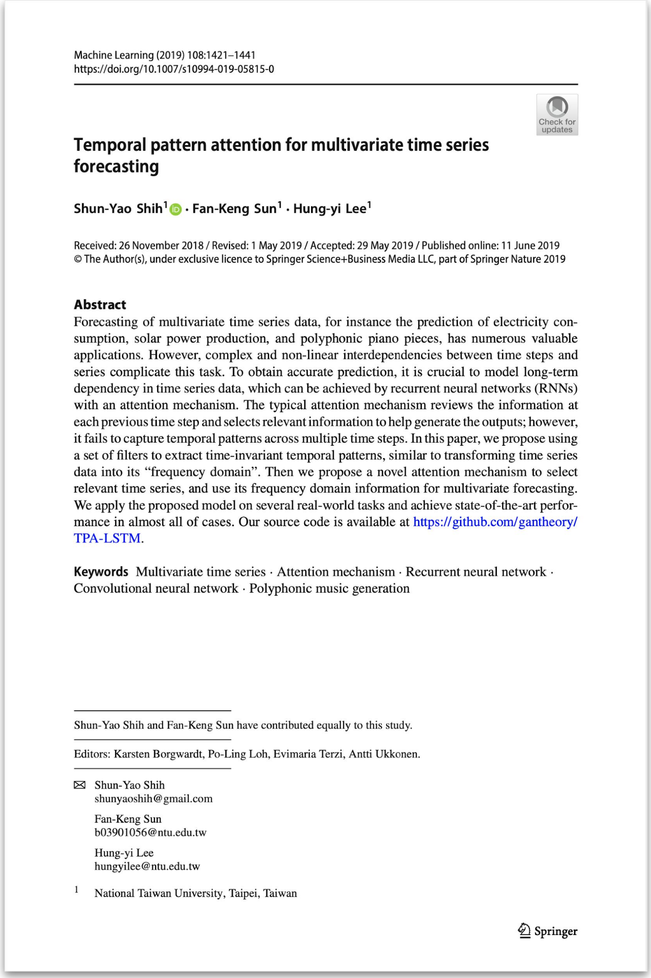
# Temporal pattern attention for multivariate time series forecasting

National Taiwan University,
Published online 11 June 2019

Presented by Henrik Grønbech


Machine Learning (2019) 108:1421–1441
https://doi.org/10.1007/s10994-019-05815-0

Check for updates

## Temporal pattern attention for multivariate time series forecasting

Shun-Yao Shih[1] · Fan-Keng Sun[1] · Hung-yi Lee[1]

Received: 26 November 2018 / Revised: 1 May 2019 / Accepted: 29 May 2019 / Published online: 11 June 2019
© The Author(s), under exclusive licence to Springer Science+Business Media LLC, part of Springer Nature 2019

**Abstract**
Forecasting of multivariate time series data, for instance the prediction of electricity consumption, solar power production, and polyphonic piano pieces, has numerous valuable applications. However, complex and non-linear interdependencies between time steps and series complicate this task. To obtain accurate prediction, it is crucial to model long-term dependency in time series data, which can be achieved by recurrent neural networks (RNNs) with an attention mechanism. The typical attention mechanism reviews the information at each previous time step and selects relevant information to help generate the outputs; however, it fails to capture temporal patterns across multiple time steps. In this paper, we propose using a set of filters to extract time-invariant temporal patterns, similar to transforming time series data into its "frequency domain". Then we propose a novel attention mechanism to select relevant time series, and use its frequency domain information for multivariate forecasting. We apply the proposed model on several real-world tasks and achieve state-of-the-art performance in almost all of cases. Our source code is available at https://github.com/gantheory/TPA-LSTM.

**Keywords** Multivariate time series · Attention mechanism · Recurrent neural network · Convolutional neural network · Polyphonic music generation

Shun-Yao Shih and Fan-Keng Sun have contributed equally to this study.

Editors: Karsten Borgwardt, Po-Ling Loh, Evimaria Terzi, Antti Ukkonen.

✉ Shun-Yao Shih
shunyaoshih@gmail.com

Fan-Keng Sun
b03901056@ntu.edu.tw

Hung-yi Lee
hungyilee@ntu.edu.tw

1 National Taiwan University, Taipei, Taiwan

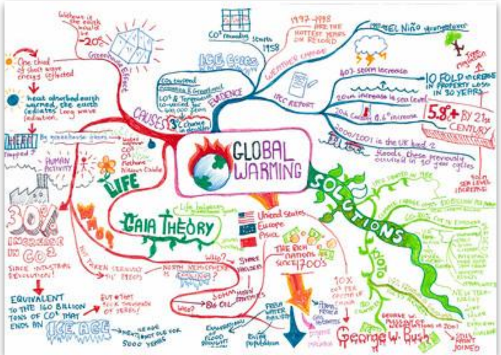
Springer

# Overview

- Motivation
- Description of the addressed problem
- Related work
- Methods
- Experimental Setting and Results
- Conclusions

# Motivation

- This paper suggests a new general technique that can be used in multivariate time series (MTS) forecasting
- Multivariate times series are everywhere
- Increase forecasting performance would be valuable in almost every domain and for humanity



"Global Warming Brainstorm" by Richard Scott 33 is licensed under CC BY-NC-SA 2.0

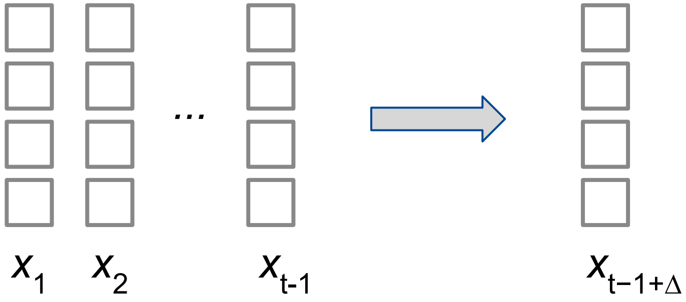

"Graph With Stacks Of Coins" by kenteegardin is licensed under CC BY-SA 2.0


"Hydro-Power-Plant_61348-480x360" by Public Domain Photos is licensed under CC BY 2.0

# Description of the addressed problem

Given X = $\{x_1, x_2, . . . , x_{t-1}\}$, where $x_i \in R^n$, the task is to predict the value of $x_{t-1+\Delta}$, where $\Delta$ is a fixed horizon with respect to different tasks

As common practice, they only use $\{x_{t-w}, x_{t-w+1}, . . . , x_{t-1}\}$ to predict $x_{t-1+\Delta}$, where w is the window size



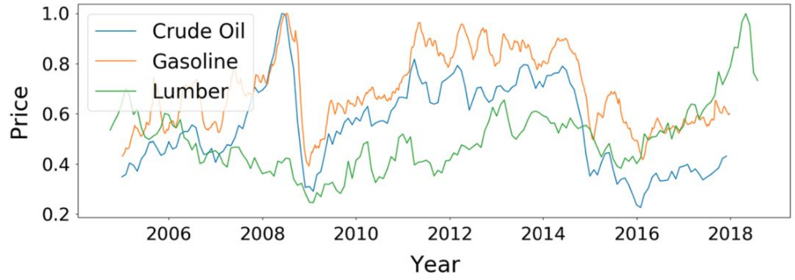




**Fig. 1** Historical prices of crude oil, gasoline, and lumber. Units are omitted and scales are normalized for simplicity

# Related work

Univariate time series

- ARIMA
- Linear support vector regression (SVR)

Multivariate time series traditional

- Vector autoregression (VAR)
- Kernel methods, ensembles, Gaussian processes, regime switching

Deep learning

- Univariate
  - RNN, LSTM
- Multivariate
  - LSTNet (2017)

# Attention mechanism in LSTNet

LSTNet paper introduces both a recurrent-skip layer and an attention mechanism to rows (time stamps)

Rationale: If you want to predict traffic at a sunday morning, look at traffic at previous sunday mornings

Shortcomings:

1. Skip-length must be manually tuned
2. Skip model only works on periodic data
3. Attention layer looks at relevant hidden states—not relevant time series

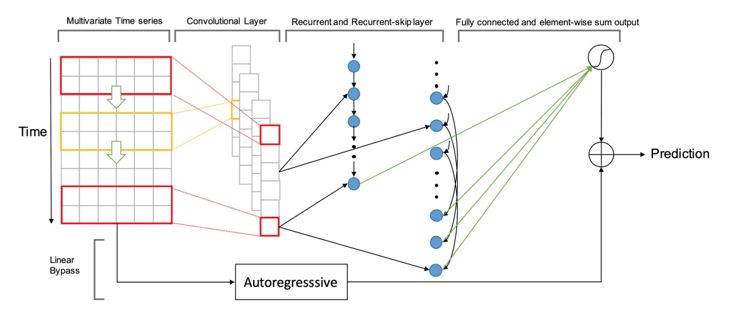
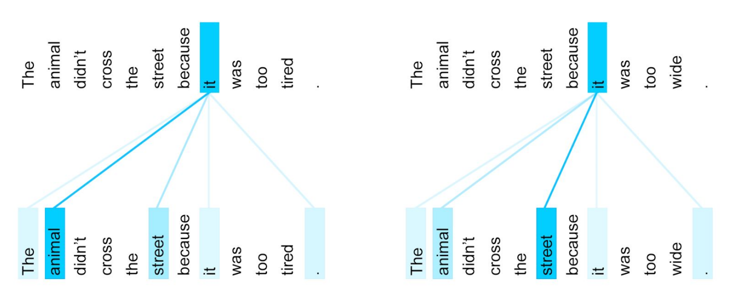


Figure 2: An overview of the Long- and Short-term Time-series network (LSTNet)

from the paper Modeling Long- and Short-Term Temporal Patterns with Deep Neural Networks




from https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html

# Attention mechanism in LSTNet

LSTNet paper introduces both a recurrent-skip layer and an attention mechanism to rows (time stamps)

Rationale: If you want to predict traffic at a sunday morning, look at traffic at previous sunday mornings

Shortcomings:

1. Skip-length must be manually tuned
2. Skip model only works on periodic data
3. Attention layer looks at relevant hidden states—not relevant time series

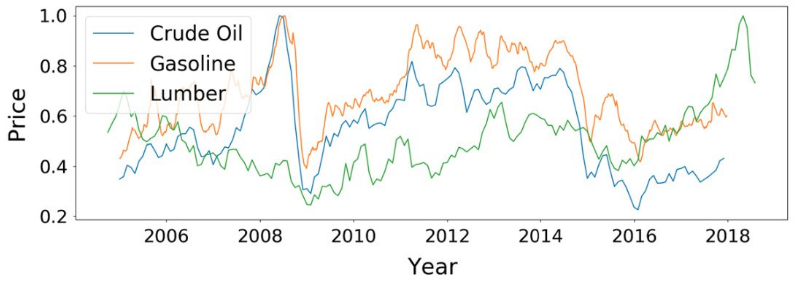



**Fig. 1** Historical prices of crude oil, gasoline, and lumber. Units are omitted and scales are normalized for simplicity

Cannot see that crude oil is more important than lumber for gasoline price!

# Methods

First calculate hidden states by an RNN (in practice an LSTM)

RNN

$$h_t = F(h_{t-1}, x_t)$$

Then apply 1-D convolution to the rows (time series) of the hidden states

Attend to the the relevant rows

Sum rows and perform matrix multiplication to get final state $h'_t$

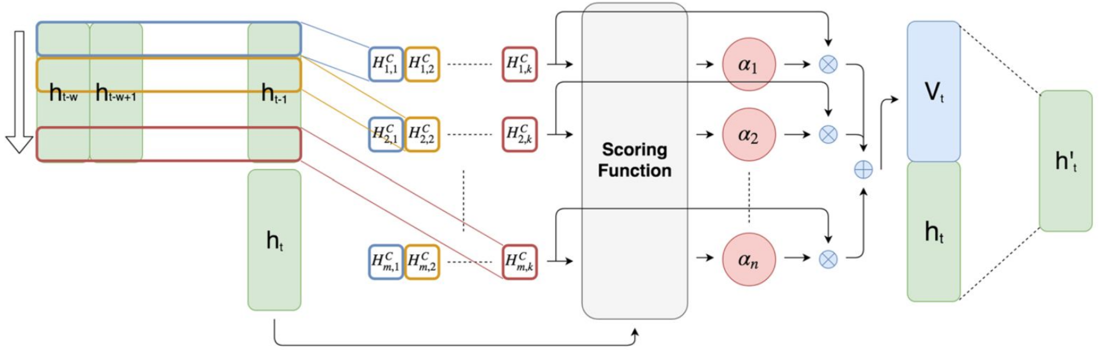



**Fig. 2** Proposed attention mechanism. $h_t$ represents the hidden state of the RNN at time step $t$. There are $k$ 1-D CNN filters with length $w$, shown as different colors of rectangles. Then, each filter convolves over $m$ features of hidden states and produces a matrix $H^C$ with $m$ rows and $k$ columns. Next, the scoring function calculates a weight for each row of $H^C$ by comparing with the current hidden state $h_t$. Last but not least, the weights are normalized and the rows of $H^C$ is weighted summed by their corresponding weights to generate $V_t$. Finally, we concatenate $V_t, h_t$ and perform matrix multiplication to generate $h'_t$, which is used to create the final forecast value (Color figure online)

# Experimental Setting and Results

The paper first presents an experiment on toy data then on real-world data



NTNU

# Toy examples

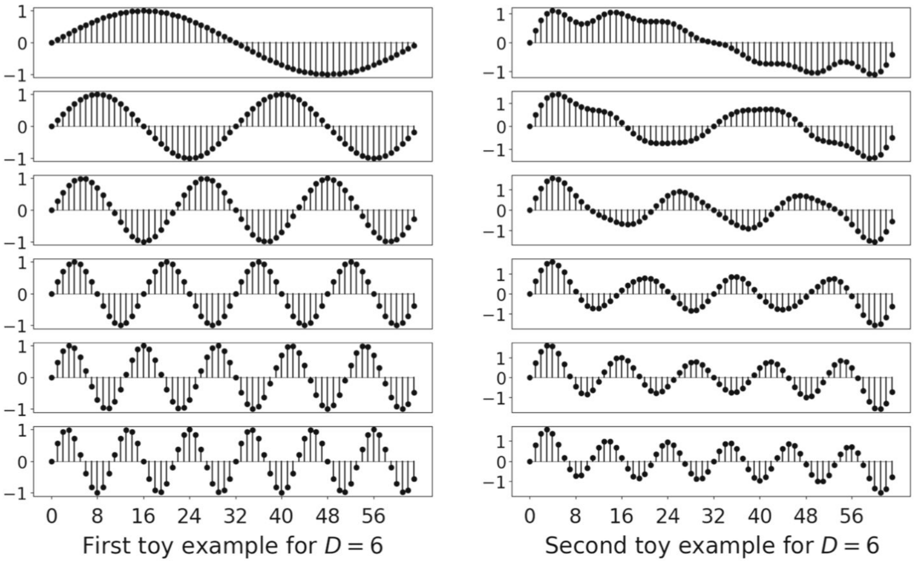



**Fig. 3** Visualization of the first type of toy examples without interdependencies (left) and the second type of toy examples with interdependencies (right) for $D = 6$, which means that there are 6 time series in each example

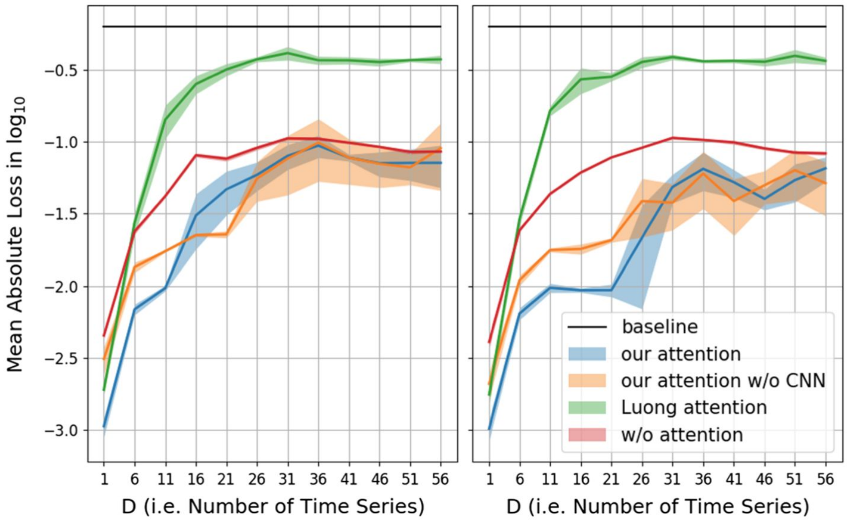



**Fig. 4** Mean absolute loss and the range of standard deviation in $\log_{10}$ of the first type of toy examples without interdependencies (left) and the second type of toy examples with interdependencies (right), both in ten runs. The baseline indicates the loss if all predicted values are zero



NTNU

# Different real-world datasets

| Dataset | $L$ | $D$ | $S$ | $B$ |
|---|---|---|---|---|
| Solar Energy | 52,560 | 137 | 10 minutes | 172 M |
| Traffic | 17,544 | 862 | 1 hour | 130 M |
| Electricity | 26,304 | 321 | 1 hour | 91 M |
| Exchange Rate | 7,588 | 8 | 1 day | 534 K |
| MuseData | 216–102,552 | 128 | 1 beat | 4.9 M |
| LPD-5-Cleansed | 1,072–1,917,952 | 128 | 1 beat | 1.7 G |

**Table 1** Statistics of all datasets, where $L$ is the length of the time series, $D$ is the number of time series, $S$ is the sampling spacing, and $B$ is size of the dataset in bytes. MuseData and LPD-5-Cleansed both have various-length time series since the length of music pieces varies.



NTNU

# Different real-world datasets

| Dataset | $L$ | $D$ | $S$ | $B$ |
|---|---|---|---|---|
| Solar Energy | 52,560 | 137 | 10 minutes | 172 M |
| Traffic | 17,544 | 862 | 1 hour | 130 M |
| Electricity | 26,304 | 321 | 1 hour | 91 M |
| Exchange Rate | 7,588 | 8 | 1 day | 534 K |
| MuseData | 216–102,552 | 128 | 1 beat | 4.9 M |
| LPD-5-Cleansed | 1,072–1,917,952 | 128 | 1 beat | 1.7 G |

**Table 1** Statistics of all datasets, where $L$ is the length of the time series, $D$ is the number of time series, $S$ is the sampling spacing, and $B$ is size of the dataset in bytes. MuseData and LPD-5-Cleansed both have various-length time series since the length of music pieces varies.



| RAE | Solar Energy | | | | Traffic | | | |
|---|---|---|---|---|---|---|---|---|
| Horizon | 3 | 6 | 12 | 24 | 3 | 6 | 12 | 24 |
| AR | 0.1846 | 0.3242 | 0.5637 | 0.9221 | 0.4491 | 0.4610 | 0.4700 | 0.4696 |
| LRidge | 0.1227 | 0.2098 | 0.4070 | 0.6977 | 0.4965 | 0.5115 | 0.5198 | 0.4846 |
| LSVR | 0.1082 | 0.2451 | 0.4362 | 0.6180 | 0.4629 | 0.5483 | 0.7454 | 0.4761 |
| GP | 0.1419 | 0.2189 | 0.4095 | 0.7599 | 0.5148 | 0.5759 | 0.5316 | 0.4829 |
| SETAR | 0.1285 | 0.1962 | 0.2611 | 0.3147 | 0.3226 | 0.3372 | 0.3368 | 0.3348 |
| LSTNet-Skip | 0.0985 | 0.1554 | 0.2018 | 0.3551 | 0.3287 | 0.3627 | 0.3518 | 0.3852 |
| LSTNet-Attn | **0.0900** | 0.1332 | 0.2202 | 0.4308 | 0.3196 | 0.3277 | 0.3557 | 0.3666 |
| Our model | 0.0918 ± 0.0005 | **0.1296 ± 0.0008** | **0.1902 ± 0.0021** | **0.2727 ± 0.0045** | **0.2901 ± 0.0095** | **0.2999 ± 0.0022** | **0.3112 ± 0.0015** | **0.3118 ± 0.0034** |

| RAE | Electricity | | | | Exchange Rate | | | |
|---|---|---|---|---|---|---|---|---|
| Horizon | 3 | 6 | 12 | 24 | 3 | 6 | 12 | 24 |
| AR | 0.0579 | 0.0598 | 0.0603 | 0.0611 | 0.0181 | 0.0224 | 0.0291 | 0.0378 |
| LRidge | 0.0900 | 0.0933 | 0.1268 | 0.0779 | 0.0144 | 0.0225 | 0.0358 | 0.0602 |
| LSVR | 0.0858 | 0.0816 | 0.0762 | 0.0690 | 0.0148 | 0.0231 | 0.0360 | 0.0576 |
| GP | 0.0907 | 0.1137 | 0.1043 | 0.0776 | 0.0230 | 0.0239 | 0.0355 | 0.0547 |
| SETAR | 0.0475 | 0.0524 | 0.0545 | 0.0565 | **0.0136** | 0.0199 | 0.0288 | 0.0425 |
| LSTNet-Skip | 0.0509 | 0.0587 | 0.0598 | 0.0561 | 0.0180 | 0.0226 | 0.0296 | 0.0378 |
| LSTNet-Attn | 0.0515 | 0.0543 | 0.0561 | 0.0579 | 0.0229 | 0.0269 | 0.0384 | 0.0517 |
| Our model | **0.0463 ± 0.0007** | **0.0491 ± 0.0007** | **0.0541 ± 0.0006** | **0.0544 ± 0.0007** | 0.0139 ± 0.0001 | **0.0192 ± 0.0002** | **0.0280 ± 0.0006** | **0.0372 ± 0.0005** |

| RSE | Solar Energy | | | | Traffic | | | |
|---|---|---|---|---|---|---|---|---|
| Horizon | 3 | 6 | 12 | 24 | 3 | 6 | 12 | 24 |
| AR | 0.2435 | 0.3790 | 0.5911 | 0.8699 | 0.5991 | 0.6218 | 0.6252 | 0.6293 |
| LRidge | 0.2019 | 0.2954 | 0.4832 | 0.7287 | 0.5833 | 0.5920 | 0.6148 | 0.6025 |
| LSVR | 0.2021 | 0.2999 | 0.4846 | 0.7300 | 0.5740 | 0.6580 | 0.7714 | 0.5909 |
| GP | 0.2259 | 0.3286 | 0.5200 | 0.7973 | 0.6082 | 0.6772 | 0.6406 | 0.5995 |
| SETAR | 0.2374 | 0.3381 | 0.4394 | 0.5271 | 0.4611 | 0.4805 | 0.4846 | 0.4898 |
| LSTNet-Skip | 0.1843 | 0.2559 | 0.3254 | 0.4643 | 0.4777 | 0.4893 | 0.4950 | 0.4973 |
| LSTNet-Attn | 0.1816 | 0.2538 | 0.3466 | 0.4403 | 0.4897 | 0.4973 | 0.5173 | 0.5300 |
| Our model | **0.1803 ± 0.0008** | **0.2347 ± 0.0017** | **0.3234 ± 0.0044** | **0.4389 ± 0.0084** | **0.4487 ± 0.0180** | **0.4658 ± 0.0053** | **0.4641 ± 0.0034** | **0.4765 ± 0.0068** |

| RSE | Electricity | | | | Exchange Rate | | | |
|---|---|---|---|---|---|---|---|---|
| Horizon | 3 | 6 | 12 | 24 | 3 | 6 | 12 | 24 |
| AR | 0.0995 | 0.1035 | 0.1050 | 0.1054 | 0.0228 | 0.0279 | 0.0353 | 0.0445 |
| LRidge | 0.1467 | 0.1419 | 0.2129 | 0.1280 | 0.0184 | 0.0274 | 0.0419 | 0.0675 |
| LSVR | 0.1523 | 0.1372 | 0.1333 | 0.1180 | 0.0189 | 0.0284 | 0.0425 | 0.0662 |
| GP | 0.1500 | 0.1907 | 0.1621 | 0.1273 | 0.0239 | 0.0272 | 0.0394 | 0.0580 |
| SETAR | 0.0901 | 0.1020 | 0.1048 | 0.1009 | 0.0178 | 0.0250 | 0.0352 | 0.0497 |
| LSTNet-Skip | 0.0864 | 0.0931 | 0.1007 | 0.1007 | 0.0226 | 0.0280 | 0.0356 | 0.0449 |
| LSTNet-Attn | 0.0868 | 0.0953 | 0.0984 | 0.1059 | 0.0276 | 0.0321 | 0.0448 | 0.0590 |
| Our model | **0.0823 ± 0.0012** | **0.0916 ± 0.0018** | **0.0964 ± 0.0015** | **0.1006 ± 0.0015** | **0.0174 ± 0.0001** | **0.0241 ± 0.0004** | **0.0341 ± 0.0011** | **0.0444 ± 0.0006** |

| CORR | Solar Energy | | | | Traffic | | | |
|---|---|---|---|---|---|---|---|---|
| Horizon | 3 | 6 | 12 | 24 | 3 | 6 | 12 | 24 |
| AR | 0.9710 | 0.9263 | 0.8107 | 0.5314 | 0.7752 | 0.7568 | 0.7544 | 0.7519 |
| LRidge | 0.9807 | 0.9568 | 0.8765 | 0.6803 | 0.8038 | 0.8051 | 0.7879 | 0.7862 |
| LSVR | 0.9807 | 0.9562 | 0.8764 | 0.6789 | 0.7993 | 0.7267 | 0.6711 | 0.7850 |
| GP | 0.9751 | 0.9448 | 0.8518 | 0.5971 | 0.7831 | 0.7406 | 0.7671 | 0.7909 |
| SETAR | 0.9744 | 0.9436 | 0.8974 | 0.8420 | 0.8641 | 0.8506 | 0.8465 | 0.8443 |
| LSTNet-Skip | 0.9843 | 0.9690 | 0.9467 | 0.8870 | 0.8721 | 0.8690 | 0.8614 | 0.8588 |
| LSTNet-Attn | 0.9848 | 0.9696 | 0.9397 | 0.8995 | 0.8704 | 0.8669 | 0.8540 | 0.8429 |
| Our model | **0.9850 ± 0.0001** | **0.9742 ± 0.0003** | **0.9487 ± 0.0023** | **0.9081 ± 0.0151** | **0.8812 ± 0.0089** | **0.8717 ± 0.0034** | **0.8717 ± 0.0021** | **0.8629 ± 0.0027** |

| CORR | Electricity | | | | Exchange Rate | | | |
|---|---|---|---|---|---|---|---|---|
| Horizon | 3 | 6 | 12 | 24 | 3 | 6 | 12 | 24 |
| AR | 0.8845 | 0.8632 | 0.8591 | 0.8595 | 0.9734 | 0.9656 | 0.9526 | 0.9357 |
| LRidge | 0.8890 | 0.8594 | 0.8003 | 0.8806 | 0.9788 | **0.9722** | 0.9543 | 0.9305 |
| LSVR | 0.8888 | 0.8861 | 0.8961 | 0.8891 | 0.9782 | 0.9697 | 0.9546 | 0.9370 |
| GP | 0.8670 | 0.8334 | 0.8394 | 0.8818 | 0.8713 | 0.8193 | 0.8484 | 0.8278 |
| SETAR | 0.9402 | 0.9294 | 0.9202 | **0.9171** | 0.9759 | 0.9675 | 0.9518 | 0.9314 |
| LSTNet-Skip | 0.9283 | 0.9135 | 0.9077 | 0.9119 | 0.9735 | 0.9658 | 0.9511 | 0.9354 |
| LSTNet-Attn | 0.9243 | 0.9095 | 0.9030 | 0.9025 | 0.9717 | 0.9656 | 0.9499 | 0.9339 |
| Our model | **0.9429 ± 0.0004** | **0.9337 ± 0.0011** | **0.9250 ± 0.0013** | 0.9133 ± 0.0008 | **0.9790 ± 0.0003** | 0.9709 ± 0.0003 | **0.9564 ± 0.0005** | **0.9381 ± 0.0008** |

**Table 2** Results on typical MTS datasets using RAE, RSE and CORR as metrics. Best performance in boldface; second best performance is underlined. We report the mean and standard deviation of our model in ten runs. All numbers besides the results of our model is referenced from the paper of LSTNet [Lai et al.(2018)Lai, Chang, Yang, and Liu].

# Different real-world datasets

| Dataset | $L$ | $D$ | $S$ | $B$ |
|---|---|---|---|---|
| Solar Energy | 52,560 | 137 | 10 minutes | 172 M |
| Traffic | 17,544 | 862 | 1 hour | 130 M |
| Electricity | 26,304 | 321 | 1 hour | 91 M |
| Exchange Rate | 7,588 | 8 | 1 day | 534 K |
| MuseData | 216–102,552 | 128 | 1 beat | 4.9 M |
| LPD-5-Cleansed | 1,072–1,917,952 | 128 | 1 beat | 1.7 G |

**Table 1** Statistics of all datasets, where $L$ is the length of the time series, $D$ is the number of time series, $S$ is the sampling spacing, and $B$ is size of the dataset in bytes. MuseData and LPD-5-Cleansed both have various-length time series since the length of music pieces varies.

| | MuseData | | |
|---|---|---|---|
| Metric | Precision | Recall | F1 |
| W/o attention | 0.84009 | 0.67657 | 0.74952 |
| W/ Luong attention | 0.75197 | 0.52839 | 0.62066 |
| W/ proposed attention | **0.85581** | **0.68889** | **0.76333** |

| | LPD-5-Cleansed | | |
|---|---|---|---|
| Metric | Precision | Recall | F1 |
| W/o attention | 0.83794 | 0.73041 | 0.78049 |
| W/ Luong attention | 0.83548 | 0.72380 | 0.77564 |
| W/ proposed attention | **0.83979** | **0.74517** | **0.78966** |

**Table 3** Precision, recall, and F1 score of different models on polyphonic music datasets.



NTNU

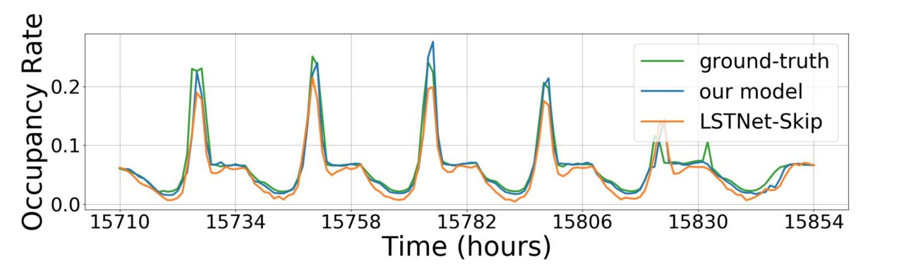
# CNN filters play the role of bases in DFT

Expect CNN filters to learn temporal MTS patterns

Authors calculated the average DFT of the 1-D convolutional filters after training on periodic dataset

Found that the prevailing frequency of the CNN filters were the same as the prevailing frequency in the dataset

This suggests that the CNN filters play the role of bases in DFT



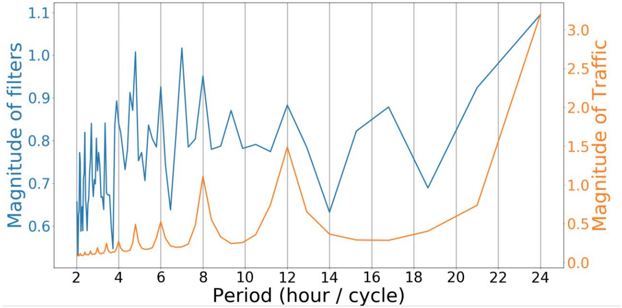




**Fig. 7** Magnitude comparison of (1) DFT of CNN filters trained on Traffic with a 3-hour horizon, and (2) every window of the Traffic dataset. To make the figure more intuitive, the unit of the horizontal axis is the period.

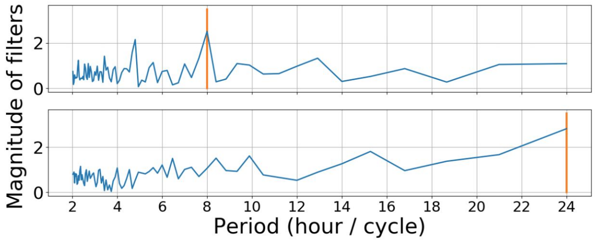



**Fig. 8** Two different CNN filters trained on Traffic with a 3-hour horizon, which detect different periods of temporal patterns.

# Conclusions

- The paper proposed a novel temporal pattern attention mechanism
- Experiments on toy examples and real-world datasets achieves state-of-the-art results
- Visualization of CNN filters verifies that they capture temporal information