TOWARDS DIVERSE AND COHERENT AUGMENTATION
FOR TIME-SERIES FORECASTING

Xiyuan Zhang⋆   *Ranak Roy Chowdhury*⋆   *Jingbo Shang*⋆   *Rajesh Gupta* ⋆   *Dezhi Hong* †

⋆ University of California, San Diego
† Amazon

ABSTRACT

Time-series data augmentation mitigates the issue of insufficient training data for deep learning models. Yet, existing augmentation methods are mainly designed for classification, where class labels can be preserved even if augmentation alters the temporal dynamics. We note that augmentation designed for forecasting requires *diversity* as well as *coherence* with the original temporal dynamics. As time-series data generated by real-life physical processes exhibit characteristics in both the time and frequency domains, we propose to combine Spectral and Time Augmentation (STAug) for generating more diverse and coherent samples. Specifically, in the frequency domain, we use the Empirical Mode Decomposition to decompose a time series and reassemble the subcomponents with random weights. This way, we generate diverse samples while being coherent with the original temporal relationships as they contain the same set of base components. In the time domain, we adapt a mix-up strategy that generates diverse as well as linearly in-between coherent samples. Experiments on five real-world time-series datasets demonstrate that STAug outperforms the base models without data augmentation as well as state-of-the-art augmentation methods.

Index Terms— Time Series, Data Augmentation, Forecasting, Decomposition, Spectral Analysis

1. INTRODUCTION

Deep learning has been successful in various time-series applications given enormous amount of data to train. However, time-series data collected through real-world sensors is often marked by irregular samples with missing values due to collection difficulties. Such data *scarcity* commonly observed in time-series data can significantly degrade the performance of deep learning methods that would otherwise perform well.

A rich line of research tries to address this problem through data augmentation, that is to generate synthetic data points to augment the original dataset [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]. However, existing augmentation methods are mainly designed for classification, where augmented samples remain effective as long as they preserve the class labels. We note that augmentation designed for forecasting requires both
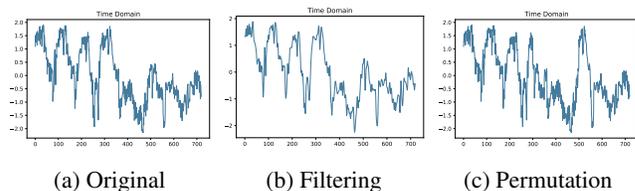

Fig. 1: Visualization of original and augmented time series from ETTm2. Augmentation methods often (b) miss diversity or (c) miss coherence with the original temporal dynamics.

diversity and *coherence* with the original temporal dynamics. Yet, existing augmentation methods generate samples that often miss one of the criteria (Figure 1). For example, filtering-based methods are deterministic processes that produce a fixed set of synthetic samples by removing noises. Permutation-based methods change the temporal order of the original series, worsening forecasting performance.

Moreover, time-series data generated by real-life physical processes exhibit characteristics both in the time and frequency domains that are not available in other data modalities like image and text. Therefore, temporal dynamics can be best captured through a joint consideration of time domain that carries changes over time and frequency domain that conveys periodic patterns. By contrast, existing augmentation methods mostly generate data in one domain, ignoring the complementary strengths of both domains.

We propose *STAug* (Figure 2), by combining Spectral and Time Augmentation for time-series forecasting task. In the frequency domain, we first apply the Empirical Mode Decomposition (EMD) [13] to decompose time series into multiple subcomponents, each representing a certain pattern embedded in the data. We then reassemble these subcomponents with random weights to generate new synthetic series. This offers a *principled* way of augmentation as it generates diverse samples while maintaining the same basic set of subcomponents. We adopt EMD for the frequency information as it better captures patterns for non-stationary time series compared with Fourier transform. In the time domain, we adapt a mix-up strategy [14] to learn linearly in-between randomly sampled pairs of training series, which produces varied and coherent
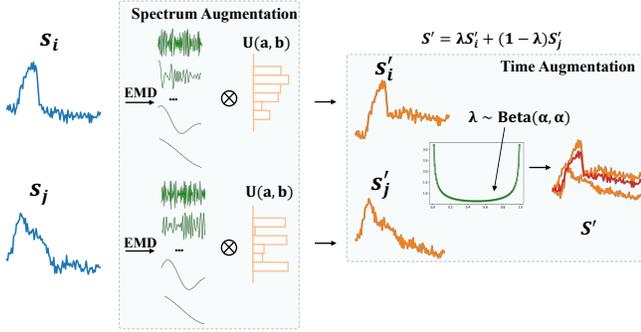
Fig. 2: Overview of STAug. For frequency-domain augmentation, we decompose two random series s_i, s_j (marked in blue) with Empirical Mode Decomposition (EMD), and then reassemble the subcomponents with uniform random weights to obtain s'_i, s'_j (marked in orange). In the time domain, we further linearly mix s'_i, s'_j to obtain the augmented series s' (marked in red) for training.

samples. We evaluate STAug on five real-world time-series datasets, and the method demonstrates state-of-the-art performance compared with existing augmentation methods.

2. RELATED WORK

Existing time-series data augmentation techniques mainly fall into four categories [15, 16] as follows.

Basic random operations include time-domain transformation, frequency-domain transformation and time-frequency domain transformation. Time-domain transformation contains scaling, rotation, jittering [1], slicing, warping [2, 17], etc. For frequency-domain transformation, RobustTAD [18] in the frequency domain makes perturbations in both magnitude and phase spectra. For time-frequency domain transformation, time-frequency features are generated from Short Fourier Transform (STFT), and then local averaging together with feature vector shuffling are applied for augmentation [7]. SpecAugment [19] proposes augmentation in Mel-Frequency by combining warping, masking frequency channels and masking time step blocks together.

Decomposition-based augmentation leverages the Seasonal-Trend Decomposition (STL) [20] or Empirical Mode Decomposition (EMD) [13] to extract patterns for generating synthetic samples. Bagging Exponential Smoothing method [3, 21, 22] uses Box–Cox transformation followed by STL decomposition, and bootstraps the reminder to assemble new series. STL decomposition components can also be adjusted and combined with a stochastic component generated by statistical models [21, 22]. Nam et al. [10] decomposes series using EMD into components from high frequency to low frequency, and adds the residue each time one IMF occurs. This can be viewed as a special case of STAug with weights equal to one for low-frequency components and zero

for high-frequency components, essentially only filtering out high-frequency noise. Moreover, the method does not benefit from time-domain augmentation.

For *pattern mixing method*, DBA calculates weighted average of multiple time series under DTW as new samples [4, 22]. Mix-up [14] constructs new examples through linear interpolation of both features and labels, but such interpolation methods mainly focus on classification tasks.

Generative method models underlying distribution of the dataset for generation, including both statistical generative models [23, 12] and deep generative adversarial network (GAN) [24] based models [5, 6, 25, 8, 9, 11].

In this work, we combine decomposition-based augmentation method in the frequency domain and pattern mixing augmentation method in the time domain to find diverse samples that preserve the original data characteristics.

3. METHODOLOGY

3.1. Overview

We focus on multivariate time series forecasting task. A multivariate time series sequence of length T and feature number c is denoted as $\mathbf{s} = [\mathbf{s_1}, \ldots, \mathbf{s_t}, \ldots, \mathbf{s_T}]^T \in \mathbb{R}^{T \times c}$, where $\mathbf{s_t} = (s_1, \cdots, s_c)^T \in \mathbb{R}^c$. In a forecasting task, we only observe history values up to timestamp $d < T$: $\mathbf{h} = [\mathbf{s_1}, \ldots, \mathbf{s_d}]^T \in \mathbb{R}^{d \times c}$, and the goal is to forecast future values at timestamp $d + 1, \ldots, T$: $\mathbf{f} = [\mathbf{s_{d+1}}, \ldots, \mathbf{s_T}]^T \in \mathbb{R}^{(T-d) \times c}$, where $\mathbf{s} = [\mathbf{h}, \mathbf{f}]$. During training, we have full access to both $\mathbf{h}$ and $\mathbf{f}$, and the training objective is to learn a model g that forecasts $\mathbf{f}$ given $\mathbf{h}$ for each $\mathbf{s}$ in the training set. During testing, we have access to *only* $\mathbf{h}$ and input $\mathbf{h}$ to model g to obtain predictions for the future part.

Our method STAug comprises augmentation in both the frequency domain and time domain. In the frequency domain, we first apply empirical mode decomposition to obtain a set of components. Then during each iteration, these components are re-combined with random weights to construct a new synthetic series. Then, we adapt mix-up as a time-domain augmentation. We linearly interpolate two randomly re-combined series to obtain the final augmented series. The augmented series are fed into the forecasting model for updating gradient. The EMD components of different series could be pre-computed, and STAug only requires randomly re-combining components or series during training, which introduces minimal computational overhead.

3.2. Frequency-Domain Augmentation

The Empirical Mode Decomposition (EMD) [13] method was originally designed to analyze nonlinear and non-stationary data, whose constituent components are not necessarily cyclic. EMD preserves temporal information in contrast with Fourier transform, and is data-driven compared with the linear

wavelet transform. It decomposes a signal into a finite number of Intrinsic Mode Functions (IMF). The first several IMFs usually carry components of higher frequency (e.g., noise), while the last several IMFs represent the low-frequency trend information embedded in the sequence. Therefore, EMD provides a principled way to decompose a signal into multiple components, and each of these components represents certain patterns embedded in the original signal.

After EMD, the original sequence could be written as

$$s(t) = \sum_i^n IMF_i(t) + r(t). \tag{1}$$

With a list of n decomposed IMFs $\{IMF_1, \ldots, IMF_n\}$ and residual r, we apply a random vector $\mathbf{w} = [w_1, \ldots, w_n]$ as weights to re-combine these IMFs as $s'(t)$:

$$s'(t) = \sum_{i=1}^n w_i \cdot IMF_i(t), \tag{2}$$

where w_i is uniformly sampled from $[0, 2]$. This way, the augmented samples are diverse by emphasizing different frequency components via random weights, and at the same time coherent with original distributions as they contain the same basic sets of components.

3.3. Time-Domain Augmentation

Complementing the frequency-domain information, time domain also provides useful patterns. Therefore, we propose to mix up sequences in the time domain, inspired by Mix-up augmentation [14]. Mix-up was originally designed for classification, and we adapt it to time-series forecasting by mixing up values at both past timestamps $1, \ldots, d$ and future timestamps $d + 1, \ldots, T$. Assume $\mathbf{s_i} = [\mathbf{h_i}, \mathbf{f_i}]$ and $\mathbf{s_j} = [\mathbf{h_j}, \mathbf{f_j}]$ are two randomly sampled sequences after spectral augmentation, where $\mathbf{h_i} = [\mathbf{s_i^1}, \ldots, \mathbf{s_i^d}]^T, \mathbf{f_i} = [\mathbf{s_i^{d+1}}, \ldots, \mathbf{s_i^T}]^T$ respectively represents past and future data, similarly for $\mathbf{s_j}$. We construct new sequence as $\mathbf{s'} = [\mathbf{h'}, \mathbf{f'}]$, where

$$\mathbf{h'} = \lambda \mathbf{h_i} + (1 - \lambda)\mathbf{h_j}, \tag{3}$$

$$\mathbf{f'} = \lambda \mathbf{f_i} + (1 - \lambda)\mathbf{f_j}, \tag{4}$$

where λ is sampled from a Beta distribution, i.e., $\lambda \sim \text{Beta}(\alpha, \alpha)$, and α is the hyper-parameter that controls how similar the newly constructed sequence is compared with the original sequences $\mathbf{s_i}$ and $\mathbf{s_j}$. Mix-up augments patterns in the time domain meanwhile by its interpolation nature generates only linearly in-between coherent samples.

3.4. Time-Series Forecasting

For each training iteration, we apply both frequency-domain and time-domain augmentation to obtain an augmented series

$\mathbf{s'} = [\mathbf{h'}, \mathbf{f'}]$. During training, we feed the augmented series history $\mathbf{h'}$ as input, and optimize the forecasting model through reconstructing future part of the series $\mathbf{f'}$. In our experiments, we adopt the state-of-the-art forecasting model Informer [26] (AAAI 2021 best paper) as the base forecaster. To minimize the reconstruction loss $\mathcal{L}$, we calculate Mean Square Error (MSE) between the forecasting model output $\mathbf{y}$ and the ground-truth future part $\mathbf{f'}$:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^N (\mathbf{y_i} - \mathbf{f_i'})^2, \tag{5}$$

where N is the number of augmented series.

4. EXPERIMENTAL SETUP AND RESULTS

4.1. Datasets, Baselines and Experimental Setup

We evaluate our augmentation method STAug on five time-series datasets: ETTh1, ETTh2, ETTm1, ETTm2, Exchange [26, 27]. We follow previous studies [26, 27] for 96 context length and 96, 192, 336, 720 forecasting horizon. All the datasets in our experiments are multivariate, and we apply EMD and mix-up separately for each original variable, and concatenate them to obtain the augmented multivariate time series. We follow previous works for $7 : 2 : 1$ train, validation and test set split, and evaluate the performance with Mean Square Error (MSE) and Mean Absolute Error (MAE).

We use Informer [26] as the base forecasting model. α of the Beta distribution equals 0.5, which is chosen from a grid search of $\{0.25, 0.5, 0.75, 1.0\}$. $p = 0.5$. a, b of the Uniform distribution are set to 0 and 2, respectively. We use Adam optimizer with a decaying learning rate starting from $1e^{-4}$. We compare STAug with base model without any augmentation (None), as well as state-of-the-art time series augmentation methods: WW [2], DBA [4, 28, 22], EMD-R [10], GAN [5], STL [3, 22], RobustTAD [18]. We repeat all the experiments for 3 runs and record the average performance.

4.2. Main Results

We evaluate STAug and report on the results in Table 1. We bold the best results, underline the second best, and mark with dash line the best baseline. STAug consistently outperforms baselines on different datasets by jointly leveraging time-domain and frequency-domain information, with an average reduction of 28.2% for MSE and 18.0% for MAE, compared with the best baseline for each dataset. We also present qualitative comparisons in Figure 4. Compared with the predictions by the base model and the best-performing baseline for the corresponding setting, STAug forecasts values that better align with the original time series.

Table 1: MSE and MAE on benchmark datasets with input context length 96 and forecasting horizon {96, 192, 336, 720}. We **bold** the best performing results, underline the second best, and mark with dashline the best baseline.

Methods		None		WW		RobustTAD		STL		EMD-R		GAN		DBA		STAug-noTime		STAug-noFreq		STAug	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	0.947	0.760	0.894	0.730	1.011	0.789	0.910	0.738	0.838	0.686	0.904	0.734	0.943	0.759	0.658	0.573	0.841	0.693	**0.645**	**0.572**
	192	0.977	0.767	0.999	0.775	0.946	0.744	0.942	0.740	0.942	0.729	0.985	0.757	0.977	0.767	**0.754**	**0.616**	0.962	0.755	0.772	0.628
	336	1.112	0.833	1.058	0.808	1.094	0.816	1.086	0.814	1.056	0.804	1.070	0.816	1.111	0.832	**0.860**	**0.681**	1.103	0.829	0.889	0.705
	720	1.182	0.864	1.201	0.884	1.146	0.841	1.187	0.863	1.187	0.866	1.221	0.892	1.184	0.866	0.994	0.752	1.162	0.857	**0.972**	**0.746**
ETTh2	96	3.084	1.383	3.399	1.463	3.028	1.380	3.042	1.374	3.440	1.451	2.906	1.377	3.005	1.370	**1.802**	**1.012**	2.984	1.371	1.869	1.046
	192	5.966	2.023	6.432	2.098	6.203	2.083	5.894	2.023	6.326	2.089	5.467	1.939	5.636	1.939	4.023	1.569	5.229	1.886	**3.467**	**1.483**
	336	4.775	1.829	4.965	1.852	5.287	1.883	4.774	1.833	5.436	1.948	4.906	1.868	4.765	1.829	3.391	**1.482**	4.374	1.736	**3.365**	1.513
	720	3.991	1.694	3.828	1.677	3.896	1.659	3.563	1.579	4.082	1.717	4.060	1.744	4.009	1.698	2.700	1.377	3.571	1.619	**2.621**	**1.340**
ETTm1	96	0.633	0.570	0.629	0.570	0.593	0.546	0.549	0.534	0.640	0.564	0.626	0.576	0.634	0.573	0.428	0.439	0.550	0.525	**0.403**	**0.424**
	192	0.797	0.673	0.792	0.665	0.802	0.663	0.738	0.646	0.682	0.590	0.842	0.689	0.803	0.675	0.538	0.505	0.668	0.591	**0.489**	**0.482**
	336	1.149	0.839	0.997	0.774	0.983	0.759	0.992	0.773	0.869	0.695	0.902	0.725	1.146	0.838	0.648	0.567	0.900	0.715	**0.627**	**0.559**
	720	1.128	0.805	1.077	0.786	1.063	0.789	1.184	0.820	1.008	0.748	1.225	0.862	1.131	0.806	0.828	0.657	0.934	0.720	**0.764**	**0.628**
ETTm2	96	0.415	0.506	0.385	0.476	0.386	0.476	0.388	0.489	0.398	0.478	0.334	0.430	0.415	0.506	0.297	0.394	0.328	0.423	**0.292**	**0.388**
	192	0.776	0.675	0.784	0.680	0.875	0.724	0.712	0.650	0.542	0.551	0.625	0.604	0.767	0.673	**0.420**	**0.490**	0.592	0.585	0.429	0.497
	336	1.515	0.946	1.425	0.923	1.390	0.909	1.402	0.908	1.287	0.875	1.453	0.929	1.568	0.962	0.943	0.735	1.287	0.876	**0.811**	**0.692**
	720	3.462	1.423	4.370	1.637	4.792	1.654	3.271	1.378	3.247	1.353	3.129	1.355	3.475	1.427	**2.645**	**1.201**	3.765	1.509	2.790	1.268
Exchange	96	0.959	0.784	0.799	0.687	0.962	0.788	0.860	0.746	0.923	0.761	0.942	0.772	0.958	0.784	0.285	0.402	0.809	0.708	**0.271**	**0.395**
	192	1.113	0.837	1.084	0.787	1.180	0.873	1.143	0.852	1.182	0.853	1.115	0.855	1.109	0.836	0.627	0.615	1.125	0.822	**0.606**	**0.605**
	336	1.605	1.009	1.697	0.995	1.562	1.006	1.548	0.987	1.606	1.001	1.465	0.972	1.595	1.007	1.007	0.807	1.438	0.944	**0.935**	**0.779**
	720	2.847	1.390	3.198	1.481	2.816	1.380	2.484	1.297	2.914	1.413	2.571	1.309	2.847	1.390	**1.409**	**0.928**	2.056	1.153	1.771	1.052

4.3. Ablation Study

To examine the effect of augmentation in different domains, we also conduct ablation study by augmenting data only in the frequency or time domain in Table 1. STAug-noFreq stands for the model that *removes* frequency-domain augmentation, and STAug-noTime stands for the model that *removes* time-domain data augmentation. The performance degrades after removing augmentation in either the frequency or time domain, validating the need of combining both domains.

4.4. Robustness Study

We sub-sample 10%, 20%, 50% of the Exchange dataset, and compare STAug with the base model and the best-performing baseline methods for the corresponding setting (horizon 96). We report on MSE and MAE in Figure 3. The performances of different methods increase as we have more data. When the sample size is small, the performance gap between augmentation methods and base model without augmentation becomes larger, which shows that augmentation is especially helpful when the original dataset is small. STAug leverages information from both time and frequency domains, and performs consistently better with respect to the original sample size. Moreover, STAug shows more significant improvement over base model and the best-performing baselines with fewer available samples in the original dataset, which demonstrates its robustness with respect to the number of data samples.

5. CONCLUSION

We proposed a generic yet effective time-series data augmentation method STAug to combine patterns in both the time domain and frequency domain. In the frequency domain, the re-combined subcompacts of the original time series are both diverse and preserve the original basic components. In the
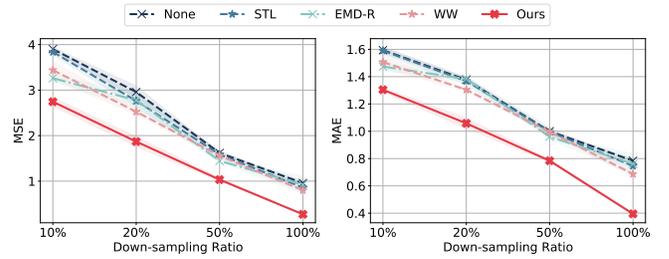

Fig. 3: MSE and MAE for different down-sampling ratios.
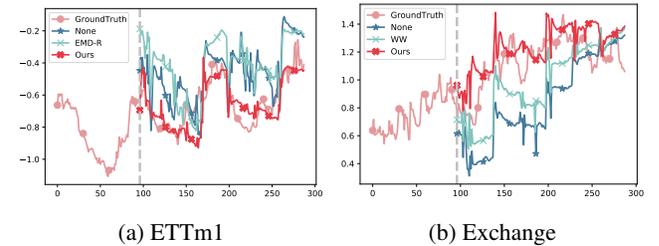

(a) ETTm1   (b) Exchange

Fig. 4: Predictions on ETTm1 and Exchange with 192 horizon. STAug predicts data that best align with the ground truth.

time domain, we adapt Mix-up to generate diverse and in-between coherent samples by linearly interpolating both past and future part of a series. Experiments on five real-world time series datasets show that STAug best reduces the forecasting errors of base model compared with existing augmentation methods. We also performed robustness analysis and observed that STAug stays robust with respect to sampling size. Our ongoing work in the area involves systematically studying the effectiveness of various time domain and frequency domain augmentation methods, and designing data-dependent selection procedure to choose the most suitable augmentation method for different datasets.

6. REFERENCES

[1] Terry T Um, Franz MJ Pfister, Daniel Pichler, Satoshi Endo, Muriel Lang, Sandra Hirche, Urban Fietzek, and Dana Kulić, "Data augmentation of wearable sensor data for parkinson's disease monitoring using convolutional neural networks," in *ICMI*, 2017.

[2] Arthur Le Guennec, Simon Malinowski, and Romain Tavenard, "Data augmentation for time series classification using convolutional neural networks," in *ECML/PKDD workshop*, 2016.

[3] Christoph Bergmeir, Rob J Hyndman, and José M Benítez, "Bagging exponential smoothing methods using stl decomposition and box–cox transformation," *International journal of forecasting*, 2016.

[4] Germain Forestier, François Petitjean, Hoang Anh Dau, Geoffrey I Webb, and Eamonn Keogh, "Generating synthetic time series to augment sparse datasets," in *ICDM*, 2017.

[5] Cristóbal Esteban, Stephanie L Hyland, and Gunnar Rätsch, "Real-valued (medical) time series generation with recurrent conditional gans," *arXiv preprint arXiv:1706.02633*, 2017.

[6] Zhengping Che, Yu Cheng, Shuangfei Zhai, Zhaonan Sun, and Yan Liu, "Boosting deep learning risk prediction with generative adversarial networks for electronic health records," in *ICDM*, 2017.

[7] Odongo Steven Eyobu and Dong Seog Han, "Feature representation and data augmentation for human activity classification based on wearable imu sensor data using a deep lstm neural network," *Sensors*, 2018.

[8] Swee Kiat Lim, Yi Loo, Ngoc-Trung Tran, Ngai-Man Cheung, Gemma Roig, and Yuval Elovici, "Doping: Generative data augmentation for unsupervised anomaly detection with gan," in *ICDM*, 2018.

[9] Jinsung Yoon, Daniel Jarrett, and Mihaela Van der Schaar, "Time-series generative adversarial networks," 2019.

[10] Gue-Hwan Nam, Seok-Jun Bu, Na-Mu Park, Jae-Yong Seo, Hyeon-Cheol Jo, and Won-Tae Jeong, "Data augmentation using empirical mode decomposition on neural networks to classify impact noise in vehicle," in *ICASSP*, 2020.

[11] Hailin Hu, MingJian Tang, and Chengcheng Bai, "Datsing: Data augmented time series forecasting with adversarial domain adaptation," in *CIKM*, 2020.

[12] Yanfei Kang, Rob J Hyndman, and Feng Li, "Gratis: Generating time series with diverse and controllable characteristics," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 2020.

[13] Norden E Huang, Zheng Shen, Steven R Long, Manli C Wu, Hsing H Shih, Quanan Zheng, Nai-Chyuan Yen, Chi Chao Tung, and Henry H Liu, "The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis," *Proceedings of the Royal Society of London. Series A: mathematical, physical and engineering sciences*, 1998.

[14] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz, "mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.

[15] Qingsong Wen, Liang Sun, Fan Yang, Xiaomin Song, Jingkun Gao, Xue Wang, and Huan Xu, "Time series data augmentation for deep learning: A survey," *arXiv preprint arXiv:2002.12478*, 2020.

[16] Brian Kenji Iwana and Seiichi Uchida, "An empirical survey of data augmentation for time series classification with neural networks," *Plos one*, 2021.

[17] Qianli Ma, Zhenjing Zheng, Jiawei Zheng, Sen Li, Wanqing Zhuang, and Garrison W Cottrell, "Joint-label learning by dual augmentation for time series classification," in *AAAI*, 2021.

[18] Jingkun Gao, Xiaomin Song, Qingsong Wen, Pichao Wang, Liang Sun, and Huan Xu, "Robusttad: Robust time series anomaly detection via decomposition and convolutional neural networks," *arXiv preprint arXiv:2002.09545*, 2020.

[19] Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le, "Specaugment: A simple data augmentation method for automatic speech recognition," *arXiv preprint arXiv:1904.08779*, 2019.

[20] Robert B Cleveland, William S Cleveland, Jean E McRae, and Irma Terpenning, "Stl: A seasonal-trend decomposition," *J. Off. Stat*, 1990.

[21] Lars Kegel, Martin Hahmann, and Wolfgang Lehner, "Feature-based comparison and generation of time series," in *SSDBM*, 2018.

[22] Kasun Bandara, Hansika Hewamalage, Yuan-Hao Liu, Yanfei Kang, and Christoph Bergmeir, "Improving the accuracy of global forecasting models using time series data augmentation," *Pattern Recognition*, 2021.

[23] Hong Cao, Vincent YF Tan, and John ZF Pang, "A parsimonious mixture of gaussian trees model for oversampling in imbalanced and multimodal time-series classification," *IEEE transactions on neural networks and learning systems*, 2014.

[24] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," *NeurIPS*, 2014.

[25] Giorgia Ramponi, Pavlos Protopapas, Marco Brambilla, and Ryan Janssen, "T-cgan: Conditional generative adversarial network for data augmentation in noisy time series with irregular sampling," *arXiv preprint arXiv:1811.08295*, 2018.

[26] 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 *AAAI*, 2021.

[27] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," *NeurIPS*, 2021.

[28] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, and Pierre-Alain Muller, "Data augmentation using synthetic data for time series classification with deep residual networks," *arXiv preprint arXiv:1808.02455*, 2018.