IEEE GlobalSIP 2019
Nov 11-14, 2019
7th IEEE Global Conference on Signal and Information Processing
Shaw Centre, Ottawa, Canada

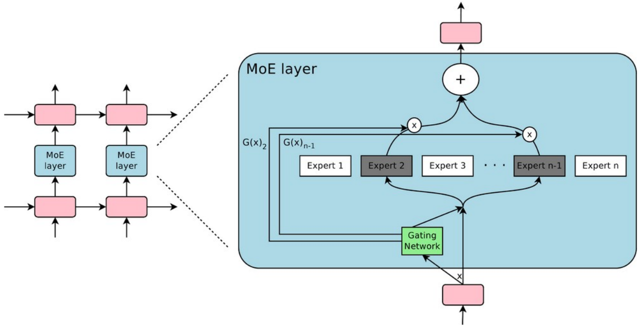
Stochastic Tucker-Decomposed Recurrent Neural Networks for Forecasting

Zachariah Carmichael* & Dhireesha Kudithipudi
Neuromorphic AI Lab
Dept. of Computer Engineering, Rochester Institute of Technology
Dept. of ECE, University of Texas - San Antonio

* Speaker

Background: Recurrent Neural Networks

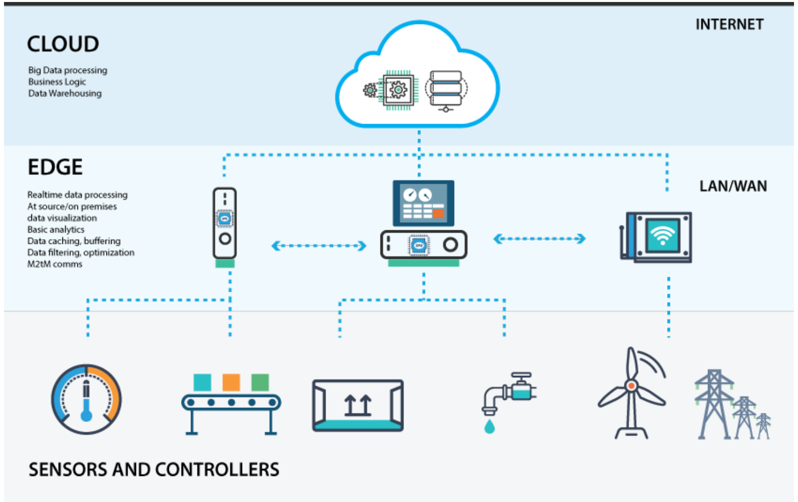
- Highly-parallel connectionist networks
- Learn a nonlinear mapping by minimizing an objective via gradient driven by data
- Successfully applied to various domains:
  - e.g. NLP, Genomics, Computer vision
- Demand copious amount of compute & memory resources



Mixture-of-Experts network with >137 billion parameters (~548 GB memory with 32-bit floats).

Shazeer, N. et al. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. arXiv preprint arXiv:1701.06538 (2017).

Motivation



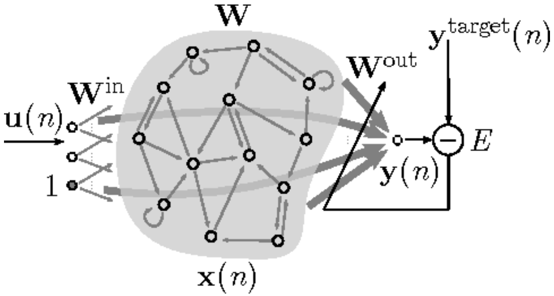
How can neural networks be tractably deployed on edge devices with constrained resources?

https://openautomationsoftware.com/blog/iiot-edge-computing-vs-cloud-computing/

Reservoir Computing

- Reservoir computing (RC)
  - Stochastic dynamics
  - Nonlinear responses

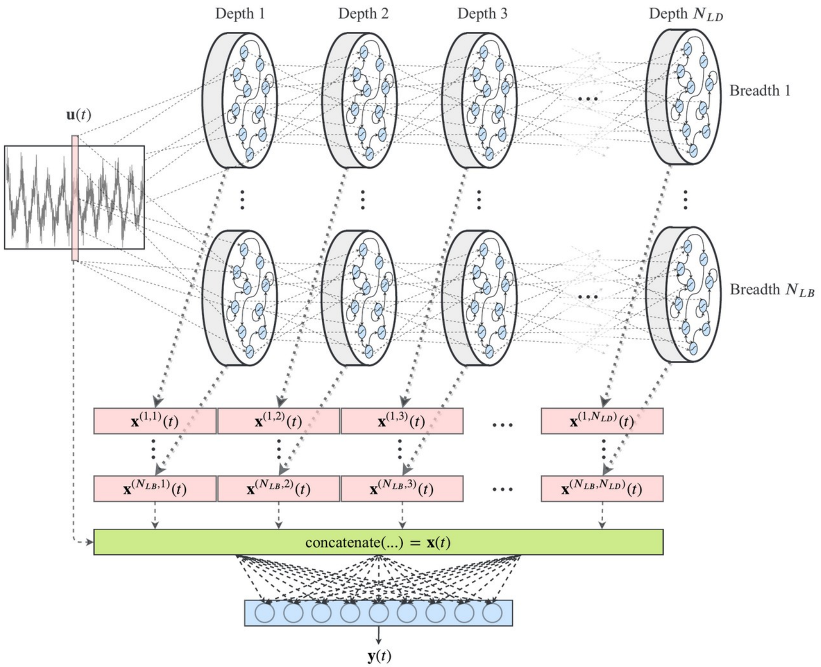
- Echo state networks (ESN)
  - Rate-base neurons
  - Dynamics & memory dictated by spectral radius
  - Teacher signal only trains readout weights



Jaeger, H. (2001). Short term memory in echo state networks. GMD Report 152, German National Research Center for Information Technology.
Lukosevicius, Mantas. “A Practical Guide to Applying Echo State Networks.” Neural Networks: Tricks of the Trade (2012).

Mod-DeepESN

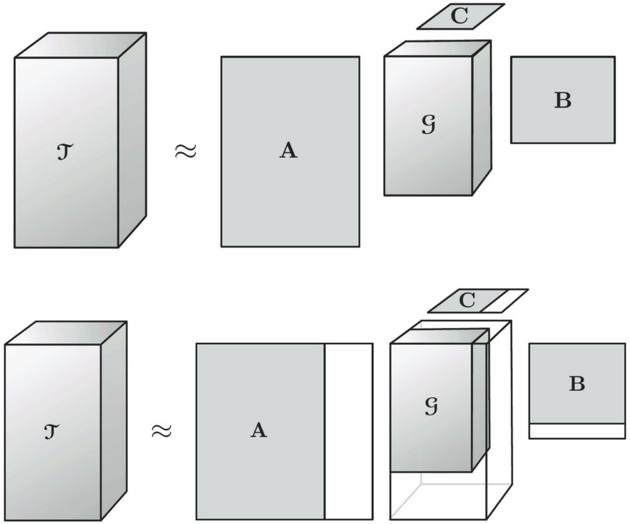
- Flexible topology for ESNs
- Elongated memory capacity
- Captures multi-scale dynamics of temporal data
- Standard components:
  - Randomly initialized weights
  - *Tanh* activations
  - Training using the pseudo-inverse (no backpropagation)
  - Training maps states matrix to a forecasted value


Depth 1
Depth 2
Depth 3
Depth N_{LD}
Breadth 1
Breadth N_{LB}
$\mathbf{u}(t)$
$\mathbf{x}^{(1,1)}(t)$
$\mathbf{x}^{(1,2)}(t)$
$\mathbf{x}^{(1,3)}(t)$
$\mathbf{x}^{(1,N_{LD})}(t)$
$\mathbf{x}^{(N_{LB},1)}(t)$
$\mathbf{x}^{(N_{LB},2)}(t)$
$\mathbf{x}^{(N_{LB},3)}(t)$
$\mathbf{x}^{(N_{LB},N_{LD})}(t)$
concatenate(...) $= \mathbf{x}(t)$
$\mathbf{y}(t)$

Z. Carmichael, H. Syed, and D. Kudithipudi, “Analysis of Wide and Deep Echo State Networks for Multiscale Spatiotemporal Time Series Forecasting,” in Proceedings of the 7th Annual Neuro-inspired Computational Elements Workshop, ser. NICE ’19. New York, NY, USA: ACM, 2019, pp. 7:1–7:10.

Tucker Decomposition

- Tucker: generalization of SVD to N-way tensors
- Factor matrices are orthogonal
- Rank reduction (tensor compression) possible by discarding Eigenvectors of each mode
- Not all modes require decomposition



$$\mathfrak{T} \approx \mathfrak{G} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} = \sum_{r_1=1}^{R_1} \sum_{r_2=1}^{R_2} \sum_{r_3=1}^{R_3} g_{r_1,r_2,r_3} \mathbf{a}_{r_1} \circ \mathbf{b}_{r_2} \circ \mathbf{c}_{r_3}$$

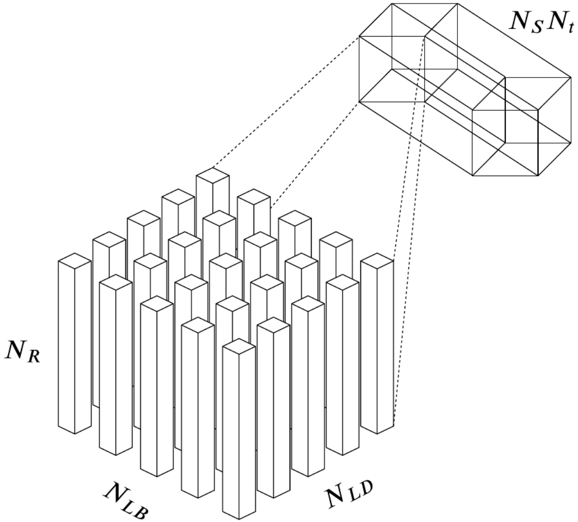
Kolda, Tamara G., and Brett W. Bader. "Tensor decompositions and applications." SIAM review 51.3 (2009): 455-500.

Tucker-Decomposed *Mod-DeepESN*

- Decomposition and compression of reservoir states as a tensor

- Orientations:
    - $N_S N_t$ x $N_L N_R$
    - $N_S N_t$ x N_L x N_R
    - $N_S N_t$ x N_{LB} x N_{LD} x N_R (shown on right)
    - $N_S N_t$ x N_{LB} x $N_{LD} N_R$
    - $N_S N_t$ x N_{LD} x $N_{LB} N_R$

- Compress along the final mode



N. Halko, P. Martinsson, and J. A. Tropp, “Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions,” SIAM Review, vol. 53, no. 2, pp. 217–288, 2011.

Training Comparison

Conventional

- SVD pseudo-inverse
- Explicit solution unstable with full states matrix for real-valued forecasting tasks

$$\mathbf{W}_{out} = \left(\mathbf{V}\frac{\mathbf{\Sigma}}{\mathbf{\Sigma}\odot\mathbf{\Sigma}+\beta\mathbb{I}}\mathbf{U}^\intercal\right)\mathbf{Y}$$

Compressed (Proposed)

- Explicit pseudo-inverse
- Overhead: SVD in HOOI algorithm
  - Replace with probabilistic algorithm - efficient for truncated SVD

$$\begin{aligned}\mathcal{G}, \mathbf{A}^{(1)}, \ldots, \mathbf{A}^{(N)} &= \mathrm{HOOI}(\mathcal{X}, R_1 \ldots R_N)\\ \mathbf{G} &= \textsc{Reshape}\left(\mathcal{G}, N_S N_t \times N_L N_R\right)\\ \mathbf{W}_{out} &= \left(\mathbf{G}^\intercal\mathbf{G}+\beta\mathbb{I}\right)^{-1}\mathbf{G}^\intercal\mathbf{Y}\end{aligned}$$

Training Comparison: Complexity

Conventional

- Complexity bottleneck: $(N_S N_t)^2$
  - Size of data squared
- Low complexity for small number of samples

$$\mathcal{O}[(N_L N_R)^2(1+N_S N_t)+(N_S N_t)^2 N_L N_R+N_S N_t N_L N_R N_Y].$$

Compressed (Proposed)

- Complexity bottleneck: k^3
  - Truncation size cubed
- Generally lower complexity here

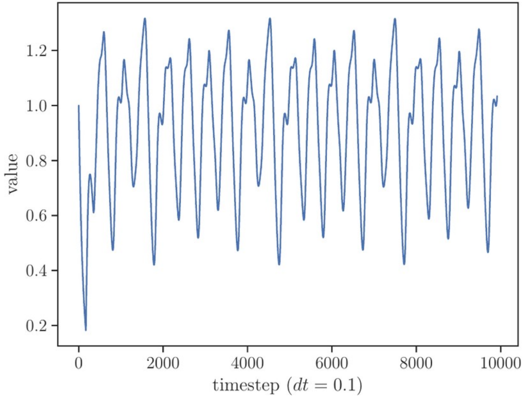
$$\mathcal{O}[k^3 + k^2(N_S N_t + N_L N_R) + k N_S N_t (N_L N_R + N_Y)]$$

Evaluation

$$\mathrm{RMSE} = \sqrt{\frac{1}{N_S N_t} \sum_{i=1}^{N_S} \sum_{t=1}^{N_t} (\mathbf{y}(t) - \hat{\mathbf{y}}(t))^2}$$

- Multi-scale nonlinear time series
    - Akin to sensory data observed by edge devices
- Compare states orientations and Tucker parameters
- Relative error to original network (average over 10 runs)
- Measurement of training FLOPs
    - Solely the FLOPs for training readout, no reservoir computations considered

Mackey Glass



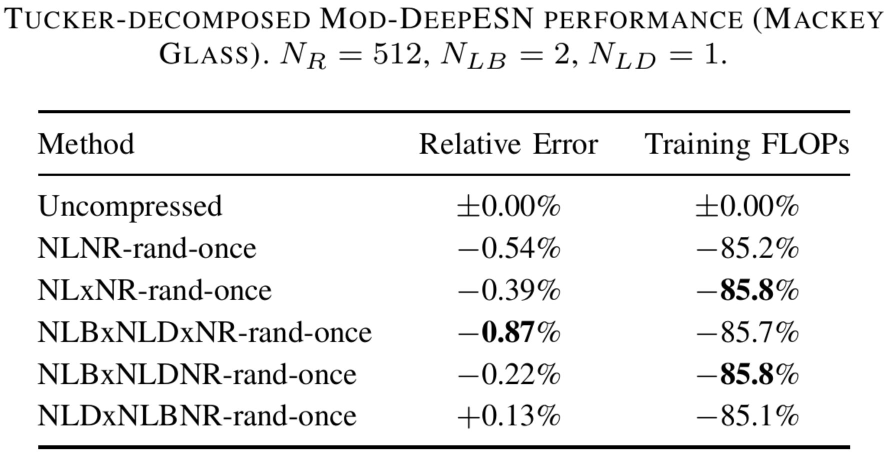
$$\frac{dx}{dt} = \beta \frac{x(t-\tau)}{1 + x(t-\tau)^n} - \gamma x(t)$$

- Classical chaotic time series benchmark for forecasting of dynamical systems

- Nonlinear differential equation generated using 4th order Runge-Kutta method

- 84-step-ahead forecasting, 10,000 samples

Mackey Glass Results

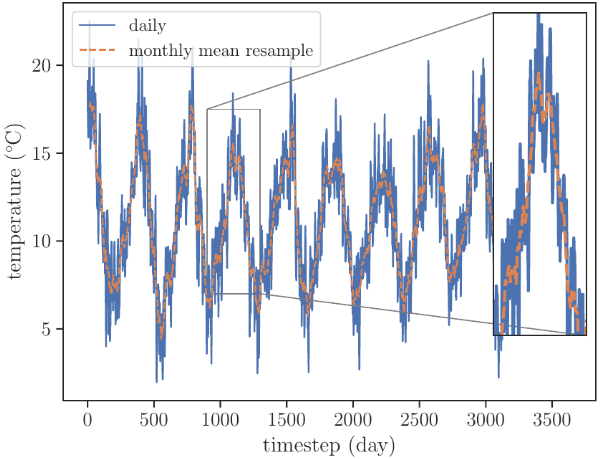


Tucker-decomposed Mod-DeepESN performance (Mackey Glass). $N_R = 512$, $N_{LB} = 2$, $N_{LD} = 1$.

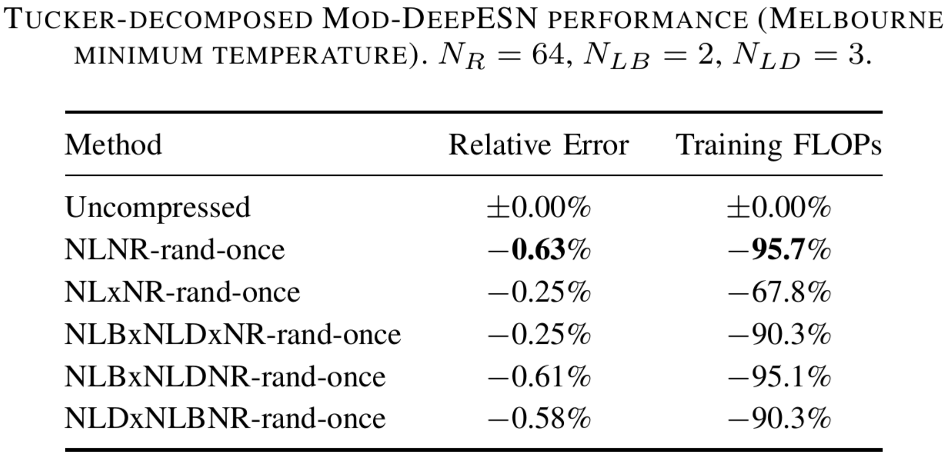
Method	Relative Error	Training FLOPs
Uncompressed	±0.00%	±0.00%
NLNR-rand-once	−0.54%	−85.2%
NLxNR-rand-once	−0.39%	−**85.8**%
NLBxNLDxNR-rand-once	−**0.87**%	−85.7%
NLBxNLDNR-rand-once	−0.22%	−**85.8**%
NLDxNLBNR-rand-once	+0.13%	−85.1%

Melbourne

- Minimum daily temperature series of Melbourne, Australia, 1981 - 1990
- 1-step-ahead forecasting task
- ~3,600 samples (days)



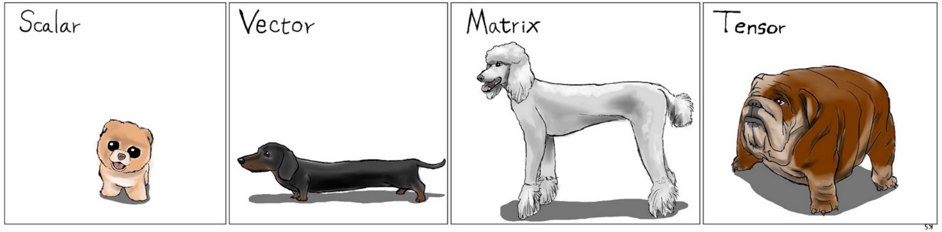
Melbourne Results



Tucker-decomposed Mod-DeepESN performance (Melbourne minimum temperature). $N_R = 64$, $N_{LB} = 2$, $N_{LD} = 3$.

Method	Relative Error	Training FLOPs
Uncompressed	$\pm 0.00\%$	$\pm 0.00\%$
NLNR-rand-once	$-\mathbf{0.63}\%$	$-\mathbf{95.7}\%$
NLxNR-rand-once	-0.25%	-67.8%
NLBxNLDxNR-rand-once	-0.25%	-90.3%
NLBxNLDNR-rand-once	-0.61%	-95.1%
NLDxNLBNR-rand-once	-0.58%	-90.3%

Conclusions



karlstratos.com/drawings/drawings.html

- Tensorization can improve forecasting performance
- Training complexity greatly reduced (~95%)
- Indicates these ideas are suitable for edge deployment
- Future work:
  - Extend to other weight matrices of *Mod-DeepESN*
  - Evaluate with more complex tasks
  - Tensor regression
  - Other decompositions: CP, tensor-train, Tucker2
  - Compare directly with deep learning counterparts

Acknowledgements

- Prof. Panos P. Markopoulos: detailed technical discussions on Tucker decomposition
- Air Force Research Laboratory: partial funding support under agreement #FA8750-16-1-0108

Questions?

Contact: zjc2920@rit.edu