Appendix: Time Series is a Special Sequence: Forecasting with Sample Convolution and Interaction

In this appendix, we first introduce the datasets and evaluation metrics used in the experiments in Section A. Then, we provide extra experimental results in Section B. In Section C, we present details of network design, training scheme, and hyper-parameter tuning.

A Datasets and Evaluation Metrics

We conduct the experiments on 7 popular time-series datasets, namely *Electricity Transformer Temperature* (ETTh1, ETTh2 and ETTm1) (Zhou et al. 2021), and *PeMS* (PEMS03, PEMS04, PEMS07 and PEMS08) (Chen et al. 2001) in the paper. In this Appendix, we show the extra experimental results on 4 other datasets: *Traffic*, *Solar-Energy*, *Electricity* and *Exchange-Rate* (Lai et al. 2018).

Here, we present the details of each dataset and the metrics used for assessing the performance of time series forecasting (TSF) models on the above datasets.

A.1 Electricity Transformer Temperature (ETT)

ETT[1] consists of 2 year electric power data collected from two separated counties of China, including hourly subsets $\{ETTh1, ETTh2\}$ and quarter-hourly subsets $\{ETTm1\}$. Each data point includes an "oil temperature" value and 6 power load features. The train, validation and test sets contain 12, 4 and 4 months data, respectively.

For data pre-processing, we perform zero-mean normalization, i.e., $X^{'} = (X - mean(X))/std(X)$, where $mean(X)$ and $std(X)$ are the mean and the standard deviation of historical time series, respectively. We use Mean Absolute Errors (MAE) (Hyndman and Koehler 2006) and Mean Squared Errors (MSE) (Makridakis et al. 1982) for model comparison.

$$MAE = \frac{1}{\tau}\sum_{i=0}^{\tau}|\hat{x}_i - x_i| \tag{1}$$

$$MSE = \frac{1}{\tau}\sum_{i=0}^{\tau}(\hat{x}_i - x_i)^2 \tag{2}$$

where $\hat{x}_i$ is the model's prediction, and x_i is the ground-truth. τ is the length of the prediction horizon.

[1]https://github.com/zhouhaoyi/ETDataset

A.2 PeMS

PeMS[2] contains four public traffic network datasets (*PEMS03, PEMS04, PEMS07 and PEMS08*) which are respectively constructed from Caltrans Performance Measurement System (PeMS) of four districts in California. The data is aggregated into 5-minutes windows, resulting in 12 points per hour and 288 points per day. We use traffic flow data from the past hour to predict the flow for the next hour. The data is pre-processed using zero-mean normalization as ETT.

Following (Hyndman and Koehler 2006), we use Root Mean Squared Errors (RMSE) and Mean Absolute Percentage Errors (MAPE) as evaluation metrics on this dataset.

$$RMSE = \sqrt{\frac{1}{\tau}\sum_{i=0}^{\tau}(\hat{x}_i - x_i)^2}, \tag{3}$$

$$MAPE = \sqrt{\frac{1}{\tau}\sum_{i=0}^{\tau}|(\hat{x}_i - x_i)/x_i|}. \tag{4}$$

A.3 Traffic, Solar-Energy, Electricity and Exchange-Rate

Traffic[3] contains the hourly data describing the road occupancy rates (ranging from 0 to 1) that are recorded by the sensors on San Francisco Bay area freeways from 2015 to 2016 (48 months in total). *Solar-Energy*[4] records the solar power production from 137 PV plants in Alabama State, which are sampled every 10 minutes in 2016. *Electricity*[5] includes the hourly electricity consumption (kWh) records of 321 clients from 2012 to 2014. *Exchange-Rate*[6] collects the daily exchange rates of 8 foreign countries from 1990 to 2016.

In our experiments, the length of the look-back window T for the above datasets is 168, and we trained independent models for different length of future horizon (i.e., $\tau = 3, 6, 12, 24$). We use Root Relative Squared Error (RSE) and Empirical Correlation Coefficient (CORR) to evaluate

[2]https://pems.dot.ca.gov

[3]http://pems.dot.ca.gov

[4]http://www.nrel.gov/grid/solar-power-data.html

[5]https://archive.ics.uci.edu/ml/datasets/ElectricityLoadDiagrams20112014

[6]https://github.com/laiguokun/multivariate-time-series-data

Table 1: Performance comparison of different approaches on *Traffic*, *Solar-Energy*, *Electricity* and *Exchange-Rate* datasets.

Methods	Metrics	Solar-Energy				Traffic				Electricity				Exchange-rate			
		Horizon				Horizon				Horizon				Horizon			
		3	6	12	24	3	6	12	24	3	6	12	24	3	6	12	24
AR	RSE	0.2435	0.3790	0.5911	0.8699	0.5991	0.6218	0.6252	0.6300	0.0995	0.1035	0.1050	0.1054	0.0228	0.0279	0.0353	0.0445
	CORR	0.9710	0.9263	0.8107	0.5314	0.7752	0.7568	0.7544	0.7591	0.8845	0.8632	0.8691	0.8595	0.9734	0.9656	0.9526	0.9357
VARMLP	RSE	0.1922	0.2679	0.4244	0.6841	0.5582	0.6579	0.6023	0.6146	0.1393	0.1620	0.1557	0.1274	0.0265	0.0394	0.0407	0.0578
	CORR	0.9829	0.9655	0.9058	0.7149	0.8245	0.7695	0.7929	0.7891	0.8708	0.8389	0.8192	0.8679	0.8609	0.8725	0.8280	0.7675
GP	RSE	0.2259	0.3286	0.5200	0.7973	0.6082	0.6772	0.6406	05995	0.1500	0.1907	0.1621	0.1273	0.0239	0.0272	0.0394	0.0580
	CORR	0.9751	0.9448	0.8518	0.5971	0.7831	0.7406	0.7671	0.7909	0.8670	0.8334	0.8394	0.8818	0.8713	0.8193	0.8484	0.8278
RNN-GRU	RSE	0.1932	0.2628	0.4163	0.4852	0.5358	0.5522	0.5562	05633	0.1102	0.1144	0.1183	0.1295	0.0192	0.0264	0.0408	0.0626
	CORR	0.9823	0.9675	0.9150	0.8823	0.8511	0.8405	0.8345	0.8300	0.8597	0.8623	0.8472	0.8651	0.9786	0.9712	0.9531	0.9223
LSTNet	RSE	0.1843	0.2559	0.3254	0.4643	0.4777	0.4893	0.4950	0.4973	0.0864	0.0931	0.1007	0.1007	0.0226	0.0280	0.0356	0.0449
	CORR	0.9843	0.9690	0.9467	0.8870	0.8721	0.8690	0.8614	0.8588	0.9283	0.9135	0.9077	0.9119	0.9735	0.9658	0.9511	0.9354
TPR-LSTM	RSE	0.1803	0.2347	0.3234	0.4389	0.4487	0.4658	0.4641	0.4765	0.0823	0.0916	0.0964	0.1006	**0.0174**	**0.0241**	**0.0341**	**0.0444**
	CORR	0.9850	0.9742	0.9487	0.9081	0.8812	0.8717	0.8717	0.8629	0.9439	0.9337	0.9250	0.9133	**0.9790**	**0.9709**	**0.9564**	**0.9381**
MTGNN	RSE	*0.1778*	*0.2348*	*0.3109*	*0.4270*	**0.4162**	*0.4754*	**0.4461**	*0.4535*	**0.0745**	*0.0878*	**0.0916**	**0.0953**	0.0194	0.0259	0.0349	0.0456
	CORR	*0.9852*	*0.9726*	*0.9509*	*0.9031*	**0.8963**	*0.8667*	**0.8794**	*0.8810*	*0.9474*	*0.9316*	*0.9278*	*0.9234*	*0.9786*	*0.9708*	*0.9551*	*0.9372*
SCINet	RSE	**0.1775**	**0.2301**	**0.2997**	**0.4081**	*0.4216*	**0.4414**	*0.4495*	**0.4453**	*0.0748*	**0.0845**	*0.0926*	*0.0976*	*0.0180*	*0.0247*	**0.0340**	**0.0442**
	CORR	**0.9853**	**0.9739**	**0.9550**	**0.9112**	*0.8920*	**0.8809**	*0.8772*	**0.8825**	**0.9492**	**0.9386**	**0.9304**	**0.9274**	0.9739	0.9662	0.9487	0.9255
IMP	RSE	0.17%	2.00%	3.60%	4.42%	-1.29%	7.15%	-0.76%	1.80%	-0.40%	3.76%	-1.09%	-2.41%	-3.45%	-2.43%	0.29%	0.45%
	CORR	0.01%	1.37%	0.43%	0.9%	-0.47%	1.64%	-0.25%	0.17%	0.19%	0.75%	0.28%	0.43%	-0.52%	-0.48%	-0.80%	-1.34%

the performance of the TSF models on these datasets following (Lai et al. 2018), which are calculated as follows:

$$RSE = \frac{\sqrt{\sum_{i=0}^{\tau}(\hat{x}_i - x_i)^2}}{\sqrt{\sum_{i=0}^{\tau}(x_i - mean(X))^2}}, \tag{5}$$

$$CORR = \frac{1}{d}\sum_{j=0}^{d}\frac{\sum_{i=0}^{\tau}(x_{i,j} - mean(X_j))(\hat{x}_{i,j} - mean(\hat{X}_j))}{\sum_{i=0}^{\tau}(x_{i,j} - mean(X_j))^2(\hat{x}_{i,j} - mean(\hat{X}_j))^2}, \tag{6}$$

where X and $\hat{X}$ are the ground-truth and model's prediction, respectively. d is the number of variates.

B Extra Experimental Results

In this section, we first conduct experiments on 4 extra forecasting datasets to see the performance of the SCINet in single-step forecasting task. Then, we add empirical study on *PEMS* datasets to show the impact of different operator combinations in *SCI-Block*.

B.1 Performance Comparison on Extra Datasets.

To further evaluate the performance of the proposed *SCINet*, we also conduct the experiments on 4 single-step forecasting datasets (Lai et al. 2018), namely *Traffic*, *Solar-Energy*, *Electricity* and *Exchange-Rate*. The task is defined as follows:

Given a long time series $\mathbf{X}^*$ and a look-back window of fixed length T, at timestamp t, the single-step forecasting is to predict the future value $\hat{\mathbf{X}}_{t+\tau:t+\tau} = \{\mathbf{x}_{t+\tau}\}$. Here, τ is the length of the forecast horizon, $x_t \in \mathbb{R}^d$ is the value at time step t, and d is the number of variates.

Loss Function To enhance the performance in single-step forecasting, we revise the loss function of the last SCINet in the stacked SCINet with $K(K \geq 1)$. The loss function contains two parts:

$$\mathcal{L}_k = \frac{1}{\tau}\sum_{i=0}^{\tau}\left\|\hat{\mathbf{x}}_i^k - \mathbf{x}_i\right\|, \quad k \neq K. \tag{7}$$

Table 2: The impact of different operators

Operators	PEMS03	PEMS04	PEMS07	PEMS08
	MAE			
+, +	15.08	19.27	21.69	**15.72**
-, -	**15.06**	**19.21**	**21.63**	15.78
+, -	15.09	19.31	21.77	15.84
-, +	15.30	19.32	21.72	15.79

For the last stack K, we introduce a balancing parameter $\lambda \in (0, 1)$ for the value of the last time-step[7]:

$$\mathcal{L}_K = \frac{1}{\tau - 1}\sum_{i=0}^{\tau-1}\left\|\hat{\mathbf{x}}_i^K - \mathbf{x}_i\right\| + \lambda\left\|\hat{\mathbf{x}}_\tau^K - \mathbf{x}_\tau\right\|. \tag{8}$$

Therefore, the total loss of the stacked SCINet can be written as:

$$\mathcal{L} = \sum_{k=1}^{K-1}\mathcal{L}_k + \mathcal{L}_K. \tag{9}$$

Results and Analyses As can be seen in Table 1, SCINet outperforms existing TSF solutions in most cases, especially for the Solar-Energy datasets. At the same time, SCINet is slightly inferior to state-of-the-art spatial-temporal models MTGNN (Wu et al. 2020) and TPA-LSTM (Shih, Sun, and yi Lee 2019) in some cases. We attribute it to the fact that these datasets have relatively strong spatial relations and hence the dedicated spatial modeling architectures in these techniques contribute to their high performance.

B.2 Empirical Study on Operator Selection

In Eq. (2) of the paper, the operators can be either "addition" or "subtraction". Although the model can learn the operation adaptively during training, the parameter initialization would affect the final performance. As shown in Table 2, the impact of operator settings is minor.

[7]This is slightly different from other practice for single-step forecasting (Lai et al. 2018), because we choose to use all the available values in the prediction window as supervision signal.

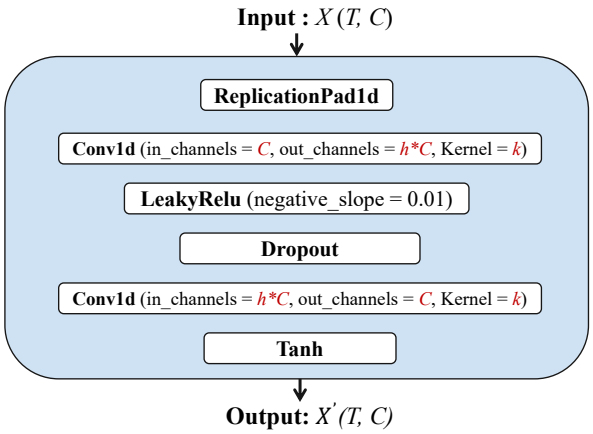

Figure 1: The structure of ϕ, ρ, ψ, and η.

C Reproducibility

Our code is implemented with PyTorch. All the experiments are conducted on an Nvidia Tesla V100 SXM2 GPU (32GB memory), which is sufficient for all our experiments.

Structure of the network modules ϕ, ρ, ψ, and η in SCI-Block: As shown in Fig. 1, ϕ, ρ, ψ, and η use the same network architecture. First, the replication padding is used to keep the border shrunk caused by the convolution operation. Then, a 1d convolutional layer with kernel size k is applied to extend the input channel C to h*C and followed with LeakyRelu and Dropout. h means a scale of the hidden size. Next, the second 1d convolutional layer with kernel size k is to recover the channel h*C to the input channel C. The stride of all the convolutions is 1. We use a LeakyRelu activation after the first convolutional layer because of its sparsity properties and a reduced likelihood of vanishing gradient. We apply a Tanh activation after the second convolutional layer since it can keep both positive and negative features into [-1, 1].

Training details: For all datasets, we fix the random seed to be 4321, and train the model for 150 epochs at most. The reported results on the test set are based on the model that achieves the best performance on the validation set.

Hyper-parameter tuning: We conduct a grid search over all the essential hyper-parameters on the held-out validation set of the datasets. The detailed hyper-parameter configurations of *ETT* are shown in Table 4. Besides, the parameters of the four datasets in *PeMS* are presented in Table6. The *Traffic, Solar-Energy, Electricity and Exchange-rate* are shown in Table 5. Notably, we only apply the weighted loss to the *Solar* and *Exchange-rate* data since they show less autocorrelation (Lai et al. 2018), which indicates the temporal correlation of the distant time-stamp cannot be well modelled by a general L1 loss. Moreover, to build a non-causal TCN[8] in the paper, we only need to remove the *chomps* in the code and make the padding equal to the dilation.

[8]https://github.com/locuslab/TCN/issues/45

References

Chen, C.; Petty, K.; Skabardonis, A.; Varaiya, P.; and Jia, Z. 2001. Freeway Performance Measurement System: Mining Loop Detector Data. *Transportation Research Record*, 1748: 102 – 96.

Hyndman, R.; and Koehler, A. 2006. Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22: 679–688.

Lai, G.; Chang, W.-C.; Yang, Y.; and Liu, H. 2018. Modeling long- and short-term temporal patterns with deep neural networks. In *SIGIR*.

Makridakis, S.; Andersen, A.; Carbone, R.; Fildes, R.; Hibon, M.; Lewandowski, R.; Newton, J.; Parzen, E.; and Winkler, R. 1982. The accuracy of extrapolation (time series) methods: Results of a forecasting competition. *Journal of Forecasting*, 1: 111–153.

Shih, S.-Y.; Sun, F.-K.; and yi Lee, H. 2019. Temporal pattern attention for multivariate time series forecasting. *Machine Learning*, 1–21.

Wu, Z.; Pan, S.; Long, G.; Jiang, J.; Chang, X.; and Zhang, C. 2020. Connecting the Dots: Multivariate Time Series Forecasting with Graph Neural Networks. In *KDD*.

Zhou, H.-Y.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; and Zhang, W. 2021. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. In *AAAI*.

Table 3: The hyperparameters in ETT datasets (Multivariate)

Model configurations		ETTh1					ETTh2					ETTm1				
Hyperparameter	Horizon	24	48	168	336	720	24	48	168	336	720	24	48	96	288	672
	Look-back window	48	96	336	336	736	48	96	336	336	736	48	96	384	672	672
	Batch size	8	16	32	512	256	16	4	16	128	128	32	16	32	32	32
	Learning rate	3e-3	9e-3	5e-4	1e-4	5e-5	7e-3	7e-3	5e-5	5e-5	1e-5	5e-3	1e-3	5e-5	1e-5	1e-5
SCI Block	h	4	4	4	1	1	8	4	0.5	1	4	4	4	0.5	4	4
	k	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
	Dropout	0.5	0.25	0.5	0.5	0.5	0.25	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
SCINet	L (level)	3	3	3	4	5	3	4	4	4	5	3	4	4	5	5
Stacked SCINet	K (stack)	1	1	1	1	1	1	1	1	1	1	1	2	2	1	2

Table 4: The hyperparameters in ETT datasets (Univariate)

Model configurations		ETTh1					ETTh2					ETTm1				
Hyperparameter	Horizon	24	48	168	336	720	24	48	168	336	720	24	48	96	288	672
	Look-back window	64	720	720	720	736	48	96	336	336	720	96	96	384	384	672
	Batch size	64	8	8	128	32	16	32	8	512	128	8	16	8	64	32
	Learning rate	7e-3	1e-4	5e-5	1e-3	1e-4	1e-3	1e-3	1e-4	5e-4	1e-5	1e-3	5e-4	1e-5	1e-5	1e-4
SCI Block	h	8	4	4	1	4	4	4	4	8	8	4	4	2	4	1
	k	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
	Dropout	0.25	0.5	0.5	0.5	0.5	0	0.5	0	0.5	0.6	0	0	0	0	0.5
SCINet	L (level)	3	4	3	4	5	3	4	3	3	3	4	3	4	4	5
Stacked SCINet	K (stack)	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1

Table 5: The hyperparameters in Traffic, Solar-energy, Electricity and Exchange-rate datasets

Model configurations		Solar				Electricity				Traffic				Exc-Rate			
Hyperparameter	Horizon	3	6	12	24	3	6	12	24	3	6	12	24	3	6	12	24
	Look-back window	160				168											
	Batch size	256	256	1024	256	32				16				4			
	Learning rate	1e-4				9e-3				5e-4				5e-3			7e-3
SCI Block	h	1	0.5	2	1	8				1	2	0.5	2	0.125			
	k	5				5				5				5			
	Dropout	0.25				0				0.5	0.25	0.25	0.5	0.5			
SCINet	L (level)	4				3				3			2	3			
Stacked SCINet	K (stack)	2			1	2				2	1	2	2	1			
	Loss weight (λ)	0.5				$\times$				$\times$				0.5			

Table 6: The hyperparameters in PeMS datasets

Model configurations		PEMS03	PEMS04	PEMS07	PEMS08
Hyperparameter	Horizon	12			
	Look-back window	12			
	Batch size	8			
	Learning rate	1e-3			
SCI Block	h	0.0625	0.0625	0.03125	1
	k	5			
	Dropout	0.25	0	0.25	0.5
SCINet	L (level)	2			
Stacked SCINet	K (stack)	1			