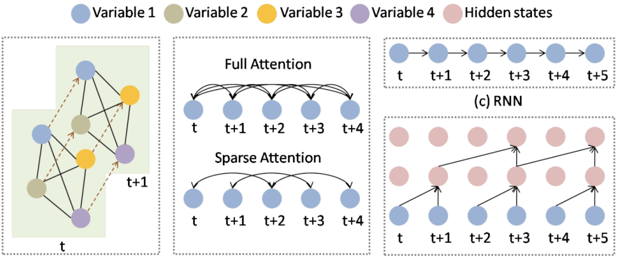
Operator learning for spatial time series
Fourier Neural Operator

Nikitina Maria

Foundation models for spatial-time series
Intelligent Systems Department, MIPT

October 16, 2025

Problem Statement

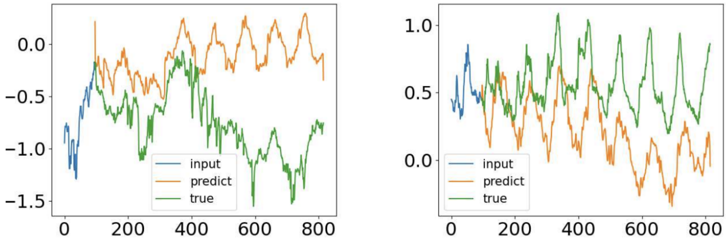


(a) GNN (b) Self-Attention (d) TCN

Common models have limited capabilities for capturing critical patterns for time series analysis. For example GNNs are constructed on variable-wise connections as illustrated in Fig. (a), and the sequential models (i.e., Transformer, RNN, and TCN) are based on timestamp-wise connections as shown in Fig. (b),(c), and (d), respectively.

Kun Yi. et.al. Neural Time Series Analysis with Fourier Transform: A Survey, 2023

Transformer problems



Different distribution between ground truth and forecasting output from vanilla Transformer in a real-world ETTm1 dataset. Left: frequency mode and trend shift. Right: trend shift. The discrepancy between ground truth and prediction could be explained by the point-wise attention and prediction in Transformer.

Tian Zhou et.al. FEDformer: Frequency Enhanced Decomposed Transformer for Long-term Series Forecasting, 2022

Operator learning definitions

Let the spatial domain $D \subset \mathbb{R}^d$ be a bounded, open set.

$\mathcal{A} = \mathcal{A}(D; \mathbb{R}^{d_a})$, $\mathcal{U} = \mathcal{U}(D; \mathbb{R}^{d_u})$ – separable Banach spaces of functions valued in $\mathbb{R}^{d_a}$ and $\mathbb{R}^{d_u}$ respectively.
We have observations $\{a_j, u_j\}_{j=1}^N$ where $a_j \sim \mu$ is an i.i.d. sequence from the probability measure μ on $\mathcal{A}$ and $u_j = G^\dagger(a_j)$ is possibly noisy.

$G^\dagger : \mathcal{A} \to \mathcal{U}$ – a (typically) non-linear map.

Operator learning task

Approximation

We approximate $G^\dagger$ by constructing a parametric map

$$G : \mathcal{A} \times \Theta \to \mathcal{U} \quad \text{or} \quad G_\theta : \mathcal{A} \to \mathcal{U},\ \theta \in \Theta$$

for finite-dimensional parameter space Θ by choosing $\theta^\dagger \in \Theta$ so that $G(\cdot, \theta^\dagger) = G_{\theta^\dagger} \approx G^\dagger$.

Cost Functional

This is a natural framework for learning in infinite-dimensions: define a cost functional $C : \mathcal{U} \times \mathcal{U} \to \mathbb{R}$ and seek

$$\min_{\theta \in \Theta} \mathbb{E}_{a \sim \mu} \left[C(G(a, \theta), G^\dagger(a)) \right],$$

Neural operator

Definition

The neural operator is an iterative architecture $v_0 \mapsto v_1 \mapsto \cdots \mapsto v_T$ where $v_j : D \to \mathbb{R}^{d_v}$ for $j = 0, \ldots, T-1$.

Iterative updates

The representation update $v_t \mapsto v_{t+1}$ is

$$v_{t+1}(x) := \sigma\left(Wv_t(x) + \left(\mathcal{K}(a;\phi)v_t\right)(x)\right), \quad \forall x \in D \tag{1}$$

where $W : \mathbb{R}^{d_v} \to \mathbb{R}^{d_v}$ is linear, and $\sigma : \mathbb{R} \to \mathbb{R}$ is a component-wise nonlinear activation. $\mathcal{K} : \mathcal{A} \times \Theta_K \to \mathcal{L}(\mathcal{U}(D;\mathbb{R}^{d_v}), \mathcal{U}(D;\mathbb{R}^{d_v}))$ maps to bounded linear operators parameterized by $\phi \in \Theta_K$:

$$\left(\mathcal{K}(a;\phi)v_t\right)(x) := \int_D \kappa\left(x, y, a(x), a(y); \phi\right) v_t(y)\, dy, \quad \forall x \in D \tag{2}$$

where $\kappa_\phi : \mathbb{R}^{2(d+d_a)} \to \mathbb{R}^{d_v \times d_v}$ is a neural network parameterized by $\phi \in \Theta_K$.

Fourier neural operator

Let $\mathcal{F}$ denote the Fourier transform of $f : D \to \mathbb{R}^{d_v}$ and $\mathcal{F}^{-1}$ its inverse:

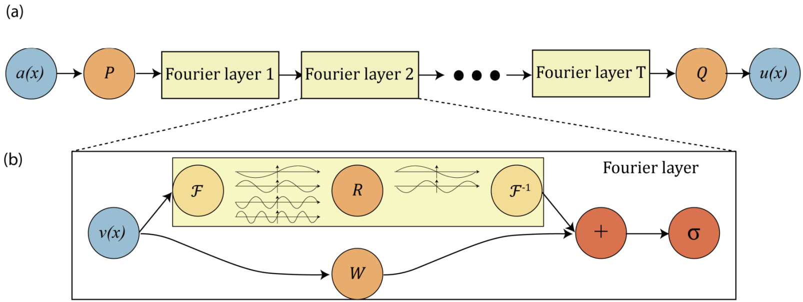
$$(\mathcal{F}f)_j(k) = \int_D f_j(x) e^{-2i\pi\langle x,k\rangle} dx, \quad (\mathcal{F}^{-1}f)_j(x) = \int_D f_j(k) e^{2i\pi\langle x,k\rangle} dk$$

Setting $\kappa_\phi(x, y, a(x), a(y)) = \kappa_\phi(x - y)$ in (2) and applying the convolution theorem yields:

$$(\mathcal{K}(a;\phi)v_t)(x) = \mathcal{F}^{-1}\left(\mathcal{F}(\kappa_\phi) \cdot \mathcal{F}(v_t)\right)(x), \quad \forall x \in D.$$

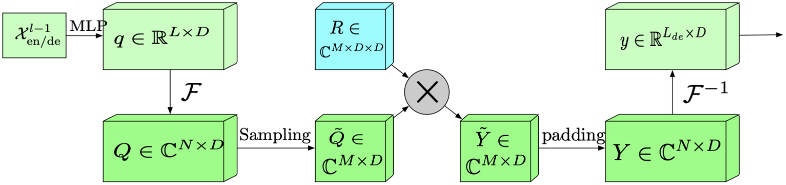
Zongyi Li et.al. Fourier neural operator for parametric partial differential equations, 2021

Architecture

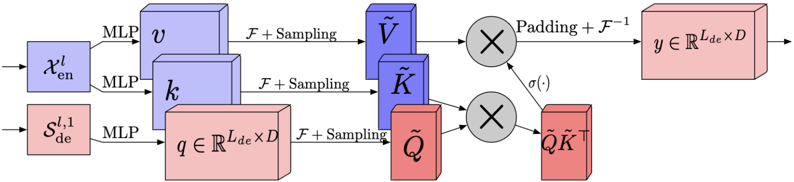


(a) The architecture of neural operator: 1. Lift to a higher dimension channel space by a neural network P. 2. Apply fourier layers. 3. Project back to the target dimension by a neural network Q.
(b) Fourier layers: Start from input v. On top: apply the Fourier transform F; a linear transform R on the lower Fourier modes and filters out the higher modes; then apply the inverse Fourier transform F^1. On the bottom: apply a local linear transform W.

Using with transformer



Frequency Enhanced Block with Fourier transform structure.



Frequency Enhanced Attention with Fourier transform structure

Conclusion

1. Operator learning is not restricted to PDEs and time series. Images can naturally be viewed as real-valued functions on 2-d domains and videos simply add a temporal structure.
2. The Fourier transform facilitates obtaining frequency spectrums that have abundant periodic information for time series analysis, e.g., seasonal patterns.
3. Frequency spectrums have a global view of time series that is helpful for capturing global characteristics of time series.
4. Another important merit of the Fourier transform is that it can be efficiently performed in the frequency domain, which saves plenty of computation costs in neural time series analysis.