*2025 International Conference on Intelligent Computing*
*July 26-29, Ningbo, China*
https://www.ic-icc.cn/2025/index.php

# MWDN:A long time series forecasting framework based on multi-scale wavelet decomposition network

Xingjie Feng and Jingyao Sun(✉) and Jiaxi Chen

College of Computer Science and Technology, Civil Aviation University of China, Tianjin 300300, China
xjfeng@cauc.edu.cn
jingyaosun01@163.com
jiaxichen12@163.com

**Abstract.** Long-term time series forecasting remains a significant challenge due to complex temporal dependencies, scale variability, and noise interference. Existing deep learning methods often struggle to capture fine-grained temporal features, particularly in multivariate scenarios where spatio-temporal correlations vary across different resolutions. To address these limitations, we propose MWDN (multi-scale wavelet decomposition network), a novel forecasting framework that integrates multi-scale decomposition with frequency-aware modeling. MWDN employs a wavelet-based module to iteratively decompose the input into detail and approximation sequences, effectively separating seasonal and trend components. These are then processed in parallel via a dual-branch architecture, enabling efficient modeling of variable dependencies across frequencies. To further enhance representation, a multi-scale fusion module aggregates information across resolutions, improving prediction accuracy while mitigating information loss. Extensive experiments on multiple benchmark datasets show that MWDN consistently achieves state-of-the-art or second-best performance on both short- and long-term forecasting tasks. Ablation studies validate the effectiveness of the decomposition strategy and architectural design. MWDN offers a robust and scalable solution for multivariate time series forecasting. The source code is publicly available at: https://github.com/take-off-ddl/MWDN.

**Keywords:** Long-term Time Series Forecasting, Wavelet Decomposition, Multi-scale Modeling, Spatio-temporal Dependency.

## 1 introduction

Long-term time series forecasting has become a pivotal technology for enabling intelligent decision-making across a wide range of domains. In the transportation sector, accurate traffic flow prediction [1] supports real-time congestion mitigation and route optimization. Meteorological agencies depend on precise weather forecasts [2] to enhance disaster preparedness and guide agricultural planning. In finance, stock market prediction [3] is essential for developing resilient investment strategies in volatile trading environments. The energy industry, in particular, benefits from electricity demand

forecasting [4] to maintain grid stability and optimize power distribution. These mission-critical applications underscore the growing demand for advanced forecasting architectures capable of capturing complex temporal dynamics over extended horizons.

Traditionally, time series forecasting [5,6] has been dominated by statistical methods. However, with the emergence of large-scale datasets and the growing availability of high-performance computing resources, deep learning approaches have gained significant attention due to their superior performance on complex forecasting tasks. The development of diverse architectures—such as recurrent neural networks (RNNs), convolutional neural networks (CNNs), and Transformer-based models—has further expanded the landscape of modeling strategies for time series data.

Multivariate time series inherently exhibit complex spatio-temporal dependencies. Accurately and efficiently modeling both inter-variable correlations and underlying temporal dynamics remains a fundamental challenge in this field. Specifically, long-term time series often display intricate patterns such as seasonality, periodicity, and long-term trends [7]. Additionally, ongoing stochastic fluctuations and dynamic external influences introduce diverse temporal behaviors across multiple scales (hours, days, weeks, and months). As the forecasting horizon extends, model performance frequently deteriorates, often leading to significant increases in MSE and MAE values.

Our research is motivated by the need to enhance the modeling of spatio-temporal and inter-variable dependencies for long-term forecasting. Previous work has shown that multi-scale decomposition of time series is an effective strategy for extracting meaningful features. Notably, Autoformer [8] proposed decomposing a time series into trend and seasonal components, resulting in improved forecasting accuracy. In real-world scenarios, many physical systems simultaneously exhibit both trend and seasonal characteristics, each associated with distinct frequency patterns.

In this paper, we introduce a novel multi-scale decomposition approach that facilitates fine-grained feature extraction across different resolutions. We propose a multi-scale wavelet decomposition network (MWDN) for long-term time series forecasting to address the aforementioned challenges. To capture the complex patterns in both the time and frequency domains, we leverage wavelet decomposition to unify analysis across these domains. Specifically, MWDN employs multi-level wavelet decomposition to split the original time series into multiple approximation and detail sequences. To prevent noise amplification from over-decomposition, we adopt an iterative decomposition strategy. For seasonal-trend separation, we introduce adaptive filters to disentangle these two components. A dual-branch architecture is then used to independently model the seasonal and trend sequences, allowing the model to more effectively understand and extract representative features from each. To integrate multi-resolution features, we further propose a multi-scale fusion module (MFM) that efficiently aggregates diverse representations, preserving valuable spatio-temporal and inter-variable dependencies. Our main contributions can be summarized as follows:

- We introduce a novel deep forecasting model, MWDN, which integrates multi-scale wavelet decomposition with three core components: the Wavelet Multi-scale Decomposition Module (WMDM), the Multi-Resolution Feature Extraction Module (MFEM), and the Multi-Scale Fusion Module (MFM). WMDM employs an iterative

decomposition strategy to suppress noise and enhance robustness, while MFEM uses a dual-branch parallel structure to independently model seasonal and trend components, efficiently capturing both variable and spatio-temporal dependencies with reduced computational cost.

- We propose a resolution-aware framework that processes time series at different scales using distinct resolution branches. The MFM effectively fuses cross-resolution features to minimize noise interference and preserve essential information, thereby enhancing the expressiveness of the learned representations.
- Extensive experiments on multiple benchmark datasets demonstrate that MWDN consistently achieves state-of-the-art (SOTA) performance in both short-term and long-term forecasting tasks, validating its effectiveness and generalizability across diverse domains.

## 2 Related Work

Time series forecasting has been a central focus across various fields, including finance, healthcare, and environmental science. Classical methods, such as [9], Holt-Winters [10], and Prophet [11], rely on predefined statistical assumptions to model temporal patterns. While effective for simple and stationary time series, these methods struggle with the complexity and non-stationarity of real-world data, limiting their applicability in dynamic and nonlinear scenarios.

In recent years, deep learning methods have significantly advanced time series modeling. RNN-based approaches [12,13] leverage recurrent architectures to capture sequential dependencies but encounter difficulties in modeling long-term patterns due to vanishing gradients. On the other hand, TCN-based models [14] address this issue by employing convolutional kernels to capture temporal variations over extended horizons. MLP-based methods [15,16] further enhance modeling capacity by encoding temporal dependencies into fixed parameter spaces. More recent variants have introduced additional mechanisms for improved feature extraction: TPA-LSTM [17] incorporates attention to capture hierarchical temporal patterns, while DeepTCN [18] utilizes dilated convolutions for multi-scale representation learning. However, these models still lack effective methods to disentangle mixed-frequency components, which are essential for modeling real-world signals involving overlapping periodicities. Despite the progress made, these methods continue to struggle with fully addressing the multi-scale dynamics inherent in time series data, underscoring the need for architectures specifically designed to handle such complexities.

Transformer-based architectures have emerged as a powerful alternative for time series forecasting. By leveraging self-attention mechanisms, these models effectively capture long-range dependencies. Autoformer [8] introduces a decomposition framework that separately models seasonal and trend components, while FEDformer [19] incorporates frequency-domain information to improve efficiency and scalability. PatchTST [20] utilizes patching mechanisms to extract both local and global temporal features, and Crossformer [21] leverages hierarchical embeddings to model cross-time and cross-variable dependencies. These advancements highlight the versatility of

Transformer-based models; however, their reliance on specific decomposition strategies or architectural assumptions limits their generalizability across diverse datasets. Recent innovations aim to bridge this gap: FEDFormer [19] incorporates Fourier transforms for frequency domain learning, while Timesnet [7] proposes adaptive multi-periodicity detection. These hybrid approaches demonstrate the benefits of combining spectral analysis with sequence modeling, though their fixed decomposition strategies limit adaptability to varying data characteristics.

Multi-scale processing plays a crucial role in time series analysis by enabling the separation of signals into components with varying frequencies. Techniques like wavelet transforms provide a flexible framework for decomposing time series into high-frequency details and low-frequency trends, thereby facilitating the extraction of both local and global patterns. Recent works have explored integrating multi-scale decomposition with deep learning architectures. For instance, TimeMixer [22] independently models trend and seasonal components to enhance predictive accuracy, while DLinear [16] adopts a simplified linear decomposition approach. Notable developments also include the introduction of a multi-level wavelet CNN by MICN [23], which effectively captures hierarchical temporal patterns, and the frequency-enhanced Transformer in FEDformer [19], which integrates wavelet analysis into the attention mechanism. These studies demonstrate the potential of combining traditional signal processing techniques with modern machine learning models. However, the effective integration of multi-scale features remains an open challenge: (1) most wavelet networks rely on fixed decomposition scales, and (2) cross-variable dependency modeling is still under-explored.

# 3 Model

In multivariate time series forecasting, we are given a historical observation sequence $\mathbf{X} = \{\mathbf{x_1}, \dots, \mathbf{x_L}\} \in \mathbf{R^{L \times N}}$ with $\mathbf{L}$ time steps and $\mathbf{N}$ variates, The forecasting objective is to predict the future $T$ time steps $Y = \{X_{L+1}, \dots, x_{L+T}\} \in R^{T \times N}$.

## 3.1 Structure Overview

The proposed model architecture is illustrated in **Fig. 1**. Our framework begins by decomposing the normalized multivariate time series using a wavelet transformation module, generating multi-scale sequences that include both approximation and detail coefficient sequences. These multi-resolution representations capture cross-variable feature relationships at different frequency levels, where each decomposition level corresponds to distinct temporal granularities. As the decomposition level increases, the approximation coefficients narrow their frequency bandwidth, producing finer-grained detail coefficients. The selection of decomposition levels and wavelet basis functions is crucial for extracting meaningful multi-resolution features. While higher decomposition levels provide more detailed coefficients, they may also introduce high-frequency noise and irrelevant patterns that can negatively impact prediction accuracy. By introducing wavelet decomposition, not only can the features of time series data at different scales be extracted, but also the impact of noise data on the model can be reduced to a certain extent.

The architecture consists of three core components: (1) Wavelet Multi-scale Decomposition Module (WMDM), (2) Multi-resolution Feature Extraction Module (MFEM), and (3) Multi-scale Fusion Module (MFM). Each resolution branch processes the coefficients through dual decomposition: seasonal components ($X_s$), which capture strong periodic fluctuations (daily/weekly cycles), are processed by multi-receptive field CNNs with hierarchical dilated convolutions to preserve temporal resolution. In contrast, trend components ($X_t$), which represent slow-evolving patterns, are modeled through a Frequency-Enhanced Transformer Encoder architecture that captures long-range dependencies via attention mechanisms. The feature extraction module fuses the seasonal and trend components using learnable parameters, followed by dynamic weight adjustment across scales in the fusion module. Final predictions are reconstructed through an inverse wavelet transform and linear projection, synthesizing information from both approximation and detail coefficients. The subsequent sections elaborate on the implementation details of each component.

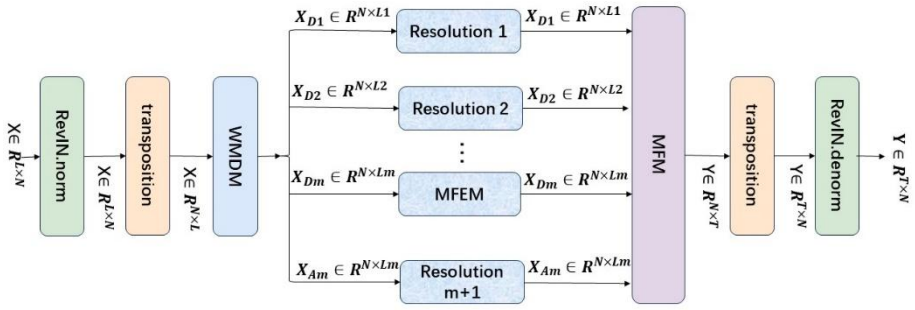


**Fig. 1.** Architecture of the proposed MWDN.

### 3.2 Wavelet Multi-scale Decomposition Module

Previous studies have primarily focused on globally processing time series as a whole, often overlooking the temporal and spatial information embedded at different granularities and resolutions. Decomposing time series components at varying granularities and resolutions can simplify processing modules and enhance the accuracy of model predictions. To address this challenge, we introduce a wavelet-based multi-scale decomposition module. We utilize a multi-level discrete wavelet transform (DWT) to decompose the time series data, which involves an iterative process using low-pass and high-pass filters at multiple levels to extract sequence representations at different granularities and resolutions. The filter coefficients depend on the selected wavelet basis

function. The high-pass filter output is considered detailed information, referred to as detail coefficients and denoted as D, while the low-pass filter output corresponds to low-frequency information, known as approximation coefficients and denoted as $A$. Through an iterative decomposition process, the approximation coefficients from each previous level are further decomposed into new approximation and detail coefficients, enabling deeper sequence analysis. The entire WMDM process for handling time series can be mathematically expressed as:

$$X_{A_m}, X_{D_m}, X_{D_{m-1}}, \dots, X_{D_1} = \mathrm{WMDM}(X, m, \psi) \quad (1)$$

Where X represents the original input time series, $m$ denotes the decomposition level, $A$ represents the low-frequency component obtained through wavelet decomposition, and $D$ represents the high-frequency component. The term $\psi$ refers to the wavelet type, and WMDM refers to the decomposition process. The sequences $\mathrm{X_{A_i}}$ and $\mathrm{X_{D_i}}$ represent the approximation and detail coefficient sequences at the $i$-th level of wavelet decomposition, respectively.

As shown in **Fig. 2**, since we adopt an iterative decomposition approach, the detail coefficient sequence at the m-th level, $\mathrm{X_{D_m}}$, is obtained by decomposing the approximation coefficient sequence at the $(\mathrm{m}-1)$-th level, yielding both the approximation and detail coefficient sequences at the $m$-th level. To prevent excessive noise from contaminating the forecasting task, we retain only the approximation coefficient sequence at the highest decomposition level. A wavelet decomposition at level m produces $\mathrm{m}+1$ outputs: m detail coefficient sequences and one approximation coefficient sequence. To process these $\mathrm{m}+1$ outputs effectively, we employ $\mathrm{m}+1$ branches, each corresponding to a different granularity and resolution. The multi-scale input-output representation is formulated as follows:

$$X \in R^{N \times L}, \qquad X_{A_i} \in R^{N \times L_i}, \qquad X_{D_i} \in R^{N \times L_i}, \qquad i = 1,2,\dots,m$$

where L represents the original length of the time series, N denotes the number of variables, and $\mathrm{L_i}$ corresponds to the reduced sequence length after decomposition.

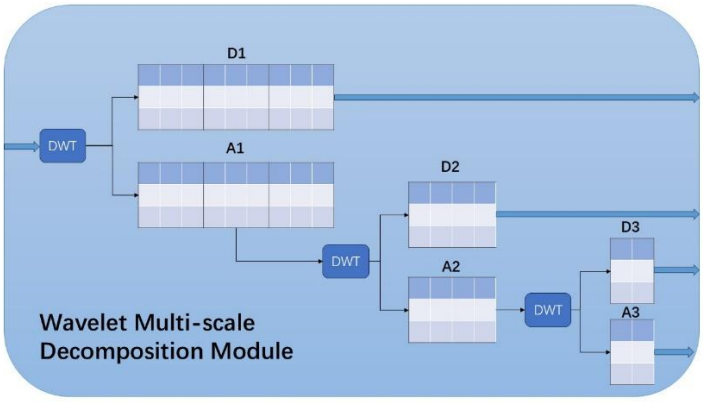


**Fig. 2.** Architecture of the WMDM.

### 3.3 Multi-resolution Feature Extraction Module

After applying the Multi-resolution Wavelet Decomposition Module (MWDM), the original time series is decomposed into $m + 1$ sub-sequences at varying resolutions and granularities, consisting of one approximation coefficient sequence and m detail coefficient sequences. These sub-sequences represent the low-frequency (coarse-scale) and high-frequency (fine-scale) components of the original series, respectively. To effectively extract frequency-specific features at multiple resolutions, we introduce a Multi-resolution Feature Extraction Module. Unlike traditional seasonal-trend decomposition methods[19,24], which rely on predefined filters and assumptions, we propose a learnable adaptive frequency filtering mechanism that can further disentangle the high-frequency and low-frequency content from each sub-sequence produced by the MWDM.

In this context, the high-frequency components are typically interpreted as seasonal patterns, capturing short-term fluctuations, while the low-frequency components correspond to trends, representing long-term dynamics. To better model these distinct temporal behaviors, we process the seasonal and trend components separately, thereby enhancing the model's ability to capture both short-term variations and long-term dependencies across multivariate time series. The proposed filtering operation is implemented via Fast Fourier Transform (FFT) and Inverse FFT (IFFT), enabling frequency-domain decomposition. Specifically, we apply a Discrete Fourier Transform (DFT) to the input series, followed by trainable low-pass and high-pass filters. These filters are parameterized by learnable cutoff frequencies and steepness factors, allowing the model to dynamically adapt the frequency separation for each individual feature. The decomposition process is formally defined as follows:

**FFT.** The Discrete Fourier Transform (DFT) maps the input sequence $\mathbf{X_t}$ from the time domain to the frequency domain, facilitating spectral analysis. The Fast Fourier Transform (FFT) can be used to efficiently compute:

$$\mathcal{F}(X) = \sum_{n=0}^{L-1} X_{t-L+n} \cdot e^{-j\frac{2\pi n}{L}} \tag{2}$$

Where X denotes a L-length window extracted from the original time series. The transformed frequency representation is then given by $\mathrm{X_{freq}} = \mathrm{FFT(X)}$.

**Adaptive Frequency Filter.** The input signal is decomposed into a high-frequency seasonal component **S** and a low-frequency trend component **T** via adaptive frequency-domain filtering. Both filters are constructed using a trainable cutoff frequency $f_{\mathrm{cut}}$ and a sharpness control parameter $\lambda$, with the sigmoid function $\sigma$ enabling a smooth transition between passed and suppressed frequencies.

The seasonal component S is extracted using an adaptive high-pass filter:

$$S = \mathcal{F}^{-1}\left(X_{\mathrm{fe}} \cdot \sigma\big((f - f_{\mathrm{cut}}) \cdot \lambda\big)\right) \tag{3}$$

Similarly, the trend component T is obtained using an adaptive low-pass filter:

$$T = \mathcal{F}^{-1}\big(X_{\mathrm{fe}} \cdot \sigma(-(f - f_{\mathrm{cut}}) \cdot \lambda)\big) \tag{4}$$

Here, $\mathcal{F}^{-1}$ denotes the inverse Fourier transform, and f represents the frequency domain variable.

The proposed frequency-based filter employs a learnable low-pass filter to capture long-term trends and a high-pass filter to extract short-term seasonal variations. This adaptive approach enables the model to separate temporal patterns across frequency bands, enhancing forecasting stability and accuracy. Unlike traditional methods with fixed heuristics, the trainable filters allow the model to automatically learn trend and periodic structures from data, improving flexibility and generalization across diverse scenarios.

To further exploit the decomposed components, a dual-branch architecture is adopted, where each branch independently models the trend or seasonal signal. As shown in **Fig. 3**, this parallel design preserves distinct temporal dynamics while enabling efficient, specialized processing.

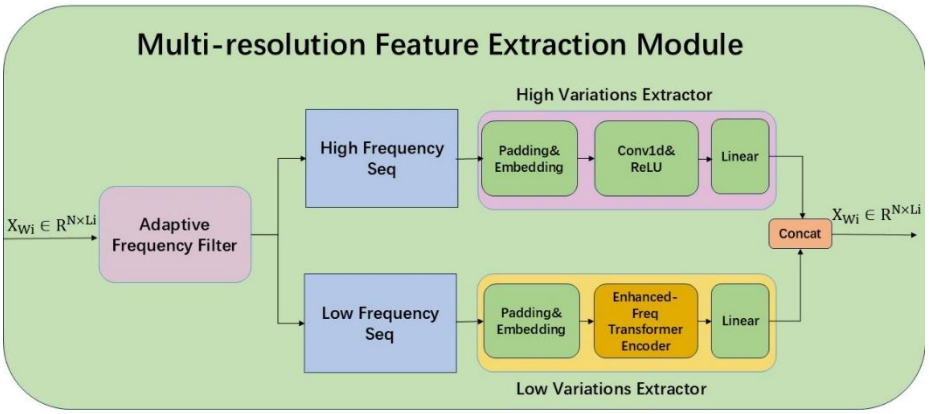


**Fig. 3.** Architecture of the proposed MFEM.

**Trend Variations Extractor.** To capture low-frequency variations and long-term dependencies within the time series, we design a Trend Variations Extractor based on a Transformer architecture enhanced with frequency-aware attention mechanisms. This module focuses on extracting trend-related features from the low-frequency component of the decomposed time series. Given a low-frequency time series extracted via the frequency-domain filters, denoted as $x_t \in \mathbf{R}^{N\times L_i}$, we first apply zero-padding and a linear projection to aligned the sequence with the model's input dimensions and embed it into a higher-dimensional latent space to facilitate the extraction of complex dependencies:

$$X_t = \text{Padding}(x_t) \in R^{N\times P} \quad (5)$$

$$X_t = \text{Embedding}(X_t) \in R^{N\times D} \quad (6)$$

Here, P denotes the padded sequence length, and D is the dimension of the latent space. The embedding step increases the representational capacity of the input, allowing the model to better learn complex patterns in subsequent stages.

Next, the embedded sequence is passed through a stack of frequency-enhanced Transformer Encoder layers, where each layer performs the following operations:

$$X_t = \text{BatchNorm}\big(X_t + \text{FEMSA}(X_t, X_t, X_t)\big) \tag{7}$$

$$X_t = \text{BatchNorm}\left(X_g + \text{MLP}(X_t)\right) \tag{8}$$

In this context, $\text{BatchNorm}(\cdot)$ denotes batch normalization[25], which stabilizes and accelerates training. $\text{MLP}(\cdot)$ is a multi-layer feedforward neural network that performs nonlinear feature transformations. As shown in **Fig. 4**, The **Frequency-Enhanced Multi-Head Self-Attention (FEMSA)** mechanism leverages multiple independent attention heads, each of which is enhanced to emphasize relevant frequency patterns. This allows the model to capture diverse long-term dependencies across temporal patches more effectively. Each head can attend to different frequency components or temporal structures, and the outputs are subsequently aggregated to form a comprehensive representation of the sequence.

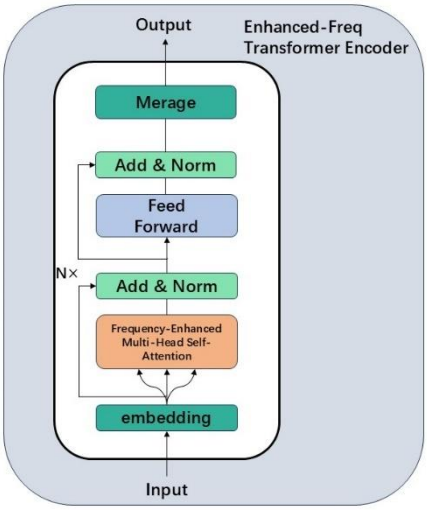


**Fig. 4.** Architecture of the frequency-enhanced multi-head self-attention.

Finally, a linear projection is applied to transform the output into the target dimensionality for downstream forecasting tasks:

$$X_t = \text{Linear}(X_t) \in R^{N \times D} \tag{9}$$

This architecture ensures that long-term trend patterns are effectively captured and modeled for use in multi-scale forecasting.

**Season Variations Extractor.** To capture high-frequency fluctuations and short-term dependencies within the time series, we design a dedicated extractor tailored for

seasonality-related features. These components often reflect abrupt changes, local periodicities, or anomalous behaviors that cannot be effectively modeled using global trend extractors. Therefore, we utilize a convolution-based structure with diverse kernel sizes to extract such local and periodic patterns.

Given the high-frequency component extracted via the frequency filter, denoted as $x_s \in R^{N \times L_i}$ , we first pad and project it into a high-dimensional representation to facilitate subsequent feature extraction:

$$X_s = \text{Padding}(x_s) \in R^{N \times P} \tag{10}$$

$$X_s = \text{Embedding}(X_s) \in R^{N \times D} \tag{11}$$

To capture local dependencies at multiple temporal scales, we apply a series of 1D convolutional blocks with varying kernel sizes. These blocks are stacked to progressively expand the receptive field:

$$X_s = \text{ReLU}\big(\text{Conv1d}(X_s)\big) \tag{12}$$

These high-frequency signals often correspond to short-term variations such as seasonal fluctuations or outlier events, which are difficult to identify through long-term modeling alone. The convolution-based structure complements the global trend extractor and enhances the model's capability in capturing dynamic local patterns and rapid transitions.

Finally, the outputs from both the trend and seasonal branches are concatenated and linearly projected to ensure dimensional consistency for subsequent multiresolution fusion. The final feature representation of the multiscale feature extraction module (MFEM) is defined as:

$$X_{W_i} = \text{Linear}\big(\text{Concat}(X_t, X_s)\big) \in R^{N \times L_i}, i = 1,2, \dots, m \tag{13}$$

Here, $X_{W_i}$ serves as the unified feature embedding, preserving both short-term and long-term dynamics across resolutions, and is fed into the subsequent Multi-resolution Fusion Module (MFM) for final prediction.

### 3.4 Multi-scale Fusion Module

To fully exploit the multi-resolution characteristics inherent in time series data, we design the Multi-scale Fusion Module (MFM). This module receives feature representations extracted at multiple temporal resolutions, each corresponding to different window lengths $L_1, L_2, \dots, L_m$. The feature representation at the i-th resolution is denoted as:

$$X_{W_i} \in R^{N \times L_i}, \qquad i = 1,2, \dots, m$$

where N denotes the number of variables, and $L_i$ indicates the temporal resolution at the i-th scale. Each representation integrates both long-term trends (from the Trend Variations Extractor) and short-term seasonal dependencies (from the Seasonal Variations Extractor).

To effectively integrate information across scales, we first apply the Inverse Discrete Wavelet Transform (IDWT) to restore the original temporal structure:

$$X_{D_1}, X_{D_2}, \dots, X_{D_m}, X_{A_m} = \text{IDWT}\big(X_{D_1}, X_{D_2}, \dots, X_{D_m}, X_{A_m}\big) \tag{14}$$

Next, all reconstructed features are concatenated along the feature axis and projected to a unified output dimension T using a linear transformation:

$$Y = \text{Linear}\left(\text{Concat}\left(X_{D_1}, X_{D_2}, \ldots, X_{D_m}, X_{A_m}\right)\right), Y \in R^{N \times T} \quad (15)$$

After feature fusion, the output $Y \in R^{N \times T}$ is transposed to align the time dimension, followed by an inverse transformation using Reversible Instance Normalization (RevIN) to restore the original value scale. The process is defined as:

$$Y_{trans} = \text{Transpose}(Y) \in R^{T \times N} \quad (16)$$

$$\hat{Y} = \text{RevIN}^{-1}(Y_{trans}) \quad (17)$$

Here, $\text{RevIN}^{-1}(\cdot)$ denotes the inverse normalization operation, which utilizes the original mean and standard deviation of each instance for de-normalization. This ensures both numerical consistency with the original input and the physical interpretability of the predicted outputs.

# 4 Experiments

## 4.1 Experimental Setup

**Datasets.** To comprehensively evaluate the performance of the proposed model on multivariate time series forecasting tasks, we conduct experiments on nine widely-used public datasets. We follow standardized preprocessing procedures adopted in previous studies to ensure fair comparison. A summary of the dataset specifications is provided in **Table 1**.

**Table 1.** Datasets for long term forecasting tasks.

| Datasets | Frequency | Variables | Length |
|---|---|---|---|
| ETTh1/ETTh2 | Hourly | 7 | 17,420 |
| ETTm1/ETTm2 | 15 minutes | 7 | 69,680 |
| Weather | 10 minutes | 21 | 52,696 |
| Electricity | Hourly | 321 | 26,304 |
| Traffic | Hourly | 862 | 17,544 |

**Baselines and Evaluation Metrics.** To validate the effectiveness of our model, **MWPM**, we benchmark it against a comprehensive set of state-of-the-art time series forecasting models. These include **PatchTST(2023)**, **TimesNet(2023)**, **DLinear(2023)**, **FEDformer(2022)**, **Autoformer(2021)**, and **Informer(2021)**, covering a diverse range of modeling paradigms. For evaluation, we adopt two standard metrics widely used in prior literature: Mean Squared Error **(MSE)** and Mean Absolute Error **(MAE)**.

**Implementation Details.** All experiments are implemented using the PyTorch framework and conducted on a single NVIDIA GeForce RTX 2080Ti GPU with 11GB memory. Following standard practice in previous works such as Informer, Autoformer and PatchTST, we normalize all datasets to zero mean and unit variance before training.

The model is evaluated under a long-term forecasting setting with prediction lengths of 96, 192, 336, and 720.

We use a batch size of 16 and train for a maximum of 30 epochs, employing early stopping based on the validation loss to prevent overfitting. The Adam optimizer is utilized for optimization. MSE is used both as the training loss function and as one of the evaluation metrics, along with MAE. All other hyperparameters are aligned with those used in recent time series forecasting benchmarks to ensure a fair and consistent comparison.

### 4.2 Multivariate Long-Term Forecasting Results

To evaluate the effectiveness of our proposed model in long-term time series forecasting, we follow prior studies and report results on seven widely-used benchmark datasets across four forecasting horizons (96, 192, 336, and 720). **Table 2** presents a comprehensive comparison between our method and several state-of-the-art baselines. Across a total of 56 evaluation points (7 datasets × 4 horizons × 2 metrics), our model consistently outperforms most Transformer-based and MLP-based architectures in terms of both MSE and MAE. For example, on the ETTh1 dataset with prediction lengths of 336 and 720, our model achieves average MSE values of 0.426 and 0.449, respectively, outperforming strong baselines such as PatchTST and TimesNet. On the ETTm1 dataset, our model surpasses the previous best result (0.387), achieving a new state-of-the-art with an MSE of 0.341—an improvement of 11.1%. Similarly, on the Electricity dataset with a prediction length of 720, both MSE and MAE results show significant improvements over existing methods, highlighting the superior forecasting ability of our model in electricity demand scenarios.

On the Weather dataset, our method yields notable performance gains at prediction horizons of 192 and 336, achieving MSE values of 0.208 and 0.266, respectively. These represent 5% and 4% improvements over the previous best results (0.219 and 0.278), clearly demonstrating the robustness and generalizability of our approach across varying temporal patterns.

These experimental results suggest that the proposed dual-branch parallel module, which separately models short- and long-term dependencies, effectively reduces forecasting errors over long horizons, enhances trend extraction, and improves model robustness. Additionally, the use of wavelet-based multi-scale and multivariate decomposition allows the model to capture cross-variable relationships at varying temporal resolutions, thereby improving the modeling of spatiotemporal dependencies in time series data. While the performance gain on the Traffic dataset is relatively marginal, we attribute this to the complex spatiotemporal dynamics inherent in traffic data. In such cases, adaptive frequency decomposition may introduce noise, slightly affecting prediction accuracy. Nonetheless, our model still achieves the best or second-best performance across the majority of datasets, underscoring its strong adaptability and effectiveness in a wide range of time series forecasting scenarios.

### 4.3 Ablation Studies

To comprehensively assess the effectiveness of each component within our proposed architecture, we conduct ablation studies on the ETT datasets. All models are trained

**Table 2.** Multivariate long-term forecasting results. Four commonly used prediction lengths (96,192,336,720) from the literature are considered for each dataset. The **bold** is the best.

| Model | | **MWDN (ours)** | | PatchTST (2023) | | TimesNet (2023) | | DLinear (2023) | | FEDformer (2022) | | Autformer (2021) | | Informer (2021) | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Metric | | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE |
| ETTm1 | 96 | **0.284** | **0.336** | 0.329 | 0.367 | 0.338 | 0.375 | 0.345 | 0.372 | 0.379 | 0.419 | 0.505 | 0.475 | 0.672 | 0.571 |
| | 192 | **0.332** | **0.329** | 0.367 | 0.385 | 0.374 | 0.387 | 0.380 | 0.389 | 0.426 | 0.441 | 0.553 | 0.496 | 0.795 | 0.669 |
| | 336 | **0.370** | **0.335** | 0.399 | 0.410 | 0.410 | 0.411 | 0.413 | 0.413 | 0.445 | 0.459 | 0.621 | 0.537 | 1.212 | 0.871 |
| | 720 | **0.376** | **0.345** | 0.454 | 0.439 | 0.478 | 0.450 | 0.474 | 0.453 | 0.543 | 0.490 | 0.671 | 0.561 | 1.166 | 0.823 |
| | Avg | **0.341** | **0.336** | 0.387 | 0.400 | 0.400 | 0.406 | 0.403 | 0.407 | 0.448 | 0.452 | 0.588 | 0.517 | 0.961 | 0.734 |
| ETTm2 | 96 | 0.234 | 0.280 | **0.175** | **0.259** | 0.187 | 0.267 | 0.193 | 0.292 | 0.203 | 0.287 | 0.255 | 0.339 | 0.365 | 0.453 |
| | 192 | 0.248 | **0.274** | **0.241** | 0.302 | 0.249 | 0.309 | 0.284 | 0.362 | 0.269 | 0.328 | 0.281 | 0.340 | 0.533 | 0.563 |
| | 336 | **0.288** | 0.373 | 0.305 | **0.343** | 0.321 | 0.351 | 0.369 | 0.427 | 0.325 | 0.366 | 0.339 | 0.372 | 1.363 | 0.887 |
| | 720 | **0.352** | **0.396** | 0.402 | 0.400 | 0.408 | 0.403 | 0.554 | 0.522 | 0.421 | 0.415 | 0.433 | 0.432 | 3.379 | 1.388 |
| | Avg | **0.281** | 0.331 | 0.281 | **0.326** | 0.291 | 0.333 | 0.350 | 0.401 | 0.305 | 0.349 | 0.327 | 0.371 | 1.410 | 0.810 |
| ETTh1 | 96 | **0.370** | **0.384** | 0.414 | 0.419 | 0.384 | 0.402 | 0.386 | 0.400 | 0.376 | 0.419 | 0.449 | 0.459 | 0.865 | 0.713 |
| | 192 | **0.392** | **0.410** | 0.460 | 0.445 | 0.436 | 0.429 | 0.437 | 0.432 | 0.420 | 0.448 | 0.500 | 0.482 | 1.008 | 0.792 |
| | 336 | **0.426** | **0.452** | 0.501 | 0.466 | 0.491 | 0.469 | 0.481 | 0.459 | 0.459 | 0.465 | 0.521 | 0.496 | 1.107 | 0.809 |
| | 720 | **0.449** | **0.466** | 0.500 | 0.488 | 0.521 | 0.500 | 0.519 | 0.516 | 0.506 | 0.507 | 0.514 | 0.512 | 1.181 | 0.865 |
| | Avg | **0.422** | **0.441** | 0.469 | 0.454 | 0.458 | 0.450 | 0.456 | 0.452 | 0.440 | 0.460 | 0.496 | 0.487 | 1.040 | 0.795 |
| ETTh2 | 96 | **0.293** | 0.376 | 0.302 | **0.348** | 0.340 | 0.374 | 0.333 | 0.387 | 0.358 | 0.397 | 0.346 | 0.388 | 3.755 | 1.525 |
| | 192 | **0.326** | **0.364** | 0.388 | 0.400 | 0.402 | 0.414 | 0.477 | 0.476 | 0.429 | 0.439 | 0.456 | 0.452 | 5.602 | 1.931 |
| | 336 | **0.390** | 0.444 | 0.426 | **0.433** | 0.452 | 0.452 | 0.594 | 0.541 | 0.496 | 0.487 | 0.482 | 0.486 | 4.721 | 1.835 |
| | 720 | **0.403** | 0453 | 0.431 | **0.446** | 0.462 | 0.468 | 0.831 | 0.657 | 0.463 | 0.474 | 0.515 | 0.511 | 3.647 | 1.625 |
| | Avg | 0.353 | 0.409 | 0.387 | **0.407** | 0.414 | 0.427 | 0.559 | 0.515 | 0.437 | 0.449 | 0.450 | 0.459 | 4.431 | 1.729 |
| Weather | 96 | **0.167** | 0.223 | 0.177 | **0.218** | 0.172 | 0.220 | 0.196 | 0.255 | 0.217 | 0.296 | 0.266 | 0.336 | 0.300 | 0.384 |
| | 192 | **0.208** | **0.251** | 0.225 | 0.259 | 0.219 | 0.261 | 0.237 | 0.296 | 0.276 | 0.336 | 0.307 | 0.367 | 0.598 | 0.544 |
| | 336 | **0.266** | **0.295** | 0.278 | 0.297 | 0.280 | 0.306 | 0.283 | 0.335 | 0.339 | 0.380 | 0.359 | 0.395 | 0.578 | 0.523 |
| | 720 | **0.352** | **0.396** | 0.402 | 0.400 | 0.408 | 0.403 | 0.554 | 0.522 | 0.421 | 0.415 | 0.433 | 0.432 | 3.379 | 1.388 |
| | Avg | **0.245** | **0.278** | 0.259 | 0.281 | 0.259 | 0.287 | 0.265 | 0.317 | 0.309 | 0.360 | 0.338 | 0.382 | 0.634 | 0.548 |
| Electricity | 96 | **0.151** | **0.243** | 0.181 | 0.270 | 0.168 | 0.272 | 0.197 | 0.282 | 0.193 | 0.308 | 0.201 | 0.317 | 0.274 | 0.368 |
| | 192 | **0.174** | **0.270** | 0.188 | 0.274 | 0.184 | 0.289 | 0.196 | 0.285 | 0.201 | 0.315 | 0.222 | 0.334 | 0.296 | 0.386 |
| | 336 | 0.208 | 0.304 | 0.204 | **0.293** | **0.198** | 0.300 | 0.209 | 0.301 | 0.214 | 0.329 | 0.231 | 0.338 | 0.300 | 0.394 |
| | 720 | **0.217** | **0.313** | 0.246 | 0.324 | 0.220 | 0.320 | 0.245 | 0.333 | 0.246 | 0.355 | 0.254 | 0.361 | 0.373 | 0.439 |
| | Avg | **0.188** | **0.283** | 0.205 | 0.290 | 0.192 | 0.295 | 0.212 | 0.300 | 0.214 | 0.327 | 0.227 | 0.338 | 0.311 | 0.397 |
| Traffic | 96 | **0.447** | **0.291** | 0.462 | 0.295 | 0.593 | 0.321 | 0.650 | 0.396 | 0.587 | 0.336 | 0.613 | 0.388 | 0.719 | 0.391 |
| | 192 | 0.493 | 0.332 | **0.466** | **0.296** | 0.617 | 0.336 | 0.598 | 0.370 | 0.604 | 0.373 | 0.616 | 0.382 | 0.696 | 0.379 |
| | 336 | 0.507 | 0.339 | **0.482** | **0.304** | 0.629 | 0.336 | 0.605 | 0.373 | 0.621 | 0.383 | 0.622 | 0.337 | 0.777 | 0.420 |
| | 720 | 0.528 | 0.367 | **0.514** | **0.322** | 0.640 | 0.350 | 0.645 | 0.394 | 0.626 | 0.382 | 0.660 | 0.408 | 0.864 | 0.472 |
| | Avg | 0.494 | 0.332 | **0.481** | **0.304** | 0.620 | 0.336 | 0.625 | 0.383 | 0.610 | 0.376 | 0.628 | 0.379 | 0.764 | 0.416 |

under identical settings, and the proposed modules are incrementally integrated into the backbone network to evaluate their individual contributions to overall performance.

**Wavelet Multi-scale Decomposition Module.** To investigate the contribution of the proposed Wavelet Multi-scale Decomposition Module (WMDM), we conduct ablation experiments on the ETT datasets by varying the decomposition depth. Specifically, we evaluate three configurations with 2, 3, and 4 levels of wavelet decomposition. All experiments are conducted under identical training settings and hyperparameters to ensure a fair comparison. **Table 3** presents the performance across different decomposition levels. The results indicate that a three-level decomposition consistently yields the best forecasting accuracy. For instance, on the ETTm1 dataset with a prediction horizon of 336, the three-level model achieves an MSE of 0.370, outperforming the two-level variant (MSE = 0.375) and the four-level counterpart (MSE = 0.374). Similar trends are observed on other horizons and datasets.

The performance degradation observed with four-level decomposition suggests that excessive decomposition may over-fragment temporal structures, thereby impeding the model's ability to capture coherent long-term dependencies. In contrast, insufficient decomposition limits the model's capacity to isolate meaningful short-term patterns, reducing its overall expressiveness. Notably, the ETTh2 dataset demonstrates the highest sensitivity to decomposition depth. With a prediction length of 336, the three-level configuration achieves an MAE of 0.444, compared to 0.449 and 0.448 for the two-level and four-level settings, respectively. These results underscore the effectiveness of moderate multi-scale decomposition in capturing both high-frequency and low-frequency components essential for accurate time series forecasting.

Overall, the ablation study validates that the proposed WMDM enhances temporal representation learning, and that an appropriate decomposition depth is critical for achieving optimal forecasting performance across varying datasets and prediction lengths.

**Table 3.** Performance comparison with different wavelet decomposition levels on ETT datasets (MSE/MAE). **Bold** indicates best results.

| Dataset | Horizon | Level=2 | Level=3 | Level=4 |
|---|---|---|---|---|
| | | MSE MAE | MSE MAE | MSE MAE |
| ETTm1 | 336 | 0.375 0.341 | **0.370 0.335** | 0.374 0.339 |
| | 720 | 0.383 0.350 | **0.376 0.345** | 0.380 0.348 |
| ETTh1 | 336 | 0.432 0.458 | **0.426 0.452** | 0.430 0.456 |
| | 720 | 0.455 0.471 | **0.449 0.466** | 0.453 0.469 |
| ETTh2 | 336 | 0.395 0.449 | **0.390 0.444** | 0.394 0.448 |
| | 720 | 0.408 0.458 | **0.403 0.453** | 0.407 0.457 |

**Multi-resolution Feature Extraction Module.** In this study, we introduce **MWDN**, a long-term time series forecasting framework based on a multi-scale wavelet decomposition model. Our model employs the Wavelet Multi-scale Decomposition Module (WMDM) to perform multi-scale decomposition of time series, effectively capturing both time and frequency domain information at different resolutions. Through the Multi-resolution Feature Extraction Module (MFEM), we propose a novel seasonal-

trend decomposition approach and a dual-branch parallel architecture that separately models components at different frequencies. Additionally, we incorporate a frequency-domain enhanced attention mechanism within the MFEM to strengthen its ability to capture spatiotemporal correlations at various frequencies. The Multi-scale Fusion Module (MFM) is used to aggregate spatiotemporal features from multiple resolutions, further improving the model's performance on multivariate time series forecasting tasks. As shown in **Table 4**, experimental results demonstrate that MWDN achieves state-of-the-art (SOTA) performance across a variety of long-term forecasting tasks. Ablation studies validate the effectiveness of the WMDM and MFEM modules in enhancing forecasting accuracy.

**Table 4.** Ablation study on seasonal-trend decomposition methods for multi-resolution feature extraction module (MSE/MAE). **Bold** indicates best results.

| Dataset | Horizon | No Decomp | Moving Avg | **Ours** |
|---|---|---|---|---|
| | | MSE MAE | MSE MAE | MSE MAE |
| ETTm1 | 336 | 0.402 0.361 | 0.381 0.347 | **0.370 0.335** |
| | 720 | 0.415 0.373 | 0.388 0.354 | **0.376 0.345** |
| ETTh1 | 336 | 0.452 0.468 | 0.437 0.457 | **0.426 0.452** |
| | 720 | 0.471 0.482 | 0.456 0.473 | **0.449 0.466** |
| ETTh2 | 336 | 0.417 0.462 | 0.403 0.451 | **0.390 0.444** |
| | 720 | 0.428 0.473 | 0.415 0.464 | **0.403 0.453** |

## 5 Conclusion

In this study, we introduce the **MWDN**, a long time series forecasting framework based on multi-scale wavelet decomposition model. Our model uses the WMDM(wavelet multi-scale decomposition module) to achieve multi-scale decomposition of time series, thereby capturing the time and frequency domain information at different resolutions. Through the MFEM (multi-resolution feature extraction module), we adopt a new seasonal trend decomposition method and a dual-branch parallel architecture to model the components of different frequencies respectively, and we use the frequency domain enhanced attention mechanism to enhance the ability of our MFEM module to capture the spatiotemporal correlation at different frequencies. We use the MFM (multi-scale fusion module) to aggregate spatiotemporal features at different resolutions to improve the performance of the model for multivariate time series forecasting. Our experimental results show that MWDN can effectively achieve SOTA performance in various long-term forecasting tasks, and we perform ablation experiments to show the effective improvement of WMDM and MFEM for forecasting tasks.

**Acknowledgments.** This study was funded by the Natural Science Foundation of Tianjin(No. 24JCYBJC00990), and the Fundamental Research Funds for the Central Universities (3122020051).

## References


1. Qin, H., Zhan, X., Li, Y., Yang, X., Zheng, Y.: Network-wide traffic states imputation using self-interested coalitional learning. Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 1370–1378 (2021)
2. Zhao, Y., Norouzi, H., Azarderakhsh, M., AghaKouchak, A.: Global patterns of hottest, coldest, and extreme diurnal variability on earth. Bulletin of the American Meteorological Society, 102(9), E1672–E1681 (2021)
3. Liu, T., Ma, X., Li, S., Li, X., Zhang, C.: A stock price prediction method based on meta-learning and variational mode decomposition. Knowledge-Based Systems, 252, 109324 (2022)
4. Ma, X., Li, X., Fang, L., Zhao, T., Zhang, C.: U-Mixer: An UNet-Mixer architecture with stationarity correction for time series forecasting. Proceedings of the AAAI Conference on Artificial Intelligence, 38, 14255–14262 (2024)
5. Hyndman, R., Koehler, A.B., Ord, J.K., Snyder, R.D.: Forecasting with Exponential Smoothing: The State Space Approach. Springer (2008)
6. Júnior, D.S.d.O.S., de Oliveira, J.F., de Mattos Neto, P.S.: An intelligent hybridization of ARIMA with machine learning models for time series forecasting. Knowledge-Based Systems, 175, 72–86 (2019)
7. Wu, H., Hu, T., Liu, Y., Zhou, H., Wang, J., Long, M.: TimesNet: Temporal 2D-variation modeling for general time series analysis. ICLR (2023)
8. Wu, H., Xu, J., Wang, J., Long, M.: Autoformer: Decomposition transformers with Auto-Correlation for long-term series forecasting. NeurIPS (2021)
9. Anderson, O., Kendall, M.: Time-Series. 2nd edn. J. R. Stat. Soc. (Series D) (1976)
10. Hyndman, R.J., Athanasopoulos, G.: Forecasting: Principles and Practice. OTexts (2018)
11. Taylor, S.J., Letham, B.: Forecasting at scale. American Statistician (2018)
12. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation (1997)
13. Lai, G., Chang, W.C., Yang, Y., Liu, H.: Modeling long- and short-term temporal patterns with deep neural networks. SIGIR (2018)
14. Franceschi, J.Y., Dieuleveut, A., Jaggi, M.: Unsupervised scalable representation learning for multivariate time series. NeurIPS (2019)
15. Oreshkin, B.N., Carpov, D., Chapados, N., Bengio, Y.: N-BEATS: Neural basis expansion analysis for interpretable time series forecasting. ICLR (2019)
16. Zeng, A., Chen, M., Zhang, L., Xu, Q.: Are transformers effective for time series forecasting? AAAI (2023)
17. Shih, S.Y., Sun, F.K., Lee, H.Y.: Temporal pattern attention for multivariate time series forecasting. (2019), https://arxiv.org/abs/1809.04206
18. Chen, Y., Kang, Y., Chen, Y., Wang, Z.: Probabilistic forecasting with temporal convolutional neural network. (2020), https://arxiv.org/abs/1906.04397
19. Zhou, T., Ma, Z., Wen, Q., Wang, X., Sun, L., Jin, R.: FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. ICML (2022)
20. Nie, Y., Nguyen, N.H., Sinthong, P., Kalagnanam, J.: A time series is worth 64 words: Long-term forecasting with transformers. ICLR (2023)
21. Zhang, Y., Yan, J.: Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. ICLR (2023)
22. Wang, S., Wu, H., Shi, X., Hu, T., Luo, H., Ma, L., Zhang, J.Y., Zhou, J.: Timemixer: Decomposable multiscale mixing for time series forecasting. arXiv preprint arXiv:2405.14616 (2024)

23. Wang, H., Peng, J., Huang, F., Wang, J., Chen, J., Xiao, Y.: MICN: Multi-scale local and global context modeling for long-term series forecasting. The Eleventh International Conference on Learning Representations (2023)
24. Li, J., Hui, X., Zhang, W.: Informer: Beyond efficient transformer for long sequence time-series forecasting. arXiv preprint arXiv:2012.07436 (2021)
25. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. (2015), https://arxiv.org/abs/1502.03167