367	0.376	0.419	0.449	0.459
	192	**0.416**	**0.424**	0.422	0.427	0.436	0.429	0.437	0.432	0.538	0.504	0.475	0.462	0.420	0.448	0.500	0.482
	336	**0.451**	**0.444**	0.464	0.454	0.491	0.469	0.481	0.459	0.574	0.521	0.518	0.488	0.459	0.465	0.521	0.496
	720	**0.470**	**0.471**	0.471	0.476	0.521	0.500	0.519	0.516	0.562	0.535	0.547	0.533	0.506	0.507	0.514	0.512
	Avg	**0.427**	**0.433**	0.433	0.439	0.458	0.450	0.455	0.451	0.542	0.509	0.461	0.462	0.440	0.459	0.496	0.487
ETTh2	96	**0.287**	**0.338**	0.302	0.348	0.340	0.374	0.333	0.387	0.340	0.391	0.297	0.349	0.358	0.397	0.346	0.388
	192	**0.359**	**0.385**	0.388	0.400	0.402	0.414	0.477	0.476	0.430	0.439	0.520	0.504	0.429	0.439	0.456	0.452
	336	**0.406**	**0.425**	0.426	0.433	0.452	0.452	0.594	0.541	0.485	0.479	0.626	0.559	0.496	0.487	0.482	0.486
	720	**0.411**	**0.434**	0.431	0.446	0.462	0.468	0.831	0.657	0.500	0.497	0.863	0.672	0.463	0.474	0.515	0.511
	Avg	**0.365**	**0.395**	0.386	0.406	0.414	0.427	0.558	0.515	0.438	0.451	0.576	0.521	0.436	0.449	0.449	0.459
Electricity	96	0.181	**0.271**	0.195	0.285	**0.168**	0.272	0.197	0.282	0.187	0.304	0.219	0.314	0.193	0.308	0.201	0.317
	192	0.186	**0.275**	0.199	0.289	**0.184**	0.289	0.196	0.285	0.199	0.315	0.231	0.322	0.201	0.315	0.222	0.334
	336	0.207	**0.296**	0.215	0.305	**0.198**	0.300	0.209	0.301	0.212	0.329	0.246	0.337	0.214	0.329	0.231	0.338
	720	0.247	0.326	0.256	0.337	**0.220**	**0.320**	0.245	0.333	0.233	0.345	0.280	0.363	0.246	0.355	0.254	0.361
	Avg	0.205	**0.292**	0.216	0.304	**0.192**	0.295	0.211	0.300	0.207	0.323	0.244	0.334	0.213	0.326	0.227	0.337
Exchange	96	**0.079**	**0.197**	0.080	0.199	0.107	0.234	0.088	0.218	0.085	0.204	0.116	0.262	0.148	0.278	0.197	0.323
	192	**0.167**	**0.291**	0.173	0.296	0.226	0.344	0.176	0.315	0.182	0.303	0.215	0.359	0.271	0.380	0.300	0.369
	336	**0.316**	**0.407**	0.323	0.412	0.367	0.448	0.313	0.427	0.348	0.428	0.377	0.466	0.460	0.500	0.509	0.524
	720	**0.787**	**0.666**	0.836	0.688	0.964	0.746	0.839	0.695	1.025	0.774	0.831	0.699	1.195	0.841	1.447	0.941
	Avg	**0.337**	**0.390**	0.353	0.398	0.416	0.443	0.354	0.413	0.410	0.427	0.384	0.446	0.518	0.499	0.613	0.539
Weather	96	**0.171**	**0.212**	0.174	0.214	0.172	0.220	0.196	0.255	0.197	0.281	0.158	0.230	0.217	0.296	0.266	0.336
	192	**0.219**	**0.254**	0.221	0.254	0.219	0.261	0.237	0.296	0.237	0.312	0.206	0.277	0.276	0.336	0.307	0.367
	336	**0.274**	**0.295**	0.278	0.296	0.280	0.306	0.283	0.335	0.298	0.353	0.272	0.335	0.339	0.380	0.359	0.395
	720	**0.353**	**0.346**	0.358	0.349	0.365	0.359	0.345	0.381	0.352	0.288	0.398	0.418	0.403	0.428	0.419	0.428
	Avg	**0.254**	**0.276**	0.257	0.278	0.259	0.286	0.265	0.316	0.271	0.308	0.258	0.315	0.308	0.360	0.337	0.381
Traffic	96	**0.492**	**0.326**	0.544	0.359	0.593	0.321	0.650	0.396	0.607	0.392	0.522	0.390	0.587	0.366	0.613	0.388
	192	**0.491**	**0.324**	0.540	0.354	0.617	0.336	0.598	0.370	0.621	0.399	0.530	0.393	0.604	0.373	0.616	0.382
	336	**0.505**	**0.327**	0.551	0.358	0.629	0.336	0.605	0.373	0.622	0.396	0.558	0.405	0.621	0.383	0.622	0.337
	720	**0.543**	**0.350**	0.589	0.375	0.640	0.350	0.645	0.394	0.632	0.396	0.589	0.428	0.626	0.382	0.660	0.408
	Avg	**0.507**	**0.331**	0.556	0.361	0.619	0.335	0.624	0.383	0.620	0.395	0.549	0.404	0.609	0.376	0.627	0.378

Table 1: Full results of multivariate forecasting. We follow the configuration of TimesNet and compare a variety of competitive models at different prediction lengths. The input sequence length for all baselines is $t = 96$, and the prediction lengths are $s \in \{96, 192, 336, 720\}$. For the ETT, Electricity, Weather, and Traffic datasets, μ is set to 2, while for the Exchange dataset, μ is set to 4, "Avg" represents the average results across all four prediction lengths.

Models		Transformer		Informer		Flowformer		Flashformer	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	Original	1.003	0.807	0.558	0.513	1.046	0.751	0.987	0.799
	Skip-time	**0.427**	**0.433**	**0.431**	**0.434**	**0.425**	**0.427**	**0.444**	**0.442**
	Promotion	57%	46.30%	22.70%	15.30%	59.30%	43.10%	55%	44.60%
Weather	Original	0.657	0.572	0.633	0.548	0.632	0.569	0.658	0.574
	Skip-time	**0.254**	**0.276**	**0.257**	**0.278**	**0.261**	**0.280**	**0.257**	**0.279**
	Promotion	61.30%	51.70%	37.60%	49.20%	58.70%	50.70%	60.90%	51.30%
Exchange	Original	1.208	0.88	1.361	0.922	1.461	0.952	1.254	0.884
	Skip-time	**0.337**	**0.39**	**0.349**	**0.396**	**0.342**	**0.393**	**0.355**	**0.400**
	Promotion	72.10%	56%	74.30%	57%	76.50%	58.70%	71.60%	54.70%

Table 2: Our skip-time interaction forecasting framework achieves significant performance improvement. Detailed results in the Table 3.

shown in Table 4.

The results surprisingly validate the rationale of utilizing multi-skip sequence embeddings in the time dimension. This allows the Transformer to benefit from an extended lookback window, leading to more accurate predictions.

4.6 Model Analysis

In Table 5, we investigate the impact of multi-skip sequence tokens and skip-time interaction forecasting. We use TimesNet as the state-of-the-art model benchmark for long sequence time-series forecasting. By comparing the results with and without the corresponding multi-skip sequence tokens (MST) and skip-time interaction forecasting (STIF), we observe that both are crucial factors in improving predictive performance.

Series Representation Analysis

To further substantiate the claim that skip-time interaction units are more adept at extracting series representations, we conduct representation analysis using Centered Kernel Alignment (CKA) similarity [Kornblith *et al.*, 2019]. Higher CKA values indicate a greater degree of similarity in representations. For both Transformer variants and Skip-Timeformer, we calculate the CKA between the output features after the first layer and those after the last layer, both embedded with multi-skip sequence tokens.

It is important to note that previous studies have demonstrated a preference for higher CKA similarity in time series forecasting, considering it as a low-level generation task, to achieve improved performance [Wu *et al.*, 2022a; Dong *et al.*, 2023]. As illustrated in Figure 4, Skip-Timeformer, STInformer, STFlowformer, and STFlashformer, through skip-

Models			Transformer		Informer		Flowformer		Flashformer	
Metric			MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	Original	96	0.989	0.787	0.873	0.707	0.889	0.728	0.936	0.762
		192	0.886	0.743	0.416	0.424	0.798	0.688	0.879	0.737
		336	0.978	0.805	0.451	0.444	0.889	0.756	0.970	0.801
		720	1.159	0.895	0.494	0.48	1.608	0.833	1.165	0.899
		Avg	1.003	0.807	0.558	0.513	1.046	0.751	0.987	0.799
	Skip-time	96	0.372	0.396	0.382	0.404	0.373	0.391	0.382	0.404
		192	0.416	0.424	0.423	0.427	0.418	0.421	0.434	0.436
		336	0.451	0.444	0.458	0.445	0.448	0.437	0.467	0.451
		720	0.471	0.471	0.464	0.462	0.462	0.460	0.494	0.477
		Avg	**0.427**	**0.433**	**0.431**	**0.434**	**0.425**	**0.427**	**0.444**	**0.442**
Weather	Original	96	0.395	0.427	0.300	0.384	0.304	0.383	0.388	0.425
		192	0.619	0.56	0.598	0.544	0.570	0.554	0.619	0.560
		336	0.689	0.594	0.578	0.523	0.707	0.601	0.698	0.600
		720	0.926	0.710	1.059	0.741	0.950	0.741	0.930	0.711
		Avg	0.657	0.572	0.633	0.548	0.632	0.569	0.658	0.574
	Skip-time	96	0.171	0.212	0.174	0.215	0.180	0.219	0.176	0.219
		192	0.219	0.254	0.222	0.255	0.227	0.259	0.222	0.256
		336	0.274	0.295	0.278	0.296	0.281	0.297	0.276	0.296
		720	0.353	0.346	0.355	0.346	0.358	0.348	0.355	0.347
		Avg	**0.254**	**0.276**	**0.257**	**0.278**	**0.261**	**0.280**	**0.257**	**0.279**
Exchange	Original	96	0.686	0.637	0.795	0.713	0.648	0.616	0.671	0.632
		192	1.029	0.786	1.035	0.818	1.197	0.849	1.022	0.786
		336	1.993	1.158	1.518	0.995	2.081	1.193	1.982	1.156
		720	1.125	0.940	2.098	1.163	1.920	1.150	1.341	0.962
		Avg	1.208	0.880	1.361	0.922	1.461	0.952	1.254	0.884
	Skip-time	96	0.079	0.197	0.081	0.199	0.079	0.198	0.083	0.201
		192	0.167	0.291	0.17	0.297	0.172	0.295	0.174	0.298
		336	0.316	0.407	0.322	0.411	0.325	0.412	0.327	0.415
		720	0.787	0.666	0.824	0.680	0.795	0.670	0.838	0.688
		Avg	**0.337**	**0.390**	**0.349**	**0.396**	**0.342**	**0.393**	**0.355**	**0.400**

Table 3: The skip-time interaction forecasting framework significantly improves the performance in several popular models.

Models		STTransformer		STInformer		STFlowformer		STFlashformer	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Weather	48	0.211	0.248	0.205	0.238	0.209	0.239	0.178	0.222
	96	0.171	0.212	0.174	0.215	0.180	0.219	0.176	0.219
	192	0.159	0.202	0.160	0.205	0.160	0.203	0.160	0.202
	336	0.153	0.202	0.150	0.200	0.151	0.200	0.152	0.201
	720	0.153	0.208	0.149	0.202	0.145	0.196	0.149	0.200

Table 4: We evaluate the predictive performance for lookback lengths $T \in \{48, 96, 192, 336, 720\}$ with a fixed prediction length $S = 96$. While the performance of Transformer-based predictors may not necessarily benefit from an increase in lookback length, our skip-time forecasting framework enhances the performance of Transformer-based models on an extended lookback window.

time interaction, learn representations more conducive to long sequence series contexts, consequently leading to more accurate predictions. The results further indicate that Skip-Timeformer merits a fundamental reconsideration of the predictive framework.

Partial Sequence Token Analysis

In our designed Skip-Timeformer, training can become cumbersome and challenging when dealing with input sequences that are excessively long, owing to the secondary complexity of self-attention. To address this challenge, in addition to introducing the skip-time interaction forecasting mechanism, we also leverage advanced techniques from previously embedded multi-skip sequence tokens. Specifically, we strategically opt to train the model using only a subset of the sequences.

Experimental results, as demonstrated in Table 6, prove that the performance of the model with partial sequence tokens is only slightly inferior to that with the full sequence when handling exchange rate datasets. Simultaneously, this approach successfully reduces memory consumption, providing robust support for the efficient training of the model.

Models		Skip-Transformer						TimesNet	
		MST+STIF		MST		STIF			
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	0.372	0.396	0.380	0.396	0.381	0.403	0.384	0.402
	192	0.416	0.424	0.432	0.428	0.429	0.435	0.436	0.429
	336	0.451	0.444	0.467	0.444	0.461	0.448	0.491	0.469
	720	0.471	0.471	0.490	0.474	0.494	0.480	0.521	0.500
ETTh2	96	0.287	0.338	0.292	0.342	0.433	0.424	0.340	0.374
	192	0.359	0.385	0.362	0.387	0.364	0.388	0.402	0.414
	336	0.406	0.425	0.424	0.439	0.414	0.429	0.452	0.452
	720	0.411	0.434	0.420	0.442	0.421	0.441	0.462	0.468
Exchange	96	0.079	0.197	0.082	0.201	0.080	0.198	0.107	0.234
	192	0.167	0.291	0.172	0.294	0.174	0.297	0.226	0.344
	336	0.316	0.407	0.323	0.412	0.327	0.414	0.367	0.448
	720	0.776	0.661	0.789	0.666	0.805	0.672	0.964	0.746
Weather	96	0.171	0.212	0.175	0.214	0.172	0.214	0.172	0.220
	192	0.219	0.254	0.223	0.256	0.221	0.256	0.219	0.261
	336	0.274	0.295	0.279	0.297	0.275	0.295	0.280	0.306
	720	0.353	0.346	0.356	0.348	0.350	0.344	0.365	0.359

Table 5: Multi-skip sequence token embedding and skip-time interaction forecasting in Skip-Timeformer experimental analysis.

Models		25%		50%		75%		100%	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Exchage	96	0.084	0.203	0.080	0.200	0.084	0.203	0.079	0.197
	192	0.175	0.300	0.171	0.294	0.172	0.298	0.167	0.291
	336	0.317	0.409	0.317	0.410	0.318	0.412	0.316	0.407
	720	0.793	0.675	0.792	0.675	0.791	0.675	0.787	0.666

Table 6: The proportion of the subsequence to the entire sequence is 25%, 50%, 75%, 100%. The performance of training variables in each subsequence is only slightly inferior, but the memory usage during the training process significantly decreases as the proportion decreases.

5 Conclusion

In this paper, we highlight that the conventional architecture of Transformer is not suitable for discovering fundamental sequence representations and inter-sequence correlations in long sequence time-series forecasting. Our approach is specifically designed to address the unique data characteristics of the LSTF dataset. Specifically, the multi-skip sequence token embedding embeds input data into a one-dimensional vector array, preserving information across multiple time dimensions. The skip-time interaction forecasting aims to capture dependencies across time dimensions of the embedded array. Through extensive experiments, we demonstrate that skip-time interaction forecasting enhances the model's understanding of long sequence patterns from highly similar training samples. Additionally, skip-time interaction forecasting effectively addresses overfitting issues in LSTF. Across eight popular datasets, Skip-Timeformer achieves state-of-the-art performance in long sequence forecasting. Compared to previous work, it benefits from longer lookback windows and demonstrates remarkable framework generality in various model scenarios. In the future, we plan to explore the application of Skip-Timeformer in diverse time series analysis tasks.

Acknowledgments

This work was supported in part by the following: the National Natural Science Foundation of China under Grant No. 62272281, the Special Funds for Taishan Scholars Project(tsqn202306274), the Youth Innovation Technology Project of Higher School in Shandong Province under Grant

No. 2023KJ212, and the NSFC Joint Fund with Zhejiang Integration of Informatization and Industrialization under Key Project(U22A2033).

Contribution Statement

Wenchang Zhang and Hua Wang contributed equally to this work and served as co-first authors†. All authors were actively involved in methodology design, research execution, data analysis, and manuscript preparation.

References

[Angryk *et al.*, 2020] Rafal A Angryk, Petrus C Martens, Berkay Aydin, Dustin Kempton, Sushant S Mahajan, Sunitha Basodi, Azim Ahmadzadeh, Xumin Cai, Soukaina Filali Boubrahimi, Shah Muhammad Hamdi, et al. Multivariate time series dataset for space weather data analytics. *Scientific data*, 7(1):227, 2020.

[Box and Jenkins, 1968] George EP Box and Gwilym M Jenkins. Some recent advances in forecasting and control. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 17(2):91–109, 1968.

[Dao *et al.*, 2022] Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in Neural Information Processing Systems*, 35:16344–16359, 2022.

[Demirel *et al.*, 2012] Ömer Fahrettin Demirel, Selim Zaim, Ahmet Çalişkan, and Pinar Özuyar. Forecasting natural gas consumption in istanbul using neural networks and multivariate time series methods. *Turkish Journal of Electrical Engineering and Computer Sciences*, 20(5):695–711, 2012.

[Devlin *et al.*, 2018] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[Dong *et al.*, 2023] Jiaxiang Dong, Haixu Wu, Haoran Zhang, Li Zhang, Jianmin Wang, and Mingsheng Long. Simmtm: A simple pre-training framework for masked time-series modeling. *arXiv preprint arXiv:2302.00861*, 2023.

[Jin *et al.*, 2023] Ming Jin, Guangsi Shi, Yuan-Fang Li, Qingsong Wen, Bo Xiong, Tian Zhou, and Shirui Pan. How expressive are spectral-temporal graph neural networks for time series forecasting? *arXiv preprint arXiv:2305.06587*, 2023.

[Kalyan *et al.*, 2021] Katikapalli Subramanyam Kalyan, Ajit Rajasekharan, and Sivanesan Sangeetha. Ammus: A survey of transformer-based pretrained models in natural language processing. *arXiv preprint arXiv:2108.05542*, 2021.

[Khan *et al.*, 2022] Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. Transformers in vision: A survey. *ACM computing surveys (CSUR)*, 54(10s):1–41, 2022.

[Kornblith *et al.*, 2019] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International conference on machine learning*, pages 3519–3529. PMLR, 2019.

[Li *et al.*, 2019] Shiyang Li, Xiaoyong Jin, Yao Xuan, Xiyou Zhou, Wenhu Chen, Yu-Xiang Wang, and Xifeng Yan. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. *Advances in neural information processing systems*, 32, 2019.

[Lim and Zohren, 2021] Bryan Lim and Stefan Zohren. Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A*, 379(2194):20200209, 2021.

[Liu *et al.*, 2021] Shizhan Liu, Hang Yu, Cong Liao, Jianguo Li, Weiyao Lin, Alex X Liu, and Schahram Dustdar. Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting. In *International conference on learning representations*, 2021.

[Liu *et al.*, 2022] Xin Liu, Junhong Guo, Hua Wang, and Fan Zhang. Prediction of stock market index based on issa-bp neural network. *Expert Systems with Applications*, 204:117604, 2022.

[Nie *et al.*, 2022] Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*, 2022.

[Patton, 2013] Andrew Patton. Copula methods for forecasting multivariate time series. *Handbook of economic forecasting*, 2:899–960, 2013.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[Wang *et al.*, 2023] Min Wang, Hua Wang, and Fan Zhang. Famc-net: Frequency domain parity correction attention and multi-scale dilated convolution for time series forecasting. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 2554–2563, 2023.

[Woo *et al.*, 2022] Gerald Woo, Chenghao Liu, Doyen Sahoo, Akshat Kumar, and Steven Hoi. Etsformer: Exponential smoothing transformers for time-series forecasting. *arXiv preprint arXiv:2202.01381*, 2022.

[Wu *et al.*, 2020] Sifan Wu, Xi Xiao, Qianggang Ding, Peilin Zhao, Ying Wei, and Junzhou Huang. Adversarial sparse transformer for time series forecasting. *Advances in neural information processing systems*, 33:17105–17115, 2020.

[Wu *et al.*, 2021] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in Neural Information Processing Systems*, 34:22419–22430, 2021.

[Wu *et al.*, 2022a] Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. *arXiv preprint arXiv:2210.02186*, 2022.

[Wu *et al.*, 2022b] Haixu Wu, Jialong Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Flowformer: Linearizing transformers with conservation flows. *arXiv preprint arXiv:2202.06258*, 2022.

[Zeng *et al.*, 2023] Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 11121–11128, 2023.

[Zhang and Yan, 2022] Yunhao Zhang and Junchi Yan. Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. In *The Eleventh International Conference on Learning Representations*, 2022.

[Zhang *et al.*, 2023a] Fan Zhang, Gongguan Chen, Hua Wang, Jinjiang Li, and Caiming Zhang. Multi-scale video super-resolution transformer with polynomial approximation. *IEEE Transactions on Circuits and Systems for Video Technology*, 2023.

[Zhang *et al.*, 2023b] Fan Zhang, Tiantian Guo, and Hua Wang. Dfnet: Decomposition fusion model for long sequence time-series forecasting. *Knowledge-Based Systems*, 277:110794, 2023.

[Zhang *et al.*, 2024a] Fan Zhang, Gongguan Chen, Hua Wang, and Caiming Zhang. Cf-dan: Facial-expression recognition based on cross-fusion dual-attention network. *Computational Visual Media*, pages 1–16, 2024.

[Zhang *et al.*, 2024b] Wenchang Zhang, Hua Wang, and Fan Zhang. Spatio-temporal fourier enhanced heterogeneous graph learning for traffic forecasting. *Expert Systems with Applications*, 241:122766, 2024.

[Zhou *et al.*, 2021] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 11106–11115, 2021.

[Zhou *et al.*, 2022] Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International Conference on Machine Learning*, pages 27268–27286. PMLR, 2022.