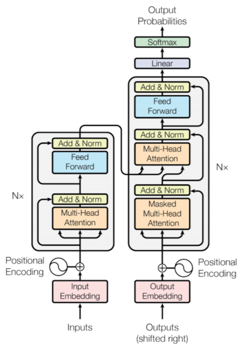
Predictive modeling in the reservoir kernel motif space

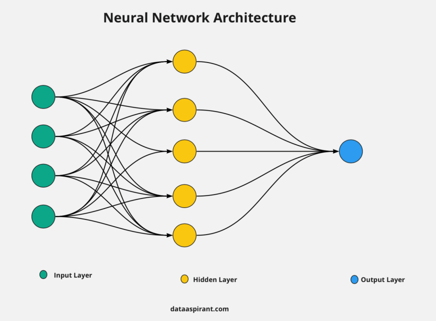
Peter Tino[†], Robert Simon Fong[†,‡], Roberto Fabio Leonarduzzi[‡]

[†] University of Birmingham; [‡]Theory Lab, Huawei

Overview

Two Paradigms of Time Series Forecasting

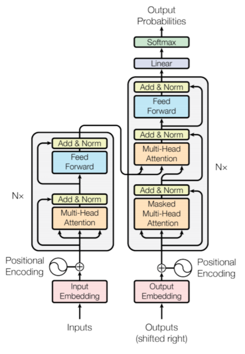




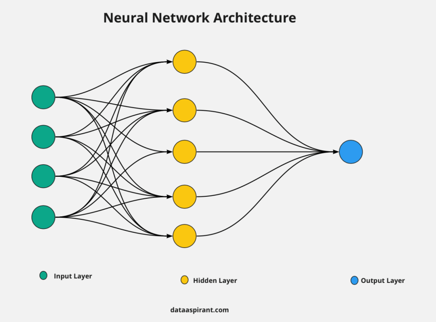
- “Trades time for space" – time series a *static input*
- Temporal correlation is disregarded due to the nature of the static input. [2]

- Captures *temporal dependencies* in the input data stream through *parametric state-space modelling*
- Sequentially encodes input time series in state space

Two Paradigms of Time Series Forecasting



- “Trades time for space" – time series a *static input*
- Temporal correlation is disregarded due to the nature of the static input. [2]



- Captures *temporal dependencies* in the input data stream through *parametric state-space modelling*
- Sequentially encodes input time series in state space

Brief overview

- Inspired by the kernel representation of linear Echo State Networks (ESN) [1]
- In linear ESN *Canonical dot product* of such feature representations $\Leftrightarrow$ *inner product* of the time series themselves (due to linearity)
- Representational structure of linear ESN is thus given by the eigenspace of the metric tensor corresponding to the inner product.
- In this work: use the positive eigenvectors (**motifs**) to build a *learnable* state space representation.

Linear Reservoir System

A **linear reservoir system** $R := (\mathbf{W}, \mathbf{w}, h)$ operating on *univariate* input has corresponding driven linear dynamical system:

$$\begin{cases} \mathbf{x}(t) = \mathbf{W}\mathbf{x}\,(t-1) + \mathbf{w}u(t) \\ \mathbf{y}(t) = h\,(\mathbf{x}\,(t))\,, \end{cases} \tag{1}$$

where:

- $\mathbf{W} \in \mathbb{R}^{N \times N}$ – dynamic coupling
- $\mathbf{w} \in \mathbb{R}^{N}$ – input couplings
- $h : \mathbb{R}^N \to \mathbb{R}^d$ – trainable readout map
- $\{\mathbf{x}(t)\}_t \subset \mathbb{R}^N$ – states
- $\{u(t)\}_t \subset \mathbb{R}$ – inputs
- $\{\mathbf{y}(t)\}_t \subset \mathbb{R}^d$ – outputs

State Space Representation

Given two sufficiently long time series of length $\tau > N$,

$$\begin{aligned}\mathbf{u} &= \left(u\left(-\tau+1\right), u\left(-\tau+2\right), \ldots, u\left(-1\right), u\left(0\right)\right) \\ &=: \left(u_1, u_2, \ldots, u_\tau\right) \in \mathbb{R}^\tau\end{aligned}$$

and

$$\begin{aligned}\mathbf{v} &= \left(v\left(-\tau+1\right), v\left(-\tau+2\right), \ldots, v\left(-1\right), v\left(0\right)\right) \\ &=: \left(v_1, v_2, \ldots, v_\tau\right) \in \mathbb{R}^\tau\end{aligned}$$

The feature space representation of $\mathbf{u}$ and $\mathbf{v}$ under $R := (\mathbf{W}, \mathbf{w}, h)$ with zero initial states is given by [1]:

$$\phi(\mathbf{u}) = \sum_{j=1}^{\tau} u_j \mathbf{W}^{\tau-j}\mathbf{w}, \quad \phi(\mathbf{v}) = \sum_{j=1}^{\tau} v_j \mathbf{W}^{\tau-j}\mathbf{w}.$$

Reservoir kernel – canonical dot product defined by:

$$K(\mathbf{u}, \mathbf{v}) = \langle \phi(\mathbf{u}), \phi(\mathbf{v}) \rangle$$

Let $\mathbf{Q} = [Q_{i,j}]$ denote the matrix corresponding to the inner product K:

$$\mathbf{u}^\top \mathbf{Q} \mathbf{v} := K(\mathbf{u}, \mathbf{v});$$
$$Q_{i,j} = \mathbf{w}^\top \left(\mathbf{W}^\top\right)^{i-1} \mathbf{W}^{j-1} \mathbf{w}.$$

Then since $\mathbf{Q}$ is symmetric positive semi-definite, it admits eigen-decomposition:

$$\mathbf{Q} = \mathbf{M} \Lambda_Q \mathbf{M}^\top.$$

Motifs – The $N_m := \text{rank}(Q) \leq N \leq \tau$ eigenvectors $\{\mathbf{m}_1, \ldots, \mathbf{m}_{N_m}\} \subset \mathbb{R}^\tau$ of $\mathbf{M}$ with *positive* (equiv. non-zero) eigenvalues are called the **motifs** of the linear system (1).

Reservoir Motifs II

Under the eigen-decomposition of $\mathbf{Q}$, we rewrite K as:

$$K(\mathbf{u}, \mathbf{v}) = \left(\Lambda_Q^{\frac{1}{2}} \mathbf{M}^\top \mathbf{u}\right)^\top \left(\Lambda_Q^{\frac{1}{2}} \mathbf{M}^\top \mathbf{v}\right) = \langle \tilde{\mathbf{u}}, \tilde{\mathbf{v}} \rangle,$$

where

$$\begin{aligned}
\varphi(\mathbf{u}) = \tilde{\mathbf{u}} = \Lambda_Q^{\frac{1}{2}} \mathbf{M}^\top \mathbf{u} \\
= \begin{bmatrix} \lambda_1^{\frac{1}{2}} \cdot \langle \mathbf{m}_1, \mathbf{u} \rangle \\ \vdots \\ \lambda_{N_m}^{\frac{1}{2}} \cdot \langle \mathbf{m}_{N_m}, \mathbf{u} \rangle \end{bmatrix} \\
= \left(\lambda_i^{\frac{1}{2}} \cdot \langle \mathbf{m}_i, \mathbf{u} \rangle\right)_{i=1}^{N_m} \in \mathbb{R}^{N_m}.
\end{aligned}$$

Reservoir Motifs III

Under the eigen-decomposition of $\mathbf{Q}$, we rewrite K as:

$$K(\mathbf{u},\mathbf{v}) = \left(\Lambda_Q^{\frac{1}{2}}\mathbf{M}^\top\mathbf{u}\right)^\top \left(\Lambda_Q^{\frac{1}{2}}\mathbf{M}^\top\mathbf{v}\right) = \langle\tilde{\mathbf{u}},\tilde{\mathbf{v}}\rangle,$$

where

$$\begin{aligned}
\varphi(\mathbf{u}) = \tilde{\mathbf{u}} &= \Lambda_Q^{\frac{1}{2}}\mathbf{M}^\top\mathbf{u} \\
&= \begin{bmatrix} \lambda_1^{\frac{1}{2}}\cdot\langle\mathbf{m}_1,\mathbf{u}\rangle \\ \vdots \\ \lambda_{N_m}^{\frac{1}{2}}\cdot\langle\mathbf{m}_{N_m},\mathbf{u}\rangle \end{bmatrix} \\
&= \left(\lambda_i^{\frac{1}{2}}\cdot\langle\mathbf{m}_i,\mathbf{u}\rangle\right)_{i=1}^{N_m} \in \mathbb{R}^{N_m}.
\end{aligned}$$

Reservoir Motif Machine

In the kernel view of Reservoir Computing the motif weights are fixed by Λ_Q, we propose to make them adaptable, denoted by the set of **motif coefficients**, $C := \{c_i \in \mathbb{R}\}_{i=1}^{N_m}$:

Reservoir Kernel:

$$\varphi(\mathbf{u}) = \begin{bmatrix} \lambda_1^{\frac{1}{2}} \cdot \langle \mathbf{m}_1, \mathbf{u} \rangle \\ \vdots \\ \lambda_{N_m}^{\frac{1}{2}} \cdot \langle \mathbf{m}_{N_m}, \mathbf{u} \rangle \end{bmatrix} = \left(\lambda_i^{\frac{1}{2}} \cdot \langle \mathbf{m}_i, \mathbf{u} \rangle \right)_{i=1}^{N_m} \in \mathbb{R}^{N_m}.$$

$\longrightarrow$

Reservoir Motif Machine

$$\varphi(\mathbf{u}; C) = \begin{bmatrix} c_1 \cdot \langle \mathbf{m}_1, \mathbf{u} \rangle \\ \vdots \\ c_{N_m} \cdot \langle \mathbf{m}_{N_m}, \mathbf{u} \rangle \end{bmatrix} = \left(c_i \cdot \langle \mathbf{m}_i, \mathbf{u} \rangle \right)_{i=1}^{N_m} \in \mathbb{R}^{N_m}.$$

RMM II: absorption of motif weights

Let $q : \mathbb{R}^{N_m} \to \mathbb{R}^d$ be a readout map of an RMM. We can defined $\tilde{q} : \mathbb{R}^{N_m} \to \mathbb{R}^d$ by:

$$\tilde{q}\left(x_1, \ldots, x_{N_m}\right) := q\left(c_1 \cdot x_1, \ldots, c_{N_m} \cdot x_{N_m}\right).$$

Therefore it suffices to consider $C = \mathbb{1}$ as a linear system:

$$\begin{aligned} \mathbf{y}(t) &= q(\varphi\left(\mathbf{z}\left(t, \tau\right); \mathbb{1}\right)) \\ &= q(\mathbf{M}^\top \mathbf{z}\left(t, \tau\right)). \end{aligned}$$

Geometric view of RMM

The feature space representation of **u** under reservoir system can be expressed as a linear operator:

$$\mathbf{A}\mathbf{u} := \phi(\mathbf{u}) = \sum_{j=1}^{\tau} u_j \mathbf{W}^{\tau-j}\mathbf{w} = \mathbf{A}\mathbf{u}.$$

Consider the SVD of $\mathbf{A} \in \mathbb{R}^{N\times\tau}$:

$$\mathbf{A}^\top \mathbf{A} = \mathbf{V}\mathbf{D}\mathbf{V}^\top,$$

Since $N < \tau$, the linear operator **A** *projects* the time series **u** onto the N singular vectors of non-zero singular values, denoted by $\{\mathbf{v}_1, \ldots, \mathbf{v}_N\}$ with weights $d_1^{\frac{1}{2}}, d_2^{\frac{1}{2}}, \ldots, d_N^{\frac{1}{2}}$.

Geometric view of RMM II

Notice $\mathbf{Q} = \mathbf{A}^\top \mathbf{A}$, hence the reservoir motifs are precisely the $N_m \leq N < \tau$ singular vectors of A with nonzero singular values.

The representation of time series by RMM, given by:

$$\varphi\left(\mathbf{u}\left(t,\tau\right);\mathbb{1}\right) = \mathbf{M}^\top \mathbf{u}\left(t,\tau\right)$$

are therefore *projections* of $\mathbf{u}\left(t,\tau\right)$ onto the feature space defined by the reservoir kernel. In other words, they are *not* an approximation of the reservoir state reached upon reading $\mathbf{u}$ up to time t.

Univariate Time Series Prediction

		Lin-RMM		Informer		LSTMa		ARIMA	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ECL	48	**0.155**	**0.301**	0.239	0.359	0.493	0.539	0.879	0.764
	168	**0.175**	**0.322**	0.447	0.503	0.723	0.655	1.032	0.833
	336	**0.166**	**0.314**	0.489	0.528	1.212	0.898	1.136	0.876
	720	**0.164**	**0.314**	0.540	0.571	1.511	0.966	1.251	0.933
	960	**0.162**	**0.312**	0.582	0.608	1.545	1.006	1.370	0.982
ETTh1	24	**0.029**	**0.127**	0.098	0.247	0.114	0.272	0.108	0.284
	48	**0.044**	**0.156**	0.158	0.319	0.193	0.358	0.175	0.424
	168	**0.079**	**0.211**	0.183	0.346	0.236	0.392	0.396	0.504
	336	**0.108**	**0.254**	0.222	0.387	0.590	0.698	0.468	0.593
	720	**0.189**	**0.353**	0.269	0.435	0.683	0.768	0.659	0.766
ETTh2	24	**0.058**	**0.180**	0.093	0.240	0.155	0.307	3.554	0.445
	48	**0.083**	**0.220**	0.155	0.314	0.190	0.348	3.190	0.474
	168	**0.146**	**0.298**	0.232	0.389	0.385	0.514	2.800	0.595
	336	**0.186**	**0.347**	0.263	0.417	0.558	0.606	2.753	0.738
	720	**0.275**	**0.427**	0.277	0.431	0.640	0.681	2.878	1.044
ETTm1	24	**0.010**	**0.073**	0.030	0.137	0.121	0.233	0.090	0.206
	48	**0.018**	**0.098**	0.069	0.203	0.305	0.411	0.179	0.306
	96	**0.028**	**0.124**	0.194	0.372	0.287	0.420	0.272	0.399
	288	**0.053**	**0.171**	0.401	0.554	0.524	0.584	0.462	0.558
	672	**0.079**	**0.209**	0.512	0.644	1.064	0.873	0.639	0.697
Weather	24	**0.091**	**0.208**	0.117	0.251	0.131	0.254	0.219	0.355
	48	**0.135**	**0.260**	0.178	0.318	0.190	0.334	0.273	0.409
	168	**0.222**	**0.345**	0.266	0.398	0.341	0.448	0.503	0.599
	336	**0.277**	**0.391**	0.297	0.416	0.456	0.554	0.728	0.730

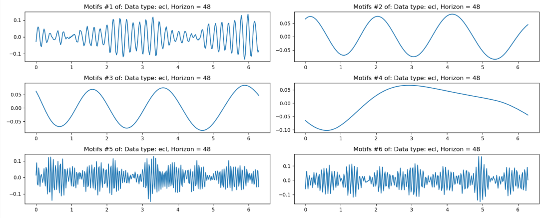
Our Lin-RMM model is compared with the Informer, LSTM, and ARIMA models based on their performance results reported in [3] through the mean square error (MSE) and mean absolute error (MAE).

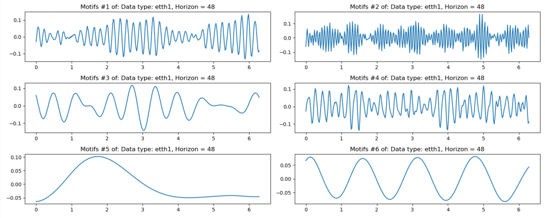
Multivariate Time Series Prediction

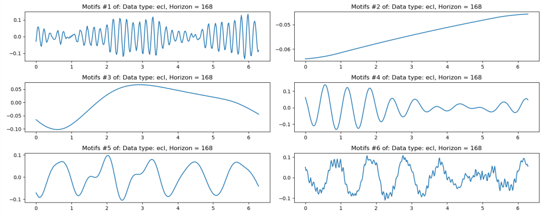
		Lin-RMM		f-Fedformer		w-Fedformer	
		MSE	MAE	MSE	MAE	MSE	MAE
ETTm2	96	**0.107**	**0.226**	0.203	0.287	0.204	0.288
	192	**0.140**	**0.263**	0.269	0.328	0.316	0.363
	336	**0.177**	**0.302**	0.325	0.366	0.359	0.387
	720	**0.223**	**0.349**	0.421	0.415	0.433	0.432
Exchange	96	0.874	0.680	0.148	0.278	**0.139**	**0.276**
	192	1.857	1.025	0.271	0.380	**0.256**	**0.369**
	336	2.819	1.306	0.460	0.500	**0.426**	**0.464**
	720	1.753	1.013	1.195	0.841	**1.090**	**0.800**
ILI	24	**1.549**	1.005	3.338	1.260	2.203	**0.963**
	36	**1.544**	1.003	2.678	1.080	2.272	**0.976**
	48	**1.279**	**0.885**	2.622	1.078	2.209	0.981
	60	**1.119**	**0.804**	2.857	1.157	2.545	1.061
Weather	96	2.677	0.876	**0.217**	**0.296**	0.227	0.304
	192	3.295	0.956	**0.276**	**0.336**	0.295	0.363
	336	2.926	0.939	**0.339**	**0.380**	0.381	0.416
	720	2.373	0.912	**0.403**	**0.428**	0.424	0.434

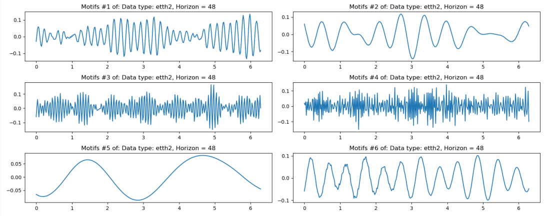
Our Lin-RMM model is compared with the Fedformer based on their performance results reported in [4] through the mean square error (MSE) and mean absolute error (MAE).

Motifs of Datasets









References I

[1] P. Tino.
Dynamical systems as temporal feature spaces.
J. Mach. Learn. Res., 21:44–1, 2020.

[2] A. Zeng, M. Chen, L. Zhang, and Q. Xu.
Are transformers effective for time series forecasting?
In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 11121–11128, 2023.

[3] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang.
Informer: Beyond efficient transformer for long sequence time-series forecasting.
In *AAAI Conference on Artificial Intelligence*, 2020.

[4] T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, and R. Jin.
Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting.
In *International Conference on Machine Learning*, pages 27268–27286. PMLR, 2022.